Laser SLAM optimization based on resampling

Shuyang Xu¹, Shukun Cao¹*, Kuizeng Gao² and Wenlong Cao², Jianzhong Ma², Song Chen¹, Chen Wang¹

¹School of mechanical engineering, University of Jinan, Jinan, Shandong, 250022, China
²Shandong juming machinery co. LTD, Zibo, Shandong, 256400, China
* xushuyang@mail.ujn.edu.cn

Abstract: In order to solve the particle degradation and particle exhaustion in the simultaneous positioning and map construction method of Rao-Blackwellized particle filter, an optimization method for simultaneous positioning and map construction is proposed. In order to alleviate the need for too much data in the re-sampling stage, the calculation time is longer, and the calculation amount is reduced by dropping some low-weight particles and reducing the number of particles. The experimental results show that compared with the traditional RBPF-SLAM method, the improved method can obtain a more accurate map with fewer particles, reduce the number of Kalman particle filtering, and reduce the amount of calculation to meet the needs of actual use.

1. Introduction

Simultaneous positioning and map construction of a mobile robot refers to the process in which the robot relies on its own sensors to perceive the surrounding environment in an unknown environment, and simultaneously estimates its own pose. Since mobile robots cannot perform accurate GPS positioning in indoor environments, SLAM, as a prerequisite for autonomous navigation, has been a research hotspot in the field of robotics in recent years.

According to the different sensors, SLAM can be divided into laser SLAM and visual SLAM[1]. Since lidar has the advantages of being less affected by light conditions and accurate ranging, it is currently the most widely used technology in the SLAM field.

The current research field divides the laser SLAM framework into four parts[2]: front-end scan matching, back-end optimization, closed-loop detection, and map construction. According to the different map format output, it is divided into three models based on raster map[3], topology map[4] and geometric map[5]. Among them, there are two methods based on filter and graph optimization in the back end. Murphy et al. combined the Rao-Blackwellization algorithm with the particle filter algorithm to make the robot state estimation shift from high-dimensional space to low-dimensional space, greatly reducing the amount of calculation.

The SLAM problem belongs to the optimal estimation problem of simultaneously estimating the pose of the mobile robot and the feature position in the environment. Researchers mainly use the method of probabilistic model to solve the SLAM problem, using methods such as extended Kalman filter and particle filter. EKF is the first method to solve the SLAM problem, but the EKF method has high computational complexity and is not suitable for multi-mode distribution. The PF-based SLAM method uses discrete random particles to represent the posterior probability density distribution of the state vector, estimates the possible position of the robot, and uses observations to weight each particle to increase the probability of the robot's most likely position. The literature [6] proposes a SLAM algorithm...
based on RBPF, which decomposes the SLAM problem into the product of the posterior probability of the robot path estimation and the posterior probability of the map part estimation, and solves the problem of excessively high dimension of the state space. It greatly improves the calculation efficiency. For robots equipped with laser sensors, the accuracy of laser information is much higher than that of odometers. Therefore, the literature [7] proposes an improved RBPF-SLAM algorithm, namely Gmapping algorithm.

2. Basic RBPF-SLAM algorithm description

The core idea of the traditional RBPF-SLAM algorithm is to decompose SLAM into two problems: robot positioning and map construction based on known robot poses[8]. The positioning robot estimates its own pose \( x_{t} = x_1, x_2, \cdots x_t \) by relying on sensor observation information \( z_{t} = z_1, z_2, \cdots z_t \) and odometer control information \( u_{t-1} = u_1, u_2, \cdots u_{t-1} \), that is, finding its posterior probability distribution \( p(x_t | u_{t-1}, z_t) \). When the robot’s own pose and sensor observation data are known, the map \( m \) can be solved in a closed form posterior probability distribution: 

\[
p(m | x_t, z_t) = p(m | x_t, z_t) \cdot p(x_t | u_{t-1}, z_t)
\]

Figure 1 is the implementation process of the RBPF-SLAM algorithm.

