Coordinated Path Planning for UAVs Based on Sheep Optimization

YANG Liuqing[1,2], *WANG Pengfei*[3*], *ZHANG Yong*[1,2]

1. Research Institute of Pilotless Aircraft, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P. R. China;
2. Key Laboratory of Advanced Technology for Small and Medium-Sized UAV, Ministry of Industry and Information Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P.R. China;
3. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, P. R. China

(Received 13 January 2020; revised 20 July 2020; accepted 20 September 2020)

Abstract: Using the traditional swarm intelligence algorithm to solve the cooperative path planning problem for multi-UAVs is easy to incur the problems of local optimization and a slow convergence rate. A cooperative path planning method for multi-UAVs based on the improved sheep optimization is proposed to tackle these. Firstly, based on the three-dimensional planning space, a multi-UAV cooperative cost function model is established according to the path planning requirements, and an initial track set is constructed by combining multiple-population ideas. Then an improved sheep optimization is proposed and used to solve the path planning problem and obtain multiple cooperative paths. The simulation results show that the sheep optimization can meet the requirements of path planning and realize the cooperative path planning of multi-UAVs. Compared with grey wolf optimizer (GWO), improved gray wolf optimizer (IGWO), chaotic gray wolf optimizer(CGWO), differential evolution (DE) algorithm, and particle swam optimization (PSO), the convergence speed and search accuracy of the improved sheep optimization are significantly improved.

Key words: multi-UAV cooperation; path planning; swarm intelligence algorithm; multi-population; improved sheep optimization (ISO)

CLC number: V279 **Document code:** A **Article ID:** 1005-1120(2020)05-0816-15

0 Introduction

With the development and maturity of the UAV technology and the continuous improvement of intelligence, UAVs will become the leader of the future sky and the main equipment of the armed forces of countries around the world.[1] In the modern warfare of informationization, networking and systematization confront high-speed development, relying on a single UAV to perform intelligence reconnaissance, battlefield strike and other tasks is far from current mission requirements[2]. Using multi-UAVs to perform combat tasks against multiple targets has become an inevitable trend[3].

Cooperative path planning is the key for multiple UAVs to achieve cooperative operations. It has the characteristics of high dimensionality, multi-constraints and spatio-temporal coordination, which is quite challenging. To solve this problem, researchers have proposed a variety of methods, including path planning algorithms, obstacle avoidance techniques and path adjustment strategies. Current methods for path planning can roughly be divided into five categories. First, on the basis of graph theory, there are the voronoi diagram method[4] and the signpost diagram method[5]. They hold a fast construction speed and a high route safety, but are difficult to be applied to multi-UAVs collaboration in three-dimensional space. Second, the method based on potential fields, like artificial potential field method (APF)[6], is simple in principle and fast in calculation, which is suitable for path re-planning with high

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

How to cite this article: YANG Liuqing, WANG Pengfei, ZHANG Yong. Coordinated path planning for UAVs based on sheep optimization[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2020, 37(5): 816-830.
http://dx.doi.org/10.16356/j.1005-1120.2020.05.016

real-time requirement. However, it is easy to fall into a state of stagnation under certain circumstances, leading to the failure of planning. Third, based on random sampling, the fast random tree method (RRT)[7] is typical. In the complex environment with known or dynamic unknown, a single track can be quickly searched out, but the cost of this method is relatively high, and the planned path is not always the optimal one. Fourth, search algorithms based on heuristic information, such as the A* algorithm[8] and the sparse A* algorithm[9], are simple and efficient, but they are easy to fall into the infinite cycle and their planned paths have many folding points. Moreover, when the environment is complex and the problem solving scale is large, the efficiency is relatively low and the parallelization ability is poor. Fifth, swarm intelligence optimization algorithms can be used for collaborative route planning of multi-UAVs due to its simple principle, fast planning speed, freedom from spatial dimension and potential parallelism. Ref.[10] designed a spatial optimization voting mechanism to solve the problem that particle swarm optimization (PSO) is prone to local optimization. Meanwhile, time coordination and space obstacle avoidance technologies were proposed for the spatial-temporal coordination of multi-UAVs, and the four-dimensional collaborative path planning of multi-UAVs was realized. Ref.[11] aimed at the problem of path planning for UAVs in three-dimensional space, and proposed the gray wolf optimization algorithm to solve the problem. A flight path that could avoid obstacles was obtained. Swarm intelligence algorithm has obvious advantages in dealing with challenging problems with high dimensions and multiple constraints.

Compared with traditional optimization algorithms, the swarm intelligence algorithm has become a widely used optimization method in practical engineering due to its good performance. However, no free lunch (NFL) theorem[12] logically proves that there is no suitable swarm intelligence algorithm for solving all optimization problems. A particular swarm intelligence algorithm may perform well on one kind, while poorly on another. In terms of path planning, most algorithms have a slow convergence speed and insufficient computational accuracy. NFL makes the study of swarm intelligence highly active, which imples a large number of scholars to improve current algorithms and propose new swarm intelligence algorithms. Among them, sheep optimization (SO) is a novel one to simulate the foraging behavior of sheep proposed by QU and XU et al. in 2018[13]. In this algorithm, three strategies of global search, local development and jumping out of local optimality are designed by simulating the three behaviors of bellwether's lead, sheep interaction and shepherd supervision. In Ref.[13], the performance of this algorithm is verified. Compared with particle swarm optimization algorithms, this algorithm can obtain higher quality solutions with a faster convergence rate and better stability. However, the algorithm has not been applied to any practical engineering scenario.

When the algorithm solves the problem, the core operator needs to calculate the fitness function value of the population for many times. The movement mode of the population is too simple; the supervising mechanism of the shepherd is too complex; and the parameter selection in the actual project is difficult. So the algorithm is too complicated and not easy to be implemented in the project. Therefore, based on simpleness in implementation and less parameters of the swarm intelligence algorithms, an improved sheep algorithm was proposed and applied to multi-UAVs cooperative path planning in order to solve the defects of slow convergence speed and low computational accuracy existing in current swarm intelligence algorithms.

In this paper, a multi-UAVs cooperative path planning method based on an improved sheep optimization (ISO) is proposed for path planning when multi-UAVs are used to carry out coordinated attack on known targets. First, mathematical modeling is carried out for the cooperative path planning of multi-UAVs. Second, the ISO is designed to solve the path planning problem, and a three-dimensional cooperative path satisfying the requirements of the planning is obtained. Finally, the benchmark function test and the simulation experiment are carried out to verify the effectiveness of this method by com-

paring it with differential evolution (DE) algorithm, PSO, gray wolf optimizer (GWO), improved gray wolf optimizer (IGWO), and chaotic gray wolf optimizer (CGWO) algorithm.

1 Modeling UAV Cooperative Path Planning

1.1 Planning space representation

In path planning, an appropriate planning space must be established in accordance with the flight environment and mission requirements. In the present work, a mountain background is taken as a task environment, and a digital elevation model is established using a random function to simulate peaks and other threat obstacles. The mountain model function was proposed in Ref.[14]. This model consists of the original digital and threat equivalent terrain models. The former is expressed as

$$z_1(x,y)=\sin(y+a)+b\cdot\sin(x)+c\cdot\cos(d\cdot\sqrt{x^2+y^2})+e\cdot\cos(y)+f\cdot\sin(f\cdot\sqrt{x^2+y^2})+g\cdot\cos(y) \tag{1}$$

where x and y refer to the point coordinates on a horizontal projection plane; z_1 refers to the height coordinate that corresponds to the coordinate points on a horizontal plane; a, b, c, d, e, f and g are the coefficients. The topography of a different landform can be obtained by changing the parameters.

