



Fast autonomous exploration with sparse topological graphs in large-scale environments

Changyun Wei¹ · Jianbin Wu¹ · Yu Xia¹ · Ze Ji²

Received: 9 October 2023 / Accepted: 30 December 2023
© The Author(s) 2024

Abstract

Exploring large-scale environments autonomously poses a significant challenge. As the size of environments increases, the computational cost becomes a hindrance to real-time operation. Additionally, while frontier-based exploration planning provides convenient access to environment frontiers, it suffers from slow global exploration speed. On the other hand, sampling-based methods can effectively explore individual regions but fail to cover the entire environment. To overcome these limitations, we present a hierarchical exploration approach that integrates frontier-based and sampling-based methods. It assesses the informational gain of sampling points by considering the quantity of frontiers in the vicinity, and effectively enhances exploration efficiency by utilizing a utility function that takes account of the direction of advancement for the purpose of selecting targets. To improve the search speed of global topological graph in large-scale environments, this paper introduces a method for constructing a sparse topological graph. It incrementally constructs a three-dimensional sparse topological graph by dynamically capturing the spatial structure of free space through uniform sampling. In various challenging simulated environments, the proposed approach demonstrates comparable exploration performance in comparison with the state-of-the-art approaches. Notably, in terms of computational efficiency, the single iteration time of our approach is less than one-tenth of that required by the recent advances in autonomous exploration.

Keywords Autonomous exploration · Topological graph · Frontier detection · Uniform sampling

1 Introduction

Autonomous exploration Juliá et al. (2012) has emerged as a prominent research topic in the field of robotics, finding extensive applications in areas such as 3D reconstruction Ma and Liu (2018), environmental monitoring, and search and rescue operations. The objective of autonomous exploration is to guide robots in expanding the mapped areas within

unknown environments, as illustrated in Fig. 1. Recently, various exploration planning methods have been proposed to address the challenges of autonomous exploration. There are two mainstream approaches to addressing the problem of autonomous robot exploration: frontier-based Yamauchi (1998) methods and sampling-based Lindemann and LaValle (2005) methods. The frontier-based approach is the conventional method for three-dimensional space exploration. It defines frontiers as the free areas adjacent to unknown regions and explores the entire space by repeatedly tracking the nearest frontiers. Building upon the frontier-based approach, subsequent research has made corresponding adjustments in terms of information gain and path length, as well as added additional constraints based on application scenarios, requirements, and platforms Selin et al. (2019); Meng et al. (2017); Batinovic et al. (2021); Schmid et al. (2021). For example, the work Dai et al. (2020) achieves implicit grouping of frontier voxels by leveraging a low-level octree map, circumventing the computationally intensive frontier clustering associated with traditional frontier-based exploration methods. Additionally, frontiers are extracted

✉ Ze Ji
jjz1@cardiff.ac.uk

Changyun Wei
c.wei@hhu.edu.cn

Jianbin Wu
lthebs@163.com

Yu Xia
mr_xia_yu@163.com

¹ College of Mechanical and Electrical Engineering, Hohai University, Changzhou, China

² School of Engineering, Cardiff University, Cardiff, Wales, UK

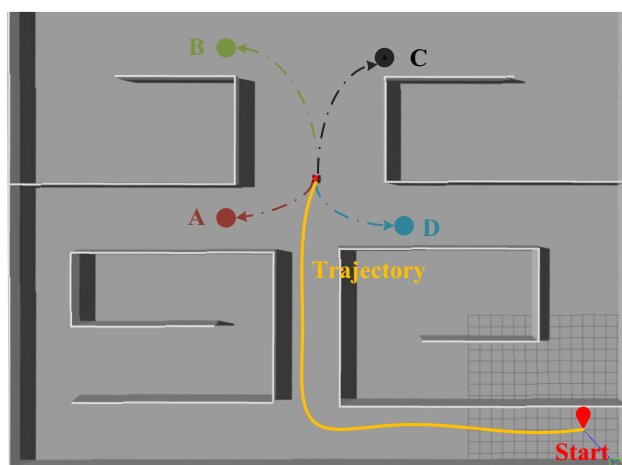


Fig. 1 An illustration of autonomous exploration task. The robot needs to decide where to go next based on the current sensor information until it has explored the entire environment

from the field of view, and a selection process is employed to choose the most suitable frontier that minimizes velocity changes, thereby facilitating high-speed flight for unmanned aerial vehicles in Cieslewski et al. (2017).

Another approach to solving the problem of autonomous exploration is based on randomization Zhu et al. (2021); Sun et al. (2022); Zhong et al. (2021); Kulkarni et al. (2022). The NBVP (Next-Best-View Planning) Bircher et al. (2016) stands as one of the most renowned methods among randomized sampling-based approaches. It leverages nodes on a Rapidly-exploring Random Tree (RRT) Kazemi et al. (2013) as viewpoints and employs a greedy strategy to select branches with the maximum collective reward from relevant viewpoints. In Dang et al. (2019), the authors present a graph-based path planning method (GBPlanner) that utilizes the Rapidly-exploring Random Graph (RRG) Karaman and Frazzoli (2009) framework to facilitate optimal movement within local subspaces for underground navigation. The work Dharmadhikari et al. (2020) introduces a motion-based path planning method (MBPlanner). Diverging from GBP, this approach takes account of the robot's motion state and guides the unmanned aerial vehicle towards positions with the highest gains after generating random samples. Due to excessive reliance on random sampling, sampling-based methods exhibit numerous suboptimal and unnecessary movements.

However, the performance of these methods in terms of exploration efficiency has been largely unsatisfactory. Some planners adopt greedy strategies Li et al. (2015) for motion planning, such as selecting viewpoints with the highest information gain or navigating to the nearest unexplored areas. While these strategies provide immediate decisions based on current situation, they neglect global optimality, resulting in overall low exploration efficiency.

Additionally, most planners rely on frontier-based or sampling-based methods, both of which require substantial computational resources. The substantial computational burden can impede the planners' ability to operate at high frequencies, potentially resulting in robots coming to a halt and waiting for computation results, particularly in large-scale scenarios.

