Optimization of the Grid Mapping Algorithm for Mobile Robots

Ruoxi Wu¹², Leizheng Shu¹*, Xin Zhao²
¹Key Laboratory of Space Utilization, Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, Beijing, China
²Changchun University of Science and Technology, College of Electronic and Information Engineering, Changchun, Jilin, China
m1816895719@163.com

Abstract. Navigation precision, which is an important performance index of mobile robots, is closely related to the precision of grid mapping (Gmapping) algorithm. The modeling of the Gmapping algorithm is completed on the basis of Rao-Blackwellized particle filter (RBPF), with a light detection and ranging (LiDAR) robot as the research background, and the factors that influence the accuracy error of the Gmapping algorithm are analyzed. Firefly algorithm is introduced into the Gmapping system to complete the modeling and simulation analysis. The simulation results show that applying the firefly algorithm to the Gmapping system can effectively smooth the fluctuations of position, attitude angle, and other data output using sensors, which can decrease the number of particles and reduce object feature degradation. This method can effectively improve the accuracy of the Gmapping algorithm, reduce navigation error, and provide reference and guidance for the Gmapping system design of mobile robots.

1. Introduction
A mobile robot that uses light detection and ranging (LiDAR) as a sensor has an extensive application prospect in commercial and civil fields given the advantages of high precision, wide range and rapid transmission [1-4]. A simultaneous localization and mapping (SLAM) subsystem is an important part of a mobile robot system, which is the premise and guarantee for the normal work of mobile robots. The SLAM subsystem based on the LiDAR technology is called LiDAR SLAM, which is extensively used by grid mapping (Gmapping) algorithm [5-8]. In 2009, Oxford University's Mobile Robotics Group debuted its first driverless Wildcat in public, thus completing a test of the SLAM subsystem’s performance. Gmapping is the first step in starting the operation of the mobile robot system, and its importance is self-evident. Moreover, the accuracy of the Gmapping algorithm directly determines the accuracy of navigation and path planning.

On the basis of the known odometer, inertial measurement unit (IMU) and LiDAR data, the localization and mapping of the Gmapping algorithm are completed by modeling with the Rao-Blackwellized particle filter (RBPF) [9-12]. LiDAR can realize 2D plane scanning within a range of distances and angles, thereby generating spatial plane point cloud information. Relatively large Gmapping errors (i.e., navigation errors) are mainly attributed to the cumulative errors in the odometer and the influences (e.g., considerable calculation and particle degradation) of the RBPF algorithm. To improve the accuracy of the Gmapping algorithm and reduce navigation errors, an optimization
algorithm can be used to process the considerable calculation and particle degradation of the RBPF, thus improving the particle distribution and ensuring the diversity of particles. The commonly used optimization algorithms include gradient descent, genetic algorithm, ant colony algorithm and firefly algorithm [12-14].

The gradient descent method is used to solve the least square problem and is often used in neural networks. The genetic algorithm is used to solve the problems of search and is often used in machine learning. The ant colony algorithm is used to find the optimal path and is often used in combinatorial optimization. The firefly algorithm, as a new intelligent optimization algorithm, has been extensively applied in the almost all disciplines and engineering, such as robot system, image processing, industrial optimization, multi-signal source positioning and detection system, since it was proposed by the Cambridge scholar Xin-sha Yang in 2009 [15-20]. On the basis of the characteristics of the firefly technology, applying the firefly algorithm to the Gmapping algorithm will improve the performance of the Gmapping system.

This study uses the LiDAR robot as the research background, completes the modeling of the Gmapping on the basis of RBPF theory, and proposes to apply the firefly algorithm to the Gmapping system to improve the accuracy of the LiDAR SLAM. The simulation results show that the Gmapping precision can be greatly improved after the optimization algorithm is applied, thereby indicating that the firefly algorithm can considerably reduce the navigation error of the mobile robot system, improve the accuracy of the path planning of the Gmapping system, and play a guiding and reference role in designing the mobile robot system.

2. Gmapping algorithm principle for mobile robots

2.1. Composition of the mobile robot system

The mobile robot is an electromechanical device. From the perspective of control, the mobile robot system can be divided into the following four parts: executive mechanism, driving system, sensing system and control system. Figure 1 illustrates the principle of the robot system.