The threat equivalent terrain model is

$$z_2(x,y)=\sum_{i=1}^{k}h(i)\cdot\exp\left(-\left(\frac{x-x_{0i}}{x_{si}}\right)^2-\left(\frac{y-y_{0i}}{y_{si}}\right)^2\right) \tag{2}$$

where x and y refer to the point coordinates on a horizontal projection plane; z_2 refers to the height of the peak; $h(i)$ the height of the highest point of peak i on a base terrain; x_{0i} and y_{0i} refer to the coordinates of the highest point of peak i; x_{si} and y_{si} the variables related to the slope of peak i along x, y axises. If x_{si} and y_{si} are large, the slope of the peak is flat and abrupt.

The final mountain threat model is obtained by integrating the original digital terrain model into the threat equivalent terrain model

$$z(x,y)=\max(z_1(x,y),z_2(x,y)) \tag{3}$$

The topography of different landforms can be obtained by changing parameters in the function. In the planning space, the flying path of UAVs can be represented by many waypoints. Consequently, the waypoints are connected to form multiple flight paths, which are linked with the starting and target points to form a flying path. We set the starting point of a certain UAV as $S(x_0, y_0, z_0)$ and the target point as $E(x_e, y_e, z_e)$. The number of waypoints is n, and the waypoints searched can be represented by $\{S, P_1, P_2, \cdots, P_n, E\}$, where the coordinate of a track node is $P_p=(x_p, y_p, z_p)$.

1.2 Path cost function

The purpose of multi-UAV cooperative path planning is that, on the premise of satisfying the requirements of a safe flight and space-time cooperation, every UAV can search the corresponding path, and the synthetic path cost of the UAV fleet must be the least. Therefore, path planning requires the establishment of a path cost function as an index to evaluate the quality of a path. The satisfaction of the spatial and temporal cooperative constraints of multi-UAVs by considering the dynamics, and threat constraints of a single UAV in multi-UAV cooperative path planning is required. Thus, given the planning objective, the following cost indexes are considered in the present work: The performance indexes of a single UAV include fuel consumption, maximum climb/slide angle, flying altitude, peak threat, and multi-UAV time cooperation. Spatial cooperation is manifested in multi-UAV path collision avoidance. We set different flight altitudes that must be avoided by each UAV. The synthetic cost function is established as

$$J=w_1k_1J_{\text{fuel}}+w_2k_2J_{\text{angle}}+w_3k_3J_{\text{height}}+w_4k_4J_{\text{threat}}+w_5k_5J_{\text{coop}} \tag{4}$$

where w_1, w_2, w_3, w_4 and w_5 refer to the weights of different cost indexes, and the sum of weights is 1. The paths that satisfy different requirements can be obtained by adjusting the weights. To ensure that all cost indexes are involved in path planning, the functions are normalized in accordance with the range of their values, and then weighted summation is performed.

Fuel consumption cost is related to the length

of flight path and flying speed. Assuming that UAVs consistently fly at a certain speed, fuel costs can be replaced by the length of the path

$$J_{\text{fuel}}=\sum_{i=1}^{n-1}\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2+(z_{i+1}-z_i)^2} \tag{5}$$

where $(x_{i+1}, y_{i+1}, z_{i+1})$ and (x_i, y_i, z_i) correspond to the coordinates of the adjacent path points.

J_{angle} refers to the cost of the maximum climb/slide angle and is expressed as

$$\begin{aligned} J_{\text{angle}}&=\sum_{i=1}^{n}\theta_i \\ \theta_i&=\arctan\left(\frac{|z_{i+1}-z_i|}{\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2}}\right) \end{aligned} \tag{6}$$

where θ_i refers to the climb/slide angle of the adjacent points of a certain path.

To satisfy the requirements of flight safety and concealment, the flight altitude cannot be overly low or high. Height cost can be expressed as

$$J_{\text{height}}=\sum_{i=1}^{n}|h_i-\text{safth}_i| \tag{7}$$

where h_i refers to the height of path point i on a certain path, and safth_i the minimum safety height for each UAV.

Collision with the mountain in the flying course of the UAV must be avoided. In Ref.[15], the peak model is represented by a cone approximate representation. The path is divided into m equal sections, and $m-1$ sampling points are obtained in the center. The threat cost of the whole path is expressed as

$$J_{\text{threat}}=\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{K}\text{threat}(j,k) \tag{8}$$

where n refers to the number of path points, K the number of peaks, and $\text{threat}(j,k)$ the threat cost of the sampling point (x_i, y_i, z_i) in the current section and a certain peak and is expressed as

$$\text{threat}(j,k)=\begin{cases} 0 & h_j>H(k)\ \text{or}\ d_T>R_T+d_{T\min} \\ R_T(h)+d_{T\min}-d_T & h_j<H(k)\ \text{and}\ d_T<R_T+d_{T\min} \end{cases} \tag{9}$$

$$R_T(h)=(H(k)-h)/\tan\theta \tag{10}$$

where n refers to the number of path points; K the number of peaks; $H(k)$ the height of peak k; R_T the maximum extension radius; h_j the flying altitude of the current UAV; d_T the distance from the UAV to the symmetrical axis of the peak; $d_{T\min}$ the minimum distance allowed on the terrain; and θ the slope of the terrain. The terrain threat is depicted in Fig.1.

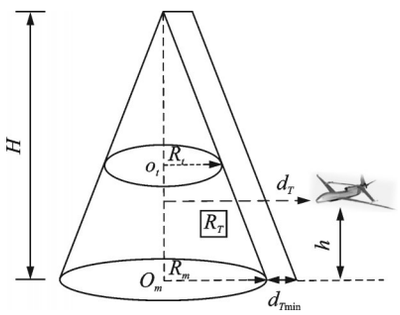


Fig.1 Terrain threat map

Cooperative cost function implies time cooperation. All UAVs are required to reach the target point simultaneously as far as possible. If the course of a certain path cannot satisfy the time cooperative constraints, the path must be corrected. Assuming that the flying speed of the UAV is in the range of $[v_{\min}, v_{\max}]$ and the course of the UAV i is L_i, its flight time period is $T_i\in[T^i_{\min}, T^i_{\max}]$. Similarly, assuming that the flying time of UAV j is in the range of $T_j\in[T^j_{\min}, T^j_{\max}]$, if the flight time of the two UAVs intersects, temporal cooperation is feasible, that is

$$T_{\text{inter}}=T_i\bigcap T_j\neq\varnothing \tag{11}$$

which is in accordance with the temporal cooperation evaluation formula between paths in Ref.[16]. Then, the temporal cooperation cost function is obtained on the basis of the planning model in the present work.

$$J_{\text{cop}T}=\begin{cases} 1 & T_{\text{inter}}=\varnothing \\ 0 & 0<T_{\text{inter}}<\text{rand}\cdot T_{\min} \\ \dfrac{3T_{\text{inter}}}{T_{\min}} & T_{\text{inter}}>\dfrac{1}{3}\cdot T_{\min} \end{cases} \tag{12}$$

where $T_{\min}$ refers to a time period with a small range of a certain path in the flight time period, and T_{inter} the intersection of the flight time for two paths.

2 Cooperative Path Planning Model Based on ISO Algorithm

2.1 Introduction of the sheep optimization algorithm

SO realizes fast global exploration by simulat-

ing the behaviour that the bellwether leads the sheep, which makes the sheep approach the known global optimal solution quickly. The mutual movement of sheep can achieve local development, and further speed up the convergence. The shepherd supervision mechanism is used to judge whether it is falling into the local optimization and quickly jump out of the local optimal solution.

