SCIENCE CHINA
Technological Sciences

• Article •

March 2020 Vol. 63 No. 3: 506–519
https://doi.org/10.1007/s11431-019-9511-7

Multi-revolution low-thrust trajectory optimization using symplectic methods

E ZhiBo[1*] & GUZZETTI Davide[2]

[1] *School of Aerospace Engineering, Tsinghua University, Beijing 100084, China;*
[2] *Department of Aerospace Engineering, Auburn University, Auburn AL 36849, USA*

Received January 22, 2019; accepted April 17, 2019; published online December 31, 2019

Optimization of low-thrust trajectories that involve a larger number of orbit revolutions is considered as a challenging problem. This paper describes a high-precision symplectic method and optimization techniques to solve the minimum-energy low-thrust multi-revolution orbit transfer problem. First, the optimal orbit transfer problem is posed as a constrained nonlinear optimal control problem. Then, the constrained nonlinear optimal control problem is converted into an equivalent linear quadratic form near a reference solution. The reference solution is updated iteratively by solving a sequence of linear-quadratic optimal control sub-problems, until convergence. Each sub-problem is solved via a symplectic method in discrete form. To facilitate the convergence of the algorithm, the spacecraft dynamics are expressed via modified equinoctial elements. Interpolating the non-singular equinoctial orbital elements and the spacecraft mass between the initial point and end point is proven beneficial to accelerate the convergence process. Numerical examples reveal that the proposed method displays high accuracy and efficiency.

low-thrust, trajectory optimization, symplectic method, multi-revolution transfers

Citation: E Z B, Guzzetti D. Multi-revolution low-thrust trajectory optimization using symplectic methods. Sci China Tech Sci, 2020, 63: 506–519, https://doi.org/10.1007/s11431-019-9511-7

1 Introduction

Low-thrust electric propulsion systems have attracted a significant amount of research interest in recent years, owing to a specific impulse higher than traditional chemical propulsion. Thus, low-thrust electric propulsion typically consumes less fuel mass and, as a result, it is an important option for interplanetary missions. Successful utilization of low-thrust electric propulsion in interplanetary missions includes Deep Space 1 [1], Dawn [2], Hayabusa [3], etc. Unfortunately, the application of low thrust usually results in long-duration orbit transfers, which may involve hundreds or even thousands of orbit revolutions. Due to such characteristic geometry, optimizing low-thrust multi-revolution transfers has been considered as a challenging problem since several decades ago, and the identification of high-performance transfer optimization frameworks is still an ongoing process. To alleviate the computational effort, developing high-precision and efficient algorithms to optimize transfers with a large number of orbit revolutions is considered to be of great significance.

Numerous computational methods for solving low-thrust optimal trajectories have been proposed in the literature, and they can be generally categorized as direct methods [4,5] and indirect methods [6,7]. Frameworks that combine direct and indirect methods are usually termed hybrid methods [8,9]. In an indirect method, by using the Pontryagins maximum principle or variation principle, the original optimization problem is transformed into a nonlinear two-point boundary value problem (TPBVP) which is generally solved via shooting methods. The solution from indirect methods is at least locally optimal, since the first-order necessary conditions for optimality are satisfied. However, it is generally difficult for

*Corresponding author (email: ezb16@mails.tsinghua.edu.cn)

indirect methods to converge to an optimal solution, since the convergence radius of the corresponding TPBVP is small. In addition, such a TPBVP is sensitive to the initial guess for the costate variables, which do not have any intuitive physical meaning. Some effective techniques to overcome the convergence challenge of indirect methods include homotopic transformation [10–12], switching detection [13] and costate variables estimation [14, 15]. For orbit transfer problems with few revolutions, these techniques are proven to be highly efficient. However, when a larger number of orbit revolutions are required for the transfer, indirect methods may struggle in finding convergent solutions. Compared to a TPBVP formulation approach, direct methods transcribe the optimal control problem into a nonlinear programming problem, which generally exhibits a larger convergence domain at the price of increased computational workload. For instance, Betts [16] uses the direct collocation method paired with sequential quadratic programing to solve a 578-revolution transfer problem, and presents an optimization problem with 416123 variables and 249674 constraints. Scheel and Conway [17] discuss a Runge-Kutta parallel-shooting method for solving a 100-revolution orbit transfer. Solving such large scale optimization problems requires a tremendous computational effort, which put forward higher demand for computational resources. In addition, the solutions that are obtained from direct methods do not generally satisfy the first-order necessary conditions for optimality. Therefore, the converged solutions are not ensured to be locally optimal. In recent years, utilizing convex optimization to solve low-thrust orbit transfer problems has attracted a significant amount of research interest [18, 19], since it is more computationally tractable compared to nonlinear programming [20]. It is proven that, convex optimization is highly efficient for solving short-duration trajectory optimization [21]. Nevertheless, when it comes to long-duration missions with multiple revolutions, there are no significant advantages from convex optimization. Hybrid methods exhibit both indirect and direct method good properties. The thrust profile is usually assumed a priori, and the optimal control is determined through the optimality conditions that define indirect methods. However, the thrust profile for orbit transfers with multi-revolutions is difficult to be guessed, and therefore, it is difficult to find optimal multi-revolutions solutions with hybrid methods.

Adding to the numerical methods mentioned above, symplectic methods exhibit promising performance in optimal control problems [22], owing to the preservation of the symplectic structure of the original problem [23]. The symplectic method first convert the nonlinear optimal control problem into a TPBVP using Hamiltonian formulation. Then, based on the dual variational principle, a symplectic form is applied to discretize the TPBVP. After discretization, the optimization problem is described by a set of nonlinear algebraic equations with sparse and symmetric coefficient matrices. Accordingly, solving such type of algebraic equations requires less computational resources. Since the symplectic method is based on the variational principle, it satisfies the first necessary conditions for optimality, which means that the solutions are at least locally optimal. Furthermore, owing to the preservation of the symplectic structure, the symplectic method can yield a reasonable approximation of the continuous solution with fewer discretization points. Peng et al. present a series of symplectic algorithms and utilize them to solve optimal orbit rendezvous problems [24], orbit transfer problems between halo orbits [25], optimal nonlinear feedback control for spacecraft rendezvous between libration point orbits [26], bound evaluation for spacecraft swarm reconfiguration on libration point orbits [27]. Li et al. [28] introduce the symplectic algorithm with quasi-linearization techniques to solve nonlinear optimal control problems with inequality path constrains, and prove its efficiency for designing spacecraft rendezvous between halo orbits. However, symplectic methods that are presented in existing studies only utilize orbit transfers with one revolution as supporting examples. In addition, the spacecraft mass variation is not taken into consideration in those studies, and should be considered in further research.

The convergence of indirect methods depend on the initial guess for the costates. Compared to indirect methods, the convergence of the symplectic methods mainly depends on the initial guess for the states. Compared to direct methods, symplectic methods require less computational resources, because the final problem formulation incorporates sparse and symmetric coefficient matrices. Consequently, symplectic methods may have large potential for solving optimal control problem with long-duration and multiple revolutions. However, to the authors best knowledge, no literature has explored the utilization of symplectic algorithms to solve low-thrust orbit transfer problems with many revolutions. That is mainly because, multi-revolution orbit transfers result in oscillation of the state variables through time, which makes difficult for symplectic methods to find convergent solutions. Another reason lies on the fact that, the supporting examples in previous references arbitrarily set the initial guess for the states variables to zero or to a constant value, without providing any reference trajectory. It is proven that, immediately supplying proper reference trajectories can accelerate the convergence of the optimal control problem

This paper investigates the application of symplectic algorithms to solve multi-revolution transfer problems as an alternative to indirect and direct methods. First, the opti-

mal orbit transfer problem is posed as a constrained nonlinear optimal control problem. Then, the constrained nonlinear optimal control problem is converted into an equivalent linear quadratic form near a reference solution. The reference solution is updated iteratively by solving a sequence of linear quadratic optimal control sub-problems, until convergence. An efficient symplectic algorithm is developed to solve each sub-problem. We also explore problem formulations that facilitate convergence to the optimal solution. That includes application of modified equinoctial elements to describe the spacecraft trajectory and bootstrapping the symplectic method with a linear interpolation of the state variables. Compared to Cartesian coordinates, modified equinoctial elements display smaller value oscillations along the final trajectory and may facilitate convergence to the optimal solution. Interpolating the non-singular equinoctial orbital elements and the spacecraft mass between the initial point and end point is also proven beneficial to accelerate the convergence process.

This paper is organized as follows. In sect. 2, spacecraft dynamics are expressed in modified equinoctial elements, and the model for low-thrust orbit transfer problem is built. The quasi-linearization method is utilized to transcribe the original nonlinear optimal control problem into a sequence of constraint linear-quadratic optimal control sub-problems. In sect. 3, a symplectic method is introduced to iteratively solve the sequence of constrained linear-quadratic optimal control sub-problems. To validate the accuracy and efficiency of the symplectic method, three examples of multi-revolution orbit transfer problems are given in sect. 4. Concluding remarks are made in sect. 5.

2 Low-thrust orbit transfer optimal control problem

Consider a transfer problem where the spacecraft is subjected only to gravity of the central body and the thrust of its own electric propulsion system. The objective is to determine the minimum-energy trajectory and thrust vector that transfer the spacecraft from the specified initial states to the specified terminal states. The low-thrust orbit transfer optimal control problem is described next.

