

Research Article

Multiphase Return Trajectory Optimization Based on Hybrid Algorithm

Yi Yang and Ying Nan

College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing, China

Correspondence should be addressed to Ying Nan; nanying@nuaa.edu.cn

Received 27 March 2016; Accepted 20 July 2016

Academic Editor: Juan F. San-Juan

Copyright © 2016 Y. Yang and Y. Nan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A hybrid trajectory optimization method consisting of Gauss pseudospectral method (GPM) and natural computation algorithm has been developed and utilized to solve multiphase return trajectory optimization problem, where a phase is defined as a subinterval in which the right-hand side of the differential equation is continuous. GPM converts the optimal control problem to a nonlinear programming problem (NLP), which helps to improve calculation accuracy and speed of natural computation algorithm. Through numerical simulations, it is found that the multiphase optimal control problem could be solved perfectly.

1. Introduction

Return trajectory, as an important part of satellite recovery mission, is flight trajectory of spacecraft from its working orbit to required reentry point. The purpose of return trajectory optimization is to obtain control variables of spacecraft that null the endpoint errors between the spacecraft final state vector and the target state vector while minimizing the total fuel mass consumption.

Return trajectory optimization problem, which also could be considered as orbit transfer optimization problem, is an optimal control problem in the functional space in principle. Typically, orbit transfer optimization algorithm could be divided into two types according to their finite dimensional processing method for dynamic equation: direct methods and indirect methods.

Indirect methods [1–4] transform the trajectory optimization problem into a two-point boundary value problem based on Pontryagin minimum principle and boundary condition. Indirect methods are of high solution accuracy and satisfy the optimality conditions strictly in theory, which have been widely used in orbit transfer optimization problem, especially in low-thrust, long-duration orbit transfer problem. However, indirect methods rely on correct estimation of unknown boundary condition and need good guess of initial value of consistent variable.

Direct methods [5–8] usually discretize the optimal control problem into a nonlinear programming problem. Direct methods have broader convergence domain and lower requirements for the accuracy of initial value evaluation. However, compared to indirect methods, direct methods have lower accuracy and are easy to converge to local optimal solution, which indicates that global optimization algorithm is the main direction of direct methods.

Pseudospectral method, as a well-known direct method, has been widely used to solve optimal control problem [9–12]. GPM transforms the original optimal control problem into a nonlinear programming problem (NLP) at the discrete points subject to continuous state equality and inequality constraints. Therefore, the dynamic equations can be transformed into the algebraic equations and the cost function can also be expressed as the function of the states and control. The NLP is usually solved by sequential quadratic programming method (SQP), but SQP easily falls into local optimal solution and is highly dependent on initial values, which make it hard to obtain global optimal solution.

Natural computation algorithm, which was first proposed by Nan Ying in 2013, is a global optimization algorithm; it numerically simulates numerous substance units, combined substances, and reciprocities among the substances, including the organized running and decomposing and compounding of substance units and combined substances.

TABLE 1: Initial orbit elements.

Universal time	Semimajor axis a	Eccentricity e	Inclination i	RAAN Ω	Argument of perigee ω	True anomaly f
2017/10/2 12:24:36	6778140 (m)	0.003	42°	122.722°	0°	227.298°

TABLE 2: Reference reentry point parameters.

Parameters	Reference value
Reentry geographic height, h	120 km
Reentry speed over ground, v_{2e}	7586.2 m/s
Reentry path angle, θ	-1.235°
Reentry heading angle, ψ_v	45.71°
Reentry longitude, λ	34.00°
Reentry latitude, ϕ	2.612°

The running purposes of substance units and combined substances are pursuing benefit performance indexes for themselves. Numerical simulation in [13, 14] indicates its global optimization ability, but it has shortcomings of numerous computation amounts and low computation accuracy according to its search principle.

Therefore, in this paper, we proposed a hybrid algorithm to combine the advantages of these two methods, where GPM is adopted to transform the original optimal control problem into a nonlinear programming problem (NLP), and then solve this NLP based on natural computation algorithm (NCA).

2. Problem Formulation

In this section, return trajectory mission of spacecraft and dynamic equation applied will be precisely described.

The spacecraft starts from a circular LEO with the altitude of 400 km and the deorbit window is shown in Table 1. Spacecraft begins to decelerate under the backward thrust of engine and flies to the reference reentry point accurately under the control of attack angle and slide angle. The reference reentry point parameters are summarized in Table 2.

The initial mass of the spacecraft is assumed to be 3000 kg, and the engine which is fixed on the bottom of the spacecraft has stable thrust 1173.5 N and specific impulse 300 s.

To minimize the performance index (i.e., fuel consumption):

$$J = \int_{t_0}^{t_f} m dt \quad (1)$$

the control variable $u(t) = [\alpha(t); \beta(t)]$, the state $x(t) = [v(t); \theta(t); \psi_v(t); h(t); \phi(t); \lambda(t)]$, the initial time t_0 , and the terminal time t_f have to be optimized.

