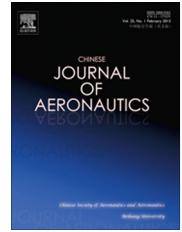




Chinese Society of Aeronautics and Astronautics  
& Beihang University  
Chinese Journal of Aeronautics

cja@buaa.edu.cn  
www.sciencedirect.com



# Trajectory optimization of multiple quad-rotor UAVs in collaborative assembling task



Chen Yongbo, Yu Jianqiao\*, Mei Yuesong, Zhang Siyu, Ai Xiaolin, Jia Zhenyue

School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China

Received 3 June 2015; revised 18 August 2015; accepted 22 October 2015  
Available online 22 December 2015

## KEYWORDS

Hierarchic optimization strategy;  
Parallel CFO-GA algorithm;  
Path planning;  
Six-degree-of-freedom rigid-body dynamic model;  
Trajectory optimization;  
Trajectory planning

**Abstract** A hierarchic optimization strategy based on the offline path planning process and online trajectory planning process is presented to solve the trajectory optimization problem of multiple quad-rotor unmanned aerial vehicles in the collaborative assembling task. Firstly, the path planning process is solved by a novel parallel intelligent optimization algorithm, the central force optimization-genetic algorithm (CFO-GA), which combines the central force optimization (CFO) algorithm with the genetic algorithm (GA). Because of the immaturity of the CFO, the convergence analysis of the CFO is completed by the stability theory of the linear time-variant discrete-time systems. The results show that the parallel CFO-GA algorithm converges faster than the parallel CFO and the central force optimization-sequential quadratic programming (CFO-SQP) algorithm. Then, the trajectory planning problem is established based on the path planning results. In order to limit the range of the attitude angle and guarantee the flight stability, the optimized object is changed from the ordinary six-degree-of-freedom rigid-body dynamic model to the dynamic model with an inner-loop attitude controller. The results show that the trajectory planning process can be solved by the mature SQP algorithm easily. Finally, the discussion and analysis of the real-time performance of the hierarchic optimization strategy are presented around the group number of the waypoints and the equal interval time.

© 2016 Production and hosting by Elsevier Ltd. on behalf of CSAA & BUAA. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

## 1. Introduction

Unmanned aerial vehicles (UAVs) have become increasingly attractive for missions in which the human presence is dangerous or difficult. Among these UAVs, the unmanned quad-rotor helicopters have become increasingly popular platforms for the study of the UAVs from many viewpoints, such as the reconnaissance, the communications relay, the individual combat and so on.<sup>1</sup> The UAVs trajectory optimization that deals with the time evolution of the flight path is a very important part of the UAVs autonomous control system. Many researches about the quad-rotor helicopters focus on the

\* Corresponding author. Tel.: +86 10 68912407.

E-mail addresses: [bit\\_chenyongbo@163.com](mailto:bit_chenyongbo@163.com) (Y. Chen), [jianqiao@bit.edu.cn](mailto:jianqiao@bit.edu.cn) (J. Yu), [mys001@bit.edu.cn](mailto:mys001@bit.edu.cn) (Y. Mei), [ylww@bit.edu.cn](mailto:ylww@bit.edu.cn) (S. Zhang), [bitaixiaolin@163.com](mailto:bitaixiaolin@163.com) (X. Ai), [jiazhenyue@163.com](mailto:jiazhenyue@163.com) (Z. Jia).

Peer review under responsibility of Editorial Committee of CJA.



Production and hosting by Elsevier

controller design field, but the trajectory optimization process specifically aiming at the quad-rotor helicopters, is scarce.

In many references, the path planning and trajectory planning are similar. They can be collectively called the trajectory optimization. But in the strict sense, the UAVs trajectory planning process is different from the UAVs path planning process. The path planning is a process in which the UAV finds a three-dimensional (3D) space path from the starting point to the destination. The 3D space path is a static geometry path. It does not include the concept of time.<sup>2</sup> However, the results of the trajectory planning process are the time-varying flying paths. The results include the flying state of the vehicles. Generally speaking, the model and solution algorithm of the trajectory planning problem are more complicated than the ones of the path planning problem. However, many ideas from the path planning algorithms can help to solve the UAVs trajectory planning problem. For many simple scenarios, the simpler path planning algorithms can offer some sketchy and flyable flying results for the UAVs automatic control system quickly and efficiently. Both of them are very significant to the UAVs autonomous control system.

Because of the simpler model, the UAVs path planning problem has been solved by many methods, such as A\* algorithm,<sup>3</sup> artificial potential field (APF) method,<sup>4,5</sup> rapidly-exploring random tree (RRT) algorithm,<sup>6,7</sup> a large number of intelligent optimization algorithms<sup>8,9</sup> and so on. Recently, the intelligent optimization algorithms have drawn a lot of attention. For example, Duan et al. introduced many novel optimization algorithms to solve this problem without assuming kinematic and dynamic constraints, including the max–min adaptive ant colony optimization approach,<sup>10</sup> the chaotic predator–prey biogeography-based optimization (CPPBBO) algorithm,<sup>11</sup> the improved gravitational search algorithm<sup>8</sup> and the chaotic artificial bee colony (ABC) approach.<sup>12</sup> In order to use these algorithms, the original path planning problem which is an infinite dimensional problem is simplified to the finite dimensional optimization problem by the partition of the two-dimensional (2D) planning space. Although these methods can ensure the near-optimality of the path under some given objective functions, the dynamic and kinematic model of the UAVs is entirely ignored. These planning results are always unacceptable for the general UAVs, especially the fixed-wing aircraft. Even for the quad-rotor helicopters, these results will strongly limit the speed scheme of the UAVs.

Because through the trajectory planning algorithm the time-varying state variables can be obtained, the models of the trajectory planning problem need to be closer to the real aircraft model. Obviously, the high-fidelity model will help the control system to get the appropriate control commands that will maintain the actual flying vehicle on the obtained trajectories accurately. But the complicated model signifies the huge amount of calculation and the large difficulty of the planning process. So the suitable model of the planning object is very important for the computing processes and planning results.

Essentially, the trajectory planning problem is a multi-constraints optimal control problem. The most intuitive approach is to use the optimal control theory. Yao and Zhao<sup>13</sup> put forward the model predictive control (MPC) algorithm to solve the UAV trajectory planning problem in an uncertain environment. In Ref.<sup>14</sup>, a novel finite horizon suboptimal controller is applied to solve the trajectory planning problems for

the approach and landing (A&L) phase of the reusable launch vehicle (RLV). In this work, the model of the vehicle is a three-degree of freedom (3-DOF) longitudinal particle model. In addition to the optimal control theory, there are some other algorithms which can solve this problem. In Ref.<sup>15</sup>, Zhang et al. applied a planning algorithm based on the inverse dynamics optimization to solve the ground attack trajectory planning problem of the unmanned combat aerial vehicle (UCAV) which is mathematically formulated as a receding horizon optimal control problem (RHC-OCP). In this work, the model of the UAV is described by the kinematic and dynamic model according to a full-blown 3-DOF particle model. In Ref.<sup>16</sup>, Kamyar and Taheri used the differential evolution-sequential quadratic programming (DE-SQP) method and the particle swarm optimization-sequential quadratic programming (PSO-SQP) method to solve the trajectory planning problem based on the high-fidelity, 6-DOF dynamic model with the integration of accurate aerodynamic and propulsion models. In the existing literature, most of these papers focused on the fixed-wing UAV. There are only a small number of scholars researching the trajectory planning problem of the quad-rotor helicopters whose model is closer to the real UAV kinematic and dynamic model. Based on the commonly employed quad-rotor UAV model, Chamseddine et al.<sup>17</sup> used the Bézier polynomial function and the differential flatness method to solve the trajectory planning problem of the quad-rotor UAV.

The above references only show the single UAV path and trajectory planning method. Some multiple UAVs cooperative trajectory planning algorithms are proposed in some references. In the path planning area, Eva et al.<sup>18</sup> presented a path planner for multiple UAVs (multi-UAVs) based on the multiple coordinated agents co-evolution evolutionary algorithms (MCACEA) for the realistic scenarios. In Ref.<sup>19</sup>, a path planning method based on the RRT was developed to generate paths for multi-UAVs in real time. In the trajectory planning area, Gu et al.<sup>20</sup> put forward a virtual motion camouflage (VMC) to solve the cooperative trajectory planning problem of the multi-UAVs, combining with the differential flatness theory, the Gauss pseudospectral method (GPM) and the non-linear programming. In this work, the model of each UAV is described by the kinematic and dynamic model according to a full-blown 3-DOF particle model. The main difficulties of the multi-UAVs trajectory planning problem are the amount of calculation and the cooperative way. Of course, the amount of calculation of the problem increases with the size of the cluster and the complexity of the model. At the same time, the cooperative way is also very important.

The motivation of this paper is to solve the cooperative trajectory optimization problem of multiple quad-rotor unmanned aerial vehicles. Our concern is to find more efficient, more realistic and suboptimal trajectories for multiple quad-rotor UAVs without making too many simplifying assumptions on the trajectories. The existing references focusing on the similar problems have some shortcomings. The first problem is the lack of the reasonable systematic research about the overall solution frameworks. The other problem is that in order to simplify the problem, there are too many simplifying assumptions on the aircraft models. Faced with these problems, the hierarchic optimization strategy based on the offline path planning process and online trajectory planning process is put forward to solve this trajectory optimization problem. This

hierarchical optimization strategy can improve the solution efficiency, enhance the factuality of the result and ensure the real-time performance of the results.

The main content, the obtained results and the contributions of this paper are presented in this paragraph. The hierarchical optimization strategy includes the path planning process and trajectory planning process. In order to make the trajectory planning process more efficient and more real-time, the off-line path planning process is introduced to complete the discrete preliminary trajectory points planning work. The dynamic and kinematic constraints of the UAV in this process just include the velocity constraint and normal overload constraint. A novel parallel intelligent optimization algorithm, the central force optimization-genetic algorithm (CFO-GA), which combines the CFO algorithm with the GA is put forward creatively to solve the multiple quad-rotor UAVs path optimization which is a multi-objective problem. The simulation results show that through the CFO-GA algorithm, the better objective function and faster convergence rate can be obtained compared with other algorithms. At the same time, because of the immaturity of the CFO algorithm, the convergence analysis of the CFO algorithm is completed by the stability theory of discrete time linear time-variant system. This proof process is original. Then, by the help of the result points, the trajectory planning process is finished based on the appropriate simplified 6-DOF rigid-body dynamic model of the quad-rotor helicopters whose inner-loop attitude is controlled by the full-blown PID controller. This controlled optimization object is unique and original. What's more, the cubic B-spline parameterization algorithm and the concept of the security time are introduced to compute the optimal control inputs and the optimal time of this optimal control problem. The full-blown SQP algorithms are introduced to compute the final suboptimal trajectories. The simulation results show that the hierarchical optimization strategy algorithm is effective for the multi-UAVs trajectory optimization problem. At last, the discussion and analysis on the real-time performance of the hierarchical optimization strategy are presented around the group number of the waypoints and the equal interval time. The obtained results show that the real-time demand of this system can be guaranteed by some parameters.

## 2. Problem description, analysis and solution frame

The multi-UAVs trajectory optimization problem widely appears in many conditions which include the surveillance, the search, the rescue missions, the geographic studies, the military and the security applications. First of all, it is very necessary to confirm the details of the textual problem.

### 2.1. Problem description and analysis

At first, the research objects of this paper are the quad-rotor UAVs. Then, this paper focuses on the multi-UAVs collaborative assembling problem which means that some UAVs fly from their own starting points to the same target point and reach the target point at the same time. The schematic diagram of this task is shown in Fig. 1. In the flying process, the multi-UAVs need to avoid the complex terrain and optimize the flight performance of the whole UAV fleet. In this problem, there are several key demands as follows: (1) the planning

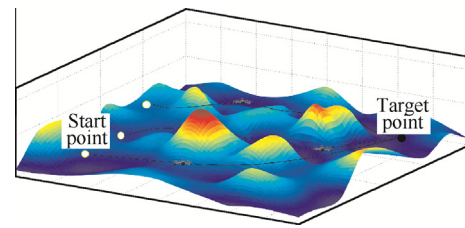


Fig. 1 Sketch diagram of multi-UAVs collaborative assembling problem.

results are effective to the real dynamic and kinematic constraints of the quad-rotor UAVs; (2) the assembling time intervals among these UAVs need to be minimized; (3) the flight performance of the whole UAV fleet needs to be optimized.

### 2.2. Solution frame

At first, the multiple quad-rotor UAVs collaborative assembling problem is a multi-objects planning problem. Obviously, the computational efficiency of the solving algorithm is very important for this problem. To solve the efficiency problem, the parallel concept is introduced to compute different individuals. But the collaborations among the UAV fleet are the taboo of the parallel algorithms. The parallel algorithms are more suitable for the UAV fleet whose individuals are independent without connection. So the collaboration information among these quad-rotor UAVs needs to be disposed into relatively independent parts. This work will be finished in the path planning process.

