Fast Trajectory Optimization for Gliding Reentry Vehicle Based on Improved Sparrow Search Algorithm

Zhanyuan Jiang1, a, Jianquan Ge2, b*, Qiangqiang Xu1, c, Tao Yang1, d
1National University of Defense Technology, Changsha, Hunan, China
2Space Engineering University, Beijing, China
aemail: 2003010218@st.btbu.edu.cn,  c email: xuqiangqiang12@126.com
demail: 1149979847@qq.com
*bcorresponding author: bemail: jiangzhanyuan1212@163.com

Abstract: In order to solve the problem of low convergence accuracy and easy to fall into local optimization when solving the reentry vehicle trajectory optimization problem for existing algorithms, an improved sparrow search algorithm (OTRSSA) is proposed. Firstly, the basic sparrow search algorithm is improved by the methods of opposition-based learning, adaptive T-distribution and random walk to improve the optimization accuracy and stability, and increase the global search ability, and the performance verification is carried out in 12 benchmark functions; secondly, the 3-DOF motion model of reentry trajectory optimization problem is established and transformed into multi-dimensional function optimization problem; finally, OTRSSA is applied to solve the trajectory optimization problem. The simulation results show that the optimization performance and convergence speed of OTRSSA are better, and a reentry trajectory with the farthest range and satisfying constraints can be obtained quickly.

1. Introduction
Reentry vehicle has the advantages of high speed, long range and strong maneuverability[1], reentry flight environment is complex, and high-speed reentry flight is greatly affected by heat flux, overload and dynamic pressure, the problem of trajectory optimization has always been a research hotspot.

There are two kinds of methods for reentry trajectory optimization: indirect method and direct method. Based on the classical variational method or Pontryagin minimum principle, the indirect method transforms the optimal control problem into a Hamiltonian two-point boundary value problem, which is sensitive to the initial value and difficult to converge, literature [2-4] uses indirect method to solve such problems. The direct method is to discretize and parameterize the variables in the optimal control problem, so as to transform the optimal control problem into a nonlinear programming (NLP) problem, and then solve it with numerical optimization method. In direct method, Gaussian pseudo spectrum method uses Gaussian pseudo Spectrum Approximation parameterization to divide the trajectory into several segments, so that all points on the flight path meet the complex constraints [5-6]. In recent years, various heuristic algorithms have been proposed and applied to reentry trajectory optimization. In reference [7], the decimal ant colony algorithm with local search strategy is used to realize the trajectory optimization design of minimizing the total heat absorption under overload constraints; in reference [8], the reentry trajectory optimization problem was solved by the improved chicken swarm algorithm with fixed angle of attack profile and discrete pitch angle; in reference [9],
the artificial bee colony algorithm is used to discretize the angle of attack at Legendre Gaussian collocation points to realize the direct configuration of trajectory optimization.

Sparrow search algorithm was proposed by Xue et al. In 2020, compared with other algorithms, it has the advantages of good stability and high convergence accuracy [10]. In reference [11], sparrow search algorithm is applied to the path planning of UAV, and the constrained path under time cooperation is obtained. But the basic sparrow search algorithm also has some problems, such as easy to fall into local optimum, long search time and so on. This paper proposes a sparrow search algorithm based on opposition-based learning, adaptive distribution and random walk strategy (OTRSSA), which can increase the search space, jump out of the local optimal solution and quickly search for the global optimal solution.

2. The improved sparrow search algorithm

2.1. Basic sparrow search algorithm

SSA algorithm is a new swarm intelligence optimization algorithm inspired by Sparrow's foraging and anti predation behavior. It divides the whole population into discoverer, follower and early warning, and they update their positions according to their respective ways.

