Research Article

Distributed SLAM Using Improved Particle Filter for Mobile Robot Localization

Fujun Pei, Mei Wu, and Simin Zhang

School of Electronic Information & Control Engineering, Beijing University of Technology, Beijing 100124, China

Correspondence should be addressed to Fujun Pei; pfj@bjut.edu.cn

Received 30 August 2013; Accepted 4 December 2013; Published 27 April 2014

Academic Editors: J. Shu and F. Yu

Copyright © 2014 Fujun Pei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The distributed SLAM system has a similar estimation performance and requires only one-fifth of the computation time compared with centralized particle filter. However, particle impoverishment is inevitably because of the random particles prediction and resampling applied in generic particle filter, especially in SLAM problem that involves a large number of dimensions. In this paper, particle filter use in distributed SLAM was improved in two aspects. First, we improved the important function of the local filters in particle filter. The adaptive values were used to replace a set of constants in the computational process of importance function, which improved the robustness of the particle filter. Second, an information fusion method was proposed by mixing the innovation method and the number of effective particles method, which combined the advantages of these two methods. And this paper extends the previously known convergence results for particle filter to prove that improved particle filter converges to the optimal filter in mean square as the number of particles goes to infinity. The experiment results show that the proposed algorithm improved the virtue of the DPF-SLAM system in isolate faults and enabled the system to have a better tolerance and robustness.

1. Introduction

Autonomous localization is a process of determining the robot's position without the use of any a priori information external to the robot except for what the robot senses about its unknown environment. This is also known as Simultaneous Localization and Mapping (SLAM), as introduced in [1, 2], where the robot has a capability for online map building, while simultaneously using the generated map to estimate and correct errors in the navigation solution obtained. It is well known that SLAM using an optimal filter, such as EKF-SLAM [3, 4], PF-SLAM [5], and FastSLAM [6–8], has been applied in many systems and solved a lot of pragmatic problems.

However, SLAM based on single centralized filter must reconfigure the entire system equation when the feature points change, which cause an exponential growth in computation quantities and difficulties in isolate potential faults. To tackle these limitations of centralized filter, Won et al. proposed a distributed particle filter for implementing vision-SLAM [9, 10]. Different from centralized particle filter, the distribute SLAM divides particle filter to some “feature point

blocks” and some “landmark blocks,” which make it easier to design the filter and simplify 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 compared with centralized particle filter.

However, particle impoverishment is inevitably induced because of the random particles prediction and resampling applied in generic particle filter [11], especially in SLAM problem that involves a large number of dimensions. If the generated particles 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 general, there are two types of improved methods for these problems.

- (1) There are many improved resampling methods that have been discussed, such as binary search [11], systematic resampling [12], and residual resampling [13]. However, these methods are not ideal because the copied samples are no longer statistically independent after resampling sample. Therefore the convergence

result obtained from previous step will be lost, which is called losing sampling diversity. These problems caused the same problem in distributed particle filter used for SLAM problem.

- (2) Another improved method was optimizing the proposal distribution using optimal estimation algorithms such as extended Kalman [14], or unscented Kalman filter [15, 16]. Since these methods, especially for unscented PF, estimate the proposal distribution in order to minimize the variance of the importance weights, which ensure that the weights are upper bounded. Thus it is not a surprise that unscented particle filter yields better proposal distribution than other methods.

In this paper, the improved particle filter was proposed to estimate the state vector of distributed SLAM. The performance of the particle filter is affected by several factors in its operation. First, the important function used to calculate the probability of the particles in local filters. To calculate the probability of these particles, the variance of these particles has to be estimated. The precision of the estimation results will be improved if the estimation variance is close to the real variance of the particles. By changing the parameter from a set of constants to a variable, determined by the variance of the particles in each local filter. Second, the fusion algorithm is applied to calculate the estimation results in the master filter. In distributed filter, the estimation results of each local filter are transmitted to the master filter to calculating the estimation results by fusion algorithm. There are two methods for calculate the weights of the local filters: the precision of the local filter and the number of effective particles. Both of these methods have their advantages and drawbacks. Then an improved method which mixed these two methods was proposed to calculate the weights of the local filters. In order to test the accuracy and tolerance of the proposed algorithm, two experiments were carried out. Experiment results show that the improved particle filter has better performance in accuracy and tolerance.

