Journal of Systems Engineering and Electronics
Vol. 34, No. 4, August 2023, pp.1053 – 1062

Discussion on neighborhood optimal trajectory online correction algorithm and its application range

LI Wanli[1,*], LI Jiong[2], and LEI Humin[2]

1. Graduate School, Air Force Engineering University, Xi'an 710051, China;
2. Air and Missile Defense College, Air Force Engineering University, Xi'an 710051, China

Abstract: This paper presents a neighborhood optimal trajectory online correction algorithm considering terminal time variation, and investigates its application range. Firstly, the motion model of midcourse guidance is established, and the online trajectory correction-regenerating strategy is introduced. Secondly, based on the neighborhood optimal control theory, a neighborhood optimal trajectory online correction algorithm considering the terminal time variation is proposed by adding the consideration of terminal time variation to the traditional neighborhood optimal trajectory correction method. Thirdly, the Monte Carlo simulation method is used to analyze the application range of the algorithm, which provides a basis for the division of application domain of the online correction algorithm and the online regeneration algorithm of midcourse guidance trajectory. Finally, the simulation results show that the algorithm has high real-time performance, and the online correction trajectory can meet the requirements of terminal constraint change. The application range of the algorithm is obtained through Monte Carlo simulation.

Keywords: midcourse guidance, online correction, neighborhood optimal control, application range.

DOI: 10.23919/JSEE.2023.000097

1. Introduction

In the mid-guidance flight of interceptor missile, affected by target maneuvering, trajectory online optimization is required to ensure that the end state of interceptor missile meets the optimality index and the changed terminal constraint conditions [1−3]. At present, trajectory online optimization can be categorized into online correction and online regeneration [4−7]. Online trajectory correction is based on the datum optimal trajectory data, and the state quantity can be modified online in its neighborhood, which saves time and improves efficiency effectively without large-scale reoptimization of control quantity. Online trajectory regeneration can well consider the process constraints and meet the adjusted terminal constraints. However, it requires high efficiency and accuracy of the algorithm, and generally used for large range maneuvering targets. The two methods have their own advantages and disadvantages, and it is worth considering how to effectively combine them to form a reasonable online trajectory optimization strategy.

This paper mainly studies the content of online trajectory correction. Mondal et al. [8,9] and Guo et al. [10] used the model predictive control theory to study the trajectory online correction problem. This method continuously corrects the control quantity based on the iterative idea until the error meets the accuracy requirements and the solution rate is greatly improved. However, due to the lack of numerical optimization process, the modified trajectory cannot guarantee the optimality. Lei et al. [11] applied the optimization theory to solve the midcourse guidance trajectory correction problem, and put forward the trajectory optimal correction algorithm. Although it ensures the second-order optimality of the corrected trajectory, it lacks effectiveness. In addition, the optimal trajectory correction algorithm designed based on the neighborhood optimal control theory is a relatively mature online trajectory correction algorithm. Mauro et al. [12,13] used the neighborhood optimal guidance method to solve the problem of lunar trajectory injection and transfer. Zheng et al. [14], from a geometric point of view, presentd a parametric approach to establish the neighboring optimal control for the problem of fixed-time multi-burn orbital transfer. Based on the improvement of neighborhood optimal control theory, Zhou et al. [15,16] and Li et al. [17] proposed a midcourse guidance trajectory correction method for intercepting hypersonic weapons. It not only avoids the disadvantage that the traditional optimization method abandons the benchmark optimal trajectory data for large-scale optimization, effec-

Manuscript received March 29, 2021.
*Corresponding author.
This work was supported by the National Natural Science Foundation of China (61873278; 62173339).

tively improves the solution efficiency, but also meets the second-order optimality conditions and has high calculation accuracy. However, the above methods do not establish the consideration of the application range of the correction method. If the target maneuvers in a large range, it is likely that the correction trajectory cannot meet the process constraints and the changed terminal constraints, resulting in the failure of interception. In other words, the neighborhood optimal trajectory correction method is effective only within its correction range.

Through the above analysis, the online trajectory correction and regeneration strategy is designed by combining the online trajectory correction and online regeneration. A neighborhood optimal trajectory online correction algorithm considering terminal time change is proposed. Based on the traditional neighborhood optimal trajectory correction method, the algorithm increases the consideration of terminal time change and solves the problem of trajectory online correction. At the same time, the application range of the algorithm is studied, and the relevant conclusions of the application range are obtained through Monte Carlo simulation analysis, which can provide a basis for the division of application domain of online correction algorithm and online regeneration algorithm of midcourse guidance trajectory.

2. Trajectory online correction regeneration strategy

The traditional interception method requires the interceptor to adjust the flight state in real time according to the instructions of the guidance law, which is not suitable for intercepting high maneuverability targets. At present, the commonly used midcourse guidance trajectory planning method based on the predicted hit area makes the trajectory break away from the limitation of the guidance law and avoid the interceptor consuming too much energy due to adjusting the trajectory. It has certain feasibility, but this method requires the real-time tracking and prediction technology of the target trajectory. As the premise of interceptor midcourse guidance trajectory planning, trajectory prediction technology has become increasingly mature [18,19]. An assumption is given as follows:

Hypothesis 1 The target can be tracked accurately, and the accuracy of predicting the shift handover point is reasonable.

2.1 Strategy design

Based on Hypothesis 1, the ground command station can calculate the optimal trajectory of the midcourse guidance benchmark before the launch of the interceptor, and import the data into the missile borne computer. Because the high sound intensity maneuvering target has a wide range of maneuverability, only planning the offline optimal trajectory of the interceptor cannot meet the interception requirements. Therefore, this paper designs the online trajectory correction regeneration strategy of midcourse guidance, as shown in Fig. 1.

