

## Research Article

# Reentry Trajectory Optimization for a Hypersonic Vehicle Based on an Improved Adaptive Fireworks Algorithm

Xing Wei,<sup>1</sup> Lei Liu ,<sup>1</sup> Yongji Wang ,<sup>1</sup> and Ye Yang<sup>2</sup>

<sup>1</sup>National Key Laboratory of Science and Technology on Multispectral Information Processing, School of Automation, Huazhong University of Science and Technology (HUST), Wuhan 430074, China

<sup>2</sup>Beijing Aerospace Automatic Control Institute, Beijing 100854, China

Correspondence should be addressed to Lei Liu; liulei@hust.edu.cn

Received 7 June 2017; Revised 4 February 2018; Accepted 21 February 2018; Published 26 April 2018

Academic Editor: Angel Velazquez

Copyright © 2018 Xing Wei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Generation of optimal reentry trajectory for a hypersonic vehicle (HV) satisfying both boundary conditions and path constraints is a challenging task. As a relatively new swarm intelligent algorithm, an adaptive fireworks algorithm (AFWA) has exhibited promising performance on some optimization problems. However, with respect to the optimal reentry trajectory generation under constraints, the AFWA may fall into local optimum, since the individuals including fireworks and sparks are not well informed by the whole swarm. In this paper, we propose an improved AFWA to generate the optimal reentry trajectory under constraints. First, via the Chebyshev polynomial interpolation, the trajectory optimization problem with infinite dimensions is transformed to a nonlinear programming problem (NLP) with finite dimension, and the scope of angle of attack (AOA) is obtained by path constraints to reduce the difficulty of the optimization. To solve the problem, an improved AFWA with a new mutation strategy is developed, where the fireworks can learn from more individuals by the new mutation operator. This strategy significantly enhances the interactions between the fireworks and sparks and thus increases the diversity of population and improves the global search capability. Besides, a constraint-handling technique based on an adaptive penalty function and distance measure is developed to deal with multiple constraints. The numerical simulations of two reentry scenarios for HV demonstrate the validity and effectiveness of the proposed improved AFWA optimization method, when compared with other optimization methods.

## 1. Introduction

In recent decades, global strike and space transportation have spurred more and more interests in hypersonic vehicles (HVs) [1] for both military and civilian applications. The development of advanced guidance and control technologies for HV is promoted to meet the need for an effective and reliable access to the space. With aerodynamic control, the unpowered HV has the ability of reentering and gliding through the atmosphere. In order to steer an efficient and safety flight in the complex conditions, the reentry trajectory optimization problem of HV has been widely of concern. The aim of reentry trajectory optimization is to find an optimal solution under the reentry flight dynamics and physics, given the aerodynamics, structural strength, and the thermal

protection system [2]. Besides, as the reentry dynamics is highly nonlinear, the reentry trajectory optimization is a nonconvex problem with multiple constraints, such as the control ability, heating rate, dynamic pressure, and aerodynamic load. Thus, it is difficult to solve these problems analytically, and numerical techniques are required to determine an approximation to the continuous solution.

There are two main kinds of numerical methods to solve trajectory optimization problems: indirect methods and direct methods [3]. In indirect methods [4], the original problem is transformed into a two-point boundary problem by applying the calculus of variations or the Pontryagin maximum principle. It has high accuracy, and it ensures that the solution satisfies the necessary optimality conditions. However, in indirect methods, it is quite complicated to

derive the necessary optimality conditions. Besides, the radius of convergence is small, and it is difficult to guess the initial value of costate variable [5].

In order to overcome the disadvantages of indirect methods, direct methods have been developed, which are classified into two main types: direct shooting method and collocation methods. In the direct shooting method, only the control variables are parameterized and explicit numerical integration is used to satisfy the differential equation constraints. As an improvement, collocation methods [6–9] discretize both the states and control variables, and they use piecewise or global polynomials to approximate the differential equations at collocation points. Pseudospectral methods (PMs) are frequently used collocation methods, which include Gauss PM (GPM) [10, 11], Legendre PM (LPM), Radau PM (RPM) [12], and Chebyshev PM (CPM) [13]. By direct methods, the reentry trajectory optimization problem with infinite dimensions is transformed into the nonlinear programming problem (NLP) with finite dimensions, which is usually a high-dimensional multimodal nonsmooth problem. Some deterministic methods, such as sequential quadratic programming (SQP) [6, 14], second-order cone programming (SOCP) [15], and sequential convex programming method [16], are used to solve it. However, deterministic algorithms are sensitive, and they are easy to be stagnated at a local optimal point.

In the recent years, intelligent algorithms with the global searching ability, which are easy to implement, have been applied to solve the reentry trajectory optimization problems with the development of computational technology [14, 17–24]. In [17], a hybrid genetic algorithm (GA) collocation method was introduced for trajectory optimization, in which the initial guesses for the state and control variables are interpolated by the best solution of GA. Zhao and Zhou employed the particle swarm optimization (PSO) to obtain the end-to-end trajectory for hypersonic reentry vehicles in a quite simple formulation [19]. Zhang et al. applied the ant colony algorithm (ACO) to generate the optimal reentry trajectory for a reusable launch vehicle (RLV) [20]. Additionally, differential evolution (DE) [21, 22] that mimicked the process of nature evolution and simulated annealing (SA) inspired by annealing in metallurgy was also introduced to deal with reentry trajectory optimization problems.

Among these intelligent algorithms, the fireworks algorithm (FWA) is a relatively new swarm intelligence method firstly proposed by Tan and Zhu [25]. Inspired by the fireworks explosion, the algorithm selects some locations as the fireworks in space, each for exploding to generate a shower of sparks. Through the explosion procedure of the FWA, the diversity of population is enhanced. Besides, owing to the powerful local search capabilities and distributed parallel search mechanism, the FWA has a faster convergence speed compared with other intelligent algorithms.

The enhanced fireworks algorithm (EFWA) [26] is an improved version of the FWA in some operators. To improve the EFWA, the adaptive fireworks algorithm (AFWA) [27] was proposed with an adaptive explosion

amplitude, which is calculated according to the fitness of individuals adaptively. Afterwards, some improved FWA were proposed and applied in solving various practical optimization problems. For example, Gao and Ming [28] combined the cultural algorithm (CA) with the FWA for the digital filter design. Zheng et al. [29] developed a hybrid FWA with DE operators to improve the diversity and avoid prematurity. Based on a self-adaption principle and bimodal Gaussian function, Xue et al. [30] proposed an advanced FWA to design the PID controllers with high-quality performances. From the previous research, it is recognized that the information interaction among all individuals of the FWA is relatively poor, whereas the interaction is vital in swarm intelligence algorithms. Thus, when solving complex multimodal problem of the optimal reentry trajectory generation, it may be easy to get trapped in a local optimum. As far as we are concerned, the application of the FWA for reentry trajectory optimization is scarce and has not been found in the published literature.

In this paper, we focus on the improvement of the AFWA and its application to the reentry trajectory optimization problems. Firstly, an improved version of the AFWA (I-AFWA) is developed by combining the standard AFWA (S-AFWA) with a new mutation strategy. In each iteration of the algorithm, the new mutation operator and auxiliary mutation individual selection strategy are applied to make Gaussian fireworks learn from more individuals (not only the best individual) and generate diverse sparks from all fireworks and explosion sparks. The interactions of the fireworks and sparks are enhanced to further improve the diversity of the population and avoid being trapped in local optima too early. Then, the I-AFWA is applied to the reentry trajectory optimization problems. The problem is transformed into NLP by using Chebyshev polynomial interpolation to discretize the angle of attack (AOA) and bank angle simultaneously, and the scope of the AOA is figured out by path constraints to simplify the problem. Next, a constraint-handling technique based on the adaptive penalty function and the distance measure is proposed to incorporate with the I-AFWA and used to deal with the multiconstrained parameter optimization problem. Finally, two reentry scenarios are given to illustrate the validity and effectiveness of the proposed I-AFWA method on the reentry trajectory optimization problem.

The remainder of this paper is organized as follows. The general reentry trajectory optimization problem is formulated in Section 2. Section 3 proposes the I-AFWA with a new mutation strategy, and a constraint-handling technique is incorporated with the I-AFWA for the reentry trajectory optimization. In Section 4, two different reentry tasks for HV are presented to evaluate the proposed algorithm. A few conclusions are made in Section 5.

## 2. Problem Formulation

*2.1. Reentry Dynamics.* Using a spherical rotating Earth model, the three-degree-of-freedom (3DOF) point mass

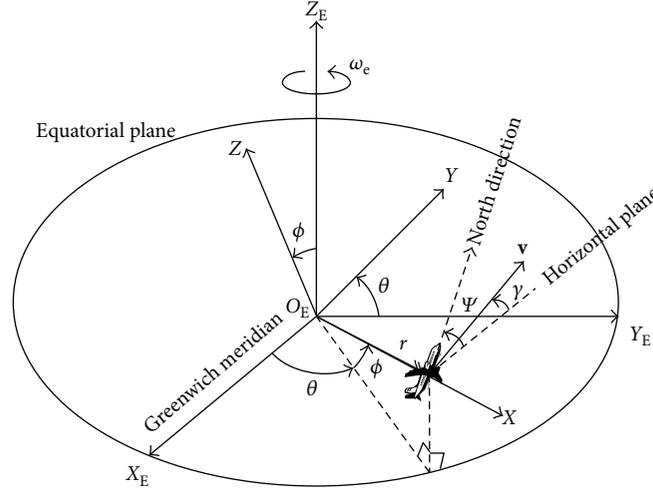


FIGURE 1: Geometric sketch of reentry flight.