2.1.1 Bellwether's lead

The bellwether refers to the sheep with the optimal fitness function value in the flock, and the bellwether's lead refers to the behavior of each sheep moving towards the bellwether. The corresponding global exploration mechanism of the algorithm is to ensure the performance of the search. The position of the new sheep is updated only when the fitness function value of the new sheep is better than the old sheep. Fig.2 is the flow chart of the algorithm for the bellwether's lead. The position of the corresponding sheep is updated when the sheep move to the bellwether

$$x_i^{\text{new}} = x_i^{\text{old}} + \text{rand}(0,1) \times (x_{\text{bellwether}} - x_i^{\text{old}}) \quad (13)$$

where x_i^{new} represents the updated position of sheep i, x_i^{old} the position which has not been updated of sheep i, and $x_{\text{bellwether}}$ the bellwether.

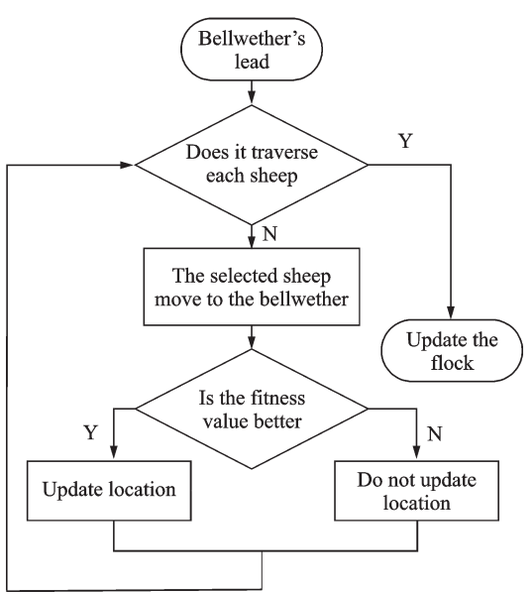


Fig.2 Flow chart of bellwether's lead algorithm

2.1.2 Sheep interaction

The sheep interaction behavior corresponds to the local development mechanism of the algorithm. Each sheep x_i in the flock will randomly select another sheep x_j for the sheep interaction strategy. If the fitness value of the selected sheep x_i is better than that of the random sheep x_j, x_i is updated to the position away from x_j, while x_j approaching to the position x_i, and vice versa. Similarly, to ensure the performance of the search, the position of the new sheep is updated only when the fitness function value of the new sheep is better than that of the old sheep. Fig.3 is the flow chart of the sheep interaction algorithm.

$$x_i^{\text{new}} = x_i^{\text{old}} + \text{rand}(0,1) \times (x_i^{\text{old}} - x_j^{\text{old}}) \quad (14)$$

$$x_j^{\text{new}} = x_j^{\text{old}} + \text{rand}(0,1) \times (x_i^{\text{old}} - x_j^{\text{old}}) \quad (15)$$

where Eq.(14) means x_i is updated to the position away from x_j, and Eq.(15) means x_j is updated to the position near x_i.

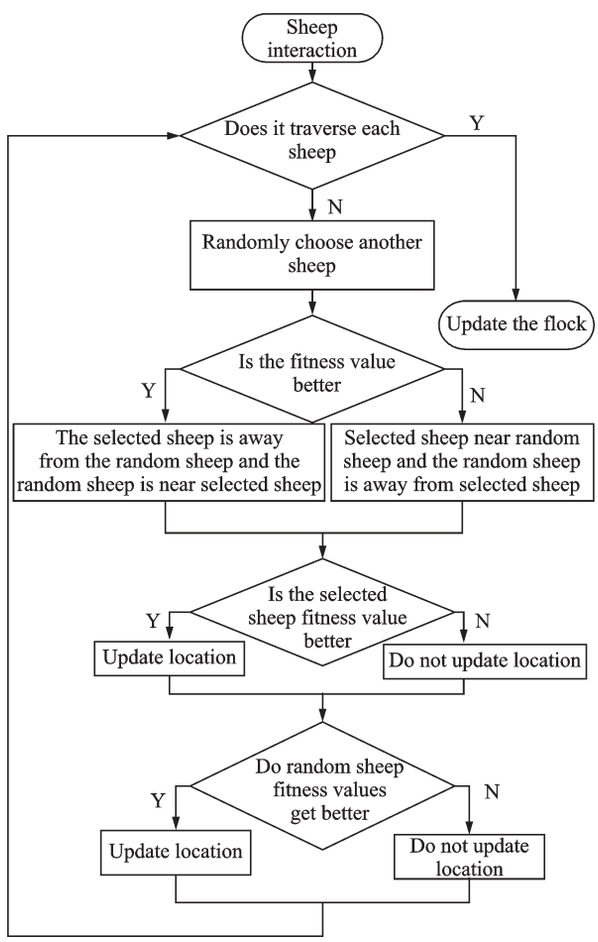


Fig.3 Flow chart of the sheep interaction algorithm

2.1.3 Shepherd supervision

When the fitness function difference between the current generation and the previous generation

is less than a threshold ε, the shepherd supervision mechanism is introduced to jump out of the local optimization. Each sheep will be herded by the shepherd with a certain probability p, that is, the sheep will be re-initialized with a probability p. Fig.4 is the flow chart of the shepherd supervision algorithm.

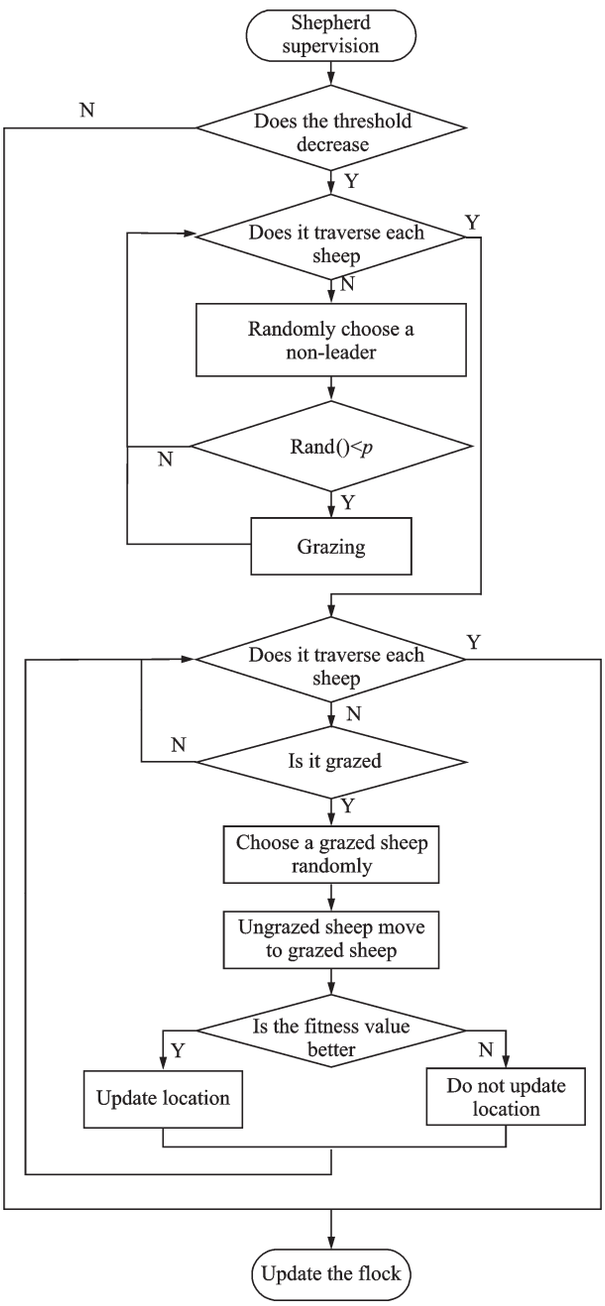


Fig.4 Flow chart of the shepherd supervision algorithm

The steps of the cooperative path planning for multi-UAVs based on sheep optimization are shown as follows.

