TELKOMNIKA, Vol. 11, No. 12, December 2013, pp. 7617~7626
e-ISSN: 2087-278X

Improved Distributed Particle Filter for Simultaneous Localization and Mapping

Mei Wu*[1], Fujun Pei[2]
School of Electronic Information & Control Engineering, Beijing University of Technology
*Corresponding author, e-mail: wumei0617@yahoo.com.cn[1], pfj@bjut.edu.cn[2]

Abstract

The Simultaneous localization and mapping (SLAM) problem have become a focus of many researches on robot navigation. Generally the most widely used filter in SLAM problems are centralized filter. It is well known that SLAM based on conventional centralized filter must reconfigure the entire state vectors when the observation dimension changes, which cause an exponential growth in computation quantities and difficulties in isolate potential faults. In this paper, we proposed improved DPF distributed particle filter-SLAM in two aspects, in DPF-SLAM one centralized filter is divided into several distributed filters which reduce the computation quantities efficiently and avoid the necessary to reconfigure the entire state vectors in every step. First, we improved the important function of the local filters in distributed particle filter. By changed a set constant in the important function to an adaptive value, we improved the robustness of the system. Second, we propose an information fusion method that mixed the innovation method and the number of effective particles method, which combined the advantages of these two methods. The result of simulations shows that the algorithms we proposed improved the virtue of the DPF-SLAM system in isolate faults and enabled the system has a better tolerance and robustness.

Keywords: *distributed particle filter, simultaneous localization and mapping (SLAM), important function, information fusion*

1. Introduction

In the last decade, SLAM has become an important technology in the localization of robot, and has received many attentions. SLAM is the technology that generates a map and estimates the position simultaneously. It performs automatic navigation without any prior information of the environment and the position. There are many different types of SLAM, such as EKF-SLAM [1-3], PF-SLAM [4], and FastSLAM [5-7], which were applied in many environments. However, these SLAM systems have an algorithm structure based on centralized filter. Because the number of observation is changing and the state vectors need to be reconfigured whenever the observation environment changes. Thus the computation quantities of centralized filter are very large and it is hard to deal with fault.

To tackle these limitations of centralized filter, Dae Hee Won proposed a distribute particle approach for vision-SLAM [8, 9]. Different from centralized particle filter, the distribute particle filter divide particle filter to a main filter and several local filter (the number of the local filter is decided by the number of observation), which make it easier to design the filter and simplified the calculation. The simulation results showed that the distributed SLAM system has a similar estimation performance and requires only one-fifth of the computation time compare with centralized particle filter. However, particle impoverishment is inevitably because of the random particles prediction and resampling applied in generic particle filter. After a number of iterations, if the particles generated are too far from the likelihood distribution, the particle weights will approach zero with only a few particles carrying significant weights, making other particles not efficient to produce accurate estimation results.

In this paper, we propose the improved distributed particle filter to estimate the state vector of SLAM. The performance of the distribute particle filter is affected by several factors in its operation. First, the improved important function was used to calculate the probability of the particles in local filters. To calculate the probability of these particles, the covariance of these particles has to be estimated. The precision of the estimate results will be improved if the estimate covariance is close to the real covariance of the particles. By changed the parameter

Received June 29, 2013; Revised August 18, 2013; Accepted September 4, 2013

from a fixed constant to a variable, which determined by the covariance of the particles in each local filter. Second, the fusion algorithm was applied to calculate the estimate result in the master filter. In distributed filter, the estimation results of each local filter are transmitted to the master filter to calculate the estimate result by fusion algorithm. There are two methods to calculate the weights of the local filters: by the precision of the local filter or the number of effective particles. Both of these methods have its advantages and drawbacks. And we propose an improved method that mixed these two methods to calculate the weights of the local filters to get a better result. In order to test the accuracy and tolerance of the proposed algorithm, two simulations were carried out. Simulation results show that the improved distributed particle filter has better performance in accuracy and tolerance.

The remainder of this paper is organized as follows. Section 2 explains the DR/SLAM system based on laser sensor, and Section 3 introduced the DPF applied in SLAM system. Section 4 proposed an improved DPF-SLAM system and introduced structure of the improved DPF-SLAM system. Section 5 presents the results of simulation, and Section 6 states our conclusion.

