Trajectory Planning Algorithms for Unmanned Aerial Vehicles

LIU Zhiping, TAN Fang

College of Computer Science and Engineering, Xian Technological University, Xian 710032
E-mail: lzp_leopard@sina.com

Abstract: in this paper, the comparison of three predominant algorithms of trajectory planning for unmanned aerial vehicles (UAV) was presented, based on the compared results, a new planar and three dimensional trajectory planning architecture and related algorithms were proposed. Furthermore, the simulation scheme was constructed under the MATLAB environment. The simulation results verified the feasibility of the presented algorithms.

Key Words: Trajectory planning, Algorithm, UAV, Simulation

1 Introduction

Unmanned aerial vehicles have been widely used in the military fields for decades, especially in the intelligence, reconnaissance and surveillance [1]. Meanwhile, with the rapid development of automation, computer and communication, the trajectory planning for the UAV has been becoming an urgent mission, because of so much obstacles should be avoided, simultaneously the limitations of the onboard equipments and maneuverability of vehicles should be seriously tackled. UAV has shown their particular importance and flexibility, the critical challenge is limited endurance and autonomy [2]. Up to now, there are three categories of trajectory planning methods proposed for UAVs, namely road map, cell decomposition and potential field [3-8].in addition to above predominant methods, there appear several more complicated methods, such as rapidly exploring random trees, receding horizon control, mixed integer linear programming and mixed integer nonlinear programming.

Although there are extensive literatures about the traditional path planning, most of them are inherited from the robotic route planning and robotic applications. As mentioned above, the road map has been used to construct a continuous path which consists of waypoints and edges. The most popular method of this kind path planning is Voronoi graph, it has been written into MATLAB functions, although the derived path is not smoothly enough to track for UAVs. According to the roadmap of the UAV released by the American department of defense, the trajectory autonomous plan could not be realized until 2015, although the autonomous aerial refueling was presented by Department of defense of USA in 2000.

Aside from the complexity of the planning algorithms, the uncertainties in the environment, limitations of the sensors and vehicle is another important factor, especially the dynamic scenario of the battle field and some urgent accidents. Meanwhile the path planning which are both optimal and dynamically feasible is complicated, because the searching space is extremely large and non-convex, and the model simplifications make the solution losing optimality and feasibility. So the path planning, especially the autonomous, intelligent path planning is becoming one of the hot issues in the field of military and civil application.

This paper is organized as follows. In the section 2, the comparison of the popular planning algorithms was presented. Ignited by the comparison, fusing the merits and demerits of the predominant methods, we propose a new path planning architecture and related algorithm in the section 3, the simulation model construction and simulation result analysis was performed in the section 4, and finally the conclusion and future work recommended was given in the section 5.

2 Comparison of the Predominant Methods

In this section we give detailed review about the three predominant method used for path planning, and then refine some inherit issues for the modification and new algorithm design.

2.1 Rapidly-exploring Random Tree (RRT)

RRT is proposed for exploring high dimensional spaces efficiently and quickly, while these spaces are not conventionally convex. As we all know that most practical optimal algorithms are suitable to convex sets.

The main idea of the RRT can be described in the figure.1, it's a random exploration algorithm that uniformly explores the search space, not limited in the planar. It use environment data to construct a so called tree, edges in the tree are directed from a child node to parent node, each node has exactly one parent, except the root. In the RRT architecture, the nodes represent physical states, and edges denote feasible path between the states. The uniformity is achieved by randomly sampling from a uniform probability distribution. As described in the figure 1, c_{ij} is the cost associated with traversing a feasible path between nodes, the leaf is the current state, which may become a parent node during the tree growing interval.

* This work was supported by the Shaan Xi Natural Special Foundation under Grant 12JK0511.

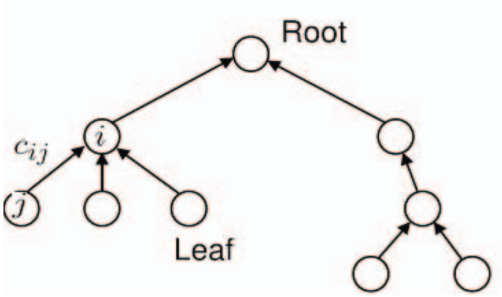

Fig.1 Illustration of RRT

The disadvantage of the RRT is that it usually couldn't give an optimal solution, even far from optimal. From the view of practical usage, it has several advantages:

1) Its computational efficiency is obvious.

2) It could find a feasible solution between the complex obstacles quickly.

3) It inherently accounts for the differential constraints and special constraints.