In response to the aforementioned limitations of existing exploration planners, this paper proposes a novel and comprehensive approach for autonomous exploration that effectively addresses those issues. To enhance the efficiency of autonomous exploration, this paper combines frontier-based and sampling-based methods by imposing constraints on the robot's sampling region using a fixed-size sliding window. By employing the quantity of frontiers as information gain for sampling viewpoints, robots are effectively guided towards exploring uncharted regions. It is important to note that frontier detection remains active throughout the entire robot exploration process. Furthermore, in order to improve the search speed of the global topological graph, this paper introduces a sparse graph construction method. We have evaluated the proposed approach across several environments and compared it with existing state-of-the-art methods. The experimental results demonstrate that our approach outperforms the benchmark methods in terms of the total exploration time and the computation time.

The main contributions of this paper are summarised as follows.

1. The frontier-based methods facilitates convenient access to environmental frontiers; however, it is associated with slower global exploration rates. On the other hand, sampling-based methods effectively explore individual regions, yet the drawback lies in the inability to comprehensively cover all sectors of the environment. To address these limitations, this paper presents a hierarchical exploration approach that combines sampling and frontier methods, effectively mitigating the shortcomings present in both methods.
2. An utility function is devised to select suitable target points by considering both information gain and the robot's forward direction, which can mitigate the issue of abrupt changes in the robot direction at branching intersections, thereby enhancing the overall efficiency of exploration.
3. This paper presents a spatially structured method to constructing a sparse topological map. This method dynamically captures the spatial structure of free space based on local sampled points and incrementally constructs a three-dimensional sparse topological graph. Consequently, it notably reduces the number of nodes and edges in global topological graphs, thereby enhancing the speed of global path generation.

2 Fast exploration with sparse topological graphs

The general framework proposed in this paper is illustrated in Fig. 2. Initially, the sensor data is fed into Octomap for map construction, with the frontier being updated as Octomap is updated. Next, the planner uniformly samples points at current position and assesses their potential gains based on the frontier points. Additionally, the sparse graph is expanded at current position. Finally, the node with the maximum gain is selected based on the reward function, and a collision-free path is generated.

2.1 Frontier detection

The frontier is defined as a spatial voxel that is adjacent to at least one unknown voxel. The objective is to guide robots towards these frontiers, iteratively reducing the map entropy in environments. In this paper, we employ OctoMap Hornung et al. (2013) as the chosen three-dimensional grid framework due to its efficient data structures and algorithms, facilitating rapid and effective map construction and updating. However, the process of detecting all frontiers in the map can be time-consuming and may adversely impact the real-time nature of robot exploration. To alleviate this computational burden, we adopt a similar approach to 3D-FBET Zhu et al. (2015), where voxels experiencing changes in their spatial state are added to a set when a new frame of point cloud data is input. Since frontiers tends to appear near the maximum field of view of the sensor, the detection efforts are concentrated on the voxels with spatial state changes in the vicinity of the sensor's maximum field of view, effectively reducing the detection time.

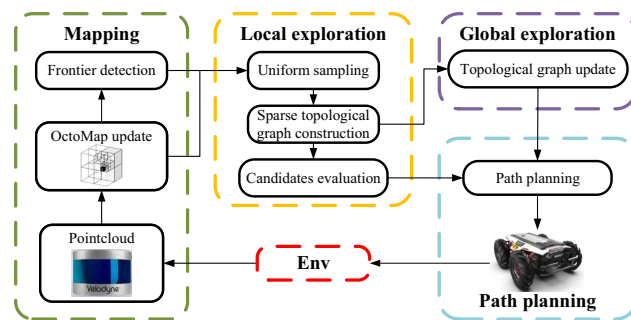


Fig. 2 An overview of the proposed exploration framework. Octomap converts the data from the sensor into spatial voxels, and boundary detection is also done in this process. Based on Octomap, uniform sampling and sparse graph construction are performed in a robot-centered location. Finally, the evaluation of candidate points and path planning are performed to guide the robot for autonomous exploration

2.2 Local exploration

The primary objective of local planning is to efficiently explore the sub-environments. In this paper, we employ a uniform sampling strategy Zhang et al. (2021) to generate viewpoints within a fixed-sized window, which are then used to construct an undirected topological graph denoted as $G(V, E)$. Uniform sampling is preferred over random sampling to mitigate the introduction of significant randomness due to an uneven distribution of data points. Random sampling may result in certain areas being overly explored while others are neglected. To verify whether the sampled viewpoints are in free space or not, collision checks are performed using Octomap, which assesses the presence of obstacles along the edges connecting the viewpoints. Prior research has predominantly estimated node gains through visibility-based ray tracing propagation, which identifies unknown voxels in the proximity of nodes Selin et al. (2019). However, the study Schmid et al. (2020) has indicated that the time spent on ray tracing estimation for node gains accounts for 95% of the total planning time.

To alleviate the computational burden, this study adopts a method that utilizes the total number of visible frontiers in the vicinity of the nodes as a measure of information gain, instead of relying on the number of unknown voxels surrounding the nodes, effectively guiding the robot towards frontier regions. An illustration for local exploration is shown in Fig. 3. To select the optimal exploration nodes considering the information gain and the motion cost, we employ the following function,

$$G(v_i) = e^{-\lambda L(r, v_i)} I(v_i), \quad (1)$$

$$\lambda = \left| \frac{yaw - \theta_i}{2\pi} \right|. \quad (2)$$

In Eq. 1, $I(v_i)$ represents the number of visible frontier points around viewpoint v_i , and $L(r, v_i)$ represents the distance from the robot's position r to viewpoint v_i . This equation is formulated to prevent the robot from greedily selecting nodes with the highest gain, which may result in overall low exploration efficiency. In Eq. 2, λ represents the weighting coefficient between the exploration cost and the information gain. To prevent sudden changes in the robot's exploration strategy, λ is defined as the difference between the current orientation of robot yaw and the angle to the target node θ_i . The nodes with gains contrary to the robot's forward direction are assigned a large motion cost, ensuring the robot maintains its forward direction,

$$v_i = \arg \max_{v_i} G(v_i). \quad (3)$$

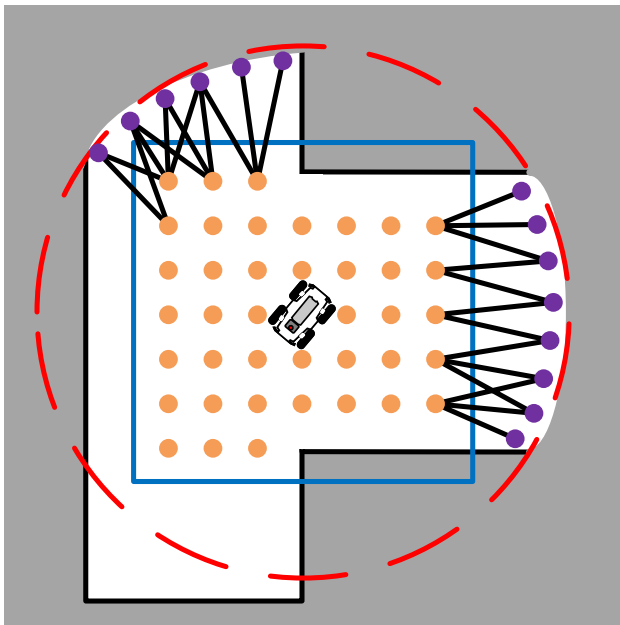


Fig. 3 An illustration of the proposed local exploration method. The red dashed line corresponds to the LiDAR's maximum field of view. The blue rectangle represents a fixed-size sampling window centered on the robot. The orange and purple dots denote the sampling and frontier points, respectively. The orange points connected to the purple ones are candidate points (color figure online)

