UAV path planning based on Improved Rapidly-exploring Random Tree

SUN-Qinpeng[1], LI-Meng[1], WANG-Tianhe[1], ZHAO-Chenpeng[2]

1. College of Electrical Engineering, University of Jinan, Jinan 250022
E-mail: 782122905@qq.com, cse_lim@ujn.edu.cn, 291554836@qq.com

2. School of Control Science and Engineering, Shandong University, Jinan 250100
E-mail: 312493792@qq.com

Abstract: The RRT algorithm can be used for path planning of Unmanned Aerial Vehicle (UAV). However, the efficiency and success rate of the basic RRT algorithm for path planning need to be improved. In view of this problem, this paper proposed an improved RRT algorithm which based on the dynamic p_g value and the dynamic step length, and bidirectional RRT (Bi-RRT) is combined. The effectiveness and advance of the improved RRT algorithm are verified by simulation experiment.

Key Words: Unmanned Aerial Vehicle, Rapidly-exploring Random Tree, Path planning

1 INTRODUCTION

The Unmanned Aerial Vehicle (UAV) has the advantages of low cost, easy operation, low environmental requirements, and excellent performance in some high-risk tasks. So it has been widely used in both military and civilian. The path planning of UAV is based on the premise of satisfying the constraint condition and planning the feasible path from the starting position to the target position according to the task demand. Path planning is an important link in autonomous control of UAV.

Existing path planning methods mainly include ant colony algorithm [1] [2], artificial potential field [3] [4], A* algorithm [5] [6] and so on. Rapidly-exploring Random Tree(RRT)is a path planning method based on spatial sampling which is proposed by S.M. LaValle [7] [8].RRT algorithm leads the search to blank region by random sampling of the state space, so it is not necessary to model the space, which has better optimization efficiency and applicability than other algorithms. Related literature research shows that RRT algorithm can more quickly than A * search algorithm to obtain feasible path [3]. RRT algorithm is also easy to apply to search for high - dimensional search and dynamic obstacle environments [9] [10]. Recent years, many modified versions of RRT have be proposed, such as Bidirectional RRT (Bi-RRT) [11], RRT-GoalBias [12], and improved RRT efficiency using potential task planner [13].

Although RRT has many good features, it does not consider the comprehensive cost of the path in the search process. Nodes extension leads to the inherent randomness of the planned path, which can't get the optimal solution [14]. Meanwhile, the efficiency and success rate of path planning need to be improved.

This work is supported by the project of Shandong province higher educational science and technology program under Grant No. J16LN27, and the Key research and development plan of Shandong Province No.2016CYJS06A01, No.2017CXGC0603.

This paper proposed an improved RRT algorithm which based on the dynamic p_g value and the dynamic step length. The specific improvement of RRT algorithm is introduced in chapter 3, and its effectiveness is verified through simulation experiments in chapter 4.

2 BASIC RRT ALGORITHM

The basic RRT algorithm is to carry out the path planning within the two-value obstacle area. Let's say the task space is C, C_{free} represents the feasible region, C_{obs} means the obstacle area, C_{free} and C_{obs} are subsets of C, and meet the following requirements:

$$C = C_{free} \cup C_{obs}$$

$$C_{free} \cap C_{obs} = \emptyset$$

Initial position $X_{init} \in C_{free}$ and target position $X_{goal} \in C_{free}$. RRT algorithm has probability completeness. As long as the target location and the initial position in the same connected area, it's sure to find the path from the initial position to the target position after enough times of node extension. The specific extension process can be shown in the figure 1.