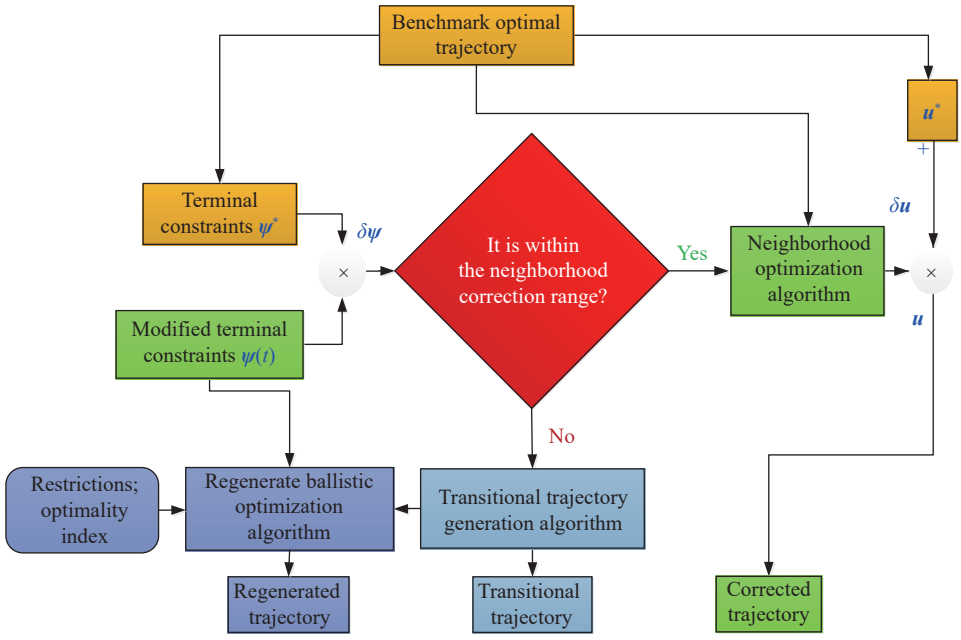


Fig. 1 Trajectory online correction regeneration strategy

Firstly, the midcourse guidance trajectory optimization problem of interceptor is a nonlinear, multi-constraint, and multi-objective optimal control problem. Using the optimization algorithm to solve the benchmark optimal trajectory not only needs to meet the terminal and process constraints, but also needs to have more accurate algorithm accuracy to make the trajectory optimal. When the target maneuver leads to a small change in the predicted shift handover point, the trajectory correction algorithm needs to solve the control correction quantity on the basis of the benchmark optimal trajectory, so that the corrected trajectory still meets the new terminal constraints and the original process constraints. However, the trajectory correction algorithm cannot be established under any conditions, and it should have a certain range of solution adaptability. When the target maneuvers in a large range, resulting in the change of the predicted shift point exceeding the application range of trajectory correction, it is difficult for the trajectory correction algorithm to meet the interception requirements. The regenerated trajectory planning method needs to be used to quickly generate the regenerated trajectory according to the new initial and terminal state conditions, so that the interceptor can still reach the updated predicted shift point to complete the shift handover of intermediate and terminal guidance.

2.2 Motion model construction

When entering the midcourse guidance phase, the rocket booster has completed the boost task and disengaged, and the interceptor glides without power. In order to facilitate the research, the assumptions are as follows.

Hypothesis 2 The interceptor is only affected by aerodynamic force and gravity in the middle guidance phase.

Hypothesis 3 According to the concept of "instantaneous balance", the interceptor is analyzed in the form of "controllable particle".

Hypothesis 4 The Earth is a uniform sphere, and the effects of the Earth's oblateness and rotation are not considered.

According to the above assumptions, the simplified motion model of the interceptor's midcourse guidance phase is

$$\dot{h} = v\sin\theta, \tag{1}$$

$$\dot{z} = v\cos\theta\sin\psi_v, \tag{2}$$

$$\dot{x} = v\cos\theta\cos\psi_v, \tag{3}$$

$$\dot{v} = -\frac{D}{m} - g\sin\theta, \tag{4}$$

$$\dot{\theta} = \frac{L\cos\sigma}{mv} - \frac{g\cos\theta}{v} + \frac{v\cos\theta}{h+r_e}, \tag{5}$$

$$\dot{\psi}_v = \frac{L\sin\sigma}{mv\cos\theta}, \tag{6}$$

where h, z, x is the position coordinate of interceptor centroid under the northeast sky coordinate; v is the velocity of interceptor; θ is the trajectory inclination angle; ψ_v is the trajectory deflection angle; m is the interceptor mass; σ is the angles of bank; r_e is the radius of the Earth and its value is 6371.2 km; g is the gravitational acceleration; D and L are drag and lift respectively. g, D, L are expressed as

$$g = g_0\left(\frac{r_e}{r_e+h}\right)^2, \tag{7}$$

$$D = 0.5\rho v^2 C_D S, \tag{8}$$

$$L = 0.5\rho v^2 C_L S, \tag{9}$$

where g_0 is the gravitational acceleration of the sea level, and its value is 9.81 m/s^2. S is the reference area of interceptor. ρ is the air density. C_D and C_L are the drag coefficient and lift coefficient respectively. ρ, C_D, C_L are expressed as

$$C_D = 0.012 - 0.01\alpha + 0.6\alpha^2, \tag{10}$$

$$C_L = -0.04 + 0.8\alpha, \tag{11}$$

$$\rho = \rho_0 \mathrm{e}^{-h/H}, \tag{12}$$

where α is the angle of attack; ρ_0 is the atmospheric density at the sea level, and its value is 1.226 kg/m^3; H is the reference height, and its value is 7254.24 m.

3. Neighborhood optimal trajectory correction algorithm considering terminal time variation

3.1 Algorithm design

Based on Pontryagin's minimum principle, the first-order necessary conditions for optimal control are derived, and the covariate variable with the same dimension as the state variables is introduced to construct the Hamiltonian equation

$$\boldsymbol{H} = \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{f} \tag{13}$$

where $\boldsymbol{\lambda}$ indicates that the covariate variable and is denoted by $\boldsymbol{\lambda} = [\lambda_h, \lambda_z, \lambda_x, \lambda_V, \lambda_\theta, \lambda_{\psi_v}]^{\mathrm{T}}$; $\boldsymbol{f}$ indicates that the state equation and is denoted by $\boldsymbol{f} = \left[\dot{h}, \dot{z}, \dot{x}, \dot{V}, \dot{\theta}, \dot{\psi}_v\right]^{\mathrm{T}}$. The optimization index in this paper is $\boldsymbol{J} = \boldsymbol{v}_{\max}$. The ter-

minal constraints are introduced into the optimality index to obtain an expanded optimization index J' as follows:

$$J'=\boldsymbol{\phi}+\boldsymbol{\nu}^{\mathrm{T}}\boldsymbol{\psi} \tag{14}$$

where $\boldsymbol{\phi}$ is an end value index; $\boldsymbol{\nu}$ indicates that Lagrange multiplier and is denoted by $\boldsymbol{\nu}=[\nu_h,\ \nu_z,\ \nu_x,\ \nu_\theta,\ \nu_{\psi_v}]^{\mathrm{T}}$; $\boldsymbol{\psi}$ indicates that the terminal constraint and is denoted by

$$\boldsymbol{\psi}=[h-h_f\ \ z-z_f\ \ x-x_f\ \ \theta-\theta_f\ \ \psi_v-\psi_{vf}]^{\mathrm{T}}=\mathbf{0}. \tag{15}$$

Then the necessary conditions to minimize the performance index under the terminal constraints can be expressed as

$$\begin{cases}\dot{\boldsymbol{x}}=\partial \boldsymbol{H}/\partial \boldsymbol{\lambda}\\ \dot{\boldsymbol{\lambda}}=-\partial \boldsymbol{H}/\partial \boldsymbol{x}\end{cases}, \tag{16}$$

$$\partial \boldsymbol{H}/\partial \boldsymbol{u}=\mathbf{0}, \tag{17}$$

$$\boldsymbol{\lambda}(t_f)=-(\partial \boldsymbol{\phi}/\partial \boldsymbol{x}+\boldsymbol{\nu}^{\mathrm{T}}\partial \boldsymbol{\psi}/\partial \boldsymbol{x})_{t=t_f}, \tag{18}$$

$$\boldsymbol{\psi}(\boldsymbol{x}(t_f))=\mathbf{0}. \tag{19}$$

In addition, due to the free time of the terminal, the cross-sectional conditions need to be added,

$$\boldsymbol{H}(t_f)+(\partial \boldsymbol{\phi}/\partial t+\boldsymbol{\nu}^{\mathrm{T}}\partial \boldsymbol{\psi}/\partial \boldsymbol{x})_{t=t_f}=\mathbf{0}. \tag{20}$$

