Hindawi Publishing Corporation
Journal of Sensors
Volume 2014, Article ID 456289, 9 pages
http://dx.doi.org/10.1155/2014/456289

Research Article

An Improved FastSLAM System Based on Distributed Structure for Autonomous Robot Navigation

Fu-jun Pei, Hao-yang Li, and Yu-hang Cheng

Department of Electronic Information and Control Engineering, Beijing University of Technology, Beijing 100022, China

Correspondence should be addressed to Fu-jun Pei; pfj@bjut.edu.cn

Received 25 June 2014; Accepted 30 September 2014; Published 27 October 2014

Academic Editor: Fanli Meng

Fast simultaneous localization and mapping (FastSLAM) is an efficient algorithm for autonomous navigation of mobile vehicle. However, FastSLAM must reconfigure the entire vehicle state equation when the feature points change, which causes an exponential growth in quantities of computation and difficulties in isolating potential faults. In order to overcome these limitations, an improved FastSLAM, based on the distributed structure, is developed in this paper. There are two state estimation parts designed in this improved FastSLAM. Firstly, a distributed unscented particle filter is used to avoid reconfiguring the entire system equation in the vehicle state estimation part. Secondly, in the landmarks estimation part, the observation model is designed as a linear one to update the landmarks states by using the linear observation errors. Then, the convergence of the proposed and improved FastSLAM algorithm is given in the sense of mean square. Finally, the simulation results show that the proposed distributed algorithm could reduce the computational complexity with high accuracy and high fault-tolerance performance.

1. Introduction

Simultaneous localization and mapping (SLAM) is the process of enabling a mobile robot to move through an unknown environment, building a map and estimating the position simultaneously, by estimating the features of the environment. In the last decade, SLAM has attracted lots of attention in the mobile robotics literature and is considered to be a key prerequisite for the truly autonomous robot navigation [1, 2].

In general, there are three main types of SLAM, the extended Kalman filter based SLAM (EKF-SLAM), the particle filter based SLAM (PF-SLAM), and FastSLAM, which have been widely used. In fact, EKF-SLAM is the first SLAM used in the real system and has been well developed over the past two decades. As the well-known algorithm to solve the Gaussian nonlinear problem, it linearizes the nonlinear model using Taylor series expansions. However, the series approximations lead to poor representations of the nonlinear functions due to the fact that EKF cannot solve the Gaussian problem [3]. Then, to own a feasible solution for non-Gaussian nonlinear SLAM, PF-SLAM is proposed in [4]. The particle filter relies on the importance of sampling to approximate the posterior distribution. However, since the feature points will change in the unknown environment, the entire system equation must be reconfigured. The computational complexity of PF-SLAM is a major problem for estimating the robot state and landmarks together. To overcome the former problems, a FastSLAM framework, using the Rao-Blackwellised particle filter (RBPF), is proposed in [5]. FastSLAM is an efficient algorithm for SLAM problems. It decomposes the entire SLAM system into a robot localization problem using PF and a collection of landmarks estimation problems using EKF. The FastSLAM approach separates the robot state from the landmarks state estimate; only the robot state equation needs to be reconfigured when the feature points change in the process. However, many limitations still need to be overcome in FastSLAM. For example, the feasible solution for non-Gaussian nonlinear SLAM and the convergence should be discussed.

In this paper, a new improved FastSLAM based on distributed structure is proposed to overcome the drawbacks of FastSLAM. In the improved FastSLAM algorithm, the SLAM problem is divided into robot localization problem and landmarks estimation problem [6]. In the localization part, the robot state is divided into several parts according to the effective landmarks based on the distributed structure,

which can estimate the distributed states at the same time without reconfiguring and changing the dimension of the robot state equation, as well as improving the fault tolerance [7]. Then, each part of the state is estimated by a local unscented particle filter (UPF) which can permit a set of sigma points to be scaled as an arbitrary scaling parameter [8]. And the computational cost will not be increased in this algorithm. In the landmarks estimation part, since the observation error is used in the updating step, the landmarks estimation can be seen as a linear optimal estimation. The federated Kalman filter, an associated set of independent Kalman filters, estimating the position of landmarks effectively, is used to estimate the landmarks instead of EKF and without the linearization process [9].

The contents of this paper are as follows. In Section 2, the basic properties and theorems of SLAM model and FastSLAM framework are reviewed. Section 3 defines the improved distributed FastSLAM and shows the process of the robot state estimation and the landmarks states estimation. Meanwhile, the weight fusion method for the local filters is given. Section 4 discusses the mean square convergence of the proposed algorithm. Section 5 gives the simulation results and the conclusions are made in Section 6.

2. Background

2.1. SLAM Model. Indeed, the absolute position of the robot cannot be obtained directly; it needs to use the indirect observation from the sensor to estimate the position of the robot while maintaining error stable in a small range [10]. The SLAM algorithm uses dead reckoning and relative observation to estimate the position of the robot and build a navigation map. Generally, the model of SLAM is composed of the motion model and the observation model.

2.1.1. Motion Model. The robot motion model can be described as a probabilistic Markov chain

$$\begin{aligned}\mathbf{x}_r(k) &= \begin{bmatrix} x_r(k) \\ y_r(k) \\ \varphi_r(k) \end{bmatrix} = f\left(\mathbf{x}_r(k-1)\right) + \boldsymbol{\gamma} \\ &= \begin{bmatrix} x_r(k-1) + \Delta x \\ y_r(k-1) + \Delta y \\ \varphi_r(k-1) + \Delta\varphi \end{bmatrix} + \boldsymbol{\gamma},\end{aligned} \tag{1}$$

where $x_r(k)$, $y_r(k)$, $\varphi_r(k)$ are the robot states defined at the external laser sensor at time k, and $\boldsymbol{\gamma}$ is state noise.