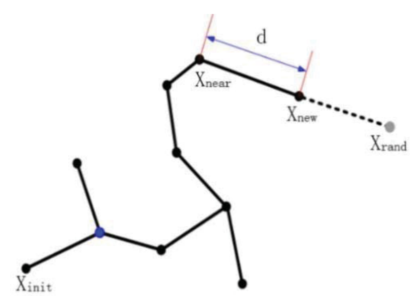

Fig 1. Node extension of RRT

When a new node is extended, the target position is selected as the X_{rand} with probability p_g, or the X_{rand} is randomly selected in the blank area with probability $(1 - p_g)$. RRT grows a step length d from X_{near} to X_{rand} when the new node is extended. The procedure of basic RRT Algorithm is shown below:

1） Algorithm initialization;

2） Whether to reach the target position, not to arrive, turn to step 3), else turn to step 7);

3） Generate a random number $p \in [0,1]$, if $p < p_g$ turns to step 4), else turn to step 5);

4） Select the target position as X_{rand}, determine X_{near}, and then get X_{new}, then turn to step 6);

5） Generate a random node X_{rand} in the search space, and then calculate the candidate new nodes X_{new};

6） To determine whether there is a threat between X_{new} and X_{near}, if there is no threat, add X_{new} to the tree, turn step 2), else turn to step 3);

7） Find the path from the Initial point X_{init} to the target point X_{goal} from the constructed random tree.

3 ALGORITHM IMPROVEMENT

3.1 Bidirectional RRT Algorithm

To improve the efficiency of path planning, the researchers proposed bidirectional RRT (Bi-RRT) [11]. The starting point X_{init} is used as the root node to generate the *Tree1*, and the endpoint X_{goal} is used as the root node to generate the *Tree2*. Double trees are generated in parallel until they are meeting, which means the algorithm converges.

3.2 Dynamic p_g Value

The main part that affects the speed of RRT algorithm is the process of selectingX_{near}. Because this process needs to traverse all nodes of the current tree, and calculate the distance from X_{rand} in order [15]. As a result, the number of random nodes generated by each new node X_{new} is an important factor that affects the speed of the algorithm. Path planning is more likely to fail with the density of the obstacle increasing. The $p_g$value should be reduced in order to increase the randomness of the search, thus improving the success rate of the path planning. N_r is the number of random nodes generated by each new node X_{new}. Adjusting the p_g value according to the N_r value (N_r=1, 2,⋯⋯, n). The p_g value is designed as follows:

$$p_g = p_{gmin} + \left(p_{gmax} - p_{gmin}\right) * \left[\frac{1}{1 + INT\left(\frac{N_r}{N_0}\right)}\right] \quad (1)$$

INT(x) is downward Integer-valued function. The lower and upper limits of p_g are p_{gmin} and p_{gmax} respectively. Between $[p_{gmin}, p_{gmax}]$, the p_g value should be reduced to increase the randomness of the tree growth as N_r increasing. However, it is not reasonable to change the p_g value immediately after the failure of the extension, so we set a threshold N_0(1,2,3⋯n)which is a positive integer .

Table1. The Change Rule of p_g Value

N_r	$INT(N_r/N_0)$	p_g
$[N_0, N_0 - 1]$	0	p_{gmax}
$[N_0, 2N_0 - 1]$	1	$\frac{(p_{gmax} + p_{gmin})}{2}$
$[2N_0, 3N_0 - 1]$	2	$\frac{p_{gmax} + 2p_{gmin}}{3}$
……	……	……
$[nN_0, (n+1)N_0 - 1]$	n	$\frac{[p_{gmax} + (n-1)p_{gmin}]}{n}$

Table 1 shows the change of p_g values with N_r.The value of p_g is going to change when N_r is an integer multiple of N_0. When the N_r value exceeds the threshold N_0, then p_g decreases with the increase of N_r.

3.3 The Dynamic Step Length

The length of the step has an important influence on the number of nodes and the success rate of path planning [15].When the step length is larger, the number of path nodes generated will be less, but the success rate of path planning will decrease. The success rate of path planning will increase when the step length is short, but the number of path nodes will increase. Using the dynamic step can achieve a good balance between efficiency and success rate. In the expansion of the new node, the step length is dynamically adjusted according to the angle α (α ∈ $[0, 180^{\circ}]$)of vector $\overrightarrow{X_{near}X_{goal}}$ and vector$\overrightarrow{X_{near}X_{rand}}$, and the more the direction of growth deviates from the target position, the shorter the step is, and the distance is shortened to some extent.