Fig. 1. Schematic of the robot system

The executive mechanism is a mechanical device of a work object that is oriented directly. The driving system is used to operate the executive mechanism and convert the command output using the control system into signals required by the executive mechanism. The internal and external sensing systems mainly realize signal input and feedback. The internal sensing system, which detects the status of position and attitude through its own signal feedback, includes odometer, gyroscope, accelerometer. The external sensing system includes cameras and LiDAR to detect information on the external environment of robots. The control system realizes task and information processing. Among these parts, the control system mainly includes three processes, namely, localization, mapping, and path planning. Figure 2 demonstrates the functional diagram of the control system.
2.2. **Gmapping algorithm principle**

The SLAM technology is mainly used to solve the problems of real-time localization and mapping of mobile robots in an unknown environment. In the SLAM technology, the classical algorithm is Gmapping algorithm, which is mainly introduced in the next subsection.

2.2.1 Working process

The Gmapping algorithm can dynamically generate 2D Gmapping by acquiring scanned LiDAR, IMU, and odometer data. Figure 3 depicts a flowchart of the Gmapping algorithm. The robot acquires its surrounding information through the LiDAR, its attitude information through the IMU, and its position information through the odometer; thereby preparing the mapping for the next moment. The RBPF uses the information of observation and sensor output to predict the position and attitude of the particle at the next moment. The scan matching algorithm is performed between the position and attitude of the robot obtained in real time and the position and attitude predicted by the RBPF, thus obtaining the optimal position and attitude of each particle and preparing for particle weight update. According to the optimal position and attitude obtained by scan matching, the particle filter takes several samples around the optimal position and attitude, and its mean and variance satisfy the Gaussian distribution (called the proposed distribution). The weight of each particle is updated and then normalized. The threshold value is used to determine whether a resampling is required (the threshold value is generally less than 2/3×N, where N is the total number of particles), and the mapping of each particle is updated to complete the SLAM process.

![Fig. 3. Flowchart of the Gmapping algorithm](image-url)
2.2.2 RBPF principle

The key idea of the SLAM is to estimate the joint posterior probability density function based on the observed value and sensor output information. The key and difficult problem of the SLAM technology is to complete the functions of simultaneous localization and mapping. From the perspective of mathematics, the RBPF algorithm can factor the joint posterior probability density function, as expressed in Formula (1). Therefore, the RBPF can estimate the position and attitude of the robot and then the mapping according with the known position and attitude. In other words, the SLAM thought is transformed into the ideas of positioning firstly and then mapping. According to the probability density function of the mapping, the mapping strongly depends on the position and attitude of the robot.

\[
p(x_{1:t}, m \mid z_{1:t}, u_{1:t-1}) = p(m \mid x_{1:t}, z_{1:t}) \cdot p(x_{1:t} \mid z_{1:t}, u_{1:t-1})
\]

(1)

where \( u_{1:t-1} = u_1, \cdots, u_{t-1} \) is the information output by the sensor; \( z_{1:t} = z_1, \cdots, z_t \) is the measured value, which is the scanning value of the LiDAR; \( x_{1:t} = x_1, \cdots, x_t \) is the real position and attitude of the robot; and \( m \) is the map.

The RBPF algorithm is derived from the Monte Carlo algorithm. The core idea of the RBPF algorithm is to use the weighted sum of a series of random samples to approximate the posterior probability density function. Namely, the probability of an event replaced with the frequency of its occurrence. The algorithm is divided into the following four stages: prediction, correction, resampling, and mapping estimation.

1) Prediction stage: This stage consists of three processes, namely, prediction particle, scan matching and proposed distribution.

2) Correction stage: With the real-time output of the observed values, the importance weight value of each particle is calculated. The calculations of the importance weight value are presented in Formulas (2) and (3).

3) Resampling stage: The sampled particles are redistribute in accordance with the weight values.

4) Mapping estimation: A map of the robot’s environment is estimated on the basis of the positions, attitudes and observations of the sampled particles.

\[
\pi x_{1:t} \mid z_{1:t}, u_{1:t-1} = \pi x_t \mid x_{1:t-1}, z_{1:t}, u_{1:t-1} \cdot \pi x_{1:t-1} \mid z_{1:t-1}, u_{1:t-2}
\]

(2)

\[
\pi x_{1:t} \mid z_{1:t}, u_{1:t-1} = \frac{p x_{1:t} \mid z_{1:t}, u_{1:t-1}}{p x_t \mid z_{1:t}, u_{1:t-1}} = \frac{\eta p z_t \mid x_{1:t} \cdot p x_t \mid x_{t-1}, u_{t-1} \cdot p x_{1:t-1} \mid z_{1:t-1}, u_{1:t-2}}{\pi x_t \mid x_{1:t-1}, z_{1:t}, u_{1:t-1} \cdot \pi x_{1:t-1} \mid z_{1:t-1}, u_{1:t-2}}
\]

