RESEARCH ON ROBOT NAVIGATION TECHNOLOGY BASED ON LASER SLAM

Xiaoyang Hu,* Sairu Liu,* and Jie Zhao*

Abstract

Taking an indoor mobile robot as the object, laser SLAM and navigation path planning as the main content, this paper aims to improve the problems in the traditional laser SLAM algorithm and navigation path-planning algorithm. The framework of an indoor mobile robot navigation system is established, and the effectiveness and practicability of the improved scheme are verified by simulation and experiments. The experimental results show that the improved laser SLAM algorithm can improve the mapping accuracy and provide an accurate map for subsequent navigation. The improved global path-planning algorithm can improve the path quality and provide an optimal path for the robot to reach the target point quickly and smoothly.

Key Words

Mobile robot, SLAM, path planning, radar, ROS

1. Introduction

In a changing environment, robots should have the functions, such as positioning, map construction, and path trajectory planning, *i.e.*, simultaneous localisation and mapping (SLAM) technology and navigation path-planning technology [1].

The main content of robot navigation is global path planning and local path planning. Global path planning can plan a feasible path from the starting point to the endpoint, which is the main research content. Local path planning can avoid obstacles and collisions in real time [2]. In practical applications, environments are complex and variable. The shapes and numbers of obstacles are different, and it is necessary for robots to plan a smooth path quickly and accurately [3]. Consequently, higher requirements are put forward to the global path planning of robots, such as increasing the planning speed of the path-planning algorithm, shortening the length of the planned trajectory, and improving the smoothness of the path, so that the robot can reach the target point quickly and smoothly in complex environments, and realise the efficient and autonomous navigation of the robot [4], [5].

Traditional path-planning algorithms include intelligent bionics algorithm and graph search algorithm. Sarmina and Khachaturov [6] improved these algorithms. Nazarahari *et al.* [7] proposed an improved artificial potential field algorithm under continuous navigation path planning of mobile robots, which improved the smoothness and safety of the path. Orozco-Rosas *et al.* [8] studied a membrane evolution artificial potential field method to solve the path-planning problem of mobile robots. This method combines membrane calculation with the artificial potential field method to find appropriate parameters and generate feasible safe path.

The above traditional path-planning algorithms [7], [8] have the problem that the computation increases exponentially in complex environments, and the algorithm is prone to fall into local minima. To reduce the complexity of the algorithm and improve the efficiency, the reduction of the complexity of the algorithm is necessary [9].

EKF-SLAM is the most representative laser SLAM. Its representative algorithm is the Hector algorithm, which can solve the nonlinearity problem, but with a large error. Rao–Blackwellsied particle filter SLAM (RBPF-SLAM) is the most generation based on particle filter SLAM table sex [10]. For non-Gaussian and nonlinear systems, the most representative of RBPF-SLAM is the Gmapping algorithm [11]. It can solve the SLAM problem in nonlinear and non-Gaussian systems. Any unevenness of the road surface or wheel friction during the robot movement will affect the accuracy of the odometer data, and it is poor at map building in outdoor environments. The Karto-SLAM graph optimisation algorithm proposed by Kümmerle *et al.* selects CSM front-end matching and G^2o library for optimisation. However, the Karto algorithm is slow in loopback detection [12], which is easy to detect false closed loops, and correctly identifies closed loops is a difficult problem.

Zhang *et al.* [13] proposed a high-stiffness hybrid spray-painting robot for touch-up painting in vehicle repair plant, which investigates the center of mass distribution from a

* School of Equipment Engineering, Shenyang Ligong University, Shenyang, Liaoning, 110159, China; e-mail: {xiaoyang_hu, 18737337900}@163.com; 2627707574@qq.com
Corresponding author: Xiaoyang Hu

Recommended by Jun Wu
(DOI: 10.2316/J.2024.206-0943)

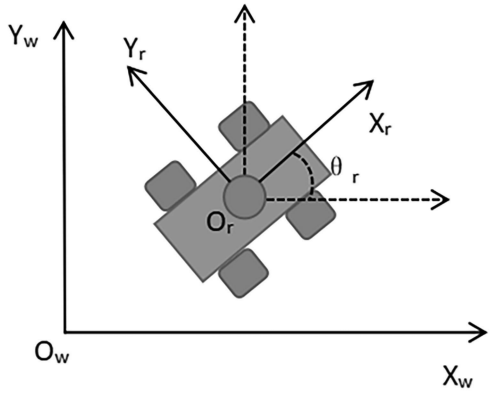

Figure 1. Robot coordinate system.

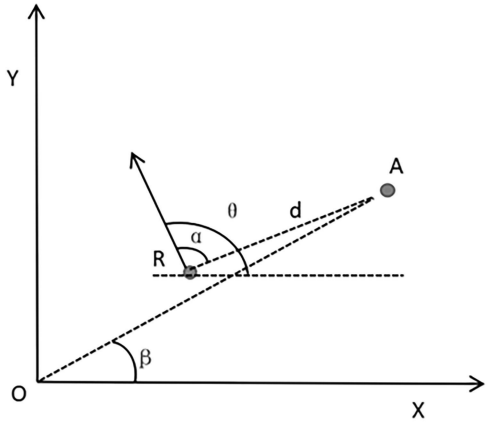

Figure 2. Laser data conversion.

kinematic model, with no improvement in the algorithms. Wu *et al.* [14] improved the multi-objective optimal design of a novel 6-degree of freedom (DOF) hybrid spray-painting robot, taking compactness, motion/force transmissibility, and energy consumption as performance indices, with less consideration given to the smoothness of the robot's motion path. Wu *et al.* [15] proposed an iterative learning method to accurately design the industrial feed-forward controllers and compensate for the external uncertain dynamic load of robots, which require installation of additional sensors in the processing system, but are difficult to widely implement in actual industrial production processes and also lack generalisability.

Taking an indoor mobile robot as the object, an improved RBPF-SLAM laser algorithm and an improved bidirectional rapidly-exploring random tree (BI-RRT) global path-planning algorithm are proposed to solve the mapping and navigation problems of robots. The results show that the RBPF-SLAM algorithm can improve the accuracy of map construction and provide accurate maps for subsequent navigation. The improved BI-RRT global path-planning algorithm can improve the path quality and provide an optimal path for the robot to reach the target point quickly and smoothly.