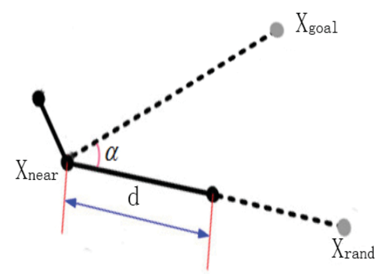

Fig 2. The dynamic step length

The dynamic step of design is as follows:

$$\begin{cases} d = d_{min} + (d_{max} - d_{min}) * \cos\alpha & ,\alpha \le 90^{\circ} \\ d = d_{min} & ,\alpha > 90^{\circ} \end{cases} \quad (2)$$

$$d = 0.618 * d_{min} \quad (3)$$

The lower limit and upper limit of the step length d are respectively d_{min} and d_{max}. When the angle α is less than or equal to 90 degrees, adjusting step length d according to angle α(Formulas (2)); When the angle is greater than 90 degrees, then the length of step is d_{min} (Formulas (2)).The extended rule is that the extension of the new node cannot run into obstacles. If the current step length is consistent with the extended rule, adding the new node to the tree. If there is an obstacle in the way that is not in line with the extended rule, then adjusting the step length to $0.618 *$

d_{min} expanded in this direction(Formulas (3)). The X_{rand} will be selected in the blank area with greater probability when p_g decreases with the increase ofN_r. Then step length is less than d_{max}, the success rate of new node extension is improved because of the shorter step length to some extent. Adjustment of step length can be shown in figure 3.

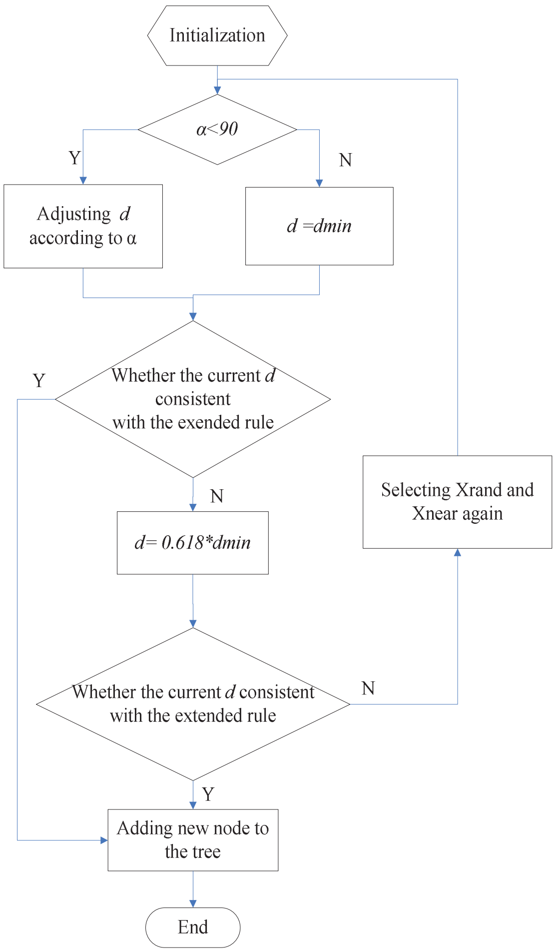

Fig 3. Flow chart of adjusting step length

3.4 The Description of the Improved RRT

First of all, selecting a temporary node X_{rand} , and determining X_{near} . Then adjusting the length of step according to angel α. If current step length consistent with the extended rule, adding the new node X_{new} to Tree1,else adjusting the length of step again. If the adjusted step length is still not satisfied with the extended rule.We need to add 1 to the value of N_r to adjust p_g value, in order to increase the randomness of random tree growth. Based on the above analysis, the improved RRT includes *Tree1* and *Tree2*. The flow chart of *Tree1* is shown in figure 4. The construction of *Tree2* is same as *Tree1*.