(3)

\[
\pi x_{1:t} \mid z_{1:t}, u_{1:t-1} \propto \frac{p z_t \mid m_{t-1}, x_{1:t} \cdot p x_t \mid x_{t-1}, u_{t-1} \cdot p x_{1:t-1} \mid z_{1:t-1}, u_{1:t-2}}{p z_t \mid z_{1:t-1}, u_{1:t-1}} \cdot w_{t-1}
\]

where \( \mu_k = \frac{1}{p z_t \mid z_{1:t-1}, u_{1:t-1}} \).
2.3. Error analysis
The abovementioned principle of the Gmapping implies that the positioning and mapping errors of the robot are caused by the cumulative errors of the odometer, considerable calculation and memory consumption, particle degradation, and other error factors and influences, thus affecting the navigation effect. Considerable calculation and memory consumption are caused by numerous particles in the RBPF. The particle degradation is caused by frequent resampling of the RBPF. To improve the accuracy of navigation, the firefly algorithm introduced into the Gmapping algorithm can effectively smooth data fluctuations, decrease the number of particles, reduce particle degradation phenomena, and enhance the detail presentation of the mapping. These advantages are consistent with the real environment of the robot.

3. Firefly algorithm introduction

3.1. Firefly algorithm principle analysis
The firefly algorithm was proposed by a Cambridge scholar in 2009 on the basis of the luminous behavior of fireflies. It is a bionic intelligent optimization algorithm. The optimization idea comes from the attraction and movement of individual fireflies. In the present study, the firefly algorithm is used to enable the sampled particles move to the high likelihood region to improve the distribution, decrease the number, and reduce the degradation of particles.

The principles of the firefly algorithm are presented as follows:

1) The formula for relative brightness between Fireflies a and b is expressed as follows:

\[ I = I_0 \times e^{-\gamma r_{ab}} \]  

(4)

where \( I_0 \) is the maximum brightness value of the firefly; \( \gamma \) is the absorption coefficient of light intensity, which decreases with the increase in distance and the absorption of a transmission medium; and \( r_{ab} \) is the spatial distance between Fireflies a and b.

2) The formula for the attraction between firefly a and firefly b is expressed as follows:

\[ \beta = \beta_0 \times e^{-\gamma r_{ab}^2} \]  

(5)

where \( \beta_0 \) is the maximum attraction of the firefly; \( \gamma \) and \( r_{ab} \) have been defined in Formula (4).

3) The updated formula of the position in which Firefly a is attracted to move toward Firefly b is expressed as follows:

\[ x_a = x_b + \beta \times (x_b - x_a) + \alpha \times \left( rand - \frac{1}{2} \right) \]  

(6)

where \( x_a \) and \( x_b \) are the spatial positions of Fireflies a and b, correspondingly; \( \alpha \in 0,1 \) is the step factor; and \( rand \) is a random number subject to uniform distribution on [0,1] to avoid falling into local optimization.

The two most important parameters of the firefly algorithm are relative brightness and attraction. The former reflects the advantages and disadvantages of the fireflies and determines the movement direction between these fireflies, whereas the latter determines the range of the firefly movement. Particle optimization can be realized by constantly updating brightness and attraction. Figure 3 exhibits the flowchart of the firefly algorithm.
3.2. Modeling
In the update formula of the position in the principle of the firefly algorithm, the calculation amount of this algorithm is very huge, which is not conducive to the implementation efficiency of the algorithm. The main reason is that in each position update, the attraction between particle a and each particle b must be recalculated in accordance with the maximum attraction, light intensity absorption coefficient, distance, and other parameters. In this study, an improved scheme using global optimal particles to interact with one another is proposed. This scheme can effectively reduce computational complexity.

The sampled particles are used as firefly individuals, and the firefly brightness is selected as the current weight of the particle. First, in accordance with the weight value of the particle, the global optimal state particle $p_{best}$ is selected. The selection principle is that the particle with the maximum weight value is regarded as the optimal particle. Thereafter, the formulas of the improved particle attraction and position update are expressed as follows:

$$\beta = \beta_0 \times e^{-\gamma r_a^2}$$  \hspace{1cm} (7)

where $r_a$ is the spatial distance between Particles a and $p_{best}$, $\beta_0$ is the maximum attraction of the firefly, and $\gamma$ is the absorption coefficient of the light intensity.

$$x_t^{(a)} = x_t^{(a)} + \beta \times (p_{best} - x_t^{(a)}) + \alpha \times (rand - \frac{1}{2})$$  \hspace{1cm} (8)

where $x_t^{(a)}$ is the state value of Particle a at time t, $\beta$ is the attraction between particles, $\alpha \in 0,1$ is the step factor, rand is a random number subject to uniform distribution on [0,1], and $p_{best}$ is the global optimal particle.