Finally, we can select the optimal node for a single planning iteration as in Eq. 3.

2.3 Incremental construction of sparse topological graphs

Inspired by Chen et al. (2022), we propose an incremental sparse graph construction method based on spatial geometric structures, as illustrated in Fig. 4. The method consists of three stages: the polygon construction stage, the vertex generation stage, and the topological graph connection stage.

Initially, the geometric structures of the robot's surrounding environment are uniformly sampled to facilitate the generation of topological vertices at appropriate locations in the polygon construction stage. This procedure involves emitting rays, each with a fixed angular distribution, from the robot's location to assess the spatial characteristics of intersected voxels within a defined range. When a voxel of an occupied status is encountered along a ray's path, progression ceases, and the voxel's position is logged, subsequently joining the set of occupied voxels. Sequentially adhering to the order of emitted rays, the vertices within the occupied set are connected to incrementally form an occupied polygon, as illustrated by the orange lines in Fig. 4a.

The subsequent phase involves leveraging the occupied polygon to generate new vertices and expanding the existing topological graph into unknown areas in the vertex generation stage. Based on real-time observations of frontiers discerned at the robot's position, each edge's midpoint within

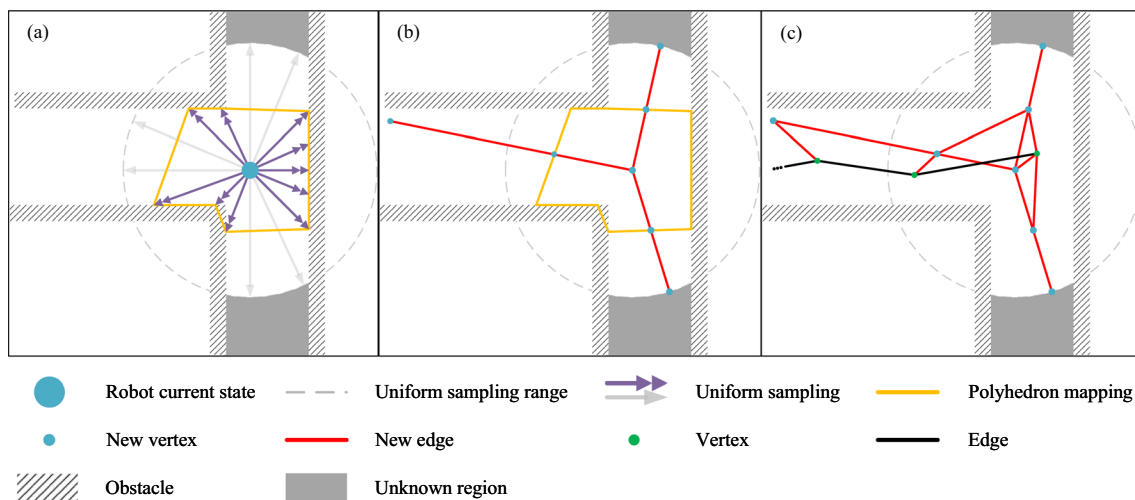


Fig. 4 Illustration of incremental construction of sparse topological graphs. This process consists of three parts: the polygon construction stage, the vertex generation stage, and the topological graph connection stage. (a) The robot captures the geometric structure of the surrounding environment through uniform sampling for constructing occupied polygon (orange). (b) Based on the polygon, we use predefined rules to generate new vertices (blue) that indicate the midpoint

of each unoccupied edge of the polygon, the robot's current position, and the determined locations near the frontiers. These new vertices are connected to from branches (red) that extend to the frontiers. (c) The newly generated branches are added to the topological graph, and connections are established with all existing vertices (green) within the neighborhood of the new vertices (color figure online)

the polygon is integrated into the ray set. Then, the spatial status of each point in the ray set is examined, and the occupied points are removed from this set. Similarly, emanating from the robot's position, rays are emitted towards each point within the ray set, scanning the spatial status of voxels along their trajectories until encountering a non-free status voxel. When the following three cases occur at the ray endpoint, it will be added as a vertex to the topological graph: firstly, when the status of the ray endpoint remains unknown; secondly, when the ray endpoint registers as occupied with proximal frontiers detected, and lastly, when the ray endpoint is occupied, but lacks nearby vertices. Upon the occurrence of these conditions, the ray endpoint, the mid-point of the polygon's edge, and the robot's current position are added as vertices to the topological graph, as illustrated in Fig. 4b. The advantage of this strategy is that the robot can effectively record the location of unexplored areas while facilitating a straightforward and rapid construction of the topological graph.

Finally, the newly generated vertices are incorporated into the topological graph. Similar to the connection edge strategy employed in the Rapidly-exploring Random Graphs (RRG), we establish connections between the newly generated vertex and all other vertices within a specified radius. This process is intended to form a more densely interconnected graph structure so as to reduce the length of the robot's traversal path. The resultant effect is illustrated in Fig. 4c. Through the employment of this method, the robot can construct a sparse topological graph for global navigation during continuous exploration.

2.4 Global exploration

The primary objective of the global planning is to guide the robot to sub-regions of the environment that still contain frontier points when none of the nodes in the local graph yield any benefits. During the activation of the global planning, the planner selects the next node to explore in the sparse global graph based on Eq. 1 and employs the Dijkstra

algorithm to determine the shortest path from the current node to the target.

Given the sparsity of the topological graph, it is imperative to optimize the nodes in the shortest path to the target node. We propose a path optimizer to trim unnecessary nodes from the sparse topological graph. The algorithmic process of the path optimizer is delineated in Algorithm 1.