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

2. Distributed SLAM System

2.1. The Centralized SLAM System. In SLAM system, although the absolute 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.

FIGURE 1: Vehicle system.

A SLAM system model combines the robot's kinematics model with the feature point model. The state vectors of the kinematics model represent position and azimuth; those of the feature point model refer to the feature point position, which is assumed to be a fixed point and its error is therefore modeled into a random constant. Equation (1) describes a system model used in SLAM system as follows:

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{W}, \quad (1)$$

where

$$\mathbf{x} = [\mathbf{x}_r \quad \mathbf{m}_1 \quad \mathbf{m}_2 \quad \cdots \quad \mathbf{m}_n]^T, \quad (2)$$

$$\dot{\mathbf{x}}_r = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\phi}_r \end{bmatrix} = f(\mathbf{x}) + \gamma = \begin{bmatrix} v_c \cdot \cos(\phi) \\ v_c \cdot \sin(\phi) \\ \frac{v_c}{L} \cdot \tan(\alpha) \end{bmatrix} + \gamma.$$

$\dot{x}_r, \dot{y}_r, \dot{\phi}_r$ is the change of the state. L is the distance between wheels. γ is the of the noise in DR system. $\mathbf{m}_{1:n}$ is feature point position.

The ranging and bearing of the feature points are 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:

$$\mathbf{z}(k) = h(\mathbf{x}_r(k), \mathbf{m}_i(k)) + \mathbf{v}(k), \quad (3)$$

where $\mathbf{z}$, $\mathbf{x}_r$, and $\mathbf{m}_i$ stand for the observed value, the vehicle related state, and the position of the i th feature point, respectively. The measurement noise, described as $\mathbf{v}N(0, R)$, can be assumed to be normal distribution with the covariance of R .

The observed value $\mathbf{z}(k) = (r, \beta)$ received by the laser sensor is the ranging and bearing information of different

feature points m_i ($i = 1 \dots n$), where r is the distance between sensor and feature points and β is the bearing measured by the laser sensor in robot coordinate. When multiple landmarks were observed, the observation vector will be represented by the following:

$$\mathbf{z} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_n \end{bmatrix} = \begin{bmatrix} h(\mathbf{x}_r(k), \mathbf{m}_1) \\ h(\mathbf{x}_r(k), \mathbf{m}_2) \\ \vdots \\ h(\mathbf{x}_r(k), \mathbf{m}_n) \end{bmatrix} + \mathbf{v}(k), \quad (4)$$

where

$$h(\mathbf{x}_r(k), \mathbf{m}_i) = \begin{bmatrix} \sqrt{(x_i - x_L)^2 + (y_i - y_L)^2} \\ a \tan\left(\frac{(y_i - y_L)}{(x_i - x_L)}\right) - \phi_L + \frac{\pi}{2} \end{bmatrix} + \mathbf{v}(k). \quad (5)$$

(x_i, y_i) is the coordinate of the i th feature point, $i = 1, 2, \dots, n$ is the number of the feature points measured, and x_L, y_L, ϕ_L is the location of the sensor in robot coordinate.

2.2. The Structure of Distributed SLAM System. Based on the system model in (1), it is evident that is a nonlinear system. The most widely applied solutions of SLAM system are EKF, UKF, and PF. However, the SLAM system based on EKF, UKF, or PF uses only one centralized filter, which uses a single filter to estimate all states of SLAM system. With the running of the robot, the number of feature points which can be observed will change. Therefore, the structure of centralized SLAM system must be reconfigured whenever the observation environment of the landmarks changes. On the other hand, these centralized filters will cause quantum of calculation to extend with the state vector increasing. Furthermore, centralized filter is hard to detect and isolate possible faults.

