RBPF algorithm combined with annealing optimization and genetic resampling

YiSun

Xi'an University of Science and Technology, School of Communication and Information Engineering

409298131@qq.com

XiaoXiaoZhang

Xi'an University of Science and Technology, School of Communication and Information Engineering  
zxx420@163.com

**ABSTRACT**

RBPF is an algorithm that effectively solves simultaneous positioning and mapping. The traditional RBPF algorithm uses a large number of particles and frequently performs resampling, resulting in particle degradation and reduced estimation ability, so that the constructed raster map is not accurate. Disadvantages, this paper combines the motion model of the robot with the observation model as its proposed distribution, and optimizes the hybrid proposed distribution with annealing parameters to make it more accurate. In the resampling process, it classifies the particles according to their weights and introduces adaptive genetics. The algorithm variation cross operation reduces the number of resampling and effectively maintains the particle diversity. Simulation verification is performed on MATLAB, and the Kobuki motion chassis is combined with the actual verification on the robot operating system (ROS). Experiments show that the algorithm can be used. Fewer particles accurately estimate the pose of the robot and create a more accurate raster map, while reducing computation time.

**Keywords**

Rao-Blackwellized particle filtering; simultaneous positioning and mapping; proposed distribution; resampling

# INTRODUCTION

In recent years, intelligent mobile robot technology has developed rapidly and has been applied to mines, security, family services and other fields, allowing robots to replace human beings to complete repetitive, boring, dangerous and even human unfinished work[[[1]](#endnote-1)].The robot is gradually becoming more intelligent and automated. With the development of artificial intelligence technology, the robot needs to have good positioning, mapping and path planning ability when assisting people to complete various tasks[[[2]](#endnote-2)]. The problem of robot positioning and mapping is complementary and inseparable, namely Simultaneous Localization and Mapping (SLAM). It combines robot positioning and mapping to lay the foundation for the next robot navigation[[[3]](#endnote-3)].

Most of the research on early SLAM technology is based on probability theory, mainly focusing on extended Kalman filter[[[4]](#endnote-4)]. In recent years, particle filters have been widely used in the field of robotics, and many studies have been conducted to solve the SLAM problem[[[5]](#endnote-5)].Murphy, Doucet et al. and Doucet introduced RBPF as an effective means to solve the SLAM problem[[[6]](#endnote-6)]. Rao-Blackwellized Particle Filter (RBPF) is more widely used for probabilistic estimation of robot position and environment than Extended Kalman Filter (EKF)[[[7]](#endnote-7)].Compared with EKF SLAM, RBPF has the advantage of robust multi-measurement data association, so RBPF SLAM will get better results than EKF SLAM when data is mis-associated[[[8]](#endnote-8)].

Literature[9], Murphy et al. used the RBPF algorithm as a new method to deal with the SLAM problem. It breaks down the SLAM problem into a two-part map estimate and pose estimation. The recorded position and laser scan data are used to estimate the position of the robot, and then the position and laser data are used again to update the map. However, there are still shortcomings such as decreased performance due to particle diversity loss and frequent resampling resulting in decreased particle diversity. Since then, many improved algorithms have been proposed:Literature[10]proposed a Rao-Blackwellized particle filter RBPF synchronization positioning and map construction optimization algorithm that incorporates the firefly algorithm. Using the firefly algorithm to improve the particle sampling process, the particle diversity is guaranteed to some extent. Literature[11]uses an RBPF-SLAM algorithm based on Gaussian distribution resampling. The particles are classified according to the weight of the particles, and the Gaussian distribution is used to disperse the high-weight particles to obtain new particles, which is not effective for mitigating particle degradation. In[12], the proposed particle distribution method is used to optimize the proposed distribution, so that the discrete particles in each region move to the high-like position of the center. The portion of dense particles remains unchanged, effectively slowing down particle degradation, but it is not effective for increasing particle diversity.

Based on the predecessors, this paper proposes that the traditional RBPF-SLAM algorithm has low distribution accuracy and the number of resampling results in the risk of depletion of particles, which reduces the diversity of particles and improves the RBPF. On the one hand, the motion model of the robot and the observation model are combined as the hybrid proposal distribution, and the annealing parameters are used to adjust the proportion of the two models in the mixed proposal distribution to improve the accuracy of the proposed distribution. On the other hand, for the number of resampling times, the particle degradation problem is classified according to the weight of the particles. For some particles, adaptive genetic algorithm cross mutation operation is introduced to generate new particles, reduce the number of resampling, and maintain the particles. Diversity. The improved algorithm in this paper can obtain more reliable pose estimation with fewer particles in less time, so that the grid map can be constructed more effectively and the path planning can be more efficient.