Steps of the SO algorithm

(1) algorithm initialization

(2) while algorithm termination conditions are not met do

(3) bellwether's lead & sheep interaction & shepherd supervision

(4) end while

(5) output result

2.2 Improved sheep algorithm

The original sheep algorithm needs to calculate the value of the fitness function of the population for many times. The movement mode of the population is too simple. The shepherd supervision is too complex. The parameter selection in the actual project is difficult, and it is not easy to realize the project. Aiming at these problems, this paper proposes an improved sheep algorithm.

In the sheep algorithm, the fitness function value of the population needs to be calculated for many times. In engineering practice, the complexity of the fitness function may increase the computation time, and the operation may make the algorithm converge too quickly and fall into a local optimization. Therefore, the proposed improved sheep algorithm removes the operation of updating the position when the fitness function is better, thereby reducing the algorithm complexity.

In order to solve the problem of simple population movement mode, this paper improves the population position update mode of the bellwether's lead and sheep interaction. The mathematical model of the position update of the bellwether's lead is shown as

$$\boldsymbol{D}=\left|\boldsymbol{C}\cdot\boldsymbol{P}(t)-\boldsymbol{X}(t)\right| \tag{16}$$

$$\boldsymbol{X}(t+1)=\boldsymbol{P}(t)-\boldsymbol{A}\cdot\boldsymbol{D} \tag{17}$$

where t indicates the current iteration; $\boldsymbol{A}$ and $\boldsymbol{C}$ are coefficient vectors. $\boldsymbol{P}$ is the position vector of the bellwether and $\boldsymbol{X}$ the position vector of a sheep.

The vectors $\boldsymbol{A}$ and $\boldsymbol{C}$ are calculated as follows

$$\boldsymbol{A}=2\boldsymbol{a}\cdot\boldsymbol{r}_1-\boldsymbol{a} \tag{18}$$

$$\boldsymbol{C}=2\cdot\boldsymbol{r}_2 \tag{19}$$

where components of $\boldsymbol{a}$ are linearly decreased from 2 to 0 over the course of iterations and $\boldsymbol{r}_1$, $\boldsymbol{r}_2$ are random vectors in [0, 1].

In the sheep interaction mechanism, when the fitness value of the randomly selected sheep is better, the updated model of the current sheep moving to the random sheep is as follows

$$\boldsymbol{X}(t+1)=\boldsymbol{D}'\cdot e^{bl}\cdot\cos(2\pi l)+\boldsymbol{R}(t) \quad (20)$$

where $\boldsymbol{D}'=|\boldsymbol{R}(t)-\boldsymbol{X}(t)|$ is the distance between the i sheep and the random sheep; l a random value in $[-1, 1]$; b a constant for defining the logarithmic spiral shapes; $\boldsymbol{R}$ the position vector of the random sheep. Conversely, the mathematical model of the random sheep moving towards the current sheep is shown as

$$\boldsymbol{R}(t+1)=\boldsymbol{D}'\cdot e^{bl}\cdot\cos(2\pi l)+\boldsymbol{X}(t) \quad (21)$$

Due to the complexity of shepherd supervision mechanism, different thresholds and probabilities have a great impact on the performance of the algorithm, so it is difficult to select appropriate parameters in practical projects. In this paper, the shepherd supervision is simplified and replaced by the lévy flight strategy. The mathematical model is shown as

$$\boldsymbol{X}(t+1)=\boldsymbol{X}(t)+\boldsymbol{X}(t)\oplus\text{Levy} \quad (22)$$

where $\oplus$ represents term-by-term multiplication. Eq.(22) is essentially a random walk equation which can prevent the sheep algorithm from falling into the local optimal solution and ensure the algorithm to be developed effectively in the search space, and the distribution equation is shown as

$$\text{Levy}\sim u=t^{-\lambda} \quad 1<\lambda\leqslant 3 \quad (23)$$

The lévy flight is a special random walk in which the step lengths have a probability distribution that is heavy-tailed. Mantegna's algorithm is used to mimic a λ-stable distribution by generating random step length s that have the same behavior with the lévy flights

$$s=\frac{\mu}{|\upsilon|^{1/\beta}} \quad (24)$$

where s is the step length of the lévy flight that is Levy and λ in Eq.(23). λ obeys the equation that $\lambda=1+\beta$, where $\beta=1.5$. $\mu=N(0,\delta_\mu^2)$ and $\upsilon=N(0,\delta_\upsilon^2)$ are both normal stochastic distributions with

$$\delta_\mu=\left[\frac{\Gamma(1+\beta)\times\sin(\pi\times\beta/2)}{\beta\times\Gamma((1+\beta)/2)\times 2^{(\beta-1)/2}}\right]^{\frac{1}{\beta}}$$
$$\delta_\upsilon=1 \quad (25)$$

The general steps of the improved sheep optimization (ISO) can be summarized in the pseudo code.

Steps of the ISO algorithm

(1) algorithm initialization
(2) calculate the fitness of each search agent
(3) calculate the best search agent
(4) while ($t<$ max_iteration)
(5) for each search agent
(6) update parameter
(7) update the position by eqs.(16—17)
(8) end for
(9) for each search agent
(10) update parameter
(11) selected another sheep at random
(12) if fitness(random)$<$fitness(i)
(13) update the position by eq.(20)
(14) else
(15) update the position by eq.(21)
(16) end if
(17) end for
(18) for each search agent
(19) update the position by eq.(22)
(20) end for
(21) calculate the fitness of each search agent
(22) update the best sheep if there is a better solution
(23) $t=t+1$
(24) end while
(25) output result

2.3 Flow of the multi-UAV cooperative path planning

Due to the potential parallel ability of the swarm algorithm, this paper constructs the cooperative path set of UAVs based on the ISO algorithm and the multi-population idea. The path of each UAV is represented by multiple subpopulations. The number of subpopulations is determined by the number of UAVs, and each subpopulation evolves independently. Information exchange is conducted only during path evaluation. As shown in Fig.5, during the evaluation of individuals in subpopulation 1, representative individuals selected from other sub-

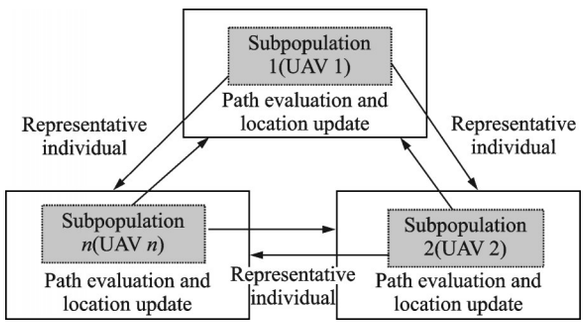


Fig.5 The process of multi-population coevolution

populations are combined with individuals in the current population to form a cooperative path, and the path cost function is used for evaluation as the fitness value of the individual, and then the fitness value of individuals in other subpopulations is calculated in turn. Individuals with small synergetic function value in the evolution process indicate that the path has good synergetic properties. Individuals in each subpopulation conduct information interaction and path evaluation with other subpopulations through the synergetic functions, and finally obtain multiple cooperative paths.