Then, in order to improve the real-time ability of the algorithm, the whole solution is divided into two processes. The first process is the early offline path planning which is computed by the intelligent optimization algorithms. In this process, the problem is degraded into a path planning problem whose purpose is to obtain a series of optimized trajectory points for the following trajectory planning problem. The offline results can help to separate the whole trajectory into several small trajectory segments. By this way, the huge planning problem can be solved by each small planning problem whose calculation amount and planning difficulty are significantly smaller than those of the original problem. At the same time, because of the hierarchical optimization strategy, the adjustment of the distance between each two waypoints helps to control the length of the computing time. In other words, it can help to realize the request of the online planning. In addition, because the small scene for the small trajectory planning problem can improve the proportion of the feasible solution in the definitional domain of the independent variables, the real-time ability of the small trajectory planning problem is further improved.

Subsequently, the second process is the online trajectory planning. The trajectory planning problem which is an optimal control problem can be solved typically by two approaches, namely, indirect approach and direct approach. The direct methods are to convert the continuous optimal control problem into a nonlinear parameter optimization problem. Generally speaking, the direct methods are easy to solve, due to the avoidance of the two point boundary value problem (TPBVP). So the direct method is applied to this paper. In this paper, the

control inputs of the whole system are parameterized by the cubic B-spline whose control points are the independent variable of the nonlinear parameter optimization problem. At the same time, the 6-DOF rigid-body dynamic model of the quad-rotor helicopters with attitude inner-loop controller is used in the optimization process to improve the practicability of the trajectory planning results.

The specific design flowchart is shown in Fig. 2.

### 3. Path planning process

The multiple quad-rotor UAVs path planning process is the first step of the whole algorithm. This process needs to obtain the discrete waypoints of the paths for the trajectory planning process. In this section, the whole path planning process, including the environmental modeling, the optimal model creation, the solution algorithm and the simulation result, is finished.

#### 3.1. Environmental modeling

Modeling of the planning space is a key part of the UAV path planning. The way used to describe the obstacles has an effect on the independent variable design, the representation of the path and the search algorithm. The reasonable modeling way can simplify the problem and improve the solution efficiency.

The commonest environment modeling way is the digital map. The digital map means the parameterization of the terrain in the flying range of the UAVs. It mainly includes the modeling of the mountains, the hills, the cities and so on. There are many ways to create the digital map. In this paper, the feature point interpolation way is used,<sup>9</sup> and the main process is as follows.

Firstly, the space rectangular coordinate system is established in the planning space. The coordinates of any point on the ground can be obtained. Secondly, the terrain in the planning space is divided into  $N_x$  and  $N_y$  parts along the  $x$ -

axis and  $y$ -axis. By this way, we can obtain  $N_x \times N_y$  feature points whose coordinates are  $(x_{i_1}, y_{i_2}, z(x_{i_1}, y_{i_2}))$  ( $i_1 = 1, 2, \dots, N_x; i_2 = 1, 2, \dots, N_y$ ), where  $z(x_{i_1}, y_{i_2})$  means the altitude coordinates. Then, these feature points are interpolated by the 4-point spline interpolation method, and we can get the digital topographic information bank. At last, by the visualization technology, the three-dimensional digital map is built. Besides the terrain, the threat and bad weather zone also need to be modeled. Some papers such as Ref.<sup>21</sup> regard them as the “soft obstacle”. The UAVs can fly into these soft obstacles. But in this paper all of them are regarded as no-fly zone where the trajectories of UAV cannot be allowed. The terrain, the threat and the bad weather zone are collectively referred to as the obstacle zero. All of the scenarios are bounded within a horizontal-plane box of  $6500 \text{ m} \times 5500 \text{ m} \times 1200 \text{ m}$ . The whole planning space is shown in Fig. 3.

#### 3.2. Optimal model creation

As for the optimization problem, the independent variables design, the objective function and the constraints are the basic elements. The multiple quad-rotor UAVs path planning problem itself is a nonlinear complex constrained optimization problem. The general form of the optimization problem<sup>22</sup> is:

$$\begin{aligned} \min_x \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & h_{k_1}(\mathbf{x}) = 0 \quad k_1 = 1, 2, \dots, l_1 \\ & g_{k_2}(\mathbf{x}) \leq 0 \quad k_2 = 1, 2, \dots, l_2 \\ & \mathbf{x} \in D \subseteq \mathbf{R}^n \end{aligned} \quad (1)$$

where  $\mathbf{x}$  is the independent variable;  $f$  is the objective function;  $D$  means the feasible region of the independent variable;  $h_{k_1}$  and  $g_{k_2}$  are the equality constraints and inequality constraints, respectively;  $l_1$  and  $l_2$  mean the number of the constraints;  $n$  is the dimension of the independent variable.

The detailed descriptions of the problem will be shown as follows.

##### 3.2.1. Independent variables

The design of the independent variables is the indispensable element of the optimization problem. It directly influences the complexity of the model and the solver, so it is very important to define the constructed variables properly. The independent variables of the multiple quad-rotor UAVs path planning problem are the coordinate representing way of the waypoints. In the path planning process, the model of the UAVs is reduced to the free particle model. Then, the simple polyline

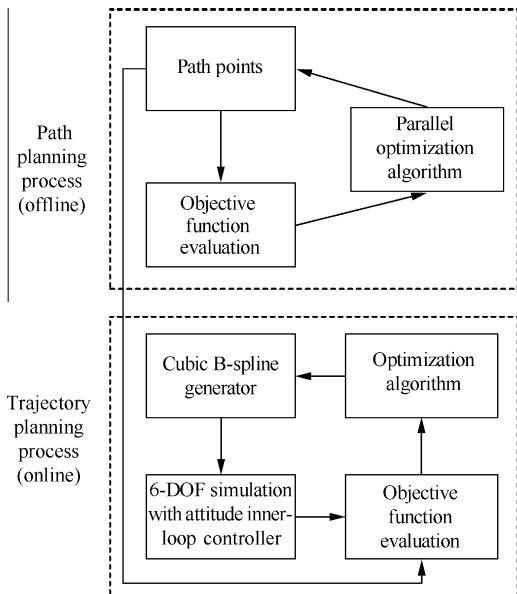


Fig. 2 Flowchart of hierarchic optimization strategy.

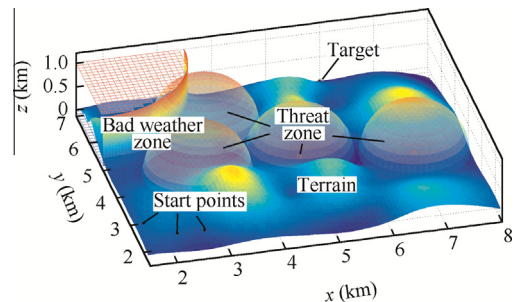


Fig. 3 Whole planning space.



paths generated by the optimization algorithm are composed of the line segments and the characteristic waypoints. At last, the smooth paths and discrete waypoints of the UAVs are obtained by the 3D circular curve.

Because this problem is the collaborative assembling problem whose targets are the same, the cylindrical coordinate system centered on the target point  $\mathbf{T} = (T_x, T_y, T_z)$  is introduced to parameterize the independent variables. The specific parametric processes are as follows: firstly, make a straight line  $l_T$  which passes through the target point and parallel to the  $z$ -axis; then, set up the cylinders according to  $l_T$ . The number of the cylinders is  $N$ , and  $N$  is the trunc of  $l_m/r$ ,  $r$  is the minimum radius of the coaxial cylinders (preestablished), and  $l_m$  is the furthest horizontal distance between the start points  $\mathbf{S}_j$  ( $j = 1, 2, \dots, M$ ) and target point. So the radiuses are  $i \times r$  ( $i = 1, 2, \dots, N$ ), where  $i$  is the serial number of the cylinders. The characteristic waypoints  $\mathbf{X}_{i,j}$  of the UAVs locate on the surface of the corresponding cylinder. Define the independent variable  $\mathbf{x}$  based on the cylindrical coordinate system as

$$\begin{aligned} \mathbf{x} &= [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_j, \dots, \mathbf{x}_M] \\ &= \underbrace{[\theta_{1,1}, \theta_{2,1}, \dots, \theta_{N,1}, h_{1,1}, h_{2,1}, \dots, h_{N,1}]}_{\text{UAV}_1}, \dots, \text{UAV}_j, \dots, \\ &\quad \underbrace{[\theta_{1,M}, \theta_{2,M}, \dots, \theta_{N,M}, h_{1,M}, h_{2,M}, \dots, h_{N,M}]}_{\text{UAV}_M} \end{aligned} \quad (2)$$

where  $\theta_{i,j}$  means the scalar from the target point to the characteristic waypoint of the  $j$ th UAV on the  $i$ th cylinder surface;  $h_{i,j}$  is the ordinate value of the characteristic waypoint of the  $j$ th UAV on the  $i$ th cylinder surface;  $M$  is the total number of the UAVs (in this paper,  $M = 3$ ).

Based on the definition of the independent variable  $\mathbf{x}$ , the characteristic waypoint of the  $j$ th UAV on the  $i$ th cylinder surface is

$$\mathbf{X}_{i,j} = (T_x - ir \cos \theta_{i,j}, T_y - ir \sin \theta_{i,j}, h_{i,j}) \quad (3)$$

Then, the smooth paths of the UAVs are obtained by the 3D inscribed circular curve. The turning radius  $r_1$  of these circular curves can be limited by the rated running conditions of the quad-rotor UAV. At last,  $M \times N_w$  discrete waypoints  $\mathbf{o}_{t,j}$  ( $t = 1, 2, \dots, N_w$ ) are chosen uniformly in these 3D curve paths. And they meet:

$$\begin{cases} \mathbf{o}_{1,j} = \mathbf{S}_j \\ \mathbf{o}_{N_w,j} = \mathbf{T} \end{cases} \quad (4)$$

### 3.2.2. Objective function

The objective of this part is to plan the paths for the UAVs from the starting points to the target. Meanwhile, the time difference of the whole fleet is minimized. Obviously, the total time difference among these UAVs is one of the objective functions. To address this problem, there are some references considering the flying time coordination problem, especially in the mission planning problem. The main time adjustment ways include the speed adjustment, the maneuver adjustment, the path length adjustment and so on. Chandler et al.<sup>23</sup> adjusted the time by changing the velocities of the UAVs. Alighanbari et al.<sup>24</sup> ensured the time coordination by the circulating flight way. Beard et al.<sup>25</sup> changed the trajectory length to realize that the UAVs reach the target at the same time. Although they can solve the time coordination problem partly, these ways cannot

ensure the optimality of each UAV or the whole UAV fleet. The UAVs waste the flight time and energy in solving this problem. Actually, the coordination of the takeoff time is the best way to solve this problem. We only need to change the takeoff times of the UAVs based on the elapsed flight time from the starting points to the target without considering changing the path. In this way, the minimized flight time and energy are the most important objectives. Because the model of the UAVs in the path planning problem is simplified into the 3-DOF free particle model, we restrict the speed scheme of the UAVs to the constant speed. Therefore, the flight time and energy optimization problem is further simplified into the shortest route distance problem. What is more, the total length of the UAVs paths is the commonest objective function in the references. The approximate expression of the objective function is shown as

$$l(\mathbf{x}) = \sum_{j=1}^M \sum_{t=1}^{N_w-1} \|\mathbf{o}_{t+1,j} - \mathbf{o}_{t,j}\| \quad (5)$$

### 3.2.3. Constraint condition

The constraint conditions of the optimization problem for the common UAV path planning process include the performance constraint and collision avoidance constraint.

Although the model of the path planning process has been greatly simplified, this part is still the foundation of the hierarchic optimization strategy. The path planning process cannot totally ignore the dynamic characteristic of the UAVs. The better flight performance can help the following trajectory planning process to obtain a better result. In fact, the performance constraint of the quad-rotor UAVs is much simpler than the one of the fixed-wing UAVs. They can complete the hovering and low speed flight maneuver which is very difficult for the fixed-wing UAVs. Originally, the usual turning radius constraint is not suitable for the quad-rotor UAVs. But we think that the straighter path can reduce the burden of the control system and ensure the high efficiency of the path. So the steering angles between each two adjacent line segments are introduced to limit the straightness of the path. The feasible scope of the following line segment of a path is shown in Fig. 4. The equation of the performance constraint is as

$$|\delta_{i,j}| \leq \delta_{\max} \quad i = 1, 2, \dots, N-1; j = 1, 2, \dots, M \quad (6)$$

where  $\delta_{i,j}$  is the angle between each two adjacent line segments, and  $\delta_{\max}$  the constraint value of the steering angle.

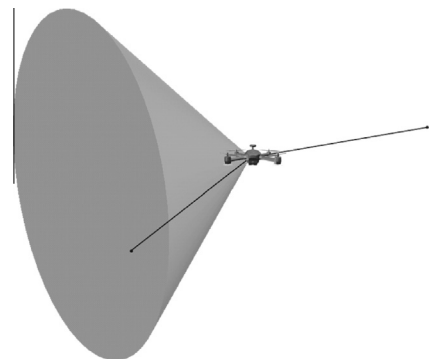


Fig. 4 Feasible scope of following line segment of path.

In addition, the collision avoidance constraint is essential for this problem. The planning results of these UAVs cannot pass through the terrain, the threat and the bad weather zone in the planning space. So the collision avoidance constraint of this problem is shown in the following expression:

$$o_{t,j}(3) > z_r \quad (7)$$

where  $o_{t,j}(3)$  is the ordinate component of  $o_{t,j}$ , and  $z_r$  the corresponding ordinate of the obstacle zone.