The states and control variables of spacecraft are determined by the following dynamic equations:

$$\begin{aligned} \dot{r} &= v \sin \theta, \\ \dot{\phi} &= \frac{v \cos \theta \cos \psi_v}{r}, \end{aligned}$$

$$\dot{\lambda} = -\frac{v \cos \theta \sin \psi_v}{r \cos \phi},$$

$$\begin{aligned} \dot{v} &= \frac{T \cos \alpha \cos \beta}{m} + g_r \sin \theta + g_\phi \cos \theta \cos \psi_v \\ &\quad - w_e^2 r \cos \phi (\sin \phi \cos \theta \cos \psi_v - \cos \phi \sin \theta), \end{aligned}$$

$$\begin{aligned} \dot{\theta} &= \frac{T \sin \alpha}{mv} + \frac{g_r \cos \theta - g_\phi \sin \lambda \cos \psi_v}{v} + \frac{v \cos \theta}{r} \\ &\quad - 2w_e \cos \phi \sin \psi_v \end{aligned}$$

$$+ \frac{w_e^2 r}{v} \cos \phi (\cos \theta \cos \phi + \sin \phi \sin \theta \cos \psi_v),$$

$$-\dot{\psi}_v = \frac{T \cos \alpha \sin \beta}{mv \cos \theta} + \frac{g_\phi \sin \psi_v}{v \cos \theta} - \frac{v \sin \psi_v}{r} \tan \phi \cos \theta$$

$$- 2w_e (\cos \phi \tan \theta \cos \psi_v - \sin \phi)$$

$$- \frac{w_e^2 r}{v \cos \theta} \sin \phi \cos \phi \sin \psi_v,$$

(2)

where $[r; \phi; \lambda]$ denote the distance from earth's core, altitude, and longitude of the spacecraft, respectively; $[v; \theta; \psi_v]$ denote the velocity, path angle, and heading angle of the spacecraft, respectively; $[\alpha; \beta]$ are attack angle and slide angle of the spacecraft; T is the thrust, which is fixed on 1173.5 N; m denotes the mass of spacecraft; w_e is the rotational-angular velocity of the earth; and $[g_\phi; g_r]$ are the acceleration of gravity component on distance and altitude.

Also, the initial and terminal constraints have to be considered, as described in (3) and (4). Moreover, the path constraints should be satisfied as (5):

$$[v; \theta; \psi_v; h; \phi; \lambda] \Big|_{t_0} \equiv [a; e; i; \Omega; \omega; f] \Big|_{t_0}, \quad (3)$$

$$[v; \theta; \psi_v; h; \phi; \lambda] \Big|_{t_f} \equiv [v_f; \theta_f; \psi_{v_f}; h_f; \phi_f; \lambda_f], \quad (4)$$

$$\sqrt{\dot{\alpha}^2(t) + \dot{\beta}^2(t)} \leq \frac{0.3^\circ}{s}. \quad (5)$$

3. Description of Hybrid Algorithm

This hybrid algorithm fully takes advantages of GPM and NCA; it converts the original multiphase trajectory optimization problem into a discrete NLP firstly by using Gauss pseudospectral method and then solves the NLP by using NCA. GPM improves accuracy and computation speed of algorithm and NCA can obtain the global optimal solution. GPM is one fitting method which uses global interpolation

polynomial, which needs less input parameters and possesses higher accuracy compared to other methods. Compared with common SQP algorithm, NCA will not fall into local optimal solution, and, independent of initial values, numerical simulation in the next section will contrast optimization results of these two methods.

3.1. Gauss Pseudospectral Method. GPM is one fitting method which uses global interpolation polynomial, which needs less input parameters and possesses higher accuracy compared to other methods.

Since the states of return trajectory $(x(t) = [v(t); \theta(t); \psi_v(t); h(t); \phi(t); \lambda(t)])$ have different orders of magnitude, states with high order of magnitude need to be normalized as follows:

$$\begin{aligned}\bar{h} &= \frac{h}{R_e}, \\ \bar{v} &= \frac{v}{\sqrt{g_0 R_e}}, \\ \bar{t} &= \frac{t}{\sqrt{R_e/g_0}}, \\ \bar{m} &= \frac{m}{m_0}, \\ \bar{T} &= \frac{T}{m_0 g_0},\end{aligned}\quad (6)$$

where m_0 is the initial mass of the spacecraft and R_e is the radius of the earth.

Gauss pseudospectral method for solving multiphase optimal control problem is described as follows.

Firstly, transform the independent variable t to the variable $\tau \in [-1, 1]$, for every phase of the problem $p \in [1, \dots, P]$, and $(\cdot)^{(k)}$ denotes information in the k th phase:

$$t^{(p)} = \frac{t_f^{(p)} - t_0^{(p)}}{2} \tau^{(p)} + \frac{t_f^{(p)} + t_0^{(p)}}{2}. \quad (7)$$

Next, choose the Legendre-Gauss (LG) points as the collection points in each phase, which are the roots of the N th-degree Legendre polynomial $P_N(\tau)$ as follows:

$$P_N = \frac{1}{2^N N!} \frac{d^N}{d\tau^N} \left\{ [\tau^2 - 1]^N \right\}. \quad (8)$$

The whole discrete points are LG points and 2 points ($\tau_0 = -1$ and $\tau_{N+1} = 1$).