2.1 Equations of motion

The state vector consists of the spacecraft position and velocity vectors, which are generally expressed in Cartesian coordinates. However, for low-thrust transfers with a large number of orbit revolutions, Cartesian coordinate values may display strong natural oscillations along the trajectory, which hinder the convergence to an optimal solution. In order to get better convergence performance, this work employs modified equinoctial elements $\boldsymbol{x} = [p, f, g, h, k, L]$ to describe the motion of the spacecraft, where p is the semi-latus rectum of the orbit, and L is the true longitude; the remaining four elements do not have any intuitive physical meaning; however, f together with g can describe the eccentricity of the orbit, and h together with k can describe the inclination of the orbit. Compared to Keplerian orbital elements or Cartesian coordinates, the equinoctial elements are non-singular for most eccentricities and inclinations, except for absolutely retrograde orbit. In addition, equinoctial elements conveniently describe the time variation of the true longitude, which acts as a phase angle. Most important for this work, when the equinoctial elements are chosen to describe the spacecraft motion with multiple revolutions, the natural oscillations of the state variable value can be reduced, and the optimal control problem is easier to solve. The equinoctial elements can be obtained from the Keplerian elements as

$$\begin{cases} x_1 = p = a(1-e^2), \\ x_2 = f = e\cos(\Omega+\omega), \\ x_3 = g = e\sin(\Omega+\omega), \\ x_4 = h = \tan(i/2)\cos\Omega, \\ x_5 = k = \tan(i/2)\sin\Omega, \\ x_6 = L = \Omega+\omega+f, \end{cases} \tag{1}$$

where a is the semi-major axis, e is the eccentricity of the orbit, i is the inclination of the orbit, Ω is the longitude of the ascending node, ω is the argument of perigee, and f is the true anomaly. We express the three-dimensional control vector in local vertical/local horizontal (LVLH) coordinates, which are attached to the spacecraft. Then, the spacecraft dynamics can be formulated as follows:

$$\dot{\boldsymbol{x}} = \boldsymbol{M}\left(\frac{T_{\max}}{m}\boldsymbol{u} + \boldsymbol{f}_p\right) + \boldsymbol{D}, \quad \dot{m} = -\frac{T_{\max}}{I_{\mathrm{sp}}g_0}\|\boldsymbol{u}\|, \tag{2}$$

where $\boldsymbol{M}$ is a 6×3 transformation matrix from the LVLH to the equinoctial elements and $\boldsymbol{D}$ is the six-dimensional gravity vector. The expressions of the matrix $\boldsymbol{M}$ and the vector $\boldsymbol{D}$ are as follows:

$$\boldsymbol{M} = \begin{Bmatrix} 0 & \frac{2x_1H}{W} & 0 & 0\,0\,0 \\ H\sin x_6 & \frac{H}{W}[(W+1)\cos x_6 + x_2] & -\frac{HG}{W}x_3 & 0\,0\,0 \\ -H\cos x_6 & \frac{H}{W}[(W+1)\sin x_6 + x_3] & \frac{HG}{W}x_2 & 0\,0\,0 \\ 0 & 0 & \frac{HS}{2W}\cos x_6 & 0\,0\,0 \\ 0 & 0 & \frac{HS}{2W}\sin x_6 & 0\,0\,0 \\ 0 & 0 & \frac{HG}{W} & 0\,0\,0 \end{Bmatrix}, \tag{3}$$

$$\boldsymbol{D} = \left[0, 0, 0, 0, 0, \frac{W^2}{H^3\mu}\right], \tag{4}$$

where μ is the gravitational constant, and the coefficients H, W, S, G are expressed as follows:

$$\begin{aligned} H &= \sqrt{\frac{x_1}{\mu}}, \\ W &= 1 + x_2\cos x_6 + x_3 \sin x_6, \\ S &= 1 + x_4^2 + x_5^2, \\ G &= x_4 \sin x_6 - x_5 \cos x_6. \end{aligned} \tag{5}$$

$T_{\max}$ is the maximum thrust magnitude, m is the instantaneous mass of the spacecraft, g_0 is the standard gravitational acceleration at sea level, and I_{sp} is the specific impulse of the thruster. The control vector is expressed by a three-dimensional vector $\boldsymbol{u}$, with norm between 0 and 1. The symbol $\boldsymbol{f}_{\mathrm{p}}$ represents the perturbation vector. In the central body reference frame, only the central body gravitational force is taken into consideration, and $\boldsymbol{f}_{\mathrm{p}}$ equals $\mathbf{0}$. In the vicinity of Earth, the J_2 perturbation is the main perturbation and should be considered. Accordingly, the vector $\boldsymbol{f}_{\mathrm{p}}$ is expressed in LVLH coordinate as follows:

$$\begin{aligned} f_{\mathrm{pr}} &= -\frac{3}{2}J_2\frac{\mu R_{\mathrm{e}}^2 W^4}{x_1^4}\left(1 - \frac{12G^2}{S^2}\right), \\ f_{\mathrm{pt}} &= -\frac{3}{2}J_2\frac{\mu R_{\mathrm{e}}^2 W^4}{x_1^4}\left\{\frac{4\left[\left(x_4^2 - x_5^2\right)\sin 2x_6 - 2x_4 x_5 \cos 2x_6\right]}{S^2}\right\}, \\ f_{\mathrm{pn}} &= -\frac{3}{2}J_2\frac{\mu R_{\mathrm{e}}^2 W^4}{x_1^4}\left[\frac{4\,(2-S)\,G}{S^2}\right], \end{aligned} \tag{6}$$

where R_{e} represents the Earth radius.

In order to facilitate numerical propagation of spacecraft dynamics, the equations of motion are normalized by appropriate characteristic length, time and mass that will be described in sect. 4, as they vary with each application. Finally, reference physical constants which will be used in all simulations for this paper, are given in Table 1.

2.2 Energy-optimal control problem

An optimal trajectory and control input to transfer the spacecraft from a given orbit state to a target orbit state can be obtained by minimization of energy consumption with appropriate constraint conditions, and the energy consumption can be expressed as

$$J = \frac{T_{\max}}{I_{\mathrm{sp}} g_0}\int_{t_0}^{t_{\mathrm{f}}} \|\boldsymbol{u}\|^2 \mathrm{d}t, \tag{7}$$

where t_0 and t_{f} denote the initial and final times, respectively, and they are both fixed. In this paper, both rendezvous and orbit transfer problems will be considered. Correspondingly, the boundary conditions for the two scenarios are described as follows.

• Boundary conditions for rendezvous problems

In rendezvous problems, the initial mass, initial states, and final states are all fixed, while the final mass is free. Thus, the following boundary constraints must be satisfied:

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \boldsymbol{x}(t_{\mathrm{f}}) = \boldsymbol{x}_{\mathrm{f}}, \tag{8}$$

$$m(t_0) = m_0, m(t_{\mathrm{f}}) = \text{Free}. \tag{9}$$

According to the transversality condition, the boundary costates are free when the corresponding boundary states are fixed. Thus, the initial costates and the final costates should be free

$$\lambda_x(t_0) = \text{Free}, \lambda_x(t_{\mathrm{f}}) = \text{Free}. \tag{10}$$

Since the final mass is free, the final costate of mass should be zero as

$$\lambda_m(t_{\mathrm{f}}) = 0. \tag{11}$$

• Boundary conditions for orbit transfer problems

In orbit transfer problems, the initial mass and initial orbit states, are both fixed. In contrast, the final mass is free. In addition, which final states are free or fixed depends on the geometry of the final orbit. If the destination orbit is circular, the following boundary constrains need to be satisfied:

$$\begin{aligned} &\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \\ &p(t_{\mathrm{f}}) = p_0, f(t_{\mathrm{f}}) = 0, g(t_{\mathrm{f}}) = 0, \\ &h(t_{\mathrm{f}}) = \text{Free}, k(t_{\mathrm{f}}) = \text{Free}, L(t_{\mathrm{f}}) = \text{Free}, \end{aligned} \tag{12}$$

$$m(t_0) = m_0, m(t_{\mathrm{f}}) = \text{Free}. \tag{13}$$

According to the transversality conditions, the initial costates and the final costates should be free or zero as follows:

$$\begin{aligned} &\lambda_x(t_0) = \text{Free}, \\ &\lambda_p(t_{\mathrm{f}}) = \lambda_f(t_{\mathrm{f}}) = \lambda_g(t_{\mathrm{f}}) = \text{Free}, \\ &\lambda_h(t_{\mathrm{f}}) = \lambda_k(t_{\mathrm{f}}) = \lambda_L(t_{\mathrm{f}}) = 0, \\ &\lambda_m(t_{\mathrm{f}}) = 0. \end{aligned} \tag{14}$$

To maintain the thrust magnitude below its maximum value during the transfer process, the following inequality

Table 1 Physical constants

Quantity	Value
g_{earth}	9.80665 m s^{-2}
μ_{earth}	3.9860047×10^{14} m^3 s^{-2}
R_{e}	6378140 m
J_2	1082.639×10^{-6}
μ_{sun}	1.327124×10^{20} m^3 s^{-2}

path constraint is enforced throughout the trajectory:

$$\|\boldsymbol{u}\| \leqslant 1. \tag{15}$$

The slack variable α is introduced to transform the inequality constraint to an equality form:

$$\|\boldsymbol{u}\| - 1 + \boldsymbol{\alpha} = 0, \alpha \geqslant 0. \tag{16}$$

Thus, the energy-optimal control problem is constructed by state eq. (2) and performance index eq. (7). with boundary conditions eqs. (8)–(11). or eqs. (12)–(14). and path constraint eq. (16).

2.3 Quasi-linearization method

To construct the symplectic-preserving condition both the state and constraint equations need to be linear in the control variable [22]. In order to apply symplectic method, quasi-linearization techniques are applied in this paper. The state and constraint equations are linearized, while the cost function is expanded up to second order around a reference solution. Thus, the original nonlinear optimal control problem is transformed into a sequence of constrained linear quadratic optimal control sub-problems that can be solved individually via a symplectic method. The solution to the quasi-linear problem is, then, utilized as new reference, and this process is iterated until convergence. Each time the reference is updated, the algorithm advances by one iteration.

