Real-Time Optimal Control for Variable-Specific-Impulse Low-Thrust Rendezvous via Deep Neural Networks

LIU Yuhang, *YANG Hongwei**

College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, P. R. China

(Received 15 January 2023; revised 7 June 2023; accepted 5 October 2023)

Abstract: This paper presents a real-time control method based on deep neural networks (DNNs) for the fuel-optimal rendezvous problem. A backward generation optimal examples method for the fuel-optimal rendezvous problem is proposed, which iterates through the dichotomy method based on the existing backward generation idea while satisfying the two integration cutoff conditions of the backward integration. We construct a DNNs structure suitable for the variable-specific-impulse model and divide the output control of networks into the thrust output and the specific impulse output. For the specific impulse output, a method is proposed that learns the optimal specific impulse first and then limits it according to its actual upper and lower limits. We propose the enhanced fault-tolerant deep neural networks (EFT-DNNs) to improve the robustness when approaching rendezvous. The effectiveness and efficiency of the proposed method are verified by simulations of the Earth-Apophis asteroid and Earth-Mars missions.

Key words: trajectory optimization; variable specific impulse; fuel-optimal control; indirect method; deep neural networks(DNNs)

CLC number: V412.4 **Document code:** A **Article ID:** 1005-1120(2023)05-0578-17

0 Introduction

Electric propulsion (EP) is a quite promising propulsion method for deep space exploration missions, and there are already many examples of successful use of EP[1-4]. The thrust of EP is very small, usually less than 1 N, and it needs to be turned on continuously to get the work done[5]. The dynamic continuous low-thrust control poses great difficulties for trajectory optimization. Traditional methods for solving the low-thrust trajectory optimization problems are direct methods[6-7] and indirect methods[8-9], and the combination of the above two[10]. The direct method is to solve the original problem by transforming it into nonlinear programming (NLP) with discrete series, while the indirect method is to transform optimal control problems (OCPs) into a two-point boundary value problem (TPBVP) by constructing co-states. Deep space exploration missions have a long period and long flight distance, with non-negligible communication time delay[11]. Considering the influence of uncertainty and other complex factors, the spacecraft needs to have the capability of autonomous trajectory optimization and real-time control. However, due to the limitation of onboard computing capability, the direct and indirect methods are computationally intensive and cannot meet the requirements of onboard real-time control.

With the progress of artificial intelligence technology, especially machine learning (ML), in the field of spaceflight is gradually increasing, which provides the possibility of real-time control onboard[12]. Deep learning and reinforcement learning are currently the most popular ML techniques in the field of deep space exploration trajectory design and optimization. Deep learning is a kind of supervised learning. In the existing research, the optimal trajec-

*Corresponding author, E-mail address: hongwei.yang@nuaa.edu.cn.

How to cite this article: LIU Yuhang, YANG Hongwei. Real-time optimal control for variable-specific-impulse low-thrust rendezvous via deep neural networks[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2023, 40(5): 578-594.

http://dx.doi.org/10.16356/j.1005-1120.2023.05.007

tory information, including the initial values of co-states, fuel consumption, transfer time, thrust, etc., is often obtained by methods such as indirect methods, and subsequently, DNNs are trained to learn the mapping relationship between inputs and outputs. Deep learning enables fast estimation of transfer costs[13-14], fast optimization of trajectories and real-time control[15-20]. For the problem of rapid estimation of low-thrust transfer costs, Zhu et al.[13] achieved rapid assessment of low-thrust transfer accessibility and optimal fuel consumption prediction by a multilayer perceptron (MLP) in the context of the seventh edition of the Global Trajectory Optimization Competition (GTOC7); Li et al.[14] used neural networks to achieve optimal time prediction; Viavattene et al.[15] applied neural networks to the rapid estimation of transfer times and costs for multi-target asteroid missions. For the difficulty of guessing initial values of co-states, Yin et al.[16] and Zhao et al.[17] proposed initial guessing methods based on neural networks for the interplanetary transfer and asteroid landing problems, respectively. However, for the real-time control, a more accurate way than guessing the co-states initial value is to use neural networks to complete the state-control mapping[17]. Cheng et al.[18] proposed a multiscale-DNN to achieve high accuracy transfer, and this network is for real-time control of the thrust direction angle. Li et al.[14] studied the time-optimal real-time control problem based on neural networks; Izzo et al.[19] solved the Earth-Venus fuel-optimal transfer problem and compared control effects of value function and policy function networks. Unlike deep learning, reinforcement learning is performed by discretizing control into a finite number of actions and later modeling the OCPs as a Markov decision process (MDP). Optimal actions are explored through the payoff function, which leads to real-time optimal control[20]. However, reinforcement learning is good for simpler OCPs, but can hardly solve complex strongly nonlinear problems. In addition, increasing the number of control actions to obtain a more continuous control also makes the problem more difficult to solve. In this paper, we choose a deep learning approach to achieve OCPs of the spacecraft.

One of the important things of using deep learning is how to get the optimal dataset efficiently. Datasets are often generated by solving traditional low-thrust trajectory optimization problems, most commonly using indirect methods to generate datasets. However, solving low-thrust problems one by one is very time-consuming, and it is worthwhile to investigate how to improve the dataset generation rate. Considering the small difference between initial values of co-states of adjacent trajectories, Yin et al.[16] proposed a method based on the fast generation of optimal nominal trajectories, which used initial values of co-states of nominal trajectories as co-states guesses of new trajectories. Liu et al.[21] proposed a data generation algorithm based on the optimal trajectory continuation method to improve the success rate of initial value guessing. However, these two methods for fast dataset generation are still essentially guessing initial values of co-states and then solving the TPBVP. Izzo et al.[19] proposed a method to satisfy the shooting equation for backward integration, which they called "the backward generation of optimal examples", and successfully applied it to the fuel-optimal transfer problem. Since this method changes the solution time of the TPBVP into one backward integration time, the solution speed is greatly improved. Izzo et.al.[22] applied this method to the constant acceleration time-optimal control problem. However, the shooting equation for the fuel-optimal rendezvous problem is different from both the fuel-optimal transfer and the constant acceleration time-optimal problem, so it cannot be applied directly. Therefore, this paper proposes a backward generation method of optimal examples for the fuel-optimal rendezvous problem.

At present, deep learning methods used in real-time OCPs all aim at the fixed-specific-impulse problems[13-14,18-19]. In fact, the specific impulse of EP engines, including magnetic plasma rockets[23], Hall effect[4], and ion EP[1,3], is variable. The variable-specific-impulse electric propulsion model is more in line with the engineering reality. In this paper, we consider the specific impulse as a controlled quantity that can be actively varied in the interval, where the engine input power is influenced by the

distance of the spacecraft from the sun[24-25]. First, for the optimal trajectory generation, the introduction of variable-specific-impulse will make the trajectory optimization problem more complicated to solve. Second, when using DNNs for optimal control, it is worthwhile to find out how to perform the accurate control of the variable-specific-impulse. In the related literature[14,18-19,22] the control quantity of the deep neural network controller is one or two kinds of thrust direction and thrust size. The control quantity under the variable ratio impulse model is increased on this basis for the control of the ratio impulse, and the more the number of control quantities, the greater the possibility of error. How to improve and ensure the control accuracy is also worth considering. Based on the above research status, this paper studies the real-time OCPs of variable-specific-impulse low-thrust rendezvous based on DNNs.

The contributions and advantages of this paper are mainly in the following three aspects. Firstly, to address the problem that the backward optimal trajectory generation method in existing studies[19,22] cannot be used for fuel-optimal rendezvous trajectories, an backward generation optimal examples method applicable to the fuel-optimal rendezvous problem is proposed to improve the efficiency of data set generation. Secondly, since existing DNN-based research[13-14,18-19] is only for the fixed-specific impulse problem, not for the variable-specific-impulse problem, this paper proposes a DNN-based real-time control method for variable-specific-impulse, and constructs a real-time optimal control deep neural networks (RTOC-DNNs) to control the thrust and specific-impulse. For the thrust control, the Monte Carlo simulation accuracy of thrust learning in Cartesian and orbital coordinate systems is compared. For the specific impulse control, DNNs are proposed to learn the optimal specific impulse, and then the output is restricted by the upper and the lower bounds of the specific impulse. Finally, EFT-DNNs are proposed for enhancing the control robustness and rendezvous accuracy of the spacecraft in the second half of the flight, and the effectiveness of the EFT-DNNs are verified by simulations.

The structure of this paper is as follows. Section 1 introduces the variable-specific-impulse low-thrust model along with the fuel-optimal rendezvous problems and the optimality conditions are derived using the indirect method. Section 2 presents the backward generation for the fuel-optimal control rendezvous problem. Section 3 presents the network structure. In Section 4, nominal trajectories and data set generation are presented, followed by the RTOC-DNN and the EFT-DNN parameters selection and construction of the networks. In Section 5, the proposed methods are applied to the Earth-Apophis asteroid and Earth-Mars rendezvous missions, and the effects of DNNs are verified using Monte Carlo simulations. Section 6 concludes the paper.