Through the secondary variation of (16) and (19), the following is obtained:

$$\delta\dot{\boldsymbol{x}}=\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{\lambda}\partial \boldsymbol{x}}\delta \boldsymbol{x}+\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{\lambda}\partial \boldsymbol{u}}\delta \boldsymbol{u}, \tag{21}$$

$$\delta\dot{\boldsymbol{\lambda}}=-\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{x}^2}\delta \boldsymbol{x}-\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{x}\partial \boldsymbol{\lambda}}\delta \boldsymbol{\lambda}-\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{x}\partial \boldsymbol{u}}\delta \boldsymbol{u}, \tag{22}$$

$$\mathbf{0}=\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}\partial \boldsymbol{x}}\delta \boldsymbol{x}+\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}\partial \boldsymbol{\lambda}}\delta \boldsymbol{\lambda}+\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}^2}\delta \boldsymbol{u}, \tag{23}$$

$$\begin{aligned}\delta\boldsymbol{\lambda}(t_f)=&\left[\left(\frac{\partial^2 \boldsymbol{\phi}}{\partial \boldsymbol{x}^2}+\boldsymbol{\nu}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{\psi}}{\partial \boldsymbol{x}^2}\right)\delta \boldsymbol{x}+\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{x}}\delta \boldsymbol{\nu}^{\mathrm{T}}+\right.\\ &\left.\left(\frac{\partial^2 \boldsymbol{\phi}}{\partial \boldsymbol{x}\partial t}+\boldsymbol{\nu}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{\psi}}{\partial \boldsymbol{x}\partial t}+\left(\frac{\partial^2 \boldsymbol{\phi}}{\partial \boldsymbol{x}^2}+\boldsymbol{\nu}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{\psi}}{\partial \boldsymbol{x}^2}\right)\dot{\boldsymbol{x}}-\dot{\boldsymbol{\lambda}}\right)\mathrm{d}t\right]_{t=t_f},\end{aligned} \tag{24}$$

$$\begin{aligned}&\left[\left(\frac{\partial^2 \boldsymbol{\phi}}{\partial t\partial \boldsymbol{x}}+\boldsymbol{\nu}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{\psi}}{\partial t\partial \boldsymbol{x}}+\dot{\boldsymbol{x}}\left(\frac{\partial^2 \boldsymbol{\phi}}{\partial \boldsymbol{x}^2}+\boldsymbol{\nu}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{\psi}}{\partial \boldsymbol{x}^2}\right)-\dot{\boldsymbol{\lambda}}\right)\delta \boldsymbol{x}+\right.\\ &\quad\left(\frac{\partial \boldsymbol{\psi}}{\partial t}+\dot{\boldsymbol{x}}\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{x}}\right)\delta \boldsymbol{\nu}^{\mathrm{T}}+\left(\frac{\partial^2 \boldsymbol{\phi}}{\partial t^2}+\boldsymbol{\nu}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{\psi}}{\partial t^2}+\right.\\ &\quad 2\left(\frac{\partial^2 \boldsymbol{\phi}}{\partial t\partial \boldsymbol{x}}+\boldsymbol{\nu}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{\psi}}{\partial t\partial \boldsymbol{x}}\right)\dot{\boldsymbol{x}}+\\ &\quad\left.\left.\dot{\boldsymbol{x}}\left(\left(\frac{\partial^2 \boldsymbol{\phi}}{\partial \boldsymbol{x}^2}+\boldsymbol{\nu}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{\psi}}{\partial \boldsymbol{x}^2}\right)\dot{\boldsymbol{x}}-\dot{\boldsymbol{\lambda}}\right)\right)\delta t\right]_{t=t_f}=\mathbf{0},\end{aligned} \tag{25}$$

$$\delta\psi(x(t_f))=\left[\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{x}}\mathrm{d}x+\left(\frac{\partial \boldsymbol{\psi}}{\partial t}+\frac{\partial \boldsymbol{\psi}}{\partial x}\dot{x}\right)\delta t\right]_{t=t_f}. \tag{26}$$

If $\partial^2 \boldsymbol{H}/\partial \boldsymbol{u}^2$ is nonsingular in the whole process, the control correction $\delta \boldsymbol{u}$ can be obtained by changing (22):

$$\delta \boldsymbol{u}=\left(\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}^2}\right)^{-1}\left(\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}\partial \boldsymbol{x}}\delta \boldsymbol{x}+\frac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}\partial \boldsymbol{\lambda}}\delta \boldsymbol{\lambda}\right). \tag{27}$$

Substitute (23) into (21) and (22):

$$\delta\dot{\boldsymbol{x}}=\boldsymbol{A}\delta \boldsymbol{x}-\boldsymbol{B}\delta \boldsymbol{\lambda}, \tag{28}$$

$$\delta\dot{\boldsymbol{\lambda}}=-\boldsymbol{C}\delta \boldsymbol{x}-\boldsymbol{A}^{\mathrm{T}}\delta \boldsymbol{\lambda}, \tag{29}$$

$$\begin{cases}\boldsymbol{A}=\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{\lambda}\partial \boldsymbol{x}}-\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{\lambda}\partial \boldsymbol{u}}\left(\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}^2}\right)^{-1}\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}\partial \boldsymbol{x}}\\ \boldsymbol{B}=\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{\lambda}\partial \boldsymbol{u}}\left(\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}^2}\right)^{-1}\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}\partial \boldsymbol{\lambda}}\\ \boldsymbol{C}=\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{x}^2}-\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{x}\partial \boldsymbol{u}}\left(\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}^2}\right)^{-1}\dfrac{\partial^2 \boldsymbol{H}}{\partial \boldsymbol{u}\partial \boldsymbol{x}}\end{cases}. \tag{30}$$

By (24), (25), and (26), $\delta\boldsymbol{\lambda}$ and $\delta\boldsymbol{\psi}$ can be expressed as linear expressions about $\delta \boldsymbol{x}$, $\delta \boldsymbol{\nu}$, and δt_f.

$$\begin{cases}\delta\boldsymbol{\lambda}=\boldsymbol{S}\delta \boldsymbol{x}+\boldsymbol{R}\delta \boldsymbol{\nu}+\boldsymbol{m}\delta t_f\\ \delta\boldsymbol{\psi}=\boldsymbol{R}^{\mathrm{T}}\delta \boldsymbol{x}+\boldsymbol{Q}\delta \boldsymbol{\nu}+\boldsymbol{n}\delta t_f\\ 0=\boldsymbol{m}^{\mathrm{T}}\delta \boldsymbol{x}+\boldsymbol{n}^{\mathrm{T}}\delta \boldsymbol{\nu}+\boldsymbol{a}\delta t_f\end{cases} \tag{31}$$

where $\boldsymbol{S}$, $\boldsymbol{R}$, $\boldsymbol{Q}$, $\boldsymbol{m}$, $\boldsymbol{n}$, and $\boldsymbol{a}$ are introduced time-varying matrices. Derivation of (31) is as follows:

$$\begin{cases}\delta\dot{\boldsymbol{\lambda}}=\dot{\boldsymbol{S}}\delta \boldsymbol{x}+\boldsymbol{S}\delta\dot{\boldsymbol{x}}+\dot{\boldsymbol{R}}\delta \boldsymbol{\nu}+\dot{\boldsymbol{m}}\delta t_f\\ 0=\dot{\boldsymbol{R}}^{\mathrm{T}}\delta \boldsymbol{x}+\boldsymbol{R}^{\mathrm{T}}\delta\dot{\boldsymbol{x}}+\dot{\boldsymbol{Q}}\delta \boldsymbol{\nu}+\dot{\boldsymbol{n}}\delta t_f\\ 0=\dot{\boldsymbol{m}}^{\mathrm{T}}\delta \boldsymbol{x}+\boldsymbol{m}^{\mathrm{T}}\delta\dot{\boldsymbol{x}}+\dot{\boldsymbol{n}}^{\mathrm{T}}\delta \boldsymbol{\nu}+\dot{\boldsymbol{a}}\delta t_f\end{cases}. \tag{32}$$