Algorithm 1 Path optimization for sparse topological graphs.

```

1: Input: Original path  $\mathbf{P}$ 
2: Output: Optimized path  $\mathbf{Q}$ 
3: Add the start point ( $p_0$ ) to the optimized path  $\mathbf{Q}$ 
4:  $n \leftarrow \text{SizeOf}(\mathbf{P})$ 
5: for  $i$  from 1 to  $n$  do
6:   for  $j$  from  $i$  to  $n$  do
7:     Flag  $\leftarrow$  Check collisions between  $p_i$  and  $p_j$ 
8:     if Flag == True then
9:        $p_{new} \leftarrow p_j, k \leftarrow j$ 
10:    end if
11:  end for
12:  Add the new point  $p_{new}$  to  $\mathbf{Q}$ , and update  $i \leftarrow k$ 
13: end for

```

Assuming the original path $\mathbf{P} = (p_0, p_1, p_2, \dots, p_n)$, where p_0 and p_n represent the start and end points, and n denotes the total number of nodes in the original path. We sequentially check whether there is a collision along the line connecting p_i and p_j , where i starts with an initial value of 0, and $j \in (i, n)$. In such a sequence, the last node, p_k , which does not collide with p_i , will be added to the optimized path \mathbf{Q} , and i is set to k . The algorithm will continue to repeat the aforementioned process until i reaches n . Once the algorithm has executed, we sequentially connect all the nodes in the set \mathbf{Q} , resulting in an optimized path shorter than the original one, as illustrated in Fig. 5a.

The path optimizer algorithm, despite effectively reducing the overall distance of the global path, might introduce sharp turns that fall outside the permissible kinematic constraints of the robot. To address this issue, we employ second-order Bezier curves as a corrective measure to smooth out the sharp turns in the modified path,

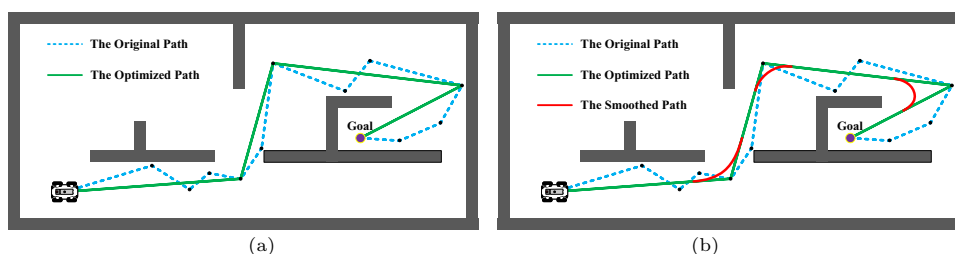


Fig. 5 An illustration of the proposed global exploration method. The blue dashed lines represent the original paths obtained from the sparse topological graph using the Dijkstra algorithm. The green lines

depict the paths pruned by the path optimizer algorithm, while the red lines represent the paths smoothed using second-order Bezier curves (color figure online)

$$B(t) = (1-t)^2P_0 + 2t(1-t)P_1 + t^2P_2, t \in [0, 1], \quad (4)$$

where P_0 , P_1 , and P_2 represent the three control points of the Bezier curve. All sharp corners in the global path undergoes a smoothing process. In this paper, the position of P_1 is strategically placed at the sharp corner, while P_0 and P_2 are positioned along the two edges connected to P_1 . The distance of P_0 and P_2 from P_1 varies with the angle formed by the two connected edges, as illustrated in Fig. 5b. Finally, we can obtain a smoother and shorter path compared to the original one.

3 Experiments and results

In order to validate the effectiveness of the proposed approach, we have conducted testing and comparisons across multiple simulated environments. The experimental robot was equipped with a Velodyne LiDAR and IMU, with its maximum linear velocity restricted to 1 m/s. For comparative analysis, we employed two cutting-edge algorithms, GBP Dang et al. (2019) and FAEL Huang et al. (2023), as benchmark algorithms. Both GBP and FAEL were configured with their default configurations. The evaluation metrics encompassed the count of nodes and edges within the global topological map, the exploration time, and the computation time per single iteration. All experiments were

conducted in real-time on an Ubuntu 20 system powered by a 2.90 GHz i5-10400F CPU.

