Proceedings of The 2021 IEEE International
Conference on Real-time Computing and Robotics
July 15-19, 2021, Xining, China

Trajectory Planning for Collaborative Transportation by Tethered Multi-UAVs

Chongxu Pei, Fan Zhang, Panfeng Huang, Hang Yu

***Abstract*— Collaborative transportation has been widely studied due to its high efficiency, and interesting research problems, specifically in trajectory planning. In this paper, a new trajectory planning scheme for collaborative transportation by tethered multi-UAVs is proposed to avoid obstacles. Firstly, the dynamics and configuration of the system is analyzed to derive the tethered force acting on each UAV, and define the angle between the rope and the vertical direction is calculated. Then the kinodynamic path search method is used to search an initial trajectory of the payload, and the initial trajectory of each UAV is solved. Finally, nonlinear optimization is used to optimize the trajectory of each UAV and the payload to achieve the effect of obstacle avoidance. Both numerical simulations and practical experiments are completed to verify the feasibility and effectiveness of the proposed planning scheme.**

(a) three QDrone UAVs transport suspended load

(b) OptiTrack indoor positioning system

Figure 1. The three drones work together to transport suspended load and avoid obstacles. The algorithm in this paper uses three QDrone UAVs (a) for implementation, and QDrone uses OptiTrack indoor positioning system (b) for positioning.

I. Introduction

In recent years, the application market of unmanned aerial vehicles (UAVs) is more and more extensive, and the application environment is getting worse and worse. Single UAV missions are relatively common and well established. However, compared to a single UAV, multiple UAVs can provide better system versatility, safety and deployability, and can reduce the overall cost of the system. For example, in the application scenario of UAV transport load, the load capacity of a single UAV is limited, and the energy consumption is fast. The use of multiple UAVs to carry load can reduce the overall cost of the system, while increase the transport capacity and robustness of the system. The cost is that complex trajectory planning algorithm is needed to generate the trajectories of each UAV, which need to satisfy more complex optimization constraints.

A single UAV transport payload has been around since the 1960s, when it was first introduced in [1]. The main application scenarios include earthquake relief, remote area material transportation, battlefield material transmission, etc. Now many applications of UAV transport load are introduced in [2-5]. At the same time, there are also some works with multiple drones to coordinate the transport of load. Research on the inverse kinematics solution of UAVs cooperative transportation scene was introduced in [6]. A series of sensors are used as the information exchange media between unmanned aerial vehicles in [7]. A method to solve the problem that the robot configuration may have multiple payload balance solutions was proposed [8]. By making constraint conditions for the robot configuration, the unique payload pose could be guaranteed.

It is essential for the UAV to plan a trajectory that conforms to the constraints of UAV dynamics, obstacle avoidance and trajectory smoothness to complete the load transportation task. In this paper, a trajectory planning method for multiple UAVs cooperative transport of suspended load (Fig.1) is introduced. Compared with the existing work, the method described in this paper can plan a flight trajectory after giving map information, UAV dynamics constraints, UAV size parameters and rope length. The trajectory is generated using this method allows UAVs to move through a narrow space or get around an obstacle in formation while meeting the requirements of a cooperative transport suspension load. The trajectories of all the UAVs and payload are simultaneously planned. However, in order to satisfy the constraint conditions first, the smoothness of the trajectory is globally suboptimal. Contribution of this paper as follows：

1) A hierarchical planning method is proposed for trajectory planning in the environment where multiple agents work together to carry loads.

2) The obstacle avoidance constraint in hybrid A* algorithm is extended, and the obstacle avoidance constraint of UAV formation is added. This ensures that the search path must be able to allow the passage of the UAV formation.

3) The nonlinear optimization problem is defined based on the idea of Minimum Jerk. At the same time, a variety of optimization constraints are defined based on the special environment of multi-UAV cooperative load handling.

The remainder of this letter is organized as follows: Section II introduces some literature related to this letter. Section III introduces some details of front-end trajectory search. Section IV formulates the cable suspension load problem as a trajectory optimization problem. Section V describes the specific details of the simulation experiment and the real experiment. The paper is concluded in Section VI.

*Research supported in part by the National Natural Science Foundation of China under Grant Nos.61803313, 91848205, 61725303, in part by the Fundamental Research Funds for the Central Universities under Grant No. 3102019HTQD003, and in part by the Young Talent Fund of University Association for Science and Technology in Shaanxi, China under Grant No. 20190102, and in part by the Natural Science Basic Research Plan in Shaanxi Province of China (No. 2019JQ-345, 2019JQ-411, 2019JM-392, 2019JM-406,). (corresponding author: Fan Zhang, phone: (+86)029-88460366; fax: (+86)029-88460366-4; e-mail: fzhang@nwpu.edu.cn)

Chongxu Pei , Fan Zhang, Panfeng Huang and Hang Yu are with the National Key Laboratory of Aerospace Flight Dynamics, Research Center for Intelligent Robotics, School of Astronautics, Northwestern Polytechnical University, 127 West Youyi Road, Xi'an, China (e-mail: peichongxu@mail.nwpu.edu.cn,fzhang@nwpu.edu.cn,pfhuang@nwpu.edu.cn, yuhang@mail.nwpu.edu.cn).