dynamics equations [31] of a hypersonic reentry vehicle are described as follows:

$$\dot{r} = v \sin \gamma, \quad (1)$$

$$\dot{\theta} = \frac{v \cos \gamma \sin \psi}{r \cos \phi}, \quad (2)$$

$$\dot{\phi} = \frac{v \cos \gamma \cos \psi}{r}, \quad (3)$$

$$\dot{v} = -D - g \sin \gamma + \omega_e^2 r \cos \phi (\sin \gamma \cos \phi - \cos \gamma \sin \phi \cos \psi), \quad (4)$$

$$\dot{\gamma} = \frac{L \cos \sigma}{v} - \frac{g}{v} \cos \gamma + \frac{v}{r} \cos \gamma + 2\omega_e \cos \phi \sin \psi + \frac{\omega_e^2 r}{v} \cos \phi (\cos \gamma \cos \phi + \sin \gamma \cos \psi \sin \phi), \quad (5)$$

$$\dot{\psi} = \frac{L \sin \sigma}{v \cos \gamma} - 2\omega_e (\tan \gamma \cos \psi \cos \phi - \sin \phi) + \frac{v}{r} \cos \gamma \sin \psi \tan \phi + \frac{\omega_e^2 r}{v \cos \gamma} \sin \psi \sin \phi \cos \phi, \quad (6)$$

where  $r$  is the radial distance from the center of the Earth to the vehicle.  $\theta$  and  $\phi$  are the longitude and latitude, respectively.  $v$  is the Earth-relative velocity.  $\gamma$  is the velocity flight path angle and  $\psi$  is the velocity heading angle measured from the North in a clockwise direction.  $\sigma$  is the bank angle, which is the angle between the vehicle longitudinal symmetry plane and the vertical plane [2].  $\omega_e$  is the Earth's self-rotation angular velocity, and  $g = \mu/r^2$  is the gravitational acceleration where  $\mu$  is the gravitational constant.  $L$  and  $D$  are the aerodynamic lift force and drag force, respectively, which are given as follows:

$$L = \frac{1}{2m} \rho v^2 S_{\text{ref}} C_L(\alpha, M_a), \quad (7)$$

$$D = \frac{1}{2m} \rho v^2 S_{\text{ref}} C_D(\alpha, M_a),$$

where  $\rho = \rho_0 \exp(-h/h_s)$  is the density of atmosphere, with  $\rho_0$  as the density of atmosphere at sea level,  $h_s$  as the scalar height coefficient, and  $h$  as the altitude from the sea level.  $h = r - r_e$ , where  $r_e$  denotes the Earth's radius.  $S_{\text{ref}}$  and  $m$  are the aerodynamic reference area and mass of the vehicle, respectively.  $C_L$  and  $C_D$  are the lift and drag coefficients determined by the AOA  $\alpha$  and Mach number  $M_a = v/V_s(r)$ .  $\alpha$  denotes the angle between the relative velocity and a reference line fixed with respect to the vehicle body [2], which does not occur explicitly in the equations and appear through the aerodynamic forces  $L$  and  $D$ . Given the altitude  $h$ , velocity  $v$ , and the AOA  $\alpha$ , the Mach number  $M_a$  is calculated, and then the corresponding lift and drag coefficient  $C_L$  and  $C_D$  values are found by looking up the table functions with the value of  $M_a$  and  $\alpha$ . The independent variable is the time  $t$ , and control variables are  $\alpha$  and  $\sigma$ . The geometric sketch of reentry flight is shown in Figure 1.

## 2.2. Multiple Constraints

**2.2.1. Heating Rate.**  $\dot{Q}_s$  indicates the heating rate at a specified point on the surface of the reentry vehicle, which normally chooses the stagnation point on the nose. It is determined by the atmospheric density and Earth-relative velocity [31], which must be limited as follows:

$$\dot{Q}_s = k_Q \sqrt{\rho} v^c \leq \dot{Q}_{s \text{ max}}, \quad (8)$$

where  $k_Q$  is the heating rate normalization constant.  $c$  is the constant coefficient that typically takes the value of 3.0, 3.15, or 3.25.  $\dot{Q}_{s \text{ max}}$  represents the maximum allowable value of heating rate.

**2.2.2. Dynamic Pressure.** The dynamic pressure  $q$  is a typical path constraint, and it is limited to control the hinge moment of an actuator in a reasonable range. The peak value of the

dynamic pressure must be less than the maximum value  $q_{\max}$  as follows:

$$q = \frac{1}{2} \rho v^2 \leq q_{\max}. \quad (9)$$

**2.2.3. Aerodynamic Load.** In order to protect the mechanical system, the instantaneous value of aerodynamic load in the body-normal direction  $n_T$  is given lower than the maximum value  $n_{T \max}$  as follows:

$$n_T = L \cos \alpha + D \sin \alpha \leq n_{T \max}. \quad (10)$$

**2.2.4. Quasi-Equilibrium Glide Condition.** As the flight path angle is small and varies relatively slowly in a major portion of the lifting reentry trajectory [32], we set  $\gamma \approx 0$ ,  $\dot{\gamma} \approx 0$  and ignore the Earth's self-rotation in (5), and it generates

$$L \cos \sigma - g + \frac{v^2}{r} = 0. \quad (11)$$

This is the so called quasi-equilibrium glide condition (QEGC) [32], which guarantees that the reentry trajectory does not oscillate acutely for the convenience of the attitude controller design. It is a soft condition that forms the upper boundary of altitude-velocity reentry corridor. Additionally, hard constraints including heating rate, dynamic pressure, and aerodynamic load form the lower boundary of altitude reentry corridor. Therefore, the altitude-velocity reentry corridor is illustrated in Figure 2.

**2.2.5. Control Boundary.** To maintain the stability of the reentry vehicle, it is necessary to limit the two control variables of AOA and bank angle as follows:

$$\begin{aligned} \alpha_{\min} \leq \alpha \leq \alpha_{\max}, \quad |\dot{\alpha}| \leq \dot{\alpha}_{\max}, \\ \sigma_{\min} \leq \sigma \leq \sigma_{\max}, \quad |\dot{\sigma}| \leq \dot{\sigma}_{\max}. \end{aligned} \quad (12)$$

**2.2.6. Terminal Conditions.** Given the different reentry flight missions and conditions, the terminal states, including terminal altitude, position, velocity, flight path angle, and heading angle, are restrained as follows:

$$\begin{aligned} |r(t_f) - r_f| &\leq \Delta r, \\ |\theta(t_f) - \theta_f| &\leq \Delta \theta, \\ |\phi(t_f) - \phi_f| &\leq \Delta \phi, \\ |v(t_f) - v_f| &\leq \Delta v, \\ |\gamma(t_f) - \gamma_f| &\leq \Delta \gamma, \\ |\psi(t_f) - \psi_f| &\leq \Delta \psi, \end{aligned} \quad (13)$$

where  $t_f$  is the terminal time of the reentry phase, and  $r_f$ ,  $\theta_f$ ,  $\phi_f$ ,  $v_f$ ,  $\gamma_f$ , and  $\psi_f$  represent the ideal reentry terminal states, respectively.

**2.3. Reentry Trajectory Optimization Problem Formulation.** The objective of the reentry trajectory optimization problem is to design a reentry trajectory that can minimize (or

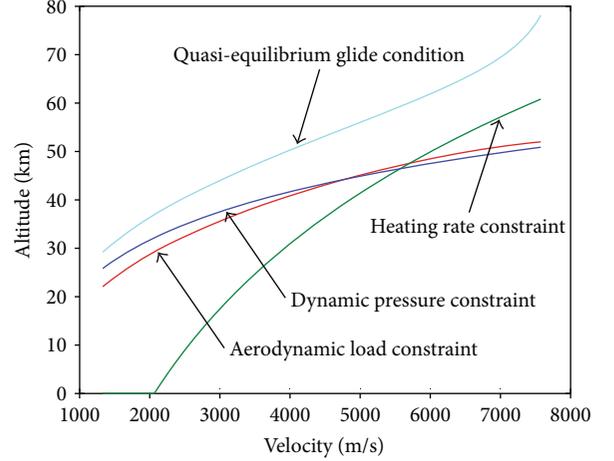


FIGURE 2: The sketch map of altitude-velocity reentry corridor.

maximize) the performance index, under multiple constraints. Hence, it is a multiconstrained nonlinear optimal control problem, and here, we consider it in the Bolza form [33].

The continuous Bolza problem is to determine the state  $\mathbf{x}(t) \in \mathbf{R}^n$  and the control  $\mathbf{u}(t) \in \mathbf{R}^m$  that minimize the Bolza objective function as follows:

$$J = \Phi(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} g(\mathbf{x}(t), \mathbf{u}(t), t) dt, \quad (14)$$

subject to the dynamics equations, boundary constraints, and path constraints, which are stated formally as follows:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t), \quad (15)$$

$$\phi(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) = \mathbf{0}, \quad (16)$$

$$\mathbf{C}(\mathbf{x}(t), \mathbf{u}(t), t) \leq \mathbf{0}, \quad (17)$$

where  $\Phi$  is the endpoint performance index and  $g$  is the integral performance index. In the reentry trajectory optimization problem,  $\mathbf{x} = [r, \theta, \phi, v, \gamma, \psi]^T$  is the state vector of the vehicle and  $\mathbf{u} = [\alpha, \sigma]^T$  is the control vector.  $t_0$  and  $t_f$  are the initial and terminal time, respectively, of the reentry flight.

Generally, the objective function is defined corresponding to the specified mission. For the reentry vehicle, the objective function can be selected to minimize downrange, to maximize crossrange, or to minimize total heat and so on.

The reentry trajectory optimization is formulated as an optimal control problem by (14), (15), (16), and (17), in which a dynamics model is represented by (1), (2), (3), (4), (5), and (6), multiple constraints are given in (8), (9), (10), (11), (12), and (13), and the performance index function is described by (14). Thus, an improved AFWA is developed to effectively solve the nonconvex optimal control problem under constraints.

### 3. Trajectory Optimization Algorithm Description

Reentry trajectory optimization problem is a continuous optimal control problem with infinite dimensions, so it is difficult to solve directly. To solve this problem, we first transform it to a finite dimensional NLP by the discretization of control curves with Chebyshev polynomial interpolation. However, it is also difficult to optimize the AOA and bank angle simultaneously, so we obtain the scope of AOA by path constraints ((8), (9), (10), and (11)). Subsequently, a constraint-handling technique based on an adaptive penalty function and a distance measure is proposed to deal with multiple constraints. Finally, a new improved AFWA with a new mutation strategy is developed to solve the NLP.