2. Mathematical Description of Robot Motion

2.1 Robot Motion Model

The robot coordinate system follows the movement of the robot, as shown in Fig. 1.

$$\begin{cases} x_w^1 = x_w^r + x_r^1 \cos\theta_r - y_r^1 \sin\theta_r \\ y_w^1 = y_w^r + y_r^1 \sin\theta_r + y_r^1 \end{cases} \tag{1}$$

The linear velocity and angular velocity of the robot and the radius r satisfy:

$$r = \left|\frac{v}{w}\right| \tag{2}$$

$$v = wr \tag{3}$$

Assuming that the initial pose of the robot is $x_{t-1} = (x, y, \theta)^T$, the coordinates of the arc center point can be expressed as:

$$x_o = x - \frac{v}{w}\sin\theta \tag{4}$$

$$y_o = y + \frac{v}{w}\cos\theta \tag{5}$$

After $\triangle t$ time passes, the pose of the robot can be obtained as:

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x_o + \frac{v}{w}\cos(\theta + w \triangle t) \\ y_o + \frac{v}{w}\sin(\theta + w \triangle t) \\ \theta + w \triangle t \end{pmatrix} \tag{6}$$

2.2 Radar Observation Model

Laser triangulation has good ranging results indoors. The laser data conversion process is shown in Fig. 2.

Through laser data conversion, the coordinates of the probe point A can be obtained as follows:

$$\begin{cases} x_A = x + d\cos(\theta - \alpha) \\ y_A = y + d\sin(\theta - \alpha) \\ \beta = \arctan\frac{y_A}{x_A} \end{cases} \tag{7}$$

During the movement progress, the laser is subject to external interference, resulting in certain errors in the obtained model. It is thence necessary to add noise factors to the laser observation model.

$$z(t) = h(s(t), \xi(t)) = \begin{pmatrix} d_t^i \\ \theta_t^i \end{pmatrix} + \xi(t) \tag{8}$$

In (8), z denotes the laser observation; h denotes the measurement function of the observation; s denotes the system state value, and ξ denotes the interference noise.

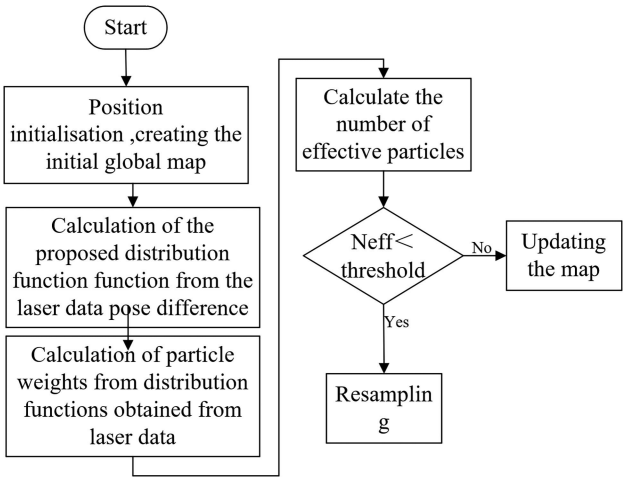

Figure 3. Flowchart of improved RBPF-SLAM algorithm.

3. Principle and Simulation of Laser SLAM Algorithm

3.1 RBPF-SLAM Algorithm

The main idea of RBPF-SLAM is to estimate the posterior probability density function by using the radar observation data and odometer data of robots [16]. Based on particle filtering [17], SLAM is divided into two parts: position and pose estimation, and map updating. The position and pose of robots are estimated by particle filtering, and the surrounding environment is estimated by EKF. RBPF-SLAM implementation includes initialising poses and maps, estimating positional and state poses using radar data and a proposed distribution function [18], updating particle weights, resampling based on valid particles, and updating the global map based on local maps and observations. The accuracy of the initial global map is due to the high precision of the robot's starting poses.

3.2 Improved RBPF-SLAM Algorithm

It can be seen that the proposed distribution function is important for an optimal position, and a good proposed distribution function is the key point to improve the positional accuracy [19]. Compared to the original algorithm, improved RBPF-SLAM algorithm improves in two ways, changing the original odometer data to the laser data pose difference as the input of the proposed distribution function and modifying the original proposed distribution function. Reasons and advantages are as follows, the numerical values obtained by laser radar are more real time and accurate than odometers [20]. The laser signal has unimodal characteristics and a smaller variance coefficient, which is the most suitable input quantity for the proposed distribution function. The proposed distribution function analysis mode of the improved RBPF-SLAM is as follows:

$$q\left(s_t|s_{t-1},z_t\right) = s_{t-1} + h_t\left(s_t,s_{t-1}\right) \quad (9)$$

$$h_t = s_t - s_{t-1} \quad (10)$$

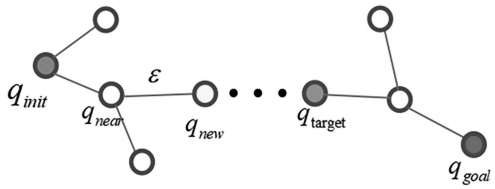

Figure 4. BI-RRT algorithm expansion.

where, s_t denotes the robot's pose at time t; z_t denotes the radar observation date, and h_t denotes the robot's front-back position difference function at time t. The improved RBPF-SLAM process is as follows:

(1) During initialisation, the radar scans the initial position and poses of the robot, processes laser data, and obtains the initial local map used as the global map.
(2) Input the position and pose difference of laser radar data at adjacent times as the proposal distribution instead of odometer data.
(3) According to the proposed distribution, the weight of the particles is obtained and normalised, and then the desired particle position and pose are obtained by the sum of the probability density function of the weighted distribution. This is used as the state position and pose at the current moment.
(4) For particle decay, the effective particle number Neff is calculated to determine whether to resample.
(5) Based on the position and pose of the current state, the data obtained by laser scanning builds a local map for updating.