The location of the discoverers is about 10% – 20% of the population, and the location renewal mode is as follows:

$$X_{\text{dis}}^{t+1} = \begin{cases} X_{t} \times \exp \left( \frac{t}{\xi \times \text{iter}_{\text{max}}} \right) & R_{t} < ST \\ X_{t} + q \times L & R_{t} \geq ST \end{cases}$$  \hspace{1cm} (1)

Where: \(t\) is the number of iterations, \(\text{iter}_{\text{max}}\) is the maximum number of iterations, \(X_{\text{dis}}^{t}\) represents the position of the \(i\)-th sparrow after iteration \(t+1\), \(\xi \in (0,1)\) is a random number with uniform distribution, \(R_{t} \in (0,1)\) indicates the warning value, \(ST \in (0.5,1)\) indicates a safe value, \(q\) is a random number that obeys normal distribution, \(L\) represents a matrix with size \(1 \times d\) and elements \(1\).

In addition to the discoverer, the rest of the sparrows are followers, which are updated according to the following formula:

$$X_{i}^{t+1} = \begin{cases} q \times \exp \left( \frac{X_{\text{worst}} - X_{i}^{t}}{i^{2}} \right) & i > \frac{n}{2} \\ X_{p}^{t} + \left| X_{i}^{t} - X_{p}^{t} \right| \times A_{\times} \times L & i \leq \frac{n}{2} \end{cases}$$  \hspace{1cm} (2)

Where: \(X_{p}\) indicates the best location for the discoverer, \(X_{\text{worst}}\) indicates the worst position, \(A\) is a vector of \(1\) and \(-1\).

The early warning persons account for about 10% – 20% of the population, and their location is updated as follows:

$$X_{i}^{t+1} = \begin{cases} X_{\text{best}}^{t} + \beta \times \left| X_{i}^{t} - X_{\text{rest}}^{t} \right| & f_{i} > f_{g} \\ X_{i}^{t} + K \times \left( \frac{X_{i}^{t} - X_{\text{worst}}^{t}}{f_{i} - f_{\text{worst}}} \right) & f_{i} \leq f_{g} \end{cases}$$  \hspace{1cm} (3)

Where: \(X_{\text{best}}\) represents the current global optimal position, \(\beta\) is the step control parameter, \(K \in (0,1)\) represents a uniformly distributed random number, \(f_{i}\) indicates the fitness of the current sparrow individual, \(f_{g}\) and \(f_{\text{worst}}\) represents the current best and worst fitness, \(\epsilon\) is a very small
number, prevent denominator from being 0.

2.2. Sparrow search algorithm based on opposition-based Learning, random walk and adaptive t-distribution strategy (OTRSSA)

2.2.1. Opposition-based Learning strategy initialization population
Opposition based learning (OBL) was proposed by Tizhuosh in 2005, which has been proved to be an effective method to improve the search ability of the algorithm [12].

Definition 1 Opposite Number: If \( x \) is any real number between \([a, b]\), then the opposite number of is:
\[
x' = a + b - x_0
\] (4)

Definition 2 Opposite Point: Let \( X = (x_1, x_2, \ldots, x_D) \) be a point in an \( D \)-dimensional space, then the opposite point corresponding to \( x_j \in [a_j, b_j] \) is:
\[
X' = (x'_1, x'_2, \ldots, x'_D) \quad x'_j = a_j + b_j - x_j
\] (5)

Because the feasible scheme and the scheme based on reverse learning are located on both sides of the search space, the reverse learning strategy can expand the search area and enhance the global search ability.

2.2.2. Adaptive t-distribution strategy to update sparrow position
T-distribution is also called student distribution. Its curve shape is related to the degree of freedom parameter \( n \). The probability density function is as follows:
\[
p_t(x) = \frac{\Gamma\left(\frac{n+1}{2}\right)}{\sqrt{n\pi}\Gamma\left(\frac{n}{2}\right)} \left(1 + \frac{x^2}{n}\right)^{-\frac{n+1}{2}}
\] (6)

Where: \( \Gamma\left(\frac{n+1}{2}\right) = \int_0^{+\infty} x^\frac{n+1}{2} e^{-x} dx \) is Euler integral of the second type.