In path planning, the fitness value of each path includes not only the information of its own path cost, but also the information of cooperative interaction with other UAVs. In other words, each UAV will refer to the path information of other UAVs when planning its path. By choosing the path with less comprehensive cost, the path with better coordination can be obtained on the basis of satisfying the single-UAV flight cost index. The planned path can avoid collision and satisfy time constraints between multiple UAVs. The specific process is shown in Fig.6.

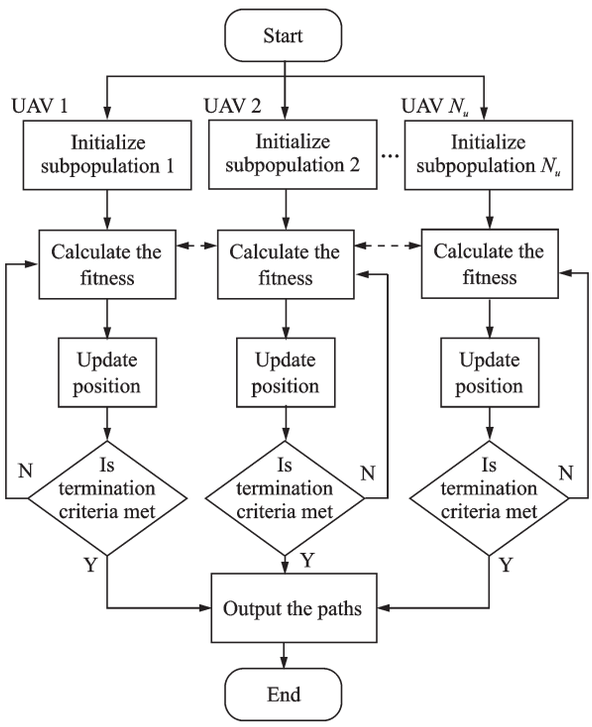


Fig.6 Cooperative path planning based on the ISO algorithm

3 Simulation Validation

3.1 Performance analysis and discussion

To verify the performance of the algorithm, we tested the performance of the improved sheep algorithm based on four classical benchmark functions[17]. The reference function is shown in Table 1, where "Dim" represents the dimension of the function, "Range" represents the boundary of the function search space, and $f_{\min}$ represents the minimum value of the function. Unimodal test function (f_1—f_2) with unique global optimal solution can test the global search ability and convergence of the

Table 1 Benchmark functions

Function	Dim	Range	$f_{\min}$
$f_1(x)=\sum_{i=1}^{n}x_i^2$	30	[−100,100]	0
$f_2(x)=\sum_{i=1}^{n}\lvert x_i\rvert+\prod_{i=1}^{n}\lvert x_i\rvert$	30	[−10,10]	0
$f_3(x)=\sum_{i=1}^{n}[\,x_i^2-10\cos(2\pi x_i)+10\,]$	30	[−5.12,5.12]	0
$f_4(x)=\frac{1}{4\,000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	30	[−600,600]	0

algorithm, while multimodal test function (f_3—f_4) with a variety of different local optimal solutions can test the ability of jumping out the local optimization.

In order to further test the performance of the algorithm and avoid contingency, the improved sheep algorithm was compared with the original SO, particle swarm optimization algorithm[18] (PSO) and gray wolf optimizer[19] (GWO) which have been widely used in recent years. Each algorithm was run 30 times on each benchmark function, and the number of each experimental population were set to 30 and the maximum number of iterations to 500. In order to make a fair comparison, all commonly used parameters of different algorithms, such as population size, dimension and maximum number of iterations, were set to the same. Related parameters of these algorithms are shown in Table 2. Test results (maximum, minimum, mean and standard deviation) are shown in Table 3, and the convergence curve is shown in Fig.7.

Table 2 Parameter values of each algorithm

Algorithm	Parameter
ISO	$a=2$
SO	$p=0.2,\varepsilon=1$
GWO	$a=2$
PSO	$c_1=c_2=2$ $w_{max}=0.9,w_{min}=0.2$

Table 3 Results of benchmark function test

F	Result	Algorithms				Rank
		ISO	SO	GWO	PSO	
f_1	Worst	3.4E−99	52.507 77	1.19E−26	0.001 535	1
	Best	2.6E−120	12.199 98	1.2E−29	6.06E−06	
	Mean	1.1E−100	29.331 59	1.29E−27	0.000 191	
	Std	6.2E−100	10.784	2.27E−27	0.000 361	
f_2	Worst	2.92E−66	6.324 963	3.76E−16	0.188 411	1
	Best	2.39E−77	2.492 533	1.71E−17	0.003 865	
	Mean	1.11E−67	3.961 088	9.81E−17	0.049 14	
	Std	5.32E−67	1.002 197	8.85E−17	0.041 166	
f_3	Worst	0	193.423 7	17.380 05	79.766 66	1
	Best	0	99.975 1	5.68E−14	26.221 24	
	Mean	0	146.539	2.588 45	51.363 83	
	Std	0	26.324 67	4.019 825	14.218 46	
f_4	Worst	0	1.294 569	0.019 912	0.022 219	1
	Best	0	1.096 585	0	1.62E−06	
	Mean	0	1.160 078	0.002 706	0.005 761	
	Std	0	0.050 48	0.005 68	0.007 798	

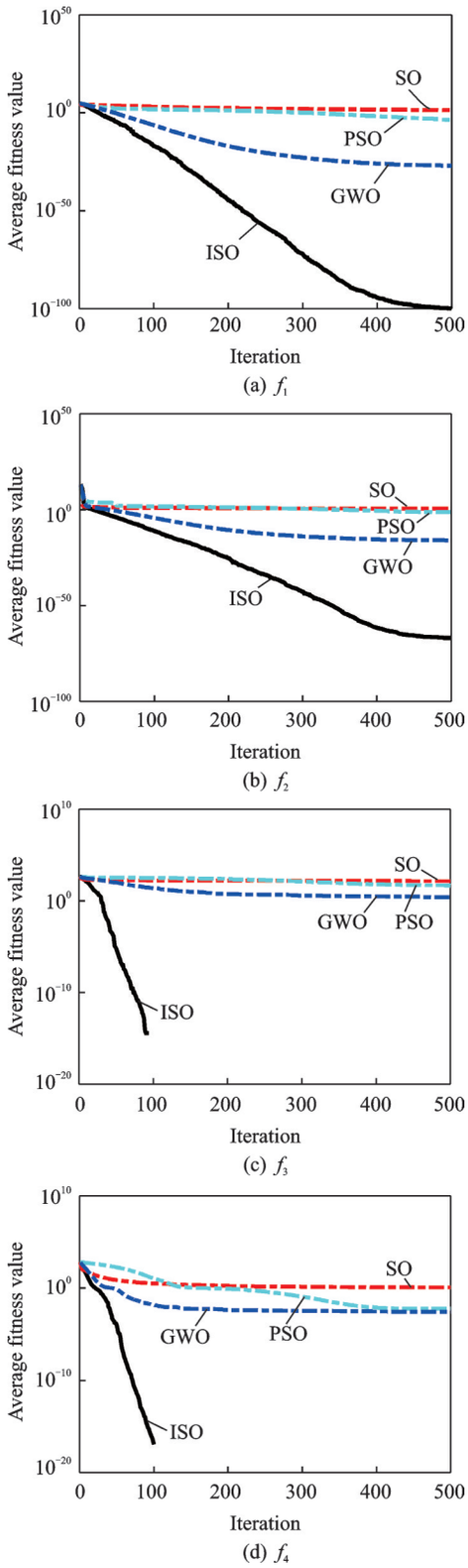


Fig.7 Convergence curves

The test results of the improved sheep algorithm on the benchmark function are obviously better than those of other algorithms, which reflects the advantages of the improved sheep algorithm in global search and local development. The improved

sheep algorithm has great potential in solving optimization problems.