**3.1. Discretization of the Control Curves.** Substitute the expression of density  $\rho$  into path constraints of (8) and (9), and we obtain the following equations:

$$\begin{aligned} r &\geq r_e - 2h_s \ln \left( \frac{\dot{Q}_s \max}{k_Q \sqrt{\rho_0} v^c} \right) \triangleq r_{\dot{Q}_s}(v), \\ r &\geq r_e - h_s \ln \frac{2q_{\max}}{\rho_0 v^2} \triangleq r_q(v). \end{aligned} \quad (18)$$

Let  $f(r, v, \alpha, \sigma) = L \cos \sigma - g + v^2/r$ . Based on the QEGC in (11),  $f(r, v, \alpha, \sigma) = 0$ . Then, by substituting (7) into it, we obtain  $\alpha = \alpha(v, r, \sigma)$ . Moreover, the derivation of  $r$ ,  $\alpha$ , and  $\sigma$  can be calculated as follows:

$$\begin{aligned} \frac{\partial f}{\partial r} &= \frac{2g}{r} - \frac{v^2}{r^2} - \frac{L}{h_s} \cos \sigma + \frac{L}{C_L} \frac{\partial C_L}{\partial r} \cos \sigma, \\ \frac{\partial f}{\partial \alpha} &= \frac{\rho v^2 S_{\text{ref}}}{2m} \frac{\partial C_L}{\partial \alpha} \cos \sigma, \\ \frac{\partial f}{\partial \sigma} &= -L \sin \sigma. \end{aligned} \quad (19)$$

Since  $\sigma \in [-90^\circ, 90^\circ]$  and  $L > 0$ ,  $\partial f / \partial \sigma > 0$  when  $\sigma \in [-90^\circ, 0^\circ)$  and  $\partial f / \partial \sigma < 0$  when  $\sigma \in (0^\circ, 90^\circ]$ . Lift coefficient  $C_L$  increases as  $\alpha$  increases; that is,  $\partial C_L / \partial \alpha > 0$ ; thus,  $\partial f / \partial \alpha > 0$ . Consider  $\partial \alpha / \partial \sigma = -(\partial f / \partial \sigma) / (\partial f / \partial \alpha)$ ; then,  $\partial \alpha / \partial \sigma < 0$  when  $\sigma \in [-90^\circ, 0^\circ)$  and  $\partial \alpha / \partial \sigma > 0$  when  $\sigma \in (0^\circ, 90^\circ]$ . Furthermore,  $L/h_s$  is much larger than  $\partial C_L / \partial r$  and  $2g/r$ , so they can be ignored. As  $\partial f / \partial r \approx (-v^2/r^2) - (L \cos \sigma / h_s) < 0$  and  $\partial \alpha / \partial r = -(\partial f / \partial r) / (\partial f / \partial \alpha) > 0$ , we have

$$\begin{aligned} \alpha &= \alpha(v, r, \sigma) \geq \alpha(v, r, 0^\circ) \\ &\geq \alpha \left( v, \max \left( r_{\dot{Q}_s}(v), r_q(v) \right), 0^\circ \right) \triangleq \alpha_{\min}(v). \end{aligned} \quad (20)$$

Denote  $\alpha_{\max}$ ,  $\sigma_{\max}$ , and  $\sigma_{\min}$  as the maximum allowable AOA, maximum allowable bank angle, and minimum allowable bank angle, respectively. Then, the scopes of control variables are determined. With the scopes, the discretization process is conducted in the following.

In the approximation of the original function, polynomial interpolation at the zeros of Chebyshev orthogonal polynomial can minimize the maximum error in interpolation

intervals. Thus, Chebyshev polynomial interpolation is the optimal approximation of the original function in all polynomial interpolations in  $L_\infty$  norm sense. Hence, the control curves are approximated by Chebyshev polynomial interpolation as follows:

$$\begin{aligned} \eta(\tau) &= \sum_{k=1}^{N_\alpha} \eta_k^{N_\alpha} l_k^{N_\alpha}(\tau), \\ \xi(\tau) &= \sum_{k=1}^{N_\sigma} \xi_k^{N_\sigma} l_k^{N_\sigma}(\tau), \\ \alpha(v) &= (1 - \eta(\tau)) \alpha_{\min}(v) + \eta(\tau) \alpha_{\max}, \\ \sigma(v) &= (1 - \xi(\tau)) \sigma_{\min} + \xi(\tau) \sigma_{\max}, \end{aligned} \quad (21)$$

where  $\tau = 2(v - v_0) / (v_f - v_0) - 1 \in [-1, 1]$ ,  $l_k^{N_\alpha}(\tau)$  is the  $k$ th interpolation basis function of  $N_\alpha$ -order Chebyshev interpolation polynomial, and  $l_k^{N_\sigma}(\tau)$  is the  $k$ th interpolation basis function of  $N_\sigma$ -order Chebyshev interpolation polynomial.  $N_\alpha$  and  $N_\sigma$  are the orders of the interpolation polynomial  $\alpha$  and  $\sigma$ , respectively.

Set  $\mathbf{X} = [\eta_1^{N_\alpha}, \eta_2^{N_\alpha}, \dots, \eta_{N_\alpha}^{N_\alpha}, \xi_1^{N_\sigma}, \xi_2^{N_\sigma}, \dots, \xi_{N_\sigma}^{N_\sigma}, \eta_k^{N_\alpha}, \xi_k^{N_\sigma} \in [0, 1]$ , in which  $\mathbf{X}$  is the decision vector of the transformed NLP. If  $\mathbf{X}$  is obtained, control curves are determined by (21), and then we can integrate (1), (2), (3), (4), (5), and (6) to figure out the state curves, objective function, and constraints. In this way, the NLP is solved.

**3.2. Adaptive Fireworks Algorithm.** In order to solve the NLP and obtain the optimal reentry trajectory, it is necessary to develop an effective intelligent optimization algorithm. The FWA [25] is a novel swarm intelligence algorithm inspired by fireworks explosion. Later, the AFWA is proposed with an adaptive explosion amplitude to improve the local search capability. It is efficient for unconstrained parameter optimization problem formulated to minimize  $D$ -dimensional function as follows:

$$\begin{aligned} \text{minimize } & f(\mathbf{X}), \mathbf{X} = [x_1, x_2, \dots, x_D], \\ & x_k \in [x_{k,\min}, x_{k,\max}], \quad k = 1, 2, \dots, D, \end{aligned} \quad (22)$$

where  $x_{k,\min}$  and  $x_{k,\max}$  are the lower and upper boundaries, respectively, of the search space in dimension  $k$ . Besides,  $\mathbf{X}_i = [x_1^i, x_2^i, \dots, x_D^i]$ ,  $i = 1, 2, \dots, n$ , represents the position of the  $i$ th firework, where  $n$  is the size of fireworks. The process of the AFWA is as follows.

**3.2.1. Initialization.** In the initialization stage,  $n$  locations are randomly generated in the search space as the fireworks of the first generation as follows:

$$\begin{aligned} x_k^i &= x_{k,\min} + \text{rand}[0, 1] \cdot (x_{k,\max} - x_{k,\min}), \\ & i = 1, 2, \dots, n, \quad k = 1, 2, \dots, D, \end{aligned} \quad (23)$$

where  $\text{rand}[0, 1]$  is the uniform distribution over interval  $[0, 1]$ .

### 3.2.2. Calculation of Spark Number and Explosion Amplitude.

For each firework, a certain number of sparks are generated within a certain explosion amplitude. The number of sparks and explosion amplitude are calculated according to their fitness. The better the firework fitness is, the more sparks it will generate and the smaller its amplitude is, vice versa.

The number of explosion sparks for each firework is calculated as follows:

$$S_i = \text{round} \left( m \cdot \frac{F(\mathbf{X}^\#) - F(\mathbf{X}_i) + \varepsilon}{\sum_{j=1}^n (F(\mathbf{X}^\#) - F(\mathbf{X}_j)) + \varepsilon} \right), \quad (24)$$

where  $m$  is the total number of explosion sparks generated by  $n$  fireworks,  $\mathbf{X}^\#$  stands for the firework with the worst (maximum) fitness among  $n$  fireworks,  $F(\mathbf{X}_i)$  refers to the fitness value of the  $i$ th firework, and  $\varepsilon$  denotes the smallest constant to avoid zero-division-error.

In order to avoid the overwhelming effects of splendid fireworks, the number of sparks is bounded as follows:

$$S_i = \begin{cases} S_{\min}, & \text{if } S_i < S_{\min}, \\ S_{\max}, & \text{else if } S_i > S_{\max}, \\ S_i, & \text{else,} \end{cases} \quad (25)$$

where  $S_{\min}$  and  $S_{\max}$  are the lower bound and upper bound, respectively, for the spark numbers.

The explosion amplitudes of the fireworks (except for the optimal firework, i.e., the firework with the best fitness) are calculated as follows:

$$A_i = A \cdot \frac{F(\mathbf{X}_i) - F(\mathbf{X}^*) + \varepsilon}{\sum_{j=1}^n (F(\mathbf{X}_j) - F(\mathbf{X}^*)) + \varepsilon}, \quad (26)$$

where  $A$  is the maximum explosion amplitude and  $\mathbf{X}^*$  stands for the firework with the best (minimum) fitness among the  $n$  fireworks.

The initial explosion amplitude of the optimal firework is  $x_{k,\max} - x_{k,\min}$ ,  $k = 1, 2, \dots, D$ .

**3.2.3. Generation of Explosion Sparks.** The explosion sparks are generated randomly within the hypersphere with the firework as its center and the amplitude as its radius. In explosion, sparks may undergo the effects of explosion from random directions (dimensions). The affected dimensions are randomly selected by  $z_k = \text{round}(\text{rand}[0, 1])$ ,  $k = 1, 2, \dots, D$ . For the selected dimensions (those dimensions, where  $z_k$  equals to 1) of the  $i$ th firework, the different offset displacements are calculated by  $\Delta x_k^i = A_i \cdot \text{rand}[-1, 1]$ . The location of each spark  $\mathbf{X}_j$  generated by  $\mathbf{X}_i$  can be obtained by adding the displacement  $\Delta x_k^i$  to  $x_k^i$ , that is,

$$x_k^j = x_k^i + \Delta x_k^i, \quad j = 1, 2, \dots, S_i, k \in Z, \quad (27)$$

where  $Z$  denotes the set of the preselected dimensions of the  $i$ th firework.