After the sparrow is updated, the adaptive \( t \)-distribution is used to update the sparrow position again. Compared with the sparrow before and after the update, the sparrow before is replaced if it is better. The updating method of adaptive \( t \)-distribution is as follows:
\[
x'_i = x_i + x_i * t(\text{iter})
\] (7)

Where: \( x'_i \) is the position of sparrow after variation; \( x_i \) is the position of the \( i \)-th sparrow; \( t(\text{iter}) \) is a \( t \)-distribution with the number of iterations as the parameter.

2.2.3. Random walk strategy perturbs sparrow position
The mathematical expression of random walk strategy is as follows:
\[
X(t) = [0, \text{cussum} (2r(t_1) - 1), \ldots, \text{cussum} (2r(t_n) - 1)]
\] (8)

Where: \( X(t) \) is defined as the set of steps of random walk; \( \text{cussum} \) represents the cumulative sum; \( t \) is the number of steps of random walk (in this paper, we take the maximum number of iterations); \( r(t) \) is a random function, defined as:
\[
r(t) = \begin{cases} 
1, \text{rand} > 0.5 \\
0, \text{rand} \leq 0.5 
\end{cases}
\] (9)
Since there is a boundary in the feasible region, it is not possible to update the position directly with equation (8), so it is necessary to normalize it:

\[ X'_i = \left( X'_i - a_i \right) \frac{\left( d_i - c_i \right)}{\left( b_i - a_i \right)} + c_i t \tag{10} \]

Where: \( a_i \) is the minimum value of random walk of the \( i \)-th dimension variable; \( b_i \) is the maximum value of random walk of the \( i \)-th dimension variable; \( c_i \) is the minimum value of the \( t \)-th iteration of the \( i \)-th dimension variable; \( d_i \) is the maximum value of the \( i \)-th dimension variable in the \( t \)-th iteration.

To sum up, the flow chart of OTRSSA algorithm is shown in figure 1.

3. Performance test

In order to verify the advanced nature of OTRSSA algorithm, OTRSSA, SSA, WOA, PSO, BA and GWO are used to solve the test function respectively. The test function is shown in table 1~3. The above test process is conducted in the Intel Core i7 CPU, 2.50GHZ, 16GB memory and windows7 64 bit test environment. The population size of each algorithm is 30 and the iteration times are 500. The 12 test functions are independently simulated 30 times, and the average values and standard deviation obtained from 30 experiments are counted, as shown in Table 4.

Table 1. Unimodal test functions (Dim = 30).

| Test function | Range       | Optimal value |
|---------------|-------------|---------------|
| \( F_1(x) = \sum_{i=1}^{n} x_i^2 \) | [-100,100] | 0             |
| \( F_2(x) = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i| \) | [-10,10]  | 0             |
| \( F_3(x) = \sum_{i=1}^{n} \left( \sum_{j=1}^{n} |x_i| \right)^2 \) | [-100,100] | 0             |
| \( F_4(x) = \max \left\{ |x_i|, -1 \leq i \leq n \right\} \) | [-100,100] | 0             |
| \( F_5(x) = \sum_{i=1}^{n} ix_i^2 + \text{random}[0,1) \) | [-1.28,1.28] | 0             |
Table 2. Multimodal test functions (Dim = 30)

| Test function | Range | Optimal value |
|---------------|-------|---------------|
| $F_6(x) = \frac{\pi}{n} \left\{ (10 \sin(\pi y_1) + \sum_{i=0}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\}$ | [-50, 50] | 0 |
| $F_7(x) = \sum_{i=1}^{n} \left[ x_i^2 - 10 \cos(2\pi x_i) + 10 \right]$ | [-5.12, 5.12] | 0 |
| $F_8(x) = -20 \exp \left(-0.2 \sqrt{\sum_{i=1}^{n} x_i^2/n} \right) - \exp \left(\sum_{i=1}^{n} \cos(2\pi x_i/n) \right) + 20 + e$ | [-32, 32] | 0 |
| $F_9(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos \left(\frac{x_i}{\sqrt{n}} \right)$ | [-600, 600] | 0 |

Table 3. Fixed-dimension test functions.

