LCFP-RRT : A Robot Exploration Algorithm Based on Local Constrained Sampling and Frontier Prioritization Classification

Hao Zhang1[0009-0003-4392-7447] , Lyu guangqiang2[0009-0009-7014-3131], Xie Yanjing3[0009-0000-1725-6270], Fangyan Dong1[0000-0001-5958-5734] ,Kewei Chen1*[0000-0002-3106-4336]

1Faculty of Mechanical Engineering & Mechanics,Ningbo University,Ningbo 315211,China
{2211090211,2211090234,dongfangyan,chenkewei} @nbu.edu.cn
*Corresponding author: Kewei Chen, chenkewei@nbu.edu.cn
2Zhejiang University Ningbo Institute of Technology, Ningbo 315000, China
guangqiang@nbt.edu.cn
3Ningbo Guangqiang Robot Technology Co.Ltd, Ningbo 315100, China
chaosphd@hotmail.com

Abstract. In order to improve the robot's exploration efficiency of the unknown environment, this paper proposes a robot autonomous exploration algorithm based on local restricted sampling and frontier priority classification, which makes the local tree "short-sighted" by restricting the sampling range of the local tree, so that the robot can focus on exploring the vicinity of the direction of travel, avoiding repeated sampling of the explored area and improving the detection efficiency. This avoids repeated sampling of the explored area, improves the detection efficiency, and avoids frequent steering of the robot. Based on the above sampling method, by prioritizing the classification of the frontiers detected by the local tree and the global tree, the priority of the frontiers detected by the local tree is increased to ensure that the robot gives priority to exploring the frontier region detected by the local tree, and then explores the frontier detected by the global tree when the local tree fails to detect the frontier, which ensures that the unknown environment in the vicinity of the robot is sufficiently explored before exploring the more distant unknown environment. Through experimental studies in different typical indoor scenes, the results show that the algorithm proposed in this paper can effectively avoid the backtracking problem of the robot, shorten the exploration time and path, and improve the exploration efficiency, which verify the effectiveness of the algorithm.

Keywords: autonomous robot exploration, local constrained sampling,frontier prioritization classification,rapidly-exploring random tree

1 Introduction

At present, most mobile robots need to construct the environment map of the application scene in advance, however, the drawing of the environment map needs to collect the environment information manually, which is a complicated process and a large amount of workload, in addition, in the actual application, there are also special environments that are inaccessible to personnel or difficult to traverse by human beings, which brings a lot of difficulties in constructing the map artificially. In order to solve this kind of problem, the autonomous exploration technology in which the robot traverses the whole environment through autonomous navigation and realizes map building in the unknown environment without a priori information has become the research focus of scholars at home and abroad. Frontier-based autonomous exploration algorithm is a classic method in the field of robot exploration, which establishes the exploration goal by detecting the boundaries of explored and unexplored regions in the unknown environment, and guides the robot to move towards the unknown region. In order to explore the environment more efficiently, the focus of this type of exploration method is mainly on how to explore and select the frontiers. The method based on Rapidly-exploring Random Tree (RRT) [1] records the top positions of the tree as frontiers. Compared with image-based frontier detection methods [2], [3], this method saves computational resources and improves the extraction speed of the frontiers, especially in larger environments where the frontier selection directly affects the exploration efficiency. Currently, most existing robot exploration strategies focus on how to design information gain models for frontiers to select frontiers with larger gains. However, the robots of these exploration strategies will greedily select frontiers with high information gain, and when they do not detect nearby frontiers in time, the robots will turn to high-gain frontiers far away from them, which will lead to incomplete exploration of the unknown environment at the current location and make the robots backtracking phenomenon.