2. The Model of SLAM

In SLAM system, although the absolutely position of the autonomous robot is not accessible, it is possible to use the information of indirect observation to estimate the position of the autonomous robot and maintain the error in a small range. SLAM, however, is rarely used alone but is combined mostly with dead reckoning (DR) or inertial navigation system (INS), because it has a low update rate for providing navigation information. The composing of SLAM system is dead reckoning (DR) and laser system, which is shown in Figure 1.

2.1. Motion Model

System model of SLAM is consisted by motion model and observation model, motion model is decided by Equation (1). Figure 1 shows motion from time k-1 to time k.

$$\mathbf{x}_r(k) = f(\mathbf{x}_r(k-1)) + \mathbf{w}(k) \qquad (1)$$

Where, $\mathbf{x}_r(k) = [x_r, y_r, \theta_r]^{\mathrm{T}}$, x_r , y_r , ϕ_r is the location of the sensor in robot coordinate. $\mathbf{w}(k) = N(0, Q)$ is state noise, which is normal distribution with the covariance of *Q*. $f(\mathbf{x}_r(k))$ is the transform function, which transform from $\mathbf{x}_r(k-1)$ to $\mathbf{x}_r(k)$.

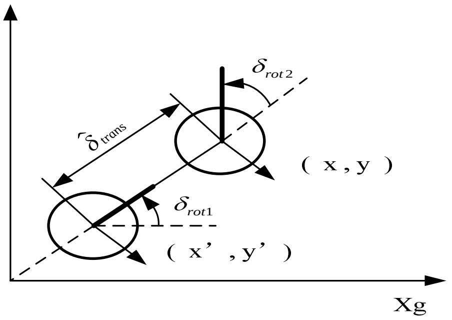

Figure 1. The Motion from Time k-1 to Time k

2.2. Observation Model

The ranging and bearing of the feature points is utilized to configure a SLAM system, and these values are determined by the azimuth of the robot and the relative position between the robot and feature points. The ranging and bearing noises can be assumed to be in a simple relationship of summation so that the observation equation is converted into following nonlinear equation:

$$\mathbf{z}(k) = (r,\beta) = h\left(\mathbf{x}_r(k), \mathbf{m}_i(k)\right) + \mathbf{v}(k)$$
$$= \begin{bmatrix} \sqrt{(x_i - x_L)^2 + (y_i - y_L)^2} \\ a\tan\left(\dfrac{(y_i - y_L)}{(x_i - x_L)}\right) - \phi_L + \dfrac{\pi}{2} \end{bmatrix} + \mathbf{v}(k) \tag{2}$$

Where, (r,β) is the observed value of the ranging and bearing information of different feature points $\mathbf{m}_i(i=1...n)$, where r is the distance between laser sensor and feature points, and β is the bearing measured by the laser sensor in robot coordinate. $\mathbf{m}_i = (x_i, y_i)$ stand for the robot related state and the position of the i-th feature point. (x_i, y_i) is the coordinate of the i-th feature point, $i = 1,2,...,n$ is the number of the feature points are measured. $\mathbf{v}$ □ $N(0,R)$ is the measurement noise, which is normal distribution with the covariance of *R*.

3. Distributed Particle Filter for SLAM

In centralized SLAM system, the dimension of state vector and observation vector changed in every step. Therefore, it is hard for centralized filter to handle this change. Because changing of the feature points alters the state dimension of the filter. However, it is easier to handle in distributed particle filter. Figure 2 shows the structure of distributed SLAM system which Dae Hee Won proposed [10]. “Feature Point Block” indicates local filters being assigned to each of the measured feature points. And “Landmark Block” is a local filter block that utilizes the precise position of landmark, if measured, to estimate the state. The results of estimation by individual local filters are transmitted to the master filter, which estimates navigation solution using particles provided by each of the local filters. The master filter estimates the state related to the robot based on the local filters in “Landmark Block”, and the estimation results of the “Feature Point Block” was used to update the map of environment.