3.2 Simulation environment setting and simulation analysis

The planning space was set as 100 km × 100 km × 500 m, including six peaks. The parameters of the original digital terrain model were set to $a=0.1$, $b=0.01$, $c=1$, $d=0.1$, $e=0.2$, $f=0.4$ and $g=0.02$. The height of the peak, horizontal coordinates of the highest point, and slope parameters are listed in Table 4. Multi-UAVs cooperative path planning was conducted under a known mission assignment scheme. In the simulation experiment, the path sub-population is initialized in accordance with the number of UAVs. The numbers of individuals in the sub-population, iterations, and path points were 50, 100 and 10, respectively. The weight coefficients of all cost functions corresponded to 0.4, 0.2, 0.1, 0.2 and 0.1. The flight speed range of the UAV was 40—60 m/s.

Table 4 Model parameters of peaks

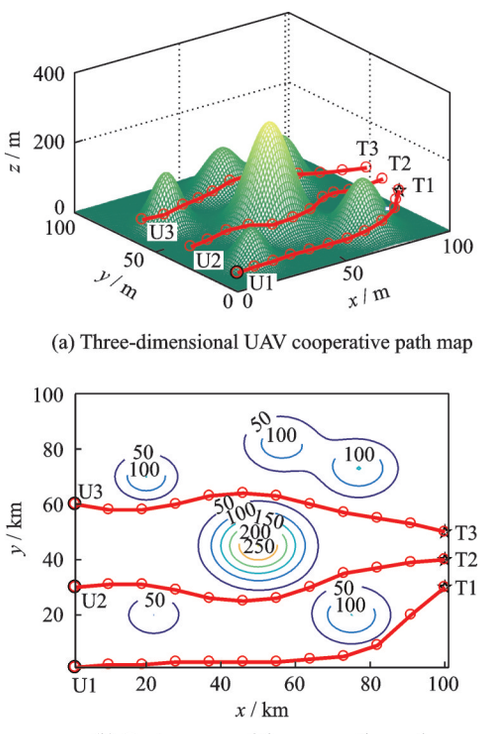
Number	Altitude/m	Center location (x,y)/km	Slope
1	130	(56,82)	(10,10)
2	150	(75,20)	(10,10)
3	300	(50,45)	(12,12)
4	100	(22,20)	(8,8)
5	150	(20,70)	(8,8)
6	150	(77,73)	(10,10)

Case 1 Three UAVs started from the starting point and arrived at the designated target point to perform tasks. The coordinates of the starting and target points are presented in Table 5.

The three-dimensional path planning and contour map of each UAV are displayed in Figs.8(a) and (b). The convergence curves of the path cost and synthetic path cost of each UAV are plotted in Figs.9(a) and (b). Fig.8 illustrates that all UAVs can effectively avoid the threat and reach the target

Table 5 Coordinates of the starting and target points of all UAVs (Case 1)

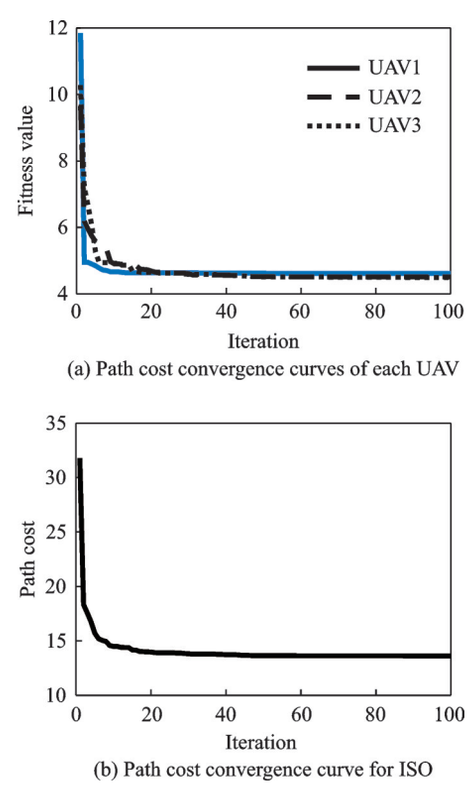
Number	Starting point	Target point
1	(1,1,0)	(100,30,70)
2	(1,30,0)	(100,40,70)
3	(1,60,0)	(100,50,70)



(a) Three-dimensional UAV cooperative path map

(b) Contour map of the cooperative path

Fig.8 Three-dimensional multi-UAV cooperative path planning and contour map (Case 1)



(a) Path cost convergence curves of each UAV

(b) Path cost convergence curve for ISO

Fig.9 Path cost convergence curve based on ISO (Case 1)

point. The cost function values of each UAV gradually converge with the increase of the iteration time, thereby verifying the effectiveness of the algorithm. Through the simulation, the flight time intervals (unit: s) and range (unit: km) of each UAV are presented in Table 6. The time intersection is [1 828.945 4, 2 562.788 5]. The time synergy requirement can be satisfied by setting different flight speeds for all UAVs.

Table 6 Range and flight time of each aircraft (Case 1)

Number	Range/km	Flight time/s
1	109.736 7	[1 828.945 4, 2 743.418 1]
2	102.511 5	[1 708.525 6, 2 562.788 5]
3	101.285 4	[1 688.089 2, 2 562.133 8]

Case 2 Four UAVs flied to two target points. The coordinates of the starting and the target points are listed in Table 7.

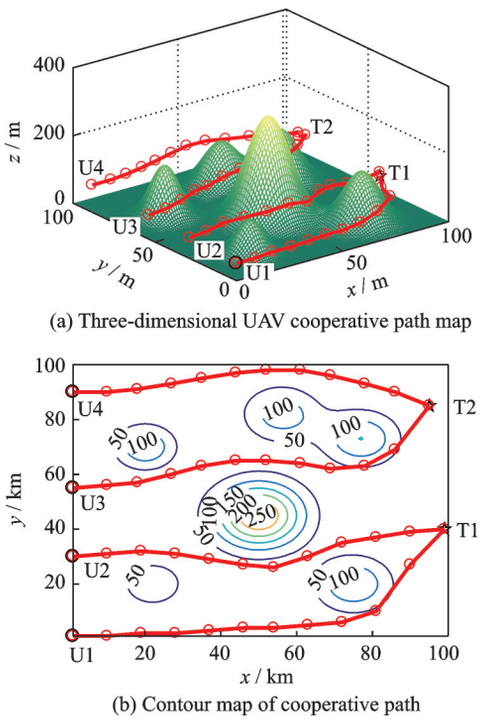
Table 7 Coordinates of the starting and target points of all UAVs (Case 2)

Number	Starting point	Target point
1	(1,1,0)	(90,40,100)
2	(1,30,0)	(95,40,100)
3	(1,55,0)	(95,85,100)
4	(1,90,0)	(80,85,100)

The planned path diagrams of UAVs were obtained, as depicted in Fig.10. The path cost convergence and synthetic cost curves of each UAV are plotted in Fig.11. Through the simulation, the flight time intervals (unit: s) and range (unit: km) of each UAV are presented in Table 8. The time intersection was [1 936.210 2, 2 443.968 2]. The planned paths and voyages were close and could effectively avoid obstacles. If the UAV is close to one another, then collisions can be avoided by setting different flight altitudes.

Case 3 Six UAVs flied to six target points. The coordinates of the starting and the target points are listed in Table 9.