In order to solve the above problems, this paper proposes a robot autonomous exploration algorithm LCFP-RRT (Local Constrained Sampling and Frontier Prioritization Classification RRT) based on local constrained sampling and frontiers classification, which restricts the sampling range of the local tree in the vicinity of the robot, making the local tree become " short-sighted", so that the robot can focus the detection range in the vicinity of its traveling direction, avoiding repeated sampling of the traveled area and improving the detection efficiency, and at the same time allowing the robot to fully explore the unknown environment in its vicinity, avoiding the backtracking problem caused by incomplete exploration, on the basis of which, the frontiers detected by the local tree and the global tree are prioritized to increase the priority of the frontiers detected by the local tree, so that the robot gives priority to exploring the frontiers detected by the local tree. When a region is fully explored, the local tree does not detect the frontiers, and then the frontiers of the global tree are explored again to make full use of the "short-sightedness" of the local tree and the global field of view of the global tree, and to improve the exploration efficiency.

2 Related Work

Frontier-based exploration methods have been favored by many scholars because of their obvious advantages over other algorithms [4]-[6], and the research topics of frontier-based exploration methods mostly focus on how to extract and select the frontier.

Yamauchi [7] proposed the original frontier-based autonomous exploration algorithm, where the frontier instructs the robot to explore unexplored areas. For frontier extraction, some scholars have used image processing techniques to extract frontiers from map images [8], [9]. This method is more suitable for small-size scenes. When the scene becomes larger, the image processing efficiency of the whole map decreases significantly. Umari et al [10] used Rapidly-exploring Random Tree (RRT) to extract frontiers. RRT can quickly grow to the frontiers of known maps, and the top nodes of the tree at the boundaries are recorded as candidate frontiers. However, due to the randomized nature of the RRT algorithm, the frontiers at the corners of the map cannot be extracted in a timely manner, resulting in backtracking during the exploration process. Therefore, Wu et al [11] proposed a method that combines RRT with image processing. In order to reduce the consumption of computational resources by RRT nodes, Qiao et al [12] proposed a sample-based frontier detection algorithm by changing the growth law of the random tree and removing redundant nodes.

In terms of frontier selection, existing methods are mainly based on utility functions and information theory [13]. Mei et al [14] considered the energy and time loss caused by the robot turning to each frontier and integrated the direction information into the utility function. Bourgault et al [15] proposed a gain function that combines the expected information gain and the expected path length.

Several scholars have proposed enrichment of the gain function and estimation of environmental information based on the geometric continuity of obstacles. Shrestha et al [16] used state-of-the-art generative neural networks to predict the unknown regions of a partially explored map and used the prediction to enhance the exploration in an information-theoretic manner. Bogoslavskyi et al [17] proposed a method for matching out-of-frontier regions with the most similar in the database of the maps and calculates the desired information gain based on the matching results.

3 Method

In this section, the autonomous robot exploration method based on locally restricted sampling and frontier prioritized classification proposed in this paper is described in detail. The method receives sensor data to update the unknown map through the SLAM module, grows global random trees and local random trees in the free region of the known map for frontier detection, clusters the frontiers through the Filter module and obtains the optimal frontier by evaluating the center of clusters through an evaluation function, and finally publishes the frontier to the robot's navigation module through the Assigner module, the navigation module guides the robot to explore autonomously, while the map is updated by the SLAM module. The overall algorithmic framework is referenced from RRT-Exploration [18].

The algorithm in this paper restricts the sampling range of the local random tree to the sector in front of the robot's traveling, so that the frontiers detected by the local random tree are all located in the robot's traveling direction, and at the same time, the frontiers detected by the global tree and the local tree are prioritized and classified, and the frontiers detected by the local tree are preferred to be explored, which avoids frequent steering of the robot and improves the robot's autonomy exploratory efficiency. The overall algorithm framework diagram is shown in **Fig. 1**.

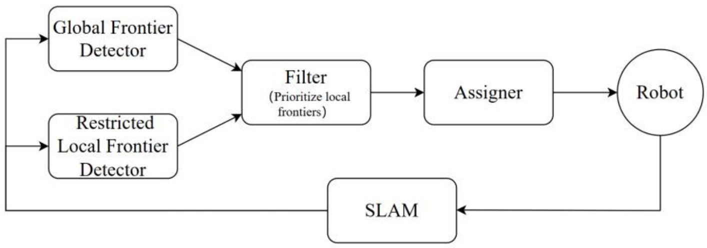


Fig. 1. Algorithm Framework of This Article.