Denoting the state vector $\boldsymbol{x} = (p, f, g, h, k, L, m)$ and the control vector $\boldsymbol{u} = (u_x, u_y, u_z)$, the constrained linear quadratic optimal control sub-problem at the $(k+1)$ iteration can be described by the following state equations:

$$\dot{\boldsymbol{x}}^{(k+1)} = \boldsymbol{A}^{(k)}\boldsymbol{x}^{(k+1)} + \boldsymbol{B}^{(k)}\boldsymbol{u}^{(k+1)} + \boldsymbol{w}^{(k)}, \tag{17}$$

where

$$\boldsymbol{A}^{(k)} = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u},t)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}^{(k)},\boldsymbol{u}^{(k)}}, \boldsymbol{B}^{(k)} = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u},t)}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}^{(k)},\boldsymbol{u}^{(k)}}, \tag{18}$$

$$\boldsymbol{w}^{(k)} = \boldsymbol{f}(\boldsymbol{x}^{(k)}, \boldsymbol{u}^{(k)}, t) - \boldsymbol{A}^{(k)}\boldsymbol{x}^{(k)} - \boldsymbol{B}^{(k)}\boldsymbol{u}^{(k)}. \tag{19}$$

Subject to the path constraints:

$$\boldsymbol{h}^{(k+1)}(\boldsymbol{x},\boldsymbol{u},t) = \boldsymbol{C}^{(k)}\boldsymbol{x}^{(k+1)} + \boldsymbol{D}^{(k)}\boldsymbol{u}^{(k+1)} + \boldsymbol{v}^{(k)} \leqslant 0, \tag{20}$$

where

$$\boldsymbol{C}^{(k)} = \left.\frac{\partial h(\boldsymbol{x},\boldsymbol{u},t)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}^{(k)},\boldsymbol{u}^{(k)}}, D^{(k)} = \left.\frac{\partial h(\boldsymbol{x},\boldsymbol{u},t)}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}^{(k)},\boldsymbol{u}^{(k)}}, \tag{21}$$

$$\boldsymbol{v}^{(k)} = \boldsymbol{h}(\boldsymbol{x}^{(k)}, \boldsymbol{u}^{(k)}, t) - C^{(k)}\boldsymbol{x}^{(k)} - D^{(k)}\boldsymbol{u}^{(k)}. \tag{22}$$

The cost function is also transformed into:

$$J^{(k+1)} = \int_{t_0}^{t_f} g^{(k+1)}\mathrm{d}t, \tag{23}$$

where

$$g^{(k+1)} = \bar{g}^{(k)} + (\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)})E^{(k)} + \frac{1}{2}(\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)})^T F^{(k)}(\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)}), \tag{24}$$

$$\bar{g}^{(k)} = \frac{1}{2}\|\boldsymbol{u}\|^2\,|_{\boldsymbol{u}^{(k)}}, \tag{25}$$

$$\boldsymbol{E}^{(k)} = \left.\frac{\partial \bar{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{u}^{(k)}}, \boldsymbol{F}^{(k)} = \left.\frac{\partial^2 \bar{g}}{\partial \boldsymbol{u}^2}\right|_{\boldsymbol{u}^{(k)}}. \tag{26}$$

Superscript in the above equations are an iteration index: the symbol $(k+1)$ denotes variable values in the current $(k+1)$ iteration, (k) refers to values at the previous k iteration, which serve as the initial reference for the current update. Therefore, the original nonlinear optimal control problem is transformed into a sequence of constrained linear quadratic control sub-problems. The iteration process ends when the variation of the orbit states is smaller than a given tolerance. The convergence criteria is defined as

$$\frac{\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|}{\|\boldsymbol{x}_k\|} \leqslant \varepsilon, \tag{27}$$

where ε is a small quantity which denotes the selected tolerance. Next, a symplectic method is proposed to obtain the solution of the linear quadratic control sub-problem at each iteration.

3 Symplectic approach for constrained linear quadratic optimal control

In this section, a symplectic method based on dual variational principle is proposed to obtain the solution of the linear quadratic optimal control sub-problems. First, the symplectic-preserving condition is obtained according to dual variational principle. Then, based on the symplecitc-preserving condition, the symplectic algorithm is constructed. To deal with path constraint in eq. (16), at the discrete points, the original linear quadratic control sub-problem is transcribed into an explicit linear complementary problem. Finally, the solution of the original optimal control problem can be obtained by solving the standard linear complementary problem. For brevity, the iteration index will be omitted in the following derivations.

3.1 Derivation of the symplectic-preserving condition

First, based on the dual variational principle, the symplectic-condition is derived. The derivation follows that in ref. [22]. The Hamilton function for each constrained linear quadratic optimal control sub-problem can be obtained as follows:

$$H = \bar{g} + (\boldsymbol{u} - \boldsymbol{u}_d)\boldsymbol{E} + \frac{1}{2}(\boldsymbol{u} - \boldsymbol{u}_d)^{\mathrm{T}}\boldsymbol{F}(\boldsymbol{u} - \boldsymbol{u}_d)$$

$$+\boldsymbol{\lambda}(\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{u}+\boldsymbol{w})+\boldsymbol{\beta}(\boldsymbol{C}\boldsymbol{x}+\boldsymbol{D}\boldsymbol{u}+\boldsymbol{v}+\boldsymbol{\alpha}), \tag{28}$$

where the subscript d represents the initial reference for the current iteration. The parameter variable β should satisfy the complementary condition:

$$\boldsymbol{\alpha}\geqslant\boldsymbol{0},\boldsymbol{\beta}\geqslant\boldsymbol{0},\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\beta}=0. \tag{29}$$

The first order necessary conditions of the optimal control problem can be obtained by the following equations:

$$\frac{\partial H}{\partial \boldsymbol{u}}=0. \tag{30}$$

Substituting eq. (28) to eq. (30), the expression for the control variable at the $(k+1)$ iteration can be obtained as

$$\boldsymbol{u}=\boldsymbol{u}_d-\boldsymbol{F}[\boldsymbol{E}+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{\lambda}+\boldsymbol{D}^{\mathrm{T}}\boldsymbol{\beta}]. \tag{31}$$

Substituting eq. (31) back into eq. (28), yields

$$\begin{aligned}H=&\bar{g}-\frac{1}{2}(\boldsymbol{E}^{\mathrm{T}}+\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{B}+\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{D})\boldsymbol{F}^{-1}(\boldsymbol{E}+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{\lambda}+\boldsymbol{D}^{\mathrm{T}}\boldsymbol{\beta})\\&+\boldsymbol{\lambda}(\boldsymbol{A}\boldsymbol{x}+\boldsymbol{w})+\boldsymbol{\beta}(\boldsymbol{C}\boldsymbol{x}+\boldsymbol{v}+\boldsymbol{\alpha})+(\boldsymbol{\lambda}\boldsymbol{B}+\boldsymbol{\beta}\boldsymbol{D})\boldsymbol{u}_d.\end{aligned} \tag{32}$$

In addition, by substituting eq. (31) back into eq. (16), the eq. (20) can be rewritten as

$$\boldsymbol{C}\boldsymbol{x}+\boldsymbol{D}\boldsymbol{u}_d-\boldsymbol{D}\boldsymbol{F}^{-1}\left[\boldsymbol{E}+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{\lambda}+\boldsymbol{D}^{\mathrm{T}}\boldsymbol{\beta}\right]+\boldsymbol{v}+\boldsymbol{\alpha}=\boldsymbol{0}. \tag{33}$$

Therefore, the Hamilton function is independent from the control variable. We define an action S in a generic time interval (a,b) as

$$S=\int_a^b[\boldsymbol{\lambda}^{\mathrm{T}}\dot{\boldsymbol{x}}-H]\mathrm{d}t. \tag{34}$$

Based on the action S, the fourth kind of generating function is produced:

$$V=\boldsymbol{\lambda}_a^{\mathrm{T}}\boldsymbol{x}_a-\boldsymbol{\lambda}_b^{\mathrm{T}}\boldsymbol{x}_b+S. \tag{35}$$

Variation of the fourth generating function yields

$$\delta V=\boldsymbol{x}_a^{\mathrm{T}}\mathrm{d}\boldsymbol{\lambda}_a-\boldsymbol{x}_b^{\mathrm{T}}\mathrm{d}\boldsymbol{\lambda}_b+\int_a^b\left[\left(\dot{\boldsymbol{x}}-\frac{\partial H}{\partial\boldsymbol{\lambda}}\right)\delta\boldsymbol{\lambda}-\left(\dot{\boldsymbol{\lambda}}+\frac{\partial H}{\partial\boldsymbol{x}}\right)\delta\boldsymbol{x}\right]\mathrm{d}t. \tag{36}$$

According to the variation principle, the optimal solution should satisfy the Hamilton canonical equations:

$$\dot{\boldsymbol{x}}=\frac{\partial H}{\partial\boldsymbol{\lambda}},\dot{\boldsymbol{\lambda}}=-\frac{\partial H}{\partial\boldsymbol{x}}. \tag{37}$$

Then, the relationship between the action V and the states and costates at two ends of the interval is given by

$$\delta V=\boldsymbol{x}_a^{\mathrm{T}}\mathrm{d}\boldsymbol{\lambda}_a-\boldsymbol{x}_b^{\mathrm{T}}\mathrm{d}\boldsymbol{\lambda}_b. \tag{38}$$

In this formulation, the costate variables at the extremes of the time interval (a,b) are the free variables, also called independent variables. And the action V must only be the function of $\boldsymbol{\lambda}_a$ and $\boldsymbol{\lambda}_b$. Numerical method that satisfys eq. (38) is symplectic-preserving, which is demonstrated in ref. [22].