Substitute (31) into (28) and (29):

$$\delta\dot{\boldsymbol{x}}=(\boldsymbol{A}-\boldsymbol{BS})\delta \boldsymbol{x}-(\boldsymbol{BR})\delta \boldsymbol{\nu}-\boldsymbol{Bm}\delta t_f, \tag{33}$$

$$\delta\dot{\boldsymbol{\lambda}}=-\left(\boldsymbol{C}+\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}\right)\delta \boldsymbol{x}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{R}\delta \boldsymbol{\nu}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{m}\delta t_f. \tag{34}$$

Substitute (33) and (34) into (32):

$$\begin{cases}\delta\dot{\boldsymbol{\lambda}}=\left(\dot{\boldsymbol{S}}+\boldsymbol{SA}-\boldsymbol{SBS}\right)\delta \boldsymbol{x}+\left(\dot{\boldsymbol{R}}-\boldsymbol{SBR}\right)\delta \boldsymbol{\nu}+(\dot{\boldsymbol{m}}-\boldsymbol{SBm})\delta t_f\\ \mathbf{0}=\left(\dot{\boldsymbol{R}}^{\mathrm{T}}+\boldsymbol{R}^{\mathrm{T}}\boldsymbol{A}-\boldsymbol{R}^{\mathrm{T}}\boldsymbol{BS}\right)\delta \boldsymbol{x}+\left(\dot{\boldsymbol{Q}}-\boldsymbol{R}^{\mathrm{T}}\boldsymbol{BR}\right)\delta \boldsymbol{\nu}+\\ \qquad\left(\dot{\boldsymbol{n}}-\boldsymbol{R}^{\mathrm{T}}\boldsymbol{Bm}\right)\delta t_f\\ \mathbf{0}=\left(\dot{\boldsymbol{m}}^{\mathrm{T}}+\boldsymbol{m}^{\mathrm{T}}\boldsymbol{A}-\boldsymbol{m}^{\mathrm{T}}\boldsymbol{BS}\right)\delta \boldsymbol{x}+\left(\dot{\boldsymbol{n}}^{\mathrm{T}}-\boldsymbol{m}^{\mathrm{T}}\boldsymbol{BR}\right)\delta \boldsymbol{\nu}+\\ \qquad\left(\dot{\alpha}-\boldsymbol{m}^{\mathrm{T}}\boldsymbol{Bm}\right)\delta t_f\end{cases}. \tag{35}$$

The dynamic differential equations of $\boldsymbol{S}$, $\boldsymbol{R}$, $\boldsymbol{Q}$, $\boldsymbol{m}$, $\boldsymbol{n}$, $\boldsymbol{\alpha}$ can be obtained by using (24)−(26), (33)−(35):

$$\begin{cases}\dot{\boldsymbol{S}}=\boldsymbol{SBS}-\boldsymbol{SA}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}-\boldsymbol{C}\\ \dot{\boldsymbol{R}}=\boldsymbol{SBR}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{R}\\ \dot{\boldsymbol{Q}}=\boldsymbol{R}^{\mathrm{T}}\boldsymbol{BR}\\ \dot{\boldsymbol{m}}=\boldsymbol{SBm}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{m}\\ \dot{\boldsymbol{n}}=\boldsymbol{R}^{\mathrm{T}}\boldsymbol{Bm}\\ \dot{\boldsymbol{\alpha}}=\boldsymbol{m}^{\mathrm{T}}\boldsymbol{Bm}\end{cases}. \tag{36}$$

The terminal values of $\boldsymbol{S}$, $\boldsymbol{R}$, $\boldsymbol{Q}$, $\boldsymbol{m}$, $\boldsymbol{n}$ and $\boldsymbol{\alpha}$ are

$$\begin{cases}\boldsymbol{S}(t_f)=\left(\dfrac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{x}}+\boldsymbol{v}^{\mathrm{T}}\dfrac{\partial^2\boldsymbol{\psi}}{\partial\boldsymbol{x}^2}\right)_{t+t_f}\\ \boldsymbol{R}(t_f)=\left(\dfrac{\partial\boldsymbol{\psi}}{\partial\boldsymbol{x}}\right)_{t+t_f}\\ \boldsymbol{Q}(t_f)=\boldsymbol{0}\\ \boldsymbol{m}(t_f)=\left(\dfrac{\partial^2\boldsymbol{\phi}}{\partial\boldsymbol{x}\partial t}+\boldsymbol{v}^{\mathrm{T}}\dfrac{\partial^2\boldsymbol{\psi}}{\partial\boldsymbol{x}\partial t}+\left(\dfrac{\partial^2\boldsymbol{\phi}}{\partial\boldsymbol{x}^2}+\boldsymbol{v}^{\mathrm{T}}\dfrac{\partial^2\boldsymbol{\psi}}{\partial\boldsymbol{x}^2}\right)\dot{\boldsymbol{x}}-\dot{\boldsymbol{\lambda}}\right)_{t+t_f}\\ \boldsymbol{n}(t_f)=\left(\dfrac{\partial\boldsymbol{\psi}}{\partial t}+\left(\dfrac{\partial\boldsymbol{\psi}}{\partial\boldsymbol{x}}\right)\dot{\boldsymbol{x}}\right)_{t+t_f}\\ \boldsymbol{a}(t_f)=\left(\dfrac{\partial\boldsymbol{\phi}}{\partial t^2}+\boldsymbol{v}^{\mathrm{T}}\dfrac{\partial^2\boldsymbol{\psi}}{\partial t^2}+2\left(\dfrac{\partial^2\boldsymbol{\phi}}{\partial t\partial\boldsymbol{x}}+\boldsymbol{v}^{\mathrm{T}}\dfrac{\partial^2\boldsymbol{\psi}}{\partial t\partial\boldsymbol{x}}\right)\dot{\boldsymbol{x}}+\right.\\ \qquad\left.\dot{\boldsymbol{x}}\left(\left(\dfrac{\partial^2\boldsymbol{\phi}}{\partial t^2}+\boldsymbol{v}^{\mathrm{T}}\dfrac{\partial^2\boldsymbol{\psi}}{\partial t^2}\right)\dot{\boldsymbol{x}}-\dot{\boldsymbol{\lambda}}\right)\right)_{t+t_f}\end{cases}. \tag{37}$$