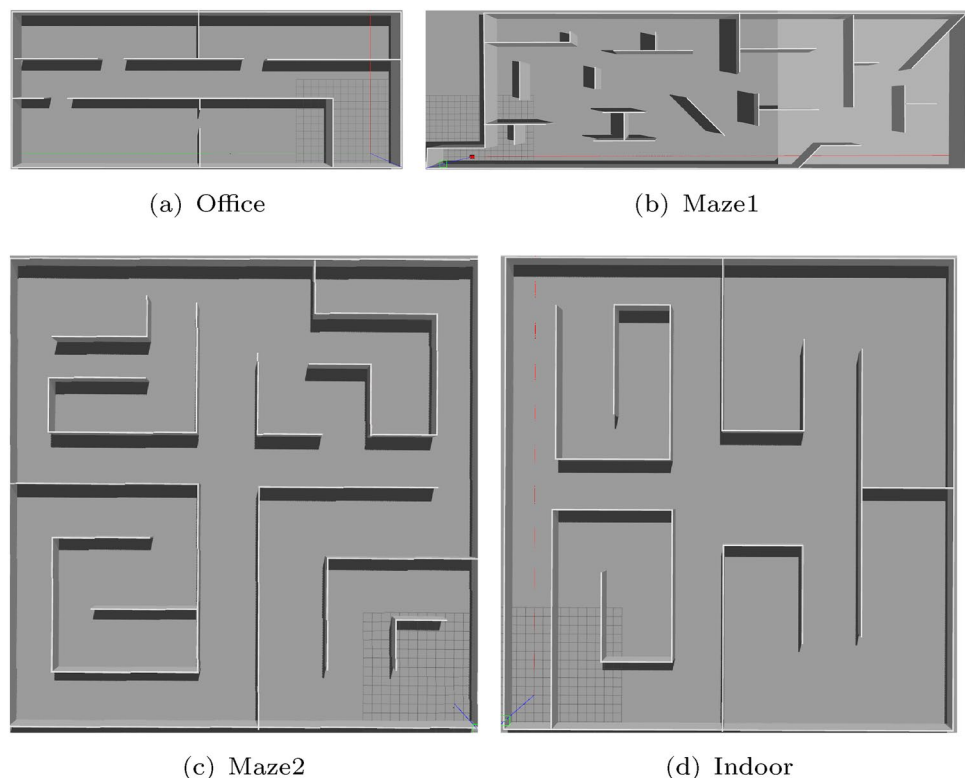
3.1 Experiment settings

We designed four benchmark scenarios encompassing environments of different sizes: an office environment (20 m × 50 m), maze1 (30 m × 70 m), maze2 (50 m × 50 m), and an indoor environment (50 m × 50 m). The environmental configurations are illustrated in Fig. 6. The office and maze1 environments exhibit distinct structural characteristics. The office layout is carefully planned, featuring various zones interconnected by narrow corridors. On the contrary, maze1 presents an irregular structure with diverse-shaped obstacles. These environments are specifically chosen to evaluate the efficacy of the proposed global topological graph construction method in scenarios involving confined passages and irregular obstacles. Both maze2 and the indoor environments represent large-scale settings. The indoor environment has a relatively straightforward structure, while maze2 incorporates multiple junctions, serving as test scenarios to assess the overall exploratory performance of different algorithms.

All experiments were conducted under the following conditions.

- The robot initiated from the same starting point with the same initial yaw angle.
- The sensor's maximum detection range was set to 15 m.

Fig. 6 The four environments used for benchmarking



- An initial action of moving forward 3 m was applied to ensure that the robot started within unoccupied voxels.
- Each experiment was considered complete when the robot ceased its movement within the environment.

In our approach, for local planning, we employed a sliding window size slightly smaller than the sensor's maximum range, specifically set at 12 m×12 m. The effective range for information gain of frontiers was limited to 5 m. Regarding the construction of a sparse graph, the radius utilized for uniform sampling was equivalent to the sensor's maximum detection range, that is, 15 m. Rays were emitted from the center of polygon edges with infinite length, yet terminated upon encountering voxels with unknown or occupied spatial states. Newly generated nodes were connected to the nodes existing within a 5 m radius in the topological graph. Lastly, the resolution of the Octomap was standardized at 0.5m.

3.2 Evaluation of the sparse topological graph construction

To evaluate the proposed sparse graph construction method, we first conducted experiments in two different spatial scenarios, the office and maze1 environments. The global graph construction method in GBP, named RRG Karaman and Frazzoli (2009), was chosen as the comparative baseline. The RRG was configured with the following parameters. It allowed for a maximum edge length of 5 m and a minimum edge length of 0.5m. The radius of the neighborhood within which new nodes were connected to existing nodes in the graph was set to 3 m.

We have recorded the quantity and spatial distribution of the nodes and edges generated by both methods during the experiments. The experimental outcomes are depicted in Fig. 7. It is evident that both the proposed method and the RRG method can successfully generate topological maps of the environments. However, the RRG method produces the nodes that nearly blanket the entire environment, while the proposed method effectively represents the skeleton of the environment with a minimal number of nodes. This

substantial discrepancy in the number of the nodes and the edges can be attributed to the randomness of the sampling points and the edge connection strategy employed in RRG.

Moreover, we use Fig. 8 to demonstrate the quantities of the global graph nodes and the edges for both methods in different environments. We can see that, in the office environment, the RRG generated 203 nodes and 1038 edges, whereas the proposed method resulted in only 56 nodes and 168 edges. In the maze1 environment, the RRG produced 393 nodes and 2238 edges, while the proposed method yielded only 95 nodes and 338 edges. We can conclude that the proposed method generates less than 30% of the nodes and edges compared to the RRG method. This is attributed to the unique ray sampling and validation mechanism in our method, which ensures stable sampling of nodes at boundaries without introducing node redundancy.

The proposed method may inadvertently omit certain regions, such as the top-left corner of office and the top-right corner of maze1. This occurrence arises because, when the sparse graph construction module is initiated, these aforementioned locations no longer exhibit frontiers,

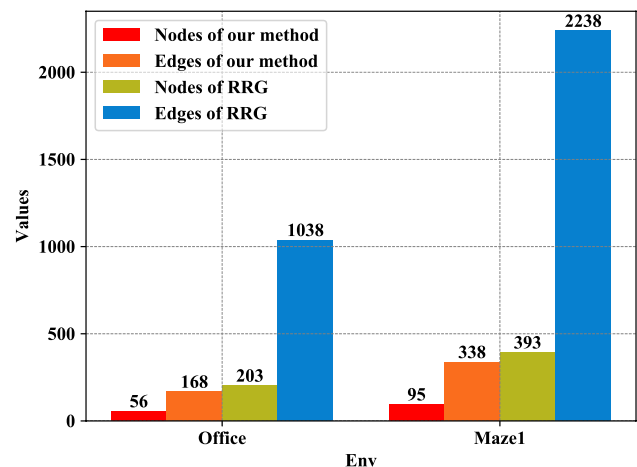
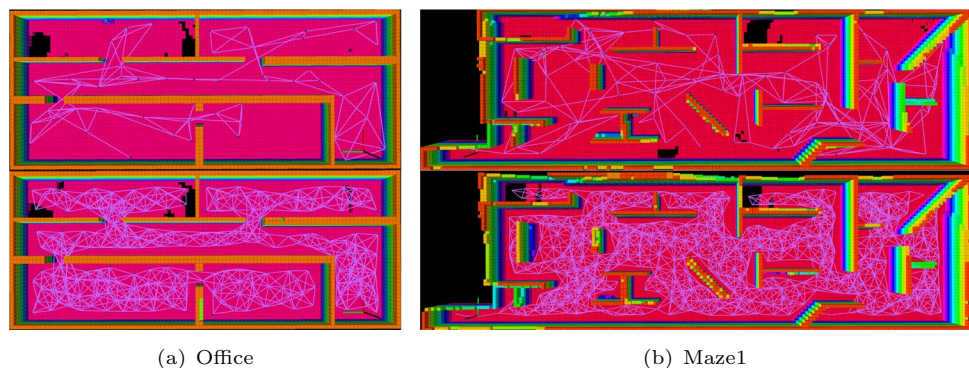


Fig. 8 A comparison of the number of elements in the global topological graph

Fig. 7 A comparison of the topological graphs in the office and maze1 environments. The images of the upper part are our proposed construction graphs, while the ones in the lower part are the graphs generated by the RRG



