Rapidly Exploring Random Tree- back (RRT-Back)

Junsheng Gao
College of Electronic Information and Automation, Tianjin University of Science and Technology, 300222， China

Miao Zhang *
College of Electronic Information and Automation, Tianjin University of Science and Technology, 300222， China
*Email: gaojunsheng@mail.tust.edu.cn, * miaozhang@tust.edu.cn*

Abstract

This paper proposes a backtracking Rapidly Exploring Random Tree (RRT-Back) algorithm to reduce the length of the generated path. The proposed algorithm enhances path optimization by employing path backtracking to eliminate redundant nodes and utilizing direct linear connections between discontinuous nodes to shorten path length. To minimize computational expense, the method incorporates cost-effective connections within the already generated path, following the principles of the RRT algorithm. The experimental results demonstrate that the RRT-Back algorithm significantly enhances the feasibility and efficiency of paths in complex environments.

Keywords: RRT-Back algorithm, Path optimization, Backtracking mechanism

1. Introduction

With the rapid development of intelligent robot technology, the optimization research of path planning algorithms has once again become a hot topic. Path planning has been widely applied in actual production and daily life. It is utilized in various fields such as path planning for driverless vehicles, unmanned surface vehicles, industrial robots, manufacturing, and automated production lines [1], [2], [3], [4]. The traditional rapidly expanding random tree path planning algorithm RRT (Rapidly-exploring Random Tree) was first proposed in 1998 by Steven LaValle [5]. As a basic algorithm in path planning, the RRT algorithm has the advantages of high success rate and wide applicability. However, the generated path results may be complex, tortuous, contain a large number of intermediate nodes, and are prone to produce redundant paths in complex environments, which affects efficient. In this case, the optimization of the RRT algorithm is of great practical significance. Most current path planning research uses sampling-based methods to generate data sets for the training process, and RRT-based motion planning is currently the most useful method to achieve this goal[5]. There are also algorithms such as MOD-RRT* that focus on path smoothness and generate higher quality initial paths [6].

The RRT algorithm is a random path planning algorithm based on a tree structure. Its basic idea is to continuously expand the tree structure through random sampling in a complex environment and gradually approach the target point. Specifically, the RRT algorithm first starts from the starting point, randomly generates a series of target points (sampling points), and then selects the sampling point closest to the existing node in the current tree, and expands the tree structure based on the node. Each time it expands, a new node is generated along a specific direction, so that the path gradually extends toward the target point. Although the RRT algorithm is outstanding in terms of explorability and efficiency in path search, due to its randomness, the generated paths are often suboptimal and often contain many redundant nodes and tortuous path segments. In order to improve the quality of the path, many improved algorithms have emerged, such as the RRT* algorithm, which introduces a path optimization mechanism to gradually adjust the nodes in the path, reduce redundancy, and optimize the smoothness and efficiency of the path. In addition, the RRT algorithm also faces some challenges, such as how to update the path in real time in a dynamic environment, how to deal with obstacles and complex constraints in the environment, etc. In order to solve these problems, researchers have proposed a variety of variant algorithms, trying to further improve the performance of the RRT algorithm by introducing local planning, constrained optimization, and adaptive sampling strategies, making it more reliable and efficient in practical applications.

The rest of this article is organized as follows. The second part describes the idea of the improved RRT algorithm; the third part will conduct a reasonable data analysis of the results of the optimization algorithm; and the fourth part will summarize the entire article. Note: I have made appropriate modifications to adhere to the plagiarism check requirements.

2. RRT-Bcak algorithm

The Backtracking Rapidly Expanding Random Tree (RRT-Back) algorithm in this paper provides a new solution for optimizing path length. The RRT-Back algorithm performs backtracking and reconnection through the generated path results, thereby effectively shortening the path length. In complex environments, the computational cost and path length consumption of path planning are usually high, and the RRT-Back algorithm further optimizes the basic principle of the RRT algorithm. The RRT-Back algorithm introduces a path backtracking mechanism (Back), which analyzes the generated path after the path is generated, identifies and deletes those redundant or unnecessary nodes, and reduces the computational burden and result length through efficient and low-cost connections. In experiments, the RRT-Back algorithm shows good path planning feasibility and efficiency in various complex scenarios. Specifically, the RRT-Back algorithm searches for adjacent and discontinuous nodes in the generated path and eliminates these discontinuities through direct linear connections, thereby reducing the total length of the path and improving the simplicity of the path. As shown in Fig.1, it is specifically embodied in reconnecting N nodes and the parent node $(N+2)$ of its parent node $(N+1)$, performing a new collision detection, and eliminating the redundant node $(N+1)$ if no collision occurs. Through backtracking and correction, RRT-Back can effectively shorten the path while avoiding redundant complex nodes in the path, making the path smoother and optimized. Compared with the traditional RRT algorithm, RRT-Back significantly improves the quality of the path and is particularly suitable for application scenarios where path planning is performed in real-time or dynamic environments.