3.1 Local Random Tree Search Strategy based on Locally Restricted Sampling

In order to avoid the robot to miss exploring small corners of the environment and to ensure detection and exploration of unknown environments far away from the robot's current position, the robot needs global random tree, RRT is probabilistic complete algorithm which is able to find all the solutions in the space in sufficient time situation, but as the global tree keeps on growing its growth rate slows down resulting in a reduction in the speed of its detection, so this necessitates the need for local random tree, the local random tree resets when it detects frontier and grows back from the current position of the robot, this speeds up exploration and ensures real time detection of the frontier.

However, the local random tree is sampled randomly around the robot, and the sampling area also includes the area that the robot has already passed through, and at the same time, this sampling method does not take into account the directionality of the robot's movement, which can lead to frequent steering of the robot. To address these problems, this paper makes the general direction of the growth of the local random tree consistent with the direction of the robot's movement, and restricts the node sampling range of the local random tree to the fan-shaped region with the robot position as the center of the fan-shaped circle, which takes the robot's current direction of motion as the symmetry axis and follows the robot's motion to change its position, as shown in **Fig. 2**, where the robot moves from position a to position b, and the search range of frontier is always restricted to the sector-shaped region in front of the robot.By limiting the sampling range of the local tree, the growth direction and range can be focused on the motion direction of the robot, which can substantially improve the detection probability of the frontier, and also make the frontier located in the direction of travel of the robot, avoiding frequent steering of the robot, effectively improving the exploration efficiency. The node sampling coordinates of the localized tree are shown in Eq. (1).

$$\begin{cases} Xr = Px + D * \cos(Ry + A) \\ Yr = Py + D * \sin(Ry + A) \end{cases} \tag{1}$$

In the equations, Xr, Yr are the x and y axis coordinates of the node sampling points of the localized tree, respectively. Px, Py are the x and y axis coordinates of the current position of the robot, respectively. D is a random number with maximum L and minimum 0. Ry is the robot's current stance. A is a random angle with maximum u/2 and minimum -u/2.

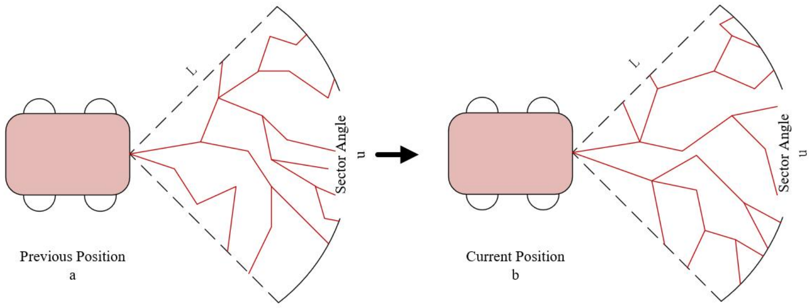


Fig. 2. Local Random Tree Search Strategy.

When the local tree does not detect the frontier, the equation gives the coordinates of a local tree node restricted to the sector range and is added to the current local tree, which grows until the frontier is detected and then reset. In order to adapt to different radar ranges and region sizes, the sector region angle u and its radius L can be changed in size to ensure that the random tree sampling range is adjustable and always larger than the radar range.

3.2 Frontier Prioritization Classification

The original algorithm does not distinguish between the frontiers detected by the global tree and those detected by the local tree, but after limiting the sampling of the local tree, the local tree can more quickly discover the frontiers in its forward direction, which are generally closer to the robot and do not require significant steering of the robot, and are more valuable to explore compared to those detected by the global tree, so the robot should have to prioritize the exploration of the frontiers detected by the local tree.

Accordingly, the frontier prioritization classification strategy is designed, in the filtering module, the frontier detected by the local tree and the global tree is divided, the frontier detected by the local tree is set as the set Frontiers_local, and that of the global tree is Frontiers_global, and priority is given to clustering on Frontiers_local, and the centroids of the clustered set is centroids published to the next module, when the published content is empty, then the local tree at this time did not detect the frontier, at this time on Frontiers_global for clustering and publishing, until Frontiers_local is not empty, at this time on behalf of the local tree to detect the frontier, continue to Frontiers_local for clustering and publishing clustering center point.