To tackle the drawbacks of centralized SLAM system, Won et al. proposed SLAM system based on distributed particle filter [10], which replaced the single centralized particle filter with local particle filters and a master filter. This technique avoids the necessity to change the structure of filter with the dimension of observation. And each local particle filter can get its own estimation result separately. Thus, even if some of the observations of the local filter contain large errors, it is easy to eliminate its effect.

Won et al. proposed the structure of distributed SLAM system including feature point position subsystem and robot's position and attitude subsystem. The distributed SLAM system model of the vehicle's position estimated system is described in the following:

$$\begin{aligned} \dot{\mathbf{x}}_v &= f(\mathbf{x}_v) + \gamma \\ z_1 &= h(\mathbf{x}_v(k), \mathbf{m}_1) + v_1(k) \\ \dot{\mathbf{x}}_v &= f(\mathbf{x}_v) + \gamma \end{aligned}$$

$$\begin{aligned} z_2 &= h(\mathbf{x}_v(k), \mathbf{m}_2) + v_2(k) \\ &\vdots \\ \dot{\mathbf{x}}_v &= f(\mathbf{x}_v) + \gamma \\ z_n &= h(\mathbf{x}_v(k), \mathbf{m}_n) + v_n(k). \end{aligned} \quad (6)$$

The estimation results were feedback to correct the position of vehicle.

The system model of the distributed landmarks position estimation system is described in the following:

$$\begin{aligned} \dot{\mathbf{m}}_1 &= 0 \\ z_1 &= h(\mathbf{x}_v(k), \mathbf{m}_1) + v_1(k) \\ \dot{\mathbf{m}}_2 &= 0 \\ z_2 &= h(\mathbf{x}_v(k), \mathbf{m}_2) + v_2(k) \\ &\vdots \\ \dot{\mathbf{m}}_2 &= 0 \\ z_2 &= h(\mathbf{x}_v(k), \mathbf{m}_2) + v_2(k). \end{aligned} \quad (7)$$

The estimation results of the feature point position subsystem are used to update the environment map. And the estimation results of the robot's position and attitude subsystem are transmitted into the master filter, which fused estimation results from each of the local filters to estimate robot's position and attitude. When fusing state vectors from the local filters, the master filter applies different weights to each local filter. Here the weights are set based on the innovation of each local filter.

3. The Improved Distributed Particle Filter

The distributed SLAM reduces the computation quantities and has similar estimation 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 section, the improved distributed particle filter was proposed to improve the performance of the distributed particle filter in two different aspects. First, the improved important function method in the local particle filters was proposed by adaptive adjusting of the estimation value of covariance. Second, an information fusion method was proposed by mixing the innovation and the number of effective particles methods.

The structure of improved DPF, proposed in this paper to realize a SLAM system, is described in Figure 2. In Figure 2, 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 Improved DPF.

FIGURE 2: The structure of improved DPF SLAM system.

3.1. The Improved Important Function. Roughly speaking, particle filters 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 $\{x_{0:t}^{(i)}; i = 1, \dots, N\}$ from posterior distribution which is used to map integrals to discrete sums. In other words, the posterior can be approximated by the following empirical estimation:

$$\hat{p}(x_{0:t} | y_{1:t}) = \frac{1}{N} \sum_{i=1}^N \delta_{x_{0:t}^{(i)}}(dx_{0:t}), \quad (8)$$

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

Base on (8), (9) can be estimate by (10)

$$E(g_t(x_{0:t})) = \int g_t(x_{0:t}) p(x_{0:t} | y_{1:t}) dx_{0:t} \quad (9)$$

$$\overline{E(g_t(x_{0:t}))} = \frac{1}{N} \sum_{i=1}^N g_t(x_{0:t}^{(i)}). \quad (10)$$

The particles $x_{0:t}^{(i)}$ are assumed to be independent and identically distributed.