| Test function | Dimension | Range | Optimal value |
|---------------|-----------|-------|---------------|
| $F_{10}(x) = 4x_1^2 - 2.1x_1^4 + \frac{x_1^6}{3} + x_1x_2 - 4x_1^2 + 4x_1^4$ | 2 | [-5, 5] | -1.0316 |
| $F_{11}(x) = -\sum_{i=1}^{n} a_i \left( x_i - p_i \right)^T$ | 3 | [0, 1] | -3.86 |
| $F_{12}(x) = -\sum_{i=1}^{10} \left[ (X - a_i)(X - a_i)^T + c^T \right]^{-1}$ | 4 | [0, 10]^10 | -10.5363 |

Table 4. Comparisons among test functions

| Test function | Statistics | OTRSSA | SSA | WOA | PSO | BA | GWO |
|---------------|------------|--------|-----|-----|-----|----|-----|
| F1            | Ave        | 2.79e-263 | 1.186e-61 | 1.450e-73 | 0.1318 | 6.8514 | 1.465e-27 |
|               | Std        | 6.486e-61 | 6.361e-73 | 0.0430 | 0.7689 | 2.178e-27 |
| F2            | Ave        | 6.004e-91 | 4.458e-30 | 8.626e-50 | 0.9664 | 272.406 | 1.119e-16 |
|               | Std        | 0        | 1.847e-25 | 16444.59 | 15.696 | 77.730 | 6.7474 |
| F3            | Ave        | 3.614e-242 | 3.668e-26 | 43173.419 | 38.881 | 156.134 | 2.255e-05 |
|               | Std        | 0        | 1.847e-25 | 16444.597 | 15.6967 | 77.730 | 6.7475 |
| F4            | Ave        | 1.888e-85 | 2.430e-30 | 48.479 | 8.6556 | 20.985 | 7.478e-07 |
|               | Std        | 1.034e-84 | 1.207e-29 | 29.660 | 4.172 | 8.679 | 8.786e-07 |
| F5            | Ave        | 0.000195 | 0.00132 | 0.00598 | 0.2418 | 54.2296 | 0.001972 |
|               | Std        | 0.000148 | 0.001667 | 0.007189 | 0.1156 | 13.3698 | 0.001099 |
| F6            | Ave        | 2.664e-08 | 5.179e-12 | 0.02564 | 9.6279 | 28.5366 | 0.04812 |
|               | Std        | 4.924e-08 | 1.397e-11 | 0.01630 | 4.9898 | 10.5094 | 0.02233 |
| F7            | Ave        | 8.8818e-16 | 8.8818e-16 | 4.0856e-15 | 2.105 | 14.0752 | 1.032e-13 |
|               | Std        | 0 | 0 | 2.355e-15 | 1.0159 | 6.9515 | 1.9378e-14 |
| F8            | Ave        | -1.0316 | -1.0316 | -1.0316 | -1.0316 | -1.0299 | -1.0316 |
|               | Std        | 3.932e-10 | 5.133e-16 | 3.819e-9 | 4.430e-15 | 0.0017 | 1.7537e-08 |
| F9            | Ave        | -3.8628 | -3.811 | -3.8564 | -3.8628 | -3.8179 | -3.8616 |
|               | Std        | 1.25e-06 | 0.1961 | 0.00888 | 1.255e-13 | 0.0591 | 0.0023 |
| F10           | Ave        | -10.403 | -8.986 | -7.101 | -5.748 | -4.817 | -10.401 |
|               | Std        | 2.9769 | 2.3907 | 3.2032 | 3.433 | 2.886 | 0.00113 |