3.2 Construction of the symplectic algorithm

Based on the symplectic-preserving condition, the symplectic algorithm is constructed next. First, the trajectory is divided into N arcs with equal time intervals $\eta=(t_{\mathrm{f}}-t_0)/N$. Therefore, the time interval of jth trajectory arc can be define as $[t_j,t_{j-1}]$, where $t_{j-1}=(j-1)\eta$ and $t_j=j\eta$. Correspondingly, state variables and costate variables at t_j are denoted as $\boldsymbol{x}_j$ and $\boldsymbol{\lambda}_j$, respectively. Within each trajectory arc, the state vector $\boldsymbol{x}(t)$ and the costate vector $\boldsymbol{\lambda}(t)$ are approximated by using Lagrange interpolation polynomials with order $m-1$ and $n-1$, respectively. We have tried to combine our methods with the symplectic methods using other interpolation schemes, including the Legendre Gauss Lobatto interpolation schemes [29], Legendre Gauss interpolation schemes, etc. The numerical results indicate that our methods also can improve the convergence of symplectic methods using other interpolation scheme. Note that, the costate variables at both ends of each arc form the set of independent variables, state and costate variables at the internal interpolation points are not considered independent variables. The parameter variables $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are assumed to be constant. A scheme of the trajectory discretization is depicted in Figure 1. The resulting system of expressions is

$$\boldsymbol{x}(t)=(\boldsymbol{M}\otimes I)\bar{\boldsymbol{x}}_j, \tag{39}$$

$$\boldsymbol{\lambda}(t)=N_1\boldsymbol{\lambda}_{j-1}+(\bar{\boldsymbol{N}}\otimes I)\bar{\boldsymbol{\lambda}}_j+N_n\boldsymbol{\lambda}_j, \tag{40}$$

$$\boldsymbol{\alpha}(t)=\boldsymbol{\alpha}_j, \tag{41}$$

$$\boldsymbol{\beta}(t)=\boldsymbol{\beta}_j, \tag{42}$$

where $\bar{\boldsymbol{x}}_j$ comprises all the state variables at both the extreme and interpolation points within the jth arc, defined as $\bar{\boldsymbol{x}}_j=[\bar{\boldsymbol{x}}_j^1,\bar{\boldsymbol{x}}_j^2,...,\bar{\boldsymbol{x}}_j^m]^{\mathrm{T}}$, $\boldsymbol{\lambda}_{j-1}$ and $\boldsymbol{\lambda}_j$ denote the costate variables at the left and right end of the jth arc, $\bar{\boldsymbol{\lambda}}_j$ is composed of the remaining dependent costate variables at the interpolation

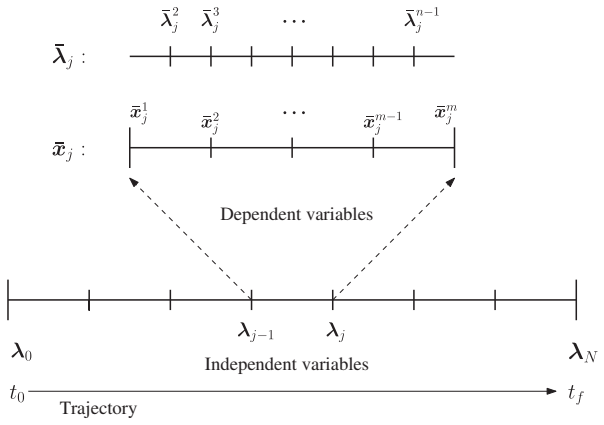

Figure 1 Trajectory discretization scheme.

points within the jth arc, defined as $\bar{\boldsymbol{\lambda}}_j = [\bar{\boldsymbol{\lambda}}_j^2, \bar{\boldsymbol{\lambda}}_j^3, ..., \bar{\boldsymbol{\lambda}}_j^{n-1}]^{\mathrm{T}}$, $\boldsymbol{I}$ denotes a $n \times n$ identity matrix, and the symbol $\otimes$ denotes the Kronecker product of two matrices. Other matrices in eqs. (39) and (40) are defined as

$$\boldsymbol{M} = [M_1, M_2, ..., M_m], \tag{43}$$

$$\bar{\boldsymbol{N}} = [N_2, N_3, ..., N_{n-1}], \tag{44}$$

$$M_i = \prod_{j=1, j\neq i}^{m} \frac{t-(j-1)\eta/(m-1)}{(i-j)\eta/(m-1)}, \tag{45}$$

$$N_i = \prod_{j=1, j\neq i}^{n} \frac{t-(j-1)\eta/(n-1)}{(i-j)\eta/(n-1)}. \tag{46}$$

Substituting interpolated state and costate variables eqs. (39) and (40) into eq. (38) gives

$$V_j = \boldsymbol{\lambda}_{j-1}^{\mathrm{T}} \bar{\boldsymbol{x}}_j^1 - \boldsymbol{\lambda}_j^{\mathrm{T}} \bar{\boldsymbol{x}}_j^r + \int_{t_{j-1}}^{t_j} (\boldsymbol{\lambda}^{\mathrm{T}} \dot{\boldsymbol{x}} - H) \mathrm{d}t. \tag{47}$$

Hence, based on eq. (38), the following equations must be satisfied at each arc

$$\boldsymbol{F}_1^j = \boldsymbol{x}_{j-1}, \tag{48}$$

$$\boldsymbol{F}_2^j = \boldsymbol{0}, \tag{49}$$

$$\boldsymbol{F}_3^j = \boldsymbol{0}, \tag{50}$$

$$\boldsymbol{F}_4^j + \boldsymbol{x}_j = \boldsymbol{0}, \tag{51}$$

where

$$\begin{aligned} \boldsymbol{F}_1^j = \frac{\partial V_j}{\partial \boldsymbol{\lambda}_{j-1}} &= \boldsymbol{K}_{11}^j \boldsymbol{\lambda}_{j-1} + \left(\boldsymbol{E}_u^{\mathrm{T}} + \boldsymbol{K}_{12}^j\right) \bar{\boldsymbol{x}}_j + \boldsymbol{K}_{13}^j \bar{\boldsymbol{\lambda}}_j \\ &+ \boldsymbol{K}_{14}^j \boldsymbol{\lambda}_j + \boldsymbol{f}_1^j, \end{aligned} \tag{52}$$

$$\begin{aligned} \boldsymbol{F}_2^j = \frac{\partial V_j}{\partial \bar{\boldsymbol{x}}_j} &= \left(\boldsymbol{K}_{21}^j + \boldsymbol{E}_u\right) \boldsymbol{\lambda}_{j-1} - \boldsymbol{K}_{22}^j \bar{\boldsymbol{x}}_j + \boldsymbol{K}_{23}^j \bar{\boldsymbol{\lambda}}_j \\ &+ \left(\boldsymbol{K}_{24}^j - \boldsymbol{E}_d\right) \boldsymbol{\lambda}_j + \boldsymbol{f}_2^j, \end{aligned} \tag{53}$$

$$\boldsymbol{F}_3^j = \frac{\partial V_j}{\partial \bar{\boldsymbol{\lambda}}_j} = \boldsymbol{K}_{31}^j \boldsymbol{\lambda}_{j-1} + \boldsymbol{K}_{32}^j \bar{\boldsymbol{x}}_j + \boldsymbol{K}_{33}^j \bar{\boldsymbol{\lambda}}_j + \boldsymbol{K}_{34}^j \boldsymbol{\lambda}_j + \boldsymbol{f}_3^j, \tag{54}$$

$$\begin{aligned} \boldsymbol{F}_4^j = \frac{\partial V_j}{\partial \boldsymbol{\lambda}_j} &= \boldsymbol{K}_{41}^j \boldsymbol{\lambda}_{j-1} + \left(\boldsymbol{K}_{42}^j - \boldsymbol{E}_d^{\mathrm{T}}\right) \bar{\boldsymbol{x}}_j + \boldsymbol{K}_{43}^j \bar{\boldsymbol{\lambda}}_j \\ &+ \boldsymbol{K}_{44}^j \boldsymbol{\lambda}_j + \boldsymbol{f}_4^j. \end{aligned} \tag{55}$$

The detailed expressions for $\boldsymbol{K}_{u,v}^j\,(u, v = 1, 2, 3, 4)$ and $\boldsymbol{f}_i^j\,(i = 1, 2, 3, 4)$ in eqs. (52)–(55) can be found in ref. [26]. Thus, the symplectic algorithm in an trajectory arc is contructed by eqs. (52)–(55). Applying eq. (38) to the each arc throughout the entire trajectory, yields the following nonlinear equation:

$$\boldsymbol{F}_1^{j+1} + \boldsymbol{F}_4^j = \boldsymbol{0}\,(j = 1, \cdots, N-1)\,. \tag{56}$$

Since the derivations of nonlinear eqs. (48)–(56) are based on dual variational principle and symplectic-preserving condition, it is a symplectic numerical method.

3.3 Formulation of the complementary problem

Note that, the number of nonlinear eqs. (48)–(56) is less than the unknown variables, which means the nonlinear equation can not be solved directly. In this subsection, the complementary problem is formulated based on the eqs. (48)–(56) and the path constraint eq. (33).