The state of spacecraft trajectory is discretized at $N + 1$ Legendre-Gauss points:

$$x^{(p)}(\tau) \approx X^{(p)}(\tau) = \sum_{i=0}^{N^{(p)}} L_i^{(p)}(\tau) X_i^{(p)}(\tau) \quad (9)$$

$$= \sum_{i=0}^{N^{(p)}} L_i^{(p)}(\tau) X_i^{(p)},$$

where $L_i^{(p)}(\tau)$ is defined as

$$L_i^{(p)}(\tau) = \prod_{j=0, j \neq i}^{N^{(p)}} \frac{\tau - \tau_j^{(p)}}{\tau_i^{(p)} - \tau_j^{(p)}}. \quad (10)$$

Differentials of state variables with respect to τ in (2) in each phase are

$$\dot{x}^{(p)}(\tau) \approx \dot{X}^{(p)}(\tau) = \sum_{i=0}^{N^{(p)}} \dot{L}_i^{(p)}(\tau) X_i^{(p)}, \quad (11)$$

$$p = 1, 2, \dots, P.$$

The derivative of each Lagrange polynomial at the LG points can be represented compactly in the form of a differentiation matrix $D \in \mathbb{R}^{N \times (N+1)}$ as

$$D_{ki}^{(p)} = \dot{L}_i^{(p)}(\tau_k) = \sum_{l=0}^N \frac{\prod_{j=0, j \neq i, l}^N (\tau_k - \tau_j)}{\prod_{j=0, j \neq i}^N (\tau_i - \tau_j)}, \quad (12)$$

where $k = 1, \dots, N$ and $i = 1, \dots, N$.

The dynamic constraint of (2) is converted into algebraic constraints in each phase $p = [1, \dots, P]$ of the problem as

$$\sum_{i=0}^N D_{ki} X_i - \frac{t_f - t_0}{2} f(X_k, U_k, \tau_k; q, t_0, t_f) = 0, \quad (13)$$

where $X_k \equiv X(\tau_k) \in \mathbb{R}^n$ and $U_k \equiv U(\tau_k) \in \mathbb{R}^m$ ($k = 1, \dots, N$). Next, define additional variable X_0 and X_f as

$$X_0 = X(\tau_0),$$

$$X_{N+1} = X_0 + \frac{t_f - t_0}{2} \sum_{k=1}^N \omega_k f(X_k, U_k, \tau_k; q, t_0, t_f). \quad (14)$$

Substituting (11) into (13), (13) will become a linear equation:

$$X_{N+1} - X_0 - \sum_{i=0}^N \sum_{k=1}^N \omega_k D_{ki} X_i = 0. \quad (15)$$

Similarly, the path constraints of (4) are discretized at LG points as

$$C_{\min} \leq C(X_k, U_k, \tau_k; t_0, q, t_f) \leq C_{\max}. \quad (16)$$

Besides, the boundary conditions of (3) are given by

$$\phi_{\min} \leq \phi(X_0, t_0, X_{N+1}, t_f) \leq \phi_{\max}. \quad (17)$$

Finally, the linking constraints are mapped using the values at the terminal and start points, respectively, of the phase pairs $(p_l, p_r) \in [1, \dots, P]$ ($l, r = [1, \dots, P]$) as

$$L_{\min}^{(s)} \leq L^{(s)}(X_{N+1}^{(p_l^s)}, t_f^{(p_l^s)}; q^{(p_l^s)}, X_0^{(p_r^s)}, t_0^{(p_r^s)}; q^{(p_r^s)}) \leq L_{\max}^{(s)}. \quad (18)$$

As the discretization process discussed above, the original multiphase continuous optimal control problem is transformed into a discrete NLP successfully, which could be solved by natural computation algorithm (NCA) later.

3.2. Natural Computation Algorithm (NCA). Natural computation algorithm is a global optimization method, which simulates running rules of all radical element units (REUs) and their combinations. In NCA, all units and combinations are pursuing for optimal performance index. Original NCA adopts average discrete method, which may lead to some shortcomings such as low accuracy and long computation time because equidistance interpolating causes fluctuations near initial and terminal points. To overcome this issue, GPM is used here to convert continuous trajectory optimization problem to NLP, which could be solved by NCA with higher accuracy and computation speed.

Computational steps of NCA are described as follows.

Initial Setting

Setting 1. Structure a $N_s = 4$ -dimensional physical space, which consists of time dimension and 3-space dimension. Define the sizes of 3-space dimension: $i_{SX_{\max}}$, $i_{SY_{\max}}$, and $i_{SZ_{\max}}$. Dimension of control and strategy's storage space is $i_{Cl_{\max}}$, $l = 1, 2, \dots, l_{\max}$, where l_{\max} is the number of control variables.

Setting 2. Transform the trajectory optimization problem into NLP by using GPM, which was described above in Section 3.1. The number of calculation steps is N , which is the sum of each phase's discrete points ($N = \sum_{p=1}^P N^p$, where N^p equals degree of Legendre polynomial in p th phase).

The detailed calculation methods for each step of natural computation are described as follows.

Step 1. Calculate the running states from time $\tau_{Y_{\min}}$ to $\tau_{Y+1, \min}$ ($\tau_{Y+1, \min} = \tau_{Y_{\max}}$), where $Y = 1$. Eliminate the combinations with poor development according to the evolution and elimination law (discussed in below). The first level's optimal control $\vec{U}^*(1) = [\vec{u}^*(1), \vec{\mu}^*(1)]$ would be obtained as follows.

Retain the suitable combinations on the basis of (23) which is posed below and also the optimal control:

$$\begin{aligned} & \vec{u}^* [\vec{X}^*(1)], \\ & J_1^* [\vec{X}^*(1), 1]. \end{aligned} \quad (19)$$

Save the states of REUs and combinations in storage $Ps_i[1][i_{SX}][i_{SY}][i_{SZ}][0]$, save the performance index in storage $Ps_i[1][i_{SX}][i_{SY}][i_{SZ}][1]$, save the interrelation function in storage $Ps_i[1][i_{SX}][i_{SY}][i_{SZ}][2]$, save the optimal control in storage $Us_i[1][l_{C, \max}]$, and save the history information $i_{Cr_{\text{Opt}}}$ in storage $\dagger_i[1][l_{C, \max}]$.