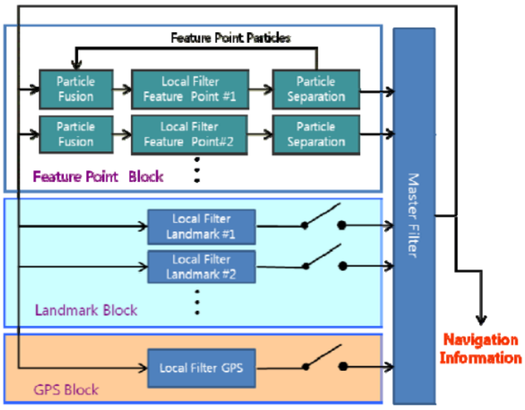

Figure 2. Configuration of the Distributed Particle Filter

The results of estimation by individual local filters are transmitted to the master filter, which estimates navigation solution using particles provided by local filters. The master filter estimates the state related to the robot only, so it is not necessary to pass all state vectors in the local filters to the master filter. Therefore, feature point-related state vectors are separated from robot-related ones in the “Particle Separation” phase, and robot-related state vectors alone are delivered to the master filter. When estimating state vectors, the master filter applies different weights to each local filter. Here the weight is set based on the innovation of each local filter and is calculated as follows:

$$\beta_i = \frac{1/innovation_i}{\sum_{j=1}^{n} 1/innovation_j} \tag{3}$$

Where β_i are the weights of the local filters.

4. The Improved Distributed Particle Filter SLAM

The distribute SLAM reduce the computation quantities, and has the similar navigation performance compared with centralized SLAM. However, there are some limitations inherent in particle filter, such as particle impoverishment and losing sampling diversity, which will cause divergence of the filter.

In this passage, we proposed the improved distribute particle filter which improved the performance of the distributed particle filter in two different aspects. First, we improved the important function of the local particle filters. Second, we put forward an information fusion method, used in master filter, by mixing the innovation and the Number of Effective Particles methods.

4.1. Improved the Important Function of Particle Filter

Roughly speaking, particle filtering are numerical algorithms to approximate the conditional distribution by an empirical distribution, constituted by a cloud of particles at each time instant. Thus one important feature of the particle filter is that a series of particles are generated randomly $\{x_{0:t}^{(i)}; i = 1,\ldots,N\}$ from posterior distribution is used to map integrals to discrete sums. In another word, the posterior can be approximated by the following empirical estimate:

$$\hat{p}(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta_{x_{0:t}^{(i)}}(d\mathbf{x}_{0:t}) \tag{4}$$

Where $\delta_{x_{0:t}^{(i)}}(dx_{0:t})$ denotes the Dirac delta function.

Base on Equation (4), Equation (5) can be estimate by Equation (6).

$$E\left(\mathbf{g}_t(\mathbf{x}_{0:t})\right) = \int \mathbf{g}_t(\mathbf{x}_{0:t}) p(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}) d\mathbf{x}_{0:t} \tag{5}$$

$$\overline{E\left(\mathbf{g}_t(\mathbf{x}_{0:t})\right)} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{g}_t(\mathbf{x}_{0:t}^{(i)}) \tag{6}$$

Denotes the particles $\mathbf{x}_{0:t}^{(i)}$ are assumed to be independent and identically distributed.

According to the law of large numbers, we have $\overline{E\left(\mathbf{g}_t(\mathbf{x}_{0:t})\right)} \underset{N\to\infty}{\to} E\left(\mathbf{g}_t(\mathbf{x}_{0:t})\right)$. And, if the posterior variance of $\mathbf{g}_t(\mathbf{x}_{0:t})$ is bounded which means $\mathrm{var}_{p(\mathbf{x}_{0:t}\mid\mathbf{y}_{1:t})}\left(\mathbf{g}_t(\mathbf{x}_{0:t})\right) < \infty$.

$$\sqrt{N}\left(E\left(\mathbf{g}_t(\mathbf{x}_{0:t})\right) - E\left(\mathbf{g}_t(\mathbf{x}_{0:t})\right)\right) \underset{N\to\infty}{\to} N\left(0, \mathrm{var}_{p(\mathbf{x}_{0:t}\mid\mathbf{y}_{1:t})}\left(\mathbf{g}_t(\mathbf{x}_{0:t})\right)\right) \tag{7}$$

Thus, if assumed that the distribution of the particles obeys gauss distribution, whose mean and the variance are observation, the probability of the particles can be estimate by Equation (8). This is the most popular used method to generate the probabilities of the particle filters.

$$w_k^{i*} = \frac{1}{(2\pi)^{n_v/2}\sqrt{\det \mathbf{R}_k}} e^{-\frac{1}{2}(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_k^{i*}))^T \mathbf{R}_k^{-1} (\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_k^{i*}))} \tag{8}$$

Where, $(\mathbf{z} - \mathbf{h}(\mathbf{x}))$ is the difference between recursive estimation and observation. $\mathbf{R}_k$ is the estimated value of the real covariance of the particles, which is an important factor for the particle filter. And the estimation accuracy of particle filter will decline when the estimation covariance diverge from the real particles covariance of the particles.