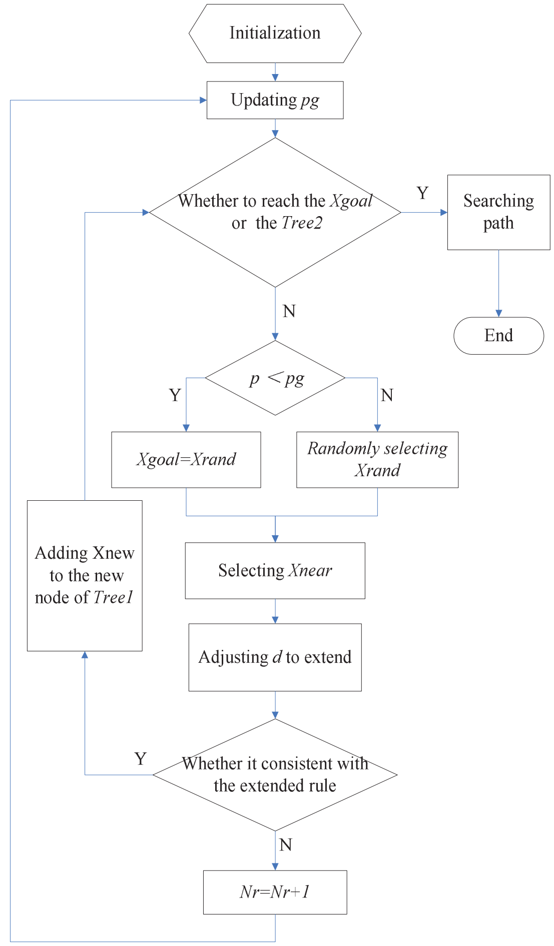

Fig 4. Flow chart of *Tree1*

4 SIMULATION

MATLAB simulation experiments are conducted to verify the improved RRT. The average of 100 simulation experiment is carried out in the environment of different obstacles, and the experimental results are analyzed qualitatively. Setting the mission environment for the dimensionless 100*100 area, the starting position of the UAV as (0, 0), and the target location is (100, 100).

The value of p_g should not be greater than 0.8, which can lead to serious reduction of the planning ability and poor real-time performance. Thinking to take p_g values in the vicinity of 0.5, this can take into account all aspects of the algorithm. RRT can get more excellent planning path and good adaptability to various threats to the environment with shorter operation time. For the basic RRT algorithm, the step length d and p_g value are respectively 7 and 0.5 respectively. For improved RRT, d_{min} and d_{max} are selected as 5 and 10 respectively; p_{gmin} and p_{gmax} are 0.4 and 0.7 respectively.

The computer used in the simulation experiment is as follows: The processor is Intel(R) Core(TM) i3-3217U CPU @ 1.80GHz; The operating system is Windows 7 64-bit operating system.

4.1 Test Result

There are 75 obstacles in the task environment. The sampling of the path planning results of improved RRT are show in the figure 5 and figure 6.

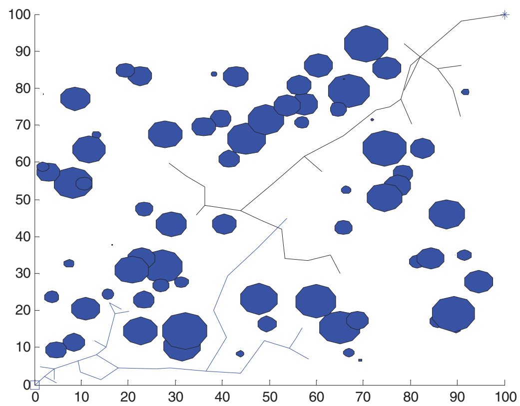

Fig 5. Tree1 and Tree2 of improved RRT

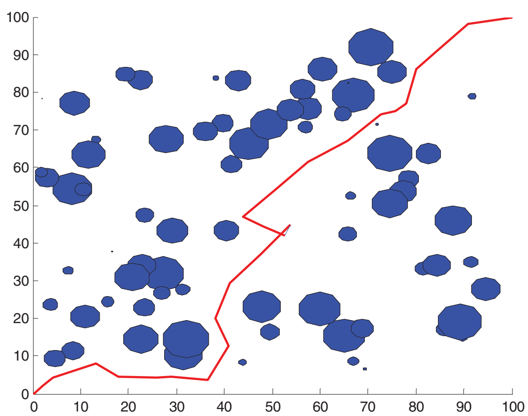

Fig 6. The path of improved RRT

The simulation results show that the improved RRT can realize the path planning. The simulation times are 1.85 seconds, the path length and the numbers of nodes are 190.4 and 62 respectively. Figure 5 shows the growth of tree1 and tree2, and the algorithm converges when the two trees meet. Figure 6 is the path that is eventually obtained.