The values of $\boldsymbol{S}$, $\boldsymbol{R}$, $\boldsymbol{Q}$, $\boldsymbol{m}$, $\boldsymbol{n}$, and $\boldsymbol{\alpha}$ in the whole process can be obtained by inversely integrating (36) with (37). Then $\delta\boldsymbol{v}$ and terminal time change value δt_f in the whole process are solved:

$$\delta\boldsymbol{v}=\left(\boldsymbol{Q}-\boldsymbol{n}\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}\right)^{-1}\left(\delta\boldsymbol{\psi}-\left(\boldsymbol{R}^{\mathrm{T}}-\boldsymbol{n}\boldsymbol{\alpha}^{-1}\boldsymbol{m}^{\mathrm{T}}\right)\delta\boldsymbol{x}\right), \tag{38}$$

$$\delta t_f=\left(\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}\left(\boldsymbol{Q}-\boldsymbol{n}\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}\right)^{-1}\left(\boldsymbol{R}^{\mathrm{T}}-\boldsymbol{n}\boldsymbol{\alpha}^{-1}\boldsymbol{m}^{\mathrm{T}}\right)-\boldsymbol{\alpha}^{-1}\boldsymbol{m}^{\mathrm{T}}\right)\delta\boldsymbol{x}-\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}\left(\boldsymbol{Q}-\boldsymbol{n}\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}\right)^{-1}\delta\boldsymbol{\psi}. \tag{39}$$

Substitute (38) and (39) into (31):

$$\delta\boldsymbol{\lambda}=\left(\left(\boldsymbol{m}\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}-\boldsymbol{R}\right)\left(\boldsymbol{Q}-\boldsymbol{n}\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}\right)^{-1}\left(\boldsymbol{R}-\boldsymbol{n}\boldsymbol{\alpha}^{-1}\boldsymbol{m}^{\mathrm{T}}\right)+\boldsymbol{S}-\boldsymbol{m}\boldsymbol{\alpha}^{-1}\boldsymbol{m}^{\mathrm{T}}\right)\delta\boldsymbol{x}+\left(\boldsymbol{R}-\boldsymbol{m}\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}\right)\left(\boldsymbol{Q}-\boldsymbol{n}\boldsymbol{\alpha}^{-1}\boldsymbol{n}^{\mathrm{T}}\right)^{-1}\delta\boldsymbol{\psi}. \tag{40}$$

Substitute (40) into (27) to obtain the control amount correction value under the condition of terminal time freedom:

$$\delta\boldsymbol{u}=\left(\frac{\partial^2\boldsymbol{H}}{\partial\boldsymbol{u}^2}\right)^{-1}\left(\left(\frac{\partial^2\boldsymbol{H}}{\partial\boldsymbol{u}\partial\boldsymbol{x}}+\frac{\partial^2\boldsymbol{H}}{\partial\boldsymbol{u}\partial\boldsymbol{\lambda}}\left(\left(\frac{\boldsymbol{m}\boldsymbol{n}^{\mathrm{T}}}{\boldsymbol{\alpha}}-\boldsymbol{R}\right)\left(\boldsymbol{Q}-\frac{\boldsymbol{n}\boldsymbol{m}^{\mathrm{T}}}{\boldsymbol{\alpha}}\right)^{-1}\left(\boldsymbol{R}^{\mathrm{T}}-\frac{\boldsymbol{n}\boldsymbol{m}^{\mathrm{T}}}{\boldsymbol{\alpha}}\right)+\boldsymbol{S}-\frac{\boldsymbol{m}\boldsymbol{m}^{\mathrm{T}}}{\boldsymbol{\alpha}}\right)\right)\delta\boldsymbol{x}+\frac{\partial^2\boldsymbol{H}}{\partial\boldsymbol{u}\partial\boldsymbol{\lambda}}\left(\boldsymbol{R}-\frac{\boldsymbol{m}\boldsymbol{n}^{\mathrm{T}}}{\boldsymbol{\alpha}}\right)\left(\boldsymbol{Q}-\frac{\boldsymbol{n}\boldsymbol{n}^{\mathrm{T}}}{\boldsymbol{\alpha}}\right)^{-1}\delta\boldsymbol{\psi}\right). \tag{41}$$

In addition, the ratio b is obtained by comparing the corrected terminal time with the terminal time of the reference optimal trajectory t_f^*:

$$b=\frac{t_f+\delta t_f}{t_f^*}. \tag{42}$$

Then the corrected instruction time step is obtained:

$$\Delta t=b\Delta t^* \tag{43}$$

where Δt^* is the command time step of the benchmark optimal trajectory.

The proof of neighborhood optimal control theory can be found in [20−22], which is not repeated here.

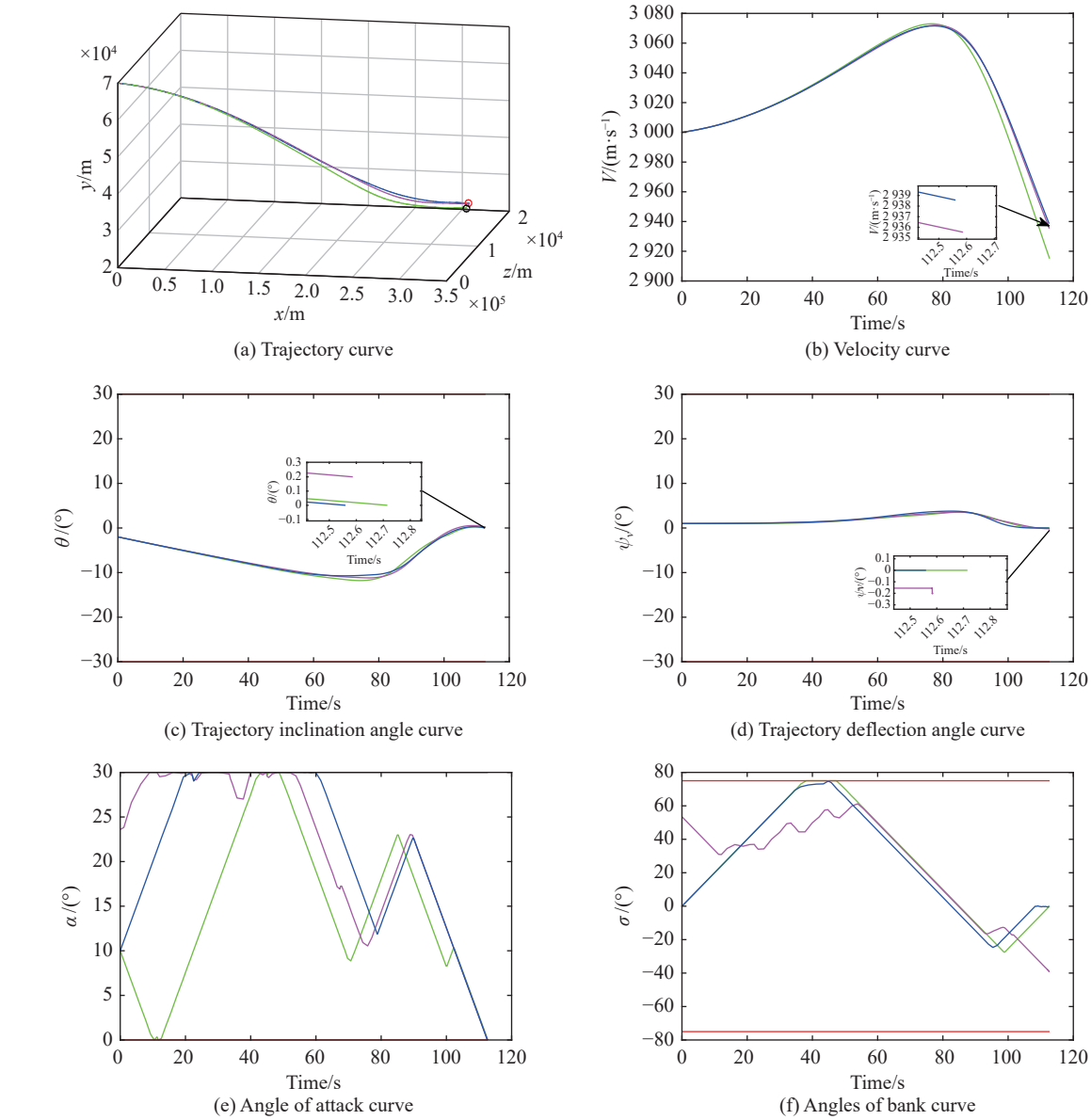
3.2 Simulation verification

In order to verify the effectiveness of the neighborhood optimal correction algorithm considering the change of terminal time to solve the problem of trajectory online correction, the following digital simulation experiments are carried out. The reference optimal trajectory in the simulation is obtained by convex optimization method [23,24], the boundary conditions of the guidance section of the interceptor are shown in Table 1, where "max" represents the optimal value of terminal velocity (i.e., the maximum value).