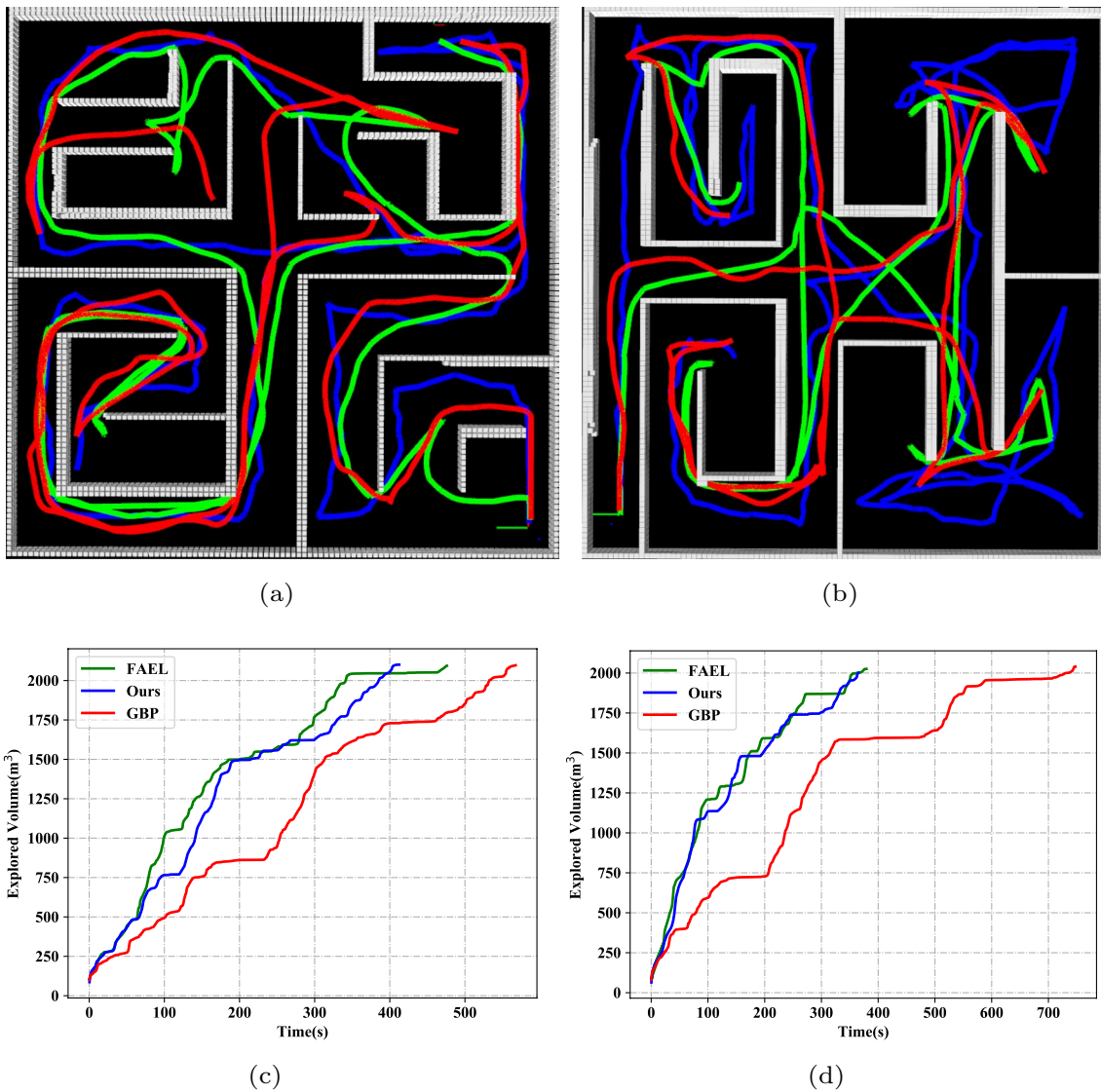


Fig. 9 Comparison of the exploration performance in the maze2 and indoor environments. **a** and **b** demonstrate the trajectory of each methods in the maze2 and indoor environments, respectively. The red line denotes our method, the blue line represents the GBP, and

the green line indicates the FAEL. **c** and **d** are the comparison of the three methods with respect to the exploration volume and the exploration time (color figure online)

leading to the non-generation of nodes in frontier-absent areas. In the context of autonomous exploration, the objective is to thoroughly explore all regions. Areas lacking frontiers imply the absence of a necessity for exploration. Hence, the omission of frontier-free regions in the global graph does not adversely affect autonomous exploration. In summary, the proposed method effectively represents the environmental skeleton with a minimal number of nodes and edges, thereby enhancing the search speed of the global topological graph in large-scale environments.

3.3 Exploration performance

We further evaluated the exploration performance in both the indoor and maze2 environments. We have recorded the key metrics such as the exploration volume, the completion time, the trajectory and travel distance for each experiment,

Table 1 Results of the travel distance in different environments

| Environment | GBP | FAEL | Ours |
|-------------|-----------|------------------|-----------------|
| Maze2 | 420.794 m | 398.16 m | 385.18 m |
| Indoor | 535.797 m | 347.072 m | 350.248 m |

as depicted in Fig. 9. In addition, Table 1 details the distances covered by each methods within the completion time.

We can see that in both the maze 2 and indoor environments, the GBP exhibits the longest completion time and covers the greatest distance during exploration compared to the proposed method. This can be attributed to the impractical utility function configuration of the GBP. After completing exploration within a specific area, the GBP may engage in multiple back-and-forth movements within that region, as illustrated in the left part of Fig. 9b, significantly impeding the efficiency of robotic exploration. In the maze2 environment, the proposed method takes 412 s and covers a distance of 385.18 ms, while the FAEL requires 472 s and covers 398.16 ms. In the indoor environment, our proposed method consumes 365 s and traverses 350.248 ms, whereas the FAEL needs 387 s and covers 347.072 ms. Thus, we can claim that the proposed method can achieve the exploration performance comparable to the state-of-the-art FAEL algorithm and outperforms the GBP algorithm based on random sampling.

With respect to the local planning, this paper incorporates the current exploration direction into the utility function, ensuring that the robot continues its exploration in the same direction. This effectively mitigates the issue of decreased exploration efficiency resulting from abrupt changes in the robot's direction. Furthermore, the behavior observed in Fig. 9a and b, where the robot can quickly enters and exits dead ends, illustrates the advantages of the proposed method in utilizing frontiers as information gains, which significantly enhances exploration efficiency.