4) It can be easily modified to meet specific computational restrictions.

5) It could deal with the optimal problems in the non convex region.

The procedure of the traditional RRT can be described as following.

1) A new random point is chosen during each iteration, the choice may be produced through random number generator.

2) The selected random point is examined against current tree of reachable points, a point which meets the nearest criteria is selected.

3) If there is no trajectory can be found, the random point will be discarded, otherwise, an edge of length is added along the feasible trajectory from the closest point to the random point.

4) To check whether the goal is reachable from the new node added to the tree. If it is feasible, a new feasible path is generated. Otherwise step 1 through 4 must be repeated until the maximum steps have been taken.

5) The RRT search may continue to run for a shortest path or low cost path.

2.2 Receding Horizon Control (RHC)

Receding horizon control also referred to as model predictive control (MPC), its basic idea is to use a model of system to predict its future behavior, the illustration of the RHC can be found in figure 2.

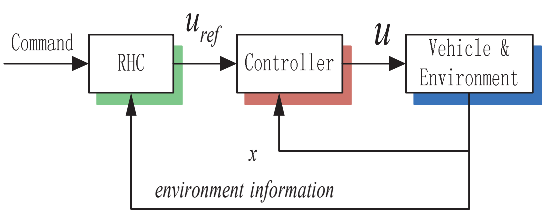

Fig.2 RHC

From the figure 2, we can see that the states and environment information will be feedback to RHC and low level controller to produce real control action and to satisfy various constraints and optimal performance, all uncertainties are included in the environment information.

From the view of the mathematics, the problems which RHC could deal with can be formulated as follow, how to make the vehicle from the current state x_0 along the designed trajectory to the ultimate state x_{final} during flight, meanwhile meeting the following constraints [10].

$$\begin{aligned}
& x_{k+1} = Ax_k + Bu_k \\
& y_k = Cx_k + Du_k \\
& u_i \in \{u : u_{i\min} \le u_i \le u_{i\max}, i = 1,2,\cdots\} \\
& J = \sum_{i=0}^{N} h(x_i, u_i)
\end{aligned} \tag{1}$$

For the reason of simplicity, the system was assumed as linear time invariant, $h(x_i, u_i)$ is a stage cost function.

As mentioned in the RRT, there is also an dynamic optimization problem in the RHC, so the online computation is a key factor to judge whether the real time optimal solution could be obtained.

It's also can be seen from the figure 2 that if the command is changing more fast than vehicle, the solution may beyond the desire. So traditionally the RHC is applied to systems whose time constant is much longer than several minutes.

The advantages of the RHC can be depicted as

1) It could operate near the constrained boundaries and obtain better optimal performance than traditional approaches.

2) It could readily cope with hard constraints, such as relative position tolerance and collision avoidance.

The demerits of RHC can be concluded as

1) Robustness is still a challenge.

2) Computation online is still a heavy burden.

3) Optimization of the solution couldn't be always guaranteed.

The theory and application of RHC for UAVs path planning has been researched and investigated in the Massachusetts Institute of Technology since 2000, both in the robotics and miniature aerial vehicles.

2.3 Voronoi Graph

Voronoi graph is suitable for the scenarios in which the obstacles are relatively small and can be modeled as points.

The typical feature of the Voronoi graph is that the edges of the graph are perpendicular bisectors, following the edges of the Voronoi graph potentially produces a feasible path. Voronoi graph is suitable for the static path planning, that means the obstacles or the threats are known before flight planning. The planned path using Voronoi graph is not smooth enough for UAV to fly.

Traditional algorithm using Voronoi graph is depicted below [3].

1) Input obstacle points Q, start point P_s and end point P_e

2) $(V, E) = constructVoronoiGraph(Q)$

3) $V^+ = V \cup \{P_s\} \cup \{P_e\}$

4) to find $\{v_{1s}, v_{2s}, v_{3s}\}$, three closest points in V to P_s, and $\{v_{1e}, v_{2e}, v_{3e}\}$, three closest points in V to P_e

5) $E^{+} = E \bigcup_{i=1,2,3} (v_{is}, P_s) \bigcup_{i=1,2,3} (v_{ie}, P_e)$

6) For each element $(v_a, v_b) \in E$ do

7) Assign edge cost $J_{ab} = J(v_a, v_b)$

8) End for

9) $W = DijkstraSearch(V^{+}, E^{+}, J)$

10) Return W

Where the cost function $J(v_a, v_b)$ for the edge defined by (v_1, v_2) , its detailed formulation is designed by user.