Because the local tree is not guaranteed to detect the frontier with every refresh, and if the global tree detects a frontier farther away at this point, this can cause the robot to backtrack, so when the local tree does not detect the frontier, it does not immediately shift to processing Frontiers_global, but instead repeats the detection for a certain number of times until the frontier is detected, and then when the detection exceeds that number of times, it shifts to processing Frontiers_global. The pseudo-code for the classification process is shown in Algorithm 1, where the MeanShift function clusters the set of frontiers and returns the set of centroids for each class.

Algorithm 1 Frontier Prioritization Classification

Input:
Local Tree Detection Frontier Set: Frontiers_local
Global Tree Detection Frontier Set: Frontiers_global
Output:
Set of Clustered Centroids: Centrists

```
count = 0
while true do
    If count > 20 then
        Centrists = MeanShift(Frontiers_global)
         count = 0
    else:
        Centrists = MeanShift(Frontiers_local)
    end if
    If Centrists == 0 then
        count ++
    end if
    return Centrists
 End while
```

4 Experimental Studies and Results

In order to verify the effectiveness of the algorithm proposed in this paper, this section will be verified through simulation experiments. The hardware configuration of the computer used in this experiment is CPU: Intel® Core™ i7-11800H, RAM: 16 GB, and the algorithm is based on the Robot Operating System (ROS), which is compiled and run on the Ubuntu 18.04 operating system.

Table 1. Size of three scenarios.

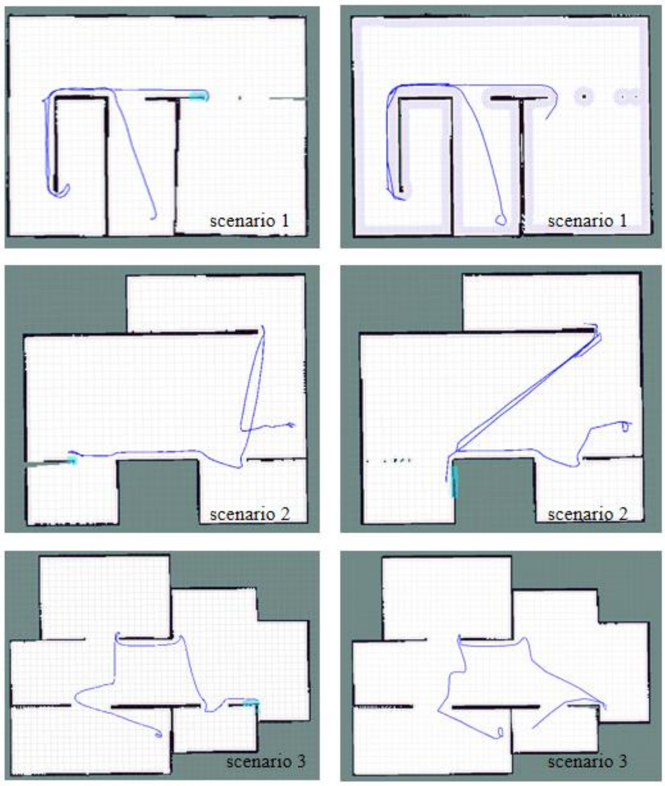
Scenario	*Scenario1*	*Scenario2*	*Scenario3*
Size	$29 \times 39\ m^2$	$31 \times 34\ m^2$	$34 \times 46\ m^2$

The experiments are conducted in three different typical indoor scenarios, the corresponding map sizes of the scenarios are shown in **Table 1**, and scenario 1, 2, and 3 are shown in **Fig. 3**. The LIDAR scanning range in the experiments is set to 180°, and the maximum scanning distance is 15 m. The experiments will run RRT-Exploration and the algorithm proposed in this paper ten times each in each environment, and the exploration time and the exploration path length are taken as the average data of the ten experiments for comparison, and in order to clearly show the exploration trajectories, the best trajectory results of the two algorithms in the three environments are selected to be Displayed.