### 3.2.4. Optimization model

In conclusion, according to the typical optimal model Eq. (1), the nonlinear optimization model of the multiple quad-rotor UAVs path planning problem is

$$\begin{aligned} \min_{\mathbf{x}} \quad & l(\mathbf{x}) \\ \text{s.t.} \quad & |\delta_{ij}(\mathbf{x})| \leq \delta_{\max} \\ & o_{t,j}(3) > z_r \end{aligned} \quad (8)$$

### 3.3. Optimization algorithms

Although the path planning process is finished offline, the higher efficiency can save the computing resource and total response time. In the real combat environment, the whole simulation time directly affects the operational effectiveness of the system. Considering the above demand, the multiple quad-rotor UAVs path planning problem is solved with the parallel CFO-sequential quadratic programming (CFO-SQP) and the parallel CFO-GA algorithms separately.

The idea of the parallel computing is introduced to improve the computational efficiency of the algorithm for the multi-agent problem. The parallel implementation of the optimization algorithm is developed, as depicted in Fig. 5. The independent variable  $\mathbf{x}$  is split into  $N$  parts, and each part represents the path of one UAV in the whole UAV fleet. Then each part of  $\mathbf{x}$  can be computed in different computing core. By this way, this problem is solved efficiently. But for the parallel optimization problem, the cooperative relation among these UAVs is an inevitable problem. The main relation exists in the computing of the objective function. In Section 3.2.2, the objective function has already been simplified due to considering the coordination problem by changing the takeoff times.

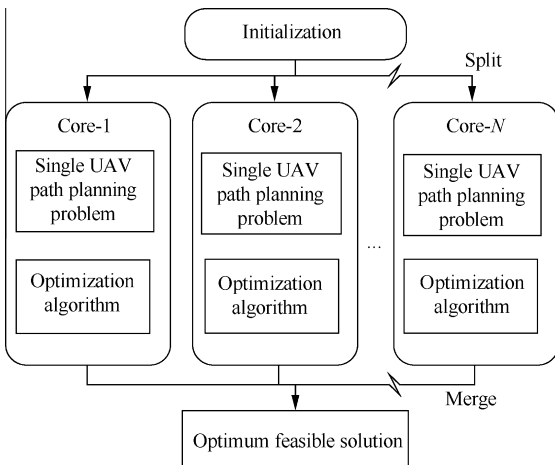


Fig. 5 Parallel optimization algorithm.

Therefore, the objective function is divided into  $N$  independent objective functions. The new objective functions are shown as

$$l_j(\mathbf{x}_j) = \sum_{t=1}^{N_w-1} \|o_{t+1,j} - o_{t,j}\| \quad (9)$$

where  $l_j(\mathbf{x}_j)$  is the objective function of the  $j$ th UAV and  $\mathbf{x}_j$  the independent variable of the  $j$ th UAV path planning problem.

The CFO algorithm is a new intelligent multi-particle optimization method, in which the search algorithm is inspired by the law of gravity. The particles update their positions in the search space according to a rule that each particle attracts every other particle with the virtual gravity force. The virtual mass of each particle is dependent on its objective function. The particles move according to the Newton's law of motion:

$$\mathbf{p}_m^{k+1} = \mathbf{p}_m^k + \mathbf{v}_m^k \Delta t + \frac{1}{2} \mathbf{a}_m^k \Delta t^2 \quad m = 1, 2, \dots, N_p \quad (10)$$

where  $\mathbf{p}_m^k \in \mathbf{R}^{1 \times 2N}$  is the position of the  $m$ th particle in the  $k$ th iteration;  $\mathbf{v}_m^k$  is the velocity vector of the  $m$ th particle in the  $k$ th iteration. In the original CFO method,  $\mathbf{v}_m^k$  is ignored ( $\mathbf{v}_m^k = \mathbf{0}_{1 \times 2N}$ );  $\Delta t$  is the time constant;  $N_p$  means the particle number;  $\mathbf{a}_m^k$  is the acceleration of the  $m$ th particle in the  $k$ th iteration, and it is defined as follow:

$$\begin{cases} \mathbf{a}_m^k = G \sum_{\substack{n=1 \\ n \neq m}}^{N_p} U(f(\mathbf{p}_n^k) - f(\mathbf{p}_m^k)) \\ \cdot (f(\mathbf{p}_n^k) - f(\mathbf{p}_m^k))^\alpha \cdot \frac{\mathbf{p}_n^k - \mathbf{p}_m^k}{\|\mathbf{p}_n^k - \mathbf{p}_m^k\|^\beta} \\ m = 1, 2, \dots, N_p \end{cases} \quad (11)$$

where  $G$  is the gravitation constant;  $U(\varpi)$  is the judgement function, and it meets  $U(\varpi) = \begin{cases} 1 & \varpi \geq 0 \\ 0 & \varpi < 0 \end{cases}$ ,  $\varpi \in \mathbf{R}$ ;  $\alpha$  and  $\beta$  are the tuning parameters, and they satisfy  $\alpha > 0$  and  $\beta > 0$ ;  $f(\mathbf{p}_m^k)$  is the objective function of the  $m$ th particle in the  $k$ th iteration.

If  $p_m^k(d)$  is bounded, the supplementary information needs to be added, as follow:

$$p_m^k(d) = \begin{cases} p_m^{\min}(d) + h(p_m^{k-1}(d) - p_m^{\min}(d)) & \text{If } p_m^k(d) < p_m^{\min}(d) \\ p_m^{\max}(d) - h(p_m^{\max}(d) - p_m^{k-1}(d)) & \text{Else if } p_m^k(d) > p_m^{\max}(d) \\ p_m^k(d) & \text{Else} \end{cases} \quad (12)$$

where  $p_m^k(d)$  is the  $d$ th components of  $\mathbf{p}_m^k$ ;  $h > 0$  is the accommodation coefficient;  $p_m^{\min}(d)$  and  $p_m^{\max}(d)$  are the minimum value and maximum value of  $p_m^k(d)$ , respectively.

The CFO is an effective and efficient deterministic search and optimization algorithm. Its effect has been tested by many test functions in the Ref.<sup>26</sup>. The convergence rate of the CFO which is based on the parameters  $G$ ,  $\alpha$  and  $\beta$  is both the strength and weakness. The appropriate rate is very difficult to grasp. Hence, the CFO algorithm is augmented with an inner-loop local optimizer. The inner-loop local optimizer is to optimize each particle  $\mathbf{p}_m^k$  in its small field. The inner-loop local optimizer can improve the regional search ability which reduces the bad influence of the convergence rate. The local optimizer is more suitable to be solved by the full-blown optimization algorithm, so SQP and GA are applied to finish this optimizer comparatively. The inner-loop local optimizer acts in the 5% range of the independent variable  $\mathbf{x}$ . The whole optimization algorithm structure is shown in Fig. 6.

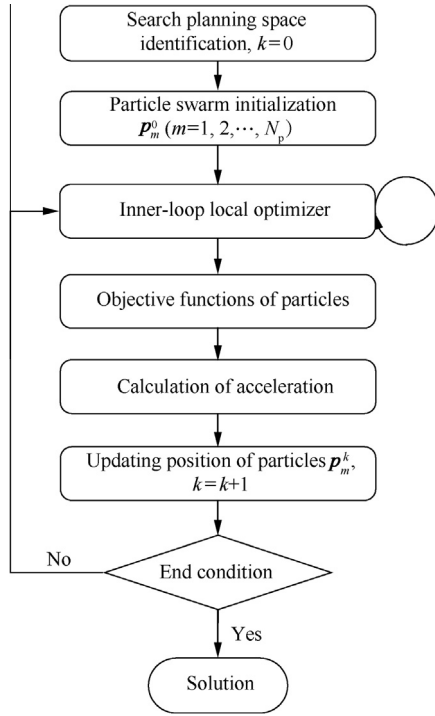


Fig. 6 Whole optimization algorithm structure.

### 3.4. Convergence analysis

The CFO algorithm is a novel optimization algorithm which needs more tests. There are only a few studies focusing on the convergence analysis of the CFO.<sup>27–30</sup> In this section, the new convergence analysis of the CFO is developed. At the same time, some convergence conditions are put forward herein. All particles using the CFO method converge to the optimum solutions or the special points. However, nothing will be said about whether these solutions represent local or global optimums. Without loss of generality, the CFO method is limited in one-dimensional situation and the selection of the particle  $m$  is arbitrary, so all related vectors  $\mathbf{p}_m^k$ ,  $\mathbf{a}_m^k$  and  $\mathbf{p}_n^k$  degenerate into the scalars  $p_m^k$ ,  $\tilde{a}_m^k$  and  $p_n^k$ .

**Definition 1 (Linear difference equations).** Let  $J^+$  be a set of the nonnegative integer, the square nonsingular matrix  $A(\cdot): J^+ \rightarrow \mathbf{R}^{n \times n}$ ,  $\mathbf{f}(\cdot): J^+ \rightarrow \mathbf{R}^n$  and  $\mathbf{x}^*(\rho) \in \mathbf{R}^n$ , for each  $\rho \in J^+$ , the linear difference equation (Eq. (13)) and corresponding homogeneous linear equation (Eq. (14)) with variable coefficients are as follows:

$$\mathbf{x}^*(\rho + 1) = A(\rho)\mathbf{x}^*(\rho) + \mathbf{f}^*(\rho) \quad (13)$$

$$\mathbf{x}^*(\rho + 1) = A(\rho)\mathbf{x}^*(\rho) \quad (14)$$

**Theorem 1.** *The movement equation of the particle  $p_m^k$  is a first-order linear difference equation with variable coefficients.*

**Proof.** According to the expression of  $U(\cdot)$ , define  $M_m = \{n | f(p_n^k) - f(p_m^k) > 0, n = 1, 2, \dots, N_p\}$ . The whole update of the CFO method is rewritten as

$$\begin{cases} p_m^{k+1} = p_m^k + \tilde{a}_m^k \\ \tilde{a}_m^k = \frac{1}{2} G \sum_{n \in M_m} \frac{(f(p_n^k) - f(p_m^k))^\alpha}{|p_n^k - p_m^k|^\beta} (p_n^k - p_m^k) \Delta t^2 \end{cases} \quad (15)$$

Define:

$$\begin{cases} \phi_m^k = G \sum_{n \in M_i} \frac{(f(p_n^k) - f(p_m^k))^\alpha}{|p_n^k - p_m^k|^\beta} p_n^k \\ \theta_m^k = G \sum_{n \in M_i} \frac{(f(p_n^k) - f(p_m^k))^\alpha}{|p_n^k - p_m^k|^\beta} \end{cases} \quad (16)$$

Eq. (15) is rewritten as

$$\begin{aligned} p_m^{k+1} &= p_m^k + \frac{1}{2} \Delta t^2 \phi_m^k - \frac{1}{2} \Delta t^2 \theta_m^k p_m^k \\ &= \left(1 - \frac{1}{2} \Delta t^2 \theta_m^k\right) p_m^k + \frac{1}{2} \Delta t^2 \phi_m^k \end{aligned} \quad (17)$$

Define:

$$\begin{cases} A_m(k) = 1 - \frac{1}{2} \Delta t^2 \theta_m^k \\ f_m(k) = \frac{1}{2} \Delta t^2 \phi_m^k \end{cases} \quad (18)$$

Go on with a variable substitution, and Eq. (17) is formed into:

$$p_m^{k+1} = A_m(k) p_m^k + f_m(k) \quad (19)$$

where  $A_m(k) \in \mathbf{R}$ ,  $f_m(k)$  is bounded. So Theorem 1 is proved.

After obtaining the differential form of the intelligent algorithm (like Eq. (19)), there are many references applying the linear system theory which is based on the characteristic equations of the differential equations to solve the convergence analysis problems.<sup>27–29</sup> Based on the linear system theory, the restriction condition to guarantee the stability of the system is that the spectral radius of the system matrix  $A_m(k)$  stays in the unit circle. This condition is the necessary and sufficient condition of the asymptotic stability for the discrete linear time-invariant system. But this conclusion is not suitable for the linear time-variant discrete system.<sup>31</sup> Xiao and Du<sup>32</sup> put forward two counter-examples respectively which show two situations. The first one shows that the movement is also unstable when the spectral radius of the system matrix  $A_m(k)$  stays in the unit circle for the linear time-variant discrete system. Then the second counter-example shows that the movement is still stable when the spectral radius of the system matrix  $A_m(k)$  stays outside the unit circle for the linear time-variant discrete system. Obviously, the differential forms of the intelligent algorithms are all time-variant. So the conclusions of above Refs.<sup>27–29</sup> are wrong and circumscribed by the assumption of the fixity coefficient.

**Theorem 2.**  $\forall \mathbf{f}^*(\rho) \in [J^+, \mathbf{R}^n]$ , the stability degree of the linear difference equations is equivalent to the stability degree of the general solution of the corresponding homogeneous linear equations.<sup>33</sup>

**Theorem 3.** *If there is a kind of matrix norm that satisfies:*

$$\|A(\rho)\|_x \leq \frac{\rho + 1}{\rho + 2} \quad \rho = \rho_0, \rho_0 + 1, \dots \quad (20)$$

where  $\|\cdot\|_x$  means a certain norm,  $\forall \rho_0 \in \mathbf{R}$ . Then the trivial solution of the Eq. (19) has the uniform asymptotic stability.<sup>33</sup> It is

only a sufficient condition for the uniform asymptotic stability of the solution.