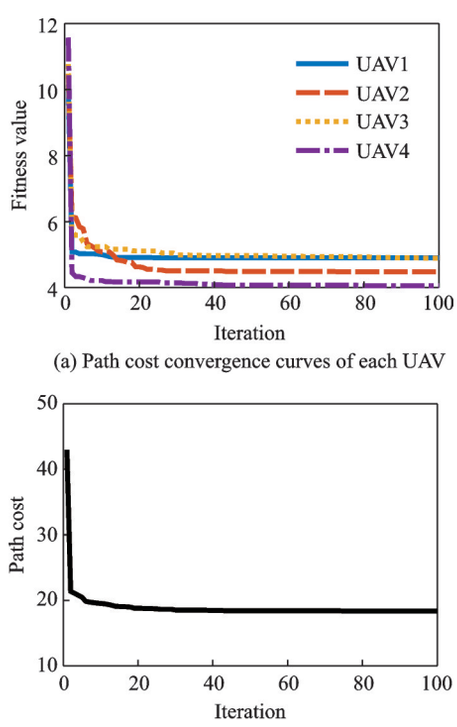
The planned path diagrams of UAVs were obtained, as depicted in Fig.12. The path cost conver-



(a) Three-dimensional UAV cooperative path map

(b) Contour map of cooperative path

Fig.10 Three-dimensional multi-UAV cooperative path planning and contour map (Case 2)



(a) Path cost convergence curves of each UAV

(b) Path cost convergence curve for ISO

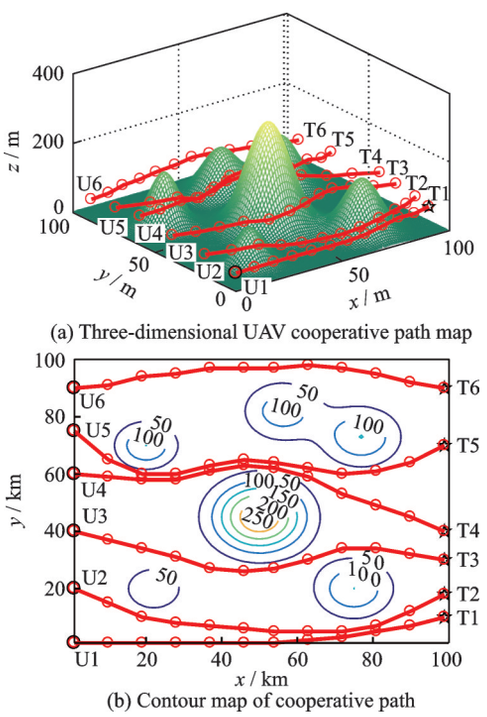
Fig.11 Path cost convergence curve based on ISO (Case 2)

Table 8 Range and flight time of each aircraft (Case 2)

Number	Range/km	Flight time/s
1	116.172 6	[1 936.210 2, 2 904.315 3]
2	101.306 5	[1 688.441 7, 2 532.662 6]
3	108.147 5	[1 802.458 7, 2 703.688 1]
4	97.570 6	[1 626.176 1, 2 439.264 1]

Table 9 Coordinates of the starting and target points of all UAVs (Case 3)

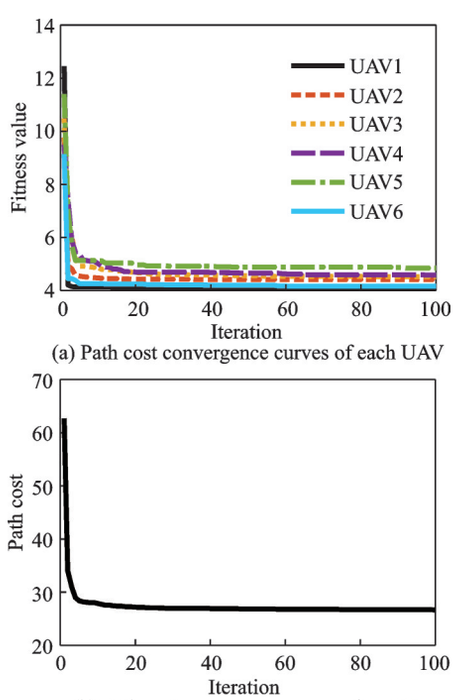
Number	Starting point	Target point
1	(1,1,0)	(99,10,70)
2	(1,20,0)	(99,18,70)
3	(1,40,0)	(99,30,70)
4	(1,60,0)	(99,40,70)
5	(1,75,0)	(99,70,70)
6	(1,90,0)	(99,90,70)



(a) Three-dimensional UAV cooperative path map

(b) Contour map of cooperative path

Fig.12 Three-dimensional multi-UAV cooperative path planning and contour map (Case 3)



(a) Path cost convergence curves of each UAV

(b) Path cost convergence curve for ISO

Fig.13 Path cost convergence curve based on ISO (Case 3)

gence and synthetic cost curves of each UAV are plotted in Fig.13. Through the simulation, the flight time intervals (unit: s) and range (unit: km) of each UAV are presented in Table 10. The time intersection is [1 779.351 9, 2 475.917 5]. The planned paths and voyages are close and can effectively avoid obstacles. If the UAV is close to one another, then collisions can be avoided by setting different flight altitudes.

Table 10 Range and flight time of each aircraft (Case 3)

Number	Range/km	Flight time/s
1	99.036 7	[1 650.611 7, 2 475.917 5]
2	104.315 7	[1 738.594 9, 2 607.892 4]
3	102.201 8	[1 703.363 5, 2 555.045 2]
4	104.176 3	[1 736.271 5, 2 604.407 3]
5	106.761 1	[1 779.351 9, 2 669.027 9]
6	99.752	[1 662.532 6, 2 493.798 9]

3.3 Comparative validation

Based on Case 3, the PSO, DE and GWO algorithms were used for multi-UAVs cooperative path planning. In addition, the improved sheep algorithm was compared with the latest two improved grey wolf algorithms IGWO[20] and CGWO[21]. The simulation results were compared with the proposed algorithm to verify the effectiveness of the improved strategy. Among these factors, the PSO algorithm parameters[20] included the number of particles as 50, learning factor $c_1 = c_2 = 2$, and inertia factor that decreased linearly from 0.96 to 0.2; the DE algorithm parameters[21] included the number of chromosomes as 50, the upper (0.6) and lower (0.2) bounds of scaling factor, and mutation rate (0.5) and crossover probability (0.6). The GWO, IGWO and CGWO algorithms were consistent with the ISO algorithm, and the numbers of individuals in the sub-population, iterations and path points were 50, 150 and 10, respectively. The algorithms were performed 30 times each. The average convergence curve after 100 iterations of each algorithm and the distribution of the minimum cost results after each iteration are obtained, as exhibited in Figs.14, 15.

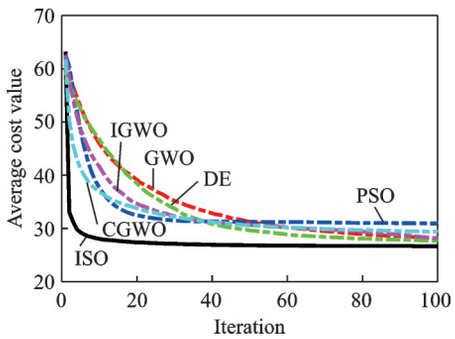


Fig.14 Average path cost convergence curves (30 times)

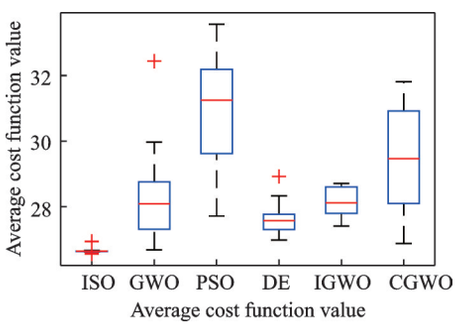


Fig.15 Distribution map of the minimum path cost value

Compared with PSO, DE, GWO, CGWO and IGWO algorithm, the final stable value of ISO was obviously better than the other algorithms, and the convergence speed was faster. The results showed that the improved sheep optimization was better than other algorithms in convergence speed and convergence accuracy and could effectively fullfill multi-UAVs cooperative path planning.