Table 1 Boundary conditions of interceptor midcourse guidance section

Parameter	Value	Parameter	Value
x_0/km	0	x_f/km	338
y_0/km	70	y_f/km	30
z_0/km	0	z_f/km	10
v_0/(km·s^{-1})	3	v_f/(km·s^{-1})	max
θ_0/(°)	−2	θ_f/(°)	0
ψ_{v0}/(°)	1	ψ_{vf}/(°)	0

It is assumed that the target maneuver leads to the change of the new predicted shift handover point (338 km, 31 km, 10.5 km). The neighborhood optimal correction algorithm considering the change of terminal time is used for correction simulation, and compared with the convex optimization method. The simulation results are given in Fig. 2 and Table 2.



(a) Trajectory curve

(b) Velocity curve

(c) Trajectory inclination angle curve

(d) Trajectory deflection angle curve

(e) Angle of attack curve

(f) Angles of bank curve

Fig. 2 Comparison of simulation results

Table 2 Comparison of terminal state simulation results

Method	x/km	h/km	z/km	$v/(\mathrm{m}\cdot\mathrm{s}^{-1})$	$\theta/(°)$	$\psi_v/(°)$	Simulation time/s
Convex optimization method	338	31	10.5	2938.6	0	0	7.59
Neighborhood correction algorithm	338	30.9	10.6	2935.6	0.2	−0.2	0.26

Fig. 2 (a) shows the three-dimensional trajectory curve. It can be seen that the neighborhood correction trajectory is similar to the convex optimization trajectory, and the trajectory is relatively smooth, which conforms to the new prediction handover point constraint and proves the rationality of the correction trajectory generated by the algorithm in this paper. Fig. 2(b) shows the speed change trend. It can be seen that the modified terminal speed is very close to the optimization result of the convex optimization method, which proves that the algorithm in this paper has certain optimization. In Fig. 2(c) and Fig. 2(d), the error between the trajectory inclination angle and trajectory deflection angle of neighborhood correction trajectory at the end is very small and can be ignored. In

Fig. 2(e) and Fig. 2(f), the control amount of neighborhood correction trajectory fluctuates obviously in the early stage, and the change trend in the later stage gradually moves closer to the convex optimization trajectory, because the change amount of terminal constraint will affect the correction amount of control amount at the initial time through feedback, and the correction amount of control amount will decrease with the decrease of terminal error value.

Table 2 shows the terminal state of neighborhood correction trajectory and convex optimization trajectory. It can be seen that the gap between them is very small, and the simulation time of the neighborhood correction algorithm is only 0.26 s, which is much less time-consuming than the convex optimization method. It proves that this algorithm has the ability of online real-time correction.

4. Application range of algorithm

4.1 Application range calculation

The algorithm introduced in Section 3 can obtain the neighborhood optimal correction trajectory considering the change of terminal time. However, the algorithm cannot meet the ballistic requirements in any case, first, sufficient conditions ($\partial^2 \boldsymbol{Ha}/\partial \boldsymbol{u}^2 > \boldsymbol{0}$, $\boldsymbol{Q} > \boldsymbol{0}$, and $\boldsymbol{S} - \boldsymbol{RQ}^{-1}\boldsymbol{R}^{\mathrm{T}}\boldsymbol{C}$ is a finite matrix) must be satisfied; secondly, through the analysis of (36)−(41), it is concluded that under the condition of unchanged initial conditions, the neighborhood optimal trajectory correction algorithm is effective only if the change of terminal constraint $\delta\psi$ is controlled within a certain range:

$$\delta\psi_{\min} < \delta\psi < \delta\psi_{\max}. \tag{44}$$

The upper and lower limits of (44) derived directly from (36)−(41) are very complex, and it is difficult to obtain its accurate value. Therefore, this paper adopts the Monte Carlo simulation method to analyze the application range of the neighborhood optimal trajectory online correction algorithm.

The analysis of the reference optimal trajectory in Section 3 shows that the range span of the interceptor in the middle guidance stage is far beyond the height and the lateral distance. The impact of target maneuver is mainly reflected in the deviation of terminal height and lateral distance. In order to ensure the smooth transition of trajectory in the shift handover stage of middle and terminal guidance, the terminal angle generally tends to zero. Therefore, the application range of $\delta\psi$ is mainly the allowable range of terminal height deviation δh_f and terminal lateral distance deviation δz_f.

Meanwhile, in order to check whether the corrected trajectory meets the interception requirements, the terminal constraint error value is set as follows:

$$\Delta\theta_f = \theta_f^* + \delta\theta_f - \theta_f, \tag{45}$$

$$\Delta\psi_{vf} = \psi_{vf}^* + \delta\psi_{vf} - \psi_{vf}, \tag{46}$$

$$\Delta x_f = x_f^* + \delta x_f - x_f, \tag{47}$$

$$\Delta h_f = h_f^* + \delta h_f - h_f, \tag{48}$$

$$\Delta z_f = z_f^* + \delta z_f - z_f, \tag{49}$$

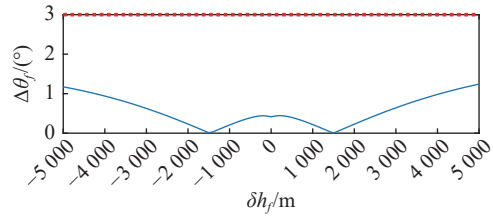
$$l = \sqrt{\Delta x_f^2 + \Delta h_f^2 + \Delta z_f^2}, \tag{50}$$

where $(\cdot)_f^*$ represents the terminal state of the benchmark optimal trajectory, $(\cdot)_f$ represents the terminal state of the field correction trajectory, $\Delta(\cdot)_f$ represents the terminal state deviation value affected by the target maneuver, and l represents the distance deviation value between the neighborhood correction trajectory terminal and the new predicted shift handover point.

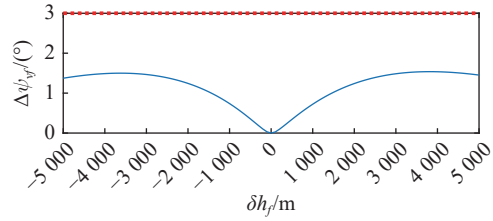
4.2 Simulation analysis

In order to study the application range of the algorithm proposed in Section 3, this section uses the algorithm for trajectory correction and Monte Carlo simulation analysis by changing the terminal constraints and based on the benchmark optimal trajectory in the third section. The maximum allowable distance error is 1 500 m and the maximum allowable angle error is 3°.