![Figure 1. Implementation process of Gaussian resampling algorithm](image)

The RBPF-SLAM algorithm implementation steps are as follows:

1) Initialization, at \( t=0 \), select \( N \) particles, and the weight of each particle is recorded as \( \omega_0^{(i)} = 1/N \)

2) Particle sampling, using the motion model as the recommended distribution \( \pi \), and then using the kinematics model \( p(x_t | x_{t-1}, u_t) \) for particle propagation, and the next generation particle set \( \{x_t^{(i)}\} \) is generated from the previous generation particle set \( \{x_{t-1}^{(i)}\} \).

3) Calculate the weight, the size of the weight \( \omega_t^{(i)} \) is calculated by the resampling formula.

\[
\omega_t^{(i)} = \frac{p(x_t^{(i)} | z_t, u_t)}{\pi(x_t^{(i)} | z_t, u_t)}
\]

4) Re-sampling. Copy high-weight particles according to the size of the weight, discard low-weight particles, and the new particles have the same weight. Grisetti et al[9] proposed an adaptive resampling algorithm \( N_{eff} \) to calculate the effective sample size to measure the degree of particle degradation. The smaller the value, the greater the variance of particle weights and the more severe the particle degradation.

\[
N_{eff} = 1 / \sum_{i=1}^{N} (\omega_t^{(i)})^2
\]

Where \( \omega_t^{(i)} \) is the normalized weight of the particles, and \( N \) is the number of particles. Generally, it can be set to resample at the time \( N_{eff} < N / 2 \).
3. Improved RBPF-SLAM algorithm

Aiming at the defect of the large amount of calculation of the RBPF-SLAM algorithm, this paper proposes a method to improve the real-time performance of the RBPF algorithm, using the importance of the particles to reduce the repeated calculation of the Kalman filter part. The original RBPF algorithm must calculate and update its probability distribution $p_i^{(t)}$ for every particle that exists, and constantly recalculate, $\mu_i^{(t)}$, $\sum_i^{(t)}$ and the inverse matrix $S_i^{-1(0)}$, resulting in a particularly large amount of calculation and poor real-time performance.

In the transmission particle filter, the particles with high weight are distributed in the high probability area of the observation posterior. The particles in the RBPF transform are in a discrete state. The existence range of these discrete particle groups in space is limited. The particles in the observation probability function area can be selected as representatives to calculate $\mu_i^{(t)}$ and $\sum_i^{(t)}$. At this time, the calculation amount of the Kalman filter in the RBPF-SLAM algorithm will be less than the actual particle N. Use the following formula for the recommended distribution of particle filtering, discard some low-weight particles, and reduce the number of particles to reduce the amount of calculation. The following formula is the recommended distribution spacing:

$$p(x_i | m_{t-1}^{(i)}, x_{t-1}^{(i)}, z_t, u_t) \approx \frac{p(z_i | m_{t-1}^{(i)}, x_{t-1}^{(i)}) \cdot p(x_i | x_{t-1}^{(i)}, u_{t-1})}{\int_{x \in \mathcal{L}^{(i)}} p(z_i | m_{t-1}^{(i)}, x) dx}$$  

(4)

$$L^{(i)} = \{x | p(z_i | m_{t-1}^{(i)}, x) > 2 / N\}$$

(5)

The improved RBPF algorithm adopts adaptive resampling technology, which is mainly divided into several steps such as prior sampling, calculating particle weights, adaptive resampling, and updating the map.

The improved algorithm flow chart is shown in Figure 2.