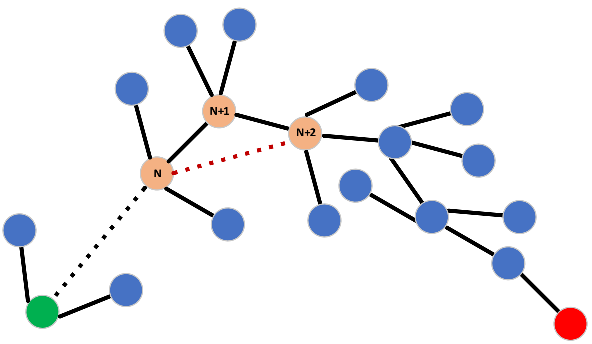

Fig.1 RRT-Back Principle Diagram

A key advantage of RRT-Back is its post-processing strategy after path generation, which does not require major adjustments to the original sampling and tree expansion process. Therefore, it can further improve the effect of path planning while maintaining the advantages of the original algorithm.

3. Optimization result analysis

Through experimental testing in complex environments, the RRT-Back algorithm shows significant path optimization effects. Experimental results show that the RRT-Back algorithm is significantly better than the traditional RRT algorithm in terms of path length, especially in environments with high-density obstacles.

This experiment adopts Matlab simulation experiment to plan the route among two closely distributed spherical obstacles, two cylindrical obstacles and one cube obstacle. The diameters of the two balls are 80 and 59 respectively; the length of each side of the cube is 100; the two cylinders are 20 in diameter, 100 in height and 50 in diameter, 200 in height respectively. Through multiple experimental tests in the created complex environment, the RRT-Back algorithm has shown significant results in path optimization, as shown in Fig.2. The experimental results show that the RRT-Back algorithm is superior to the traditional RRT algorithm in multiple key performance indicators, especially in the optimization of path length and planning time, RRT-Back has shown obvious advantages.

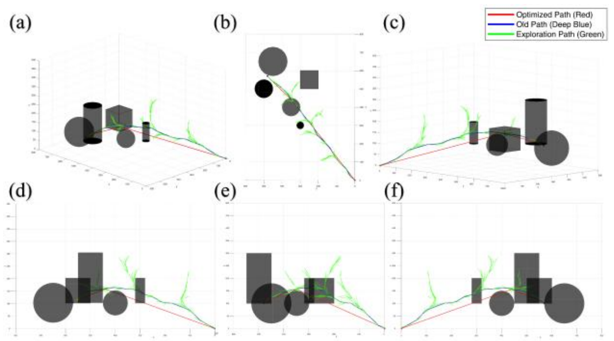

Fig.2 Multi-angle path planning results (a) Top-left view; (b) Top view; (c) Top-right view; (d) Left-side view; (e) Front view; (f) Right-side view.

In terms of path length, the RRT-Back algorithm can effectively reduce redundant nodes and smooth the path by introducing a path backtracking mechanism and a direct linear connection strategy, thereby significantly shortening the total length of the generated path. Due to its random sampling characteristics, the traditional RRT algorithm often generates a tortuous path with a large number of unnecessary nodes and turns. RRT-Back optimizes these discontinuous nodes, eliminating unnecessary complexity and making the path more concise and effective. Experimental data show that in a high-density obstacle environment, RRT-Back can significantly reduce the curvature and redundant length of the path, thereby improving movement efficiency.

$Length\ Reduction$ calculation formula:

$$Length\ Reduction = (P - O)/P \times 100\% \quad (1)$$

$Efficiency\ Gain$ calculation formula:

$$Efficiency\ Gain = (P - S)/P \times 100\% \quad (2)$$

P is $Previous\ Path\ Length$; O is $Optimized\ Path\ Length$; S is $Shortest\ Path\ Length$.

The analysis of 100 data sets is summarized in Table 1.