As shown in Table 4, for high-dimensional unimodal test functions F1 ~ F5, OTRSSA greatly improves the optimization accuracy and stability compared with the other five algorithms, for F1 ~ F4, the average and standard deviation of OTRSSA are improved by more than 25 orders of magnitude.
compared with the other five algorithms, for F5, although the optimization performance of OTRSSA is not as good as F1 ~ F4, it is still improved by an order of magnitude. For high-dimensional multimodal test function F6 ~ F9, in F7 ~ F9, OTRSSA and SSA are far better than the other four algorithms in the optimization accuracy and stability, which shows that OTRSSA can jump out of the local optimum, find the global optimum stably, and has strong robustness. Although the performance of OTRSSA is not as good as SSA in F6, the optimization accuracy and stability are still very high. For low dimensional functions F10 ~ F12, although the average value and standard deviation of multiple optimization by OTRSSA are not significantly improved compared with the other five algorithms, for each test function F10 ~ F12, OTRSSA can find its approximate optimal solution, which shows that OTRSSA has strong robustness and good stability. In order to further test the advantage of OTRSSA in iteration speed, F6 and F10 are selected to test with OTRSSA and SSA respectively. The convergence curves are shown in figure 2 and figure 3.

![Figure 2. Curve of iterative convergence of F6](image)

![Figure 3. Curve of iterative convergence of F10](image)

It can be seen from table 4, figure 2 and figure 3 that although the average and standard deviation of multiple optimizations by OTRSSA in test functions F6 and F10 are not as accurate as those by SSA, its convergence speed is fast and the global approximate optimal solution can be obtained stably.

To sum up, compared with other algorithms, OTRSSA has a greater improvement in the optimization accuracy and stability, and its iterative convergence speed is faster, the search ability is strong, it can jump out of the local optimal solution, and find the global optimal solution or global approximate optimal solution stably.

### 4. Dynamic model of trajectory optimization problem

#### 4.1. Motion model of reentry process

Considering that the earth is a stationary homogeneous sphere and the vehicle's bank angle is zero. The reentry motion equation is:

\[
\begin{align*}
\dot{r} &= v \sin \gamma \\
\dot{v} &= -\frac{D}{m} - g \sin \gamma \\
\dot{\gamma} &= \frac{L}{mv} + \frac{1}{v} \left( \frac{v^2}{r} - g \right) \cos \gamma \\
\dot{\beta} &= \frac{v}{r} \cos \gamma
\end{align*}
\]

Where: \( r \) is the radial distance from the Earth center to the vehicle, \( v, \gamma, \beta, g, m \) are the velocity, the flight path angle, the range angle and the acceleration of gravity respectively. \( L \) and \( D \) are the aerodynamic lift and drag accelerations respectively, the specific expression is:

\[
\begin{align*}
L &= \rho V^2 C_L S_{ref} / 2 = \rho_0 e^{-k/r} V^2 C_L S_{ref} / 2 \\
D &= \rho V^2 C_D S_{ref} / 2 = \rho_0 e^{-k/r} V^2 C_D S_{ref} / 2
\end{align*}
\]
Where: $\rho$ is the atmospheric density; $\rho_0 e^{-h/h_s}$ is an exponential atmospheric model; $h_s = 7110$; $C_L$, $C_D$ are the lift and drag coefficients, which depend on the angle of attack angle and Mach number of the entry vehicle. $S_{ref}$ is the characteristic area of the aircraft.

### 4.2 Constraints

#### 4.2.1 Process constraints

Four process constraints, including dynamic pressure, overload, heating rate and angle of attack, are considered in reentry trajectory optimization.

$$[q(t) = \frac{\rho v^2}{2} \leq q_{\text{max}}, \quad n = \frac{\sqrt{L^2 + D^2}}{mg} \leq n_{\text{max}}, \quad q = C \rho^{0.5} v^3 \leq q_{\text{max}}, \quad \alpha_{\text{min}} \leq \alpha \leq \alpha_{\text{max}}]$$ (13)

Where: $q_{\text{max}}$ is the maximum dynamic pressure, $n_{\text{max}}$ is the maximum overload value, $q_{\text{max}}$ is the maximum heating rate, $\alpha_{\text{min}}$, $\alpha_{\text{max}}$ are the maximum and minimum angle of attack.

#### 4.2.2 Terminal constraints

The terminal constraints of reentry process mainly consider the terminal height and velocity constraints.

$$[r(t_f) = r_f, v(t_f) \geq v_f]$$ (14)

Where: $r_f$ is the terminal height, $v_f$ is the terminal velocity.