According to Theorems 2 and 3, a sufficient condition which can ensure that the movement equation of the particles  $p_m^k$  based on the CFO method is stable (in other words, each particle  $p_m^k$  will converge to its own stable  $P_m$ ) is shown as follows:

$$|\lambda| = \left| 1 - \frac{1}{2} \Delta t^2 \theta_m^k \right| \leq \frac{k+1}{k+2} \quad k = k_0, k_0 + 1, \dots \quad (21)$$

where  $\lambda$  is the eigenvalue of  $A_m(k)$ ,  $\forall k_0 \in \mathbf{R}$ .

The condition is further simplified into:

$$\theta_m^k \in \left[ \frac{2}{(k+2)\Delta t^2}, \frac{4k+6}{(k+2)\Delta t^2} \right] \quad (22)$$

Based on the above sufficient condition, the particles  $\{p_m^k\}$  converge to the stable values  $\{P_m\}$ ,  $\lim_{k \rightarrow \infty} p_m^k = P_m$ . In the Ref.<sup>27</sup>, one of its conclusions is that all particles using the CFO method converge to a stable  $P$ . But we find that this conclusion is wrong because some special circumstances were not taken into consideration. Then, we need to prove our own conclusion.

**Theorem 4.** Under condition of Eq. (22),  $\{P_m, m = 1, 2, \dots, N_p\}$  will converge to the optimum solutions (local/global) or the special points.

**Proof.** Based on Eq. (15) and  $\lim_{k \rightarrow \infty} p_m^k = P_m$ , we have

$$\lim_{k \rightarrow \infty} \tilde{a}_m^k = \lim_{k \rightarrow \infty} (p_m^{k+1} - p_m^k) = P_m - P_m = 0 \quad (23)$$

Further, substitute Eq. (15) into Eq. (22), and we have

(1) Situation 1

For  $\forall m$  if it exists:  $P_n = P_m$ , you can get that all particles using the CFO method converge to a stable optimum solution  $P$ :

$$P_1 = P_2 = \dots = P_{M_m} = P \quad (24)$$

(2) Situation 2

If it exists:  $f(P_n) = f(P_m) \cap P_n \neq P_m$ , you can get that all particles using the CFO method converge to several stable values  $\{P_{m'}, m' = 1, 2, \dots, N'_p; N'_p \leq N_p\}$  which have the equal corresponding objective function values (the local/global optimum).

$$\begin{aligned} \{P_m, m = 1, 2, \dots, N_p\} \rightarrow \\ \{P_m, m' = 1, 2, \dots, N'_p; N'_p \leq N_p\} \end{aligned} \quad (25)$$

(3) Situation 3

If some  $m$  exists:  $f(P_n) \neq f(P_m) \cap P_n \neq P_m \cap \sum_{m \in M_m} \frac{(f(P_n) - f(P_m))^2}{|P_n - P_m|^\beta} \cdot (P_n - P_m) = 0$ , you can get that these particles using the CFO method converge to several special points  $\{P_{m''}, m'' = 1, 2, \dots, N''_p; N''_p \leq N_p\}$  which are not the local/global optimum. But their accelerations are zero. This situation only exists in rare circumstances. They are defined as the special points.

So Theorem 4 is proved.

As an example, these three situations are introduced further by Fig. 7 (3 particles).

The special points are only a smaller proportion of the definitional domain for most optimization problems. In the overwhelming majority of cases, the  $\{P_m, m = 1, 2, \dots, N_p\}$  will converge to the optimum solutions (local/global) under condition of Eq. (22). Even if the Situation 3 exists, there are at least two other stable  $P_m$  whose objective functions are better than the objective functions of the special points. The final results will rule out these special points in the sorting process.

In conclusion, the convergence of the CFO method can be guaranteed as long as  $\theta_m^k$  satisfies the Eq. (22).

### 3.5. Simulation results

In this part, the simulation results of the path planning problem are shown and analyzed. The CFO-GA algorithm and CFO-SQP algorithm are contrasted in the same simulation settings. The simulation settings are shown as follows.

The number of UAVs is  $M = 3$ ; the starting points of the UAVs are:  $S_1 = (1500, 2000, 500)$  m,  $S_2 = (2500, 1500, 500)$  m and  $S_3 = (2000, 1500, 500)$  m; the common target point  $T = [6000, 7000, 0]$  m; the number of the particles in CFO  $N_p = 20$ ; the maximum number of iterations in CFO is 300; the gravitation constant  $G = 0.01$ ; the tuning parameters  $\alpha = 0.4$  and  $\beta = 0.5$ ; the accommodation coefficient  $h = 0.5$ ; the number of the discrete waypoints  $N_w = 1300$ ; the minimum radius of the coaxial cylinders which is very important to the time-consumption of the following trajectory planning process will be discussed deeply in Section 5 and is provisionally defined as  $r = 1000$  m, so the total number of the cylinders  $N = 6$ ; the constraint value of the steering angle which is related to the turning radius of the UAVs  $r_1$  and the minimum radius of the coaxial cylinders  $r$  is provisionally defined as  $\delta_{\max} = \pi/2$  rad; the maximum number of iterations of the inner-loop local optimizer changes from 5 to 30; the other settings of the local optimizer are the defaults of the MATLAB R2011b.

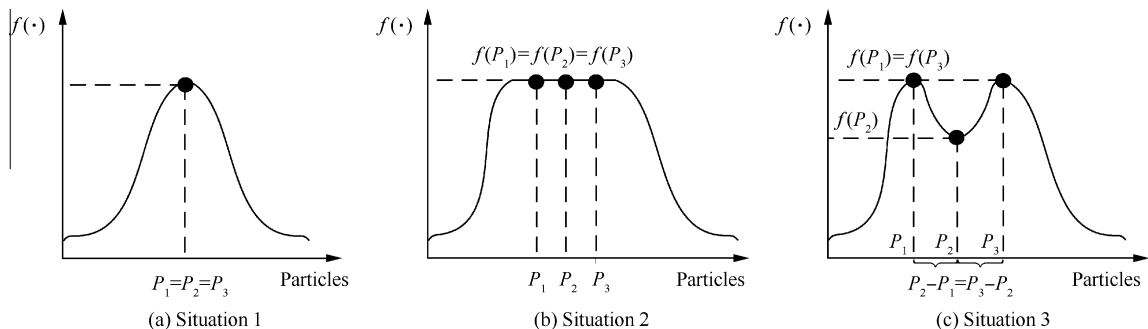


Fig. 7 Three possible convergence results of CFO algorithms.



In order to further verify the effectiveness of the proposed algorithms, the other state-of-the-art methods (GA and PSO) are introduced. The same simulation settings are shown as follows: The simulation parameters of the planning scene are all same as the parameters of the CFO-GA algorithm and CFO-SQP algorithm; the maximum number of iterations is 300; the population size in each generation is 20; the number of the discrete waypoints  $N_w = 1300$ ; the minimum radius of the coaxial cylinders  $r = 1000$  m, so the total number of the cylinders  $N = 6$ ; the constraint value of the steering angle is provisionally defined as  $\delta_{\max} = \pi/2$  rad. The different simulation settings are shown as follows: Because the GA is programmed based on the library function of the MATLAB R2011b, the settings of the GA are the defaults of the MATLAB R2011b. The learning factor, the acceleration coefficient and the inertia weight of the PSO algorithm respectively are 0.5, 0.5 and 0.8.<sup>34</sup>

The simulation results of the parallel CFO-GA method, the parallel CFO-SQP method, the parallel GA method, the parallel PSO method and the parallel CFO method depend on the same parameter settings. The effects of the maximum number of iterations and the maximum number of iterations of the inner-loop local optimizer will be presented in the latter part. So we only present the best situation of the path effect picture whose maximum number of iterations is 300 and the maximum number of iterations of the inner-loop local optimizer is 30 (See Fig. 8).

Fig. 8 shows the planning results of five algorithms. The dotted lines represent the planning results obtained by the parallel PSO algorithm. The star full lines represent the planning results obtained by the parallel GA algorithm. The dot dash lines represent the planning results obtained by the parallel CFO-GA algorithm. The imaginary lines represent the planning results obtained by the parallel CFO algorithm and the full lines without stars represent the planning results obtained by the parallel CFO-SQP algorithm. And the black lines, the purple lines and red lines represent the planning results whose starting points are  $S_1$ ,  $S_2$  and  $S_3$  respectively. They all keep away from the obstacle zones and reach the same target. It is observed from Fig. 8 that the parallel CFO-GA has selected a shorter and smoother path compared with the path found by the parallel GA, PSO, CFO-SQP and CFO. The path obtained by the parallel CFO-GA fly above the slit which is formed by the upper surface of the radar areas. Because this planning space is complex, the optimization algorithm is easy to fall into the local minimum value. The results of the other algorithms are limited in the other slit which presents a distinct local minimum. Therefore, we can see that the parallel CFO-GA algorithm has the stronger global convergence ability than the other algorithms. The objective functions changing with the iterations (1–300) are shown in Fig. 9.

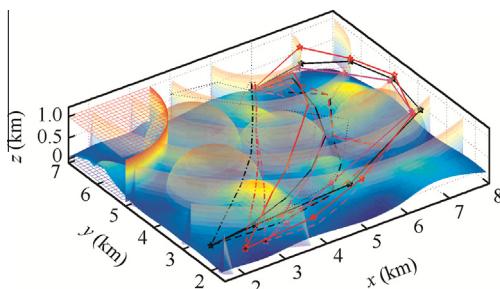


Fig. 8 Planning results of five algorithms.

From Fig. 9, we can see that the traditional GA and PSO algorithms have the poorer performance compared with three CFO-based algorithms. So in the following discussion, we only consider the parallel CFO-GA algorithm, the parallel CFO-SQP algorithm and the parallel CFO algorithm.

From the view of the fluctuation of the objective function values, the parallel CFO-GA algorithm is more suitable for the complicated path planning than the parallel CFO-SQP algorithm and the parallel CFO algorithm. At the initial segment (about 1–100 iterations), because of the unsatisfied constraints and penalty function method, some fluctuations appear in the objective function values. They respectively meet the constraints at the 107th (parallel CFO), the 90th (parallel CFO-SQP) and the 15th (parallel CFO-GA) iteration. Therefore, although the optimization range of the inner-loop local minimum problem is small, because of the complexity of the planning space, the local minimum problem also needs some algorithm like the GA whose global convergence is better than the traditional algorithm like the SQP.

From the view of the overall trend of the objective function values, the objective function values converge to their convergence values with the number of iterations increasing at last. At the initial segment (about 1–25 iterations), the convergence speeds of these three algorithms are all rapid. The main reason of this phenomenon is the strong convergence property of the outer loop optimization algorithm (the parallel CFO method). But because the convergence speed of the parallel CFO algorithm largely depends upon the differences between the particles, when the differences between the particles decrease (i. e.  $|f(\mathbf{p}_n^k) - f(\mathbf{p}_m^k)|^\alpha$  converges to zero), the convergence speed will greatly slow down. Just then the power of the inner-loop local optimizer is shown. Especially the local GA algorithm still propels the optimization after the initial segment.

In brief, the analyses of the convergence property of these three algorithms further provide that the parallel CFO-GA algorithm has a better performance for this problem than the parallel GA, PSO, CFO and CFO-SQP algorithm. After obtaining the above conclusion, we will further discuss the effect of the maximum number of the iterations of the inner-loop local optimizer in the convergence situation of the whole algorithm. The convergence values changing with the maximum number of the iterations (5–30) are shown in Fig. 10.

From Fig. 10, we can see that the convergence values decrease with the maximum number of the iterations of the inner-loop local optimizer. In the beginning, the decreasing velocity is slow. Then, the convergence values converge rapidly

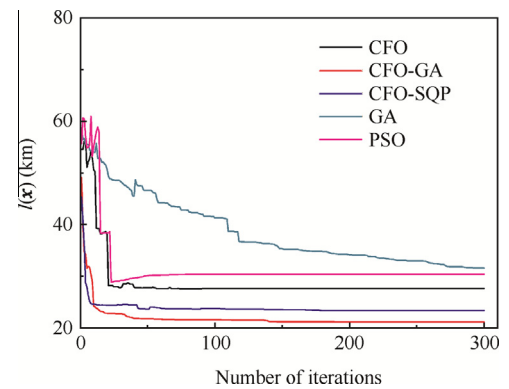
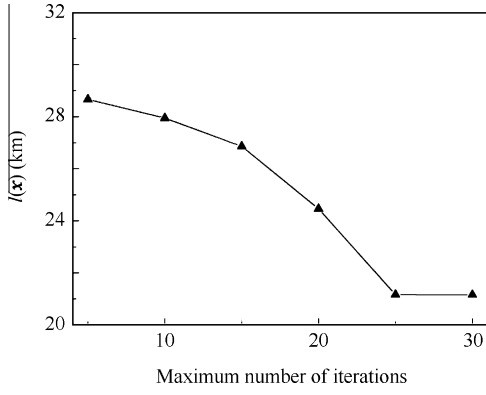


Fig. 9 Convergence property of five algorithms.