While the paths generated by the proposed sparse graph construction method may not match the quality of dense graphs generated by the RRG and the FAEL, it is evident from the latter part of the global planning phase in Fig. 9c and d that our global planning approach still can guide the

Table 2 Average computation times of the components of our proposed method

| Environment | Sampling | Gain evaluation | Find path |
|-------------|----------|-----------------|-----------|
| Maze2 | 1.594 ms | 0.012 ms | 0.273 ms |
| Indoor | 1.500 ms | 0.015 ms | 0.388 ms |

robot to another subregion with the frontiers in the shortest possible time. It demonstrates that our global planning method produces smoother high-quality paths with shorter travel distances.

3.4 Computation efficiency

In terms of computation time, the algorithm runtime of the GBP is around 150ms, while the our proposed method and the FAEL run in less than 50 ms. Thus, the experimental results only display our proposed method and FAEL, as shown in Fig. 10.

We can find that the iteration computation time of the FAEL gradually increases with the exploration time due to the increasing number of nodes and edges in the global graph, resulting in increased search costs. In contrast, the iteration time of our method remains around 2ms throughout the entire exploration time, only one-tenth of that of the FAEL. This is achieved because of the fixed-size sampling window and the sparse topological graph. As depicted in Fig. 7, the sparse graph construction method proposed in this paper can generate a graph that covers the entire global map, similar to the RRG, while being sparser. Compared to the RRG or other global graphs, the search time is significantly reduced. Additionally, the evaluation computation times of each component of our proposed method are presented in Table 2. It is worth mentioning that the sampling time

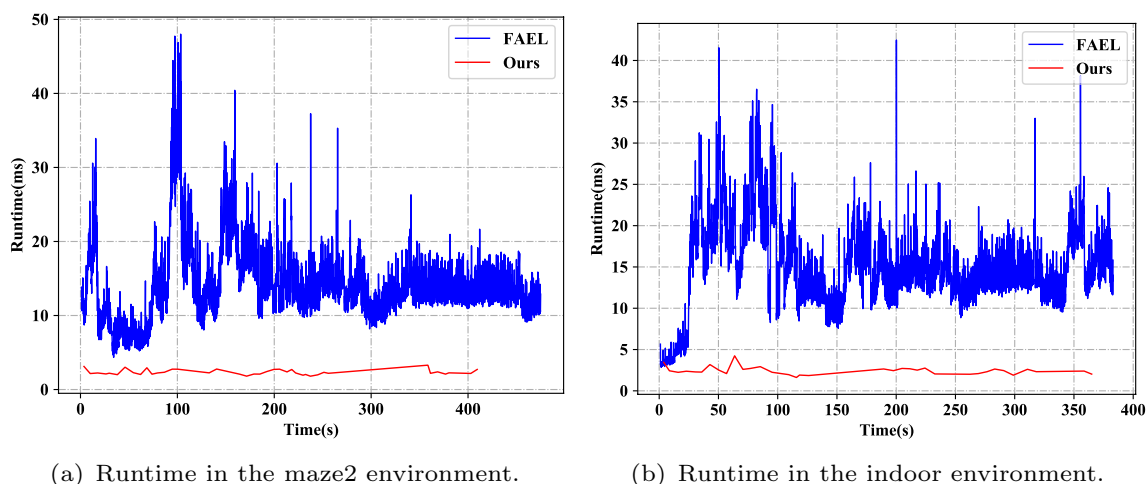


Fig. 10 The runtime evaluation in the maze2 and indoor environments

contains the construction time of the sparse graph. We can claim that the proposed method exhibits relatively low computation times in both the maze2 and indoor environments.

4 Conclusion

This paper proposes a hierarchical exploration approach that integrates the frontier and sampling methods. This approach comprises a local exploration stage, which swiftly expands the free space within the environment, and a global stage that guides the robot to different sub-regions. To enhance the search speed of the global topological graph, this paper introduces a method for constructing a sparse topological graph. During the exploration process, the planner incrementally constructs a three-dimensional sparse topological graph by dynamically capturing the spatial structure of free space through uniform sampling. In various challenging simulated environments, the proposed approach attains comparable exploration performance to the state-of-the-art methods while demonstrating significantly improved computational efficiency. In the future, we will continue to explore optimization methods for constructing sparse graphs, with the aim of ensuring that the regions with smaller openings will not be missed. For example, when constructing a sparse graph, frontier cluster points are used as nodes and added to the sparse topological graph that can be generated in any region with frontiers.

Author Contributions All authors contributed to the study conception and design. CW and JW performed material preparation, data collection and analysis. CW, JW, and YX wrote the first draft of the manuscript and ZJ revised the manuscript based on previous versions. All authors read and approved the final manuscript.

Funding This work was supported in part by the National Natural Science Foundation of China under Grant 52371275. This work was supported in part by the Royal Academy of Engineering under the Industrial Fellowships programme under Grant IF2223-199.

Code availability The simulation data will be available upon request.

Declarations

Ethics approval Not applicable. Our manuscript does not report results of studies involving humans or animals.

Consent to participate Not applicable. Our manuscript does not report results of studies involving humans or animals.

Consent for Publication All authors have approved and consented to publish the manuscript.