1 Low-Thrust Trajectory Optimization

1.1 Dynamical model

We consider a spacecraft with a variable-specific-impulse EP engine, and this spacecraft moves in the gravity of the Sun. In the Cartesian coordinate system, the dynamical equations with the spacecraft of mass m can be expressed as

$$\dot{\boldsymbol{r}} = \boldsymbol{v} \tag{1}$$

$$\dot{\boldsymbol{v}} = -\frac{\mu}{r^3}\boldsymbol{r} + \frac{T_{\max} u}{m}\boldsymbol{\alpha} \tag{2}$$

$$\dot{m} = -\frac{T_{\max} u}{I_{\mathrm{sp}} g_0} \tag{3}$$

where $\boldsymbol{r} = (r_x, r_y, r_z)^{\mathrm{T}}$ and $\boldsymbol{v} = (v_x, v_y, v_z)^{\mathrm{T}}$ are the position and the velocity vectors of the spacecraft, respectively; the spacecraft-sun distance is expressed as $r = \| \boldsymbol{r} \|$; $T_{\max}$ and $\boldsymbol{\alpha}$ denote the maximum thrust and the unit direction vector of the thruster, respectively. The power throttle level is denoted as u, and $u \in [0, 1]$. $u = 0$ and $u = 1$ mean that the thruster is in the off and fully on state. g_0 and μ are the acceleration at ground level and the solar gravitational constant, respectively, and they take values of $g_0 = 9.806\ 65\ \mathrm{m/s^2}$ and $\mu = 1.327\ 124\ 400\ 18 \times 10^{11}\ \mathrm{km^3/s^2}$. The specific impulse of the engine is expressed as

$$I_{\mathrm{sp\text{-}min}} \leqslant I_{\mathrm{sp}} \leqslant I_{\mathrm{sp\text{-}max}} \tag{4}$$

where $I_{sp\text{-}min}$ and $I_{sp\text{-}max}$ are the lower and upper limits of the specific impulse, respectively, and the specific impulse I_{sp} can change actively.

The amplitude of thrust is obtained as

$$T_{max}=\frac{2\eta P_{max}}{I_{sp}g_0} \tag{5}$$

where P_{max} is the maximum input power; and η is the engine input power utilization efficiency, which varies linearly with I_{sp}.

$$\eta=\beta_0+\beta_1 I_{sp} \tag{6}$$

where β_0 and β_1 are empirical coefficients of the effect of specific impulse on engine efficiency. Our work takes performance parameters of NASA's Evolutionary Xenon Thruster (NEXT) as a reference[26], so $\beta_0=0.2916$ and $\beta_1=0.9624\times10^{-4}\ \mathrm{s}^{-1}$.

In the actual spacecraft work, in order to ensure the normal operation of the system function, the electrical energy generated by the solar panel first supplies the equipment except the engine and then supplies the engine to produce the thrust. The output power of the solar panel is affected by the distance between the spacecraft and the sun, and the specific relationship is as[27]

$$P_{SA}=\frac{P_{AU}}{r^2}\left(\frac{d_1+d_2r^{-1}+d_3r^{-2}}{1+d_4r+d_5r^2}\right) \tag{7}$$

where P_{SA} is the solar panel output power; P_{AU} the solar panel output power at 1 AU distance; the part in parentheses represents the empirical value of the solar panel efficiency changing with the instrumental day distance, where $d_1,\cdots,d_5$ are the empirical coefficients. Further, the maximum input power of the engine P_{max} can be listed[27]

$$P_{max}=\begin{cases}\delta P_{P\text{-}max} & P_{SA}>P_{P\text{-}max}+P_L\\ \delta(P_{SA}-P_L) & P_{SA}\leqslant P_{P\text{-}max}+P_L\end{cases} \tag{8}$$

where $P_{P\text{-}max}$ and P_L are the maximum input power of the engine power processor and the total power required by systems other than the engine system; respectively. δ is the duty cycle, which represents the power conversion efficiency.

1.2 Indirect method for low-thrust trajectory optimization

We consider a fixed-time interplanetary rendezvous problem with the performance metric of the fuel-optimal control problem

$$J=\lambda_0\int_{t_0}^{t_f}\left[\frac{2\eta P_{max}u}{I_{sp}{}^2g_0{}^2}-\varepsilon\ln(u-u^2)\right]\mathrm{d}t \tag{9}$$

where t_0, t_f and ε are the departure moment, the completion rendezvous moment, and parameter of homotopy, respectively.

In order to achieve the fuel-optimal control, it is necessary to minimize J. The functional J is chosen based on the work of Chi et al.[24] In their work, the decoupling of the control quantity u and the control quantity specific impulse I_{sp} is completed, which makes the problem easy to solve. The indirect method for solving OCPs introduces co-states with no specific physical meaning. Therefore, there is no clear guessing range in selecting initial values of co-states, which greatly affects the efficiency of the solution. In this paper, we suggest the normalization of co-states according to the co-states normalization method in Ref.[9]. The initial values of eight co-states, including λ_0, are restricted to an eight-dimensional unit hypersphere. This can narrow the guessing range and greatly improve the guessing efficiency. Since the fuel-optimal control problem is difficult to solve directly, a common approach is to introduce the homotopy parameter. First, solving the problem at $\varepsilon=1$. After that, ε is gradually reduced from 1 to 0 using the homotopy method. When ε is 0 corresponds to the fuel-optimal control problem. We can list the Hamiltonian function

$$H=\lambda_0\frac{2\eta P_{max}u}{I_{sp}^2g_0^2}-\lambda_0\varepsilon\ln(u-u^2)+\boldsymbol{\lambda}_r\cdot\boldsymbol{v}+\boldsymbol{\lambda}_v\cdot\left(-\frac{\mu}{r^3}\boldsymbol{r}+\frac{2\eta P_{max}u}{mI_{sp}g_0}\boldsymbol{\alpha}\right)-\lambda_m\frac{2\eta P_{max}u}{I_{sp}^2g_0^2} \tag{10}$$

where $\boldsymbol{\lambda}_r$, $\boldsymbol{\lambda}_v$ and λ_m denote co-states corresponding to position, velocity, and mass, respectively. According to the Pontryagin's minimum principle (PMP)[28], if we want to achieve the optimal control, we need to make the Hamiltonian function take the minimum value. That is $\boldsymbol{\lambda}_v\cdot\boldsymbol{\alpha}$ takes the minimum value, at this time $\boldsymbol{\lambda}_v$ and $\boldsymbol{\alpha}$ the direction opposite. Therefore, the optimal thrust direction is

$$\boldsymbol{\alpha}^*=-\frac{\boldsymbol{\lambda}_v}{\|\boldsymbol{\lambda}_v\|} \tag{11}$$

According to the first order necessary condition we can write the Euler-Lagrange equation[24,29-30]

$$\dot{\boldsymbol{\lambda}}_r = -\frac{\partial H}{\partial \boldsymbol{r}} = \frac{\mu}{r^3}\boldsymbol{\lambda}_v - \frac{3\mu \boldsymbol{r}\cdot\boldsymbol{\lambda}_v}{r^5}\boldsymbol{r} - \frac{2\eta u}{I_{sp}g_0}\left(\frac{\lambda_0}{I_{sp}g_0} - \frac{\|\boldsymbol{\lambda}_v\|}{m} - \frac{\lambda_m}{I_{sp}g_0}\right)\frac{\partial P_{max}}{\partial \boldsymbol{r}} \tag{12}$$

$$\dot{\boldsymbol{\lambda}}_v = -\frac{\partial H}{\partial \boldsymbol{v}} = -\boldsymbol{\lambda}_r \tag{13}$$

$$\dot{\lambda}_m = -\frac{\partial H}{\partial m} = -\frac{2\eta P_{max}\|\boldsymbol{\lambda}_v\| u}{m^2 I_{sp} g_0} \tag{14}$$

where $\frac{\partial P_{max}}{\partial \boldsymbol{r}}$ can be derived from Eq.(8)[27].

Since the engine specific impulse varies actively in the interval, the optimal actual specific impulse I_{sp}^* is calculated as

$$I_{sp}^* = \begin{cases} I_{sp\text{-}max} & I_{sp\text{-}opt} > I_{sp\text{-}max} \\ I_{sp\text{-}opt} & I_{sp\text{-}min} \leqslant I_{sp\text{-}opt} \leqslant I_{sp\text{-}max} \\ I_{sp\text{-}min} & I_{sp\text{-}opt} < I_{sp\text{-}min} \end{cases} \tag{15}$$

where optimal specific impulse $I_{sp\text{-}opt}$ is expressed as

$$I_{sp\text{-}opt} = \frac{2m(\lambda_0 - \lambda_m)}{\|\boldsymbol{\lambda}_v\| g_0} \tag{16}$$

For Eq.(9) with the modified logarithmic homotopy function, the optimal power throttle level is written as[24]

$$u^* = \frac{2\lambda_0 \varepsilon}{\rho_{log} + 2\varepsilon + \sqrt{\rho_{log}^2 + (2\varepsilon)^2}} \tag{17}$$

Switching function for logarithmic homotopy method ρ_{log} is

$$\rho_{log} = \frac{2\lambda_0 \eta P_{max}\rho}{I_{sp}^2 g_0^2} \tag{18}$$

where $\rho = -\frac{I_{sp}g_0\|\boldsymbol{\lambda}_v\|}{\lambda_0 m} - \frac{\lambda_m}{\lambda_0} + 1$.