The flowchart of improved RBPF-SLAM algorithm is shown in Fig. 3.

4. Global Path-Planning Algorithm

4.1 BI-RRT Algorithm

BI-RRT algorithm searches with the starting point q_{init} and the endpoint q_{goal} as the beginning of the expansion tree. The two expansion trees continuously generate random points for expansion. The search ends when both tree searches meet the threshold criteria. The two expansion trees of the BI-RRT algorithm are shown in Fig. 4.

4.2 Improved BI-RRT Algorithm

BI-RRT algorithm samples and expands the bidirectional spanning tree, which can save time, but cannot guarantee the quality of the path [21], [22]. Aiming at the problem of blind expansion, target bias sampling is proposed, so that the random point has a certain probability in the direction of the end. In view of the low generation path quality, the optimal parent node selection method of spanning tree is proposed to optimize the path quality.

4.2.1 Target Bias Sampling

To avoid the blind generation of random sampling points, the target bias sampling is added to make the expansion

tree have a certain direction [23]. The expansion tree can find the endpoint faster, and the target bias sampling is as follows:

$$q_{\text{rand}} = \begin{cases} q_{\text{goal}}, & \text{if } p_{\text{rand}} > p_{\text{bias}} \\ \text{SampleFree}(), & \text{else} \end{cases} \quad (11)$$

where, P_{bias} denotes the target bias threshold; P_{rand} denotes the random sampling probability, and SampleFree() denotes the security status space. The random sampling probability is in the range of (0, 1), and a threshold is set to determine whether the random sampling point is the endpoint. If the random sampling probability is greater than the threshold, the endpoint is selected as the random sampling point; otherwise, a randomly generated node is selected as the random sampling point.

Adding target bias cannot only preserve the global expansion characteristics of the RRT algorithm, ensure its probabilistic completeness, and enable the expansion of nodes to be spread throughout the state space, but also has local characteristics with a certain probability of convergence towards the end point [24]. The threshold selection of the target bias is a key issue, because when the threshold values are set on the large side, the probability of expanding to the endpoint is too small and has no significant effect on the expansion speed of the endpoint. When the threshold values on the small side are selected, the opportunity to expand to the endpoint is too large. In the environment with many obstacles, it is easy to fall into the local minimum and cannot find the endpoint. As such, a reasonable threshold value should be selected to allow the algorithm to have a higher efficiency, expanding the search to the endpoint [25]. The new node is expanded as follows:

$$q_{\text{new}}(x, y) = q_{\text{near}}(x + \varepsilon \cos\theta, y + \varepsilon \sin\theta) \quad (12)$$

$$\theta = \arctan(q_{\text{rand}}(y) - q_{\text{near}}(y), q_{\text{rand}}(x) - q_{\text{near}}(x)) \quad (13)$$

where, q_{new} is the result of the nearest node expansion q_{near}, and θ is the expansion direction of the new node.

4.2.2 Optimal Parent Node Selection

To reduce tortuous nodes and unnecessary travel, the optimal parent node selection is added. After planning the path through the BI-RRT algorithm, all nodes of the path are optimally selected and subsequent child nodes are traversed from the starting point to determine whether a child node can be reached directly without obstacle collision. If there are child nodes, this new local path is selected to replace the original path. This node is then used to traverse the remaining child nodes and iterate continuously to find the endpoint. In this way, many unnecessary process nodes can be skipped and several key nodes can be used to represent the path, thereby improving the quality of the path and reducing the traveling process for robot navigation.

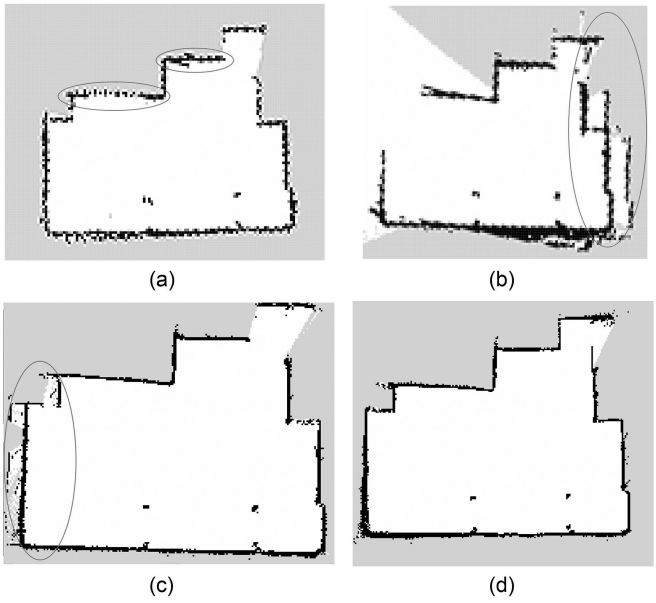

Figure 5. Mapping effect of the four algorithms in the environment of Experiment I: (a) Gmapping algorithm; (b) Hector algorithm; (c) Karto algorithm; and (d) Improved algorithm.

5. Robot Laser SLAM and Navigation Experiment

5.1 Laser SLAM Experiment

The following is SLAM Experiment 1. The environment of Experiment 1 is an indoor room with an overall size of 4.5 × 4.5 m, containing the obstacles, such as tables, cabinets, refrigerators, and more.

Figure 5 shows the overall mapping effect of the four algorithms in Experiment 1. The overall accuracy of the Gmapping algorithm is low, and burr noise points appear at the edge of obstacles, but no ghosting occurs in the overall map construction. The Hector algorithm can build a rough outline in a small environment, and ghosting is serious. The Karto algorithm can build a complete map, but ghosting appears on the right edge. The proposed improved RBPF-SLAM algorithm has the best overall mapping effect among the four algorithms, without ghosting and fewer rough edges at the edge of the obstacle.

The detail accuracy of the mapping effect in Experiment 1 is measured, and the feature position is selected to compare with the error of the actual accuracy. Since the Hector algorithm cannot construct a complete map, the comparison is not involved, as shown in Table 1 and Fig. 6.