Conflict of Interest The authors have no relevant financial or nonfinancial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Batinovic, A., Petrovic, T., Ivanovic, A., Petric, F., Bogdan, S.: A multi-resolution frontier-based planner for autonomous 3d exploration. *IEEE Robot. Autom. Lett.* **6**(3), 4528–4535 (2021)
- Bircher, A., Kamel, M., Alexis, K., Oleynikova, H., Siegwart, R.: Receding horizon “next-best-view” planner for 3d exploration. In: *IEEE International Conference on Robotics and Automation, IEEE*, pp. 1462–1468 (2016)
- Chen, X., Zhou, B., Lin, J., Zhang, Y., Zhang, F., Shen, S.: Fast 3d sparse topological skeleton graph generation for mobile robot global planning. In: *International Conference on Intelligent Robots and Systems, IEEE*, pp. 10283–10289 (2022)
- Cieslewski, T., Kaufmann, E., Scaramuzza, D.: Rapid exploration with multi-rotors: a frontier selection method for high speed flight. In: *International Conference on Intelligent Robots and Systems*, pp. 2135–2142 (2017). *IEEE*
- Dai, A., Papatheodorou, S., Funk, N., Tzoumanikas, D., Leutenegger, S.: Fast frontier-based information-driven autonomous exploration with an mav. In: *2020 IEEE International Conference on Robotics and Automation (ICRA), IEEE*, pp. 9570–9576 (2020)
- Dang, T., Mascarich, F., Khattak, S., Papachristos, C., Alexis, K.: Graph-based path planning for autonomous robotic exploration in subterranean environments. In: *International Conference on Intelligent Robots and Systems, IEEE*, pp. 3105–3112 (2019)
- Dharmadhikari, M., Dang, T., Solanka, L., Loje, J., Nguyen, H., Khedekar, N., Alexis, K.: Motion primitives-based path planning for fast and agile exploration using aerial robots. In: *International Conference on Robotics and Automation, IEEE*, pp. 179–185 (2020)
- Hornung, A., Wurm, K.M., Bennewitz, M., Stachniss, C., Burgard, W.: Octomap: An efficient probabilistic 3d mapping framework based on octrees. *Auton. Robot.* **34**, 189–206 (2013)
- Huang, J., Zhou, B., Fan, Z., Zhu, Y., Jie, Y., Li, L., Cheng, H.: Fael: Fast autonomous exploration for large-scale environments with a mobile robot. *IEEE Robot. Autom. Lett.* **8**(3), 1667–1674 (2023)
- Juliá, M., Gil, A., Reinoso, O.: A comparison of path planning strategies for autonomous exploration and mapping of unknown environments. *Auton. Robot.* **33**, 427–444 (2012)
- Karaman, S., Frazzoli, E.: Sampling-based motion planning with deterministic μ -calculus specifications. In: *The 48th IEEE Conference on Decision and Control, IEEE*, pp. 2222–2229 (2009)
- Kazemi, M., Gupta, K.K., Mehrandezh, M.: Randomized kinodynamic planning for robust visual servoing. *IEEE Trans. Rob.* **29**(5), 1197–1211 (2013)
- Kulkarni, M., Dharmadhikari, M., Tranzatto, M., Zimmermann, S., Reijgwart, V., De Petris, P., Nguyen, H., Khedekar, N., Papachristos, C., Ott, L., et al.: Autonomous teamed exploration of subterranean environments using legged and aerial robots. In: *2022 International Conference on Robotics and Automation (ICRA), IEEE*, pp. 3306–3313 (2022)

- Li, S., Xu, X., Zuo, L.: Dynamic path planning of a mobile robot with improved q-learning algorithm. In: IEEE International Conference on Information and Automation, IEEE, pp. 409–414 (2015)
- Lindemann, S.R., LaValle, S.M.: Current issues in sampling-based motion planning. In: The Eleventh International Symposium, pp. 36–54. Springer, Berlin (2005)
- Ma, Z., Liu, S.: A review of 3d reconstruction techniques in civil engineering and their applications. *Adv. Eng. Inform.* **37**, 163–174 (2018)
- Meng, Z., Qin, H., Chen, Z., Chen, X., Sun, H., Lin, F., Ang, M.H.: A two-stage optimized next-view planning framework for 3-d unknown environment exploration, and structural reconstruction. *IEEE Robot. Autom. Lett.* **2**(3), 1680–1687 (2017)
- Schmid, L., Pantic, M., Khanna, R., Ott, L., Siegwart, R., Nieto, J.: An efficient sampling-based method for online informative path planning in unknown environments. *IEEE Robot. Autom. Lett.* **5**(2), 1500–1507 (2020)
- Schmid, L., Reijgwart, V., Ott, L., Nieto, J., Siegwart, R., Cadena, C.: A unified approach for autonomous volumetric exploration of large scale environments under severe odometry drift. *IEEE Robot. Autom. Lett.* **6**(3), 4504–4511 (2021)
- Selin, M., Tiger, M., Duberg, D., Heintz, F., Jensfelt, P.: Efficient autonomous exploration planning of large-scale 3-d environments. *IEEE Robot. Autom. Lett.* **4**(2), 1699–1706 (2019)
- Sun, Z., Wu, B., Xu, C., Kong, H.: Ada-detector: adaptive frontier detector for rapid exploration. In: 2022 International Conference on Robotics and Automation (ICRA), pp. 3706–3712, IEEE (2022)
- Yamauchi, B.: Frontier-based exploration using multiple robots. In: The Second International Conference on Autonomous Agents, pp. 47–53 (1998)
- Zhang, X., Chu, Y., Liu, Y., Zhang, X., Zhuang, Y.: A novel informative autonomous exploration strategy with uniform sampling for quadrotors. *IEEE Trans. Industr. Electron.* **69**(12), 13131–13140 (2021)
- Zhong, P., Chen, B., Lu, S., Meng, X., Liang, Y.: Information-driven fast marching autonomous exploration with aerial robots. *IEEE Robot. Autom. Lett.* **7**(2), 810–817 (2021)
- Zhu, H., Cao, C., Xia, Y., Scherer, S., Zhang, J., Wang, W.: Dsvp: dual-stage viewpoint planner for rapid exploration by dynamic expansion. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, pp. 7623–7630 (2021)
- Zhu, C., Ding, R., Lin, M., Wu, Y.: A 3d frontier-based exploration tool for mavs. In: The 27th International Conference on Tools with Artificial Intelligence, IEEE, pp. 348–352 (2015)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Dr. Changyun Wei received the PhD degree in Artificial Intelligence from the Delft University of Technology, Netherlands, in 2015. Currently, he is an Associate Professor with the College of Mechanical and Electrical Engineering, Hohai University, China. He has published more than 30 papers in the fields of computer vision, artificial intelligence and robotics. His research interests include robotics, multi-agent systems, intelligent systems and computer vision.



Jianbin Wu received B.S. degree in Mechanical Engineering from Jimei University, Xiamen, China, in 2020. He is currently pursuing the M.S. degree in the College of Mechanical and Electrical Engineering, Hohai University, China. His research interests include intelligent autonomous unmanned systems and robot navigation.



Yu Xia received the B.S. degree in Mechanical Engineering from Hohai University of Wentian College, Ma'anshan, China, in 2020. He is currently pursuing the M.S. degree in the College of Mechanical and Electrical Engineering, Hohai University, China. His research interests are deep learning and robotics.



Ze Ji received the Ph.D. degree from Cardiff University, U.K., in 2007. He is a Senior Lecturer (Associate Professor) with the School of Engineering, Cardiff University, U.K. Prior to his current position, he was working in industry (Dyson, Lenovo, etc) on autonomous robotics. His research interests include autonomous robot navigation, robot manipulation, robot learning, computer vision, simultaneous localization and mapping (SLAM), acoustic localisation and tactile sensing.