2021 IEEE International Conference on Real-time Computing and Robotics (RCAR) | 978-1-6654-3678-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/RCAR52367.2021.9517336

Authorized licensed use limited to: Tsinghua University. Downloaded on August 28,2023 at 06:36:08 UTC from IEEE Xplore. Restrictions apply.

II. Related Work

There are a large number of algorithms and applications for single robot motion/trajectory planning. A fast and robust trajectory planning algorithm was proposed in [9]. This algorithm first uses the hybrid A* algorithm to find a path that satisfy the dynamics requirements. Then, based on the known path, the nonlinear optimization algorithm is used to optimize the path to generate a trajectory of the UAV. This algorithm has a good effect in trajectory planning of single UAV. But this algorithm has not been used in multi-drone cooperation.

The trajectory of robot motion can be represented by polynomials, of which there are some special advantages to using Bezier curves to represent the trajectory of robot motion. Bezier curve is used for path generation and tracking in [10], and the generated path is used for autonomous driving vehicles. Bezier curves are used to limit the steering speed of vehicles. Bezier curves are also easy to use for obstacle avoidance.

A distributed trajectory optimization based suspension load for multi-UAV transport is proposed in [11]. This method uses the optimization method to optimize the generation of the trajectory. This method puts forward the constraint of the trajectory optimization problem of multi-UAV cooperative transportation. Different from the trajectory optimization constraints of a single UAV, collision prevention constraints and rope length constraints between multiple UAVs are also required in this scenario. By using the ALTRO optimization algorithm proposed in [12] to solve the trajectory optimization problem, a trajectory solution is obtained. Then each trajectory is solved in a circular way, thus reducing the time needed for the optimization algorithm. However, this algorithm is not distributed optimization in the complete sense, because its optimization is carried out sequentially rather than simultaneously. And the method needs a good initial solution.

All states of the UAV can be represented only by its three-dimensional coordinates and yaw angle, and this proof was proposed by [13]. The idea of minimum snap method for trajectory planning of UAV is also proposed. If the trajectory of a UAV is represented by a polynomial, then the first derivative of this polynomial is the velocity and the second derivative is the acceleration. Because the acceleration of UAV is linearly related to the combination Angle of UAV roll Angle and pitch Angle. So the third derivative of the track(jerk) is the angular velocity, and the fourth derivative(snap) is the angular acceleration. Jerk is related to the smoothness of the drone's track, and snap is related to the energy consumption of the drone. Under this idea, the minimum jerk of UAV is optimized to ensure the high smoothness of UAV trajectory.

Based on the ideas presented in the above paper, a novel trajectory planning algorithm is proposed, which can be used in the special scenario of multiple UAVs cooperating to carry loads, and solve the flight trajectory of each UAVs with only map information given. At the same time, the optimization problem can be solved globally, and the smoothness of all trajectories is guaranteed under the condition of satisfying the constraints.

III. Front-end Trajectory Search

This front-end path searching module is based on the Kinodynamic path search proposed in [9], which is originated from the hybrid-state A* search first proposed for autonomous vehicle in [14]. The idea of A* algorithm is to extend each extended raster point in all directions by conducting an extended search in A known raster map. And the cost of expansion is solved by the cost function **costGn()**, and the path cost can be guaranteed to get the trajectory with the least cost by comparing the path cost. The heuristic function **costHn()** is used to give the grid a priority expansion direction and accelerate the expansion speed. As shown in Algorithm 1, the cyclic idea of Hybrid A* algorithm is basically the same as that of A* algorithm. The main difference is that the expansion mode is transformed from A straight line to A curved path that conforms to the kinematics constraints of the robot, and its **costGn()** and **costHn()** are changed. The path explored in this way meets the requirements of robot dynamics and can be directly used by the robot.

Algorithm 1: A Star Search.

```
Initialize();
while open_num > 0 do
    cur_node ← node(min_fc);
    if Terminate then
        return path_node,coef_shot,t_shot;
    end if
    pro_node = stateTransit(cur_node, um, tau);
    if not_feasible then
        continue;
    end if
    pro_fc ← costGn(pro_node) + costHn(pro_node);
    if not_same then
        node ← insert(pro_node);
    else if same_fc > pro_fc
        node(same) ← pro_node;
    end if
end while
```

A. Node Expansion Mode

The three dimensions of the UAV are completely decoupled, so the state on each dimension can be calculated separately. For each dimension, the maximum acceleration is discretized. The acceleration as the input variable, give a time, combine the current position and velocity of the node to calculate, and get the position and velocity of the new node. The expansion formula on one dimension is

$$p_{\mathrm{new}} = p_{cur} + v_{cur} t_{\exp} + \frac{1}{2} a_{\exp} t_{\exp}^2$$
$$v_{new} = v_{cur} + a_{\exp} t_{\exp}, \tag{1}$$

where $a_{\exp} \in (-a_{\max}, a_{\max})$ is the acceleration of this expansion. $t_{\exp} \in (t_{rat} t_{\max}, t_{\max})$ is the duration of this expansion, t_{rat} is the discrete scale coefficient and $t_{\max}$ is the maximum time of single expansion. v_{cur} is the speed of the current node, p_{cur} is the position of the current node. v_{new} is the speed of the extended node. p_{new} is the position of the extended node.