The algorithm flow in the figure 2: at time t-1, the sampling stage, $x_{t-1}^{(i)} \sim p(x_{t-1} | x_{t-2}^{(i)})$, get unweighted particles $\{p_{t-1}^{(i)}, \mathcal{N}^{-1}\}$, get weighted particles $\{p_{t-1}^{(i)}, w_{t-1}^{(i)}\}$; in the re-sampling stage, only keep high-weight particles, get $\{p_{t-1}^{(i)}, \mathcal{N}^{-1}\}$, Kalman filter only updates particles at different positions at this time. Reduce the calculation amount of $\mu_i^{(t)}$, $\sum_i^{(t)}$ Repeat sampling $x_t^{(i)} \sim p(x_i | x_{t-1}^{(i)})$ at time t to get next generation particles $\{p_t^{(i)}, \mathcal{N}^{-1}\}$. 

3. Improved RBPF-SLAM algorithm

Aiming at the defect of the large amount of calculation of the RBPF-SLAM algorithm, this paper proposes a method to improve the real-time performance of the RBPF algorithm, using the importance of the particles to reduce the repeated calculation of the Kalman filter part. The original RBPF algorithm must calculate and update its probability distribution $p_i^{(t)}$ for every particle that exists, and constantly recalculate, $\mu_i^{(t)}$, $\sum_i^{(t)}$ and the inverse matrix $S_i^{-1(0)}$, resulting in a particularly large amount of calculation and poor real-time performance.

In the transmission particle filter, the particles with high weight are distributed in the high probability area of the observation posterior. The particles in the RBPF transform are in a discrete state. The existence range of these discrete particle groups in space is limited. The particles in the observation probability function area can be selected as representatives to calculate $\mu_i^{(t)}$ and $\sum_i^{(t)}$. At this time, the calculation amount of the Kalman filter in the RBPF-SLAM algorithm will be less than the actual particle N. Use the following formula for the recommended distribution of particle filtering, discard some low-weight particles, and reduce the number of particles to reduce the amount of calculation. The following formula is the recommended distribution spacing:

$$p(x_i | m_{t-1}^{(i)}, x_{t-1}^{(i)}, z_t, u_t) \approx \frac{p(z_i | m_{t-1}^{(i)}, x_{t-1}^{(i)}) \cdot p(x_i | x_{t-1}^{(i)}, u_{t-1})}{\int_{x \in \mathcal{L}^{(i)}} p(z_i | m_{t-1}^{(i)}, x) dx}$$  

(4)

$$L^{(i)} = \{x | p(z_i | m_{t-1}^{(i)}, x) > 2 / N\}$$

(5)

The improved RBPF algorithm adopts adaptive resampling technology, which is mainly divided into several steps such as prior sampling, calculating particle weights, adaptive resampling, and updating the map.

The improved algorithm flow chart is shown in Figure 2.

The algorithm flow in the figure 2: at time t-1, the sampling stage, $x_{t-1}^{(i)} \sim p(x_{t-1} | x_{t-2}^{(i)})$, get unweighted particles $\{p_{t-1}^{(i)}, \mathcal{N}^{-1}\}$, get weighted particles $\{p_{t-1}^{(i)}, w_{t-1}^{(i)}\}$; in the re-sampling stage, only keep high-weight particles, get $\{p_{t-1}^{(i)}, \mathcal{N}^{-1}\}$, Kalman filter only updates particles at different positions at this time. Reduce the calculation amount of $\mu_i^{(t)}$, $\sum_i^{(t)}$ Repeat sampling $x_t^{(i)} \sim p(x_i | x_{t-1}^{(i)})$ at time t to get next generation particles $\{p_t^{(i)}, \mathcal{N}^{-1}\}$.
4. Realization of simulation research in ROS

Download the open source data set on the official website to simulate and analyze the RBPF-SLAM algorithm and the improved RBPF-SLAM algorithm. The process is shown in the figure 3.

