Route Planning of Unmanned Aerial Vehicle Based on Sparse A* Algorithm

Tian-shu BI, Ping YE, Yu-nong XU and Fei-hu ZHANG
Department of Automation, Beijing University of Posts and Telecommunications, Beijing, China

Keywords: Optimized route planning, Unmanned aerial vehicle, Dynamic re-planning.

Abstract. Considering the matters of accuracy, real-time, task adaptability and dynamic re-planning for the route planning of unmanned aerial vehicle. When the unmanned aerial vehicle is planning its route on a large-scale map, it needs to ensure better real-time performance. If new obstacles are encountered during flight, the route planning algorithm needs to be able to carry out dynamic planning to re-plan the route to avoid the new obstacles. This paper presented an optimized route planning of unmanned aerial vehicle algorithm, which mainly through the management of number of search nodes and stratification strategy while still maintaining a safe route. We also presented a dynamic route re-planning function when encountering new obstacles by updating the map. The algorithm has better real-time performance and robustness. It has great reference value for the practical application of unmanned aerial vehicle.

Introduction

Route planning of unmanned aerial vehicle is based on comprehensive consideration of the arrival time of the aircraft, fuel consumption, threats and flight areas. Under the premise of the factors, it can plan an optional or satisfactory route to ensure a satisfactory completion of flight mission[1]. With the development of military forces in various countries, the complexity of modern environment situations has gradually increased, and the traditional route planning algorithms have been unable to meet the needs of high-complexity battlefield environments[2]. At present, dynamic track has also become an important method to improve the environmental adaptability and safety of unmanned aerial vehicle. Therefore, the unmanned aerial vehicle is required to avoid the threat in the mission area and reach the specified target safely when panning the route. It puts high demands on the accuracy, real-time, task adaptability. Dynamic route planning needs receive and analyze threat area quickly, then update it to the map and change the pre-planned route to reduce the probability of the unmanned aerial vehicle meeting the dynamic threat. However, due to the uncertainty of the external environment, the results of the dynamic planning of the track are unpredictable, and it is possible that the unmanned aerial vehicle cannot complete the tasks.

The unmanned aerial vehicle needs to plan a safe route according to the threat area in the map, and can adaptive change pre-planned trajectory to perform real-time obstacle avoidance when updating the dynamic obstacle in the map. Due to limitations of the unmanned aerial vehicle hardware platform, such as fuel consumption. If the algorithm cannot plan the route with high real-time performances, this route panning algorithm is difficult to get practical application. Therefore, this paper proposes a method of track planning, which can plan the route that the unmanned aerial vehicle will fly according to the planned route. The new threat area will be updated in the map while the flight is in process and has an impact on the route. The new algorithm can quickly adaptive change the pre-planned route, which is a route planning method combining route pre-planning with dynamic obstacle avoidance. The experimental platform of this paper is PC.

The methods for solving the route problem of unmanned aerial vehicle are mainly divide into graph theory method and grid algorithm, such as graph theory-based Voroni algorithm[3], PRM algorithm[4], grid-based A* algorithm and dynamic programming algorithm[5][6]. Considering the real-time demands of this paper. We choose A* algorithm. In this paper, we use sparse A* algorithm to optimize the A* algorithm. The A* algorithm is optimized based on the hierarchical strategy of
large step size rough plan and small step size fine plan. When the map updates dynamic obstacles, the original is partially localized changed so that the target can still be reached.

This paper is organized as follows: The problems faced by route planning of unmanned aerial vehicle in practical application are described in Section I. In Section II, the route planning algorithm proposed in this paper is introduced. In Section III, we detail our method about optimized route planning of unmanned aerial vehicle. The conducted experiments and result are shown in Section IV, followed by conclusions in Section V.

**System Overview**

Aiming at the unmanned aerial vehicle path planning algorithm proposed in this paper, we establish a threat model that is used in the obstacle map[7]. This paper proposed three basic assumptions. The first assumption is that the unmanned aerial vehicle is in a fixed-high and constant-speed flight state, which can simplify the path planning to the route planning in the 2-dimensional map. The second assumption is that threats include anti-aircraft guns, missiles, radars, etc are all described as a circular area and all threats are measured by a uniform threat level. And the third assumption is that route pre-planning only considers known static threats and the dynamic threats operate according to a certain motion law and can be detected in real time, and then can be updated in the map.

The sparse A* algorithm[8] is an optimize of the A* heuristic search algorithm. The basic idea is similar to the standard A* algorithm. First, the planning space is expressed in a grid form, and then optimal route is searched by a predetermined cost function or heuristic function. When searching for the optimal route, the cost of each reachable mesh node relative to the current position is calculated, the node with the least cost is added to the search space, and then more nodes are expanded with the least cost node.

In the process of node expansion, the sparse A* algorithm fully considers the constraints of the maximum corner angle, maximum climb/slip angle and minimum route step of the planning object and extends only to some node units in the neighborhood that satisfy the relevant constraints. The route planning process based on sparse A* algorithm is shown in Figure 1.