The flowchart of the RBPF Gmapping algorithm on the basis of the firefly algorithm optimization is displayed in Figure 5, with the following specific steps:

Step 1: Estimate the initial position and attitude of the robot in accordance with the position and attitude $x_{t-1}$ and the sensor control information $u_{t-1}$. 
Step 2: Perform the scan matching algorithm.
Step 3: Sample in the proposed distribution to update the current particle.
Step 4: Calculate the particle weight value.
Step 5: Calculate the global optimal particle $p_{best}$, and the attraction between Particles $a$ and $p_{best}$, as expressed in Formula (7).

Step 6: Update the particle state in accordance with the improved position update expressed in Formula (8).
Step 7: Calculate the optimized particle weight and conduct normalization processing.
Step 8: Resample.
Step 9: Update the particle map.

---

4. Simulated analysis

4.1. Simulation conditions and parameter setting

In this study, a 2D LiDAR backpack at the Deutsches Museum was used as the dataset. The dataset gives the polar coordinates of the LiDAR scanning point cloud to the center of the mobile robot, that is, the angles and distances of the point cloud deviate from the center of the mobile robot. MATLAB software was utilized for the simulation experiment. Three groups of experiments were conducted on the robot. The experimental conditions are respectively without the filtering, application of the RBPF, combination of the RBPF and firefly algorithm. The initial simulation conditions are presented as follows:

1) The number of the scan is 5522.
2) The number of the point in a scan is 1079.
3) The maximum attraction $\beta_0$ is 1.
4) The step factor $\alpha$ is 0.001.
5) The maximum absorption coefficient of light intensity $\gamma$ is 1.
6) The particle number of the RBPF is 20.
7) The iteration number of the firefly algorithm is 10.
8) Process noise: The mean is $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$, and the variance is $\begin{bmatrix} 10^{-3} & 0 & 0 \\ 0 & 10^{-3} & 0 \\ 0 & 0 & 10^{-3} \end{bmatrix}$.

---

Fig. 5 Flowchart of the Optimize Gmapping algorithm
9) Observation noise: The mean is \( \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \), and the variance is \( \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix} \).

4.2. Result and analysis

Figure 6 presents the results from the above-mentioned simulation conditions. (a) is the Gmapping without the RBPF and firefly algorithm, (b) is to enlarge the map without the RBPF, (c) is the Gmapping in the case of the RBPF, (d) is to enlarge the map in the case of the RBPF, (e) is the Gmapping in the case of the RBPF and firefly algorithm, and (f) is to enlarge the map in the case of the RBPF and firefly algorithm. The map has a white background, purple line of mapping, black line of walking path of the robot, and green line of laser scanning.

Figure 6 illustrates that among the three cases, the effect of the maps inside the rectangular box is evidently different, and that for (e) and (f) is the optimal. The simulation results show that the adding of the firefly algorithm into the RBPF can effectively reduce the number of particles, enhance the clarity of the mapping, and improve the accuracy and navigation effect of the Gmapping algorithm.

---

Fig. 6. Gmapping of a robot
The matching information of the grid map is further analyzed under three different simulation conditions. In the grid map, the predicted scanned point cloud is matched with the point cloud of the known dataset, and the distance matrix from the useless point cloud to the available point cloud is calculated as the column vector, whose dimension is $5522 \times 1$. By summing the distance matrix, the overlap ratio of the match can be obtained, which is the so called evaluation function: the smaller the distance is, the higher the overlap ratio of the two maps will be. The simulation results are shown in Figure 7. (a) is the image of overlap ratio without the filtering; (b) is the image of overlap ratio under the RBPF, and (c) is the image of overlap ratio under the RBPF and firefly algorithm. It can be seen from the images that by applying the firefly algorithm, the overlap ratio of the map is relatively low, which can improve the matching effect of the map, and further improving the positioning and mapping ability of Gmapping.
5. Conclusion

Navigation precision is an important performance index of mobile robots. The improvement of navigation accuracy is crucial for the SLAM performance of LiDAR. This study investigates the methods for improving the navigation accuracy, and presents the RBPF and firefly algorithm models. The validity of the method is verified by simulation. In addition, the Gmapping performance and navigation precision are improved considerably, thereby enhancing the clarity of the map. After appropriate modifications in the particle filter and firefly algorithm models, the technology that is the combination of these two algorithm models can also be applied to the SLAM technology, such as multi-robot, unmanned aerial vehicle, and unmanned ship. These models provide guidance and reference for the SALM system design.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant No.11603057).