Step κ . Calculate the running states from time $\tau_{Y_{\min}}$ to $\tau_{Y+1, \min}$ ($\tau_{Y+1, \min} = \tau_{Y_{\max}}$), where $Y = \kappa$. Eliminate the combinations with poor development according to evolution and elimination law. The κ th level's optimal control $\vec{U}^*(\kappa) = [\vec{u}^*(\kappa), \vec{\mu}^*(\kappa)]$ would be obtained as follows:

$$\begin{aligned} & J_\kappa^* [\vec{X}^*(\kappa), \kappa] \\ & = \min_{\vec{u}_{k,i}(\kappa-1) \rightarrow \vec{u}_{k,i}(\kappa)} \{L_{k,i,j} [\vec{X}(\kappa), \vec{u}_{k,i}(\kappa), \kappa] \\ & + J_{\kappa-1}^* [\vec{x}^*(\kappa-1), \kappa-1]\}, \\ & \vec{x}_{k,i}(\kappa) = \vec{f}_{k,i} [\vec{X}(\kappa-1), \vec{u}_{k,i}(\kappa-1), \kappa-1]. \end{aligned} \quad (20)$$

Retain the suitable combinations on the basis of (23) and also the optimal control:

$$\begin{aligned} & \vec{u}^* [\vec{X}^*(1)], \\ & J_1^* [\vec{X}^*(1), 1]. \end{aligned} \quad (21)$$

Save the states of REUs and combinations in storage $Ps_i[\kappa][i_{SX}][i_{SY}][i_{SZ}][0]$, save the performance index in storage $Ps_i[\kappa][i_{SX}][i_{SY}][i_{SZ}][1]$, save the interrelation function in storage $Ps_i[\kappa][i_{SX}][i_{SY}][i_{SZ}][2]$, save the optimal control in storage $Us_i[\kappa][l_{C, \max}]$, and save the history information $i_{Cr_{\text{Opt}}}$ in storage $\dagger_i[\kappa][l_{C, \max}]$.

Step N . Calculate the running states from time $\tau_{Y_{\min}}$ to $\tau_{Y+1, \min}$ ($\tau_{Y+1, \min} = \tau_{Y_{\max}}$), where $Y = N$. Eliminate the combinations with poor development according to (21). The N th level's optimal control $\vec{U}^*(N-1) = [\vec{u}^*(N-1), \vec{\mu}^*(N-1)]$ would be obtained as follows:

$$\begin{aligned} & J_N^* [\vec{X}^*(N), N] \\ & = \min_{\vec{u}_{k,i}(N-1) \rightarrow \vec{u}_{k,i}(N)} \{L_{k,i,j} [\vec{X}(N), \vec{u}_{k,i}(N), N] \\ & + J_{N-1}^* [\vec{x}^*(N-1), N-1]\}, \end{aligned} \quad (22)$$

$$\vec{x}_{k,i}(N) = \vec{f}_{k,i} [\vec{X}(N-1), \vec{u}_{k,i}(N-1), N-1].$$

The final states have to satisfy the terminal constraints (17). Similar to Step κ , save the calculation data in storage corresponding to N .

After the calculations, all the pieces of information for each step, including the state variables, control variables, and performance indexes, have been saved successfully and could be read freely. The final optimal combination and its history are the optimal control plan.

3.2.1. The Evolution and Elimination Law

(1) Survival of the Fittest. By the end of each time interval $[\tau_{Y_{\min}}, \tau_{Y+1, \min}]$, numerous combinations' running states are generated in the physical space (each REU generates $Q_{k,i}^2$ states at the next time point, and each combination generates $Q_{\alpha,\delta}^2$ states). In the natural computation algorithm, all REUs and combinations need to follow the law of nature, which means that the survival space is finite and the total number of REUs and combinations cannot exceed the size of physical space. In each space unit, there exists only one fittest REU, and in the physical space, the number of combinations is also restricted.

Suppose that \wp_{Agv} is the capacity of one space unit; for all \wp combinations in the physical space, only λ combinations can survive:

$$\lambda = \begin{cases} 0, & \text{if } \wp = 0 \\ 1, & \text{if } 1 \leq \wp < \wp_{\text{Agv}} \\ \xi, & \text{if } \wp_{\text{Agv}} \cdot \xi \leq \wp < \wp_{\text{Agv}} \cdot \xi + \wp_{\text{Agv}} - 1. \end{cases} \quad (23)$$

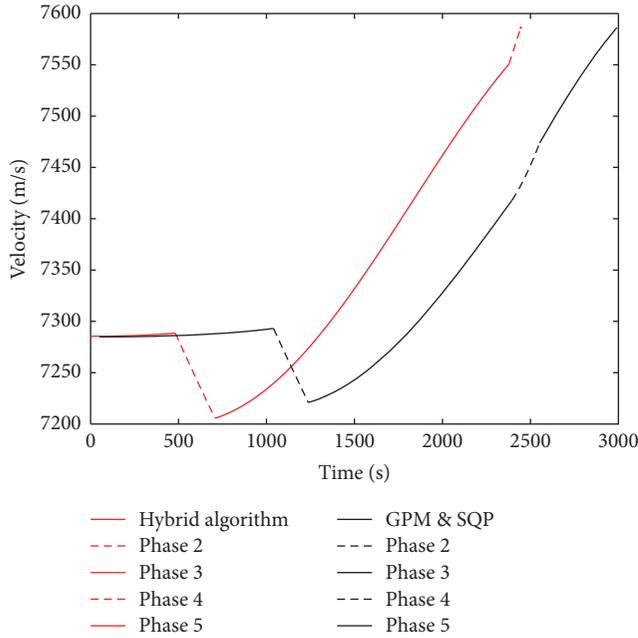


FIGURE 1: Time history of velocity of spacecraft in return trajectory.