#### 4.2.3 Performance index

In this paper, the farthest voyage is selected as the cost function, and the objective function is composed of cost function and constraint function:

$$J = \max \beta(t_i) + j_1 + j_2 + j_3 + j_4 + j_5$$ (15)

Where: the expression of $j_1$, $j_2$, $j_3$, $j_4$, $j_5$ is:

$$\begin{align*}
  j_1(p) &= \begin{cases}
    0, & v(t_i) > v_f, \\
    \left(\frac{v_f - v(t_i)}{100}\right)^2, & v(t_i) \leq v_f
  \end{cases}, \\
  j_2(p) &= \begin{cases}
    0, & |r(t_i) - r_f| < 100, \\
    \left(\frac{r_f - r(t_i)}{100}\right)^2, & |r(t_i) - r_f| \geq 100
  \end{cases}, \\
  j_3(p) &= \begin{cases}
    0, & q_{\text{max}} < \bar{q}_{\text{max}}, \\
    \left(\frac{\bar{q}_{\text{max}} - \bar{q}_{\text{max}}}{1000}\right)^2, & \bar{q}_{\text{max}} \geq \bar{q}_{\text{max}}
  \end{cases}, \\
  j_4(p) &= \begin{cases}
    0, & n_{\text{max}} < \bar{n}_{\text{max}}, \\
    \left(\frac{n_{\text{max}} - \bar{n}_{\text{max}}}{1000}\right)^2, & n_{\text{max}} \geq \bar{n}_{\text{max}}
  \end{cases}, \\
  j_5(p) &= \begin{cases}
    0, & q_{\text{max}} < \bar{q}_{\text{max}}, \\
    \left(\frac{q_{\text{max}} - \bar{q}_{\text{max}}}{100000}\right)^2, & q_{\text{max}} \geq \bar{q}_{\text{max}}
  \end{cases}
\end{align*}$$ (16)

The reentry trajectory optimization problem is described as: the total flight time $t_f$ is divided into $n-1$ segments, and the corresponding control variable (angle of attack) at each time node is $[\alpha_1, \alpha_2, \ldots, \alpha_n]$, then there are $n+1$ variables in total which are $[t_f, \alpha_1, \alpha_2, \ldots, \alpha_n]$, the angle of attack at
other times is obtained by linear interpolation, and the flight trajectory satisfying the constraints and the optimal performance index is obtained by integrating the motion equation and optimizing the variables, which belongs to the multi constraint optimization problem.

5. Simulation calculation and verification
The trajectory optimization problem of reentry vehicle is a multi constrained optimization problem. Using OTRSSA to solve the reentry trajectory optimization problem is to use OTRSSA to optimize the objective function. The position of sparrow individual represents a group of optimization variables. Individual fitness is the value of objective function. The objective function is composed of cost function and constraint function. The discoverer in sparrow population is the individual with better value of current objective function, and the follower is the individual with worse value of current objective function, after several iterations, the optimal solution returned is the optimal value of the variable corresponding to the farthest voyage.

5.1 Simulation parameters
The aerodynamic reference area of the aircraft is $0.35m^2$, the quality is $907kg$, the initial condition of reentry point is $(v_0, r_0, \gamma_0, \beta_0) = (7000m/s, 100km, -0.2^\circ, 0 rad)$, the range of attack angle is $(0, 20^\circ)$, the heating rate, dynamic pressure and overload constraints are $2200/kWm^2, 80kPa$ and $3g$, the terminal height is $20km$, terminal speed is greater than $1000m/s$.

5.2. Simulation results and analysis
Figure. 4 and figure. 5 are the range angle curves obtained by using the six algorithms of OTRSSA, SSA, WOA, PSO, Ba and GWO, respectively. It can be seen from figure. 4 that the range angle obtained by using the optimization of OTRSSA is the largest, which is, converted into the range of about; as can be seen from figure 5, OTRSSA converges rapidly to the optimal value at the beginning of iteration, and then stabilizes at the optimal value, which indicates that OTRSSA has strong global optimization ability.