As shown in **Fig. 3**, the left column shows the exploration trajectories of the proposed algorithm in three environments, and the right column shows the exploration trajectories of the RRT-Exploration algorithm. From the exploration trajectory routes, the RRT-Exploration algorithm has some backtracking phenomena, and the robot will go to explore the unknown environments far away from the robot when the unknown environments in the direction of travel have not been explored completely. This will make the robot turn frequently and produce repetitive routes, while the trajectory of the proposed algorithm does not have large zigzags and repetitive routes.

As shown in **Fig. 4**, the experimental results show that on the scenario 1 map, the proposed algorithm in this paper reduces 13.04% in terms of exploration time and 32.79% in terms of exploration path compared to the original algorithm. For scenario 2 map, this paper's algorithm reduces 38.46% in exploration time compared with the original algorithm, and also reduces 30.69% in exploration path. On the scenario 3 map, the exploration time of this paper's algorithm is reduced by 5.34%, while the exploration path is shortened by 9.33%.

The experimental results show that compared with the RRT-Exploration algorithm, the algorithm in this paper has a large improvement in exploration efficiency. The RRT-based RRT-Exploration exploration algorithm often fails to detect the frontiers in the corners near the robot in a timely manner due to the randomness of the sampling, and the robot will greedily choose the frontiers with the largest gain at each moment, which leads to frequent backtracking of the robot and lower exploration efficiency. The method in this paper, on the other hand, restricts the sampling range of the local tree to the vicinity of the robot's traveling direction and prioritizes the exploration of the frontiers detected by the local tree, while the global tree also ensures the complete exploration of the map, which effectively avoids the backtracking phenomenon and improves the exploration efficiency. In summary, the experiment fully proves the effectiveness of the algorithm in terms of two indicators: exploration time and path length.



(a)LCFP-RRT (b)RRT-Exploration

Fig. 3. Exploratory Trajectories in Three Scenarios.

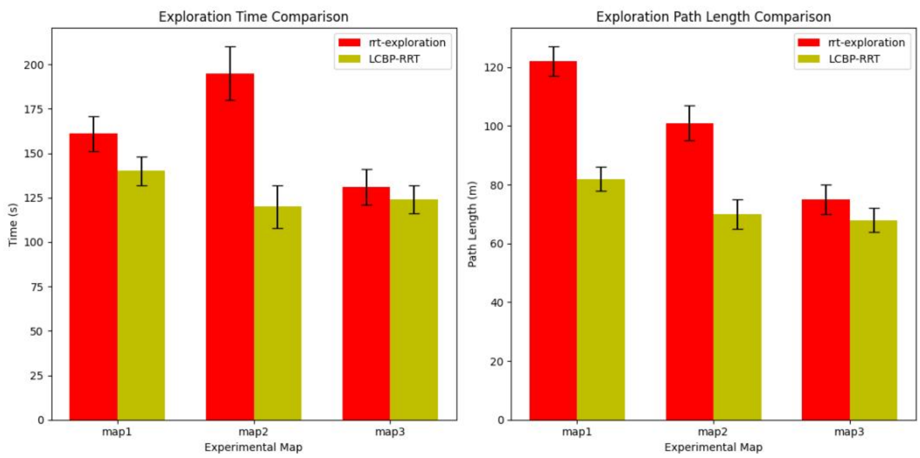


Fig. 4. Results of the Exploration Time and Path Comparison between the Two Algorithms.

5 Conclusion

This paper proposes a robot autonomous exploration algorithm LCFP-RRT based on local restricted sampling and frontier prioritization classification, which is based on the RRT-Exploration algorithm. This algorithm improves the probability of detecting unknown environments in the robot's travelling direction by restricting the growth range of the local random tree to the robot's movement direction, and at the same time prioritizes the local random tree with the global random tree detected by the frontiers are prioritized to increase the exploration priority of the local tree frontiers, so that the robot can fully explore the unknown environment in the vicinity with minimal steering. When the local tree does not detect the frontier by repeating a specific number of times, it means that the unknown environment near the robot is fully explored at this time, and then the frontier detected by the global tree will be explored, and when the frontier is detected by the local tree again, the frontier detected by the local tree will guide the robot again. The local tree is responsible for fully exploring the environment near the robot, and the global tree is responsible for fully exploring the whole map.