**3.2.4. Generation of Gaussian Mutation Sparks.** To keep the diversity of sparks and avoid falling into the local optimum,

the Gaussian mutation process is introduced. Each selected firework stretches along the direction between the current location of the firework and the location of the best firework. For the selected dimensions of the randomly selected firework  $\mathbf{X}_i$ , the location of the Gaussian mutation spark  $\mathbf{X}_j$  is generated as follows:

$$x_k^j = x_k^i + (x_k^* - x_k^i) \cdot e, \quad j = 1, 2, \dots, \hat{m}, k \in Z, \quad (28)$$

where the offset displacement  $e = \text{Gaussian}(0, 1)$  is the Gaussian distributed with mean 0 and standard deviation 1, and it is utilized to define the coefficient of the mutation.  $\hat{m}$  is the number of Gaussian mutation sparks and  $x_k^*$  is the  $k$ th dimensional position of the best firework in the current generation.

**3.2.5. Mapping Rule.** When the location of a new generated spark  $\mathbf{X}_j$  (explosion spark or mutation spark) exceeds the search range in dimension  $k$ , this spark will be mapped to another location within the search space with uniform distribution according to

$$x_k^j = x_{k,\min} + \text{rand}[0, 1] \cdot (x_{k,\max} - x_{k,\min}). \quad (29)$$

**3.2.6. Calculation of the Adaptive Amplitude of the Optimal Firework.** The optimal firework in the next generation is the best individual found (could be a spark generated or a firework) in this generation. To calculate the adaptive amplitude, we choose an individual and use its distance to the best individual as the amplitude of the optimal firework in the next explosion. The chosen individual should subject to the following conditions:

- (1) Its fitness is worse than that of the optimal firework in this generation.
- (2) Its distance to the best individual is minimal among all individuals satisfying 1.

Namely,

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X}_i} (d(\mathbf{X}_i, \mathbf{X}_{G+1}^*)), \quad (30)$$

with constraint  $f(\mathbf{X}_i) > f(\mathbf{X}_G^*)$ , where  $\mathbf{X}_{G+1}^*$  denotes the best individual among all sparks and fireworks in generation  $G$ , namely, the optimal firework of generation  $G + 1$ , and  $\mathbf{X}_G^*$  is the optimal firework of generation  $G$ .  $\mathbf{X}_i$  here represents all individuals in this generation.  $d$  is a certain measurement of distance.

Then, the adaptive amplitude of the optimal firework in generation  $G + 1$  is calculated as follows:

$$A_{G+1}^* = 0.5 \cdot \left( A_G^* + \lambda \cdot \|\hat{\mathbf{X}} - \mathbf{X}_{G+1}^*\|_\infty \right), \quad (31)$$

where  $A_G^*$  and  $A_{G+1}^*$  represent the adaptive amplitude in generation  $G$  and  $G + 1$ , respectively.  $\|\cdot\|_\infty$  denotes the infinity norm, which is used as the distance measure, namely, the maximum difference among all dimensions.  $\lambda$  is a certain coefficient (usually bigger than 1) that is

used to further slow down the convergence rate and improve the global search.

**3.2.7. Selection.** Elitism-random selection method [34] is used for the selection. The best individual will be selected first as the optimal firework in the next generation. Then, the other individuals are selected randomly.

**3.3. Constraint Handling.** In this subsection, we consider how to deal with the various constraints of the optimal reentry trajectory generation. Typically, a penalty function is added to the objective function to solve the constrained parameter optimization problem. However, it is difficult to choose appropriate weight coefficient for each constraint. In this case, a distance measure method and an adaptive penalty function are incorporated to form a fitness function to deal with multiple constraints and check dominance in the population. The values of the two functions vary with the sum of constraint violations and the objective function value of an individual.

**3.3.1. Distance Measure.** The distance measure is obtained by introducing the effect of an individual's constraint violation into its objective function. Firstly, the equality constraints defined in (16) are transformed into the following inequality constraints:

$$|\phi(\mathbf{X})| - \delta \leq 0, \quad (32)$$

where  $\delta$  is the small constant that represents the tolerance value for equality constraints.

Since the magnitude of an individual's constraint value and its objective function are different, it is necessary to normalize the two indexes. With the maximum and minimum objective function values of the current generation, the normalized objective function is described as follows:

$$\tilde{f}(\mathbf{X}) = \frac{f(\mathbf{X}) - f_{\min}}{f_{\max} - f_{\min}}. \quad (33)$$

The constraint violation  $v(\mathbf{X})$  of an individual is then calculated as the summation of the normalized violations of each constraint divided by the total number of constraints.

$$v(\mathbf{X}) = \frac{1}{\hat{n}} \sum_{i=1}^{\hat{n}} \frac{c_i(\mathbf{X})}{c_{i,\max}}, \quad (34)$$

with the constraint violation that is defined as follows:

$$c_i(\mathbf{X}) = \begin{cases} \max(0, C_i(\mathbf{X})), & i = 1, 2, \dots, l, \\ \max(0, |\phi_i(\mathbf{X})| - \delta), & i = l + 1, \dots, \hat{n}, \end{cases} \quad (35)$$

where  $c_{i,\max} = \max_{\mathbf{X}} c_i(\mathbf{X})$  is the maximum value of the  $i$ th constraint violation.  $\hat{n}$  is the total number of constraints,  $l$  is the number of inequality constraints, and  $\hat{n} - l$  is the number of equality constraints. If the individual  $\mathbf{X}$  satisfies the  $i$ th constraint, then the constraint violation  $c_i(\mathbf{X})$  is set to zero; otherwise,  $c_i(\mathbf{X})$  will be the actual constraint value, which is greater than zero. Particularly, if  $\mathbf{X}$  is a feasible individual, namely, all the constraints of  $\mathbf{X}$  are satisfied,

then  $c_i(\mathbf{X}) = 0, i = 1, \dots, \hat{n}$ , and  $c_{i,\max} = 0$ , which make the constraint violation in (34) run into the zero-division-error. Hence, we define  $v(\mathbf{X}) = 0$  in this case.

Thus, the distance measure for each individual is formulated with the objective value and constraint violation as follows:

$$d(\mathbf{X}) = \begin{cases} v(\mathbf{X}), & \text{if } r_f = 0, \\ \sqrt{\tilde{f}^2(\mathbf{X}) + v^2(\mathbf{X})}, & \text{otherwise,} \end{cases} \quad (36)$$

where  $r_f$  is the proportion of feasible solutions in the current population, and it is described as

$$r_f = \frac{\text{the number of feasible individuals in the current population}}{\text{the size of the current population}}, \quad (37)$$

**3.3.2. Adaptive Penalty Function.** Additionally, an adaptive penalty function is added to the fitness value of infeasible individuals, which contains two parts: one is based on the constraint violation and the other is based on the objective function. The weight between the two components is controlled by the number of feasible individuals in the current population. The adaptive penalty function is formulated as follows:

$$p(\mathbf{X}) = (1 - r_f)A(\mathbf{X}) + r_f B(\mathbf{X}), \quad (38)$$

with

$$A(\mathbf{X}) = \begin{cases} 0, & \text{if } r_f = 0, \\ v(\mathbf{X}), & \text{otherwise,} \end{cases} \quad (39)$$

and

$$B(\mathbf{X}) = \begin{cases} 0, & \text{if } \mathbf{X} \text{ is a feasible individual,} \\ \tilde{f}(\mathbf{X}), & \text{otherwise.} \end{cases} \quad (40)$$

The weight  $r_f$  in the adaptive penalty function allows the tradeoff between finding more feasible solutions and finding better solutions during the evolutionary process.

Thus, the final fitness function of each individual  $\mathbf{X}$  is formulated as the sum of the distance measure and penalty function.

$$F(\mathbf{X}) = d(\mathbf{X}) + p(\mathbf{X}). \quad (41)$$

The fitness function formulation of the optimal reentry trajectory generation is flexible, and it helps to solve the multiconstrained optimization problem more efficiently and effectively.

**3.4. Improved Adaptive Fireworks Algorithm.** Although there are some variants for the AFWA, premature convergence when solving complex multimodal problems of the optimal reentry trajectory generation is still the main deficiency of the AFWA. The search process of the AFWA indicates that the diversification mechanism is not very flexible, and in

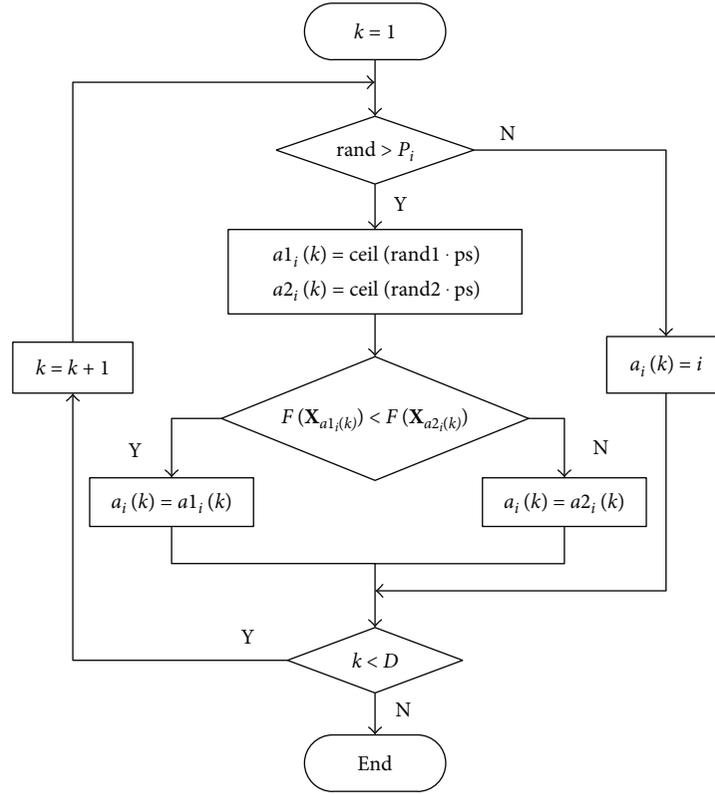


FIGURE 3: Selection procedure of auxiliary mutation individuals for the firework  $\mathbf{X}_i$  in all dimensions.

particular it does not utilize more information about other better quality solutions in the swarm. Especially, the interaction of fireworks in the explosion operator is not sufficient when obtaining the number and amplitude of the fireworks, which can be easily polarized. In other words, the best firework may generate an overwhelming number of sparks, while the other fireworks produce very few sparks. In the mutation operator, the interaction between the explosion sparks is ignored, and thus the diversity of sparks is reduced. Besides, the mutation sparks can hardly be passed down to the next generation [35]. For a  $D$ -dimensional optimization problem, the fitness of an individual may be determined by values in all dimensions, and an individual that has discovered the region corresponding to the global optimum in some dimensions may have a low fitness value because of the poor solutions in the other dimensions [36]. Thus, it is vital to generate more diverse sparks in all dimensions.