3 New Trajectory Planning Architecture

Based on the above comparison, we can find insights in the currently used planning algorithms. So in this section, we propose some modified algorithms and make some modifications to the traditional methods.

3.1 Modified RRT

In the numerous literatures, the modified RRT can be found mainly focus on the choice of searching algorithm and adding additional points during the iteration.

Now we present a modified RRT, it has four steps during one iteration interval.

1) Define a search region, this region is constrained by the onboard sensors, during every searching step, this area of region is constant, if and only if the random point was selected and within this region, the next step can be initiated.

2) Generating the next node, meanwhile integrating the dynamical constraints of the vehicle and geographic information.

3) Smoothing the new generated path.

4) Updating the smoothed cost.

It 's depicted in the Fig.3.

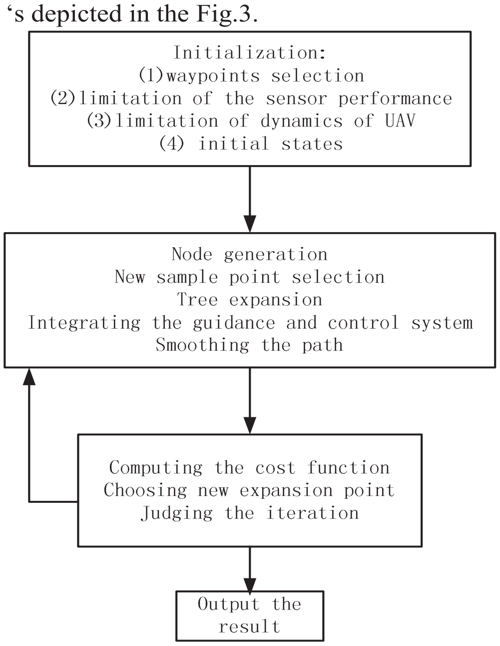

Fig.3 RRT Architecture

From the aforementioned procedure, we can find that the difference lies in the integrating the dynamical constraints in- to the traditional RRT.

In the realistic design and simulation, the guidance and control system of UAVs must be incorporated into the path planner and mission manager system.

The dynamic constraints which should be considered were listed below.

1) overload
2) minimum turn radius
3) flight endurance
4) angle of attack and sideslip angle
5) limitation of the actuators
6) limitation of sensors
7) priori geographical information

Combining the proposed architecture and dynamic constraints, we can see that the function of practical trajectory planning is identical to a typical guidance, navigation and control system.

4 Simulation and Analysis

In this section, the path planning algorithms and the UAV dynamics are taken into consideration simultaneously. Firstly, we give the simulation results of the static path planning in Fig.4 and Fig.5, in this procedure, we assume that the threats are located on ground, they are detection radar system, and geographical positions of radar stations are a prior knowledge for the UAV in the detection field.

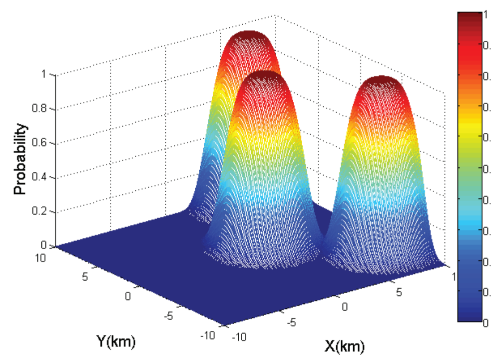

Fig.4 Detection Probability

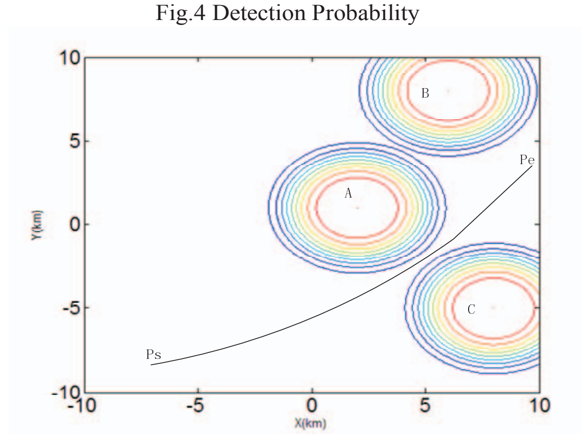

Fig.5 Path Planning Result

In the Fig.4 and Fig.5, three threats locate at A (2.0, 1.0), B (6.0, 8.0) and C (8.0, -5.0), then using the threats model presented in the reference [7], then the simulation results can be found under the MATLAB environment. The planned path can be smooth curves projected on ground, and the detection threshold is defined as 0.01 in the Fig.4 and Fig.5, and it's obviously that the scenario is within the planar.