It can be seen from Table 1 and Fig. 6 that the errors of the Gmapping algorithm and the Karto algorithm at different feature positions are different. The feature position error of the proposed improved RBPF-SLAM algorithm is smaller than that of the previous two algorithms, and the mapping accuracy is the highest.

Next, SLAM Experiment 2 is carried out. Experiment 2 is an indoor corner corridor environment, with a total length of 15 m and a width of 1–3 m. The obstacles include regular wooden doors, irregular walls, corner walls,

Table 1
Error of Experiment I

Algorithm		Gmapping		Karto		Improved	
Feature Points	Actual Value/cm	Measured Value/cm	Absolute Error/cm	Measured Value/cm	Absolute Error/cm	Measured Value/cm	Absolute Error/cm
1	42	45.3	3.3	45.48	3.48	46.92	2.5
2	41	47.11	6.11	50.94	9.94	46.92	4.92
3	50	48.92	1.08	49.12	0.88	50.55	0.55
4	112	108.72	3.28	110.98	1.02	111.89	0.11

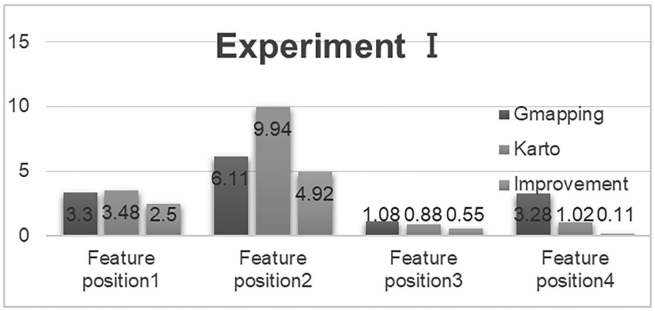

Figure 6. Error histogram of Experiment I.

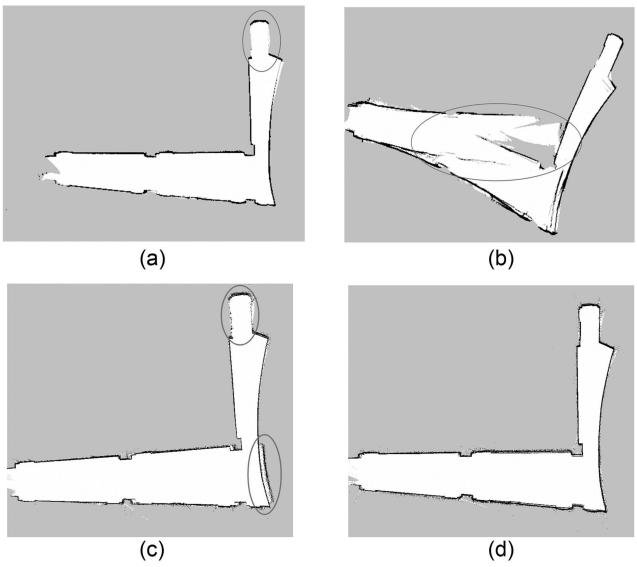

Figure 7. Drawing effect of the four algorithms in Experiment II: (a) Gmapping algorithm; (b) Hector algorithm; (c) Karto algorithm; and (d) Improved algorithm.

and more. The robot is controlled to scan and construct a map.

Figure 7 shows the mapping effect of the four algorithms in the corner corridor environment of Experiment 2. The red circle is a matter of detail in the mapping. The Gmapping algorithm cannot detect wall obstacles at the red circle and has burrs in the overall details, so that its mapping accuracy is average. The Hector algorithm has the same effect as the simulation, and only using

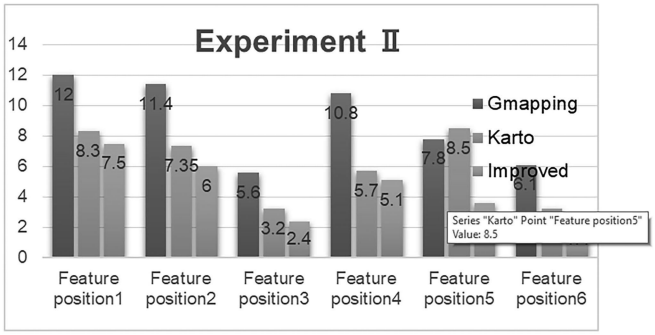

Figure 8. Error histogram of Experiment II.

laser data will result in a large mapping bias and a complete map cannot be built. The Karto algorithm has low accuracy in the upper right corner of the red circle, and there is ghosting in the lower right corner of it. To solve these problems, many loops are needed with a lot of time. Compared with the previous three algorithms, the improved RBPF-SLAM algorithm has higher mapping accuracy, less burrs, no ghosting, and accurate obstacle contour recognition. The mapping effect of Experiment 2 is measured with accuracy, and the selected feature position is compared with the actual accuracy, as shown in Table 2 and Fig. 8.

It can be seen from Table 2 and Fig. 8 that, except for feature position 3, the error of other details of the Gmapping algorithm is the largest in the three algorithms. The maximum error is feature position 1, with 12 cm. The overall detail error of the Karto algorithm is better than that of the Gmapping algorithm, and the maximum error is the feature position 5, with 8.5 cm. Compared with the previous two algorithms, the error of the proposed improved RBPF-SLAM algorithm is the smallest at all feature positions, and the maximum error is the feature position 1, with 7.5 cm.

Experiment 3 is an indoor conference room, with an area of m, including tables, chairs, uneven walls, and monitor stands. It is an environment with relatively dense obstacles, and the environment for Experiment 3 is shown in Fig. 9.