**Fig. 10** Convergence values changing with the maximum number of iteration.

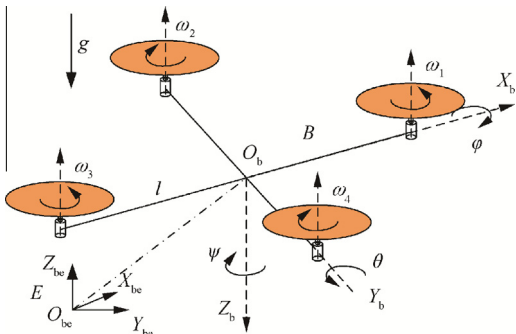
to the good solution of this problem after 25 iterations. This shows that the effective inner-loop local optimizer iterations for this problem need to be 25 at least. What is more, the bigger maximum number of the iterations of the inner-loop local optimizer will not improve the results of this problem and will increase the computing time of this problem after 25 iterations.

#### 4. Trajectory planning process

With the help of the hierarchic optimization strategy, the large-scale optimization problem is decomposed and simplified into many small trajectory planning problems. Then, the small trajectory planning problems, namely the optimal control problems, are converted into several nonlinear parameter optimization problems by the parameterization process. At last, they can be solved by the mature optimization algorithm quickly enough in the real situation during processing. In this section, the whole trajectory planning process including the flight simulator modeling, planning inputs definition and parameterization (independent variable definition), the optimization model creation, the solution algorithm and the simulation result is finished.

##### 4.1. Flight simulator model

In this work, the nonlinear 6-DOF dynamic equations of the quad-rotor UAVs are used to model the UAV motion. The quad-rotor helicopter UAV is an under-actuated, highly nonlinear and strongly coupled system with four fixed pitch angle rotors as shown in Fig. 11.<sup>35</sup>



**Fig. 11** Quad-rotor helicopter structural diagram.

The realistic and complicated model can better simulate the real dynamical system of the quad-rotor UAVs, but the complicated model increases the amount of calculation of the trajectory planning problem, which is not conducive to realizing the demand of real-time ability. So the appropriately simplified model of the UAV is as follows:<sup>36</sup>

$$\begin{cases} \ddot{x}_b = \frac{1}{m_u} (\cos \varphi \cos \psi \sin \theta + \sin \varphi \sin \psi) u_{c1} - \frac{K_x}{m_u} \dot{x}_b \\ \ddot{y}_b = \frac{1}{m_u} (\cos \varphi \sin \psi \sin \theta - \sin \varphi \cos \psi) u_{c1} - \frac{K_y}{m_u} \dot{y}_b \\ \ddot{z}_b = \frac{1}{m_u} \cos \theta \cos \varphi u_{c1} - \frac{K_z}{m_u} \dot{z}_b - g \\ \ddot{\varphi} = \frac{l}{I_x} u_{c2} + \dot{\theta} \dot{\psi} \left( \frac{I_y - I_z}{I_x} \right) \\ \ddot{\theta} = \frac{l}{I_y} u_{c3} + \dot{\varphi} \dot{\psi} \left( \frac{I_z - I_x}{I_y} \right) \\ \ddot{\psi} = \frac{l}{I_z} u_{c4} + \dot{\varphi} \dot{\theta} \left( \frac{I_x - I_y}{I_z} \right) \end{cases} \quad (26)$$

where  $\psi$ ,  $\theta$  and  $\varphi$  are the Euler angles between the body-axis coordinate system  $B$ :  $O_b X_b Y_b Z_b$  and the inertial coordinate system  $E$ :  $O_{be} X_{be} Y_{be} Z_{be}$ ;  $x_b$ ,  $y_b$  and  $z_b$  are the positions of the UAV;  $m_u$  is the mass of the UAV;  $g$  is the acceleration of gravity;  $l$  is the half length of the quad-rotor helicopter;  $K_x$ ,  $K_y$  and  $K_z$  are the drag coefficients;  $I_x$ ,  $I_y$  and  $I_z$  are the moments of inertia with respect to the axes  $X_b$ ,  $Y_b$  and  $Z_b$ , respectively; the control inputs  $u_{ci}$  ( $i = 1, 2, 3, 4$ ) satisfy the following equations:<sup>37</sup>

$$\begin{cases} u_{c1} = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ u_{c2} = b(\omega_2^2 - \omega_4^2) \\ u_{c3} = b(\omega_1^2 - \omega_3^2) \\ u_{c4} = c(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{cases} \quad (27)$$

where  $b$  is the scaling factor of the thrust generated by the rotor;  $c$  is the scaling factor of the propeller torque constant generated by the rotors;  $\omega_i$  ( $i = 1, 2, 3, 4$ ) is the rotational speed of the  $i$ th propeller.

Above models Eq. (26) are based on the following assumptions:<sup>38</sup>

- (1) The UAV structure is symmetrical and rigid.
- (2) The center of mass and  $O_b$  coincide.
- (3) Thrust and drag are proportional to the square of the propellers speed.
- (4) Ignore the rotational resistance of each UAV.
- (5) Ignore the non-determinacy of the UAV system.

##### 4.2. Planning inputs definition and parameterization

According to the direct method of the optimal control problem, the planning inputs need to be defined and parameterized. In Ref.<sup>16</sup>, the inputs of the trajectory planning process are directly the control inputs of the F-16 control system. Drawn from above, the control inputs  $u_{ci}$  ( $i = 1, 2, 3, 4$ ) may be chosen for the planning inputs of the quad-rotor UAV trajectory planning. If so, the  $u_{ci}$  ( $i = 1, 2, 3, 4$ ) can make up the independent variable of the new optimization problem.

But this assumption proved to be wrong in the simulation process. The quad-rotor helicopter is a strong unstable system

no matter from the viewpoint of the attitude inner-loop or from the viewpoint of the position outer-loop. It cannot fly stably without the effect of the control system. Obviously, the stability of the attitude inner-loop is the basis of the stability of the whole system. As a double-integral system, a few unreasonable inputs on the system will lead to the divergence and unreasonable results (large attitude angle). If the trajectory planning is finished without the attitude inner-loop control, most values of the independent variables in the definitional domain are infeasible. The percentage of success is too low to realize the real-time demand. Most of the planning time is wasted in searching the feasible solutions rather than searching the optimal solution. At the same time, the optimization effect will be very poor. The schematic diagram of two situations is shown in Fig. 12.

Therefore, the planning inputs need to be re-selected. By the help of the above analysis, the planning inputs need to ensure that the attitude parameters are in the proper range at least. The classical control system of the quad-rotor helicopter UAV is shown in Fig. 13, where  $x_p, y_p$  and  $z_p$  are the coordinates of the reference trajectory;  $\psi_r, \theta_r$  and  $\varphi_r$  are the reference attitude angles. In order to realize the stabilization of the attitude angles, the attitude inner-loop PD control is introduced in the planning process. So the planning inputs change into the control inputs  $\psi_r$  ( $-30^\circ \leq \psi_r \leq 30^\circ$ ),  $\theta_r$  ( $-30^\circ \leq \theta_r \leq 30^\circ$ ),  $\varphi_r$  ( $-30^\circ \leq \varphi_r \leq 30^\circ$ ) and  $u_{c1}$

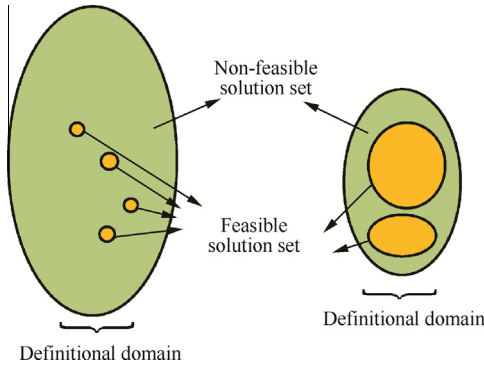


Fig. 12 Schematic diagram of two situations.

( $0.9m_0g \leq u_{c1} \leq 1.1m_0g$ ) of the attitude inner-loop controller. At the same time, the planning object also turns into the nonlinear 6-DOF dynamic equations of the quad-rotor UAVs with the attitude inner-loop PD control. The algorithm of the PD controller is shown as

$$u_{\text{follow}}(t) = K_P e(t) + K_D \frac{de(t)}{dt} \quad (28)$$

where  $u_{\text{follow}}(t)$  is the control variable; the tracking error  $e(t)$  is defined as  $e(t) = W_r(t) - W(t)$ , with  $W_r(t)$  the reference output and  $W(t)$  the real output;  $K_P$  and  $K_D$  are controller gains associated with proportional (P) and derivative (D) actions, respectively.

The control inputs are discretized at a number of equally spaced nodes (control points) between the initial time  $t_0$  and the indefinite final time  $t_f$  of the flying process. The values of the control points are limited in  $[LB, UB]$ , where LB and UB are the upper and lower bound of the control inputs. Then, these control points are parameterized by the cubic B-spline. But as a TPBV optimal control problem, the optimal final time  $t_f$  is very difficult to obtain. In order to obtain it, the security time  $T_f$  which is bigger than the indefinite final time  $t_f$  is presented. The discretized and parameterization processes are finished between the initial time  $t_0$  and the security time  $T_f$ . The security time  $T_f$  will obtain an integrated trajectory. The end waypoint whose corresponding time is  $t_f$  is defined as the closest waypoint to the target in the whole trajectory. So although the control inputs between  $t_0$  and  $T_f$  are used in the following optimization process, only the control inputs between  $t_0$  and  $t_f$  act on the objective function of the trajectory planning problem. In other words, the optimization only occurs between  $t_0$  and  $t_f$ . By this way, the optimal final time  $t_f$  and control inputs between  $t_0$  and  $t_f$  are obtained and used in the real flying state. The parameterization of control inputs is shown in Fig. 14.

From the above, the optimal control problem has been converted into the parameter optimization problem in this section. It's worth noting that the planning inputs have been changed into the control inputs  $\psi_r, \theta_r, \varphi_r$  and  $u_{c1}$  of the attitude inner-loop controller. Because  $\psi_r, \theta_r$  and  $\varphi_r$  are the reference attitudes of the UAV, they need to ensure its continuity. Therefore, the first control point of each small optimal control problem needs to be limited by the final value of the control input of the previous problem. Above all, the total number

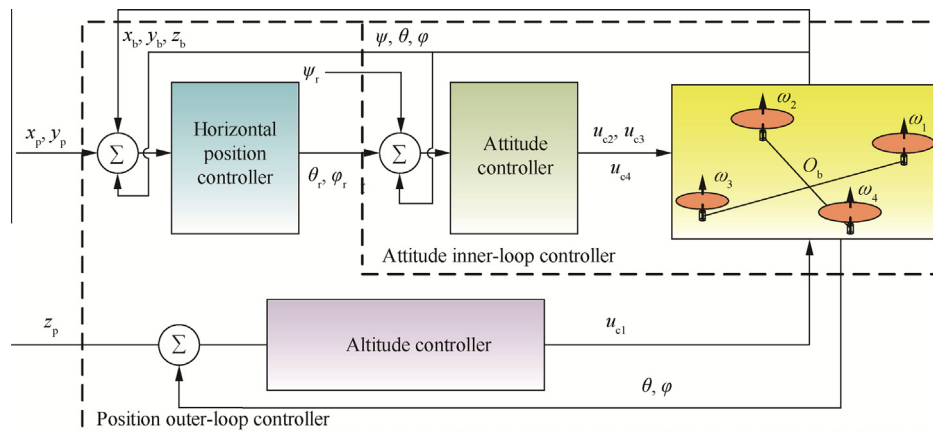


Fig. 13 Classical control structure for quad-rotor helicopter UAV.

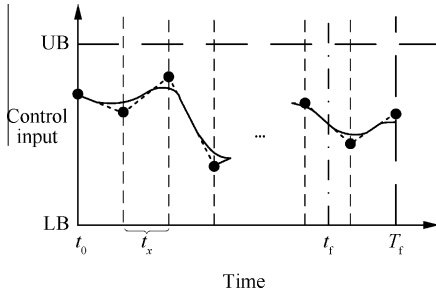


Fig. 14 Parameterization of control inputs.

of the independent variables of the trajectory planning problem except the first one is  $4(T_f - t_0)/t_x$ , with  $t_x$  the equal interval time. Moreover, the total number of the first small trajectory planning problem is  $4 + 4(T_f - t_0)/t_x$ . Because this process is solved on-line, the optimization efficiency which is greatly influenced by the scale of the optimization problem is very important. The value of  $t_x$  needs to be relatively big to reduce the number of the independent variables which can improve the optimization efficiency. At the same time, the value of  $t_x$  needs to be small enough to ensure the availability and optimality of the results. In conclusion, the new structure of the whole optimization problem is shown in Fig. 15.

#### 4.3. Optimization model creation and solution

The objective function and constraint conditions which are the basic elements of the optimization problem will be presented in this section. Then, the whole trajectory planning problem of each UAV in the fleet is established. In addition, the solving method is put forward.