The $\bar{\boldsymbol{x}}_j$ and $\bar{\boldsymbol{\lambda}}_j$ vectors can be expressed using the independent variables $\boldsymbol{\lambda}_{j-1}$ and $\boldsymbol{\lambda}_j$ by solving eqs. (49) and (50). After that, substituting the expression for vectors $\bar{\boldsymbol{x}}_j$ and $\bar{\boldsymbol{\lambda}}_j$ into eqs. (48) and (51), yields

$$\boldsymbol{F}_1^j = \boldsymbol{S}_{11}^j \boldsymbol{\lambda}_{j-1} + \boldsymbol{S}_{12}^j \boldsymbol{\lambda}_j + \boldsymbol{\zeta}_{11}^j + \boldsymbol{\zeta}_{12}^j \boldsymbol{\beta}_j, \tag{57}$$

$$\boldsymbol{F}_4^j = \boldsymbol{S}_{21}^j \boldsymbol{\lambda}_{j-1} + \boldsymbol{S}_{22}^j \boldsymbol{\lambda}_j + \boldsymbol{\zeta}_{21}^j + \boldsymbol{\zeta}_{22}^j \boldsymbol{\beta}_j. \tag{58}$$

The detailed expression of $\boldsymbol{S}_{u,v}^j\,(u, v = 1, 2)$ and $\boldsymbol{\zeta}_{pq}^j\,(p, q = 1, 2)$ can be seen in ref. [26] (see eqs. (50)–(57) and appendix). Utilizing eqs. (48), (51) and (55), the eqs. (57) and (58) can be expressed in a impact form as

$$\hat{\boldsymbol{\lambda}} = \boldsymbol{A}^{-1} \boldsymbol{\Psi} \hat{\boldsymbol{\beta}} + \boldsymbol{A}^{-1} \boldsymbol{\Phi}. \tag{59}$$

Refer to ref. [28] for a detailed expressions for the above matrices $\boldsymbol{A}, \boldsymbol{\Psi}, \boldsymbol{\Phi}$. Then, the state vector can be expressed in the parameter $\hat{\boldsymbol{\beta}}$ by utilizing eqs. (51) and (58)

$$\boldsymbol{x}_j = -\boldsymbol{F}_4^j = -(\boldsymbol{S}_{21}^j \boldsymbol{\lambda}_{j-1} + \boldsymbol{S}_{22}^j \boldsymbol{\lambda}_j + \boldsymbol{\zeta}_2^j). \tag{60}$$

That allows to express the state and costate vectors at the discretization points along the trajectory through the parameter variable $\hat{\boldsymbol{\beta}}$. Moreover, the complementary conditions in eqs. (33) and (29) also need to be satisfied. Substituting eqs. (59) and (60) back to eq. (33), a standard linear complementary problem can be obtained:

$$\hat{\boldsymbol{\alpha}} - \boldsymbol{M}_{\mathrm{new}} \hat{\boldsymbol{\beta}} = \boldsymbol{q}_{\mathrm{new}}, \tag{61}$$

$$\hat{\boldsymbol{\alpha}} \geqslant \boldsymbol{0}, \hat{\boldsymbol{\beta}} \geqslant \boldsymbol{0}, \hat{\boldsymbol{\beta}}^{\mathrm{T}} \hat{\boldsymbol{\alpha}} = \boldsymbol{0}. \tag{62}$$

The symbol $\boldsymbol{M}_{\mathrm{new}}$, $\boldsymbol{q}_{\mathrm{new}}$ and derivation process follows that in ref. [28]. In general, the parameter $\hat{\boldsymbol{\beta}}$ at the discretization points can be obtained by solving the standard linear complementary problem. In this paper, we utilize lemke method to solve the formulated linear complementary problem. Next, the costate vector can be obtained by substituting $\hat{\boldsymbol{\beta}}$ into eq. (59); similarly the state vector is produced by eq. (60). Finally, the control input is derived by eq. (31). Following this procedure, constrained linear quadratic optimal control with given terminal states can be solved. In addition, the matrices $\boldsymbol{A}, \boldsymbol{\Psi}, \boldsymbol{\Phi}, \boldsymbol{M}_{\mathrm{new}}$ all have sparse structure with small band width, which makes the numerical implementation of the proposed method highly efficient. It should be noted that, eqs.

(59) and (61) can be modified to reflect the desired boundary conditions.

As it is described in sect. 2.2, boundary conditions for rendezvous problems and orbit transfer problems are considered in this paper. In the case of boundary conditions for rendezvous problems, the costate for the final mass equals zero. Thus, the last row of vector $\boldsymbol{\lambda}_N$ is removed from the list of unknown variables. Correspondingly, the last row of the $\hat{\boldsymbol{\beta}}$, eqs. (59) and (61) should also be deleted. In the case of boundary conditions for orbit transfer problems, the costate of the final mass and the costate of the last three components of the state vector are zero. Thus, the last four rows of $\boldsymbol{\lambda}_N$ are removed from the list of unknown variables. Similarly, the last four rows of $\hat{\boldsymbol{\beta}}$, eqs. (59) and (61) should be deleted.

4 Numerical examples and discussion

This section presents three examples of energy-optimal transfers with multiple revolutions to illustrate the accuracy and efficiency of the techniques and methods presented in the last two sections. To capture the oscillation of the state variables well, the number of the interpolation points for the state variables in a sub-interval is set to be 4, and that for the costate variables is set to be 5. All computations are performed in MATLAB (R2016b) on a desktop computer with a CPU of 4.00 GHz. The heliocentric position and velocity vectors of the planets, when needed, are computed online using the Jet Propulsion Laboratory Horizons system. The value of convergence tolerance in eq. (27) is set to be 1.0×10^{-6}.

4.1 Generation of nominal trajectories

Since the state equations of spacecraft are linearized around a sequence of reference trajectories, the iteration process for achieving an acceptable error is impacted by the quality of the initial guess, especially for orbit transfer problems with multiple revolutions. If the initial nominal trajectory is too far from the true optimal trajectory, the symplectic method presented in this paper may not converge to the optimal solution. In contrast to other studies [22, 27, 28], the initial reference trajectory is generated by linear interpolation of the state variables, which are the non-singular equinoctial orbital elements and the mass of the spacecraft between the initial and final trajectory points. Empirically, that results in an effective strategy for multi-revolution transfers. Initially, the control variable value at every discretization point is identically set to 0.005 N. Although the initial nominal trajectory may be neither optimal nor feasible, an optimal, feasible trajectory can be generally obtained after a small number of iterations with the symplectic method.

4.2 Rendezvous from Earth to Venus

A low-thrust rendezvous problem from Earth to Venus is considered in this section. Namely, the spacecraft starts at the instantaneous Earth heliocentric position and velocity and arrives at Venus with its same instantaneous heliocentric position and velocity. This example exactly replicates that in ref. [6], and is presented to illustrate the accuracy of results that are obtained by our optimization strategy. In ref. [6], the optimal trajectory is obtained via an indirect method and will serve as comparison. The method in ref. [6] includes an homotopic transformation of the solution. Since we search for the energy-optimal trajectory, we only consider the solution for an homotopic parameter equals to one. All the input parameters are listed in Table 2. For computational convenience, length, time, and mass are nondimensionalized by the astronomical unit (AU, 149597870.66 km), solar year (yr, 365.25×86400 s), and spacecraft initial mass, respectively.

Converged results obtained by the symplectic method with different number of trajectory arcs are listed in Table 3. It can be seen in Table 3 that, the number of iterations to converge to the optimal solution is not influenced by the number of arcs. In contrast, a dozen of grams may add to the final mass if the number of arcs is increased. We also note from Table 3 that, the third decimal place of the final mass value is converged when the number of arcs equals 20, while the fourth decimal place of the final mass value is converged when the number

Table 2 Parameters for a representative Earth-to-Venus transfer

Parameter	Value	Units
Initial date	October 7, 2005 0:0:0.0	Coordinate time
Flight time	1000	d
Initial position	$\left[9.708322\times10^{-1}, 2.375844\times10^{-1}, -1.671055\times10^{-6}\right]$	AU
Initial velocity	$\left[-1.598191, 6.081958, 9.443368\times10^{-5}\right]$	AU/yr
Final position	$\left[-3.277178\times10^{-1}, 6.389172\times10^{-1}, 2.765929\times10^{-2}\right]$	AU
Final velocity	$\left[-6.598211, -3.412933, 3.340902\times10^{-1}\right]$	AU/yr
I_{sp}	3800	s
T_{max}	0.33	N
m_0	1500	kg

Table 3 Converged solutions from the symplectic method with different number of arcs

Number of arcs	Number of iterations	Computational time (s)	Final mass (kg)
5	9	0.136186	1274.747782
10	9	0.326455	1274.959963
15	9	0.363343	1274.959691
20	9	1.220348	1274.957674
25	9	1.749006	1274.957120
30	9	3.145607	1274.956974
35	9	4.625240	1274.956923

of arcs equal 30. The indirect method predicts a final mass of 1274.956883 kg, with a variation of 0.0304 kg when the number of arcs equals 10, and a variation of 0.00043 kg when the number of arcs equals 35. Thus, the relationship between the number of arcs and the accuracy of the optimal solution can be inferred. That is, if one revolution contains 3 or 4 arcs, the symplectic method can produce the optimal solution with relatively high accuracy. When the number of arcs equals to 9 or 10 in a revolution, the symplectic method can achieve the same precision of the indirect method. This fact is also demonstrated in other numerical examples. The accuracy of the proposed method can be improved by increasing the number of discretization points, as demonstrated by Figure 2. Figure 2 clearly illustrates that the difference of final mass, m_f, between the indirect and the symplectic method decreases as the number of the trajectory arcs increases. Since solutions via the indirect method are guaranteed to be at least locally optimal, the optimality of the trajectories produced by the symplectic method is also demonstrated in this example.

A comparison of the optimal low-thrust multi-revolution trajectories solved by the symplectic method and indirect method is displayed in Figure 3. The symplectic method uses 35 arcs. Parameters of the indirect method are set to follow that in ref. [6]. The symbol represents the trajectory obtained from the symplectic method, while the dashed orange line renders the trajectory obtained from the indirect method. Figure 3 portrays the path of the spacecraft from the initial Earth heliocentric position and velocity to the Venus rendezvous by matching Venus heliocentric position and velocity after 3 orbital revolutions. Both the symplectic method and the indirect method produce nearly identical optimal low-thrust trajectories.

The time evolution for costate variables of the symplectic and indirect method is depicted in Figure 4, denoted by stars and lines respectively. From Figure 4, it is clear that the costate variables obtained from the two methods are in a close agreement. From Figure 4, it is easy to verify that the terminal mass costate λ_m satisfies the transversallity condition in eq. (14), i.e. $\lambda_m(t_f) = 0$, and demonstrates that boundary conditions for the spacecraft mass costate are also satisfied.

The optimal thrust profile is displayed in Figure 5. Both, the symplectic and indirect method converge on nearly identical optimal thrust profiles. In addition, the thrust magnitude is below one during the entire transfer, and the path constraint in eq. (15) is satisfied.