The change of p_g is shown in figure 7, p_g value is changed between 0.7 and 0.4. The p_g value will decrease to increase the randomness of the random tree growth when the extension has not got the new node many times. The change of step d length is shown in figure 8. The step length is adjusted according to formula (2) and formula (3). Step length is changed between d_{max} and $0.618 * d_{min}$.

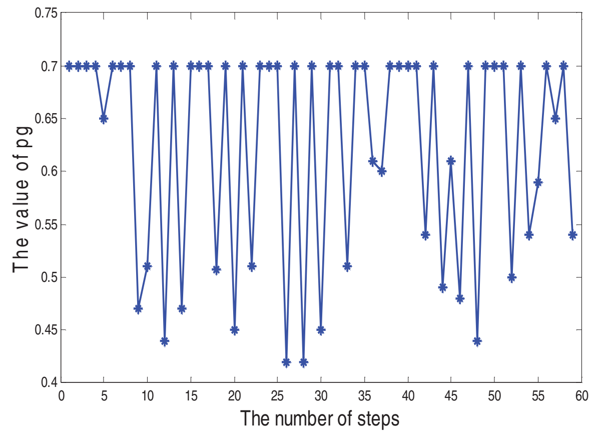

Fig 7. The change of p_g value

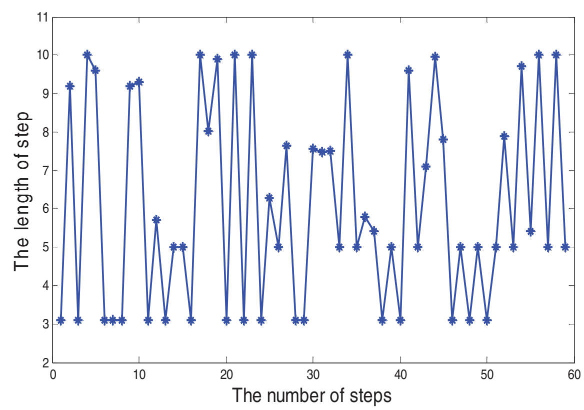

Fig 8. The change of step length

4.2 Data Analysis

Experiments are carried out in environments where the numbers of obstacles are 25, 75 and 100 respectively. 100 maps are randomly generated for each task environment, with the average of 100 paths as the final simulation results. If there is no path generated for 1000 nodes, then path planning fails. In table 2, Ba-RRT is basic RRT, Bi-RRT is bidirectional RRT, and Im-RRT is improved RRT.

Table2. Comparison of Simulation Results

Obstacle	RRT	Time/s	Length	Node	Failure
25	Ba-RRT	0.15	163.07	55.03	0
	Bi-RRT	0.41	167.07	32.73	0
	Im-RRT	0.46	161.50	26.67	0
75	Ba-RRT	1.37	193.16	136.9	8
	Bi-RRT	0.88	192.48	55.24	6
	Im-RRT	1.12	168.18	41.46	3
100	Ba-RRT	2.98	198.88	163.0	16
	Bi-RRT	1.74	197.10	59.47	14
	Im-RRT	1.96	173.20	52.48	6

When the number of obstacles is 25, the time of the improved algorithm has no advantage. The path length of improved RRT is 5.57 less than bidirectional RRT, and the total number of nodes in improved RRT is 18.52% less than bidirectional RRT. When the number of obstacles is 75, the improved RRT planning path is 0.25 seconds faster than basic RRT. Comparing with bidirectional RRT, the path length and the total number of nodes decreased by 12.62% and 24.95% respectively, with the failure rate dropped from 6% to 3%. When the number of obstacles is 100, the time of the improved RRT planning path decreased by 1.02 seconds than basic RRT. Comparing with bidirectional RRT, the path length decreased by 12.13%, the number of nodes decreased by 11.75%, and the failure rate decreased by 6%.