In order to make use of the beneficial information in the population more effectively to generate better quality solutions, a new mutation strategy is proposed to improve the S-AFWA, which can enhance the diversity of the population and improve the global search capability. In the new mutation strategy, the location of the mutation spark  $\mathbf{X}_j$  is generated by the selected firework  $\mathbf{X}_i$  in all dimensions as follows:

$$x_k^j = x_k^i + \left( x_k^{a_i(k)} - x_k^i \right) \cdot e, \quad j = 1, 2, \dots, \hat{m}, k = 1, 2, \dots, D, \quad (42)$$

where  $\mathbf{a}_i = [a_i(1), a_i(2), \dots, a_i(D)]$  defines which individual the selected firework  $\mathbf{X}_i$  should follow; that is,  $\mathbf{X}_i$  stretches along the direction between the selected individual and itself.  $x_k^{a_i(k)}$  is the corresponding  $k$ th dimension of any individual (all fireworks and explosion sparks, including itself), and the decision depends on the mutation probability  $P_i$ , which can take different values for different fireworks.

For each dimension of the firework  $\mathbf{X}_i$ , a random number is generated. If the random number is greater than  $P_i$ , the corresponding dimension will learn from another individual; otherwise, it will learn from itself. We employ a selection procedure of  $a_i(k)$  when the dimension of the firework learns from another individual as follows.

Firstly, two individuals (excluding the selected firework  $\mathbf{X}_i$ ) are randomly selected among the population. Subsequently, compare the fitness values of the two individuals and select the better one. At last, the winner is used as the exemplar to learn from for that dimension.

After the above selection procedure, if all exemplars in different dimensions of the firework are itself (i.e.,  $a_i(k) = i$  in all dimensions; due to that, all generated random numbers above are less than  $P_i$ ), we will randomly choose one dimension to learn from the corresponding dimension of another individual. The detailed procedure of choosing  $\mathbf{a}_i$  in (42) is given in Figure 3, where the ceil operator obtains the smallest integer greater than or equal to the specified expression and  $ps = n + m - 1$  is the population size.

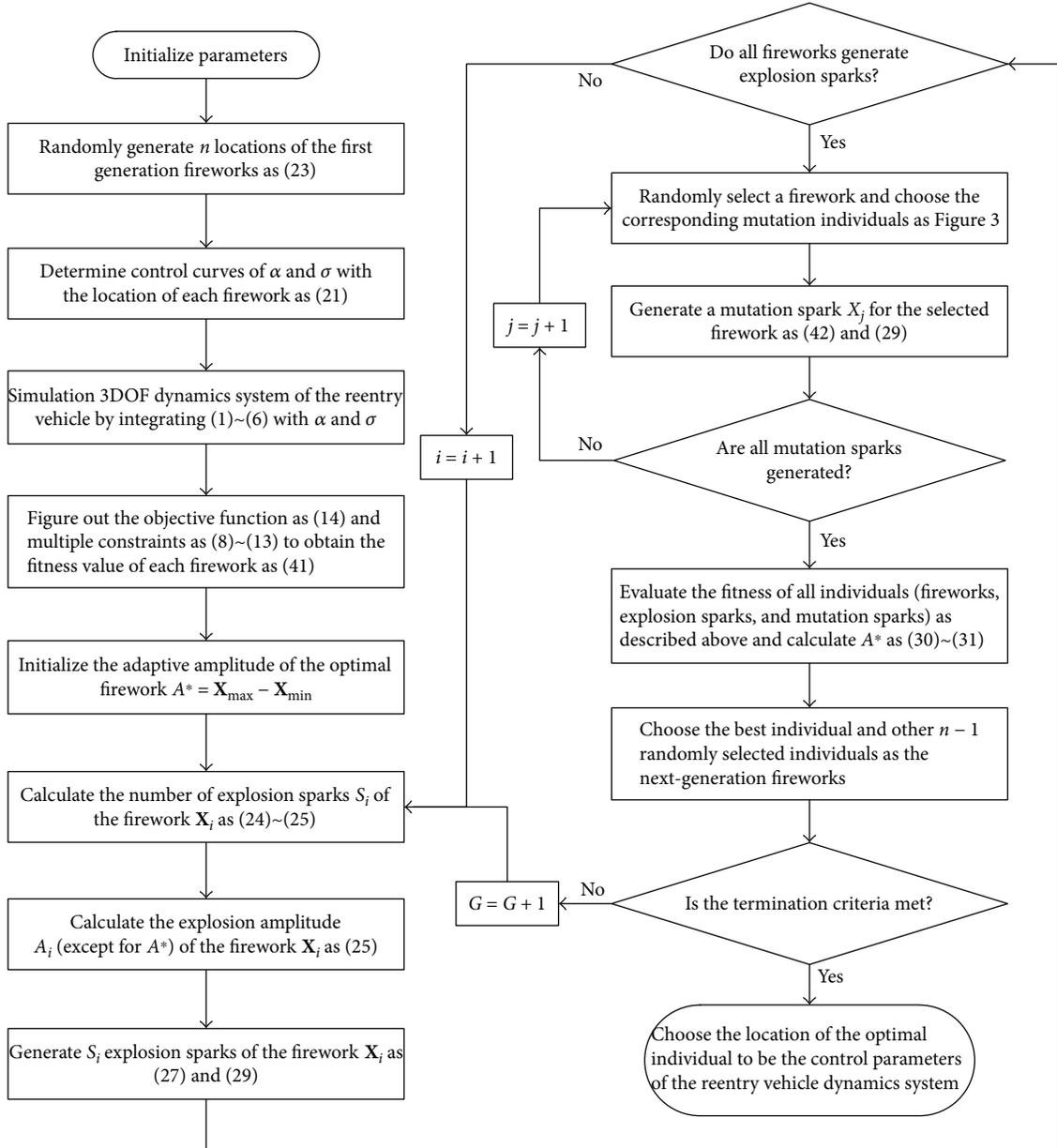


FIGURE 4: Reentry trajectory optimization based on the I-AFWA with a new mutation strategy.

TABLE 1: The main parameter settings of six intelligent algorithms.

Method	Main parameters
I-AFWA/S-AFWA/FWA	$n = 5, m = 50, S_{\max} = 40, S_{\min} = 4, \hat{m} = 5, A = (x_{k,\max} - x_{k,\min})/5, k = 1, 2, \dots, D, \lambda = 1.3$
DE	Mutation proportional coefficient $F = 0.7$ , crossover probability $C_R = 0.3$
PSO	Inertia weight $\omega_{\max} = 0.9, \omega_{\min} = 0.4$ , linear decreasing learning factor $c_1 = c_2 = 2.0$ , individual speed limitation $[-0.5, 0.5] \cdot (x_{k,\max} - x_{k,\min}), k = 1, 2, \dots, D$
GA	Binary coded chromosome length (8), crossover probability $P_c = 0.7$ , mutation probability $P_m = 0.02$

The new mutation strategy can generate new spark in the search space using the information from all the individuals, and thus the interaction between the fireworks and sparks

is enhanced, not limited to the selected firework and the best firework. The proposed mutation strategy exhibits a larger potential search space for mutation sparks than that of the

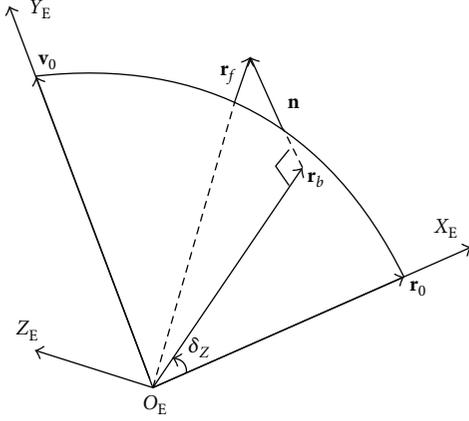


FIGURE 5: The schematic diagram of the reentry downrange.

S-AFWA, and thus the diversity is also enhanced. Especially when the best firework falls into the local optimum region, the mutation sparks have the increasing ability to jump out of the local optimum via the cooperative behavior of the whole population.

In order to address optimization problems in a general manner, each firework has a different mutation probability  $P_i$ . Therefore, fireworks have different levels of exploration and exploitation ability in the population, and they are able to solve diverse problems. We empirically set the mutation probability for each firework as follows:

$$P_i = 0.3 + 0.2 \cdot \frac{(\exp(10(i-1)/n-1) - 1)}{\exp(10) - 1}. \quad (43)$$

In detail, the procedure of reentry trajectory optimization based on the I-AFWA with a new mutation strategy is depicted in Figure 4.

#### 4. Simulation and Results

In this section, one HV model [32] is used to validate the proposed algorithm. The reentry scenario is set as follows:

$$\begin{aligned} [h_0, \theta_0, \phi_0, v_0, \gamma_0, \psi_0] &= [70 \text{ km}, 0^\circ, 0^\circ, 6500 \text{ m/s}, 0^\circ, 90^\circ], \\ [h_f, \theta_f, \phi_f, v_f, \gamma_f, \psi_f] &= [30 \text{ km}, 50^\circ, 0^\circ, 1600 \text{ m/s}, -3^\circ, 90^\circ], \\ [\Delta h, \Delta \theta, \Delta \phi, \Delta v, \Delta \gamma, \Delta \psi] &= [2 \text{ km}, 0.05^\circ, 0.05^\circ, 10 \text{ m/s}, 1^\circ, 5^\circ], \end{aligned} \quad (44)$$

where  $h = r - r_e$  and  $r_e = 6378 \text{ km}$  is the Earth's radius. Hence,  $r_0 = 6448 \text{ km}$  and  $r_f = 6408 \text{ km}$ .