Thus, for this problem, since the solution from the two methods are in close agreement, we can conclude that the symplectic method converges on the locally optimal solution

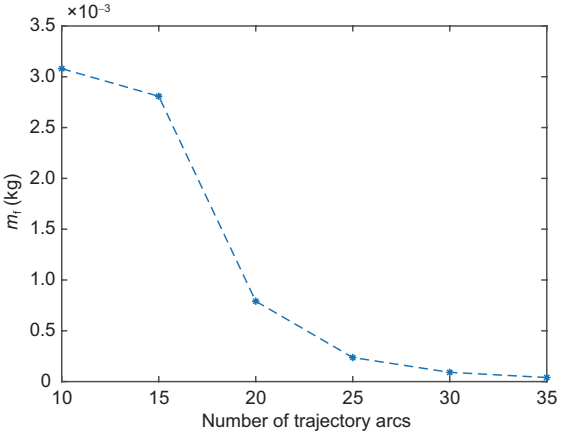

Figure 2 (Color online) Final mass difference between the indirect and symplectic algorithm as a function of number of trajectory arcs.

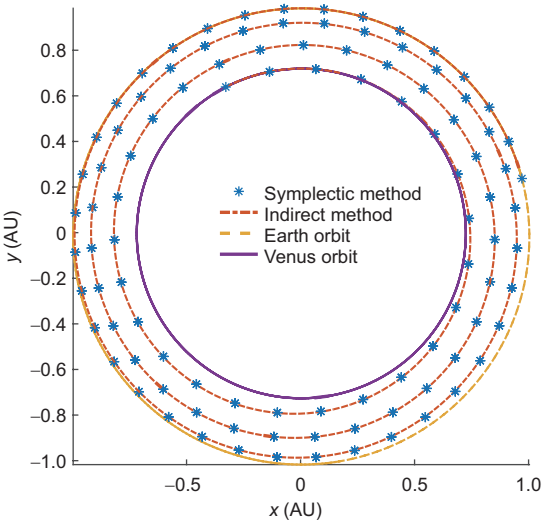

Figure 3 (Color online) Low-thrust trajectory from Earth to Venus.

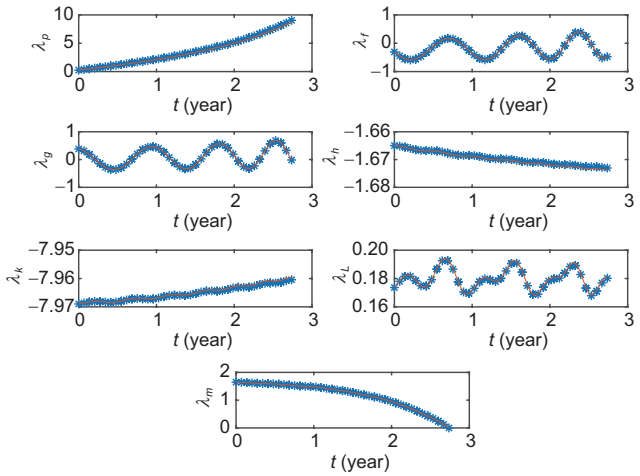

Figure 4 (Color online) Costate variables time histories for the optimal rendezvous trajectory from Earth to Venus.

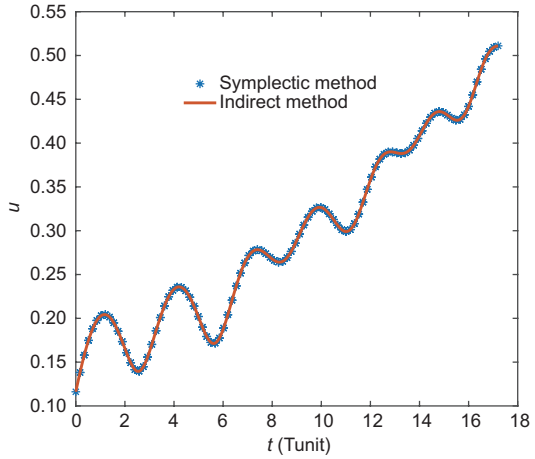

Figure 5 (Color online) Optimal thrust profile of the rendezvous from Earth to Venus.

with high accuracy.

4.3 Orbital transfer between two circular orbits

In this section, a low-thrust orbit transfer problem between two circular orbits around the Sun is considered: the spacecraft starts from the instantaneous Earth heliocentric position and velocity and arrives at a final, given circular orbit. This example replicates that in ref. [30], and is presented to illustrate the efficiency of the symplectic method. The specific impulse I_{sp} is assumed to be constant and equals to 2000 s and the initial mass of the spacecraft is 1000 kg. The initial epoch is 00:00:00, April 10th, 2007, and the corresponding Earth position and velocity vectors at this epoch are retrieved from JPL ephemerides DE405:

$$r_0 = [-140699693, -51614428, 980] \text{ km}, \quad (63)$$

$$v_0 = [9.774596, -28.07828, 4.337725 \times 10^{-4}] \text{ km/s}. \quad (64)$$

The final spacecraft terminates in a circular orbit with radius $a_{\text{target}} = 1.95$ AU. Since the final orbit is a circular orbit, the eccentricity is zero. The remaining four Keplerian elements are free. The corresponding equinoctial orbit elements are

$$p(t_f) = 1.95 \text{ AU},$$
$$f(t_f) = 0,$$
$$g(t_f) = 0,$$
$$h(t_f) = \text{Free},$$
$$k(t_f) = \text{Free},$$
$$L(t_f) = \text{Free}.$$

To facilitate numerical computations, length, time, and mass are nondimensionalized as in the last section. Since in both the current and previous example, the central body is the Sun, both problems can be nondimensionalized by the same characteristic quantities.

To better understand the influence of the number of revolutions on the optimization process, the optimal transfer is solved for a set of four different times of flight (which correspond to a different number of revolutions). For each given time of flight, the maximum thrust magnitude is adjusted to ensure that there exists a feasible low-thrust transfer. Resulting parameters for the four time of flight cases are listed in Table 4.

The results produced by the optimal control software GPOPS are chosen for comparison to illustrate the efficiency of the symplectic method. GPOPS is an open source MATLAB software developed by ref. [31], for solving complex optimal control problems using the nonlinear programming solver SNOPT, which is developed by Gill et al. [32]. To make a legit comparison of the algorithm efficiency, the initial guess of state variables and control variables are set the same for both the symplectic method and the GPOPS. As for the other parameters of GPOPS, they are listed in the Table 5.

Table 4 Parameters for the numerical examples

Case	N_{rev}	$T_{\max}$ (N)	TOF (d)
1	2	0.2	1165.65
2	5	0.14	2325.30
3	9	0.05	4650.60
4	17	0.015	8719.88

Table 5 Parameters for the optimal control software GPOPS

Parameters	Value
Setup.mesh.tolerance	1×10^{-6}
Setup.mesh.iteration	30
Setup.derivatives	Finite-difference
Setup.checkDerivatives	0
Setup.autoscale	off

The results for different time of flight cases are listed in Table 6. From Table 6, it can be found that, the final mass obtained by symplectic methods is nearly the same as that of GPOPS, which means the solution produced by symplectic methods can have the same accuracy as that of GPOPS. Moreover, it should be noted that, the symplectic method converges to the optimal solution with fewer discretization points when compared to GPOPS. Thus, it can be concluded that the symplectic method can preserve the continuous nature of the original dynamics with fewer discretization points when compared to GPOPS. Besides, the symplectic method is significantly faster than GPOPS in terms of total computational time. Thus, for this example, the efficiency and optimality of the symplectic method can be demonstrated.

For reference, the optimal solution obtained by symplectic method for Case 3 is depicted in Figures 6–9. Figure 6 shows the optimal spacecraft trajectory obtained from the symplectic method and SNOPT solver. The complete orbit transfer contains nearly 9 revolutions. The thrust profiles are portrayed in Figure 7. From Figures 6 and 7, it can be noticed that, the optimal transfers obtained from the two methods are in close agreement. The evolution of the final mass and final true anomaly with the number of iterations during the optimization process for the symplectic method is depicted in Figures 8 and 9. The final mass approximately converges after 3 iterations. During the remaining iterations, the solution slowly updates its final true longitude. As a consequence, the efficiency of the symplectic method can be further improved by giving better initial guesses for the true longitude.

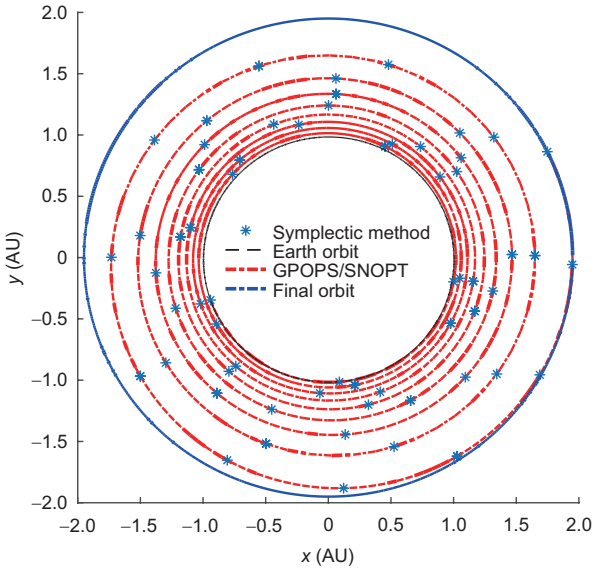

Figure 6 (Color online) Low-thrust trajectory from Earth to a circular orbit.