Each time the nodes are extended, the locations and speeds are basically different, and there will be multiple extension points in the same grid. In order to reduce the amount of calculation and improve the efficiency of calculation. In the expansion of each point, only the one extension point with the lowest total cost of f_c is retained in each grid.

Authorized licensed use limited to: Tsinghua University. Downloaded on August 28,2023 at 06:36:08 UTC from IEEE Xplore. Restrictions apply.

B. Actual Cost and Heuristic Costs

Acceleration is defined as an input variable and add a time penalty. So as far as possible to find the path to reach the minimum energy consumption and the fastest. The cost of the trajectory is defined as

$$J(T) = \int_0^T \left\| a(t) \right\|^2 dt + \rho T$$
$$g_c = \sum_{\mu \in \{x,y,z\}} J_\mu(T) \quad , \tag{2}$$

where T is the sum of the time from the starting point to the current expansion point. $J(T)$ is the real cost in one dimension. g_c is the actual total cost.

In order to find the end point faster, a heuristic function h_c need to be given. The Pontryagins Minimum Principle is used to calculate the cost from the current point to the end point. The detailed derivation is described in [15], the final calculation formula as

$$p^*(t) = \frac{1}{6}\alpha t^3 + \frac{1}{2}\beta t^2 + v_{cur} t + p_{cur}$$
$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \frac{1}{T^3} \begin{bmatrix} -12 & 6T \\ 6T & -2T^2 \end{bmatrix} \begin{bmatrix} p_{goal} - p_{cur} - v_{cur} T \\ v_{goal} - v_{cur} \end{bmatrix},$$
$$J^*(T) = \sum \left(\frac{1}{3}\alpha^2 T^3 + \alpha\beta T^2 + \beta^2 T\right)$$
$$h_c = \sum_{\mu \in \ x,y,z} J_\mu^*(T) \tag{3}$$

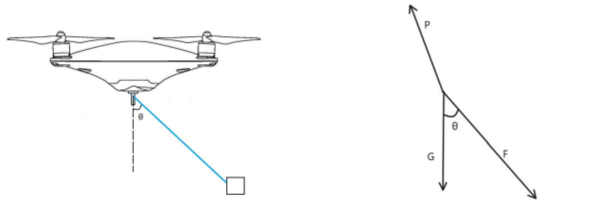
where v_{goal}、p_{goal} is the speed and position of the end point. T is the time from the current point to the end point. $J(T)$ is the predicted cost in one dimension, and the derivative of $J(T)$ with respect to T is equal to zero $\frac{\partial J(T)}{\partial T}=0$ to find the optimal time T, and thus to get the minimum $J^*(T)$. h_c is the predicted total cost. Finally, f_c is defined as $f_c = g_c + h_c$.

C. Obstacle Avoidance Judgment in Formation

The dynamics analysis of a UAV's cable-pulled load is shown in Figure 2. There are three forces on a single drone: its own gravity(G), the upward push of the wings(P), and the pull on the ropes(F). The relationship between these forces as

$$P = F + G$$
$$\mathrm{F} = \mathrm{G}_{load} \cos^{-1}\theta \quad , \tag{4}$$
$$l_{dis} = l_{cab} \sin\theta$$

where G_{load} is the load weight, θ is the Angle between the rope and the vertical direction, l_{dis} is the horizontal distance from the UAV to the payload, l_{cab} is the length of the rope. From these forces, the maximum allowable distance between the drones can be solved. According to the maximum thrust P_{max} of the UAV and G, the maximum tension F_{max} can be solved, and then the maximum Angle θ_{max} can be solved. Thus, the maximum distance l_{dis}^{max} can be solved. The minimum distance between the drones l_{dis}^{min} is the horizontal distance from the center of the drone to the boundary.



(a) The UAV cable-suspended the load (b) Force analysis diagram

Figure 2. The UAV cable-stayed suspension load is shown in Figure (a), where θ is the Angle between the rope and the vertical direction. The force analysis of the UAV is shown in Figure (b), where P is the thrust of the UAV, F is the pull on the rope, and G is the gravity of the UAV itself.

Obstacle-avoidance judgment is needed for each expansion node to determine whether the whole UAVs handling formation encounters obstacles. Therefore, after obstacle collision detection on the extension node itself, a cuboid bounding box is generated according to the overall size of the UAVs formation. Obstacle collision detection is carried out on the eight vertices and central points of the cuboid. If all the points do not hit the obstacle, it can be considered that the entire UAVs formation did not hit the obstacle.