In order to verify the effectiveness of the proposed algorithm, the I-AFWA is tested in comparison with the S-AFWA, DE, PSO, GA, and FWA. These algorithms all run 20 times repeatedly based on a similar simulation environment. The population size (spark size in the I-AFWA,

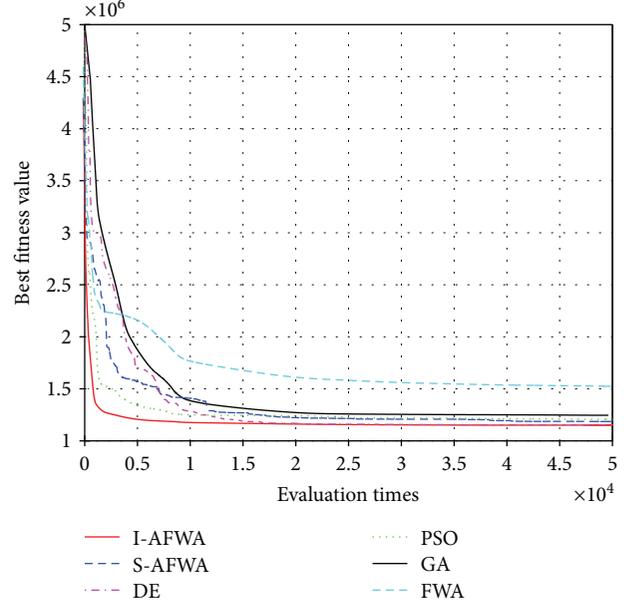


FIGURE 6: The average convergence curves of the fitness function for each algorithm in minimum downrange reentry trajectory optimization.

S-AFWA, and FWA and particle number in DE, PSO, and GA) is set as 50, and the maximum number of function evaluations  $G_{\max} = 100000$  for all algorithms. The AOA discrete point number  $N_\alpha = 5$ , and bank angle discrete point number  $N_\sigma = 10$ . Besides, other main parameters are shown in Table 1.

The hard path constraint limits and control boundaries remain fixed throughout all simulations:

$$\begin{aligned} \dot{Q}_s \max &= 5000 \text{ kW/m}^2, \\ q_{\max} &= 30 \text{ kPa}, \\ n_{\max} &= 5.0g, \\ 8^\circ &\leq \alpha \leq 30^\circ, \\ |\dot{\alpha}| &\leq 5^\circ/\text{s}, \\ |\sigma| &\leq 90^\circ, \\ |\dot{\sigma}| &\leq 30^\circ/\text{s}. \end{aligned} \quad (45)$$

In order to deal with multiple constraints, these six algorithms all adopt the constraint-handling technique proposed in this paper. The simulation of reentry trajectory optimization is conducted on two cases using different objective functions: minimum downrange and minimum total heat.

**4.1. Minimum Downrange Trajectory.** For convenience of description, the definition of the standard reentry longitudinal plane and related concepts is given. The schematic diagram of the downrange is presented in Figure 5, where

$\mathbf{r}_0$  and  $\mathbf{v}_0$  are the position vector and velocity vector, respectively, of the reentry point.  $O_E$  is the center of the Earth, and  $\mathbf{r}_f$  is the position vector of the reentry terminal point. The standard reentry longitudinal plane is a plane composed of  $\mathbf{r}_0$  and  $\mathbf{v}_0$ , and  $\mathbf{r}_b$  is the projection of  $\mathbf{r}_f$  on this plane. The vector  $\mathbf{n}$  is the unit normal vector of the plane. The motion in the standard longitudinal plane is called the longitudinal motion, and the motion deviating from the plane is called the lateral motion. Therefore, the reentry downrange  $Z$  is defined as the arc length of a circle with the radius of  $r_f$  corresponding to the angle  $\delta_Z$  between vectors  $\mathbf{r}_b$  and  $\mathbf{r}_0$ , which is formulated as follows:

$$Z = r_f \cdot \delta_Z = r_f \cdot \langle \mathbf{r}_b, \mathbf{r}_0 \rangle = r_f \cdot \arccos\left(\frac{\mathbf{r}_b \cdot \mathbf{r}_0}{|\mathbf{r}_b| \cdot |\mathbf{r}_0|}\right), \quad (46)$$

where the relevant parameters are defined as

$$\begin{aligned} \mathbf{r}_b &= \mathbf{r}_f - (\mathbf{r}_f \cdot \mathbf{n})\mathbf{n}, \quad \mathbf{n} = (\mathbf{v}_0 \times \mathbf{r}_0) / |\mathbf{v}_0 \times \mathbf{r}_0|, \\ \mathbf{r}_0 &= r_0 \cdot [\cos \phi_0 \cos \theta_0, \cos \phi_0 \sin \theta_0, \sin \phi_0], \\ \mathbf{r}_f &= r_f \cdot [\cos \phi_f \cos \theta_f, \cos \phi_f \sin \theta_f, \sin \phi_f], \\ \mathbf{v}_0 &= v_0 \cdot [\cos \theta_0 \cos \phi_0 \sin \gamma_0 - \sin \theta_0 \cos \gamma_0 \sin \psi_0 \\ &\quad - \cos \theta_0 \sin \phi_0 \cos \gamma_0 \cos \psi_0, \sin \theta_0 \cos \phi_0 \sin \gamma_0 \\ &\quad + \cos \theta_0 \cos \gamma_0 \sin \psi_0 \\ &\quad - \sin \theta_0 \sin \phi_0 \cos \gamma_0 \cos \psi_0, \sin \phi_0 \sin \gamma_0 \\ &\quad + \cos \phi_0 \cos \gamma_0 \cos \psi_0], \end{aligned} \quad (47)$$

where  $r_0, \theta_0, \phi_0, v_0, \gamma_0$ , and  $\psi_0$  represent the initial states of the reentry phase.

In the minimum downrange reentry trajectory optimization, the terminal constraints of the longitude, latitude, and heading angle need not be considered, and the objective function is given as

$$J = \min(Z). \quad (48)$$

The average convergence curves of the six comparative algorithms are illustrated in Figure 6. As we can see, in the minimum downrange reentry trajectory optimization, both the convergence speed and the final optimization result of our I-AFWA are better, when compared with those of the other five algorithms. Figure 7 presents the simulation results under running the I-AFWA 20 times, including the 3D trajectory, ground trajectory, control curves, and path constraint curves. It can be found that all constraints are satisfied and the reentry trajectory and control profiles are smooth throughout the flight, which indicate that our algorithm is quite efficient in handling multiple constraints.

The mean and standard deviation of the optimization performance index, path constraints, and terminal

constraints of minimum downrange reentry trajectory obtained by the six intelligent algorithms are presented in Table 2. The performance index of the minimum downrange generated by the I-AFWA is much better than that generated by the S-AFWA, DE, PSO, GA, and FWA, which demonstrates that the optimization precision of the I-AFWA is higher and our improvement strategy is effective. Besides, the distribution range of the minimum downrange is very small, which shows that the I-AFWA achieves better performance and robustness.

The minimum downrange of the reentry vehicle is 1151.8km, and the corresponding trajectory has a larger AOA, which can increase the aerodynamic drag force and decrease the lift-drag ratio, thus consuming more energy and reducing the maneuver capacity of reentry vehicles, resulting in smaller flight downrange.

**4.2. Minimum Total Heat Trajectory.** The objective function of minimum total heat reentry trajectory optimization problem is given as follows:

$$J = \min(Q) = \min\left(\int_{t_0}^{t_f} \dot{Q}_s dt\right). \quad (49)$$

Figure 8 presents the average convergence curves of the comparative algorithms. It can be seen that the I-AFWA also has the fastest convergence speed in the six algorithms. The simulation results using the I-AFWA, including the state and control profiles, are depicted in Figure 9. The minimum total heat and constraints of minimum total heat reentry trajectory with the six algorithms are compared in Table 3. The terminal constraints, such as the final altitude, longitude, latitude, velocity, flight path angle, and heading angle, are satisfied accurately. The heating rate, dynamic pressure, and aerodynamic load are less than the maximum allowable values strictly. The minimum total heat of the I-AFWA is much smaller than that of the other five algorithms, which proves once again that the optimization results of the I-AFWA are better. Additionally, the variance of the performance index is also relatively small.

From the simulation results, it can be seen that the minimum total heat of this reentry scenario is about  $1.47 \times 10^6$  kJ, maybe smaller in fact. The reentry trajectory with minimum total heat corresponds to larger AOA and smaller bank angle, where the aerodynamic lift force of the vehicle is relatively large, the flight altitude is higher, and the atmospheric density is lower, thus reducing the heating rate (in terms of (8)) and decreasing the total heat; when the velocity is lower (close to the end of the reentry phase), pull down the AOA, so that the flight height drops rapidly, and meanwhile lateral maneuver is carried out to consume excess energy, making the vehicle reach the reentry terminal states and complete the flight mission.

Tables 2 and 3 also give the run time of each algorithm in the last row. On the two objective functions, we can see that the run time of the I-AFWA is typically