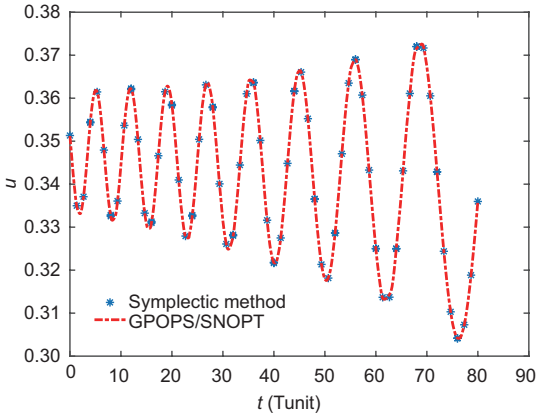

Figure 7 (Color online) Optimal thrust profile of the orbit transfer from Earth to a circular orbit.

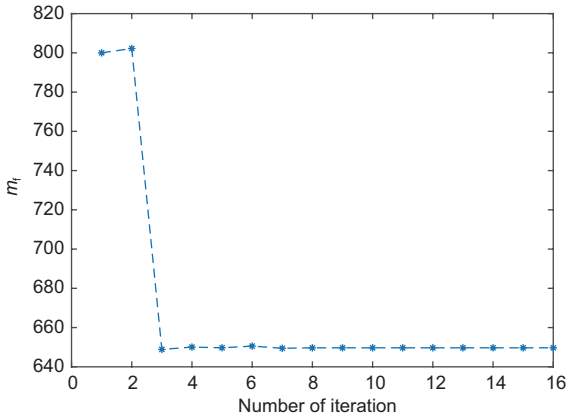

Figure 8 (Color online) Evolution of the final mass during the optimization process for the symplectic method.

Table 6 Comparison between the symplectic method and SNOPT solver for multi-revolution orbital transfers

Case	Method	m_f (kg)	Number of discretization points	CPU time (s)
1	Symplectic method	647.5883	40	0.2269
	GPOPS/SNOPT	647.5883	321	2.2056
2	Symplectic method	649.1790	60	0.3747
	GPOPS/SNOPT	649.1790	466	3.7903
3	Symplectic method	649.6894	80	0.4891
	GPOPS/SNOPT	649.6878	897	8.0261
4	Symplectic method	649.6167	200	3.9055
	GPOPS/SNOPT	649.6168	1661	38.1206

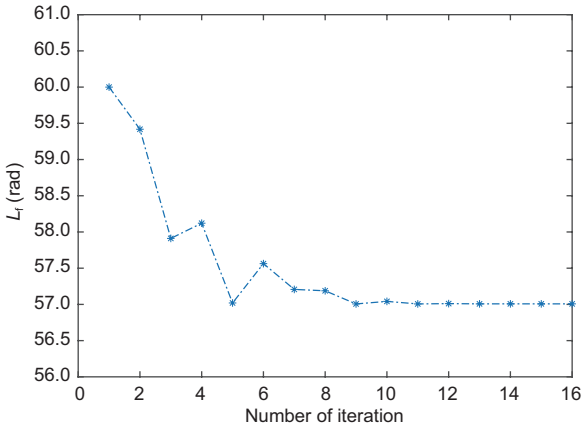

Figure 9 (Color online) Evolution of the final true longitude during the optimization process for the symplectic method.

4.4 Low Earth orbit sacecraft rendezvous

Consider a spacecraft rendezvous mission in low Earth orbit (LEO): the chaser satellite starts from a sun-synchronous orbit and transfer to another sun-synchronous orbit to rendezvous with the target satellite. Unlike the last two numerical examples, the spacecraft dynamics around Earth include the J_2 perturbation, which makes the optimal control problem much challenging to solve [33]. This example illustrates that, the symplectic method can also be applied to optimize low-thrust trajectories with a very large number of revolutions within perturbed dynamics. The specific impulse fixed to 3800 s and the initial mass of the chaser spacecraft is equal to 1500 kg. The maximum thrust magnitude is 0.33 N. The initial state vector of the chaser spacecraft is specified in terms of equinoctial orbit elements as

$$\begin{aligned} &p_0(t_0) = 6899.4468 \text{ km}, \ f_0(t_0) = -0.00008344, \\ &g_0(t_0) = 0.00067183, \ h_0(t_0) = -0.06749657, \\ &k_0(t_0) = -1.13743783, \ L_0(t_0) = 1.85174464 \text{ rad}. \end{aligned} \tag{65}$$

The initial state of the target spacecraft is also specified in terms of equinoctial orbit elements as

$$\begin{aligned} &p_1(t_0) = 6897.4283 \text{ km}, \ f_1(t_0) = -0.00026998, \\ &g_1(t_0) = 0.00040531, \ h_1(t_0) = -0.06750251, \\ &k_1(t_0) = -1.13753794, \ L_1(t_0) = 1.99497980 \text{ rad}. \end{aligned} \tag{66}$$

It should be noted that, the target spacecraft is subject only to the Earth gravity, while the chaser spacecraft is subject both Earth gravity and the thrust of its own electric propulsion system. The characteristics quantities that normalize the problem are changed to reflect the fact that the Earth is, now, the central body (in contrast to the previous examples). The characteristic length is set to $L = 6878.137$ km. Then, the characteristic time can be defined as $T = (L^3/\mu_e)^{0.5} = 903.52$ s, so to make the normalized μ_e equal to 1. The initial spacecraft mass is chosen as the characteristic mass.

Initially, we set the transfer time of flight to 2 d, which corresponds to a trajectory with 30 revolutions. Again, we solve this numerical example with different number of trajectory arcs. The converged results are listed in Table 7. Observations from sect. 4.2 are still valid in Table 7. That is, the symplectic method may reach relatively high accuracy with 3 or 4 arcs, and the accuracy of the symplectic method can further improve when more arcs are added. As a reference, the optimal solution for 100 intervals is portrayed through Figures 10–12.

Next, we consider a longer time of flight, i.e., 15 d which corresponds to a 228-revolutions trajectory. Solving low-thrust trajectory with such a large number of revolutions is considered a challenging problem. The optimal solution can be successfully obtained by the symplectic method, when a good initial guess is supplied. The converged optimal solutions are listed in Table 8. The CPU cost could be further reduced by improving code quality. For completeness, the trajectory, the costate variables and the thrust profile are depicted in Figures 13–15. This example supplies preliminary

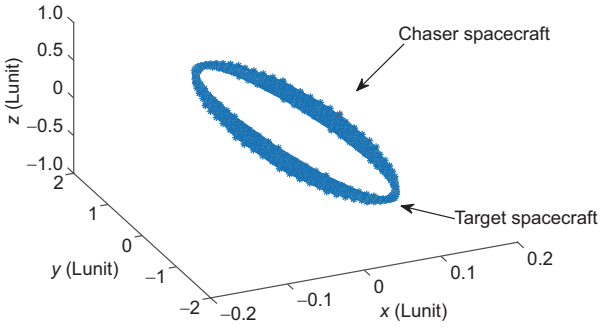

Figure 10 (Color online) Low-thrust trajectory with 30 revolutions for the chaser spacecraft.

Table 7 Converged solutions from the symplectic method with different number of arcs

Number of arcs	Number of iterations	Computational time (s)	Final mass (kg)
50	3	2.172762	1499.764308
80	3	6.768178	1499.761093
100	3	11.512130	1499.761217
200	3	70.278182	1499.761266
300	3	216.140373	1499.761265

Table 8 Converged solutions from the symplectic method with different number of arcs

Number of arcs	Number of iterations	Computational time (s)	Final mass (kg)
600	3	1719.7184	1499.943624
800	3	3476.3645	1499.943511

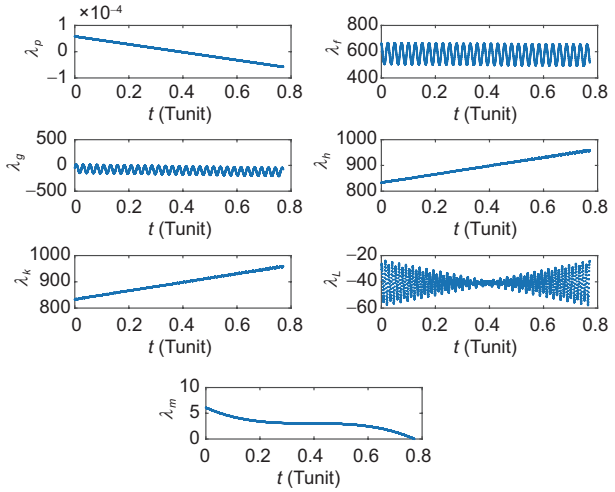

Figure 11 (Color online) Costate variables time histories of the chaser spacecraft along a 30 revolutions trajectory.

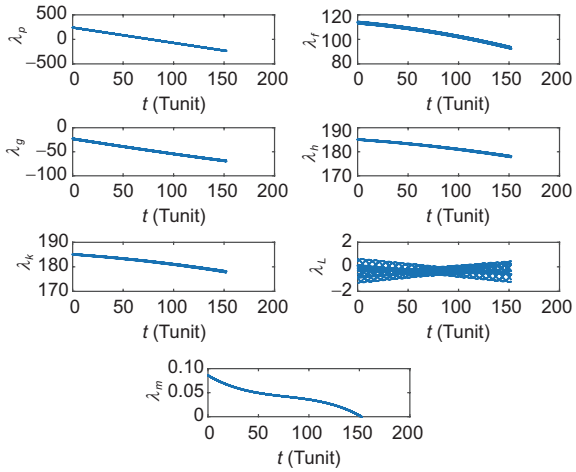

Figure 14 (Color online) Low-thrust trajectory of the chaser spacecraft with 228 revolutions.

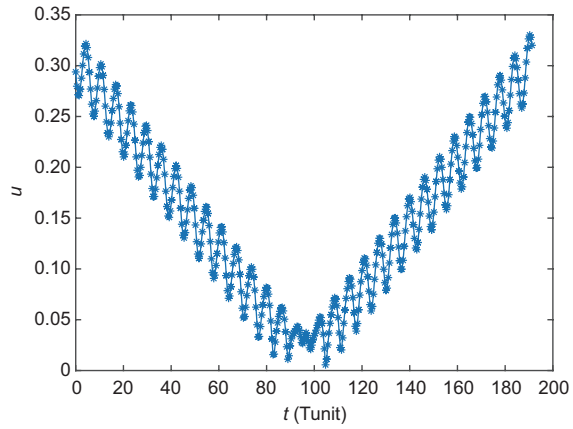

Figure 12 (Color online) Optimal thrust profile of the chaser spacecraft along a 30 revolutions trajectory.