Generally the value of $\mathbf{R}_k$ is set roughly a fixed constant to generate the weights of the particles. But this method has its limitation, for the particle covariance may vary in iterative process and will different in individual local filters. Setting $\mathbf{R}_k$ as a fixed constant will cause $\mathbf{R}_k$ to diverge from the real covariance of the particles.

In this paper, we proposed the adaptive method to adjust $\mathbf{R}_k$ by the covariance of the particles in iterative process. Firstly, a fixed constant $\mathbf{R}_k$ is used to generate the probability of particles, then the covariance of these particles is used to generate $\mathbf{R}'_k$ by Equation (9). Secondly, $\mathbf{R}'_k$ is used to generate probability of particles by Equation (10).

$$\mathbf{R}'_k = \sqrt{\sum_{i=1}^{N} w_k^{i*}(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_k^{i*}))^T(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_k^{i*}))} \tag{9}$$

$$w_k^{i*}{}' = \frac{1}{(2\pi)^{n_v/2}\sqrt{\det \mathbf{R}'_k}} e^{-\frac{1}{2}(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_k^{i*}))^T(\mathbf{R}'_k)^{-1}(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_k^{i*}))} \tag{10}$$

In the improved method, the value of $\mathbf{R}_k$ is substitute by $\mathbf{R}'_k$, which is an adaptive value which calculated in each iteration to improve the robustness of the system. The process of recalculation allots a better weight to the particles by a recalculation step. Since $\mathbf{R}_k{}'$ is calculated in every step, the improved system has a better robustness. As a result, the improved distribute particle filter system are better than the original system by change the important function, and a better important weights make the estimate more accuracy.

4.2. Information Fusion Method

For one of the most important advantages of DPF is that by set different weights to local filters the master filter can reduce the probability of the particle filter with bad performance. From the principle of information sharing scheme, the weights set to individual local filter can be described by Equation (11).

$$\beta_1 + \beta_2 + \cdots + \beta_n = 1 \tag{11}$$

Where β_n represent the weights allots to each local filter.

In the previous study, these weights were fixed value. But fixed value can’t simultaneously reflect the performance of the local filters. However, there are several adaptive methods to generate the weights of the local filters in the main filter. In [10], the weight is set based on the innovation of each local filter and is calculated as Equation (3). In this method, the innovations were used to evaluate the performance of the local filter, a larger innovation represent a worse quality of the observation, and a lesser weight in master filter. This algorithm can reduce the affection of the observation noise effectively, but when the estimate result of the local filter is not accurate enough, the effective of this method will diminish. Therefore, the weight allocation for individual local filters must consider the estimation performance of individual local filters.

The Neff (number of effective particles) represents the efficient of the local filter because the value of Neff is calculated by the weight of the particles in each filter, which is indicated in Equation (12).

$$Neff_j = \frac{1}{\sum_{j=1}^{N}(\omega_k^j)^2} \tag{12}$$

Where, ω_k^j is the weight of the particles in each filter, j is the index of local filters and N is the number of the local filters.

Then the weight, which represents the estimation performance of individual local filter, can be set based on the Neff of each local filter and is calculated as Equation (13).

$$\beta_j = \frac{Neff_j}{\sum_{j=1}^{N}(Neff_j)} \qquad (13)$$

In sum, these two methods have respective advantage and disadvantage, and represents different performance of the local filters. The innovation method can evaluate the quality of the observation efficiently. But if the estimation of the state is not accurate enough, the performance of this method is limited. On the other hand, the performance of the Neff method is not as good as the innovation method, when the estimation of the state is accurate enough. Because the number of effective particles method evaluates the efficient of the local filter, the performance of this method will less likely affect by the accuracy of the estimation of states.