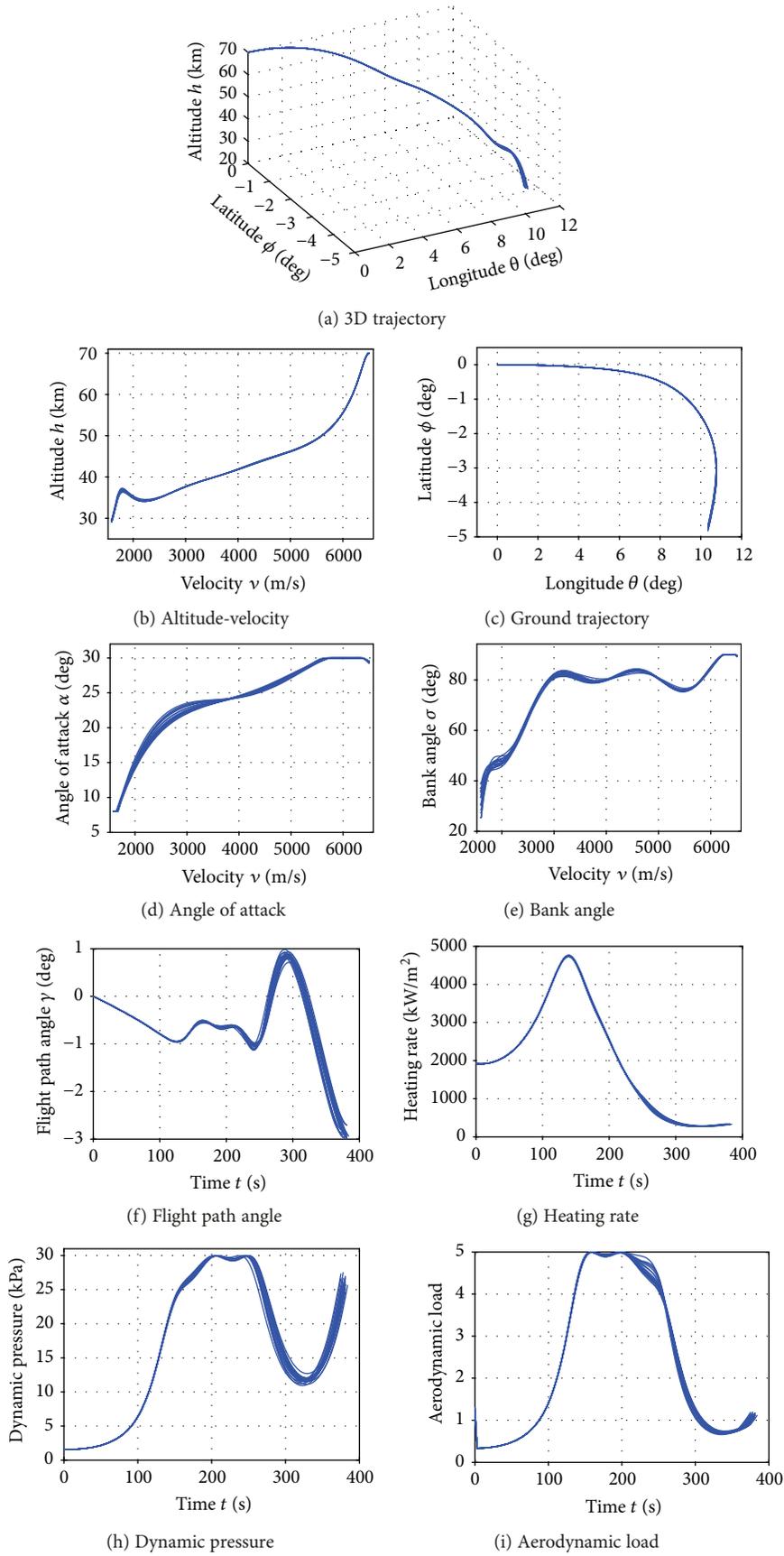


FIGURE 7: Simulation results of 20 times minimum downrange reentry trajectory optimization using the I-AFWA.

TABLE 2: Performance index and constraints in minimum downrange reentry trajectory optimization with the six comparative algorithms.

Evaluation index	I-AFWA		S-AFWA		DE		PSO		GA		FWA	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
$Z_f$ (km)	1152.1	0.8	1182.6	2.2	1155.8	1.5	1187.5	26.1	1240.3	1.8	1522.8	63.5
$\max(\dot{Q}_s)$ (kW/m <sup>2</sup> )	4974.1	9.8	4745.3	12.5	4899.3	38.1	4699.6	113.9	4830.5	59.2	4720.4	150.8
$\max(q)$ (kPa)	29.991	$7e-3$	29.469	0.036	29.840	0.071	29.955	0.121	29.535	0.096	28.737	0.334
$\max(n_T)$ (g)	4.995	$1e-3$	4.967	0.118	4.999	$3e-4$	4.996	$3e-3$	4.984	0.047	4.835	0.686
$ \Delta r(t_f) $ (km)	0.96	0.07	0.89	0.53	0.75	0.46	1.32	0.41	1.05	0.38	1.18	0.77
$ \Delta v(t_f) $ (m/s)	9.82	0.18	9.95	0.09	9.96	0.10	9.81	0.60	9.89	0.21	8.84	1.08
$ \Delta \gamma(t_f) $ (°)	0.197	0.102	0.784	0.173	0.879	0.084	0.886	0.124	0.913	0.236	0.656	0.455
Run time (s)	78.3	—	76.5	—	91.5	—	70.6	—	87.5	—	74.2	—

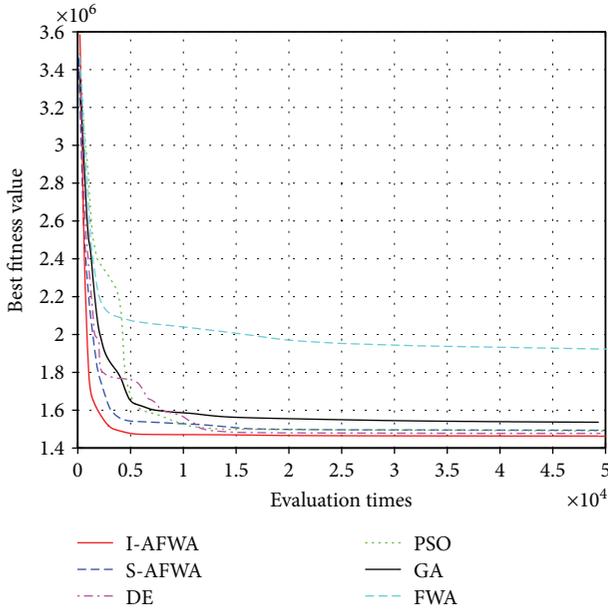


FIGURE 8: The average convergence curves of the fitness function for each algorithm in minimum total heat reentry trajectory optimization.

a little more than that of the S-AFWA, which is mainly because the I-AFWA takes more computational time on the selection of auxiliary mutation individuals. Note that the run time of PSO is the smallest and that of DE is the largest. However, for reentry trajectory optimization problem, compared with the long flight time, the little gap between the run times of the I-AFWA and S-AFWA is almost negligible. Although PSO runs much faster, the optimization result is relatively poor; DE is just the opposite. The running speed of the FWA is second only to that of PSO, but its optimization result is the worst. GA presents relatively poor performances in both two aspects.

In summary, the results of the numerical simulations demonstrate that the proposed new mutation strategy,

which enhances the information interactions between the fireworks and sparks to a great extent, can greatly improve the population diversity and thus avoid premature convergence during the search. Consequently, the I-AFWA can achieve a much better balance of exploration and exploitation than S-AFWA, DE, PSO, GA, and FWA and thus improve convergence speed and solution accuracy effectively.

## 5. Conclusion

In this paper, based on the AFWA and a new mutation strategy, a new optimization algorithm called the I-AFWA is proposed to generate an optimal trajectory for a hypersonic reentry vehicle under multiple constraints. By incorporating the new mutation operator and auxiliary mutation individual selection strategy to the S-AFWA, the information sharing among the fireworks and sparks is enhanced to generate more diverse individuals, and thus the global search capability is improved. Additionally, based on the distance measure and adaptive penalty function, a constraint-handling technique is developed to deal with all path and terminal constraints. The I-AFWA-based optimal reentry trajectory generation method takes all control variables into consideration, which makes it have high fidelity.

The simulation results of two different reentry missions are used to validate the reentry trajectory optimization algorithm, and some analyses are conducted. It is indicated that the I-AFWA exhibits better search efficiency and achieves better optimization performance than S-AFWA, DE, PSO, GA, and FWA. Besides, the convergence speed of the I-AFWA is also the fastest. The I-AFWA can also be applied to other optimization functions and other optimization problems. However, due to the increased computational cost caused by the selection process of auxiliary mutation individuals, the running speed of the I-AFWA is not the fastest one among all the algorithms. In the future research, the proposed algorithm will be further studied and tested on more complicated reentry scenarios involving waypoint and no-fly zone constraints.

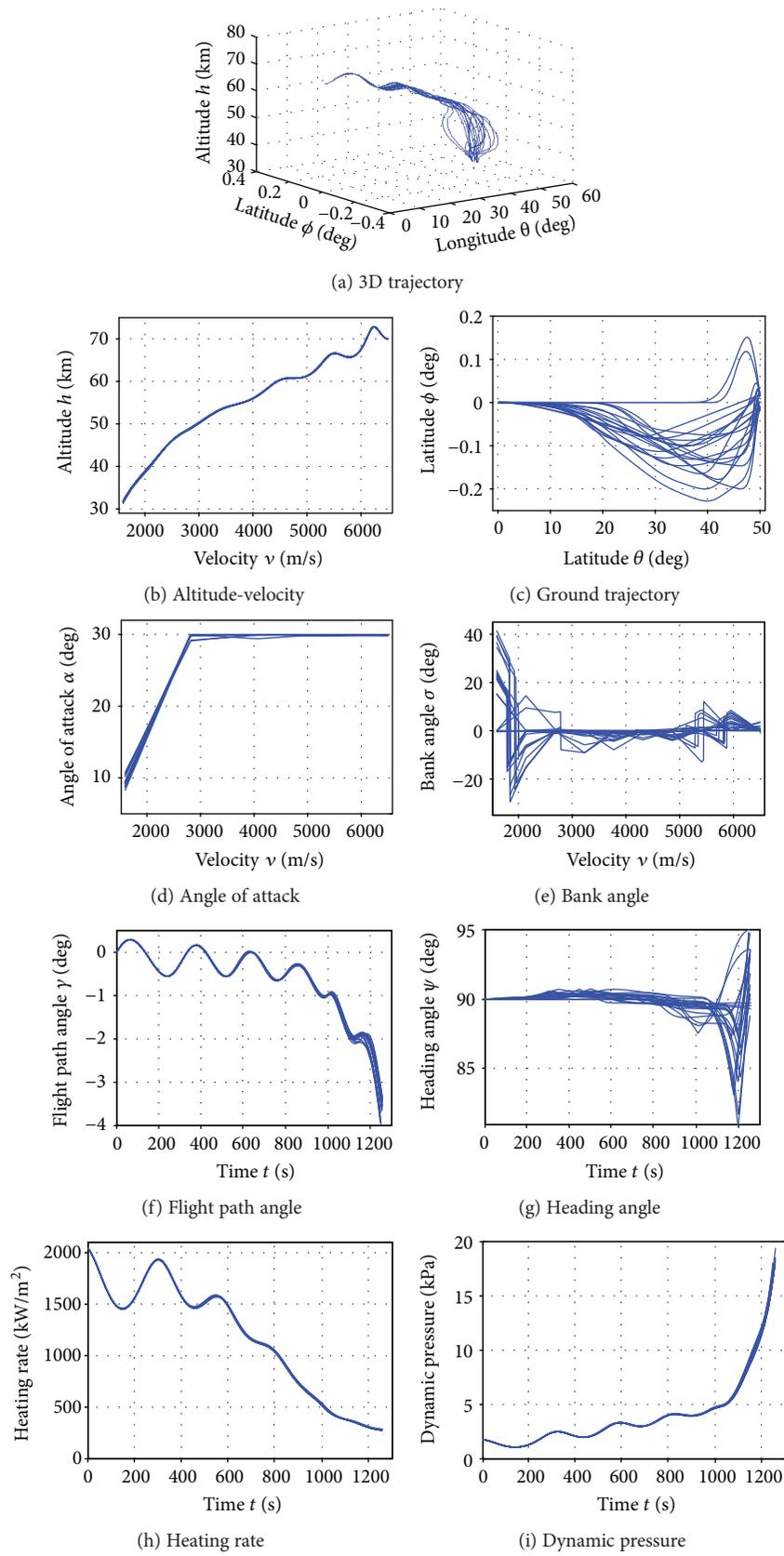


FIGURE 9: Simulation results of 20 times minimum total heat reentry trajectory optimization using the I-AFWA.