Secondly, we give dynamic obstacle avoidance planning demonstration, the obstacles were assumed to move in the three dimensional space, meanwhile we consider the dynamics of the vehicle during the path planning, the model of the UAV was referenced to [9].

The movement of the obstacles was assumed within a sphere, the radius of the sphere is two kilometers, the initial position of each obstacle is A (0, 0, 0), B (10,-8, 4) and C (0,-7, 6). When considering the minimum turning radius of the vehicle, the path planning is becoming much complicated and much more time consuming. The Fig.6 illustrated the relatively simple scenario.

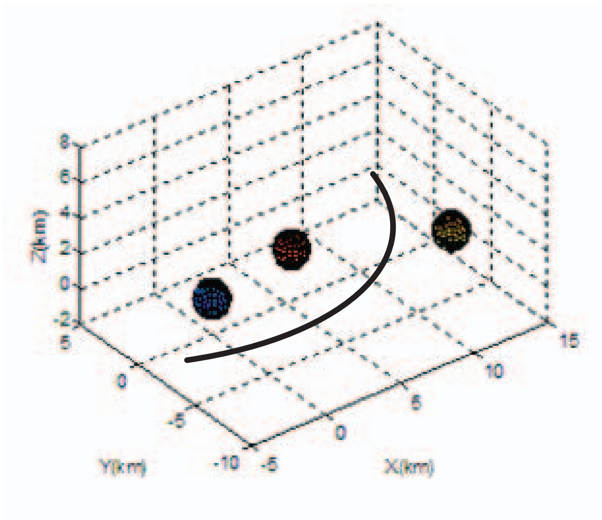

Fig.6 Spatial Path

Furthermore, if we take practical requirements such as overload factor, time consuming and minimal exposure time, we find that all presented methods are unfeasible.

After analyzing the simulation results, we find that following key factors which affect the planning solution and performance.

1) the prior knowledge about obstacles or threats
2) the maneuverability of the vehicle
3) tactical requirements
4) computation burden on the onboard computer

5 Conclusions

The detailed comparison of popular path planning algorithms was presented in this paper, based on the merits and demerits of these algorithms, we add UAV dynamics and dynamic obstacles in the whole path planning scheme. Under the MATLAB environment, we performed the simulation, the simulation results shown that the path planning scheme considering the vehicle dynamics is much more time consuming and challenging in the satisfaction of the various requirements

References

[1] [1] Kim, Y., Gu, D., Real Time Path Planning with Limited Information for Autonomous Unmanned Air Vehicles, Automatica, Vol. 44, 2008, pp. 696-712.

[2] [2] Redding, J., Amin, J.N., and Boskovic, J.D., A Real-time Obstacle Detection and Reactive Path Planning System for Autonomous Small-scale Helicopter, AIAA-2007-6413.

[3] [3] Beard, B.W., McLain, T.W., Navigation, Guidance, and Control of Small Unmanned Aircraft, Brigham Young University, 2009.

[4] [4] Liu, X., Zhou, C.P., and Ding, M.Y., Efficient Path Planning Algorithm for UAV, Journal of Huazhong University of Science and Technology, Vol.39, No.4, 2011,pp: 45-48.

[5] [5] Erazzoli, E., Dahleh, M.A., and Feron, E., Real-time Motion Planning for Agile Autonomous Vehicles, Journal of Guidance, Control, and Dynamics, Vol. 25, No. 1, 2002, pp. 116-129.

[6] [6] Lavalle, S.M., Planning Algorithms, Cambridge University Press, 2006.

[7] [7] Yang, X., Wang, R.M., Research on Autonomous Flying-trace Plan for UAVs, Journal of Computer Engineering, Vol.38, No.5, 2012, pp:192-195.

[8] [8] Chen, H.H., Liu, Y., and Du, Y.L., Path Planning of Unmanned Aerial Vehicle, Journal of Computer Applications, Vol. 31, No.9, 2011, pp: 2574-2577.

[9] [9] Dogan, A., Venkataramanan, S., Nonlinear Control for Reconfiguration of Unmanned-Aerial Vehicle Formation, Journal of Guidance, Control, and Dynamics, 2005, Vol.28, No.4, pp:667-678.

[10] [10] Levine, D.S., Information-rich Path Planning under General Constraints using Rapidly-exploring Random Trees, Thesis of Master Thesis, Massachusetts Institute of Technology, 2010.