According to the law of large numbers, we have $\overline{E(g_t(x_{0:t}))} \xrightarrow{N \rightarrow \infty} E(g_t(x_{0:t}))$. And if the posterior variance of $g_t(x_{0:t})$ is bounded which means. Consider

$$\begin{aligned} & \sqrt{N} (E(g_t(x_{0:t})) - \overline{E(g_t(x_{0:t}))}) \\ & \xrightarrow{N \rightarrow \infty} N(0, \text{var}_{p(x_{0:t}|y_{1:t})}(g_t(x_{0:t}))). \end{aligned} \quad (11)$$

Thus, if we assumed that the distribution of the particles obeys gauss distribution, whose mean and variance are observation, the probability of the particles can be estimated by (12). This is the most popular used method to generate the probabilities of the particle filters:

$$w_k^{i*} = \frac{1}{(2\pi)^{n_v/2} \sqrt{\det R_k}} e^{-(1/2)(z_k - h_k(x_k^{i*}))^T R_k^{-1} (z_k - h_k(x_k^{i*}))}, \quad (12)$$

where $(z - h(x))$ is the difference between recursive estimation and observation. R_k is the estimation value for 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 diverges from the real particles covariance of the particles.

Generally, R_k is set roughly as 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 be different in individual local filters. Setting R_k as a fixed constant will cause R_k to diverge from the real covariance of the particles.

In this paper, the adaptive calculation method was proposed to adjust R_k by the covariance of the particles in iterative process. Firstly, a fixed constant R_k is used to generate the basic probability of particles. Secondly, the covariance of these particles is used to generate the new value R'_k by (13). The new value R'_k is used to generate the final weights by (12).

$$R'_k = \sqrt{\sum_{i=1}^N w_k^{i*} (z_k - h_k(x_k^{i*}))^T (z_k - h_k(x_k^{i*}))} \quad (13)$$

$$w_k^{i*'} = \frac{1}{(2\pi)^{n_v/2} \sqrt{\det R'_k}} e^{-(1/2)(z_k - h_k(x_k^{i*}))^T (R'_k)^{-1} (z_k - h_k(x_k^{i*}))}. \quad (14)$$

In the improved method, R_k is substituted by the estimation covariance R'_k , which is calculated in each iteration to improve the robustness of the particle filter. The adaptive adjustment process of estimation covariance receives a better weight for the particles by recalculation step. Since R'_k is calculated in every iteration step, the improved particle filter has a better robustness. Therefore, the improved distributed particle filter systems are better than the original system in accuracy and robustness.

3.2. Information Fusion Method. One of the most important advantages of DPF is that by setting different weights to local filters in the master filter, which 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

$$\beta_1 + \beta_2 + \dots + \beta_n = 1, \quad (15)$$

where β_n represents the weights allots to each local filter.

In [10], the weights are allotted based on the innovation of each local filter and are calculated in (4). In this method, the innovations (the difference between observation and estimation) were used to evaluate the performance of the local filter; a larger innovation represents a worse quality of the local filter and allotted a lesser weight in master filter. This method can reduce the affection of the observation noise uncertainty, but this method will fail when the estimation results of the local filters are not accurate enough. Therefore, the weight allocation for individual local filters must consider the estimation performance of individual local filters.

The number of effective particles (N_{eff}) represents the efficiency of the local filter because the value is calculated

Step 1. Initialization, initialize the position of the robot $\mathbf{X}_v$, and the particles $\mathbf{X}_k^j$ in local filters.

for 1 : n

{

Step 2. Reconstruct the local filters

 {

 Delete the local filters without input observation

 Added local filters for new observation

 }

Step 3. Generate the probabilities of the particles

The probabilities of the particles in local filters are calculated by (*). The particles closer to the observation have larger probabilities.

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

 (a) *Update the weights of the particles in time k by (*).*

 (b) *Calculate the variance of the particles and use the variance as parameter $\mathbf{R}_k'$ to recalculate the w_k^{i*} by (*).*

 (c) *Update ω_k^i by (**)*

$$\omega_k^i = \omega_{k-1}^i p(\mathbf{z}_k | \mathbf{x}_k^i) = \omega_{k-1}^i w_k^{i*} \quad (**)$$

Step 4. Evaluate the state in every local filter by (* * *).

$$\mathbf{x}_k = \sum_{i=1}^n \mathbf{x}_k^i * w_k^i \quad (* * *)$$

Step 5. Calculate the evaluation result of the master filter

The result of evaluation calculated by each local filter is transformed to the master filter. And the weight of each local filter is calculated by (17).