Figure 10 shows the mapping effect of the four algorithms in Experiment 3. There are many obstacles in the conference room, which requires high mapping details of the algorithm. The Gmapping algorithm has

Table 2
Errors of Experiment II

Algorithm		Gmapping		Karto		Improved	
Feature Points	Actual Value/cm	Measured Value/cm	Absolute Error/cm	Measured Value/cm	Absolute Error/cm	Measured Value/cm	Absolute Error/cm
1	139	151	12	149.6	8.3	146.5	7.5
2	115	126.4	11.4	124.35	7.35	121	6
3	84	92.6	5.6	93	3.2	89.4	2.4
4	104	114.8	10.8	112.9	5.7	111.2	5.1
5	115	131.3	7.8	127.7	8.5	118.6	3.6
6	57	63.1	6.1	60.2	3.2	58.4	1.4

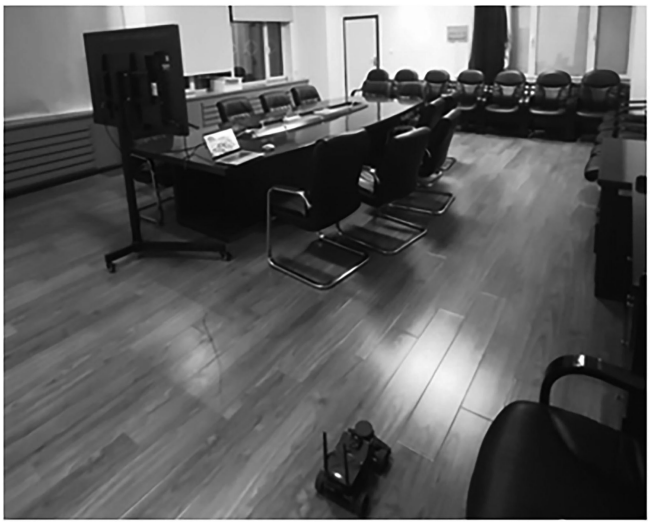

Figure 9. Real environment of SLAM Experiment III.

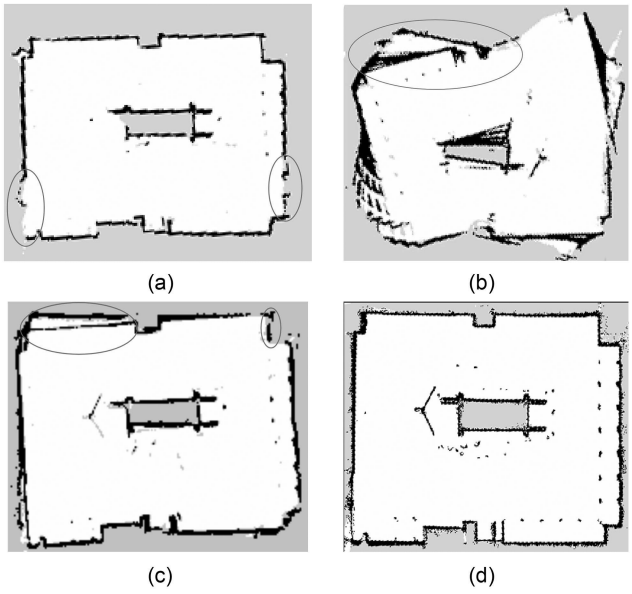

Figure 10. Mapping effect of the four algorithms in Experiment III: : (a) Gmapping algorithm; (b) Hector algorithm; (c) Karto algorithm; and (d) Improved algorithm.

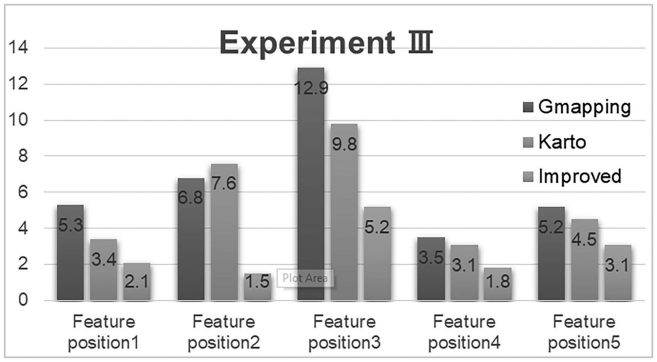

Figure 11. Error histogram of Experiment III.

low accuracy in the construction of the two red circles. The construction of the wall is incomplete, and the support legs of many chairs are not identified. The overall mapping effect of the Hector algorithm is poor, with large areas of ghosting, and the support frame of the chair is basically not recognised. The Karto algorithm can identify the surrounding walls, but ghosting appears in the red circle with low mapping accuracy. Compared with the previous three algorithms, the proposed improved RBPF-SLAM algorithm can completely scan the wall contour in the complex meeting room. It has high recognition rate for the supporting legs of chairs, no ghosting, and has high precision for building details [26]. The accuracy of the mapping effect in Experiment 3 is measured, and the feature position is selected to compare with the actual accuracy, as shown in Table 3 and Fig. 11.

Table 3 and Fig. 11 represent a visual comparison of the accuracy of the graph construction details of the three algorithms in a complex conference room. Although the Gmapping algorithm can roughly build the map outline, there is a large error in some obstacle details, and the overall error is larger than that of the Karto algorithm and the improved RBPF-SLAM algorithm. The Karto algorithm has ghosting in construction, but it is better than the Gmapping algorithm in the error of details.

Table 3
Errors of Experiment III

Algorithm		Gmapping		Karto		Improved	
Feature Points	Actual Value/cm	Measured Value/cm	Absolute Error/cm	Measured Value/cm	Absolute Error/cm	Measured Value/cm	Absolute Error/cm
1	57	64.3	5.3	53.6	3.4	59.1	2.1
2	41	47.8	6.8	45.6	7.6	42.5	1.5
3	370	382.9	12.9	379.8	9.8	378.7	5.2
4	43	46.5	3.5	39.9	3.1	44.8	1.8
5	39	42.2	5.2	43.5	4.5	42.1	3.1

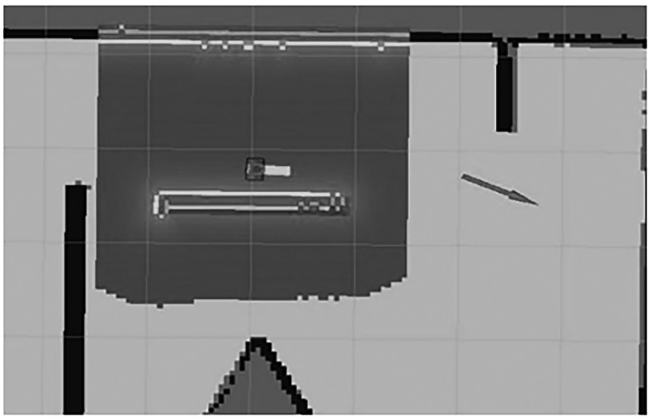

Figure 12. Local path planning of DWA algorithm.