The fuel-optimal control problem is transformed into a two-point boundary value problem (TPBVP) with a spacecraft with a fixed velocity at the beginning, end positions and a fixed mission time. If the end-state mass is free, the value of the co-state at the end is zero. The shooting equation is

$$\boldsymbol{\phi} = \left[\boldsymbol{r}(t_f) - \boldsymbol{r}_f, \boldsymbol{v}(t_f) - \boldsymbol{v}_f, \lambda_m(t_f), \|\boldsymbol{\lambda}(t_0)\| - 1\right] \tag{19}$$

where $\boldsymbol{r}_f$ and $\boldsymbol{v}_f$ represent the position and the speed at the rendezvous moment, respectively, and $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_r, \boldsymbol{\lambda}_v, \boldsymbol{\lambda}_m, \lambda_0]^T$.

2 Backward Generation of Fuel-Optimal Rendezvous Examples

The essence of DNNs is to construct mapping relationships between inputs and outputs, which are often complex and nonlinear. In order to obtain such complex mapping relationships, DNNs must be trained based on existing datasets consisting of a large number of samples. However, generating a large amount of sample data is extremely time-consuming. Izzo et al.[19] proposed the method of backward generation of optimal examples in their work. The idea of this method is to assume that an optimal nominal trajectory has been obtained, to perturb the variables except for the constraint of the shooting equation and to carry out the backward integration. Although the new trajectory is different from the nominal trajectory, it is still optimal because it satisfies the first-order necessary condition of optimality and does not change the shooting equation. Since the method of backward generation of optimal examples changes the TPBVP solving problem to numerical integration of trajectories, the speed of data generation is substantially improved compared to solving TPBVP directly. However, only the fuel-optimal transfer and the time-optimal rendezvous with constant acceleration have been solved[19,22]. For the fuel-optimal rendezvous control problem, the backward generation method cannot be used directly because shooting equations are different. Therefore, this paper proposes a method for backward generation of fuel-optimal rendezvous examples based on Ref.[19].

In the shooting Eq.(19), $\|\boldsymbol{\lambda}(t_0)\| - 1$ is the co-states initial value normalization condition, which is introduced to reduce the difficulty of guessing the initial value of the co-states. However, in the backward integration from the moment t_f to the moment t_0, it does not need to guess the initial value of the co-states. So it is not necessary to satisfy this condition when performing the backward integration. The conditions are satisfied by the backward integral as

$$\boldsymbol{\phi}_{back} = \left[\boldsymbol{r}(t_f) - \boldsymbol{r}_f, \boldsymbol{v}(t_f) - \boldsymbol{v}_f, \lambda_m(t_f)\right] = 0 \tag{20}$$

The position co-states, velocity co-states, and

mass at the rendezvous moment are not in Eq.(20), so changing these variables has no effect on the shooting equation ϕ_{back} and does not change the optimality of the solution. We define the equation

$$\begin{cases} \boldsymbol{\lambda}_r(t_f) = \boldsymbol{\lambda}_r^*(t_f) + \delta\boldsymbol{\lambda}_r \\ \boldsymbol{\lambda}_v(t_f) = \boldsymbol{\lambda}_v^*(t_f) + \delta\boldsymbol{\lambda}_v \\ m_f = m_f^* + \delta m_f \end{cases} \tag{21}$$

where the superscript " * " indicates the value of the nominal trajectory; and $\delta\boldsymbol{\lambda}_r$, $\delta\boldsymbol{\lambda}_v$ and δm_f denote perturbation values. The backward integration of $\boldsymbol{\lambda}_r(t_f)$, $\boldsymbol{\lambda}_v(t_f)$, m_f as initial conditions can obtain an optimal trajectory different from the nominal trajectory. The backward integral equation consists of the dynamical Eqs.(1—3) and the Euler-Lagrange Eqs.(12—14).

For the rendezvous problem, there are two conditions for the backward integration cutoff. One for the initial mass m_0^* to be reached at the integration cutoff, and the other for the integration time to be the spacecraft flight time t. When both integral cutoff conditions are satisfied, a new rendezvous optimal trajectory can be generated in backward. In this work, we use flight time as the integration cutoff condition. After that, we use the perturbed $\boldsymbol{\lambda}_r(t_f)$, $\boldsymbol{\lambda}_v(t_f)$ and m_f as the initial condition for the backward integration and integrate the equation to solve for the initial mass m_0. Then, We determine the difference between m_0 and the nominal initial mass m_0^*, and use the idea of dichotomy to gradually and iteratively select the value of m_f. The iteration cutoff condition is

$$|m_0 - m_0^*| \leqslant \sigma \tag{22}$$

where σ denotes the backward integration accuracy.

Now we turn the time to solve TPBVP once into the sum of multiple iterations of integration time. Trajectories solved after iteration have the same end state and different initial states as the nominal trajectory. New trajectories are still optimal since they do not change the shooting equation. The backward generation method of fuel-optimal rendezvous examples(Fig.1) is as follows:

Step 1 Calculate the nominal trajectory of $\boldsymbol{\lambda}_r^*(t_f)$, $\boldsymbol{\lambda}_v^*(t_f)$ and m_f^*, and determine the amount of random disturbance $\delta\boldsymbol{\lambda}_r$ and $\delta\boldsymbol{\lambda}_v$.

Step 2 Set the initial mass iteration step m_{step} and the iteration precision σ.

Step 3 Backward integration with $\boldsymbol{\lambda}_r(t_f)$, $\boldsymbol{\lambda}_v(t_f)$ and m_f as the initial value of integration; and solve the initial mass m_0.

Step 4 Calculate the absolute value of the difference between the m_0 and the nominal value m_0^*. If it is greater than the iteration accuracy σ, reduce the mass iteration step m_{step} by half and proceed to Step 3.

Step 5 If the absolute value of the difference between m_0 and the nominal value is less than the iterative precision, the iteration ends, and the optimal trajectory is output.

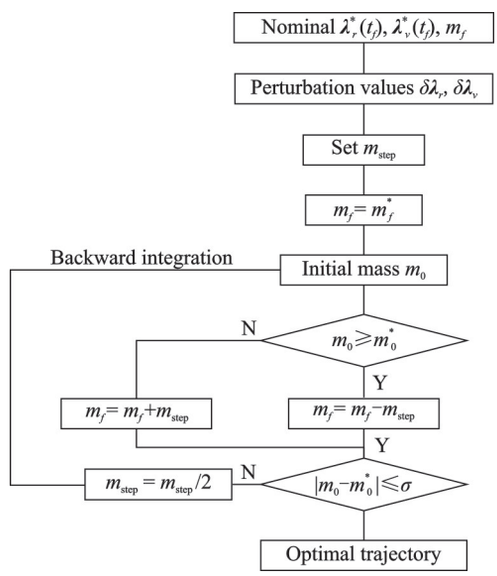

Fig.1 Algorithm for backward generation of fuel-optimal rendezvous examples

3 Deep Neural Networks for Real-Time Optimal Control

DNNs refer to the working mechanism of biological neural networks and use multiple layers and connections between multiple neurons to achieve the mapping between input and output layers. With the increase in the number of neurons and the number of hidden layers of DNNs, DNNs can theoretically achieve the mapping between any input and output. We use feedforward neural networks to learn the mapping relationship between states and control quantities of fuel-optimal control data sam-

ples. Trained DNNs can provide optimal control in real-time based on the current state information.

We use the modified equinoctial orbit elements[31] $\boldsymbol{x}=[p, f, g, h, k, L]^{\mathrm{T}}$ and time as inputs to DNNs, as shown in Fig.2. The reason why $\boldsymbol{r}$ and $\boldsymbol{v}$ in the Cartesian coordinate system (CCS) are not used as the input of DNNs is that the six-state quantities are all fast variables, and the severe data changes are not conducive to the learning of DNNs. Modified equinoctial orbit elements have only one fast variable and it differs from classical orbit elements in that modified equinoctial orbit elements are singularity-free at zero eccentricity and zero orbital inclination. Therefore, modified equinoctial orbit elements are more suitable for describing the spacecraft state.

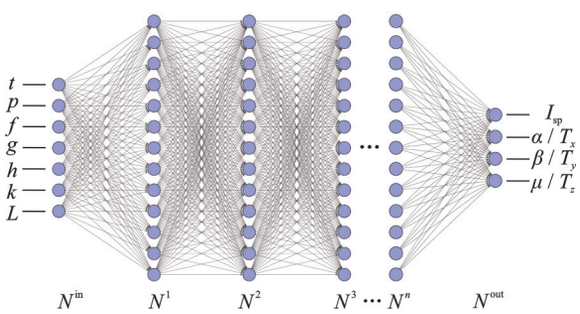

Fig.2 Network structure

The output of the DNN is divided into two types: The thrust and the specific impulse. We use the Cartesian and orbital coordinate systems (OCS) to describe the thrust

$$T_{\mathrm{CCS}}=\left[T_x, T_y, T_z\right]^{\mathrm{T}} \tag{23}$$

$$T_{\mathrm{OCS}}=\left[\alpha, \beta, \mu\right]^{\mathrm{T}} \tag{24}$$

where the thrust T_{CCS} in the Cartesian coordinate system is represented by three components T_x, T_y and T_z; the thrust T_{OCS} in the orbital coordinate system is described by the yaw control angle α, pitch control angle β, and μ.