Because of the hierarchic optimization strategy, the objective function of the single UAV trajectory planning problem is based on the waypoints  $\mathbf{o}_{t,j}$  ( $t = 1, 2, \dots, N_w$ ) of the offline path planning results. The number of the waypoints  $\mathbf{o}_{t,j}$  ( $t = 1, 2, \dots, N_w$ ) is too large, so we cannot build the small trajectory planning problem in every two waypoints. Therefore, the waypoints are divided into  $N_G$  groups. At the same time, the large-scale optimization problem is decomposed and simplified into  $N_G$  small trajectory planning problems. The last waypoint of the prior group is the first point of the later one. The first

waypoint and the last waypoint of each group respectively are the starting point and the target point of the small trajectory planning problem. The equation of the grouping is

$$G_{s,j} = \{ \mathbf{o}_{(s-1) \times \text{round}(N_w/N_G) + 1, j}, \mathbf{o}_{(s-1) \times \text{round}(N_w/N_G) + 2, j}, \dots, \mathbf{o}_{s \times \text{round}(N_w/N_G) + 1, j} \} \quad (29)$$

where  $s$  is the serial number of the group and the serial number of the small trajectory planning problem;  $\text{round}(\cdot)$  is the round down function;  $G_{s,j}$  is the  $s$ th group of the waypoints of the  $j$ th UAV.

The objective of this paper is to finish the multiple quad-rotor UAVs collaborative assembling problem. So the objective function of the trajectory planning problem also focuses on the minimized length and time difference. With the consideration of the path planning process, the length of the trajectory is guaranteed by the staged target points. So the length-optimal problem can be reduced into the problem whose goal is to minimize the distance between the end point and the staged target point. For the distance/time difference-optimal trajectory, based on the weighting method, the objective function of the  $j$ th UAV in the  $s$ th trajectory planning problem is defined as

$$J_{s,j} = \omega_{u1} \| (x_f^{s,j}, y_f^{s,j}, z_f^{s,j}) - \mathbf{o}_{s \times \text{round}(N_w/N_G) + 1, j} \| + \omega_{u2} |t_f^{s,j} - t_0^{s,j} - T_r^{s,j}| \quad (30)$$

where  $\omega_{u1}$  and  $\omega_{u2}$  are the adjustment coefficients, and  $\omega_{u1} > 0$ ,  $\omega_{u2} > 0$ ;  $(x_f^{s,j}, y_f^{s,j}, z_f^{s,j})$  is the end waypoint of the  $j$ th UAV in the  $s$ th small problem;  $t_0^{s,j}$  and  $t_f^{s,j}$  mean the initial time and final time of this problem, respectively;  $T_r^{s,j}$  is the predicted flight time of the  $j$ th UAV in the  $s$ th small problem whose purpose is to allocate the flight time among the different small trajectory planning problems and minimize the time difference. It is defined as

$$T_r^{s,j} = \begin{cases} \Phi(G_{s,j})/v & s = 1 \\ (t_f^{s-1,j} - t_0^{s-1,j} - T_r^{s-1,j}) + \Phi(G_{s,j})/v & s > 1 \end{cases} \quad (31)$$

where  $\Phi(G_{s,j})$  is the sum length of the path which is composed of line segments according to the initial and final waypoints of  $G_{s,j}$ .

The constraint conditions of the optimization problem include the dynamic constraints, the obstacle constraints and the initial/final conditions. The dynamic constraints have been

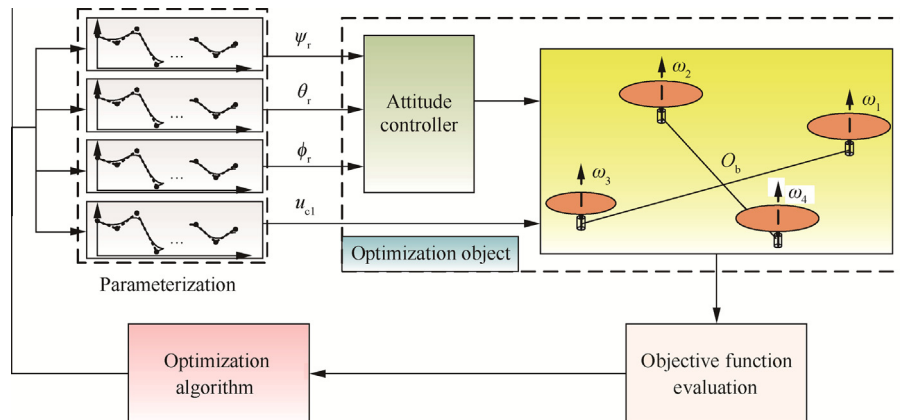


Fig. 15 New structure of whole optimization problem.



studied in the above section. It is manifested in the 6-DOF nonlinear differential dynamic equation of the quad-rotor UAV with the attitude inner-loop PD controller and the value range constraints of the planning inputs. Obviously, the UAV cannot fly through the obstacle, so the flight height meets  $z_b^{s,j} > z_r$ , where  $z_b^{s,j}$  and  $z_r$  respectively mean the UAV flight height and the obstacle height of the  $j$ th UAV in the  $s$ th trajectory planning problem. The initial/final conditions of the  $j$ th UAV in the  $s$ th trajectory planning problem are defined as follows:

$$\begin{cases} \mathbf{X}_0^{s,j} = \begin{cases} [\boldsymbol{o}_{1,j}, \mathbf{0}_{1 \times 9}]^T & s = 1 \\ \mathbf{X}_r^{s-1,j} & s > 1 \end{cases} \\ (\mathbf{X}_r^{N_G,j}, \mathbf{y}_r^{N_G,j}, \mathbf{z}_r^{N_G,j}) \in \mathcal{O}_{r=100}(\boldsymbol{o}_{N_w,j}) \end{cases} \quad (32)$$

where  $\mathbf{X}_0^{s,j}$  and  $\mathbf{X}_r^{s,j}$  are the initial and terminal state vectors of the 6-DOF differential dynamic equation; the state vectors can then be formed as  $[x_b^{s,j}, y_b^{s,j}, z_b^{s,j}, \dot{x}_b^{s,j}, \dot{y}_b^{s,j}, \dot{z}_b^{s,j}, \varphi_{s,j}, \theta_{s,j}, \psi_{s,j}, \dot{\varphi}_{s,j}, \dot{\theta}_{s,j}, \dot{\psi}_{s,j}]^T$ ; they represent the flight state of the  $j$ th UAV in the  $s$ th trajectory planning problem which includes the position, the speed, the Euler angle and the Euler angular velocity;  $\mathcal{O}_{r=100}(\boldsymbol{o})$  is a sphere whose center and radius respectively are  $\boldsymbol{o}$  and 100 m.

In conclusion, the  $j$ th UAV in the  $s$ th optimization problem is defined as

$$\begin{aligned} & \min_{\psi_r, \theta_r, \varphi_r, u_{ci}} J_{s,j} \\ & \text{subject to } \begin{cases} \text{Differential Eqs. (26)–(28)} \\ z_b^{s,j} > z_r \\ \text{Initial/final conditions in Eq. (32)} \end{cases} \end{aligned} \quad (33)$$

This trajectory problem needs to be optimized on-line in the airborne computer of the UAV, so the mature, simple and efficient method must be used in this process. The SQP method which is the most widely-used algorithm is used to solve this problem. In this part, the whole trajectory planning problem is finished by the Microsoft Visual C++ V 6.0. The maximum number of allowed iterations is 15, and all other optimization settings imitate the default parameters of the MATLAB R2011b. At the same time, the initial values of the control inputs of  $\theta_r$ ,  $\varphi_r$  and  $u_{ci}$  respectively are  $[0, 0, \dots, 0]_{1 \times (T_f - t_0)/t_x}$ ,  $[0, 0, \dots, 0]_{1 \times (T_f - t_0)/t_x}$  and  $[m_{uG}, m_{uG}, \dots, m_{uG}]_{1 \times (T_f - t_0)/t_x}$ .

#### 4.4. Simulation results

In this section, the simulation results of the same trajectory planning problem are shown and analyzed. The simulation parameters are shown as follows: the mass of the UAVs is  $m_u = 0.504$  kg; the moments of inertia  $I_x$ ,  $I_y$  and  $I_z$  respectively are  $1.9757 \times 10^{-4}$  kg·m<sup>2</sup>,  $1.9757 \times 10^{-4}$  kg·m<sup>2</sup> and  $3.9514 \times 10^{-4}$  kg·m<sup>2</sup>; the half length of the quad-rotor helicopter UAV is  $l = 0.028$  m; the group number is  $N_G = 10$ ; the interval between the initial  $t_0$  and the security time  $T_f$  is 90 s; the value of the equal interval time is  $t_x = 10$  s; the rated speed of the rotor-UAVs is  $v = 9$  m/s. The simulation results are shown as follows.

Fig. 16 shows the trajectory planning results of the collaborative assembling task. The multi-UAVs take off from different starting points (black line: UAV<sub>1</sub> (S<sub>1</sub>), purple line: UAV<sub>2</sub> (S<sub>2</sub>) and red line: UAV<sub>3</sub> (S<sub>3</sub>)), avoid the obstacles and

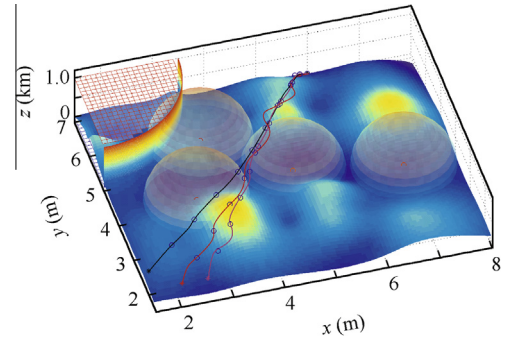


Fig. 16 Trajectory planning results of collaborative assembling task.

finish the collaborative assembling task successfully. The distances between the terminal positions and the target respectively are 13.9 m, 7.4 m and 8.6 m.

It is observed that the trajectories are not as straight as the paths in the path planning process. The main reason of this phenomenon is that the small trajectory planning problems are short of the terminal attitude constraints and the terminal flight state of the previous small problem will impact the optimization process of the next small problem. Certainly, this is the defect of this algorithm, and it forms the objectives of the future research and works. Then the flight data which include the optimal inputs, the flight velocities in three directions and the attitude angles versus time are shown in Figs. 17 and 18.

The above results (see Figs. 17 and 18) show that the variation ranges of the optimal inputs, the flight velocities and the attitude angles all satisfy the constraint conditions. At the same time, because the initial values of the control inputs are chosen as the zero vectors or the row vector whose values are all  $m_{uG}$ , the values of the optimal control inputs keep in the small values. This phenomenon means that the UAVs are in the stable flight state. The total flight time of these three UAVs respectively is 819.76 s (UAV<sub>1</sub>), 761.52 s (UAV<sub>2</sub>) and 832.69 s (UAV<sub>3</sub>). Because the rated speed of the rotor-UAVs is 9 m/s and the sum lengths of the whole paths respectively are 7232 m, 6835 m and 7228 m, the predicted flight time respectively is 803.6 s (UAV<sub>1</sub>), 759.4 s (UAV<sub>2</sub>) and 803.1 s (UAV<sub>3</sub>). It follows that the algorithm ensures the rated flight time to some extent. So in order to ensure that the multi-rotor-UAVs can reach the target at the same time, the takeoff time of UAV<sub>1</sub> and UAV<sub>2</sub> needs to be postponed for 12.93 s and 71.17 s, respectively.

In conclusion, these results prove that the hierarchic optimization strategy algorithm is effective for the multi-UAVs trajectory optimization problem without the consideration of the real-time ability. In this section, we also do not discuss the real-time requirement of the trajectory planning process and we will discuss it in the following section.

## 5. Discussion and analysis of real-time performance of hierarchic optimization strategy

The offline path planning process is computed in ground station before the takeoff of the UAVs. Then, the results will be transferred and loaded into the airborne memory device

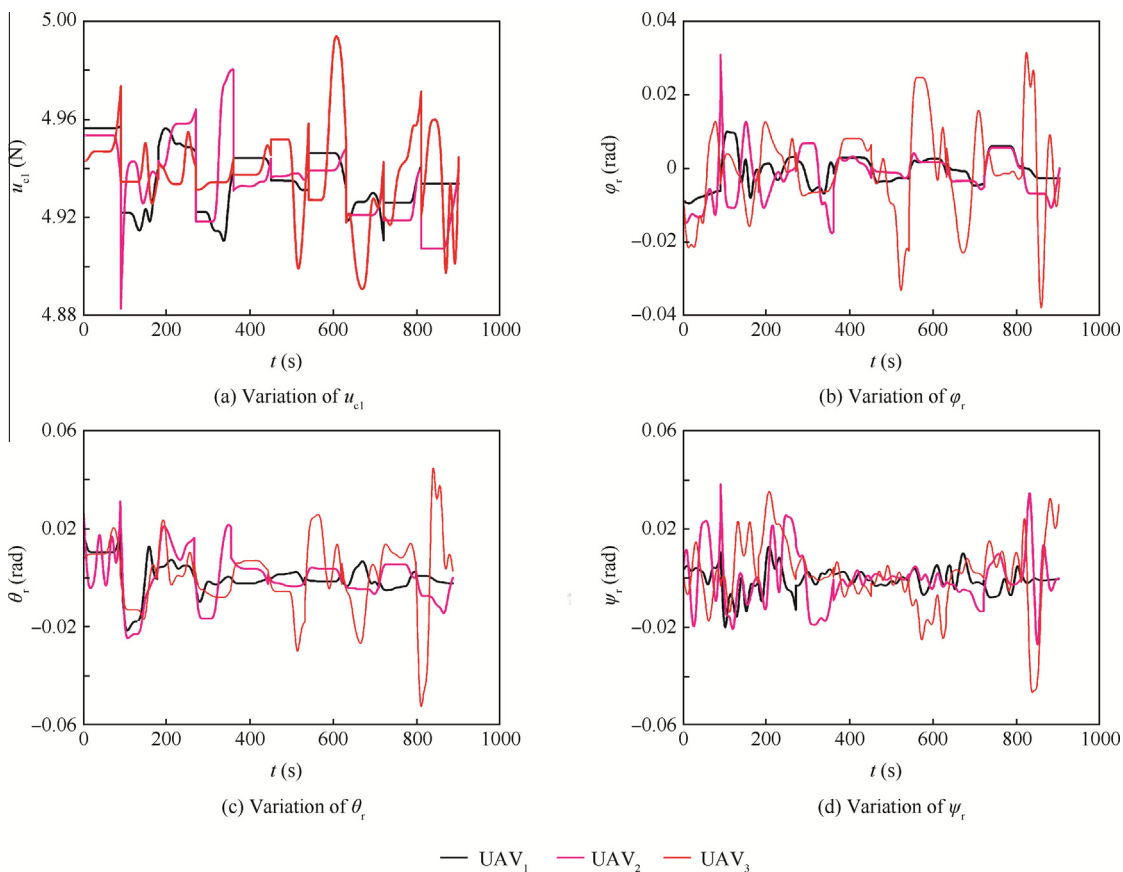


Fig. 17 Variations of optimal inputs.