# THE BASIC PRINCIPLE OF RBPF-SLAM

SLAM mainly estimates the joint posterior probability density function based on sensor observation data and robot odometer data. Bayesian filtering is used to divide the formula into two processes: prediction and observation, which correspond to two models: the motion model and the observation model.Control the motion model of the mobile robot according to the input data obtained from the robot, or calculate the relative value of the current posture and the last moment of the robot encoder, the gyro motion detects the sensor data, calculates the last-time positioning result of the robot as a model input, and obtains the robot positioning. The prior probability distribution. The observation model is based on measurement data obtained by sensors such as a laser radar and other sensors on the mobile robot, and calculates the observed probability compared to the existing map.

SLAM uses Markov's hypothesis that the continuous motion of a mobile robot is separated into discrete system states by time, forming a Markov chain. At this time, the positioning result of the robot is used as the input of the next positioning algorithm. The sensor is used for real-time positioning of mobile and environmental ranging information, and the positioning result is used for real-time map construction. Unlike traditional location maps, SLAM can be used for real-time location and map construction without the need to complete map input in advance, and an environmental map will be generated during the movement and positioning of the robot. However, there are some difficulties in implementing SLAM, mainly to locate the robot, and a very accurate map must be established before. However, for a very accurate map to be built, the robot must know exactly where it is.

RBPF-SLAM is a particle filter based SLAM algorithm that uses particles to represent the position and attitude of the robot. Widely used in robotic simultaneous positioning and map construction. The RBPF (Rao-Blackwellized Particle Filter) algorithm uses factor (1) to factor factor the joint probability density function.



RBPF allows the location of the robot to be estimated using recorded ranging and laser scan data, and then the map is updated based on the position and laser data. Separate the pose estimation from the mapping. First, pose estimation is performed according to the motion model. The RBPF algorithm uses particle samples to represent the probability distribution of the positioning results, and each particle represents the possible pose of the robot. The map is updated in accordance with the obtained pose and the observation model.

The steps of the RBPF particle filter are as follows:

1. Initialization: When t=0, select N particles according to the prior probability  of the robot motion model, and record that the weight corresponding to each particle of  is .
2. Sampling: The next generation of particle sets is generated from the set of particles , according to the proposed distribution of π samples. The odometer motion model  is usually distributed as a proposed distribution π.
3. Calculating particle weights: Calculate the weight of each particle from equation (2) according to the principle of importance resampling:



1. Resampling: Calculate the number of effective particles according to equation (3) and set a threshold . When , resampling is performed, and after resampling, all particle weights are unified.



1. Update map: Update the corresponding map according to the pose  of the particle and the historical observation information : .

# IMPROVEMENT OF RBPF-SLAM ALGORITHM

## Adaptive Optimization Of Hybrid Distribution

For the resampling process, the next generation of particles needs to be sampled according to the proposed distribution. The basic RBPF uses the robot motion model as the proposed distribution, resulting in higher weights of particles with only higher observed posterior likelihood values. The difference in weight between particles becomes larger and the particles degenerate seriously. Therefore, the built environment map is not accurate. In order to solve the above problem, an observation model is added to the motion model as its mixed proposal distribution, as shown in equation (4):



Unlike the motion model, the observation model presents a relatively concentrated peak distribution. The proposed hybrid distribution cannot be directly sampled, and the Gaussian function is used to construct the proposed distribution. Firstly, the prediction is obtained according to the motion model, and then the prediction value is used as the initial value to perform a scan matching, and a region with high probability is obtained. K data is randomly selected in the region, and the variance and the mean are calculated by using the observation model and the motion model. New particles can be obtained from the simulated Gaussian function:



After obtaining the mixed proposal distribution of the Gaussian function simulation, the sampling of the robot pose information at the next moment can be performed. The formula for calculating the weight of particles at this time is:



After adding the sensor observation data, the variance of the importance particle weight is made smaller, but the integral is difficult, and when the observation model exhibits the kurtosis distribution, the sampling efficiency is reduced, which will cause the filter to diverge. Therefore, the annealing parameter α is introduced to regulate The ratio of the two models in the mixed distribution is shown in the following formula (7):



The weight calculation formula at this time is:



Through continuous experiment and comparison of the relationship between the observed data and the real distribution, in general, when the motion model plays a leading role, α is 0.6, and vice versa, when the sensor's observation model is closer to the real distribution, α is 0.02 to increase the proportion of the observed model.

## Improved Resampling

The traditional RBPF algorithm results in reduced particle diversity and even particle depletion due to the high number of resampling. In order to maintain the diversity of particles, optimize the obtained particle set, and introduce an adaptive genetic algorithm to cross-mutate some particles. The basic idea is to classify the particles according to the calculated particle weights, high-weight particles, medium-weight particles and low-weight particles. The appropriate high weight and low weight threshold are set by equation (9). Medium weight particles.