For DNNs to learn the specific impulse control more accurately, we use the optimal specific impulse $I_{\mathrm{sp\text{-}opt}}$ of Eq.(16) as the output of DNNs. Subsequently, the actual optimal specific impulse I_{sp}^{*} is obtained by limiting the DNN output specific impulse $I_{\mathrm{sp\text{-}opt}}$ using in Eq.(15).

The loss function measures the difference between the DNNs model and the real model. For regression problems, the mean-square error (MSE) is the most commonly used loss function. For outlier-free data, MSE is a good measure of the error between the predicted value and the true optimal value. In this paper, MSE is still used as the loss function. The reason why DNNs can accomplish very complex nonlinear mappings is the inclusion of a nonlinear activation function in the hidden layers. The choice of the activation function and other hyper-parameters are discussed in the following sections.

4 Network Training

We use Fortran to code the optimal trajectory generation program, and Python for DNN training and simulation. The positions and velocities of Earth and Mars in the example are calculated by DE421. Asteroid status information is from the Jet Propulsion Laboratory Horizons system. All simulations are implemented on a desktop computer with Intel Core i7-8700K CPU @3.70 GHz.

4.1 Nominal trajectory

Two interplanetary rendezvous missions are considered in our work: The Earth-Apophis asteroid 230 d rendezvous mission and the Earth-Mars 600 d rendezvous mission. Assume that the spacecraft in both missions use only one NEXT. According to NEXT parameters, the adjustable specific impulse range setting is as [2 210, 4 100] s[27]. Other parameters are set as $P_{\mathrm{AU}}=10$ kW, $P_{\mathrm{P\text{-}max}}=6.9$ kW, $P_{\mathrm{L}}=0.4$ kW and duty cycle $\delta=0.94$. The part in parentheses $d_1, d_2, \cdots, d_5$ in Eq.(7) are chosen as[32]: $d_1=1.106\,3$, $d_2=0.149\,5$, $d_3=-0.299\,0$, $d_4=-0.043\,2$ and $d_5=0$. The specific mission information is given in Tables 1,2.

By solving Eq.(20), we can obtain the optimal control. After that, using the homotopy method, we iterate the program and gradually reduce the value of the homotopy parameter ε to finally find the fuel-optimal control. Fig.3 displays the power throttle level u for the two missions with different ε. When ε decreases from 10^{-2} to 10^{-6}, u gradually approaches the bang-bang control. We consider the fuel-opti-

Table 1 Earth-Apophis asteroid 230 d rendezvous mission

Parameter	Value
Initial date(Coordinate time)	15 Apr. 2029 0:0:0.0
Initial position/km	$[-1.361\ 215\times10^{8},\ -6.320\ 800\times10^{7}, 4.633\ 328\times10^{3}]$
Initial velocity/(km•s^{-1})	$[1.206\ 830\times10, -2.713\ 887\times10,\ 2.276\ 979\times10^{-3}]$
Final position/km	$[6.482\ 640\times10^{7}, 1.807\ 479\times10^{8},\ -5.421\ 290\times10^{6}]$
Final velocity/(km•s^{-1})	$[-2.174\ 530\times10, 1.020\ 122\times10, -6.997\ 165\times10^{-1}]$
Initial mass/kg	500

Table 2 Earth-Mars 600 d rendezvous mission

Parameter	Value
Initial date(Coordinate time)	3 Mar. 2026 0:0:0.0
Initial position/km	$[-1.410\ 638\times10^{8},\ 4.569\ 714\times10^{7},\ -1.968\ 576\times10^{3}]$
Initial velocity/(km•s^{-1})	$[-9.658\ 780,\ -2.843\ 915\times10^{1},\ 2.909\ 212\times10^{-3}]$
Final position/km	$[-5.084\ 734\times10^{6},\ -2.180\ 468\times10^{8},\ -4.445\ 691\times10^{6}]$
Final velocity/(km•s^{-1})	$[2.513\ 562\times10,\ 1.521\ 453,\ -5.843\ 683\times10^{-1}]$
Initial mass/kg	1 000

mal control is achieved for $\varepsilon=10^{-6}$. The fuel-optimal consumption for the Earth-Apophis asteroid rendezvous mission is 128.717 8 kg, and the fuel-optimal consumption for the Earth-Mars rendezvous mission is 157.886 0 kg.

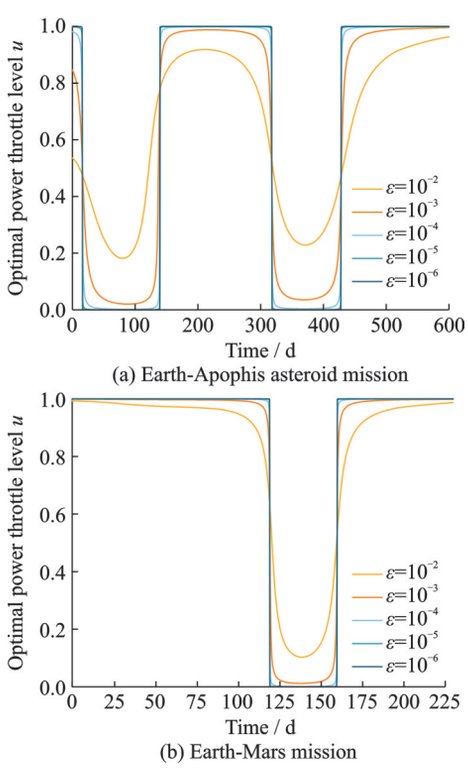

Fig.3 Effect of homotopy parameter changes on the throttle

4.2 Database generation

We use the backward generation method of fuel-optimal rendezvous examples to generate the dataset. It is worth noting that our dataset is generated based on the two nominal trajectories described above. The mass iteration step is set as m_{step} and the iteration precision as $\sigma=10^{-5}$. This parameter setting allows the initial mass of the generated trajectory to be within 0.01 g of the nominal mass. For each task, we solve the fuel-optimal control problem only once. The Earth-Apophis asteroid mission is calculated to take 28.125 s to generate 1 000 trajectories using the backward generation method and 683.766 s using the traditional homotopy method. The Earth-Mars mission takes 29.922 s to generate 1 000 trajectories using the backward generation method and 1 367.906 s using the traditional homotopy method. This shows the backward generation method of fuel-optimal rendezvous examples can greatly improve the speed of the dataset generation.

4.3 DNN optimization

4.3.1 Real-time optimal control networks

The RTOC-DNNs implement the state-control mapping. For DNNs involved in this paper, we only consider control in the spatial extent of the dataset. The dataset spatial extent refers to the maximum extent in space that can be reached theoretically when the dataset is generated. The spatial extent of the dataset depends on the extent of perturbation of the position co-states $\delta\boldsymbol{\lambda}_r$ and velocity co-states $\delta\boldsymbol{\lambda}_v$

in the backward generation method. We set the $\delta\boldsymbol{\lambda}_r$ and $\delta\boldsymbol{\lambda}_v$ perturbation range of the dataset for training the RTOC-DNNs at

$$\begin{cases}\delta\boldsymbol{\lambda}_r\in[-0.5\%,0.5\%]\cdot\boldsymbol{\lambda}_r^*(t_f)\\ \delta\boldsymbol{\lambda}_v\in[-0.5\%,0.5\%]\cdot\boldsymbol{\lambda}_v^*(t_f)\end{cases}\tag{25}$$

For easy description, we refer to this range of perturbations as the 1% perturbation and also denote the following perturbation range as such. Fig.4 visualizes the range of data sets for both tasks.

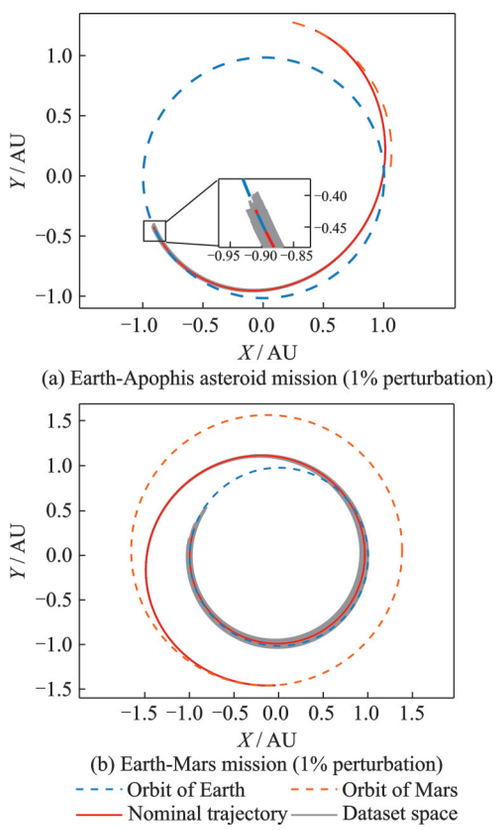

Fig.4 Range of RTOC-DNNs data sets

For both missions, 10 000 trajectories are randomly selected in all training. A total of 1 001 points are taken for each trajectory at equal time intervals, for a total of 10 010 000 sample points. Where 80% of the dataset is used as the training set and 20% as the validation set. After our attempts, the activation function is hyperbolic tangent function (Tanh), and the number of hidden layers is set to five, and each layer contains 256 neurons, which can often achieve better results. To ensure the output margin of the hyperbolic tangent activation function[33], the range of the dataset normalization is set to [−0.9, 0.9]. The Adam algorithm[34] is chosen as the neural network optimization algorithm, and the Adam method is very efficient in optimization because it can correct both the gradient descent direction and the learning rate l_r during the optimization process. Parameters of Adam's method: The learning rate is set to $l_r = 0.001$, and the parameters controlling the first-order momentum and the second-order momentum are set to $\beta_1 = 0.9$, $\beta_2 = 0.999$. Because of the large number of training sets, we use the mini-batch gradient descent method and set the number of batches to 2 048 and the number of training sessions to 100. In addition, the optimization of neural networks using GPUs for speed.