In this paper, we proposed the fusion method mixed the innovation method and the Neff method. The fusion algorithm set the weights of the local filters by both the innovation method and the Neff method. The structure of the improved fusion method is shows in Figure 3. In this proposed method, the weight of the local filters reflected both the observation performance and the estimation performance of the local filters. The weights of the local filters are calculated by Equation (14).

$$\beta_N \cdot \alpha + \beta_I \cdot \chi = \beta_M \qquad (14)$$

Where, β_N is the weights generated by Neff algorithm. β_I is the weights generated by innovations algorithm. β_M is the mixed weights. α and χ are the proportion parameters, which stands for the proportion of β_N and β_I in the mixed weights, and is restricted by Equation (15).

$$\alpha + \chi = 1 \qquad (15)$$

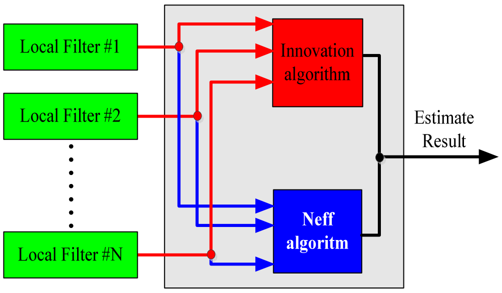

Figure 3. The Structure of Improved Fusion Method

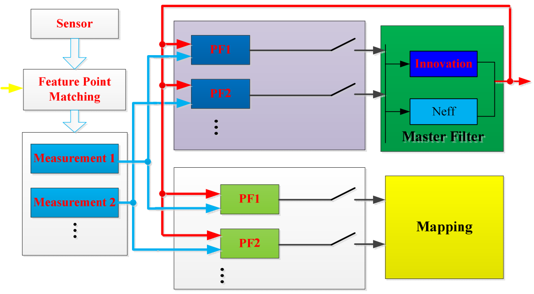

Figure 4. The Structure of Improved DPF-SLAM System

In conclusion, we proposed the improved distributed particle filter is given in Algorithm 1 and it is briefly described below.

The structure of Improved DPF, configured in this paper to realize a SLAM system, is described in Figure 4. In Figure 4, there are two subsystem blocks in the IDPF-SLAM system: state subsystem block and mapping subsystem block. State subsystem estimates the state of the robot using the proposed Improved DPF. Mapping subsystem estimates the position of the landmarks and updates the map of the environment using DPF.

After the measurement divided, these measurements are transformed to the local filters in two subsystems separately. Each local filter in these two subsystems has their own estimation results. Different from the mapping subsystem, each local filter estimate one landmark, all of the local filters in state subsystem estimate the state of the robot using improved DPF. The estimation results of local filters were transmitted into the master filter to estimate the state of the robot by the proposed fusion method as mentioned above.

Then the estimation results of these two subsystems are utilized to update the whole system. The result of the master filter is used to update the state of the robot while the result of the mapping is used to update the map generated by the SLAM system.

5. Simulation

To verify the accuracy and tolerance of the improved algorithm, two experiments were performed to verify its performance. It was assumed that the process of extracting and tracking feature points was carried out independently by an external module. For the performance comparison, results from the distributed particle filter and improved distributed particle filter are demonstrated.

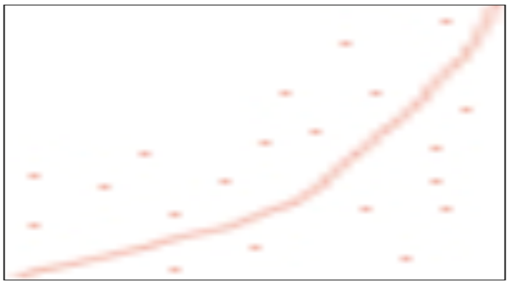

Figure 5. Track of the Robot and the Feature Points

Figure 5 shows the map used in this paper. It is 50*50 meter large, there are 20 feather points. The time of simulation is 15 second, the rate of velocity and attitude update is 100 Hz. The points in Figure 5 are the feather points. The line is the track of the robot. In this simulation, position estimation at a rate of 10Hz. It takes 20 seconds for the robot move to the top right corner from the lower left corner of the map. Due to the fact that there is not enough feature points in the right corner of the map, we only use part of the track (15 second in 20 second). In these experiments, the noise of the observation is 0.05m in range and 0.01radian in bearing. The initial error of position is 0.2m in x and y direction respectively. The initial error in bearing is 0.005radian.