TABLE 3: Performance index and constraints of minimum total heat reentry trajectory optimization with the six comparative algorithms.

Evaluation index	I-AFWA		S-AFWA		DE		PSO		GA		FWA	
	Mean	Std										
$Q$ (kJ)	1.472e6	2.1e3	1.489e6	8.0e3	1.482e6	1.5e3	1.493e6	6.5e3	1.535e6	4.3e3	1.922e6	9.7e3
$\max(\dot{Q}_s)$ (kW/m <sup>2</sup> )	2037.9	7.8e-3	2452.2	8.2	2448.5	1.3	2532.8	11.8	2483.2	4.8	2756.5	32.8
$\max(q)$ (kPa)	18.053	0.342	24.439	3.0e-2	19.854	0.771	20.955	1.210	21.386	1.005	24.852	2.363
$\max(n_T)$ (g)	1.081	2.3e-2	2.883	0.375	1.975	0.126	2.293	0.531	2.132	0.154	2.379	0.612
$ \Delta r(t_f) $ (km)	1.70	0.12	0.95	0.61	1.78	0.06	1.03	0.55	1.43	0.23	1.15	0.64
$ \Delta \theta(t_f) $ (°)	2.50e-2	1.14e-2	3.78e-2	6.73e-3	3.52e-2	1.24e-2	3.89e-2	1.33e-2	4.06e-2	8.57e-3	3.12e-2	1.81e-2
$ \Delta \phi(t_f) $ (°)	1.53e-2	4.14e-3	3.34e-2	8.35e-3	2.48e-2	1.15e-2	3.14e-2	1.85e-2	4.37e-2	1.02e-2	2.89e-2	1.44e-2
$ \Delta v(t_f) $ (m/s)	5.52	0.28	7.95	2.23	7.63	0.87	8.32	0.43	6.68	0.95	7.32	1.76
$ \Delta \gamma(t_f) $ (°)	0.598	0.199	0.753	0.272	0.680	0.142	0.833	0.105	0.768	0.176	0.712	0.235
$ \Delta \psi(t_f) $ (°)	2.432	1.691	2.833	1.125	3.435	0.693	3.236	1.135	3.585	0.937	4.018	1.244
Run time (s)	258.4	—	250.5	—	287.1	—	232.8	—	274.6	—	242.7	—

## Conflicts of Interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work was supported in part by the National Nature Science Foundation of China (Grant nos. 61573161 and 61473124).

## References

- [1] B. Xu and Z. Shi, "An overview on flight dynamics and control approaches for hypersonic vehicles," *Science China Information Sciences*, vol. 58, no. 7, pp. 1–19, 2015.
- [2] N. X. Vinh, *Optimal Trajectories in Atmospheric Flight*, Elsevier, New York, NY USA, 1981.
- [3] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.
- [4] Q. Fang, X. Wang, C. Sun, and J. Yuan, "A shape-based method for continuous low-thrust trajectory design between circular coplanar orbits," *International Journal of Aerospace Engineering*, vol. 2017, Article ID 9234905, 10 pages, 2017.
- [5] K. Zhang and W. Chen, "Reentry vehicle constrained trajectory optimization," in *17th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, pp. 1–16, San Francisco, CA, USA, April 2011.
- [6] C. R. Hargraves and S. W. Paris, "Direct trajectory optimization using nonlinear programming and collocation," *Journal of Guidance, Control, and Dynamics*, vol. 10, no. 4, pp. 338–342, 1987.
- [7] P. Williams, "Hermite-Legendre-Gauss-Lobatto direct transcription in trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 4, pp. 1392–1395, 2009.
- [8] M. J. Poustini, R. Esmaelzadeh, and A. Adami, "A new approach to trajectory optimization based on direct transcription and differential flatness," *Acta Astronautica*, vol. 107, pp. 1–13, 2015.
- [9] L. Yang, W. Chen, X. Liu, and H. Zhou, "Steady glide dynamic modeling and trajectory optimization for high lift-to-drag ratio reentry vehicle," *International Journal of Aerospace Engineering*, vol. 2016, Article ID 3527460, 14 pages, 2016.
- [10] J. Zhao and R. Zhou, "Reentry trajectory optimization for hypersonic vehicle satisfying complex constraints," *Chinese Journal of Aeronautics*, vol. 26, no. 6, pp. 1544–1553, 2013.
- [11] W. Zhang, Y. Zhang, W. Li, and Y. Wang, "Path planning for rapid large-angle maneuver of satellites based on the gauss pseudospectral method," *Mathematical Problems in Engineering*, vol. 2016, Article ID 1081267, 7 pages, 2016.
- [12] D. Garg, M. A. Patterson, C. Francolin et al., "Direct trajectory optimization and costate estimation of finite-horizon and infinite-horizon optimal control problems using a Radau pseudospectral method," *Computational Optimization and Applications*, vol. 49, no. 2, pp. 335–358, 2011.
- [13] X. Guo and M. Zhu, "Direct trajectory optimization based on a mapped Chebyshev pseudospectral method," *Chinese Journal of Aeronautics*, vol. 26, no. 2, pp. 401–412, 2013.
- [14] D. Zhang and Y. Liu, "RLV reentry trajectory optimization through hybridization of an improved GA and a SQP algorithm," in *AIAA Guidance, Navigation, and Control Conference*, pp. 1–11, Portland, OR, USA, August 2011.
- [15] X. Liu, Z. Shen, and P. Lu, "Entry trajectory optimization by second-order cone programming," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 2, pp. 227–241, 2016.
- [16] Z. Wang and M. J. Grant, "Constrained trajectory optimization for planetary entry via sequential convex programming," in *AIAA Atmospheric Flight Mechanics Conference*, pp. 1–23, Washington, DC, USA, June 2016.
- [17] K. Subbarao and B. M. Shippey, "Hybrid genetic algorithm collocation method for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 4, pp. 1396–1403, 2009.
- [18] A. Rahimi, K. Dev Kumar, and H. Alighanbari, "Particle swarm optimization applied to spacecraft reentry trajectory," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 1, pp. 307–310, 2013.
- [19] J. Zhao and R. Zhou, "Particle swarm optimization applied to hypersonic reentry trajectories," *Chinese Journal of Aeronautics*, vol. 28, no. 3, pp. 822–831, 2015.
- [20] Q. Zhang, C. Liu, B. Yang, and Z. Ren, "Reentry trajectory planning optimization based on ant colony algorithm," in *2007 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1064–1068, Sanya, China, December 2007.
- [21] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [22] S. Chang, Y. Wang, L. Liu, and D. Zhao, "Reentry trajectory optimization based on differential evolution," *World Academy of Science, Engineering and Technology*, vol. 80, pp. 597–601, 2011.
- [23] Q. Zhang, Z. Wang, F. Tao, B. R. Sarker, and L. Cheng, "Design of optimal attack-angle for RLV reentry based on quantum particle swarm optimization," *Advances in Mechanical Engineering*, vol. 6, article 352983, 2015.
- [24] Z. Su and H. Wang, "A novel robust hybrid gravitational search algorithm for reusable launch vehicle approach and landing trajectory optimization," *Neurocomputing*, vol. 162, pp. 116–127, 2015.
- [25] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," in *International Conference in Swarm Intelligence (ICSI)*, pp. 355–364, Beijing, China, June 2010.
- [26] S. Zheng, A. Janecek, and Y. Tan, "Enhanced fireworks algorithm," in *2013 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2069–2077, Cancun, YUC, Mexico, June 2013.
- [27] J. Li, S. Zheng, and Y. Tan, "Adaptive fireworks algorithm," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 3214–3221, Beijing, China, July 2014.
- [28] H. Gao and M. Diao, "Cultural firework algorithm and its application for digital filters design," *International Journal of Modelling, Identification and Control*, vol. 14, no. 4, pp. 324–331, 2011.
- [29] Y.-J. Zheng, X.-L. Xu, H.-F. Ling, and S.-Y. Chen, "A hybrid fireworks optimization method with differential evolution operators," *Neurocomputing*, vol. 148, pp. 75–82, 2015.
- [30] J. Xue, Y. Wang, H. Li, X. Meng, and J. Xiao, "Advanced fireworks algorithm and its application research in PID parameters tuning," *Mathematical Problems in Engineering*, vol. 2016, Article ID 2534632, 9 pages, 2016.

- [31] N. X. Vinh, A. Busemann, and R. D. Culp, *Hypersonic and Planetary Entry Flight Mechanics*, University of Michigan Press, Ann Arbor, MI, USA, 1980.
- [32] Z. Shen and P. Lu, "Onboard generation of three-dimensional constrained entry trajectories," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 1, pp. 111–121, 2003.
- [33] J. Z. Ben-Asher, *Optimal Control Theory with Aerospace Applications*, American Institute of Aeronautics and Astronautics, Reston, VA, USA, 2010.
- [34] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*, John Wiley & Sons, New York, NY USA, 2006.
- [35] C. Yu, J. Li, and Y. Tan, "Improve enhanced fireworks algorithm with differential mutation," in *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 264–269, San Diego, CA, USA, October 2014.
- [36] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 3, pp. 281–295, 2006.



**Hindawi**

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