4.3.2 Enhanced fault-tolerant networks

The RTOC-DNNs' dataset has a small spatial distribution in the period close to the rendezvous. This creates a problem in that RTOC-DNNs barely allow for errors in the period close to rendezvous. Larger errors may cause the spacecraft to deviate from the training range of RTOC-DNNs, which in turn makes the error at rendezvous increase dramatically. To improve this situation, Izzo et al.[22] adds larger perturbed data to the dataset and make the neural network learn its control mechanism. Although this approach makes it much less likely that the spacecraft will deviate from the range of the neural network training set. However, it also increases the difficulty of neural network learning due to the introduction of larger perturbed data. For the rendezvous problem, the data set under small perturbations for the flight start moment is sufficient for neural network training, and data with larger perturbations are not needed.

In this paper, we propose the use of EFT-DNNs for enhancing the control accuracy of neural networks in the second half of the flight time. EFT-DNNs are independent of RTOC-DNNs, and EFT-DNNs are used as a backup network to deal with the situation when the spacecraft deviates too much from the spatial extent of the training set of RTOC-DNNs. For the Earth-Apophis asteroid mission, the dataset of EFT-DNNs uses a 20% perturbation. For the Earth-Mars mission, the dataset of EFT-DNNs uses 8% perturbation. The spatial extent of the training data set for EFT-DNNs is shown in

Fig.5. There are 10 000 randomly generated data in the dataset, and 501 feature points are selected for each trajectory at equal time intervals, for a total of 5 010 000 input and output data. Similarly, 80% of the dataset is used as the training set and 20% as the validation set. The hyper parameters settings are kept the same as those for training RTOC-DNNs.

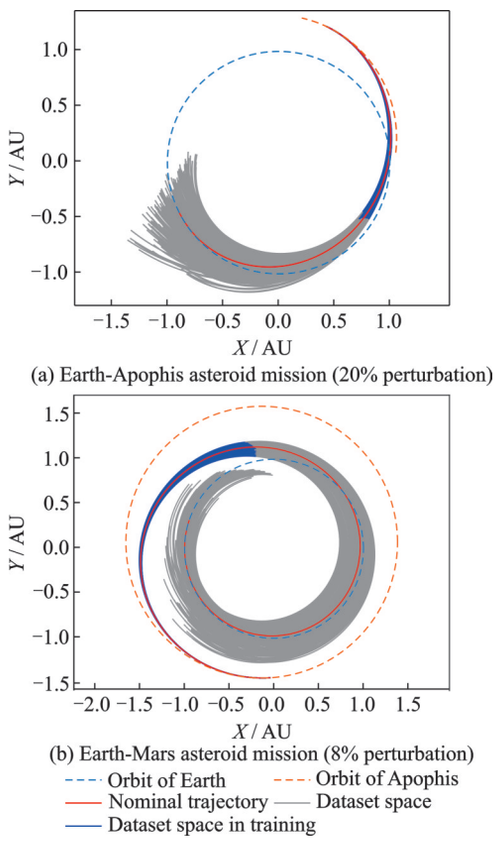

Fig.5 Range of EFT-DNNs data sets

4.4 Network usage scheme

As shown in Fig.6, the overall scheme for the real-time control using DNNs is divided into two parts: Offline training, and online real-time control. The offline training consists of two parts: Dataset generation using the backward fuel optimal trajectory generation method and neural network. The weights and biases of each neuron in RTOC-DNNs and EFT-DNNs eventually obtained by offline training are passed to the online real-time control module. The online real-time control module inputs the real-time spacecraft status and flight time into the DNN, which outputs real-time thrust and specific impulse control. Whether RTOC-DNNs or EFT-DNNs are used for control is determined by the network switching module. The network switching module switches the network depending on whether the time of flight or the spacecraft state is to exceed the DNN training space range. The DNN outputs two parts of control, thrust and specific impulse. The thrust is used directly for real-time control and the specific impulse requires truncation based on the actual specific impulse limit range for real-time control. The controlled spacecraft state is then used as input to the DNN to produce the control output. This process continues until the end of the flight.

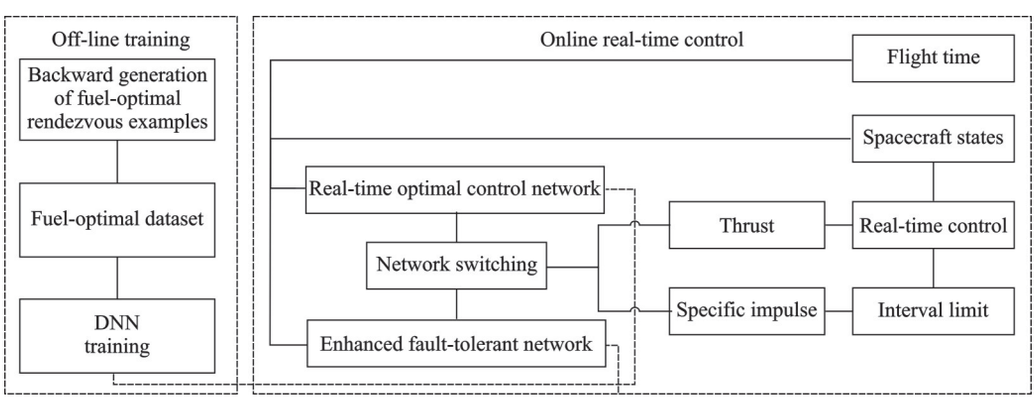

Fig.6 DNN-based real-time control master plan

5 Numerical Results

In this section, we investigate the effect of the homotopy parameters on the training effect of DNNs. To test the effect of DNNs, we perform Monte Carlo simulations for two cases, and the number of Monte Carlo simulations is set to 1 500. Finally, the simulation results using RTOC-DNNs

and EFT-DNNs are analyzed.

5.1 Exmple 1: Rendezvous from the Earth to Apophis

5.1.1 Simulation analysis in deterministic environment

One way to determine whether DNNs are better at learning the optimal control law is to investigate the size of the loss function. Table 3 shows the values of the loss function of RTOC-DNNs in two coordinate systems with different homotopy parameters. From Table 3, it can be seen that the loss function has a tendency to increase gradually as the congruence parameter decreases. This shows that the reduction of the homotopy parameter makes it more difficult for DNNs to learn the optimal control law. The smaller the homotopy parameter, the closer the control law is to bang-bang control. The bang-bang control is reflected in the sharp change of the three output thrusts in the CCS and in the OCS in the sharp change of the power throttle level u. The bang-bang control causes a large error between the predicted and the true values of the DNN near the switch, which in turn makes the loss function increase sharply and makes it more difficult for DNNs to learn the optimal control law. But with a somewhat larger homotopy parameter, the power throttle level u becomes more smoothly with state and flight time, the loss function is lower, and DNNs are more likely to learn the optimal control law.