Step 6. Save the distribution of the particles in every local filter

 The probabilities of the particles in every local filter are saved to generate the distribution of the particles in next iteration.

Step 7. Resample

 Where $N_{\text{threshold}}$ is the threshold of the number of peffective particle, if the N_{eff_i} of a local filter lower than $N_{\text{threshold}}$, this local filter should be resample.

 if ($N_{\text{eff}_i} < N_{\text{threshold}}$)

 {

 Resample this local filter;

 }

}

ALGORITHM 1: Improved distributed particle filter (Improved DPF).

by the weight of the particles in each local filter, which is indicated in the following:

$$N_{\text{eff}_j} = \frac{1}{\sum_{j=1}^N (w_k^j)^2}, \quad (16)$$

where w_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 N_{eff} of each local filter and is calculated as

$$\beta_j = \frac{N_{\text{eff}_j}}{\sum_{j=1}^N (N_{\text{eff}_j})}. \quad (17)$$

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

3.3. Theoretical Convergence. For the proposed Improved DUPE, the convergence of the particle filter with the distributed implementation is an important issue and requires careful investigation, as it is crucial for the successful applications. In this section, the proof of the mean square convergence of the distributed particle filter to the optimal filter was described. First, the convergence results for particle filter were extended to prove that distributed particle filter convergence to the optimal filter in mean square as the number particles go to infinity.

An extensive treatment of the currently existing mean square convergence results of particle filter can be found in

[17] and [18] These convergence results demonstrate that, under very loose assumptions, convergence of the particle filter is ensured and that the convergence rate of the method is independent of dimension of the state space. The convergence results for particle filter are described as follows.

Let $B(\mathbb{R}^n)$ be the space of bounded, Borel measurable functions on $\mathbb{R}^n$ and denote $\|f\| \triangleq \sup_{x \in \mathbb{R}^n} |f(x)|$. Then the following theorem is a consequence result of the convergence of particle filter.

Theorem 1. *If the importance weight*

$$w_t \propto \frac{p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{x}_{t-1})}{q(\mathbf{x}_t | \mathbf{x}_{0:t-1}, \mathbf{y}_{1:t})} \quad (18)$$

is upper bounded for any $(\mathbf{x}_{t-1}, \mathbf{y}_t)$, then for all $t \geq 0$, there exists c_t independent of N such that for any $f_t \in B(\mathbb{R}^n)$

$$\mathbf{E} \left[\left(\frac{1}{N} \sum_{i=1}^N f_t(\mathbf{x}_{0:t}^i) - \int f_t(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right)^2 \right] \leq c_t \frac{\|f_t\|^2}{N}. \quad (19)$$

The convergence of distributed particle filter is as follows: for any fixed time instance t , under what conditions and for what kind of function $f(x)$ does the distributed particle filter approximation converge to the optimal filter? As mentioned above, distributed particle filter is configured by a number of local particle filters. Therefore, the convergence of distributed particle filter can be proved using Theorem 1 and Deduction 1 for the convergence results of particle filter. In the following, the convergence result for distributed particle filter was proved based on Theorem 1.

Deduction 1. If all of the local particle filters in distributed particle filter are convergence, the distributed particle filter is surely convergence.

Proof. From the mean square convergence, the distributed particle which filter converges to the optimal filter is the estimation results approximate to the conditional expectation $E(f_t(\mathbf{x}_{0:t})/\mathbf{y}_{1:t})$, which can be calculated by

$$E\left(\frac{f_t(\mathbf{x}_{0:t})}{\mathbf{y}_{1:t}}\right) = \int f_t(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}). \quad (20)$$

From Algorithm 1, the conditional expectation estimated by distributed particle filter $\hat{E}(f_t(\mathbf{x}_{0:t})/\mathbf{y}_{1:t})$ can be calculated by

$$\hat{E}\left(\frac{f_t(\mathbf{x}_{0:t})}{\mathbf{y}_{1:t}}\right) = \frac{1}{N} \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) = \frac{1}{N} \sum_{q=1}^m \beta_m \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i), \quad (21)$$

where, $q = 1, 2, \dots, m$ is the number of the local filter.