References

[1] Y. Cheng, G. Wang, “Mobile Robot Navigation Based on Lidar,” Chinese Control And Decision Conference (CCDC). IEEE, pp. 1243-1246 (2018).
[2] J. Park, J. Y. Kim, et al., “Global Map Generation using LiDAR and Stereo Camera for Initial Positioning of Mobile Robot,” International Conference on Information and Communication Technology Robotics, pp. 1-4 (2018).
[3] D. Ghorpade, A. D. Thakare, S. Doiphode, “Obstacle Detection and Avoidance Algorithm for Autonomous Mobile Robot using 2D LiDAR.” International Conference on Computing, Communication, Control and Automation. IEEE, pp. 1-6 (2018).
[4] R. Li, J. Liu, et al., “LIDAR/MEMS IMU integrated navigation (SLAM) method for a small UAV in indoor environments,” DGON Inertial Sensors and Systems. IEEE, pp. 1-15 (2015).
[5] K. Wongsuwan, K. Sukvichai, “Generalizing corrective gradient refinement in RBPF for occupancy grid LIDAR SLAM,” International Conference on Robotics and Biomimetics. IEEE, pp. 495-500 (2017).
[6] D. Shen, Y. Huang, et al., “Research and Implementation of SLAM Based on LIDAR for Four-Wheeled Mobile Robot,” International Conference of Intelligent Robotic and Control Engineering. IEEE, pp. 19-23 (2018).
[7] Y. Abdelrasoul, A. B. S. H. Saman, P. Sebastian, “A Quantitative Study of Tuning ROS
Gmapping Parameters and Their Effect on Performing Indoor 2D SLAM,” International Symposium on Robotics & Manufacturing Automation. IEEE, pp. 1-6 (2017).

[8] B. L. E. A. Balasuriya, B. A. H. Chathuranga, et al., “Outdoor robot navigation using Gmapping based SLAM algorithm,” Moratuwa Engineering Research Conference. IEEE, pp. 403-408 (2016).

[9] R. P. Guan, B. Ristic, et al., “Combining KLD-sampling with Gmapping proposal for grid-based Monte Carlo localization of a moving robot,” International Conference on Information Fusion. IEEE, pp. 1-8 (2017).

[10] G. Grisetti, C. Stachniss, W. Burgard, “Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling,” International Conference on Robotics & Automation. IEEE, pp. 2432-2437 (2005).

[11] G. Grisetti, C. Stachniss, W. Burgard, “Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters,” Transactions on Robotics. IEEE, 23(1), 34-46 (2017).

[12] M. Chen, C. Bian, Y. Wu, “Research on Rao-Blackwellised Particle Filter SLAM algorithm based on QPSO,” Chinese Control Conference, pp. 5179-5185 (2018).

[13] L. Qing, Z. Chao, et al., “Path planning of mobile robots based on specialized genetic algorithm and improved particle swarm optimization,” Proceedings of the 31st Chinese Control Conference, pp. 7204-7209 (2012).

[14] J. Lee, B. Choi, et al., “Comparison between heterogeneous ant colony optimization algorithm and Genetic Algorithm for global path planning of mobile robot,” International Symposium on Industrial Electronics. IEEE, pp. 881-886 (2011).

[15] L. Ma, P. Cao, “Comparative study of several improved firefly algorithms,” IEEE International Conference on Information & Automation, pp. 910-914, February 2017.

[16] S. K. Sarangi, R. Panda, et al., “A new modified firefly algorithm for function optimization” International Conference on Electrical. IEEE, pp. 2944-2949 (2016).

[17] M. Gupta, S.N. Tazi, et al., “Edge Detection Using Modified Firefly Algorithm,” International Conference on Computational Intelligence & Communication Networks. IEEE, pp. 167-173, March 2015.

[18] A. Sharma, S. Sehgal, “Image segmentation using firefly algorithm,” International Conference on Information Technology. IEEE, pp. 99-102, February 2017.

[19] C. Yu, B. Jin, et al, “Multi-threshold Image Segmentation Based on Firefly Algorithm” International Conference on Intelligent Information Hiding & Multimedia Signal Processing. IEEE, pp. 415-419, July 2014.

[20] S. Goel, V. K. Panchal, “Performance evaluation of a new modified firefly algorithm” International Conference on Reliability. IEEE, January 2015.