Introducing an adaptive genetic algorithm and selecting the weight as the fitness function of the particle, the cross mutation operation at time t is as follows.

Cross-operation: randomly select two particle individuals from the obtained high-weight and low-weight particle groups as parents to cross the particles according to the adaptive crossover rate shown in equation (10) to obtain new individuals.



Mutation operation: From the new particle set obtained according to the above crossover rate, randomly select one as the parent individual to operate the new particle according to the adaptive mutation rate p of equation (11).



Where is the maximum fitness value of the particles in the set;is the average fitness value of the particles in each generation group;is the larger fitness value of the two individuals in the crossover operation;is the particle performing the mutation operation Fitness value.

Improve the RBPF algorithm flow:

1. When t=0, select N particles, calculate the particle weight as ; set the values of  and .
2. Find the mixed proposal distribution and sample the particles according to equation (7).
3. The particle weights are calculated and updated according to equation (8).
4. Calculate the number of effective particles, determine whether resampling is performed according to equation (3), perform step (5) if resampling is performed, and perform step (6) otherwise.
5. According to the particle weight, the adaptive genetic algorithm (10) and (11) cross mutation operation are performed on high-weight and low-weight particles.

(6) The map m is calculated and updated based on the postureof the robot and the observation informationof the sensor.

## 4 EXPERIMENTAL RESULTS AND ANALYSIS

**4.1** **Simulation**

In order to verify the effectiveness of the proposed algorithm, the robot firstly estimates and compares its own poses on MATLAB, and sets the true pose state in its actual running trajectory, using the basic RBPF\_SLAM, literature [11] algorithm and improved algorithm in the particle. The number N is estimated to be 50 and 100 respectively to estimate the true pose of the robot. As shown in Figure 1 below:



(a)N=50



(b)N=100

Figure 1 Robot pose estimation

Table 1 Comparison of three algorithm data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Trajectory  RMSE | Road sign  rmse | Number of particles | Run time / s |
| RBPF | 2.492 | 2.674 | 80 | 5.627 |
| Literature [11] | 1.864 | 1.835 | 58 | 4.235 |
| Improve RBPF | 0.967 | 1.321 | 43 | 3.564 |

It can be seen from the data in Figure 1 and Table 1. In the case of the same number of particles, the root mean square error of the improved RBPF algorithm proposed in this paper is smaller than that of the basic RBPF and the literature [11], and is closer to the real state. As the number of particles increases, Although the improved algorithm runs longer, the root mean square error is smaller and more consistent with the real state. It can be seen from the data that the improved algorithm can obtain 50 particles compared with the RBPF using 100 particles. Better estimate the effect. Therefore, the improved algorithm can obtain more accurate estimates with fewer particles, provide more reliable and accurate data for subsequent map creation, and can create more accurate maps.

Secondly, the actual trajectory of the robot and the road sign are estimated, as shown in Figure 2 and Table 2.



Figure 2 Robot trajectory and road sign estimation

Table 2 Comparison of three algorithm data

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Number of particles | Root mean square error(rmse) | Run time / s |
| RBPF | 50  100 | 0.600  0.294 | 0.446  0.574 |
| Literature [11] | 50  100 | 0.463  0.210 | 0.395  0.617 |
| Improve RBPF | 50  100 | 0.262  0.101 | 0.510  0.713 |

From the data of Fig. 2 and Table 2, in the aspect of trajectory estimation, the improved algorithm is smaller than the basic RBPF and the error estimated by the literature [11], which is closer to the real trajectory; the improved algorithm in the road sign estimation is closer to the real landmark position, the estimated The error is smaller, and the basic RBPF and the road sign estimated by the literature [11] algorithm differ greatly from the actual landmark position, and the improved RBPF uses fewer and shorter time to estimate. Therefore, the improved algorithm can obtain more accurate results in robot trajectory estimation and road sign estimation, and can more effectively establish a higher precision raster map.

**4.2 Actual Verification**

ROS (Robot Operating System) is a robotic software platform that provides libraries and tools to help software developers create robotic applications. In the ROS system, the RBPF-SLAM algorithm is packaged as a Gmapping mapping package, and laser data can be used to create a two-dimensional environment raster map with high precision.

The experimental platform of this paper is the Kobuki sports chassis, which contains an odometer and carries a lidar. It performs simultaneous positioning and mapping on the linux (Ubuntu16.04) mobile platform with ROS.

Part of the laboratory was selected as the experimental environment for this experiment. As shown in Figure 3, the selected area was 6m × 3.2m. The robot used odometer data and laser observation data based on RBPF, literature [11] and improved RBPF-SLAM. The algorithm builds a raster map.