D. One Shot Find Path

Using Hybrid A* for the path search, the search points are not in grid units. So extending at one point is hard to search straight to the end. The One Shot method is used to extend directly to the finish line. As shown in algorithm 1, a termination judgment is required for each loop. In the termination judgment, (3) is used to find a path connecting the current point to the end point. Then obstacle detection, velocity detection and boundary constraint detection are carried out on this path. If all constraints are satisfied, the trajectory is output directly, thus finding the path from the beginning to the end. This method greatly reduces the time required for the search.

IV. Back-end Trajectory Optimization

The front-end trajectory search based on Hybrid A* algorithm cannot find a globally optimal trajectory, only find one suboptimal solution. And the front-end trajectory search only looks for the trajectory of the load, while the trajectory of the UAV needs to be solved by the size of the formation. So the trajectory of the drone is likely to hit an obstacle. For these problems above, the trajectory generated by the front end can be further optimized by means of optimization, so as to generate a trajectory that can satisfy various constraints and is globally suboptimal. In order to reduce the number of constraints and simplify the expression of the constraint, Bezier curves are used to represent trajectories that need to be optimized. The Bezier curve is explained in detail in IV-C. The optimization variable is the control point of the Bezier curve. The idea of minimum jerk is used to optimize the problem to minimize the 3rd derivative of the trajectory, so that the trajectory smoothness is higher.

A. Initial Solution of The Optimize Variable

Firstly, the load trajectories generated by the front end are extended by the UAV formation size to obtain the trajectories of three UAVs. Then every trajectory is sampled, and the

Authorized licensed use limited to: Tsinghua University. Downloaded on August 28,2023 at 06:36:08 UTC from IEEE Xplore. Restrictions apply.

Polynomial coefficient $P(n)$ of trajectory is fitted with the sampled values. The relationship between $P(n)$ and Bezier curve control points $B(n)$ can be expressed as

$$B(n) = M^{-1}P(n)\,, \tag{5}$$

where n is the order of the polynomial and the Bezier curve. M is a fixed matrix whose dimension and its values are related to n. There is a total of one payload track and n_{uav} bar drone track. Each trajectory has an n_{seg} segment and has three dimensions. The total number of optimized variables is expressed as

$$n_{poly} = 3(1+n_{uav})n_{seg}(n+1)\,. \tag{6}$$

B. Optimization Problem

Minimum jerk is used here, the integral of square jerk is used to represent the optimization cost. Jerk is the third derivative of polynomial trajectory, which represents the sum of the angular velocity of pitch and roll of UAV, and is directly related to the smoothness of trajectory.

$$f(t) = \sum_i P_i t^i$$

$$(f^{(3)}(t))^2 = \sum_{i,l\ge 3} i(i-1)(i-2)l(l-1)(l-2)t^{i+l-6}P_iP_l$$

$$J(T) = \int_{T_{j-1}}^{T_j} (f^{(3)}(t))^2 dt = P^TQP = B^TM^TQMB\,, \tag{7}$$

where $f(t)$ is polynomial expression. $J(T)$ is the cost over a trajectory in one segment. $J(T)$ in terms of $B(n)$ can be derived from (7). The total cost is equal to the sum of the costs of each of the trajectories, the optimization problem is as follows:

$$\underset{B}{minimize}\, J_{all}(T) = \sum_{b=1}^{1+n_{uav}} \sum_{m=1}^{3} \sum_{n=1}^{n_{seg}} J(T)\,. \tag{8}$$

C. Optimization Constraints

This optimization problem uses Bezier curves to represent the trajectory of the UAV and the payload. Bezier curves have some nice properties that are useful for trajectory planning problems. A fourth-order Bezier curve is shown in Figure 3.

1) The Bezier curve only passes through the first control point and the last control point.
2) Bezier curves must be contained within the convex hull where all control points are located.
3) The derivative of a Bezier curve is still a Bezier curve, and the coefficient of the derivative is $n(B_{i+1} - B_i)$.
4) The time of the Bezier curve must be in [0,1].

By virtue of the fine properties of Bezier curves, it is facility to add constraints to the optimization variables. The trajectory optimization constraints of multi-UAVs cooperative handling of suspension load mainly include starting point and finishing point constraints, dynamic constraints, obstacle avoidance constraints, rope length constraints, anti-collision constraints between unmanned aerial vehicles, and constraints of maximum formation of unmanned aerial vehicles. A constraint on a trajectory is introduced here:

$$V_i^l = n(B_{i+1}^l - B_i^l),\ i\in[1,\ n-1],\ l\in[1,\ n_{seg}] \tag{9.1}$$

$$A_i^l = n(V_{i+1}^l - V_i^l),\ i\in[1,\ n-2],\ l\in[1,\ n_{seg}] \tag{9.2}$$

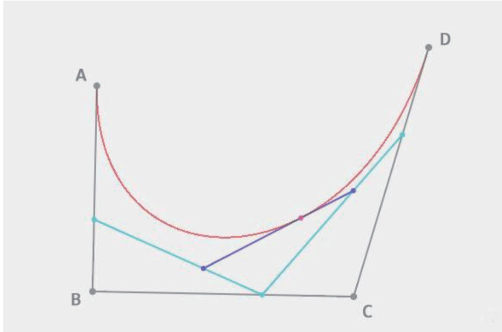


Figure 3. The red line is A fourth-order Bezier curve, and points A,B,C and D are the four control points of the curve.Bezier curves only pass through the first control point and the last control point.And the bezier must be contained in a convex hull consisting of control points.

$$B_0^1 = p_{start},\ V_0^1 = v_{start},\ A_0^1 = a_{start} \tag{9.3}$$

$$B_n^{n_{seg}} = p_{end},\ V_{n-1}^{n_{seg}} = v_{end},\ A_{n-2}^{n_{seg}} = a_{end} \tag{9.4}$$

$$B_n^l = B_0^{l+1},\ V_{n-1}^l = V_0^{l+1},\ A_{n-2}^l = A_0^{l+1},\\ l\in[1,\ n_{seg}-1] \tag{9.5}$$

$$p_{min}^l \le B_a^l \le p_{max}^l, p_{min}^{l+1} \le B_b^l \le p_{max}^{l+1},\\ l\in[1,\ n_{seg}] \tag{9.6}$$

$$V_{min} \le V_i^l \le V_{max},\ i\in[1,\ n-1],\ l\in[1,\ n_{seg}] \tag{9.7}$$

$$A_{min} \le A_i^l \le A_{max},\ i\in[1,\ n-2],\ l\in[1,\ n_{seg}] \tag{9.8}$$

$$\left\|(r_b)_i^l - r_i^l\right\|_2 = l_{cab},\ b\in(1,2,3),\ l\in[1,\ n_{seg}] \tag{9.9}$$

$$2l_{dis}^{min} \le \left\|(r_m)_i^l - (r_n)_i^l\right\|_2 \le 2l_{dis}^{max},\\ m,\ n\in(1,2,3),\ m\ne n,\ \ l\in[1,\ n\_seg] \tag{9.10}$$

where V_i^l is the control point of the velocity curve. A_i^l is the control point of the acceleration curve. p_{start}, v_{start}, a_{start} is the starting position, velocity, and acceleration, and p_{end}, v_{end}, a_{end} is the end position, velocity, and acceleration. p_{min}^l and p_{max}^l is the maximum and minimum values of the flight corridor around a path point. V_{min} and V_{max} is maximum and minimum velocities. A_{min} and A_{max} is maximum and minimum acceleration. l_{cab} is the length of the rope. B_i^l represents the i-th control point of the trajectory of segment l. V_i^l and A_i^l are analogous to B_i^l.

Start state constraints and end state constraints are represented by (9.3) and (9.4). (9.5) represents continuity constraint, the last control point of each trajectory should be equal to the first control point of the next trajectory, so that the trajectory can be continuous. Obstacle avoidance constraints are expressed in (9.6), where $a\in[1, floor(n/2)]$ and $b\in[floor(n/2)+1, n\_seg]$. The **floor()** is an integer down function. The control points of a trajectory are separated

Authorized licensed use limited to: Tsinghua University. Downloaded on August 28,2023 at 06:36:08 UTC from IEEE Xplore. Restrictions apply.

into two halves. The first half is in the flight corridor found at the starting point of the trajectory, and the second half is in the flight corridor found at the end point of the trajectory. Using this method can increase the success rate and quality of trajectory optimization when the intersection volume of two adjacent flight corridors is small.

The path point is the starting point and the end point of each segment on the trajectory. Each path point is extended to the periphery to obtain a cuboid that does not touch obstacles and is within the boundary of the map. The rectangular body of the UAV and the payload are considered safe. The specific form of the flight corridor is shown in Figure 4. The dynamic constraints of the UAV are reflected in (9.7) and (9.8), and each control point on the Bezier curve of velocity and acceleration should be within the allowable range of velocity and acceleration.

In (9.9) and (9.10), R is used to represent the three dimensional coordinates of control points on the trajectory of each UAV or payload. The order of Bezier curve can be regarded as the sampling times of the curve. When the length of a trajectory is small enough and the order is large enough, Bezier's control points will be very close to the curve. In this case, the control points can be regarded as real-time sampling points on the trajectory. The distance between control points can be regarded as the real-time distance between two tracks. (9.9) represents the rope length constraint, where the distance from each UAV to the load is equal to the rope length. (9.10) represents the obstacle avoidance constraint between UAVs. The distance between two UAVs cannot be less than the minimum or greater than the maximum of the formation.