5.1. Experiment 1

In experiment 1 two simulations were taken to test the performances of the distributed particle filter and the distributed particle filter with improved important function.

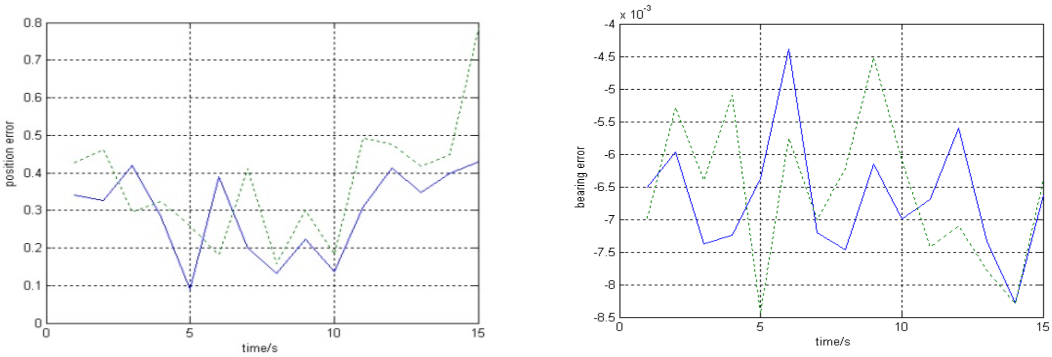

Figure 6. Position and Heading Error in Experiment 1.1

In experiment 1.1 the estimated position error and heading error of distribute particle filter and improved distribute particle filter is shown in Figure 6. The real line is the error of improved method, and the dotted line is the distributed particle filter SLAM.

From Figure 6, it is easy to find that the estimate results of the improved algorithm are more accuracy than the former algorithm which has the larger maximum error and the minimum error in position than the improved method. This simulation results proved that with the improved important function, we can get a better probability of the particles in local filters. An adaptive estimation method for R_k can get a better weight of the particles in local filters which improved the performances of the distributed particle filter.

In experiment 1.2, to exam the robustness of the improved SLAM system, some disturb noise were added in the process of SLAM at step 5-15. Figure 7 shows the estimation errors in position and heading of robot. There is a dramatic increase in the estimated error of the former SLAM system comparing with the improved SLAM system. It shows that the robustness of the improved SLAM system is better than the former SLAM system. This simulation results, in Figure 7, proved that an adaptive important function can set a better weights of the particles in the local filters when some noisy were added to the SLAM system, and made the improved SLAM system have a better robustness.

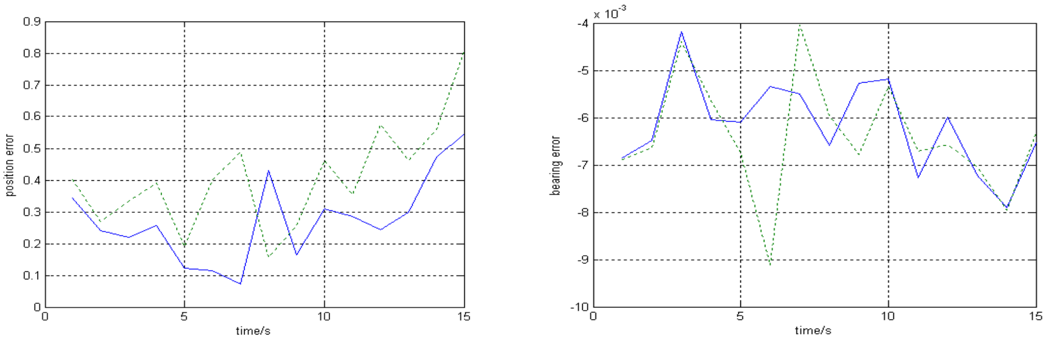

Figure 7. Position and Bearing Error in Experiment 1.2

5.2. Experiment 2

To test the tolerance of the improved fusion method, three simulations were designed and the simulation conditions in experiment 2 were same as experiment 1.