Table 3 Training results (Earth-Apophis) of RTOC-DNNs

Case	Homotopy parameter	MSE(CCS)	MSE(OCS)
A	10^{-2}	$1.075\,0\times10^{-5}$	$6.940\,2\times10^{-6}$
B	10^{-3}	$3.070\,9\times10^{-5}$	$2.501\,7\times10^{-5}$
C	10^{-4}	$3.208\,9\times10^{-4}$	$3.463\,5\times10^{-4}$
D	10^{-5}	$5.051\,9\times10^{-4}$	$8.048\,6\times10^{-4}$
E	10^{-6}	$8.112\,4\times10^{-4}$	$6.875\,7\times10^{-4}$

Another way to determine whether DNNs are better at learning the optimal control law is to perform Monte Carlo simulations. We perform 1 500 Monte Carlo simulations in which the departure positions of the spacecraft are randomly selected outside the training and validation sets. And the departure positions are obtained by the backward generation method with a perturbation of 1%. In this section and later simulations, the integrator uses the fourth-order Runge-Kutta algorithm (RK4) with a fixed step size, and the number of integrations is set to 1 000. The fuel consumption deviation is defined as the difference between the DNN-controlled fuel consumption and the optimal fuel consumption. Fig.7 shows the Euclidean distance, velocity, and fuel consumption deviation between the arrival state

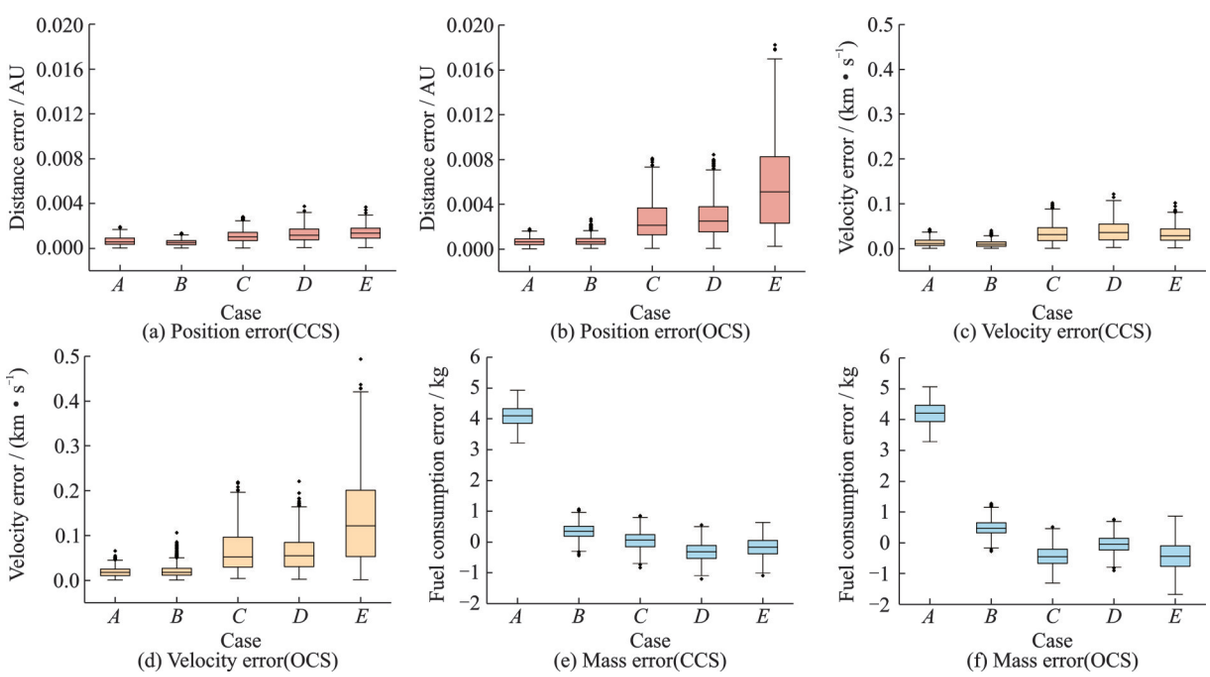

Fig.7 Monte Carlo simulation results (Earth-Apophis asteroid)

and the target state for the Monte Carlo simulations of the Earth-Apophis asteroid mission. The horizontal coordinates A to E corresponds to cases with the homotopy parameter ε of 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5} and 10^{-6}, respectively. For the position deviation and the velocity deviation, there is a tendency for error values to become larger as ε become progressively smaller, and this tendency is more obvious in the OCS. For the fuel consumption deviation, case A in both coordinate systems has the highest fuel consumption, consuming on average more than 4 kg of fuel. while the B to E cases in both coordinate systems consume less than 0.5 kg of fuel on average. The selection of a DNN suitable for real-time control onboard requires not only making its rendezvous position and velocity deviations small, but also a comprehensive consideration of fuel consumption. We choose the network in the coordinate system B case with the smallest distance deviation as real-time control DNNs for this mission. The average error of distance, in this case, is $5.064\ 9\times10^{-4}$ AU, and the average error of speed is $1.075\ 7\times10^{-2}$ km/s; the average consumption is 0.350 41 kg more than the optimal fuel consumption, which is about 0.272 2% of the optimal fuel consumption.

5.1.2 Simulation analysis in uncertain environment

As the above mentioned, simulations are performed in a deterministic environment, while there are various uncertainties in the real flight environment[20], so we further investigate the performance of DNNs in a state-uncertain environment. It is worth noting that the dataset used for DNNs training in this paper does not consider the uncertain environment. So when considering uncertainty, theoretically DNNs do not make the position and velocity errors of end rendezvous less, and often all produce larger errors than they in a deterministic environment. Although EFT-DNNs are used to compensate for the spacecraft being out of the training range of RTOC-DNNs, EFT-DNNs will not stay out of the network training range in the case of large errors, ensuring the accuracy of the control.

We simulate both cases without and with EFT-DNNs. Not using EFT-DNNs means that RTOC-DNNs are used throughout the mission. For the Earth-Apophis asteroid mission, we use EFT-DNNs case and set the flight time from 0 to 130 d use of RTOC-DNNs for control, and after 130 d we use EFT-DNNs for control. RTOC-DNNs and EFT-DNNs are selected for the Cartesian coordinate system B case. 1 500 Monte Carlo simulations are performed in a state uncertain environment satisfying a Gaussian distribution $\sigma_{rx}=\sigma_{ry}=\sigma_{rz}=5$ km, $\sigma_{vx}=\sigma_{vy}=\sigma_{vz}=0.003$ km/s. The simulation results: The average position error is $1.107\ 5\times10^{-2}$ AU; the average speed error is 0.446 9 km/s, and the average fuel consumption error is −1.724 4 kg without EFT-DNNs; the average position error is $5.434\ 9\times10^{-3}$ AU, the average speed error is 0.112 0 km/s, and the average fuel consumption error is 1.535 0 kg with EFT-DNNs. After calculation, the percentage of position error less than 0.01 AU is 59.33% without the use of EFT-DNNs, and this percentage is increased to 89.20% with the use of EFT-DNNs.

Fig.8(a) shows the thrust in the deterministic environment and the thrust in the uncertain environ-

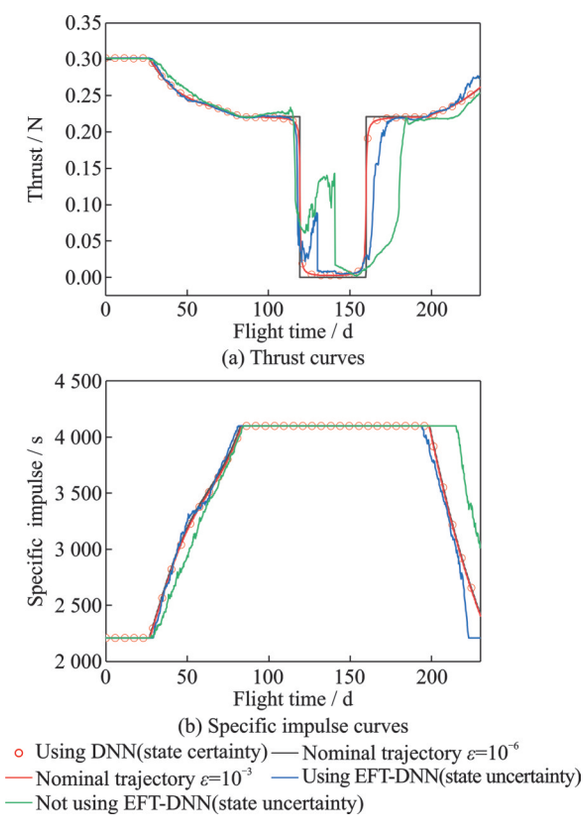

Fig.8 Variation curves of control quantities Earth-Apophis asteroid (under conditions of environmental certainty and environmental uncertainty)

ment for the nominal departure position condition of this task, and Fig.8(b) shows the corresponding specific impulse control. First, it can be seen that the thrust and specific impulse of the DNN output in the determined environment match well when $\varepsilon = 10^{-3}$, indicating that RTOC-DNNs learn the optimal control well. The thrust and specific impulse curves after considering the state uncertainty are significantly different from those when the uncertainty is not considered. This also shows that the DNN outputs optimal thrust and specific impulse control based on the time and state of the current input.

5.2 Exmple 2: Rendezvous from the Earth to Mars

5.2.1 Simulation analysis in deterministic environment

We first analyze the training results (Table 4) of RTOC-DNNs. They are the same as the results of Example 1, and the loss function has a gradually increasing trend as ε decreases. This again validates our idea that a larger homotopy parameter ε will make it easier for DNN to learn the optimal control law.