V. Simulation and Experiment

The order of Bezier curve is chosen as 4. The drone has a length of 0.4m, a width of 0.4m and a height of 0.2m.The rope length is 1.27m.Enlarge the actual map by 10 times, and then raster the map by unit to get the information of obstacles on the raster map. The first expansion time of the front end is 0.8s, the other expansion time is 0.5s each time, the maximum speed is 1.5m/s, the maximum acceleration is 0.5 m/s^2, the time penalty coefficient is 10, and the number of obstacle avoidance detection is 10.The sampling time of the front-end path is 0.3s, and the path time of each segment after fitting is 1s. The maximum velocity and maximum acceleration of the back-end are 3 m/s and 1.5 m/s^2, and the time of each segment of Bezier curve is 1s. The fmincon function in Matlab was selected for nonlinear optimization, and the optimization algorithm was set as internal-point, MaxFunEvals as 1e6,TolCon as 0.1,TolFun as 0.1, and TolX as 1e-10.

A. *Simulation result*

The algorithms were implemented on a desktop computer containing an Intel Core i5 8400 processor and 8 GB of RAM. In the simulation environment, a series of continuous rectangular obstacles are set up. For different starting points and end points, the algorithm can intelligently judge whether to pass through obstacles or bypass obstacles to reach the end point (Fig. 5). The path through the obstacle to the destination will automatically judge whether the distance of the obstacle object can meet the demand of the overall formation of the UAV, and it will give priority to the area with large safe space. The problem cost and constraint cost as well as optimization time are expressed in Table 1. Getting around the obstacle is scenario 1, and going through the obstacle is scenario 2. Either way, the obstacle avoidance constraint can be well satisfied. None of the drones, loads, or ropes hit the obstacle. And the cost of optimization problems is kept below 10. It indicates that the smoothness of the planned trajectory also reaches a high level. The cost of the optimization constraint is also kept below 1, indicating that the satisfaction degree of the overall constraint is very high. The individual velocity, acceleration curve, anti-collision judgment curve and rope length constraint curve of one UAV are shown in Fig. 6.

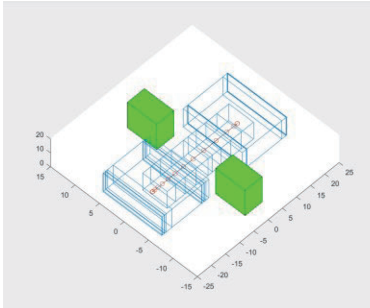
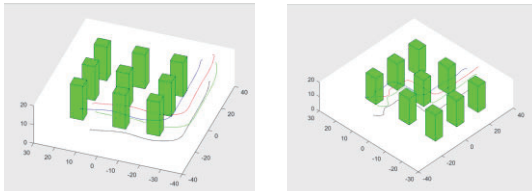


Figure 4. The figure is a flight corridor diagram of a flight path.Green cuboids represent obstacles, orange circles represent path points, and blue boxes represent flight corridors around each path point. The coordinate value is 10 times the actual coordinate.



(a) Get around an obstacle (b) Pass through an obstacle

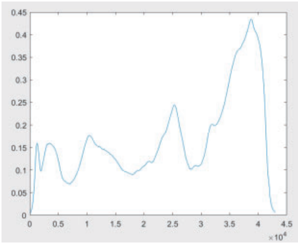
Figure 5. The figure shows the track of multiple unmanned aerial vehicles' cooperative transportation of load to the destination in the environment of multiple cuboid obstacles.The black line in the figure is the trajectory of the load, and the other three colored lines are the trajectories of the three UAVs.

TABLE I. Simulation Results

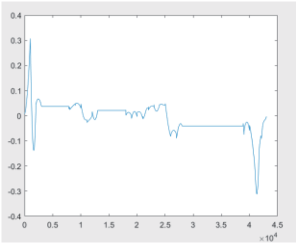
Scenarios	Optimization Results		
	Problem Cost	*Cost constraints*	*Time*
1	5.762640	0.1682	1246s
2	5.126579	0.2531	1439s
3	3.797553	0.04016	278s
4	1.068686	0.03985	358s
5	0.4496941	0.004995	479s

As represented by UAV 1, the velocity curve and acceleration curve of this UAV are presented, which are shown in Fig. 6 (a) and (b).It can be seen that the speed of the UAV is limited within the allowable range of speed and does not exceed 3m/s. The acceleration of UAV does not exceed 1.5m/s^2, which is limited within the allowable range of acceleration, satisfying the dynamic constraints of UAV operation. In addition, the position curve, velocity curve and acceleration curve are all

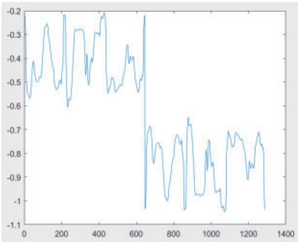
Authorized licensed use limited to: Tsinghua University. Downloaded on August 28,2023 at 06:36:08 UTC from IEEE Xplore. Restrictions apply.