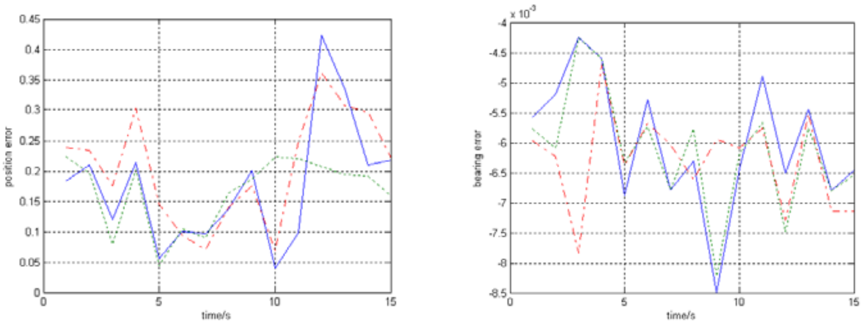

Figure 8. Position and Heading Error in Experiment 2.1

Figure 8 describes the estimated position and heading error in experiment 2.1. The real line demonstrates the error of Neff method, the dotted line showed the error of the innovation method, and the dash-dotted line represents the error of the proposed fusion method.

According to Figure 8, the proposed fusion method sharing a same accuracy with the innovation method, the max error of the innovation method is less than 0.25m in position, and the max error of the number of effective particles method in position is larger than 0.4m. In addition, the Neff method gets the largest error in heading. The results of experiment 2.1 demonstrated that if the state of the system is accuracy enough, the innovation method can reduce the affective of the observation error to the system.

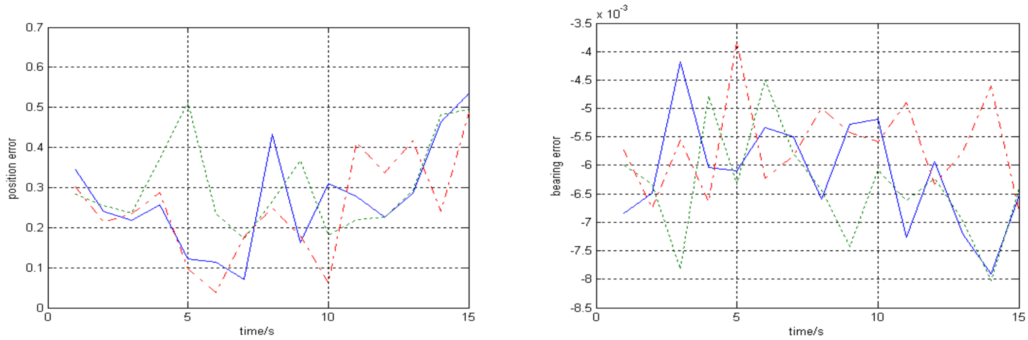

Figure 9. Position and Heading Error in Experiment 2.2

In experiment 2.2 some disturb noise was added in the process of SLAM between step 5 to step 15, which is a white noisy 0.1 in range added in both *x* and *y* direction. The estimation errors of three methods were shown in Figure 9. According to the Figure 9, the proposed fusion method has the best estimation accuracy in position and heading, and the Neff method is better than the innovation method.

The result of experiment 2.2 proved that the performance of innovation method is bad when the state of the system is not accuracy enough. If the estimation results of state are not accuracy enough, in SLAM system, the effective of using innovation to evaluate the accuracy of the observation is limited. In this condition, the performance of the Neff method is better than the innovation method, because the Neff method evaluate the weights of the local filters by the effective of the particles in the local filters which is relate to the accuracy of state estimation. The proposed fusion method has better performance than the other two methods because the fusion method considers both the innovation and the Neff of the particles in local filters.

In experiment 2.3 a disturb noise, which is a white noisy 0.5m in range and 0.01radian in bearing, was added into observation data in step 10. The results of the experiment 2.3 were shown in Figure 10. The disturb noise added was a white noisy 0.5m in range and 0.01radian in bearing. In this experiment, the number of effective particles method has the worst estimated performance, and the proposed fusion method has better performance than the other two methods.