4. Numerical Simulation and Analysis

Five phases are assumed for the whole return trajectory, and [OFF, ON, OFF, ON, OFF] are engine work states corresponding to five phases separately. The simulation results of optimal return trajectory are shown in Figures 1–12, the red lines are optimal trajectory obtained by the hybrid algorithm, and the black lines are optimal trajectory obtained by GPM and SQP.

From the simulation results shown in Figures 1–12, the hybrid algorithm gets a better return trajectory with lower fuel consumption compared to SQP algorithm, which indicates that NCA has a better ability for global optimization.

The optimal return trajectory can be described as follows: at the beginning of the second phase, the attack angle is almost 180° , which means that the thrust has been turned to the same direction as that of the velocity; then the engine works to decline the velocity of spacecraft until the end of second phase; in the third phase, the spacecraft flies without power in a transition track; the attack angle and slide angle are adjusted to optimal values; and, in fourth phase, the engine works to accelerate reaching the reentry point.

The final optimal results are as follows:

- (1) The maximum final mass is 2885.23 kg, which means that the performance index (fuel consumption) is 114.77 kg.
- (2) The terminal state of flight trajectory is $[v(t); \theta(t); \psi_v(t); h(t); \phi(t); \lambda(t)] = [7586.20, -1.2351, 314.3132, 119992.011, 2.5951, 43.0032]$.
- (3) The state errors between terminal state of optimal trajectory and reference reentry point state are listed in Table 3, which satisfy required precision.

TABLE 3: Terminal state errors.

Trajectory state	Errors
V (m/s)	0.0107501755680642
θ (rad)	$-7.66425969788223e - 06$
ψ_v (rad)	$-5.64008839631924e - 07$
h (m)	-8.8399759697844
ϕ (rad)	$3.40221324294371e - 06$
λ (rad)	$4.11743958561583e - 06$

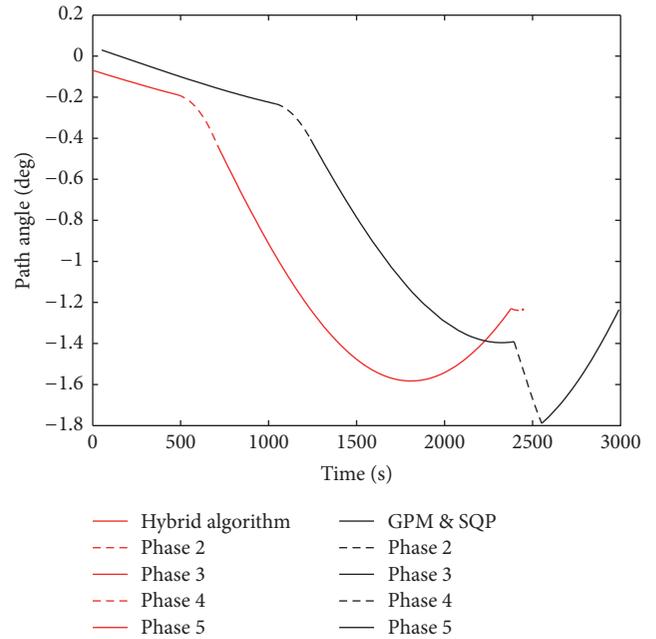


FIGURE 2: Time history of path angle of spacecraft in return trajectory.

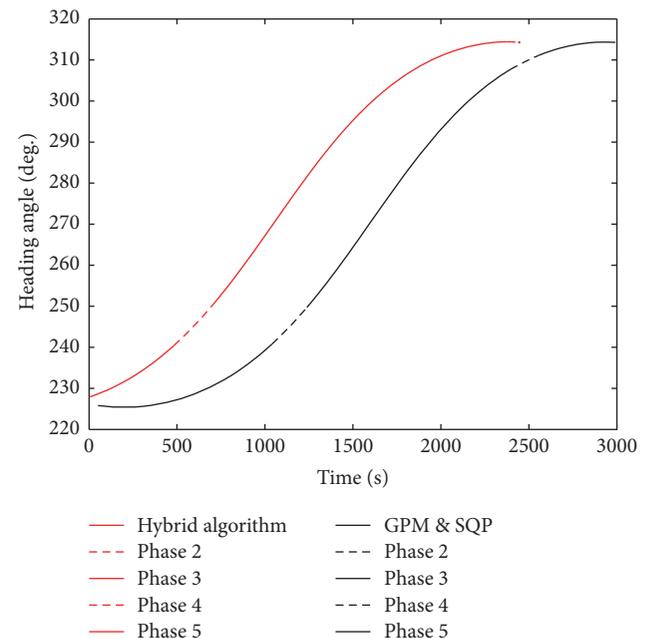


FIGURE 3: Time history of heading angle of spacecraft in return trajectory.