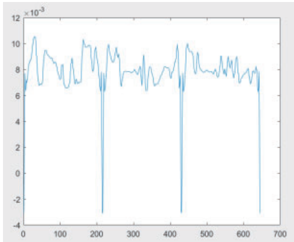
(a) The speed curve of UAV 1



(b) Acceleration curve of UAV 1



(c) UAV anti-collision verification curve



(b) Rope length constraint curve

Figure 6. In (a), the flying velocity curve of UAV 1 is shown; in (b), the velocity curve of UAV 1 is shown; in (c), the test value of anti-collision constraint between UAVs is shown; and in (d), the test value of rope length constraint is shown.

continuous curves, which satisfy the continuity constraint. The spacing detection formula between UAVs is set as

$$\begin{aligned} c_i &= \left\| (r_m)_i^l - (r_n)_i^l \right\|_2 - 2l_{dis}^{max} \\ c_{i+n_{uav}n_g(n+1)} &= 2l_{dis}^{min} - \left\| (r_m)_i^l - (r_n)_i^l \right\|_2 \quad , \\ i &\in [1,\ n_{uav}n_g(n+1)] \end{aligned} \tag{10}$$

where c_i is test value. It can be seen from (10) that as long as c_i is constant and less than 0, the spacing between unmanned aerial vehicles is guaranteed to be within the required range. The trajectory generated by this method satisfies the anti-collision constraint between UAVs, which is shown in Fig. 6 (c). The rope length constraint is detected as

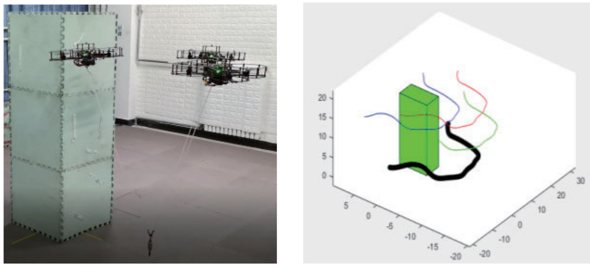
$$c_{eq} = \left\| (r_b)_i^l - r_i^l \right\|_2 - l_{cab},\ b \in (1,2,3),\ l \in [1,\ n_{seg}], \tag{11}$$

where c_{eq} is test value. Only when c_{eq} is equal to 0 does the distance from the UAV to the payload equal the rope length. As can be seen from Figure 6 (d), CEQ is generally less than 1e-2, which is a very good value. It shows that the rope length constraint is well realized.

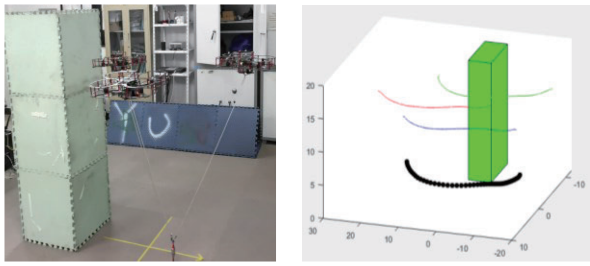
B. Experiment Result

The flying around a single obstacle and passing through two obstacles were verified. In the experiment, three Quanser QDrone drones and the OptiTrack indoor positioning system comprised of 12 cameras were used. The effect of the actual flight is shown in Figure 6, and the result of path optimization is shown in Table 1. Passing between two obstacles is Scene 3, bypassing from the left of the obstacle is Scene 4, and bypassing from the right of the obstacle is Scene 5. The full flight video is more than 200MB, and the video can be obtained at https://www.bilibili.com/video/BV1hK4y1S7MW?from=search&seid=661029385619294354.

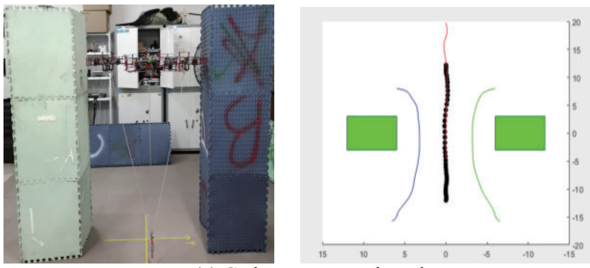
Numerically, the optimization cost in these three scenarios is all below 5, and the constraint cost is guaranteed to be below



(a) Go around the obstacle on the right



(b) Go around the obstacle on the lift



(c) Go between two obstacles

Figure 7. Three QDrone carry load together and pass the experimental verification of obstacles. The blue, red and green lines represent the flight path of each drone, and the black lines represent the flight path of the payload. In (b) and (c), the black dots on the black line represent the control points of the Bezier curve.

0.1, and the overall flight trajectory is very smooth. From the actual experimental results, according to the optimized trajectory, the UAV can avoid obstacles perfectly in the actual flight test. Even in the case of passing through the narrow space between two obstacles, as demonstrated in Figure 7 (c), the drones do not collide with each other or hit the obstacles. And in flight, the UAVs can maintain a whole formation between each other, which can be well kept within the range of rope length constraints. There were no instances where a single drone left the group and disorganized the formation. The speed of each drone in flight was maintained at maximum speed, and there was no violent acceleration. The overall flight was very smooth.