Table 4 Training results (Earth-Mars) of RTOC-DNNs

Case	Homotopy parameter	MSE(CCS)	MSE(OCS)
A	10^{-2}	$5.771\,9\times10^{-6}$	$9.854\,1\times10^{-6}$
B	10^{-3}	$1.938\,4\times10^{-5}$	$3.490\,1\times10^{-5}$
C	10^{-4}	$2.194\,5\times10^{-4}$	$2.932\,6\times10^{-4}$
D	10^{-5}	$2.689\,4\times10^{-4}$	$9.771\,8\times10^{-4}$
E	10^{-6}	$4.183\,4\times10^{-4}$	$1.194\,7\times10^{-3}$

Similarly, we perform a Monte Carlo simulation with the number of simulations of 1 500. The departure position of the spacecraft in the simulation is generated by the backward generation method with 1% random perturbation. Fig.9 shows the Euclidean distance deviation, velocity deviation, and deviation relative to the optimal fuel consumption between the arrival state and the target state for the Monte Carlo simulation of the Earth-Mars mission. The meaning of the horizontal coordinate is the same as those in Example 1. Under the CCS, the average position and the velocity errors of C are large, and those of A are the smallest of $3.448\,7\times10^{-3}$ AU and $5.963\,7\times10^{-2}$ km/s. However, the deviation of fuel consumption of A in CCS is larger, 11.457 7 kg, which is about 7.256 9% of the optimal fuel consumption. In the OCS, the average position and the velocity errors tend to become larger as ε becomes smaller. The errors at the same ε are all larger than those under the corresponding CCS, so we only select the network at the output of the CCS. The deviation of the combustion consumption of A in the CCS is too large, and the average position and the velocity errors of C are large, both of which

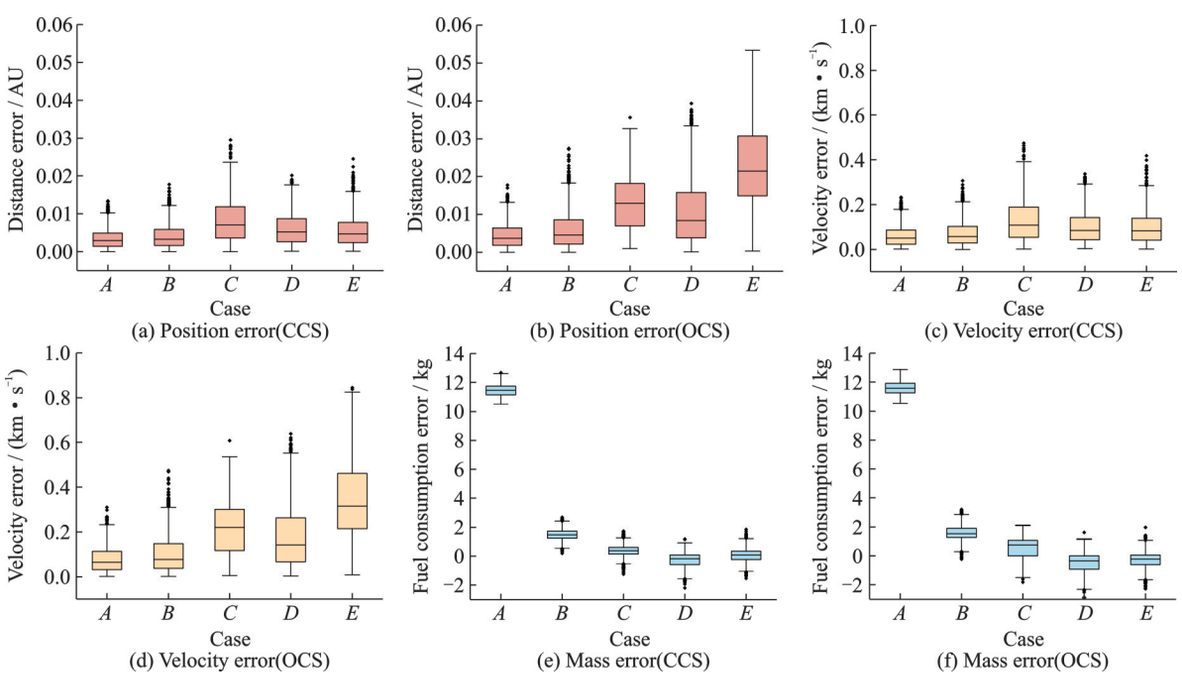

Fig.9 Monte Carlo simulation results (Earth-Mars)

are not selected. The network picks among the remaining three, B, D, and E. We finally chose B, which has the smallest position and speed errors among these three. Although the fuel consumption error of B is larger than that of D and E, it is within the acceptable range. The average error in case B is $4.092\ 2\times10^{-3}$ AU, and the average error in speed is $7.141\ 3\times10^{-2}$ km/s; the error in fuel consumption is 1.494 6 kg, which is about 0.946 6% of the fuel-optimal consumption.

5.2.2 Simulation analysis in uncertain environment

For the Earth-Mars mission, we use EFT-DNNs case and set the flight time from 0 to 350 d use of RTOC-DNNs for control, and after 350 d we use EFT-DNNs for control. RTOC-DNNs and EFT-DNNs are selected for the Cartesian coordinate system B condition. 1 500 Monte Carlo simulations are performed in a state uncertain environment satisfying a Gaussian distribution $\sigma_{rx}=\sigma_{ry}=\sigma_{rz}=1$ km, $\sigma_{vx}=\sigma_{vy}=\sigma_{vz}=0.001$ km/s. Simulation results: The average position error is $1.662\ 8\times10^{-2}$ AU, the average speed error is 0.289 2 km/s, and the average fuel consumption error of 1.342 1 kg without EFT-DNNs; the average position error is $1.040\ 7\times10^{-2}$ AU, the average speed error is 0.188 0 km/s, and the average fuel consumption error is 2.167 1 kg with EFT-DNNs. After calculation, the percentage of position error less than 0.01 AU is 38.20% without the use of EFT-DNNs, and this percentage is increased to 57.47% with the use of EFT-DNNs. Both Example 1 and Example 2 show that the use of EFT-DNNs can lead to a significant improvement in the rendezvous accuracy considering state uncertainty conditions.

Fig.10 shows the thrust in the deterministic environment and the thrust in the uncertain environment for the nominal departure position conditions of the Earth-Mars mission, and the corresponding specific impulse control is also shown. In this example, RTOC-DNNs also learns the optimal control very well. The thrust and specific impulse curves after considering the state uncertainty are more significantly different from those when the uncertainty is not considered. This is also the same as it in Example 1, which shows that DNNs tend to control the spacecraft to reach the rendezvous position optimally from the current state.

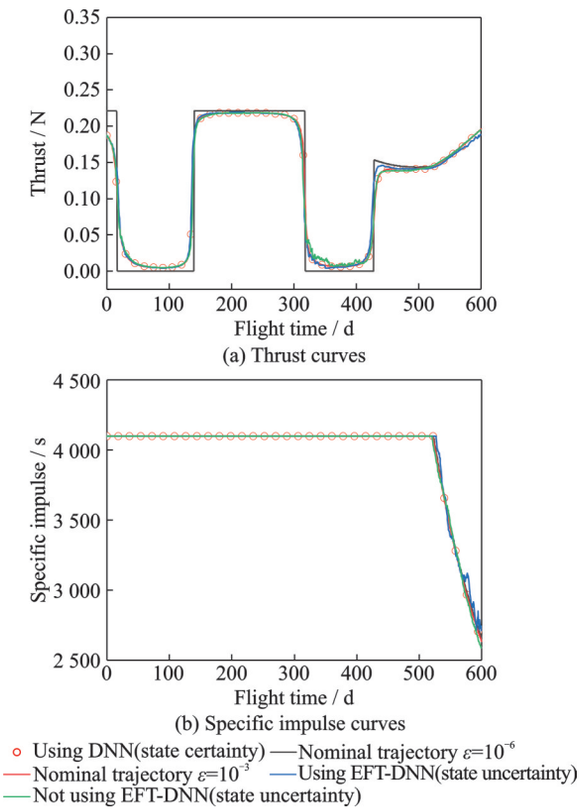

Fig.10 Variation curves of control quantities in the Earth-Mars mission (under conditions of environmental certainty and environmental uncertainty)

5.3 Network speed

The reason why DNNs are suitable for the real-time optimal control of starboard is the fast computation speed, which is shown in Table 5, for the DNNs we have chosen. The output time of the networks we use is about 0.000 23 s at a rate of 4 000 Hz or more, which proves the advantage of DNNs for real-time control onboard.

Table 5 Speed of the networks

Mission	Network	Time/s	Frequency/Hz
Earth-Apophis asteroid	RTOC-DNNs	0.000 234	4 270.43
Earth-Mars	EFT-DNNs	0.000 227	4 400.50
	RTOC-DNNs	0.000 243	4 107.60
	EFT-DNNs	0.000 231	4 319.80

6 Conclusions

This paper presents a method to achieve real-time optimal control of variable-specific-impulse

low-thrust rendezvous via DNNs. First, the method of backward generation optimal examples for the optimal fuel rendezvous problem is proposed. The method generates datasets 24.312 times faster than the homotopy method for the Earth-Apophis asteroid mission, and 46.814 times faster for the Earth-Mars mission. Second, DNN structure is constructed for low-thrust model with variable specific impulse, and the network output control is divided into the thrust output and the specific impulse output. For the specific impulse output, a method is proposed that first learns the optimal specific impulse first and then limits it according to its actual upper and lower limits. DNNs are trained using optimal datasets with different homotopy parameters. The results show that DNNs can learn the optimal thrust and the optimal specific impulse well, and that it is more difficult for the neural network to learn the control law as the homotopy parameter decreases. After that, we conduct Monte Carlo simulations in deterministic and uncertain environments. The simulation results show that the EFT-DNNs can effectively enhance the control range in the second half of flight and improve the rendezvous accuracy. Finally, the network processing speed is calculated. The single processing time of the networks is about 0.000 23 s and the processing frequency is above 4 000 Hz, which is sufficient to show the potential of the proposed method for the real-time control onboard.