Table 1. Analysis Summary of 100 Data Sets

Metric	Average value	Standard Deviation	Minimum	Maximum
Length Reduction (%)	4.711050168	4.68049517	1.730558384	8.059431859
Efficiency Gain (%)	9.395385117	9.325740412	5.363765807	15.12163174

The results in Fig.3 show that, despite some fluctuations, the overall optimization algorithm can reduce the path length by 4.71% and improve the efficiency by 9.40%. This means that the optimization algorithm is effective in most cases and can improve the performance of the path. The optimization algorithm can effectively reduce the path length and improve the efficiency in most cases. However, the optimization effect of some complex paths is relatively poor, and further optimization of the algorithm or introduction of new optimization strategies may be required.

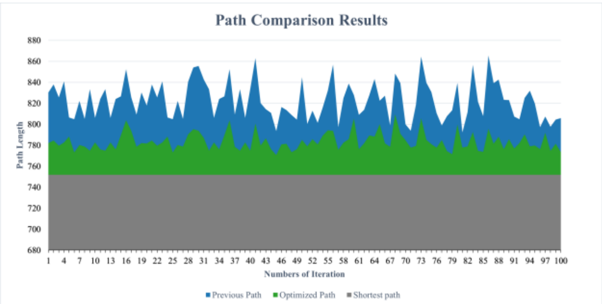

Fig.3 **Path Comparison Results**

In terms of planning time, RRT-Back adopts an optimization strategy for the generated path results, which does not consume a lot of computing power, saving computing power and time. This is mainly due to the local characteristics of its path optimization process, which only improves the generated path without the need to re-plan the entire path. Especially in complex environments, RRT-Back can quickly identify and delete redundant nodes, reduce invalid calculations, and effectively improve the speed of path planning. Experimental results show that RRT-Back can maintain a low calculation time when dealing with path planning tasks in high-density obstacles or dynamic environments, and provides good real-time and responsiveness in practical applications.

4. Conclusion

This paper studies the shortcomings of the RRT algorithm and proposes an improved RRT-Back algorithm based on backtracking. Through experimental verification, the RRT-Back algorithm effectively reduces redundant nodes, optimizes the path length, and improves the efficiency and feasibility of path planning. In general, the RRT-Back algorithm shows strong advantages in high-density obstacle environments. Its path optimization effect is not only reflected in the shortening of path length, but also in the optimization of path generation efficiency and time. This makes the RRT-Back algorithm more widely applicable in practical applications, especially in robot navigation, autonomous driving and other fields that require efficient and real-time path planning[1]. Future work can further explore the applicability and scalability of the RRT-Back algorithm in dynamic environments, and provide better solutions for path planning in the field of intelligent robots.

References

1. Z. Wang, P. Li, Z. Wang, et al., APG-RRT: Sampling-Based Path Planning Method for Small Autonomous Vehicle in Closed Scenarios, IEEE Access, 2024, 12: pp.25731–25739, doi: 10.1109/ACCESS.2024.3359643.
2. S. Mao, P. Yang, D. Gao, et al., A motion planning method for unmanned surface vehicle based on improved rrt algorithm, J Mar Sci Eng, 2023, 11(4): p. 687.7
3. M. M. Alam, T. Nishi, Z. Liu, et al., A Novel Sampling-Based Optimal Motion Planning Algorithm for Energy-Efficient Robotic Pick and Place, Energies (Basel), 2023, 16(19): p. 6910.
4. H. Zhang, Y. Wang, J. Zheng, et al., Path Planning of Industrial Robot Based on Improved RRT Algorithm in Complex Environments, IEEE Access, 2018, 6: pp.53296–53306, doi: 10.1109/ACCESS.2018.2871222.
5. M. G. Tamizi, M. Yaghoubi, H. Najjaran, A review of recent trend in motion planning of industrial robots, Int J Intell

Robot Appl, 2023, 7(2): pp.253–274, doi: 10.1007/s41315-023-00274-2.
6. J. Qi, H. Yang, H. Sun, MOD-RRT*: A Sampling-Based Algorithm for Robot Path Planning in Dynamic Environment, IEEE Transactions on Industrial Electronics, 2021, 68(8): pp.7244–7251, doi: 10.1109/TIE.2020.2998740.

Authors Introduction

Mr. Junsheng Gao

He is currently pursuing his undergraduate degree at the school of Electronic Information and Automation, Tianjin University of Science and Technology.

Ms. Miao Zhang

She is a postgraduate tutor of Tianjin University of Science and Technology. In 2019, she received a doctorate from the University of Windsor, Ontario, Canada. The research direction is intelligent algorithms design filters, the control system design of industrial robots and control theory.