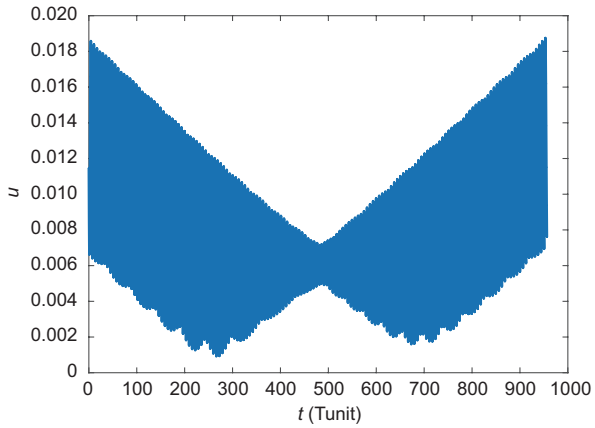

Figure 15 (Color online) Low-thrust trajectory of the chaser spacecraft with 228 revolutions.

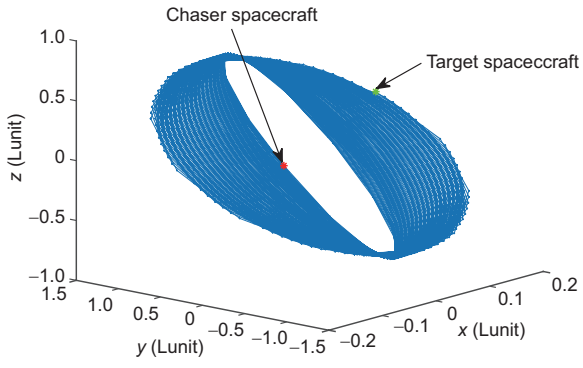

Figure 13 (Color online) Low-thrust trajectory of the chaser spacecraft with 228 revolutions.

evidence that the proposed symplectic method is a promising tool to optimize low-thrust transfers with a large number of revolutions.

5 Conclusion

A symplectic method is presented in this paper to optimize multi-revolution low-thrust orbit transfer trajectories. To reduce the oscillatory nature of the Cartesian coordinates along spiraling trajectories with multiple revolutions, osculating equinoctial elements are chosen to describe the motion of the spacecraft. In addition, an initial reference solution is given to accelerate the optimization process. These two techniques

may enable the symplectic method to converge rapidly, when it is applied to the optimization of orbit transfers with multiple revolutions.

A representative renhdezvous problem from the Earth to Venus is successfully solved by the proposed method. The accuracy and optimality of the symplectic method are demonstrated by comparison with known results obtained from an indirect method. In addition, the relationship between the number of intervals and the accuracy attainable with the symplectic method is discussed, and may be a reference for future research. The symplectic method is also compared to the well-known SNPOT solver. In optimizing an orbit transfer between two circular orbits, which serves as a benchmark problem, the symplectic method is significantly faster than SNOPT in terms of computational time. In addition, compared to SNOPT, the proposed method can produce a reasonable approximation of the continuous solution with fewer discretization points. Finally, low Earth orbit spacecraft rendezvous with a very large number of revolutions are successfully solved by the proposed symplectic method, within J_2-perurbed orbit dynamics. In conclusion, the symplectic methods prove valid in solving known problems and seem to behave well when applied to more complex dynamics. In future work, we envision the application of symplectic methods to optimize more complex transfers within higher fidelity environments.

This work was supported by the National Natural Science Foundation of China (Grant Nos. 11672146, 11432001), and the 2015 Chinese National Postdoctoral International Exchange Program. In addition, we are grateful to Mr. Boyang Shi for his help with computational programming and the reviewers for their detailed comments that have helped improve this manuscript.

1 Rayman M D, Chadbourne P A, Culwell J S, et al. Mission design for deep space 1: A low-thrust technology validation mission. Acta Astronaut, 1999, 45: 381–388
2 Rayman M D, Fraschetti T C, Raymond C A, et al. Coupling of system resource margins through the use of electric propulsion: Implications in preparing for the dawn mission to ceres and vesta. Acta Astronaut, 2007, 60: 930–938
3 Kuninaka H, Nishiyama K, Funakai I, et al. Asteroid rendezvous of HAYABUSA explorer using microwave discharge ion engines. In : Proceedings of the 29th International Electric Propulsion Conference. Princeton, 2005
4 Enright P J, Conway B A. Discrete approximations to optimal trajectories using direct transcription and nonlinear programming. J Guidance Control Dyn, 1992, 15: 994–1002
5 Hargraves C R, Paris S W. Direct trajectory optimization using nonlinear programming and collocation. J Guidance Control Dyn, 1987, 10: 338–342
6 Jiang F H, Baoyin H X, Li J F. Practical techniques for low-thrust trajectory optimization with homotopic approach. J Guidance Control Dyn, 2012, 35: 245–258
7 Kechichian J A. Optimal leo-geo intermediate acceleration orbit transfer. Spaceflight Mech, 1994, 1994: 885–903
8 Gao Y, Kluever C. Low-thrust interplanetary orbit transfers using hybrid trajectory optimization method with multiple shooting. In: Proceedings of the AIAA/AAS Astrodynamics Specialist Conference and Exhibit. Providence, 2004. 5088
9 Kluever C A, Pierson B L. Optimal low-thrust three-dimensional Earth-moon trajectories. J Guidance Control Dyn, 1995, 18: 830–837
10 Bertrand R, Epenoy R. New smoothing techniques for solving bang-bang optimal control problems—numerical results and statistical interpretation. Optim Control Appl Meth, 2002, 23: 171–197
11 Haberkorn T, Martinon P, Gergaud J. Low thrust minimum-fuel orbital transfer: A homotopic approach. J Guidance Control Dyn, 2004, 27: 1046–1060
12 Pan B, Lu P, Pan X, et al. Double-homotopy method for solving optimal control problems. J Guidance Control Dyn, 2016, 39: 1706–1720
13 Martinon P, Gergaud J. Using switching detection and variational equations for the shooting method. Optim Control Appl Meth, 2007, 28: 95–116
14 Jiang F H, Tang G J, Li J F. Improving low-thrust trajectory optimization by adjoint estimation with shape-based path. J Guidance Control Dyn, 2017, 40: 3282–3289
15 Yang H W, Li S X, Bai X L. Fast homotopy method for asteroid landing trajectory optimization using approximate initial costates. J Guidance Control Dyn, 2019, 42: 585–597
16 Betts J T. Very low-thrust trajectory optimization using a direct SQP method. J Comput Appl Math, 2000, 120: 27–40
17 Scheel W A, Conway B A. Optimization of very-low-thrust, many-revolution spacecraft trajectories. J Guidance Control Dyn, 1994, 17: 1185–1192
18 Yang H W, Bai X L, Baoyin H X. Rapid generation of time-optimal trajectories for asteroid landing via convex optimization. J Guidance Control Dyn, 2017, 40: 628–641
19 Tang G, Jiang F H, Li J F. Fuel-optimal low-thrust trajectory optimization using indirect method and successive convex programming. IEEE Trans Aerosp Electron Syst, 2018, 54: 2053–2066
20 Liu X, Lu P, Pan B. Survey of convex optimization for aerospace applications. Astrodyn, 2017, 1: 23–40
21 Yang H W, Bai X L, Baoyin H X. Rapid trajectory planning for asteroid landing with thrust magnitude constraint. J Guidance Control Dyn, 2017, 40: 2713–2720
22 Peng H J, Gao Q, Wu Z G, et al. Symplectic adaptive algorithm for solving nonlinear two-point boundary value problems in Astrodynamics. Celest Mech Dyn Astr, 2011, 110: 319–342
23 Zhong W X. Duality System in Applied Mechanics and Optimal Control. Boston: Springer Science & Business Media, 2006
24 Peng H J, Gao Q, Wu Z, et al. Symplectic approaches for solving two-point boundary-value problems. J Guidance Control Dyn, 2012, 35: 653–659
25 Peng H, Chen B, Wu Z. Multi-objective transfer to libration-point orbits via the mixed low-thrust and invariant-manifold approach. Nonlinear Dyn, 2014, 77: 321–338
26 Peng H, Jiang X, Chen B. Optimal nonlinear feedback control of spacecraft rendezvous with finite low thrust between libration orbits. Nonlinear Dyn, 2014, 76: 1611–1632
27 Peng H J, Li C. Bound evaluation for spacecraft swarm on libration orbits with an uncertain boundary. J Guidance Control Dyn, 2017, 40: 2690–2698
28 Li M, Peng H, Zhong W. A symplectic sequence iteration approach for nonlinear optimal control problems with state-control constraints. J Franklin Institute, 2015, 352: 2381–2406
29 Peng H J, Wang X W, Li M W, et al. An hp symplectic pseudospectral method for nonlinear optimal control. Commun Nonlinear Sci Numer Simul, 2017, 42: 623–644
30 Lantoine G, Russell R P. A hybrid differential dynamic programming algorithm for constrained optimal control problems. Part 1: Theory. J Optim Theor Appl, 2012, 154: 382–417
31 Patterson M A, Rao A V. Gpops-ii: A matlab software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming. ACM Trans Math Softw, 2014, 41: 1–37
32 Gill P E, Murray W, Saunders M A. SNOPT: An SQP algorithm for large-scale constrained optimization. SIAM Rev, 2005, 47: 99–131
33 Zhao S, Zhang J, Xiang K, et al. Target sequence optimization for multiple debris rendezvous using low thrust based on characteristics of SSO. Astrodyn, 2017, 1: 85–99