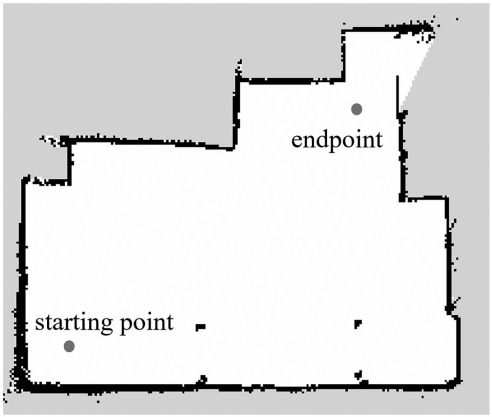

Figure 13. Global path planning of Experiment I map.

Compared with the previous two algorithms, the improved RBPF-SLAM algorithm has small overall error, high accuracy of map building, and can well detect the overall state of the environment.

5.2 Robot Navigation Experiment

5.2.1 Experiment of Local Path-Planning DWA Algorithm

In this navigation experiment, the local path-planning algorithm is the DWA algorithm.

In Fig. 12, the purple area is a local cost map where the robot samples multiple sets of speeds. The velocity sampled is the green segment in the plot that moves in the direction of the highest scoring velocity trajectory when it is sampled.

5.2.2 Experiment of Global Path-Planning Algorithm

Global path-planning experiments are performed on three maps obtained after the improved RBPF-SLAM laser SLAM experiments. The path-planning effect of the RRT algorithm, BI-RRT algorithm, and improved BI-RRT algorithm on these three maps are compared to summarise the data recording. Figures 13–15 are the three maps obtained by laser SLAM experiment. The starting point and end point of path planning are selected, respectively. Tables 4–6 show the route planning experiment data of the three algorithms on three maps. Each algorithm obtains average data values after 20 experiments on each map.

In Experiment 1, the map is an indoor room. Table 4 shows that in global path planning, the RRT algorithm takes the most time and has the longest path length. This is because the path planned by the RRT algorithm makes the robot rotate an average of 2.65 times during its movement, resulting in the increase in time and length. Due to the high number of folds, the BI-RRT algorithm also has an increase in time and length, with an average of 2.6 folds per turn. The improved BI-RRT algorithm can increase the selection of optimal nodes, make the path smooth and reduce the number of turns, with an average of 0.45 times each time, thereby reducing time and shortening the path length.

The map of Experiment 2 is a corner corridor, and the overall environment is longer and has corners. The number of global path folding for the three algorithms increases. The RRT algorithm averages 7.55 folds per turn, takes up to 31.25 s, and has a path length of 18.14 m. In general, the BI-RRT algorithm is slightly better than the RRT algorithm, which can reduce the number of folds by 0.35, save 1.35 s of time, and reduce the path length by 0.83 m. Compared with the first two algorithms, the improved BI-RRT algorithm can reduce the average number of folds

Table 4
Global Path Planning Data of Experiment I Map

Path-Planning Algorithms	Time (s)	Length (m)	Number of Folds
RRT algorithms	6.95	5.97	2.65
BI-RRT algorithms	6.9	5.96	2.5
Improved algorithms	4.55	5.24	0.45

Table 5
Global Path-Planning Data of Experiment II Map

Path-Planning Algorithms	Time (s)	Length (m)	Number of Folds
RRT algorithms	31.25	18.14	7.55
BI-RRT algorithms	29.9	17.31	7.2
Improved algorithms	26.15	16.42	4.75

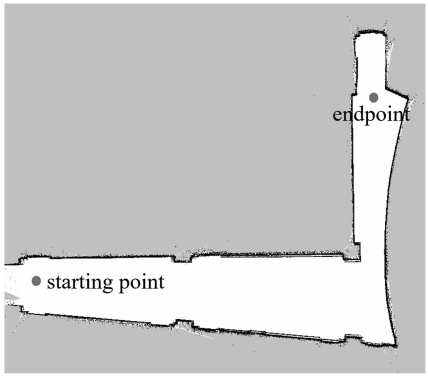

Figure 14. Global path planning of Experiment II map.

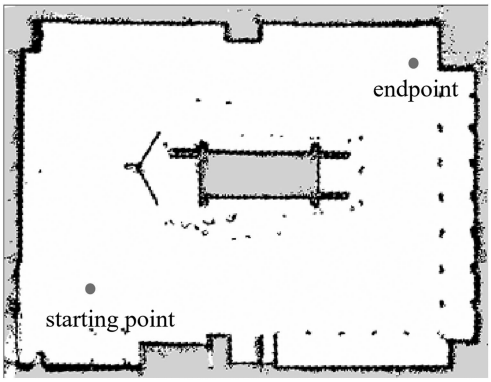

Figure 15. Experiment III map.

to 4.75, save 26.15 s, and has a path length of 16.42 m, which is the best in the three algorithms.

The map of Experiment 3 is a conference room with many obstacles, which tests the planning ability of the algorithm under multiple obstacles. The RRT algorithm and the BI-RRT algorithm have poor planning results, with more number of folds, more time consumption, and longer paths. Compared with the previous two algorithms, the improved BI-RRT algorithm has advantages in all aspects, with less folding, less time, and shorter paths.

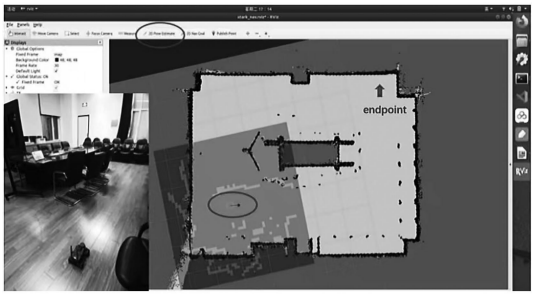

Figure 16. Initial position and pose of the navigation robot.