Figure 3 experimental environment



1. RBPF experiment results



1. Literature [11] experimental results



1. Improve RBPF experiment results

Figure 4 Rviz construction

Table 3 three algorithm construction data

|  |  |  |
| --- | --- | --- |
| Algorithm | Number of particles | Run time / s |
| RBPF | 50 | 413 |
| Literature [11] | 30 | 213 |
| Improve RBPF | 8 | 110 |

From the data of Figure 4 and Table 3, the accuracy of the map constructed by the traditional RBPF is not accurate enough for the grid maps with the same complexity environment. The improved algorithm of the literature [11] improves the accuracy of the constructed map with 30 particles, but The effect is not particularly obvious. The improved algorithm in this paper uses only 8 particles to construct a more accurate map in a shorter time, so the improved algorithm can build a more accurate map with fewer particles.

# 5 CONCLUSION

In this paper, an improved RBPF-SLAM algorithm is proposed for the traditional RBPF algorithm with low proposed distribution accuracy and reduced particle diversity. First, the motion model of the robot and the observation model are combined as a hybrid proposal distribution. At the same time, the annealing parameters are introduced to control the ratio of both. For the resampling process, the particles are classified according to the particle weights, and the high-weight and low-weight particles are introduced into the cross-variation operation in the adaptive genetic algorithm to maintain the particle diversity. The simulation is carried out on MATLAB to verify the effectiveness of the proposed algorithm. At the same time, the Kobuki robot is used for verification on ROS. Experiments show that the proposed algorithm can build more accurate maps with fewer particle numbers, and the running time is greatly reduced, which enables the robot to better carry out the next path planning.

**REFERENCES**

1. Chen Zhuo, Su Weihua, Consolation, Qin Xiaoli.Implementation of mobile robot SLAM and path planning in ROS framework [J]. Medical and Health Equipment, 2017, 38 (02): 109-113.
2. Endres F , Jürgen Hess, Engelhard N , et al. An evaluation of the RGB-D SLAM system[C]// 2012 IEEE International Conference on Robotics and Automation. IEEE, 2012.
3. Cheein, Toibero, Sciascio D , et al. Monte Carlo uncertainty maps-based for mobile robot autonomous SLAM navigation[C]// IEEE International Conference on Industrial Technology. IEEE, 2010.
4. Liu Chang. Research progress of synchronous positioning and map construction (SLAM) algorithm based on extended Kalman filter[J].Equipment Manufacturing Technology, 2017(12): 41-43

[5] Giorgio Grisetti,Gian Diego Tipaldi,Cyrill Stachniss,Wolfram Burgard,Daniele Nardi. Fast and accurate SLAM with Rao–Blackwellized particle filters[J]. Robotics and Autonomous Systems,2006,55(1).

[6] Wang Zhiyuan, Cheng Lan, Xie Gang. An Improved Particle Filter Algorithm and Its Application in Multipath Estimation[J]. Computer Engineering, 2017, 43(06): 289-295.

[7] Wang Fasheng, Lu Mingyu, Zhao Qingjie, Yuan Zejian. Particle Filtering Algorithm[J]. Chinese Journal of Computers, 2014, 37(08): 1679-1694.

[8] Yuvapoositanon P. Fast computation of look-ahead Rao-Blackwellised Particle Filter in SLAM[C]//2014 International Electrical Engineering Congress (iEECON). IEEE, 2014: 1-4

[9] Murphy K P. Dynamic Bayesian Networks : Representation, Inference and Learning[M]// Dynamic Bayesian Networks: Representation, Inference and Learning. Ph.D. thesis. 2002.

1. Zheng Bing, Chen Shili, Liu Rong. Research on Gmapping Based on Firefly Algorithm Optimization[J]. Computer Engineering, 2018, 44(09): 22-27.
2. Zhang Yi, Zheng Yifeng, Luo Yuan, Pang Dongxue.Rao-Blackwellized particle filter SLAM algorithm based on Gaussian distribution resampling[J].Control and Decision,2016,31(12):2299-2304.
3. Wang Tian-Yan, Cai Yun-fei, Tang Zhen-min. SLAM Method Based on Regional Particle Swarm Optimization and Partial Gaussian Resampling[J]. Computer Engineering, 2017, 43(11): 310-316.

1. [↑](#endnote-ref-1)
2. [↑](#endnote-ref-2)
3. [↑](#endnote-ref-3)
4. [↑](#endnote-ref-4)
5. [↑](#endnote-ref-5)
6. [↑](#endnote-ref-6)
7. [↑](#endnote-ref-7)
8. [↑](#endnote-ref-8)