The altitude and speed change are shown in figure 6 and figure 7. From figure 6, it can be seen that the aircraft shall be wave like jumping after re-entry and pull, with a smaller jump range, and the re-entry glide time is about 3000s and the terminal height is $20km$; it can be seen from figure 7 that the overall velocity of the aircraft decreases in the course of re-entry and glide, and the terminal speed is slightly higher than $1000m/s$, which meets the design requirements.
Figure 8-11 shows the change of attack angle, heating rate, dynamic pressure and overload during reentry of aircraft. It can be seen from the figure that the reentry process of the aircraft meets the design constraint value and the optimized trajectory is effective and feasible.
6. Summary

1) In this paper, the basic sparrow search algorithm is improved from three aspects: opposition-based learning, adaptive t-distribution and random walk, and the improved sparrow search algorithm (OTRSSA) is compared with other five algorithms. The results of 12 test functions show that compared with other algorithms, OTRSSA has higher optimization accuracy and stability, faster iterative convergence speed, and it can effectively jump out of the local optimal solution and quickly find the global optimal solution;

2) In this paper, OTRSSA is used to solve the trajectory optimization problem. The simulation results show that OTRSSA can optimize a reentry trajectory with the farthest range and meet the constraints, which has a good application value in the reentry trajectory optimization problem.

3) There is still room to improve the stability of multi-dimensional function optimization for OTRSSA, which can be further applied to the cooperative trajectory optimization design of reentry vehicle.

References

[1] Liu Siyuan, Liang Zixuan, Ren Zhang, et al. Overview of guidance methods for hypersonic gliding vehicle reentry stage[J]. Chinese Space Science and Technology, 2016, 36(6).

[2] ZHANG Peng-cheng, LIXin-guo. Based on Trajectory Optimization of Glide Phase with GPM for Hypersonic Glide Vehicle[J]. Electronic Design Engineering, 2013, 21(15): 105-108.

[3] TANG Liang, YANG Jian-min, CHEN Feng-yu. Trajectory Optimization with Multi-Constraints for a Lifting Re-entry Vehicle[J]. Missiles and Space Vehicles, 2013, 1 (1):1-5.

[4] Jiang Z , Rui Z . Reentry trajectory optimization for hypersonic vehicle satisfying complex constraints[J]. Chinese Journal of Aeronautics, 2013, 26(006):1544-1553.

[5] Y F MAO,D L ZHANG. Reentry trajectory optimization for hypersonic vehicle based on improved Gauss pseudospectral method[J]. Soft Computing,2017,21(6):4583-4592.

[6] A CRAIN,S ULRICH. Nonlinear optimal trajectory planning for free-floating space manipulators using a Gauss pseudospectral method[C]/AIAA/AAS Astrodynamics Specialist Conference,Long Beach,California,2016.

[7] Wang Yin, Wang Bin. Spacecraft reentry trajectory optimization based on improved ant colony algorithm[J]. Journal of Taiyuan University of Science and Technology, 2020, 041(004):270-274.

[8] Wu Y , Yan B , Qu X . Improved Chicken Swarm Optimization Method for Reentry Trajectory Optimization[J]. Mathematical Problems in Engineering,2018,(2018-1-31), 2018, 2018(Pt.2):1-13.

[9] Artificial Bee Colony-Based Direct Collocation for Reentry Trajectory Optimization of Hypersonic Vehicle[J]. IEEE Transactions on Aerospace & Electronic Systems, 2015.

[10] XUE J, SHEN B. A novel swarm intelligence optimization approach: sparrow search algorithm[J]. Systems Science & Control Engineering, 2020, 8(1):22-34.

[11] HAN Tong, TANG Andi. Multiple UAV cooperative path planning based on LASSA method[J]. Systems Engineering and Electronics,2021.

[12] TIZHOOSH H R. Opposition-based learning: A new scheme for ma- chine Intelligence [C] / /Computational Intelligence for Modelling. Vienna, Austria: Computer Society, 2005: 695 －701.