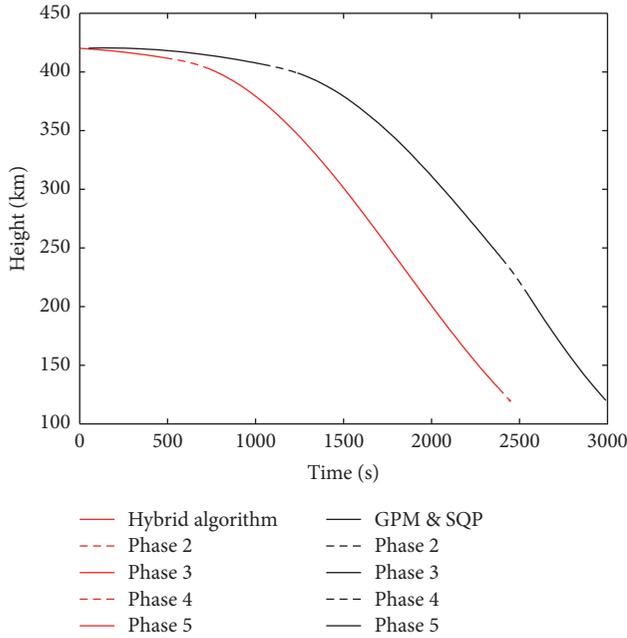


FIGURE 4: Time history of height of spacecraft in return trajectory.

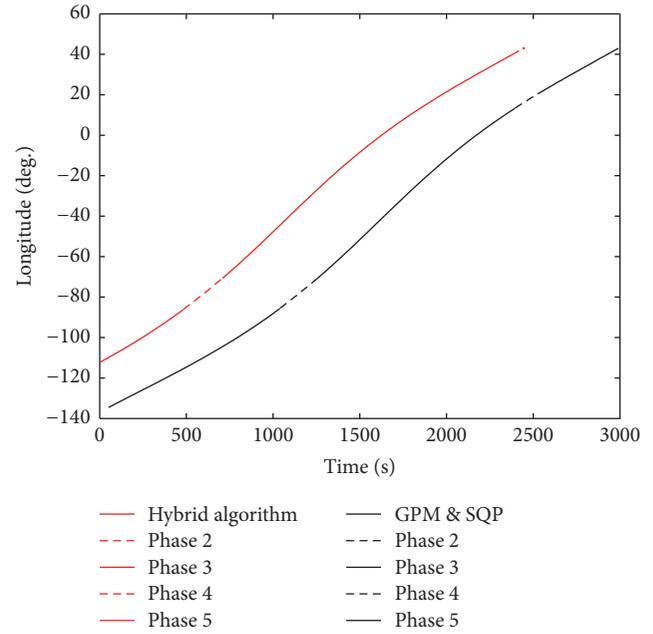


FIGURE 6: Time history of longitude of spacecraft in return trajectory.

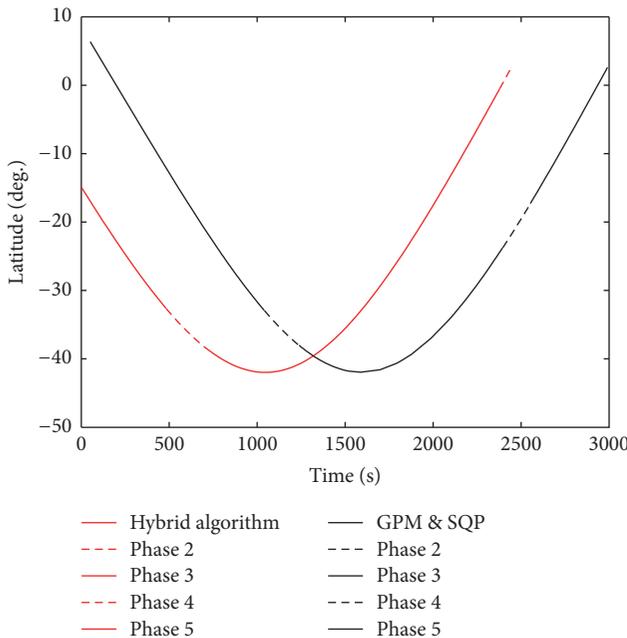


FIGURE 5: Time history of latitude of spacecraft in return trajectory.

5. Conclusion

We have proposed a novel hybrid algorithm, consisting of Gauss pseudospectral method and natural computation algorithm, to solve the multiphase optimal control problem. First, Gauss pseudospectral method (GPM) is used to discretize the original optimal control problem to NLP; and then, natural computation algorithm is used to search the optimal solution of NLP. GPM is characterized as high-grade precision fitting method while NCA is capable of searching the optimal

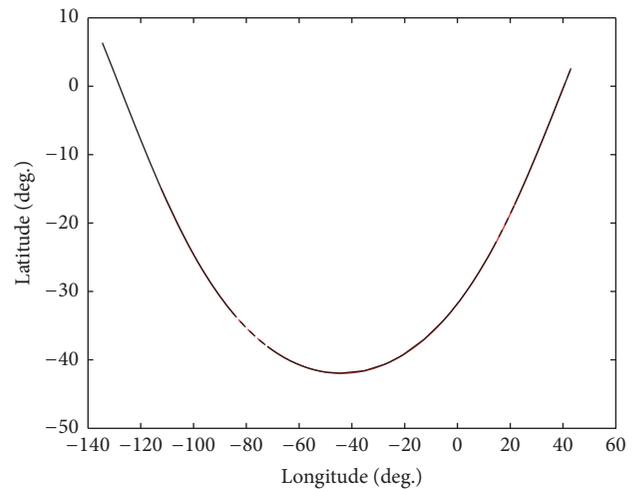


FIGURE 7: Satellite ground track of the return trajectory.

solution globally. We take advantages of both GPM and NCA in this hybrid algorithm. To verify the effectiveness, a multiphase return trajectory design has been illustrated, where the fuel consumption is minimized; meanwhile all the terminal constraints and path constraints are satisfied.