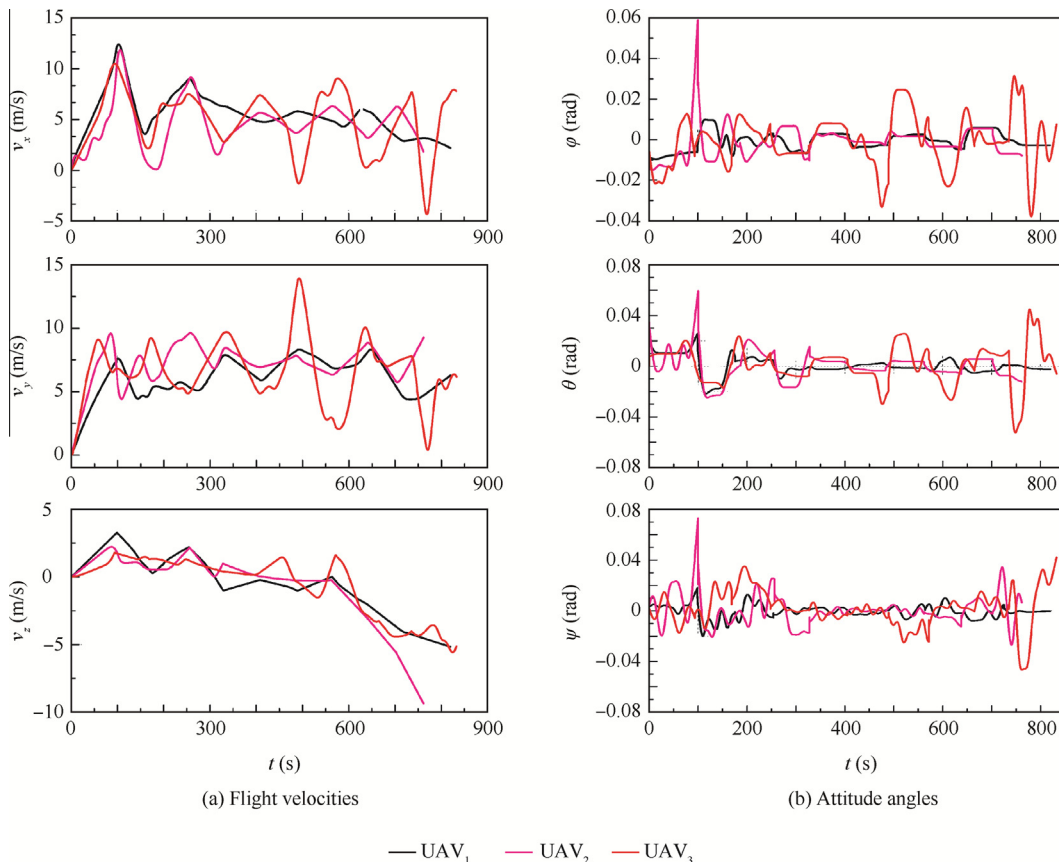


Fig. 18 Variations of flight velocities and attitude angles.

which can be called the airborne computers. After the storage of the offline data, the UAVs are ready to hop off. The optimal time-varying flying states which are the inputs of the real UAV control system are obtained by the online trajectory planning process in the airborne computers. In the real flying process, the UAVs need to fly steadily and plan quickly enough. So the airborne computers need to simultaneously calculate the guidance law and control command. The graphical illustration of the online trajectory planning process is shown in Fig. 19.

As you can see, the calculation and implementation of the UAV system are finished in several time units. In every unit, the airborne computer needs to complete the operations of the data loading, the trajectory planning resolving and the data post-processing in due order. Accordingly, the fixed length of the time unit  $T_{\text{step}}$  includes the constant loading time  $\Delta t_{\text{loading}}$ , the planning time  $\Delta t_{\text{planning}}$ , and the constant post-processing time  $\Delta t_{\text{post-processing}}$  ( $T_{\text{step}} = \Delta t_{\text{loading}} + \Delta t_{\text{planning}} + \Delta t_{\text{post-processing}}$ ,  $\Delta t_{\text{loading}} + \Delta t_{\text{post-processing}} = 5\text{ s}$ ). Of course, for a UAV in one task, the time units  $T_{\text{step}}$  of the different small trajectory planning problems are different, so we define that the value of the time unit  $T_{\text{step}}(s, j)$  means the time unit of the  $j$ th UAV in the  $s$ th trajectory planning problem. Before takeoff, the UAVs wait for at least a time unit  $T_{\text{step}}(1, j)$ . During this period of time, the airborne computer needs to accomplish the first small trajectory planning process which is from the starting point  $S_j$  to the later point  $(\theta_{s \times \text{round}(N_w/N_G)+1, j})$ . Then, the UAVs receive the planning results and begin to follow the trajectory until they reach the later points  $(\theta_{N_G \times \text{round}(N_w/N_G)+1, j})$  at time intervals  $T_{\text{flying}}(1, j)$ , with  $T_{\text{flying}}(1, j)$  the optimal final time  $t_f$  of the  $j$ th UAV in the first trajectory planning problem. By this way, the planning results need to be obtained before the implementation for the same flight process. So the feasible condition which can ensure the real-time ability is

$$T_{\text{step}}(s+1, j) \leq T_{\text{flying}}(s, j) \quad s = 1, 2, \dots, N_G - 1 \quad (34)$$

Therefore, in order to meet the above condition, the coordinated relations between the planning time  $\Delta t_{\text{planning}}$  and the implementation time  $T_{\text{flying}}$  are studied in the following part. The influence factors of the planning time  $\Delta t_{\text{planning}}$  mainly include the equal interval time of the control inputs  $t_x$  and the group number of the waypoints  $N_G$ . However, the main influence factor of the implementation time  $T_{\text{flying}}$  is the group number of the waypoints  $N_G$ . So in the following part, we will discuss the impact degree of these factors by some simulations.

Because the planning time  $\Delta t_{\text{planning}}$  not only depends upon the complexity level of the optimization problem, but also depends upon the performance of the simulation computer,

all these trajectory planning problems are solved by the following computer systems shown in Table 1.

The first influence factor is the group number of the waypoints  $N_G$  which directly affects the distance between each two starting waypoints in the adjacent groups, then further affects the implementation time  $T_{\text{flying}}$  and the planning time  $\Delta t_{\text{planning}}$ . In order to separately study the impact of  $N_G$ , we keep the value of the equal interval time  $t_x$  ( $t_x = (T_f - t_0)/9$ ) fixed. Accordingly, the discretized and parameterization time between the initial time  $t_0$  and the security time  $T_f$  is defined as  $900/N_G$ . For a collaborative assembling task, the distances between the multi-UAVs terminal positions and the target are the evaluation criteria to judge whether the task is achieved. In this part, the success sign of the simulation is defined as that the distances between the multi-UAVs terminal positions and the target are less than 100 m. When  $N_G$  is small, the small maximum number of iterations cannot ensure that the distances are small enough. So in order to achieve the collaborative assembling task, the maximum number of iterations is unlimited when  $N_G$  changes from 1 to 9. Even so, because of the complexity of this problem, the completion status of the simulations is also very bad for some situations whose  $N_G$  is too small. Then, the real-time performance of the simulation is shown in Table 2 and Fig. 20. Where average  $\Delta t_{\text{planning}}$  means the average value of the planning time in all small trajectory planning problems; average  $T_{\text{flying}}$  means the average value of the flying time in all small trajectory planning results; completion status means whether the simulations reach the success sign or not;  $\infty$  means there is no successful case; real-time ability means all simulation results satisfy Eq. (22).

In Fig. 20, we can see that the average time unit  $T_{\text{step}}$  and average flying time  $T_{\text{flying}}$  are both getting smaller with the increase of  $N_G$ . When  $N_G$  is too small, the SQP algorithm falls into local minima and cannot obtain a successful simulation result. At the same time, we can realize that the gradients of them are different and these two curves intersect. The intersection of these two curves is very important to judge the real-time performance of the hierarchic optimization strategy. When  $N_G$  is bigger than 6, the real-time performance of the algorithm basically meets the demand of the online system. Or else it cannot meet the online demand definitely. In Table 2, a special phenomenon needs to be paid attention to. When  $N_G$  is 6, the real-time demand still is not satisfied although the average time unit is smaller than the average flying time. The reason of this phenomenon is that, for different UAVs and different small trajectory planning problem, the time unit  $T_{\text{step}}(s, j)$  and the flying time  $T_{\text{flying}}(s, j)$  are different and around the average value.

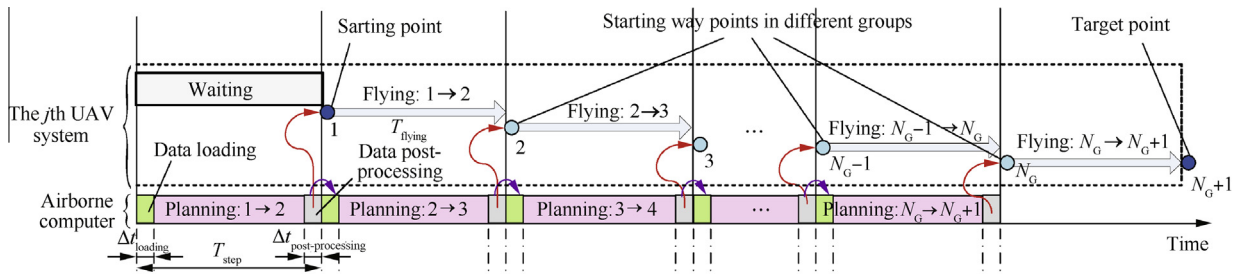


Fig. 19 Graphical illustration of online trajectory planning process.

**Table 1** Computer system used for experimental test.

Computer	CPU	RAM	OS	IDE
ThinkStation S20	Intel(R) Xeon(R) E5620	4.00 GB DDR3 1600 MHz	Windows 7 professional	Microsoft Visual C++
4105-D22	CPU @ 2.40 GHz		64 bits	V6.0

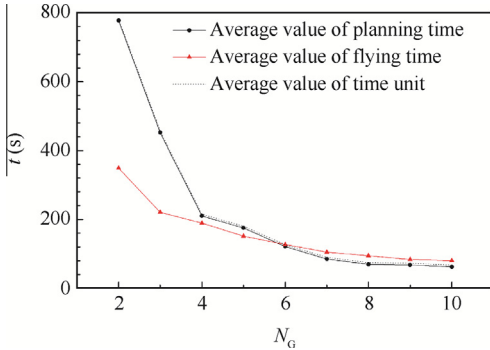
The second influence factor is the equal interval time  $t_x$  which directly affects the independent variables of the optimization problem, thus affects the planning time  $\Delta t_{\text{planning}}$ . In order to study the influence of  $t_x$  separately, the group number of the waypoints  $N_G$  and the maximum number of allowed iterations are defined as 10 and 15 respectively. Then, the value of the control inputs  $t_x$  changes from  $(T_f - t_0)/9$  to  $(T_f - t_0)/2$ . The real-time performance of the simulation is shown in Table 3.

After the corrections of the constant loading time  $\Delta t_{\text{loading}}$  and the constant post-processing time  $\Delta t_{\text{post-processing}}$ , the curves of the planning time  $\Delta t_{\text{planning}}$  and the time unit  $T_{\text{step}}$  with the changes of  $t_x$  are shown in Fig. 21.