VI. Conclusion

In the present work, a novel method is proposed to solve the trajectory planning problem of multi-UAV cooperative transport suspension load. In this algorithm, the hybrid A* algorithm is used to search the initial path, and an initial solution is obtained. The Bezier curve is used to describe the path of the UAV, and the control points of the Bezier curve are taken as the optimization variables. The smoothness of the trajectory of the UAV is taken as the optimization problem

Authorized licensed use limited to: Tsinghua University. Downloaded on August 28,2023 at 06:36:08 UTC from IEEE Xplore. Restrictions apply.

using the idea of the Minimum Jerk. Thus, the trajectory planning problem of UAV is transformed into a nonlinear optimization problem to solve. Then the proposed algorithm is verified by simulation and real experiment. It is proved that the proposed method can generate trajectories for multi-UAV be smooth enough to satisfy all the constraints. The proposed method can simultaneously plan the trajectories of three unmanned aerial vehicles and their load, so that a path with high smoothness and low energy consumption can be obtained under the condition of satisfying the constraints. Moreover, the proposed method does not need to obtain a good initial path in advance, and can directly solve the trajectory of the UAV after the map information is given. This method also has some disadvantages: the optimization time of this method is greatly affected by the number of UAVs, and the optimization time is relatively long. Therefore, this method is only applicable to offline global programming. In the future, the optimization algorithm can be replaced to improve the calculation speed and achieve the effect of implementation planning.

REFERENCES

[1] T. Lancashire, R. T. Lytwyn, G. Wilson, and D. Harding, “Investigation ofthe mechanics of cargo handling by aerial crane-type aircraft,” Boeing CoMorton Pa V ertol Div, Tech. Rep., 1966.

[2] K. Sreenath, T. Lee, and V . Kumar, “Geometric control and differential flatness of a quadrotor UAV with a cable-suspended load,” in Proc. 52nd IEEE Conf. Decis. Control, 2013, pp. 2269–2274.

[3] S. Tang and V . Kumar, “Mixed integer quadratic program trajectory generation for a quadrotor with a cable-suspended payload,” inProc. IEEE Int. Conf. Robot. Autom., 2015, pp. 2216–2222.

[4] C. De Crousaz, F. Farshidian, and J. Buchli, “Aggressive optimal control for agile flight with a slung load,” in Proc. IROS Workshop Mach. Learn.Planning Control Robot Motion, 2014.

[5] P . Foehn, D. Falanga, N. Kuppuswamy, R. Tedrake, and D. Scaamuzza,“Fast trajectory optimization for agile quadrotor maneuvers with a cable-suspended payload,” in Proc. Robot.: Sci. Syst., 2017, pp. 1–10.

[6] Q. Jiang and V. Kumar, "The Inverse Kinematics of Cooperative Transport With Multiple Aerial Robots," in IEEE Transactions on Rbotics, vol. 29, no. 1, pp. 136-145, Feb. 2013

[7] Tagliabue, Andrea , et al. "Robust Collaborative Object Transportation Using Multiple MAVs." International Journal of Robotics Rsearch, 2017.

[8] Fink, Jonathan , et al. Planning and Control for Cooperative Manipulation and Transportation with Aerial Robots. Springer Berlin Heidelberg, 2011.

[9] Zhou, Boyu , et al. "Robust and Efficient Quadrotor Trajectory Generation for Fast Autonomous Flight." IEEE Robotics and Automation Letters ,2019.

[10] Bae, Il , et al. "Path generation and tracking based on a Bézier curve for a steering rate controller of autonomous vehicles." International IEEE Conference on Intelligent Transportation Systems IEEE, 2013.

[11] Jackson, Brian E. , et al. "Scalable Cooperative Transport of Cable-Suspended Load With UAVs Using Distributed Trajectory Optimization." IEEE Robotics and Automation Letters PP.99(2020):1-1.

[12] Howell, Taylor A. , B. E. Jackson , and Z. Manchester . "ALTRO: A Fast Solver for Constrained Trajectory Optimization." 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) IEEE, 2020.

[13] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," 2011 IEEE International Conference on Robotics and Automation, Shanghai, 2011, pp. 2520-2525

[14] Dolgov, Dmitri , et al. "Path Planning for Autonomous Vehicles in Unknown Semi-structured Environments." The International Journal of Robotics Research 29.5(2010):485-501.

[15] Mueller, Mark W. , M. Hehn , and R. Dandrea . "A Computationally Efficient Motion Primitive for Quadrocopter Trajectory Generation." IEEE Transactions on Robotics 31.6(2017):1294-1310.

Authorized licensed use limited to: Tsinghua University. Downloaded on August 28,2023 at 06:36:08 UTC from IEEE Xplore. Restrictions apply.