Thus, the mean square can be calculated in the following:

$$\begin{aligned} & \mathbf{E} \left[\left(\hat{E}\left(\frac{f_t(\mathbf{x}_{0:t})}{\mathbf{y}_{1:t}}\right) - E\left(\frac{f_t(\mathbf{x}_{0:t})}{\mathbf{y}_{1:t}}\right) \right)^2 \right] \\ &= \mathbf{E} \left[\left(\frac{1}{N} \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) - \int f_t(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right)^2 \right] \\ &= \mathbf{E} \left[\left(\frac{1}{N} \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) - \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \int f_t^q(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right. \right. \\ &\quad \left. \left. + \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \int f_t^q(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) - \int f_t(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right)^2 \right], \end{aligned} \quad (22)$$

when $N \rightarrow \infty$, there is

$$\begin{aligned} & \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \int f_t^q(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \\ & - \int f_t(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) = 0. \end{aligned} \quad (23)$$

Thus, we have

$$\begin{aligned} & \mathbf{E} \left[\left(\frac{1}{N} \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) - \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \int f_t^q(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right)^2 \right] \\ &= \mathbf{E} \left[\left(\sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \right. \right. \\ &\quad \left. \left. \times \left(\frac{1}{N} \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) - \int f_t^q(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right) \right)^2 \right] \\ &= \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \mathbf{E} \\ &\quad \times \left[\left(\left(\frac{1}{N} \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) - \int f_t^q(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right) \right)^2 \right], \end{aligned} \quad (24)$$

where, $E[(((1/N) \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) - \int f_t^q(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}))^2]$ is the mean square for each local filter. From Theorem 1, the local filters are convergence; we have

$$E \left[\left(\left(\frac{1}{N} \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) - \int f_t^q(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right) \right)^2 \right] \leq c_t \frac{\|f_t^q\|^2}{N}. \quad (25)$$

Then, (24) is changed to be

$$E \left[\left(\frac{1}{N} \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} \sum_{i=1}^N f_t^q(\mathbf{x}_{0:t}^i) - \int f_t(\mathbf{x}_{0:t}) p(d\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) \right)^2 \right] \leq \sum_{q=1}^m \frac{N_{\text{eff}_q}}{\sum_{q=1}^m N_{\text{eff}_q}} c_t \frac{\|f_t^q\|^2}{N}. \quad (26)$$

It is obvious, from (26), that the distributed particle filter will converge to the optimal filter when all of the local filters are convergence. $\square$

4. Experiment Results

In this section, the proposed Improved DPF-SLAM system was tested in three experiments. The first experiment was used to test the estimation accuracy by comparing it with DPF-SLAM. The second experiment was used to test the robustness of the proposed Improved DPF-SLAM by adding disturbance noise to the estimation process. The third experiment was used to test the tolerance of the Improved DPF-SLAM by comparing it with the exiting methods.

These experiments were tested using the experiment data that come from an experiment that was finished in Victoria Park in Sydney, Australia [19, 20]. The trees can be considered one of the most relevant features that a laser range sensor can identify in this outdoor environment. The algorithm implemented in [17] was used in this paper, which tracks the center of the trunk by clustering a number of laser observations. The extract algorithm was shown in Figures 3 and 4.

Experiment 1. To test the performance of the proposed improved DPF-SLAM, the experiment was finished to be compared with DPF-SLAM.

Figure 5 describes the experiment results of Improved DPF-SLAM and DPF-SLAM. In Figure 5, the straight line indicates the estimation results of robot's position from Improved DPF, and the plus sign indicates the estimation results of feature points' position from Improved DPF. Correspondingly, the dashed line indicates the estimation results of robot's position from DPF, and the rhombus indicates the estimation results of feature points' position from DPF. The stars indicate the measurement results from GPS, which are used as the reference information for these experiments.