As a new global optimization algorithm, NCA has some pros and cons. The main drawbacks are low fitting accuracy and large computation amount. The newly proposed hybrid algorithm uses GPM to replace the average discrete method to discretize the original continuous problem, which improves both calculation accuracy and speed.

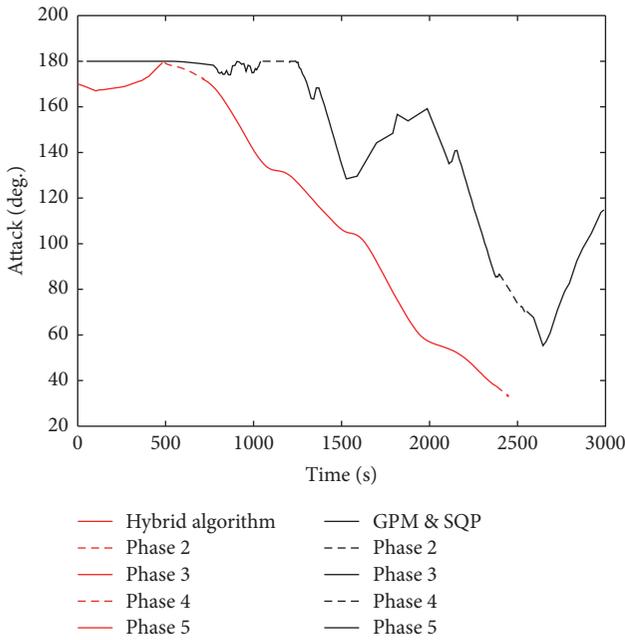


FIGURE 8: Time history of attack angle α of spacecraft in return trajectory.

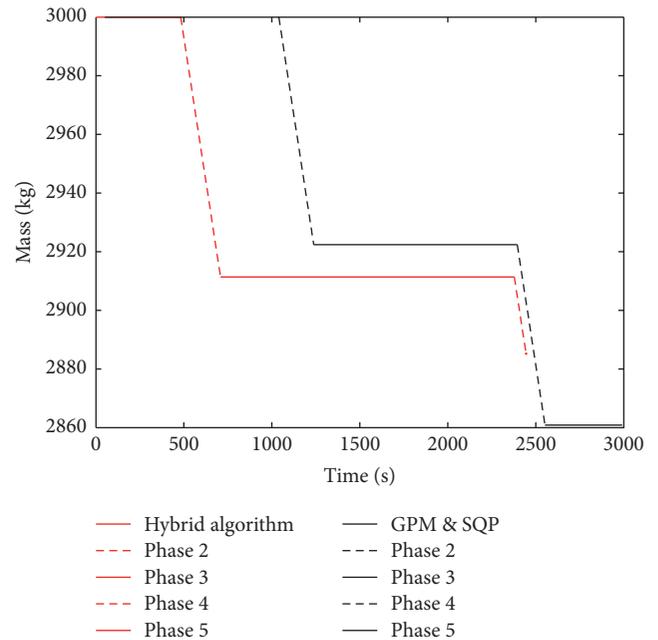


FIGURE 10: Time history of mass of spacecraft in return trajectory.

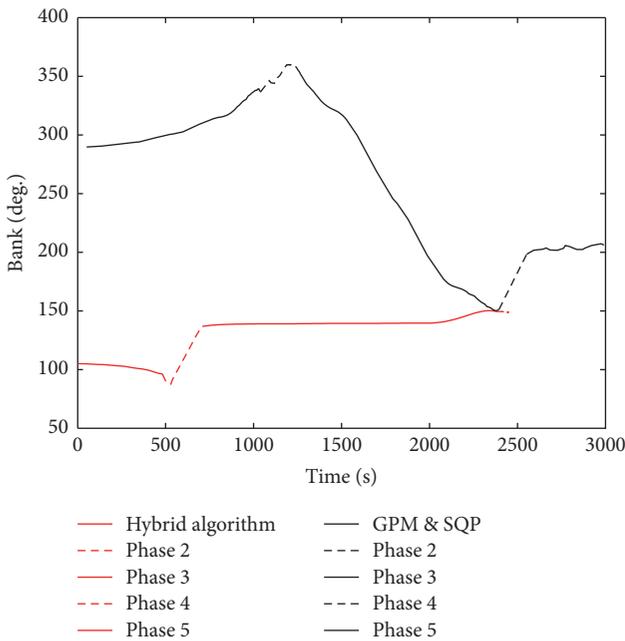


FIGURE 9: Time history of slide angle β trajectory of spacecraft in return trajectory.

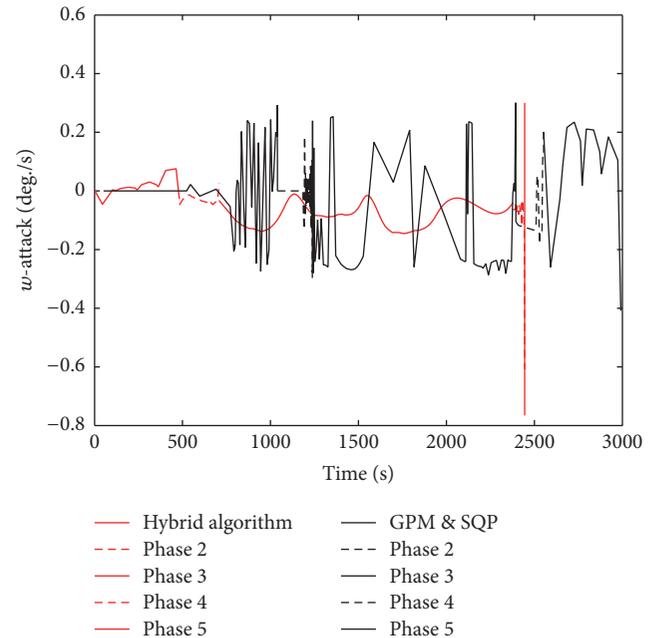


FIGURE 11: Time history of attack angle rate $\dot{\alpha}$ of spacecraft in return trajectory.