Simulation scenario 1 Only the terminal height constraint is changed, and other terminal constraints remain unchanged. Taking the original terminal height as the benchmark, the interval is 100 m, and the deviation δh_f range is ±5 km. The trajectory is corrected by using the algorithm in this paper, and the Monte Carlo simulation analysis is carried out. The simulation results are shown in Fig. 3.



(a) Trajectory inclination error



(b) Trajectory deflection error

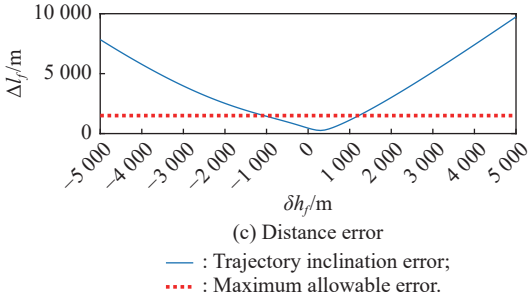


Fig. 3 Simulation scenario 1 results

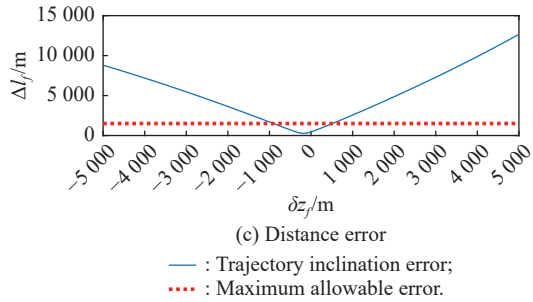


Fig. 4 Simulation scenario 2 results

Fig. 3(a) and Fig. 3(b) show the errors of trajectory inclination and trajectory deflection angle. It can be seen that under the condition of only considering the change of terminal height constraint, the error between the terminal trajectory inclination and trajectory deflection angle of corrected trajectory and the changed terminal constraint is small, which basically meets the requirements of trajectory. Fig. 3(c) shows the distance error. It can be concluded that the distance error value of the corrected trajectory increases with the increase of the absolute value of the terminal height constraint change. Under the limitation of the maximum allowable error, the application range of this algorithm considering only the change of terminal height constraint is from −1 000 m to 1 200 m.

Simulation scenario 2 Only the terminal lateral constraint is changed, and other terminal constraints remain unchanged. Taking the original terminal lateral as the benchmark, the interval is 100 m, and the deviation δz_f range is ±5 km. The trajectory is corrected by using the algorithm in this paper, and the Monte Carlo simulation analysis is carried out. The simulation results are shown in Fig. 4.

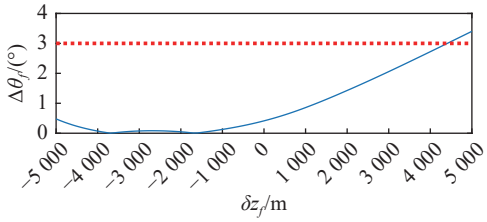
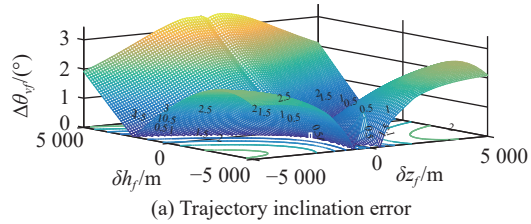
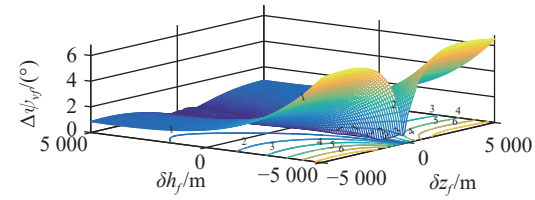


Fig. 4(a) and Fig. 4(b) show the errors of trajectory inclination and trajectory deflection angle. It can be concluded that under the condition of only considering the change of terminal lateral constraint, the error value between the terminal trajectory inclination angle and trajectory deflection angle of the corrected trajectory and the changed terminal constraint also almost meets the maximum allowable error limit. Different from the result of simulation scenario 1, the change trend of terminal trajectory inclination angle error in this case is opposite to that of terminal trajectory deflection angle error. Fig. 4(c) shows the distance error. It can be concluded that the variation trend of the distance error of the modified trajectory is similar to that of the simulation scenario 1. Under the limitation of the maximum allowable error, the algorithm in this paper only considers that the application range of the change of terminal deviation constraint is from −900 m to 500 m. Compared with the simulation scenario 1, it can be concluded that the change of terminal deviation constraint has a greater impact on the modified trajectory than the change of terminal height constraint.

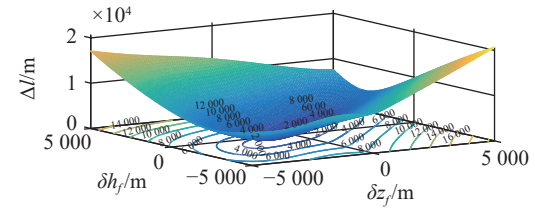
Simulation scenario 3 Change the terminal height and deflection constraints at the same time, and other terminal constraints remain unchanged. Taking the original terminal height and lateral as the benchmark, the interval is 100 m, and the deviation δh_f and δz_f range are ±5 km. The trajectory is corrected by using the algorithm in this paper, and the Monte Carlo simulation analysis is carried out. The simulation results are shown in Fig. 5.



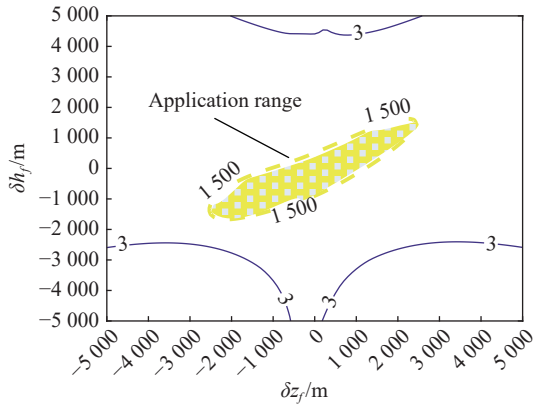
(a) Trajectory inclination error



(b) Trajectory deflection error



(c) Distance error



(d) Application range diagram

Fig. 5 Simulation scenario 3 results

Fig. 5(a) and Fig. 5(b) show the errors of trajectory inclination and trajectory deflection angle. It can be concluded that considering the simultaneous change of terminal height constraint and terminal lateral constraint, most of the error values between the terminal trajectory inclination and trajectory deflection angle of the modified trajectory and the changed terminal constraint are within the maximum allowable error limit. Combined with the analysis of the results of the first two simulation scenarios, it can be concluded that within a certain range of terminal distance, the influence of this algorithm on the terminal angle error is very small. Fig. 5(c) shows the distance error. It can be concluded that considering the simultaneous change of terminal height constraint and terminal lateral constraint, the distance error value of the corrected trajectory does not simply increase with the increase of δh_f and δz_f, but presents an elliptical range on the error plan, as shown in the yellow part of Fig. 5(d). This shows that when the terminal height constraint and terminal bias constraint change at the same time, the influence of δh_f and δz_f on the accuracy of this algorithm is not linear superposition.