From Figure 5, it is obvious that the estimation results of Improved DPF-SLAM are more approximate to the results of

(a)

(b)

(c)

FIGURE 3: Trees with different shape, size and inclination.

GPS than DPF-SLAM, which means that the Improved DPF-SLAM has better accuracy than DPF-SLAM.

Table 1 describes the mean value and variance of the estimation results from these two algorithms. From Table 1, it is obvious that the estimation results from Improved DPF are smaller than the estimation results from DPF in mean and variance, which also means that the Improved DPF-SLAM has better accuracy than DPF-SLAM.

Experiment 2. To test the robustness of the Improved DPF-SLAM, the experiment was finished and the experiment data used in Experiment 2 were the same as Experiment 1. The disturbance noise was added into the SLAM system process from the 1000th step.

Figure 6 describes the estimation results using Improved DPF-SLAM and DPF-SLAM. In Figure 6, the straight line

FIGURE 4: Tree information and trunk approximation.

FIGURE 5: The comparison of estimation results in Experiment 1.

TABLE 1: The comparison of mean and variance in Experiment 1.

Algorithm	MSE	
	Mean	Variance
Improved DPF	2.2367	2.2367
DPF	3.65898	3.65898

indicates the estimation results of robot's position from Improved DPF, and the plus sign indicates the estimation results of feature points' position from Improved DPF. Correspondingly, the dashed line indicates the estimation results of robot's position from DPF, and the rhombus indicate the

TABLE 2: The comparison of mean and variance in Experiment 2.

Algorithm	MSE	
	Mean	Variance
Improved DPF	3.92396	3.18096
DPF	5.41226	4.7737

TABLE 3: The comparison of mean and variance in Experiment 3.

Algorithm	MSE	
	Mean	Variance
Improved DPF	3.3139	3.1364
N_{eff} DPF	4.65898	4.75815
Ino DPF	3.31508	3.20737

estimation results of feature points' position from DPF. The stars indicate the measurement results from GPS, which are used the reference information for these experiments.

From Figure 6, it is obvious that the estimation results of Improved DPF-SLAM are more approximate to the results from GPS than DPF-SLAM, which means that the Improved DPF-SLAM has better robustness than DPF-SLAM in disturbance condition.

Table 2 describes the mean value and variance of the estimation results from these two algorithms. From Table 2, the estimation results from Improved DPF are obviously smaller than the estimation results from DPF in mean and variance, which also means that the Improved DPF-SLAM has better performance than DPF-SLAM in disturbance condition.

Experiment 3. To test the tolerance of the Improved DPF-SLAM, the experiment was finished and the experiment data used in Experiment 2 were the same as Experiment 1. The disturbance noise was added into the SLAM system process from the 1000th step.

Figure 7 describes the estimation results using Improved DPF-SLAM and DPF-SLAM. In Figure 7, the straight line indicates the estimation results of robot's position from Improved DPF, and the plus sign indicates the estimation results of feature points' position from Improved DPF. Correspondingly, the dashed line indicates the estimation results of robot's position from DPF using N_{eff} method, and the rhombus indicates the estimation results of feature points' position from DPF using N_{eff} method. And the dotted line indicates the estimation results of robot's position from DPF using Innovation method, and the triangles indicate the estimation results of feature points' position from DPF using Innovation method. The stars indicate the measurement results from GPS, which are used as the reference information for these experiments.

From Figure 7, it is obvious that the estimation results of Improved DPF-SLAM are more approximate to the results from GPS than DPF-SLAM using N_{eff} method or using Innovation method, which means that the Improved DPF-SLAM has better robustness than these two methods.

Table 3 describes the mean value and variance of the estimation results from these three algorithms. From Table 3,

FIGURE 6: The comparison of estimation results in Experiment 2.

FIGURE 7: The comparison of estimation results in Experiment 3.

it is obvious that the estimation results from Improved DPF are smaller than the estimation results from other methods in mean and variance, which also means that the Improved DPF-SLAM has better accuracy than other methods.