5.2.3 Robot Navigation Process

The effectiveness and practicability of the improved BI-RRT algorithm are proved by path-planning experiments. Next, the autonomous navigation process of the robot is carried out in a known map built by laser SLAM.

As shown in Fig. 16, the lower left corner is the position and pose of the robot in the actual environment. The red circle in the top menu bar is 2DPoseEstimate, which determines the initial position and pose of the robot. The red circle in the map is the determined initial position and pose, and the shaded square around the robot is the local cost map, which is the scope of local path planning. The navigation endpoint in the upper right corner of the map is selected for the robot to operate autonomous navigation.

Figures 17 and 18 show the initial position of the robot and the planning information during the movement. The yellow line segment in front of the robot is the trajectory of the local path-planning DWA algorithm, if the robot is too close to an obstacle during the movement, the DWA algorithm will avoid the obstacle. Fig. 19 below shows that the robot navigates to the endpoint and completes the navigation process.

Table 6
Global Path-Planning Data of Experiment III Map

Path-Planning Algorithms	Time (s)	Length (m)	Number of folds
RRT algorithms	17.95	11.29	4.85
BI-RRT algorithms	17.7	11.05	4.7
Improved algorithms	14.55	8.7	2.5

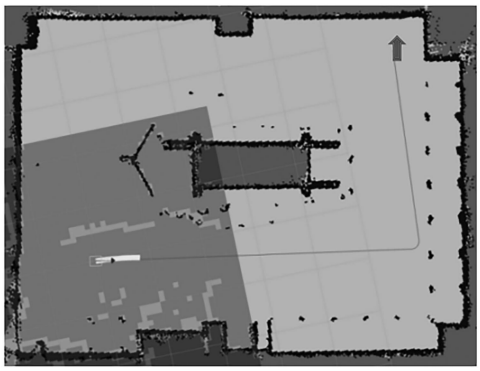

Figure 17. Initial planning information.

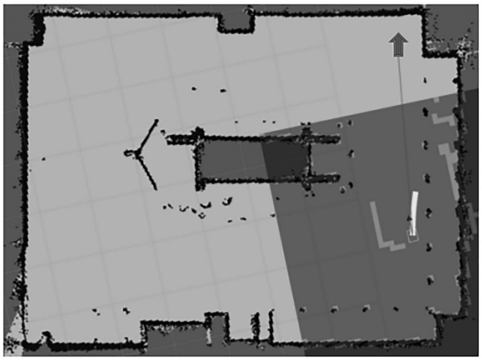

Figure 18. Motion process planning information.

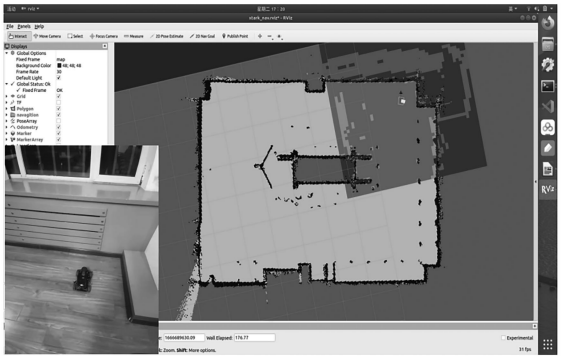

Figure 19. Position and pose of the end point.

6. Conclusion

Three sets of experiments are carried out for laser SLAM. The mapping effect and detail accuracy of the Gmapping algorithm, Hector algorithm, Karto algorithm, and improved RBPF-SLAM algorithm are compared in each set of experiments. The results show that the improved RBPF-SLAM algorithm has the advantages of less rough edges, no double image, and less precision error. The main content of navigation is divided into local path planning and global path planning. Local path planning uses the DWA algorithm to achieve local obstacle avoidance in navigation to keep the robot away from obstacles. The global path planning compares the planning effects of the RRT algorithm, BI-RRT algorithm, and the improved BI-RRT algorithm. Experiments are carried out on three maps, and each algorithm takes the average value of data after 20 experiments on each map. The data comparison verifies that the improved BI-RRT algorithm has significant advantages in the number of folds, path length, and time consumption. Finally, the specific process of navigation is introduced, which is achieved by command control of the SSH-connected robot on the computer. The results show that compared with the RRT algorithm and BI-RRT algorithm, the improved algorithm has faster planning speed, fewer path nodes, and improved path smoothness, which can quickly provide an optimal path for robot navigation.

References

[1] Y. Cong, C. Chen, B. Yang, F. Liang, R. Ma, and F. Zhang, CAOM: Change-aware online 3D mapping with heterogeneous multi-beam and push-broom LiDAR point clouds, *ISPRS Journal of Photogrammetry and Remote Sensing*, *195*, 2023, 204–219. DOI: https://doi.org/10.1016/j.isprsjprs.2022.11.017

[2] S. Zeng and K. Liu, Research status and development trend of UAV path planning algorithms, *Journal of Physics: Conference Series*, *2283*(1), 2022, 012004. DOI:https://doi.org/10.1088/1742-6596/2283/1/012004

[3] Y.J. Cui, Y.C. Wang, Z. He, D. Cao, L. Ma, and K. Li, Global path planning of kiwifruit harvesting robot based on improved RRT algorithm, *Transactions of the Chinese Society of Agricultural Machinery*, *53*(6), 2022, 151–158.

[4] D. Chen and X. Wu, Research on improved laser SLAM algorithm for mobile robots, *Computer Engineering and Application*, *58*(4), 2022, 163–168.