Through experimental studies in different typical indoor scenarios, the results show that the algorithm proposed in this paper can effectively avoid the backtracking problem of the robot, shorten the exploration time and exploration path, and improve the exploration efficiency of the robot, which proves the effectiveness of the algorithm.

In the future, the semantic information in the environment can be considered to use the region prediction of the unknown environment, so that the robot can prioritize the exploration of the prediction region, based on this point can also be considered to improve the growth strategy of the global tree, when the robot enters the prediction region, a global tree will be grown in the center of the region, to solve the problem that the local tree can not detect the frontier in time in the complex environment.

References

1. S. M. Lavalle, “Rapidly-exploring random trees : A new tool for path planning,” Computer ence Dept. Oct, vol. 98, 1998.
2. R. C. Gonzalez and R. E. Woods, Digital Image Processing (3rd Edition). Prentice-Hall, Inc., 2007.
3. R. Szeliski, Computer Vision: Algorithms and Applications. Springer-Verlag New York, Inc., 2011.
4. A. A. Makarenko, S. B. Williams, F. Bourgault, and H. F. Durrant Whyte, “An experiment in integrated exploration,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 1, 2002, pp. 534–539 vol.1.
5. González-Baños, Héctor, H., Latombe, and Jean-Claude, “ Navigation strategies for exploring indoor environments.” International Journal of Robotics Research, vol. 21, no. 10-11, pp. 829–848, 2002.
6. W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, “Collaborative multi-robot exploration,” in Proceedings 2000 ICRA. Millennium Conference. IEEE International

Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065), vol. 1, 2000, pp.476–481 vol.1.
7. B. Yamauchi, “A frontier-based approach for autonomous exploration,” Intelligence in Robotics and Automation CIRA’97. ’Towards New Computational Principles for Robotics and Automation’, 1997, pp. 146–151.
8. R. C. Gonzalez and R. E. Woods, Digital Image Processing, 3rd Ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2007.
9. R. Szeliski, Computer Vision: Algorithms and Applications. New York, NY, USA: Springer-Verlag, 2011.
10. H. Umari and S. Mukhopadhyay, “Autonomous robotic exploration based on multiple rapidly-exploring randomized trees,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2017, pp. 1396–1402.
11. C. Wu and H. Lin, “Autonomous mobile robot exploration in unknown indoor environments based on rapidly-exploring random tree,” in Proc. IEEE Int. Conf. Ind. Technol., 2019, pp. 1345–1350.
12. W. Qiao, Z. Fang, and B. Si, “Sample-based frontier detection for autonomous robot exploration,” in Proc. IEEE Int. Conf. Robot. Biomimetics, 2018, pp. 1165–1170.
13. C. Wang, W. Chi, Y. Sun, and M. Q.-H. Meng, “Autonomous robotic exploration by incremental road map construction,” IEEE Trans. Autom. Sci. Eng., vol. 16, no. 4, pp. 1720–1731, Oct. 2019.
14. Y. Mei, Y.-H. Lu, C. Lee, and Y. Hu, “Energy-efficient mobile robot exploration,” in Proc. IEEE Int. Conf. Robot. Autom., 2006, pp. 505–511.
15. F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte, “Information based adaptive robotic exploration,” in IEEE/RSJ International Conference on Intelligent Robots and Systems,vol. 1, 2002, pp. 540–545 vol.1.
16. R. Shrestha, F. Tian, W. Feng, P. Tan, and R. Vaughan, “Learned map prediction for enhanced mobile robot exploration,” in 2019 International Conference on Robotics and Automation (ICRA), 2019, pp. 1197–1204.
17. I. Bogoslavskyi, M. Mazuran, and C. Stachniss, “Robust homing for autonomous robots,” in 2016 IEEE International Conference on Robotics and Automation (ICRA), 2016, pp. 2550–2556.
18. H. Umari and S. Mukhopadhyay, “Autonomous robotic exploration based on multiple rapidly-exploring randomized trees,” in 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS), 2017, pp. 1396–1402.