5. Conclusions

The trajectory online correction-regeneration strategy is designed to provide a new idea for the research of trajectory online generation. A neighborhood optimal trajectory correction algorithm considering terminal time variation is studied, and the simulation results show that the algorithm can effectively solve the problem of trajectory online correction for small-scale maneuvering of targets. Through the Monte Carlo simulation experiment of this algorithm, the adaptability of this algorithm is analyzed, and the relevant conclusions are as follows.

(i) The online correction algorithm cannot effectively solve the problem of terminal constraint change caused by target maneuver in any case, and it has a limited application range.

(ii) The influence of terminal lateral constraint on the error of corrected trajectory is greater than that of terminal height constraint.

(iii) When the terminal lateral constraint and terminal height constraint change at the same time, there is a functional relationship between them on the error of corrected trajectory

The above conclusions can provide assistance for the research on the application domain of midcourse trajectory online correction algorithm.

References

[1] CHEN B H, CHE T L. Design of missile mid-course guidance for constrained impact angle. Journal of Physics: Conference Series, 2020, 1509: 012007.

[2] ZHANG Y L, XIE Y. Review of trajectory planning and guidance methods for gliding vehicles. Acta Aeronautica et Astronautica Sinica, 2020, 41(1): 50–62. (in Chinese)

[3] CHEN W Y, SHAO L, LEI H M. On-line trajectory generation of midcourse cooperative guidance for multiple interceptors. Journal of Systems Engineering and Electronics, 2022, 33(1): 197–209.

[4] ZHAO H, SU Z. Real-time estimation of roll angle for trajectory correction projectile using radial magnetometers. IET Radar Sonar & Navigation, 2020, 14(10): 1559–1570.

[5] FU S N, ZHOU G Q, XIA Q L. A trajectory shaping guidance law with field-of-view angle constraint and terminal limits. Journal of Systems Engineering and Electronics, 2022, 33(2): 426–437.

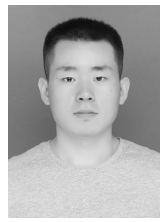
[6] ZHANG X, HUANG H, LI W. Neural-network-based real-time trajectory re-planning for Mars entry guidance. Proceedings of the Institution of Mechanical Engineers Part G: Journal of Aerospace Engineering, 2016, 231(14): 2634–2645.

[7] LI Z H, HU C, DING C B, et al. Stochastic gradient particle swarm optimization based entry trajectory rapid planning for hypersonic glide vehicles. Aerospace Science and Technology, 2018, 76(5): 176–186.

[8] MONDAL S, PADHI R. State and input constrained missile guidance using spectral model predictive static programming.

Proc. of the AIAA Guidance, Navigation, and Control Conference, 2018. https://arc.aiaa.org/doi/abs/10.251416.2018-1584.
[9] MONDAL S, PADHI R. Angle-constrained terminal guidance using quasi-spectral model predictive static programming. Journal of Guidance, Control & Dynamics, 2018, 41(3): 779–787.
[10] GUO X, FU W X, FU B, et al. Penetration trajectory programming for air-breathing hypersonic vehicles during the cruise phase. Journal of Astronautics, 2017, 38(3): 287–295. (in Chinese)
[11] LEI H M, ZHOU J, ZHAI D L, et al. Midcourse guidance optimal trajectory modification using the second variation. Systems Engineering and Electronics, 2016, 38(12): 2807–2813. (in Chinese)
[12] MAURO P, FABIO C. Variable-time-domain neighboring optimal guidance and attitude control of low-thrust lunar orbit transfers. Acta Astronautica, 2020, 175: 616–626.
[13] MAURO P, FABIO C. Lunar ascent and orbit injection via neighboring optimal guidance and constrained attitude control. Journal of Aerospace Engineering, 2018, 31(5): 04018071.
[14] ZHENG C. Neighboring optimal control for fixed-time multi-burn orbital transfers. Aerospace Science and Technology, 2017, 61(3): 57–65.
[15] ZHOU J, LEI H M, ZHAI D L, et al. Optimal midcourse trajectory cluster generation and trajectory modification for hypersonic interceptions. Journal of Systems Engineering and Electronics, 2017, 28(6): 1162–1173.
[16] ZHOU J, LEI H M, ZHANG D Y. Online optimal midcourse trajectory modification algorithm for hypersonic vehicle interceptions. Aerospace Science and Technology, 2017, 63(4): 266–277.
[17] LI N B, LEI H M, ZHOU J, et al. Variable-time-domain online neighboring optimal trajectory modification for hypersonic interceptors. International Journal of Aerospace Engineering, 2017, 2017: 9456179.
[18] GUILLERMO F, JUAN A B, JAVIER L-L. Generation of aircraft intent based on a microstrategy search tree. IEEE Trans. on Intelligent Transportation Systems, 2017, 18(6): 1405–1421.
[19] ZHANG K, XIONG J J, LI F, et al. Bayesian trajectory prediction for a hypersonic gliding reentry vehicle based on intent inference. Journal of Astronautics, 2018, 39(11): 1258–1265. (in Chinese)
[20] BRYSON A E, HO Y C. Applied optimal control. New York: Blaisdell Publishing Company, 1975.
[21] MAURO P, GIAMPAOLO C, PAOLO T, et al. Variable-time-domain neighboring optimal guidance, Part 1: algorithm structure. Journal of Optimization Theory and Applications, 2015, 166(1): 76–92.
[22] MAURO P, GIAMPAOLO C, PAOLO T. Variable-time-domain neighboring optimal guidance, Part 2: application to lunar descent and soft landing. Journal of Optimization Theory and Applications, 2015, 166(1): 93–114.
[23] WANG Z, GRANT M J. Constrained trajectory optimization for planetary entry via sequential convex programming. Journal of Guidance, Control & Dynamics, 2017, 40(10): 2603–2615.
[24] LIU X F, SHEN Z J, LU P. Exact convex relaxation for optimal flight of aerodynamically controlled missiles. IEEE Trans. on Aerospace and Electronic Systems, 2016, 52(4): 1881–1892.

Biographies



LI Wanli was born in 1997. He received his M.S. degree in control science and engineering from Air Force Engineering University in 2021. He is currently working toward his Ph.D. degree in control science and engineering at Research College, Air Force Engineering University, Xi'an, China. His research interests focus on mid-guided missile trajectory optimization, correction, and regeneration.
E-mail: liwl170@163.com.

LI Jiong was born in 1979. He received his M.S. and Ph.D. degrees in control science and engineering from Air Force Engineering University in 2003 and 2007, respectively. He is now an associate professor and a doctoral supervisor at Air and Missile Defense College, Air Force Engineering University, Xi'an, China. His research interests include advanced guidance law design, hypersonic interception, and missile automatic controller design.
E-mail: gracefuloo1@126.com

LEI Humin was born in 1960. He received his M.S. and Ph.D. degrees in navigation, guidance and control from Northwestern Polytechnical University, in 1989 and 1999, respectively. He is a professor and a doctoral supervisor in Air and Missile Defense College, Air Force Engineering University, Xi'an, China. His current research interests are in the field of advanced guidance law design, hypersonic vehicle controller design, and hypersonic interception strategy.
E-mail: hmleinet@126.com