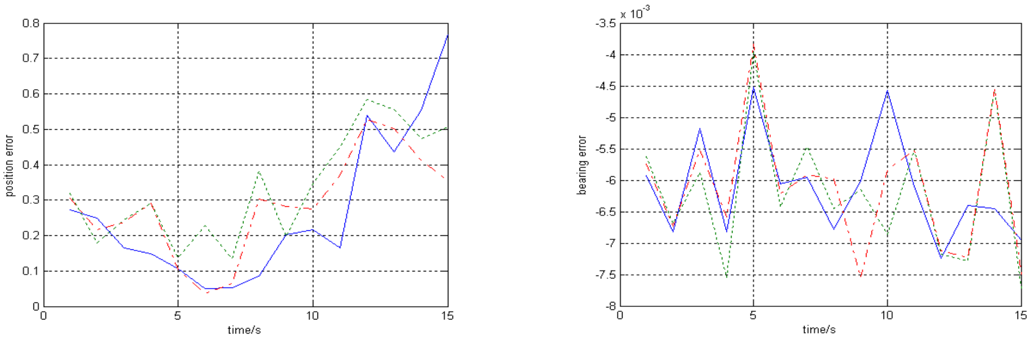

Figure 10. Position and Heading Error in Experiment 2.3

From this experiment, we have the conclusion that when noise in observation increased, the innovation method has poor performance, and the proposed fusion method has the best performance in the three methods, which proved that the improved method has a better tolerance.

According to the experiment results, the innovations method can get better estimated result, when there is no disturb noise. When some disturb noise was added in estimation process, the Neff method has better estimated performance than the innovations method. In contrary, when there is disturb noise in the sensor measured data, the experiment results described that the performance of the Neff method is worse than the innovation method. However, when there are different disturb noise in the system, neither the innovation method nor the Neff method is the best choice, and the fusion method proposed in this paper has the best estimated performance.

6. Conclusion

In this paper, we proposed an improved distributed particle filter SLAM to generate a map and localize the robot. Firstly, the variance estimation of particle distribution in local filters is a vital parameter in generating the weights of the particles. An optimized variance estimation of particle distribution made the filter has a better performance. Instead of a fixed parameter in variance estimation, we use the particles variance as the variance estimation adaptively. Secondly, the appropriate weights of local filters in master filter is also important factor in distribute particle filter. The information fusion method proposed by this paper mixes the innovation method and the Neff method. To test the performance of the improved distributed particle filter algorithm for SLAM, two simulation experiments were designed. The simulation results show that the improved algorithm has the advantage in robustness and accuracy and has the better tolerance than the former algorithm.

References

[1] H Durrant, T Bailey. Simultaneous localization and mapping: part I*. Robotics & Automation Magazine. IEEE Transactions on*. 2006; 13(2): 99-110.

[2] T Bailey, H Durrant. Simultaneous localization and mapping (SLAM): part II. *Robotics & Automation Magazine. IEEE Transactions on.* 2006; 13(3):108-117.

[3] H Shoudong, D Gamini. Convergence and Consistency Analysis for Extended Kalman Filter Based SLAM. *Robotics, IEEE Transactions on*. 2007; 23(5); 1036-1049.

[4] D Törnqvist, TB Schön R Karlsson, F Gustafsson. Particle Filter SLAM with High Dimensional Vehicle Model. *Journal Intelligent Robot System.* 2009; 55: 249–266.

[5] M Montemerlo, S Thrun, D Koller, et al. *Fast SLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges.* The 18th International Joint Conference on Artificial Intelligence. 2003: 1151-1156.

[6] Y Ma, H Ju, P Cui. *Research on Localization and Mapping for Lunar Rover Based on RBPF-SLAM.* Intelligent Human Machine Systems and Cybernetics 2009, International Conference on. 2009; 2: 306-311.

[7] S Christof, G Horst-Michael. A sensor-independent approach to RBPF SLAM - Map Match SLAM applied to visual mapping. *IEEE/RSJ International Conference on Intelligent Robots and Systems,* 2008: 2078-2083.

[8] D Won, S Chun, S Sung, et al. *Improving Mobile Robot Navigation Performance using Vision based SLAM and Distributed Filters.* International Conference on Control, Automation and Systems. 2008; 186-191.

[9] D Won, S Chun, S Sung, et al. INS/vSLAM System Using Distributed Particle Filter. *International Journal of Control, Automation, and Systems*. 2010; 1232-1240**.**