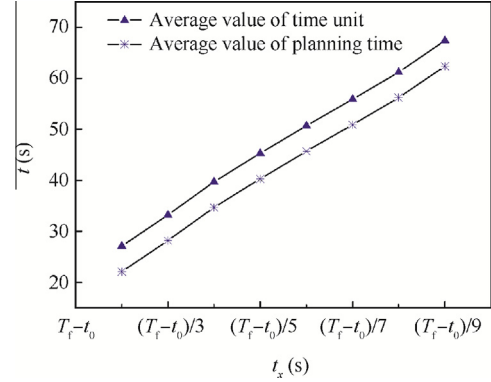
With the decrease of  $t_x$ , the dimensions of the independent variables become larger and larger. Of course, the planning time and time unit also become longer and longer. For this problem, the relationship between the planning time and  $t_x$  is in the linearity range (from  $(T_f - t_0)/2$  to  $(T_f - t_0)/9$ ). Clearly, all of these results can meet the real-time requirement. But the

**Table 2** Real-time performance changes with  $N_G$ .

$N_G$	Average $\Delta t_{\text{planning}}$ (s)	Average $T_{\text{flying}}$ (s)	Completion status	Real-time ability
1	$\infty$	$\infty$	Failure	No
2	777.3	349.2	Success	No
3	452.1	220.5	Success	No
4	210.1	189.4	Success	No
5	175.2	151.1	Success	No
6	121.3	127	Success	No
7	84.9	105.2	Success	Yes
8	69.4	94.6	Success	Yes
9	67.7	83.6	Success	Yes
10	62.4	80.5	Success	Yes

**Fig. 20** Variation curves of average planning time, average flying time and average time unit.**Table 3** Real-time performance changes with  $t_x$ .

$t_x$	Average $\Delta t_{\text{planning}}$ (s)
$(T_f - t_0)/2$	22.1
$(T_f - t_0)/3$	28.2
$(T_f - t_0)/4$	34.7
$(T_f - t_0)/5$	40.3
$(T_f - t_0)/6$	45.7
$(T_f - t_0)/7$	50.9
$(T_f - t_0)/8$	56.2
$(T_f - t_0)/9$	62.4

**Fig. 21** Changing curves of time unit and planning time.

larger dimensions of the independent variables can expand the resolution space, which is favorable to solve the more complex planning space. So  $t_x$  is set to 10 s in Section 4.

In short, the real-time demand of this system can be guaranteed for this hardware system when the group number of the waypoints  $N_G$  is bigger than 6 and the value of the control inputs  $t_x$  is bigger than  $(T_f - t_0)/9$ . For different computer hardware configurations, the real-time conditions need to be adjusted by the adjustment of the group number of the waypoints  $N_G$  and the value of the control inputs  $t_x$ .

## 6. Conclusions

- (1) A hierarchic optimization strategy which is composed of the offline path planning process and online trajectory planning process is used to solve the multiple quad-rotor UAVs collaborative assembling task. By the help of the hierarchic optimization strategy, the trajectory optimization problem is divided into a path planning problem and several small trajectory planning problems.
- (2) In the path planning process, a novel parallel intelligent optimization algorithm which combines the CFO algorithm with the GA algorithm is presented. Moreover, the sufficient conditions of the CFO algorithm are obtained by the stability theory of linear time-variant discrete system, under which it is guaranteed to converge to the global/local optimal value or the special points under a predefined initial distribution.
- (3) In the trajectory planning process, the trajectory planning optimization problem is established based on a high-fidelity and six-degrees-of-freedom nonlinear



dynamic model of the quad-rotor UAVs and the planning results of the front path planning process. In order to limit the range of the attitude angle and guarantee the flight stability of the planning result, the optimized object is changed from the ordinary quad-rotor UAVs dynamic model to the 6-DOF rigid-body dynamic model of the quad-rotor helicopters with an inner-loop attitude controller. What's more, the cubic B-spline parameterization algorithm and the concept of the security time are introduced to compute the optimal control inputs and the optimal time of this optimal control problem. All the three UAVs fly over the regions and reach the target at the same time.

- (4) Finally, the discussion and analysis on the real-time performance of the hierarchic optimization strategy are presented around the group number of the waypoints and the equal interval time.

## References

- Peng H, Su F, Bu YL, Zhang GZ, Shen LC. Cooperative area search for multiple UAVs based on RRT and decentralized receding horizon optimization. *Proceedings of 2009 the 7th Asian control conference*; 2009 Aug 27–29; HongKong, China. Piscataway (NJ): IEEE Press; 2009. p. 298–303.
- Shen LC, Xu X, Zhu HY. *Autonomous mobile robot control theory and technology*. Beijing: Science Press; 2011. p. 151–68 [Chinese].
- Garcia M, Viguria A, Ollero A. Dynamic graph-search algorithm for global path planning in presence of hazardous weather. *J Intell Rob Syst* 2013;**69**(1–4):285–95.
- Khatib O. Real-time obstacle avoidance for manipulators and mobile robots. *Int J Rob Res* 1986;**5**(1):90–8.
- Cetin O, Zagli I, Yilmaz G. Establishing obstacle and collision free communication relay for UAVs with artificial potential fields. *J Intell Rob Syst* 2013;**69**(1–4):361–72.
- Dunbar WB, Caveney DS. Distributed receding horizon control of vehicle platoons: stability and string stability. *IEEE Trans Autom Control* 2012;**57**(3):620–33.
- Ichnowski J, Alterovitz R. Scalable multicore motion planning using lock-free concurrency. *IEEE Trans Rob* 2014;**30**(5):1123–36.
- Li P, Duan HB. Path planning of unmanned aerial vehicle based on improved gravitational search algorithm. *Sci China Technol Sci* 2012;**55**(10):2712–9.
- Roberge V, Tarbouchi M, Labonté G. Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. *IEEE Trans Ind Inf* 2013;**9**(1):132–41.
- Duan HB, Zhang XY, Wu J, Ma GJ. Max–min adaptive ant colony optimization approach to multi-UAVs coordinated trajectory replanning in dynamic and uncertain environments. *J Bionic Eng* 2009;**6**(2):161–73.
- Zhu WR, Duan HB. Chaotic predator–prey biogeography-based optimization approach for UCAV path planning. *Aerosp Sci Technol* 2014;**32**(1):153–61.
- Xu CF, Duan HB, Liu F. Chaotic artificial bee colony approach to uninhabited combat air vehicle (UCAV) path planning. *Aerosp Sci Technol* 2010;**14**(8):535–41.
- Yao M, Zhao M. Unmanned aerial vehicle dynamic path planning in an uncertain environment. *Robotica* 2015;**33**(3):611–21.
- Heydari A, Balakrishnan S. Path planning using a novel finite horizon suboptimal controller. *J Guid Control Dyn* 2013;**36**(4):1210–4.
- Zhang Y, Chen J, Shen LC. Real-time trajectory planning for UCAV air-to-surface attack using inverse dynamics optimization method and receding horizon control. *Chin J Aeronaut* 2013;**26**(4):1038–56.
- Kamyar R, Taheri E. Aircraft optimal terrain/threat-based trajectory planning and control. *J Guid Control Dyn* 2014;**37**(2):466–83.
- Chamseddine A, Zhang Y, Rabbath CA. Trajectory planning and re-planning for fault tolerant formation flight control of quadrotor unmanned aerial vehicles. *Proceedings of the 2012 American control conference (ACC)*; 2012 Jun 27–29; Montreal (QC). Piscataway (NJ): IEEE Press; 2012. p. 3291–6.
- Eva BP, Lius T, Cruz JM, Bonifacio AT. Evolutionary trajectory planner for multiple UAVs in realistic scenarios. *IEEE Trans Rob* 2010;**26**(4):619–34.
- Yang K, Keng GS, Sukkarieh S. A Gaussian process-based RRT planner for the exploration of an unknown and cluttered environment with a UAV. *Adv Rob* 2013;**27**(6):431–43.
- Gu XQ, Shen LC, Chen J, Zhang Y, Zhang WP. A virtual motion camouflage approach for cooperative trajectory planning of multiple UCAVs. *Math Prob Eng* 2014;**2014**:1–15.
- Luo GC, Yu JQ, Mei YS, Zhang SY. UAV path planning in mixed-obstacle environment via artificial potential field method improved by additional control force. *Asian J Control* 2014;**17**(4):1–11.
- Zhang XY, Duan HB, Yu YX. Receding horizon control for multi-UAVs close formation control based on differential evolution. *Sci China Inf Sci* 2010;**53**(2):223–35.
- Chandler P, Rasmussen S, Pachter M. UAV cooperative path planning. *Proceedings of AIAA guidance, navigation, and control conference and exhibit*; 2000 Aug 14–17; Denver (CO). Reston: AIAA; 2000. p. 1255–65.
- Alighanbari M, Kuwata Y, How JP. Coordination and control of multiple UAVs with timing constraints and loitering. *Proceedings of the 2003 American control conference*; 2003 Jun 4–6. Piscataway (NJ): IEEE Press; 2003. p. 5311–6.
- Beard RW, McLain TW, Goodrich MA, Anderson EP. Coordinated target assignment and intercept for unmanned air vehicles. *IEEE Trans Rob Autom* 2002;**18**(6):911–22.
- Formato RA. Central force optimization with variable initial probes and adaptive decision space. *Appl Math Comput* 2011;**217**(21):8866–72.
- Ding DS, Qi DL, Luo XP, Chen JF, Wang XJ, Du PY. Convergence analysis and performance of an extended central force optimization algorithm. *Appl Math Comput* 2012;**219**(4):2246–59.
- van den BF, Engelbrecht AP. A study of particle swarm optimization particle trajectories. *Inf Sci* 2006;**176**(8):937–71.
- Emara HM, Fattah HAA. Continuous swarm optimization technique with stability analysis. *Proceedings of the 2004 American control conference*; 2004 Jun 30–Jul 2; Boston (MA). Piscataway (NJ): IEEE Press; 2004. p. 2811–7.
- Chen YB, Yu JQ, Mei YS, Wang YF, Su XL. Modified central force optimization (MCFO) algorithm for 3D UAV path planning. *Neurocomputing* 2016;**171**(1):878–88.
- Jin XL, Ma LH, Wu TJ, Qian JX. Convergence analysis of the particle swarm optimization based on stochastic processes. *Acta Autom Sin* 2007;**33**(12):1263–8 [Chinese].
- Xiao Y, Du XY. Theorem and algorithm of asymptotic stability test for time variant discrete system. *J Northern Jiaotong Univ* 1998;**22**(6):1–7 [Chinese].
- Liao WD, Liu ZX, Liao XX. Stability conditions based on Cauchy-matrix for some classes of time-varying linear difference equations. *J Huazhong Normal Univ (Nat Sci)* 2001;**35**(4):386–9 [Chinese].
- Wang Q, Zhang A, Qi LH. Three-dimensional path planning for UAV based on improved PSO algorithm. *Proceedings of the 26th Chinese control and decision conference (CCDC)*; 2014 May 31–Jun 2; Changsha, China. Piscataway (NJ): IEEE Press; 2014. p. 3981–5 [Chinese].
- Mohamed HA, Yang S, Moghavvemi M. Sliding mode controller design for a flying quadrotor with simplified action planner.

- Proceedings of 2009 ICCAS-SICE*; 2009 Aug 18–21; Fukuoka, Japan. Piscataway (NJ): IEEE Press; 2009. p. 1279–83.
36. Bouabdallah S, Siegwart R. Full control of a quadrotor. *Proceedings of IEEE/RSJ international conference on intelligent robots and systems*; 2007 Oct 29–Nov 2; San Diego (CA). Piscataway (NJ): IEEE Press; 2007. p. 153–8.
37. Bai YQ, Liu H, Shi ZY, Zhong YS. Robust flight control of quadrotor unmanned air vehicles. *Robot* 2012;34(5):519–24 [Chinese].
38. Salih AL, Moghavvemi M, Mohamed HA, Gaeid KS. Modelling and PID controller design for a quadrotor unmanned air vehicle. *Proceedings of 2010 IEEE international conference on automation quality and testing robotics (AQTR)*; 2010 May 28–30; Cluj-Napoca. Piscataway (NJ): IEEE Press; 2010. p. 1–5.

**Chen Yongbo** received his B.S. degree in Beijing Institute of Technology in 2012. He is currently working toward a doctoral degree at School of Aerospace Engineering, Beijing Institute of Technology. His research interest is UAV path planning.

**Yu Jianqiao** received B.S., M.S. and Ph.D. degrees from Beijing Institute of Technology in 1994, 1997 and 2007, respectively, and now is a professor there. His main research interests include flight dynamics and control, cooperative control, flight vehicle system design and robust control.

**Mei Yuesong** received his Ph.D. degree in flight vehicle design from Beijing Institute of Technology in 2007. He is now a lecturer in School of Aerospace Engineering, Beijing Institute of Technology. His research interests include flight vehicle system design, flight dynamics and control.

**Zhang Siyu** received his B.S. degree in Mechanical Design Manufacturing and Automation from Beijing University of Technology in 2010. He is currently working toward a Ph.D. degree at School of Aerospace Engineering, Beijing Institute of Technology. His research interests include UAV path planning and area search.

**Ai Xiaolin** received his B.S. degree in Beijing Institute of Technology in 2013. He is currently working toward a doctoral degree at School of Aerospace Engineering, Beijing Institute of Technology. His research interests are flight dynamics and control, cooperative control and flight vehicle system design.

**Jia Zhenyue** received his B.S. degree in Beijing Institute of Technology in 2014. He is currently working toward a doctoral degree at School of Aerospace Engineering, Beijing Institute of Technology. His research interests are flight dynamics and control, cooperative control and UAV mission planning.