However, there is still a lot of work that can be done to improve the results. In this paper, we set the whole trajectory as a REU; as a result, the length of the REU is the sum of five phases' discrete points, which may cause dimension disaster and large calculation amount. The possible solution is to set every phase as a REU and use combination of REUs to

represent the whole trajectory. We believe that it will reduce the dimension of calculation greatly.

Competing Interests

The authors declare that they have no competing interests.

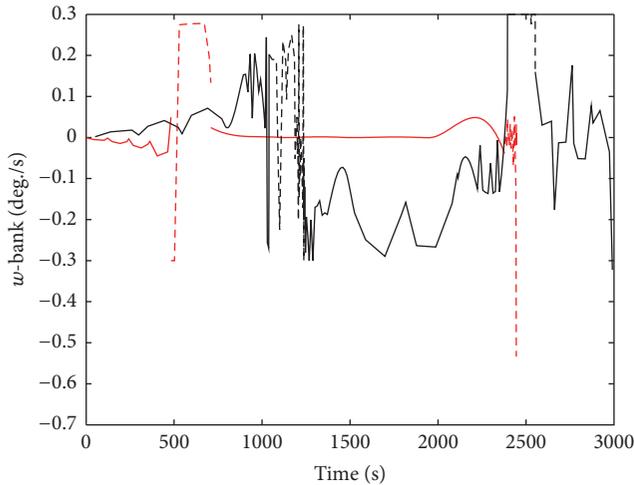


FIGURE 12: Time history of slide angle rate $\dot{\beta}$ of spacecraft in return trajectory.

Acknowledgments

This work was supported by funding of Jiangsu Innovation Program for Graduate Education, KYLX15_0319, “the Fundamental Research Funds for the Central Universities.”

References

- [1] T. Goodson, *Fuel-optimal control and guidance for low-and medium-thrust orbit transfer [Ph.D. thesis]*, Georgia Institute of Technology, Atlanta, Ga, USA, 1995.
- [2] C.-H. Chuang and J. L. Spreyer, “Periodic optimal hypersonic scramjet cruise,” *Optimal Control Applications and Methods*, vol. 8, no. 3, pp. 231–242, 1987.
- [3] C. H. Chuang, T. Goodson, and L. A. Ledsinger, “The second variation and neighboring optimal feedback guidance for multiple burn orbit transfers,” in *Proceedings of the 1995 AIAA Conference on Guidance, Navigation, and Control*, pp. 1689–1699, AIAA, Baltimore, Md, USA, 1995.
- [4] R. Armellin and F. Topputo, “A sixth-order accurate scheme for solving two-point boundary value problems in astrodynamics,” *Celestial Mechanics & Dynamical Astronomy*, vol. 96, no. 3-4, pp. 289–309, 2006.
- [5] P. N. Desai and B. A. Conway, “A two-timescale discretization scheme for collocation,” *Advances in the Astronautical Sciences*, vol. 119, no. 2, pp. 2053–2063, 2005.
- [6] J. T. Betts, “Very low-thrust trajectory optimization using a direct SQP method,” *Journal of Computational and Applied Mathematics*, vol. 120, no. 1-2, pp. 27–40, 2000.
- [7] A. L. Herman and D. B. Spencer, “Optimal, low-thrust earth-orbit transfers using higher-order collocation methods,” *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 40–47, 2002.
- [8] V. Coverstone-Carroll, J. W. Hartmann, and W. J. Mason, “Optimal multi-objective low-thrust spacecraft trajectories,” *Computer Methods in Applied Mechanics and Engineering*, vol. 186, no. 2–4, pp. 387–402, 2000.
- [9] R. Tawfiqur, H. Zhou, Y. Sheng, Y. Yamin, and K. Zhang, “Trajectory optimization of hypersonic vehicle using gauss pseudospectral method,” *Applied Mechanics and Materials*, vol. 110, no. 1, pp. 5232–5239, 2011.
- [10] G. T. Huntington and A. V. Rao, “Optimal configuration of spacecraft formations via a gauss pseudospectral method,” *Advances in the Astronautical Sciences*, vol. 120, no. 1, pp. 33–50, 2005.
- [11] Y. Xuan, W. Zhang, and Y. Zhang, “Pseudospectral method based trajectory optimization and fairing rejection time analysis of solid launch vehicle,” *Science in China, Series E: Technological Sciences*, vol. 52, no. 11, pp. 3198–3206, 2009.
- [12] W. Jianping and T. Hasegawa, “Numerical simulation on compressible turbulence by spectral method,” *Acta Mechanica Sinica*, vol. 14, no. 3, pp. 205–207, 1998.
- [13] N. Ying, C. Shilu, and D. Quanxin, “Global integrative trajectory optimization for multitudinous flight vehicles based on natural running algorithm,” *Science China Technological Sciences*, vol. 43, pp. 636–659, 2013 (Chinese).
- [14] Y. Yang, Y. Nan, and H. Chen, “New theory and algorithm of optimal control based on natural computation,” in *Proceedings of the 11th International Conference on Natural Computation (ICNC '15)*, pp. 492–501, Anaheim, Calif, USA, 2015.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