References

[1] RAYMAN M D, VARGHESE P, LEHMAN D H, et al. Results from the deep space 1 technology validation mission[J]. Acta Astronautica, 2000, 47(2/3/4/5/6/7/8/9): 475-487.

[2] RAYMAN M D, FRASCHETTI T C, RAYMOND C A, et al. Dawn: A mission in development for exploration of main belt asteroids Vesta and Ceres[J]. Acta Astronautica, 2006, 58(11): 605-616.

[3] KUNINAKA H, NISHIYAMA K, FUNAKAI I, et al. Asteroid rendezvous of HAYABUSA explorer using microwave discharge ion engines[C]//Proceedings of the 29th International Electric Propulsion Conference. [S.l.]:[s.n.], 2005.

[4] KOPPEL C R, ESTUBLIER D. The SMART-1 Hall effect thruster around the moon: In flight experience[C]//Proceedings of the 29th International Electric Propulsion Conference. [S.l.]:[s.n.], 2005: 1-9.

[5] LI J, JIANG F. Survey of low-thrust trajectory optimization methods for deep space exploration[J]. Mechanics in Engineering, 2011, 33(3): 1-6.

[6] HARGRAVES C R, PARIS S W. Direct trajectory optimization using nonlinear programming and collocation[J]. Journal of guidance, control, and dynamics, 1987, 10(4): 338-342.

[7] ENRIGHT P J, CONWAY B A. Discrete approximations to optimal trajectories using direct transcription and nonlinear programming[J]. Journal of Guidance, Control, and Dynamics, 1992, 15(4): 994-1002.

[8] RANIERI C L, OCAMPO C A. Indirect optimization of three-dimensional finite-burning interplanetary transfers including spiral dynamics[J]. Journal of Guidance, Control, and Dynamics, 2009, 32(2): 445-455.

[9] JIANG F, BAOYIN H X, LI J. Practical techniques for low-thrust trajectory optimization with homotopic approach[J]. Journal of Guidance, Control, and Dynamics, 2012, 35(1): 245-258.

[10] GAO Y, KLUEVER C. Low-thrust interplanetary orbit transfers using hybrid trajectory optimization method with multiple shooting[C]//Proceedings of AIAA/AAS Astrodynamics Specialist Conference and Exhibit. Providence, USA: AIAA, 2004: 5088.

[11] LIU B, LI X, GAO J, et al. A method of GNC system driven control based on routing forwarding technology[J]. Flight Control and Detection, 2022(3): 44-51.

[12] IZZO D, MÄRTENS M, PAN B. A survey on artificial intelligence trends in spacecraft guidance dynamics and control[J]. Astrodynamics, 2019, 3(4): 287-299.

[13] ZHU Y, LUO Y Z. Fast evaluation of low-thrust transfers via multilayer perceptions[J]. Journal of Guidance, Control, and Dynamics, 2019, 42(12): 2627-2637.

[14] LI H, BAOYIN H, TOPPUTO F. Neural networks in time-optimal low-thrust interplanetary transfers[J]. IEEE Access, 2019, 7: 156413-156419.

[15] VIAVATTENE G, CERIOTTI M. Artificial neural networks for multiple NEA rendezvous missions with continuous thrust[J]. Journal of Spacecraft and Rockets, 2022, 59(2): 574-586.

[16] YIN S, LI J, CHENG L. Low-thrust spacecraft trajectory optimization via a DNN-based method[J]. Advances in Space Research, 2020, 66(7): 1635-1646.

[17] ZHAO Y, YANG H, LI S. Real-time trajectory optimization for collision-free asteroid landing based on

deep neural networks[J]. Advances in Space Research, 2022, 70(1): 112-124.

[18] CHENG L, WANG Z, JIANG F, et al. Real-time optimal control for spacecraft orbit transfer via multi-scale deep neural networks[J]. IEEE Transactions on Aerospace and Electronic Systems, 2018, 55(5): 2436-2450.

[19] IZZO D, ÖZTÜRK E. Real-time guidance for low-thrust transfers using deep neural networks[J]. Journal of Guidance, Control, and Dynamics, 2021, 44(2): 315-327.

[20] ZAVOLI A, FEDERICI L. Reinforcement learning for robust trajectory design of interplanetary missions[J]. Journal of Guidance, Control, and Dynamics, 2021, 44(8): 1440-1453.

[21] LIU Y, YANG H, LI S. Efficient machine learning Method for co-state estimation of low-thrust optimal trajectories[J]. Journal of Astronautics, 2022, 43(5) 593-602.

[22] IZZO D, ORIGER S. Neural representation of a time optimal, constant acceleration rendezvous[J]. Acta Astronautica, 2023, 204: 510-517.

[23] BRUNO C, DUJARRIC C. In-space nuclear propulsion[J]. Acta Astronautica, 2013, 82(2): 159-165.

[24] CHI Z, YANG H, CHEN S, et al. Homotopy method for optimization of variable-specific-impulse low-thrust trajectories[J]. Astrophysics and Space Science, 2017, 362(11): 1-13.

[25] CHI Z, WU D, JIANG F, et al. Optimization of variable-specific-impulse gravity-assist trajectories[J]. Journal of Spacecraft and Rockets, 2020, 57(2): 291-299.

[26] ARYA V, TAHERI E, JUNKINS J L. A composite framework for co-optimization of spacecraft trajectory and propulsion system[J]. Acta Astronautica, 2021, 178: 773-782.

[27] MENGALI G, QUARTA A A. Fuel-optimal, power-limited rendezvous with variable thruster efficiency[J]. Journal of Guidance, Control, and Dynamics, 2005, 28(6): 1194-1199.

[28] GEERING H P. Optimal control with engineering applications[M]. Germany: Springer, 2007.

[29] YANG H, BAI X, BAOYIN H. Rapid generation of time-optimal trajectories for asteroid landing via convex optimization[J]. Journal of Guidance, Control, and Dynamics, 2017, 40(3): 628-641.

[30] YANG H, BAI X, BAOYIN H. Rapid trajectory planning for asteroid landing with thrust magnitude constraint[J]. Journal of Guidance, Control, and Dynamics, 2017, 40(10): 2713-2720.

[31] WALKER M J H, IRELAND B, OWENS J. A set modified equinoctial orbit elements[J]. Celestial mechanics, 1985, 36(4): 409-419.

[32] WILLIAMS S N, COVERSTONE V. Benefits of solar electric propulsion for the next generation of planetary exploration missions[J]. The Journal of the Astronautical Sciences, 1997, 45(2): 143-159.

[33] PÉREZ I, SAN-JUAN J F, SAN-MARTÍN M, et al. Application of computational intelligence in order to develop hybrid orbit propagation methods[J]. Mathematical Problems in Engineering, 2013(2013): 613628.

[34] KINGMA D P, BA J. Adam: A method for stochastic optimization[EB/OL]. (2017-01-30) [2023-10-1]. http:// arxiv.org/abs/1412.6980.

Acknowledgements This work was supported by the National Natural Science Foundation of China (No.12102177) and the Natural Science Foundation of Jiangsu Province (No.BK20220130).

Authors Mr. **LIU Yuhang** received his M.S. degree from the College of Astronautics, Nanjing University of Aeronautics and Astronautics in 2023. His research interest is low-thrust trajectory optimization.

Dr. **Yang Hongwei** received his B.S. degree from the Department of Engineering Mechanics, Shandong University, China, in 2012 and his Ph.D. degree from the School of Aerospace Engineering, Tsinghua University, China, in 2017. Since 2017, he has been with the College of Astronautics, Nanjing University of Aeronautics and Astronautics, China, where he is a research professor now. His current research interests include astrodynamics, trajectory optimization and spacecraft guidance & control.

Author contributions Mr. **LIU Yuhang** designed the DNN-based control strategy, performed simulations, and wrote the manuscript. Dr. **YANG Hongwei** designed the study and contributed to the discussion and background of the study. Both authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: ZHANG Bei)

基于深度神经网络的变比冲小推力交会实时最优控制

刘宇航,杨洪伟

(南京航空航天大学航天学院,南京 211106, 中国)

摘要:针对燃料最优交会问题,提出了一种基于深度神经网络的实时控制方法。首先,发展了面向燃料最优交会问题的轨迹反向生成方法,该方法基于现有的反向生成思想进行二分法迭代,以满足反向积分的两个截止条件。然后,构造了一种适用于变比冲模型的深度神经网络结构,并将网络的输出控制分为推力输出和比冲输出。提出了先学习最优比冲,然后根据比冲的实际上下限约束对其进行限制以获得比冲输出的方法。进一步,通过设计增强容错深度神经网络以提高交会任务末端接近段的鲁棒性。最后,通过对地球至阿波菲斯小行星和地球至火星的任务仿真,验证了所提方法的有效性和高效性。

关键词:轨迹优化;变比冲;燃料最优控制;间接法;深度神经网络