![Figure 1. Route planning system.](image)

In the process of node expansion, the sparse A* algorithm fully considers the constraints of the maximum corner angle, maximum climb/slip angle and minimum route step of the planning object and extends only to some node units in the neighborhood that satisfy the relevant constraints. The route planning process based on sparse A* algorithm is shown in Figure 1.

Optimizing the node expansion strategy and the node table update strategy in the sparse A* algorithm can further improve the efficiency.

In the planning strategy, the idea of stratified planning is adopted. The idea is solving the main
problem quickly firstly and then considering the global optimal detailed optimization. The first level is a rough plan and the second level is a fine plan. Then according to the above algorithm flow, the detection of dynamic obstacles can be found in time and obstacles avoidance can be performed[9].

**Optimized Methods**

**Sparse A* Algorithm Node Strategy**

In the node table update strategy, node arrange detection is added and if the new extended node is dominated by the existing node, it is not reserved; if the new code dominates the existing node, it replaces the existing controlled node; if the new node and all existing nodes reserved if they are not dominated. Through the node dominance detection, the sparse A* algorithm no longer extends the useless nodes, improving the efficiency of the algorithm.

In the node expansion strategy, in order to obtain the nodes of the same height as the planned target point in the search point in the search process, when the heights of the target points are not the same, if the height of the target points is within the current reachable range, then the extension is performed. Based on the original high node, adding a node that can reach the height of the target point and regarding it as a point on the same height as, so we can expand the node with the same height as the target point in the space search process and simplify it on a 2D map. Then we can simplify the judgement of subsequent convergence conditions.

**Rough Planning**

Although the efficiency of the sparse A* algorithm for track planning is significantly improved compared to the standard A* algorithm. However, when the unmanned aerial vehicle is planning the flight route, it has higher requirements on the real-time performance of the algorithm because of the large-scale map, all kinds of target task and so on. The sparse A* algorithm may fall into the phenomenon of local search or dimensional explosion. The improvement of the planning algorithm alone cannot effectively meet the requirements of the demands of real-time. Therefore, the hierarchical planning idea can be adopted on the basis of the sparse A* algorithm.

Firstly, a rough trajectory plan is quickly solved with the entire map and the threat idea, that is, having a point set of the route which is planned by using the large-step sparse A* algorithm. Due to the use of large step algorithm to plan the basic route, the speed of the rough plan is much better than the standard sparse A* algorithm plan. The step size of the large step is determined by the size of the map to be planned and the case of the threat area. The various constrains in the route planning process are not considered for the time being in the rough planning process.

The rough plan mainly solves the problem that the algorithm falls into local search and dimension explosion[10]. In the process of route planning, planning for small step for all threat regions will reduce the efficiency of the entire algorithm. And there is not much difference in the real-time, the accuracy of the planned route and the security compared to the algorithm with small step. Therefore, the rough plan can better solve the problem of falling into the local search and dimension explosion. And it can also improve the efficiency of the algorithm.

**Fine Planning**

For the route planning algorithm, various constraints need to be considered in the planning process. When the constraint is added to the search space, the invalid points can be effectively eliminated. This paper mainly considers two constraints, the minimum turning radius constraint and the maximum turning angle constraint[11][12]. The turning radius is the three consecutive nodes generated in the search process in Figure 2. The condition that A3 can be extended is:

\[ R < r_{\text{min}} \]

\[ r_{\text{min}} \] is the minimum turning radius.
The condition that can be extended is that the maximum turning angle is constrained by the performance limitations of the unmanned aerial vehicle in actual flight. And the planned route needs to avoid a large turn corner to ensure that the route is reasonable and feasible. Assuming that the maximum turning angle constraint can be expressed as:

$$\theta_i < \theta_{\text{max}}$$

$$\theta_i$$ indicates the turning angle at the i-th route point. \(i = 1,2,...,n\).

The route generated during the rough planning process may have a larger turning angle and a smaller turning radius due to the use of a larger step size. And the planned trajectory of some parts of the route may not be smooth enough. In response to this situation, we need to optimize the large step strategy to deal with the situation. This paper proposes the fine planning, that is small-step sparse A* algorithm. The fine plan takes the point set of the route segment which is not in conformity with the constraint requirements as the starting point and the end point of the plan, then planning the fine route between the two points. The fine plan considers the constraints of the track, adjusting and modifies the unfeasible route in the rough plan. It ensures the feasibility of the final route and improves the advantages of the route on the basis of rough plan. The fine planning is on the basis of the rough plan, so it does not need to re-plan all the existing route. The planning process is relatively simple and it will not fall into the local search and latitude explosion again.

The fine planning is that the large-step route planning obtained by the rough plan combined with the small-step planning. It ensures the feasibility and advantages of the planned track planning method can be improved. Thereby we obtain an optimal planning track with more symbol requirements.