5. Conclusion

In this paper, we proposed an improved particle filter for distributed SLAM to generate a map and localize the robot. Firstly, the variance estimation of particle distribution in local particle filters is a vital parameter in generating the

weights of the particles. An optimized variance estimation of particle distribution will ensure that the particle filter has a better performance. Instead of a fixed parameter in variance estimation, the particles variance was adjusted with the variance estimation adaptively. Secondly, the weight of local filters in master filter is also an important factor in distributed particle filter. The information fusion method mixing the N_{eff} method and the innovation method were proposed by this paper. To test the performance of the Improved DPF-SLAM algorithm, three experiments were designed. The experiment results show that the improved DPF-SLAM algorithm has the advantage in robustness and accuracy and has the better tolerance than the DPF-SLAM algorithm.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National Science Foundation of China under Grant no. 60975065 and by Program for the Top Young Innovative Talents of Beijing CITCD201304046.

References

- [1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.
- [2] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): part II," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.
- [3] S. Huang and G. Dissanayake, "Convergence and consistency analysis for extended Kalman filter based SLAM," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 1036–1049, 2007.
- [4] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.
- [5] S. Garrido, L. Moreno, and D. Blanco, "Exploration and mapping using the VFM motion planner," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 8, pp. 2880–2892, 2009.
- [6] 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," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI '03)*, pp. 1151–1156, Acapulco, Mexico, 2003.
- [7] Y. Zhong, S. Gao, and W. Li, "A quaternion-based method for SINS/SAR integrated navigation system," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 1, pp. 514–524, 2012.
- [8] Y. Ma, H. Ju, and P. Cui, "Research on localization and mapping for lunar rover based on RBPF-SLAM," in *Proceedings of the International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC '09)*, vol. 2, pp. 306–311, Zhejiang, China, August 2009.
- [9] D. H. Won, S. Chun, S. Sung, T. Kang, and Y. J. Lee, "Improving mobile robot navigation performance using vision based SLAM

- and distributed filters,” in *Proceedings of the International Conference on Control, Automation and Systems (ICCAS '08)*, pp. 186–191, Seoul, Republic of Korea, October 2008.
- [10] D. H. Won, S. Chun, S. Sung et al., “INS/vSLAM system using distributed particle filter,” *International Journal of Control, Automation and Systems*, vol. 8, no. 6, pp. 1232–1240, 2010.
- [11] A. Doucet, S. Godsill, and C. Andrieu, “On sequential Monte Carlo sampling methods for Bayesian filtering,” *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.
- [12] M. Isard and A. Blake, “Condensation—conditional density propagation for visual tracking,” *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.
- [13] J. U. Cho, S. H. Jin, X. D. Pham, J. W. Jeon, J. E. Byun, and H. Kang, “A real-time object tracking system using a particle filter,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '06)*, pp. 2822–2827, Beijing, China, October 2006.
- [14] L.-Q. Li, H.-B. Ji, and J.-H. Luo, “The iterated extended Kalman particle filter,” in *Proceedings of the International Symposium on Communications and Information Technologies (ISCIT '05)*, vol. 2, pp. 1213–1216, Beijing, China, October 2005.
- [15] S. J. Julier and J. K. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.
- [16] Q. Cheng and P. Bondon, “A new unscented particle filter,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 3417–3420, Las Vegas, Nev, USA, April 2008.
- [17] X.-L. Hu, T. B. Schön, and L. Ljung, “A basic convergence result for particle filtering,” *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1337–1348, 2008.
- [18] X.-L. Hu, T. B. Schön, and L. Ljung, “A general convergence result for particle filtering,” *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3424–3429, 2011.
- [19] J. E. Guivant and E. M. Nebot, “Optimization of the simultaneous localization and map-building algorithm for real-time implementation,” *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 242–257, 2001.
- [20] J. E. Guivant, F. R. Masson, and E. M. Nebot, “Simultaneous localization and map building using natural features and absolute information,” *Robotics and Autonomous Systems*, vol. 40, no. 2–3, pp. 79–90, 2002.