4 Conclusions

In this paper, a mathematical model of multi-UAVs cooperative path planning is established on the basis of three-dimensional planning space. It is easy to fall into the problem of local optimization and slow convergence rate if the traditional swarm intelligence algorithm is used to solve the multi-UAVs cooperative path planning problem. To address this, a new method which solves the three-dimensional cooperative path planning problem of multi-UAVs by using improved sheep optimization is proposed. The simulation results show that this method can obtain the cooperative path while satisfying the constraint conditions. Compared with the other algorithms, the improved sheep optimization algorithm is effective in solving the cooperative path planning, and the search accuracy and convergence speed are significantly improved.

However, only the path planning problem with known environmental information is considered in this paper. In the future research, we will model the dynamic path planning problem according to the possible emergency situations during the flight of UAVs and extend the application scenarios of the sheep optimization to dynamic cooperative path planning for multi-UAVs.

References

[1] FAN B, LI Y, ZHANG R, FU Q. Review on the technological development and application of UAV systems[J]. Chinese Journal of Electronics, 2020, 29(2): 199-207.

[2] YANG W, LEI, DENG J. Optimization and improvement for multi-UAV cooperative reconnaissance mission planning problem[C]//Proceedings of the 11th International Computer Conference on Wavelet Actiev Media Technology and Information Processing (ICCWAMTIP). [S.l.]: IEEE, 2014: 10-15.

[3] ZHAO Z, YANG J, NIU Y F, et al. A hierarchical cooperative mission planning mechanism for multiple unmanned aerial vehicles[J]. Electronics, 2019, 8(4): 443.

[4] CHEN X, CHEN X M. The UAV dynamic path planning algorithm research based on Voronoi diagram[C]//Proceedings of the 26th Chinese Control and Decision Conference (2014 CCDC). Changsha, China: IEEE, 2014: 1069-1071.

[5] BAUMANN M, LEONARD S, CROFT E A, et al. Path planning for improved visibility using a probabilistic road map[J]. IEEE Transactions on Robotics, 2010, 26(1): 195-200.

[6] MASOUD A A. Decentralized self-organizing potential field-based control for individually motivated mobile agents in a cluttered environment: A vector-harmonic potential field approach[J]. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2007, 37(3): 372-390.

[7] LU L, ZONG C, LEI X, et al. Fixed-wing UAV

path planning in a dynamic environment via dynamic RRT algorithm[C]//Proceedings of the International Conference of Mechanism and Machine Science, Asian MMS 2016. Singapore: Springer, 2016: 271-282.

[8] SZCZERBA R J, GALKOWSKI P, GLICJTEIN I S, et al. Robust algorithm for real-time route planning[J]. IEEE Transactions on Aerospace and Electronic Systems, 2000, 36(3): 869-878.

[9] YONG F M, LUO Z, LUO S X. Application of improved sparse A* algorithm in UAV path planning[J]. Information Technology Journal, 2013, 12 (17): 4058-4062.

[10] LIU Y, ZHANG X J, ZHANG Y, et al. Collision free 4D path planning for multiple UAVs based on spatial refined voting mechanism and PSO approach[J]. Chinese Journal of Aeronautics, 2019, 32(6): 1504-1519.

[11] DEWANGAN R K, SHUKLA A, GODFREY W W. Three dimensional path planning using grey wolf optimizer for UAVs[J]. Applied Intelligence, 2019, 49(6): 2201-2217.

[12] WOLPERT D H, MACREADY W G. No free lunch theorems for optimization[J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1): 67-82.

[13] QU D, XU L, LU Y, et al. A new swarm intelligence algorithm by simulating sheep behaviors[J]. Acta Electronica Sinica, 2018, 46(6): 1300-1305.

[14] CHENG Z, TANG Y, LIU Y. 3-D path planning for uav based on chaos particle swarm optimization[J]. Applied Mechanics and Materials, 2012, 232: 625-630.

[15] HU Z H. Research on some key techniques of UAV path planning based on intelligent optimization algorithm[D]. Nanjing: Nanjing University of Aeronautics and Astronautics, 2011. (in Chinese)

[16] YE Y, MIN C. A co-evolutionary method for cooperative UAVs path planning[J]. Computer Simulation, 2007, 24(5): 37-39.

[17] DIGALAKIS J G, MARGARITIS K G. On benchmarking functions for genetic algorithms[J]. International Journal of Computer Mathematics, 2001, 77 (4): 481-506.

[18] POLI R, KENNEDY J, BLACKWELL T. Particle swarm optimization[J]. Swarm Intelligence, 2007, 1 (1): 33-57.

[19] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer[J]. Advances in Engineering Software, 2014, 69: 46-61.

[20] LIU C A, WANG X P, LIU C Y, et al. Three-dimensional route planning for unmanned aerial vehicle based on improved grey wolf optimizer algorithm[J]. Journal of Huazhong University of Science and Technology(Natural Science Edition), 2017, 45(10): 38-42. (in Chinese)

[21] KOHLI M, ARORA S. Chaotic grey wolf optimization algorithm for constrained optimization problems[J]. Journal of Computational Design and Engineering, 2018, 5(4): 458-472.

Acknowledgement This work was supported in part by the Fundamental Research Funds for the Central Universities (No.NZ18008).

Authors Mr. **YANG Liuqing** is an associate professor. He received the M. S. degree in control engineering from Nanjing University of Aeronautics and Astronautics in 2012. His research is related to navigation guidance and control.

Mr. **WANG Pengfei** is pursuing the M.S. degree in Nanjing University of Aeronautics and Astronautics. His research is related to multi-UAV cooperative path planning and task allocation.

Author contributions Mr. **YANG Liuqing** contributed to the design, the discussion and background of the study, and conducted the analysis. Mr. **WANG Pengfei** programmed the new algorithm, interpreted the results, and wrote the manuscript. Mr. **ZHANG Yong** contributed to the discussion and the background of the study. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing inrests.

(Production Editor: ZHANG Bei)

基于羊群算法的多机协同航路规划

杨柳庆[1,2]，王鹏飞[3]，张　勇[1,2]

（1.南京航空航天大学无人机研究院，南京 210016；
2.南京航空航天大学中小型无人机先进技术工业和信息化部重点实验室，南京 210016；
3.南京航空航天大学自动化学院，南京 211106）

摘要：针对传统群智能算法解决多机协同航路规划问题易陷入局部最优和收敛速度慢的问题，提出了一种基于羊群算法的多机航路规划方法。首先在三维规划空间基础上，根据航路规划要求建立多机协同代价函数模型，并结合多种群思想构造初始航迹集合。然后利用羊群算法对航路规划问题进行求解得到多条协同航路。仿真结果表明，该算法能够满足航路规划相关约束，实现多无人机协同航路规划，相比较灰狼算法（Gray wolf optimizer，GWO）、差分进化算法（Differential evolution algorithm，DE）、粒子群算法（Particle swam optimization，PSO）、混沌灰狼算法（Chaotic gray wolf optimizer，CGWO）和改进灰狼算法（Improved gray wolf optimizer，IGWO），改进羊群算法收敛速度和搜索精度均有明显提高。

关键词：多机协同；航路规划；群智能算法；多种群；改进羊群算法