The improved RRT is faster than basic RRT to plan a path when obstacles density is large. It is shown that the improved RRT algorithm is more effective for UAV path planning. The path of improved RRT is shorter than before. The range of UAV is limited, and a shorter path is more conducive to the implementation of its missions. The improved RRT significantly reduces the total number of nodes, which greatly improves the efficiency of the path search. When there are more obstacles, the success rate of path planning will be reduced when the density of obstacles increases. Obviously, compared with basic RRT and bidirectional RRT, the improved RRT has less failure.

5 CONCLUSION

According to the RRT algorithm in the path planning of UAV application, an improved RRT algorithm is proposed. Step length can be dynamically adjusted according to angle and constraint rules, and combines bidirectional RRT (Bi-RRT) and dynamic p_g various. The simulation results show that the improved RRT algorithm can greatly improve the efficiency and the success rate of the path planning, planning time, path length，especially in complex environment. However, the algorithm is still only using computer simulation experiments, because the flight environment is also complex and changeable, it needs further verification and analysis in practical applications.

REFERENCES

[1] F. H. Jin, H. J. Gao, X. J. Zhong, Research on path planning of robot using adaptive ant colony system, Journal of Harbin Institute of Technology, Vol.42, No.7, 1014-1018, 2010.

[2] C. Z. Sun, G. S. Gui, D. Han, L. Zhang, Algorithm for mobile robot path planning based on the ant colony algorithm and its application, Journal of Hefei University of Technology, Vol.29, No.10, 1208-1211, 2006.

[3] Z. Z. Yu, J. H. Yan, J. Zhao, Z. F. Chen, Y. H. Zhu, Mobile robot path planning based on improved artificial potential field method, Journal of Harbin Institute of Technology, Vol.43, No.1, 349-354, 2011.

[4] J. Y. Zhang, Z. P. Zhao, D. Liu, A path planning method for mobile robot based on artificial potential field, Journal of Harbin Institute of Technology, Vol.38, No.8, 1306-1309, 2006.

[5] D. Brandt, Comparison of A and RRT-Connect Motion Planning Techniques for Self-Reconfiguration Planning, International Conference on Intelligent Robots and Systems, 892-897, 2007.

[6] F. Ma, H. S. Yang, Q. Gu, Y. Meng, Navigation path planning of unmanned underground LHD based on improved A* algorithm, Transactions of the Chinese Society of Agricultural Machinery, Vol.46, No.7, 303-309, 2015.

[7] S. M. Lavalle, Rapidly-Exploring Random Trees: A New Tool for Path Planning, Algorithmic & Computational Robotics New Directions, 293--308,1998.

[8] S. M. Lavalle, J. J. Kuffner, Rapidly-Exploring Random Trees: Progress and Prospects, Algorithmic & Computational Robotics New Directions, 293--308, 2000.

[9] J. Bruce, M. Veloso, Real-time randomized path planning for robot navigation, International Conference on Intelligent Robots and Systems, IEEE, 2383-2388, 2002.

[10] J. Lee, C. Pippin, T. Balch, Cost based planning with RRT in outdoor environments, International Conference on Intelligent Robots and Systems, IEEE, 684-689, 2008.

[11] X. Wu, C. Guo, Y. Li, Variable probability based bidirectional RRT algorithm for UAV path planning, Chinese Control and Decision Conference, 2217-2222, 2014.

[12] H. B. Tang, Z. Q. Sun, Parameter adaptive RRT-GoalBias algorithm, Dynamics of Continuous Discrete and Impulsive Systems-series B-Applications & Algorithms, 381-386, 2005.

[13] I. Garcia, J. P. How, Improving the Efficiency of Rapidly-exploring Random Trees Using a Potential Function Planner, Decision and Control, 2005 and 2005 European Control Conference, IEEE, 7965-7970, 2012.

[14] G. Y. Yin, S. L. Zhou, Q. P. Wu, An Improved RRT Algorithm for UAV Path Planning, Acta Electronica Sinica, Vol.45, No.7, 2017.

[15] M. Li, Q. Song, Q. J. Zhao. Experimental Study of Parameters for Rapidly-exploring Random Tree Algorithm. Proceedings of the 8th International Conference on Computer Modeling and Simulation 19-23, 2017.