[5] S. Wen, Z. Wen, D. Zhang, H. Zhang, and T. Wang, A multi-robot path-planning algorithm for autonomous navigation using meta-reinforcement learning based on transfer learning, *Applied Soft Computing*, *110*(4), 2021, 107605. DOI: https://doi.org/10.1016/j.asoc.2021.107605

[6] B.G. Sarmina and G. Khachaturov, QPA*: Design of a searching and path planning algorithm for intelligent agents in two dimensions, *Proc. IEEE 19th International Conf. on Cognitive Informatics & Cognitive Computing (ICCI*CC)*, Beijing, 2020, 202–209. DOI: https://doi.org/10.1109/ICCICC50026.2020.9450270

[7] M. Nazarahari, E. Khanmirza, and S. Doostie, Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm,

Expert Systems with Applications, *115*, 2019, 106–120. DOI: https://doi.org/10.1016/j.eswa.2018.08.008

[8] U. Orozco-Rosas, O. Montiel, and R. Sepúlveda, Mobile robot path planning using membrane evolutionary artificial potential field, *Applied Soft Computing*, *77*, 2019, 236–251. DOI: https://doi.org/10.1016/j.asoc.2019.01.036

[9] G.S. Zhang, J.J. Wei, J.Q. Liu, C.B. Wang, T. Wang, Z.X. Mao, C.K. Luo, Path planning strategy for mobile robot navigation, *Mechanical & Electrical Engineering Technology*, *50*(04), 2021, 14–24.

[10] Y. Song, Q.L. Li, and Y.F. Kang, Conjugate unscented fastSLAM for autonomous mobile robots in large-scale environments, *Cognitive Computation*, *6*(3), 2014, 496–509. DOI: https://doi.org/10.1007/s12559-014-9258-z

[11] G. Grisetti, C. Stachniss, and W. Burgard, improved techniques for grid mapping with rao-blackwellized particle filters, *IEEE Transactions on Robotics*, *23*(1), 2007, 34–46. DOI: https://doi.org/10.1109/TRO.2006.889486

[12] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, G^2o: A general framework for graph optimization, *Proc. IEEE International Conf. on Robotics and Automation, Shanghai,* 2011, 3607–3613. DOI: https://doi.org/10.1109/ICRA.2011.5979949

[13] R. Zhang, J. Wu, and Y. Wang, Stability analysis of a novel mobile spray-painting robot for touch-up painting in vehicle repair plant, *Journal of Mechanical Science and Technology*, *36*(5), 2022, 2571–2584. DOI: https://doi.org/10.1007/s12206-022-0438-6

[14] J. Wu, X. Wang, B. Zhang, and T. Huang, Multi-objective optimal design of a novel 6-DOF spray-painting robot, *Robotica*, *39*(12), 2021, 2268–2282. DOI: https://doi.org/10.1017/S026357472100031X

[15] J. Wu, B. Zhang, L. Wang, and G. Yu, An iterative learning method for realizing accurate dynamic feedforward control of an industrial hybrid robot, *Science China Technological Sciences*, *64*(6), 2021, 1177–1188. DOI: https://doi.org/10.1007/s11431-020-1738-5

[16] R. Edlinger and A. Nüchter, Terrain prediction with a low-cost LIDAR sensor for mobile robots, *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *48*, 2022, 81–86. DOI: https://doi.org/10.5194/isprs-archives-XLVIII-2-W1-2022-81-2022

[17] Y. Wang and X. Wang, Research on SLAM road sign observation based on particle filter, *Computational Intelligence and Neuroscience*, *2022*, 2022, 4478978. DOI: https://doi.org/10.1155/2022/4478978

[18] L. Chen, A. Yang, H. Hu, and W. Naeem, RBPF-MSIS: Toward Rao-Blackwellized particle filter SLAM for autonomous underwater vehicle with slow mechanical scanning imaging sonar, *IEEE Systems Journal*, *14*(3), 2020, 3301–3312. DOI: https://doi.org/10.1109/JSYST.2019.2938599

[19] X.J. Li, D. Wang, and D.T. Li, Design of indoor mobile robot based on ROS and SLAM, *Digital Design, 1*(11), 2022, 19–23.

[20] A.H. Tan, A. Al-Shanoon, H. Lang, and Y. Wang, Mobile robot docking with obstacle avoidance and visual servoing, *International Journal of Robotics and Automation*, *38*(2), 2023, 97–108. DOI: https://doi.org/10.2316/J.2023.206-0782

[21] X.Y. Yang and H. Yan, Rapidly-exploring random trees algorithm with guided extension, *Modern Computer, 36*, 2020, 58–63. DOI: https://doi.org/10.3969/j.issn.1007-1423.2020.36.011

[22] G.C. Liu, P.C. Shi, Ni, Xuan, T.N. Liang, An improved bidirectional rapidly expanding random tree algorithm for path planning of unmanned vehicles, *Journal of Anhui Polytechnic University*, *36*(04), 2021, 41–50. DOI: https://doi.org/10.3969/j.issn.2095-0977.2021.04.007

[23] H.F. Wang, Y.Y. Cui, M.F. Li, and G.Y. Li, Path planning algorithm for mobile robot based on improved RRT * FN, *Journal of Northeastern University(Natural Science)*, *43*(9), 2022, 1217–1224+1249.

[24] L.J. Chen, W.L. Zhao, and J.J. Lou, RRT-based dynamic avoidance trajectory planning algorithm, *Modern Computer, 22*, 2021, 72–76+80.

[25] L. Hong, C. Song, P. Yang, and W. Cui, Two-layer path planner for AUVs based on the improved AAF-RRT algorithm. *Journal of Marine Science and Application*, *21*(1), 2022, 102–115. DOI: https://doi.org/10.1007/s11804-022-00258-x

[26] Y. Li, W. Liu, and X. Lei, Charging trajectory planning and motion control for indoor mobile robots, *International Journal of Robotics and Automation*, *37*(6), 2022, 520–528. DOI: https://doi.org/10.2316/J.2022.206-0527

Biographies

Xiaoyang Hu was born in Shenyang, Liaoning, in 1979. He received the Ph.D. degree in engineering, completed his doctoral defense in 2009 from the Institute of Mechanical Engineering, Russian Academy of Sciences, majoring in "Rigid Body Deformation Mechanics." He is an Associate Professor and a Master Tutor with the School of Equipment Engineering.

Sairu Liu was born in 2000. She is currently pursuing the graduation degree majoring in weapons science and technology with Shenyang Ligong University.

Jie Zhao was born in 1995. He is currently pursuing the graduation degree with Shenyang Ligong University.