**Dynamic Route Planning**

The dynamic obstacles do not appear in the map when planning the route. The unmanned aerial vehicles encounter such obstacles during flight and need to be able to perform real-time obstacle avoidance to ensure flight safety.

In the case of dynamic obstacles, we also require algorithm that has a high real-time. The system in this paper will detect whether there is a dynamic obstacle during flight. If it does not detect a new obstacle, it will fly according to the original route. If a new obstacle is detected, it is updated to the map according to the threat information and the algorithm of the dynamic route path planning is used. If the new threat area affects the original route, we select the point of the affected part of the track as the starting point and the target point. The small-step sparse A* algorithm is used to re-plan the route. Since only a part of the route is re-planned, high real-time performance can be ensured. The dynamic obstacle avoidance can be completed, so that the unmanned aerial vehicle can accurately complete the reservation tasks.
Experimental Result

In order to verify that the proposed algorithm can solve the problems faced by the unmanned aerial vehicle flight path planning, the typical proposed mission is simulated using the proposed algorithm. The trajectory planned by the algorithm is required to avoid the threat on the map and reach the target accurately.

The simulation platform uses the Intel-core i5-200 @ 2.20hz processor and the simulation environment is ROS environment under Linux. The simulation of the unmanned aerial vehicle route dynamic planning is carried out in a map environment of 10km*12km.

We select three points on the map to compare the route results of the original algorithm and modified algorithm. And we calculate the time that it takes to plan a new path.

Sparse A* Algorithm Result and Runtime

Based on the original sparse A* algorithm, we improve the real-time performance of the whole algorithm by optimizing the node expansion strategy and update strategy to meet the requirements in practical applications. As shown in Figure 3 and Figure 4, the optimized sparse A* algorithm has better real-time performance. And the number of nodes that we used is shown in Table I.

![Figure 3](image1.png)  
Figure 3. Route result comparison of target point is (2500,6600).

![Figure 4](image2.png)  
Figure 4. Route result comparison of target point is (9300,11000).

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>(3000,3000)</th>
<th>(2500,6600)</th>
<th>(9300,11000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original algorithm</td>
<td>14855</td>
<td>23901</td>
<td>57421</td>
</tr>
<tr>
<td>Optimized algorithm</td>
<td>8917</td>
<td>19463</td>
<td>40950</td>
</tr>
</tbody>
</table>

Hierarchical Strategy Result and Runtime

We use a layered strategy that uses a combination of large step size and small step size to further improve the real-time and robustness performance of the algorithm of the algorithm. We still select
the three target points above to get the route result and calculate the time consumption. Figure 5 shows that the route result of the target point of (2500,6600) and (9300,11000). In the experiment, the large step is 10 and the small step is 1.

Figure 5. Route result of target point is (2500,6600) and (9300,11000).

Figure 5 shows that the accuracy of the route is acceptable. The planned path can avoid threats in the map and reach the target point safely. We also calculate the time consumption of the hierarchical strategy. Table II shows the comparison of the runtime.

Table II. Planning Time Comparison.

<table>
<thead>
<tr>
<th>Planning time (s)</th>
<th>(3000,3000)</th>
<th>(2500,6600)</th>
<th>(9300,11000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original algorithm</td>
<td>0.146</td>
<td>7.454</td>
<td>536.284</td>
</tr>
<tr>
<td>Optimized algorithm</td>
<td>0.075</td>
<td>12.181</td>
<td>432.131</td>
</tr>
<tr>
<td>Hierarchical algorithm</td>
<td>0.024</td>
<td>1.19</td>
<td>35.905</td>
</tr>
</tbody>
</table>

Table II shows that the hierarchical strategy can have a better real-time performance. The speed of the planning path has been greatly improved. And the strategy can guarantee trajectory security with acceptable robustness.

**Dynamic Route Planning Result**

If the unmanned aerial vehicle meets the dynamic obstacles, it is supposed to change the follow-up route. Figure 8 shows the dynamic route planning function.

Figure 6. Pre-planned route and actual dynamic planning route.

Figure 6 shows that when the unmanned aerial vehicle encounter a dynamic obstacle during the flight (the square in the Figure 6), it can modify the previously planned route and successfully avoid the dynamic obstacle. And then it can reach the target point.
Practical Application
One of the best ways to verify the performance of an algorithm is to put the algorithm in practice. We built an entire system on a quadrotor drone to verify the algorithm. We planned the route for the playground and recorded the trajectory. The algorithm preforms well in addition to real-time and robustness.

Conclusion
This paper is aiming at real-time, task adaptability and achievable comprehensive matching requirements of unmanned aerial vehicle path planning based on the sparse A* algorithm. We optimize the node strategy of sparse A* algorithm and construct a route planning strategy based on layered idea. This paper also propose a scheme for solving the route planning of dynamic obstacles to ensure the real-time planning. The performance of UAV dynamic track planning amplification based on hierarchical strategy is verified by simulation. The simulation results show that the proposed algorithm effectively improves the timeliness and robustness of the algorithm, shortens the planning time and has high engineering applicability.

References