Table 1. RBPF algorithm and improved RBPF algorithm error statistics

| Data set            | RBPF-SLAM | Improved RBPF-SLAM |
|---------------------|-----------|--------------------|
| Intel               | 0.065     | 0.052              |
| FABMAP              | 1.236     | 0.849              |
| NewCollege          | 1.114     | 1.058              |
| COLD                | 3.297     | 2.564              |
| Marulan             | 8.512     | 6.149              |
| St Lucia Multiple Times | 2.126   | 2.009              |

It can be concluded from Table 1 that in the same environment, the improved RBPF has higher...
accuracy than the map created by the comparison algorithm. At the same time, the number of resamples for creating consistent maps was compared in the FABMAP and Intel data sets.

| Two sets of experiments | FABMAP | Intel |
|-------------------------|--------|-------|
| Comparison algorithm    | 74     | 124   |
| Improve algorithm       | 56     | 99    |

Raster maps have a large amount of data, and colleagues need to copy a large amount of map data during the re-sampling stage. Yes, it takes a long time to run. The improved algorithm uses the method of discarding the bottom weight particles and reducing the number of particles, which can reduce the number of resampling. Therefore, the operating efficiency of the algorithm is improved. It can be concluded from the table and table that the improved RBPF-SLAM algorithm not only improves the accuracy of mapping, but also improves the efficiency of the algorithm.

5. Conclusion

Based on the research of the Rao-Blackwellized particle filter algorithm, this paper proposes a method to change the particle distribution and discard the unweighted particles to reduce the number of Kalman filters. In the process of resampling, the improved algorithm uses changing the particle distribution to obtain a new particle distribution, reducing the number of Kalman particle filtering, and improving the accuracy of the map construction and the efficiency of the algorithm. Finally, simulation analysis and experiments with open source datasets on ROS verify that the improved algorithm can create high-precision environmental maps.

Acknowledgments

This paper was supported by "Taishan industrial leading talent project" (Project No. tscy20160206), Jinan "Twenty items for Colleges and Universities" funded project (Project No. 2019GXRC019) and by the Natural Science Foundation of Shandong Province (Project No.ZR2019MEE087).

References

[1] WANG Y P, TANG X H. Visual SLAM loop closure detection based on improved ORB[J]. China Sciencepaper, 2018, 13(14):1621-1625. (in Chinese)
[2] CADENA C, CARLONE L, CARRILLO H, et al. Past, present, and future of simultaneous localization and mapping: toward the robust-perception age[J]. IEEE Transactions on Robotics, 2016, 32(6):1309-1332.
[3] KUNDU A S, MAZUMDER O, DHAR A, et al. Oc-cupancy grid map generation using 360° scanning xtion prolive for indoor mobile robot navigation [C]//Proceedings of 2016 IEEE First International Conference on Control, Measurement and Instrumentation. New York: IEEE, 2016:15804079.
[4] AKAI N, OZAKI K. A navigation method based on topological magnetic and geometric maps for outdoor mobile robots [C]/Proceedings of 2015 IEEE/SICE International Symposium on System Integration. New York: IEEE, 2015:15799275.
[5] RAMAITHITIMA R, WHITZER M, BHATTA CHARYAS, et al. Automated creation of topological maps in unknown environments using a swarm of resource-constrained robots [J]. IEEE Robotics and Automation Letters, 2016, 1(2):746-753.
[6] DOUCET A, de FRETAS N, Murphy K, et al. Rao-Blackwellized Particle Filtering for Dynamic Bayesian Networks [C]//Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence. New York, USA: ACMPress, 2000:176-183.
[7] GRISSETTI G, STACHNISS C, BURGARD W. Improved Techniques for Grid Mapping with Rao-blackwellized Particle Filters [J]. IEEE Transactions on Robotics, 2007, 23(1):34-46.
[8] CHEN JJ, LIAO XF, WU Y, et al. RAO-Blackwellised SLAM based on niched genetic optimized method[J]. Application Research of Computers, 2017, 34(8): 2368-2371. (in Chinese)

[9] SHI LQ, QIU M, WANG Y, et al. Evaluation of water inrush from under lying aquifers by using a modified water-inrush coefficient model and water-inrush index model: a case study in Feicheng coalfield, China [J]. Hydrogeology Journal, 2019, 27(6): 2105-2119.