The landmarks are assumed to be static, and the landmark motion model is given by

$$\mathbf{x}_I^j(k+1) = \mathbf{x}_I^j(k) = \begin{bmatrix} x_I^j & y_I^j \end{bmatrix}, \tag{2}$$

where $\mathbf{x}_I^j$ is the state of the jth landmark.

2.1.2. Observation Model. Normally, the observation equation relating the robot states to the landmarks is

$$\begin{aligned}\mathbf{z}(k) &= \begin{bmatrix} z_r^j \\ z_\beta^j \end{bmatrix} = h\left(\mathbf{x}_r(k), x_I^j, y_I^j\right) + \boldsymbol{\eta} \\ &= \begin{bmatrix} \sqrt{\left(x_I^j - x_r\right)^2 + \left(y_I^j - y_r\right)^2} \\ \arctan\left(\dfrac{\left(y_I^j - y_r\right)}{\left(x_I^j - x_r\right)}\right) - \varphi_r \end{bmatrix} + \boldsymbol{\eta},\end{aligned} \tag{3}$$

where z is the observation vector. h is the measurement noise. r is the distance from the beacon to the laser sensor and β is the laser sensor bearing measured with respect to the robot coordinate frame. (x_I^j, y_I^j) means the coordinate of the landmarks.

2.2. FastSLAM Framework. The FastSLAM algorithm, introduced by Montemerlo et al., is an efficient algorithm especially for the SLAM problems [11]. It utilizes a particle filter to approximate the ideal recursive Bayesian filter for estimating the robot pose. At each point of time, the FastSLAM algorithm maintains a set of particles to represent the posterior, denoted S_t. Each particle $s_t^{[i]} \in S_t$ represents a possible position of robot

$$S_t = \left\{s_t^{[i]}\right\} = \left\{s_1^{[i]}, s_2^{[i]}, \ldots, s_t^{[i]}\right\}, \tag{4}$$

where $[i]$ refers to the ith particle. And the set of particles in S_t is considered for both the robot control u_t and the measurement z_t, which is denoted by the following sampling distribution:

$$s_t^{[i]} \sim p\left(s_t \mid s^{t-1[i]}, u_t, z_t\right). \tag{5}$$

After obtaining the new set S_t, each particle $s^{t[i]}$ is drawn with a probability proportional called the importance weight $w_t^{[i]}$, and then the final robot position and orientation estimation results can be gotten. Finally, the landmarks estimation is represented by EKF, and this estimation is conditioned on the robot position and orientation.

In the FastSLAM algorithm, the SLAM posterior was partitioned into a localization problem with three positions and orientation parameters of robot and independent landmarks position estimation problem with n landmarks with two state parameters conditioned on the robot position and orientation estimation. So that the update process in FastSLAM only involves a Gaussian of dimension two, whereas a Gaussian of size $2n + 3$ has to be updated in EKF-SLAM [3].

However, the FastSLAM algorithm also suffers from three limitations. Firstly, the FastSLAM algorithm belongs to the centralized approach. The robot position and orientation n estimation are considered under measurement information. Since the number of effective landmarks is changing over time, the dimension of robot state equation must be reconfigured. It is difficult for the centralized approach to deal with the reconfiguration in runtime. Secondly, the fault tolerance of centralized approach is low. Thirdly, since EKF is used for

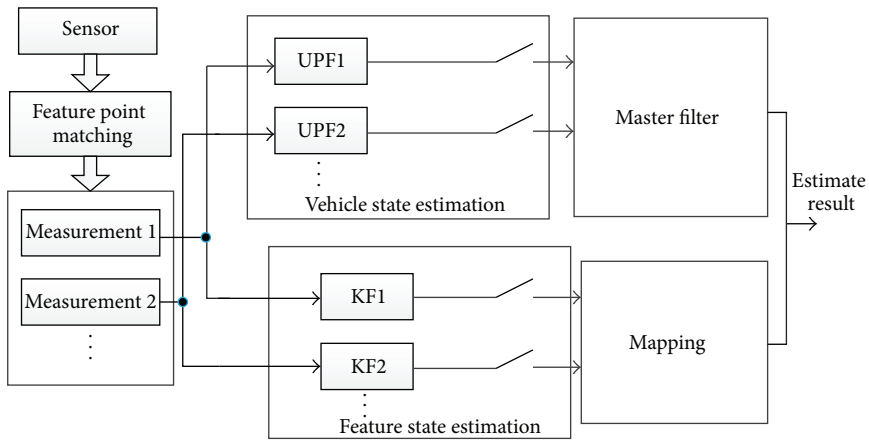

Figure 1: Framework of the distributed FastSLAM.

landmarks estimation, it needs to approximates the nonlinear model using a linear function and take the derivative of the Jacobian matrices with complicated calculations.

3. Improved FastSLAM Based on Distributed Structure

In this section, the improved FastSLAM algorithm is introduced. The improved distributed FastSLAM is based on the distribution structure. And the federated Kalman filter algorithm is used to replace the EKF in FastSLAM. This improved FastSLAM algorithm consists of three parts: the robot state estimation, the landmarks state estimation, and the importance weights calculation. Figure 1 shows the framework of the proposed improved FastSLAM.

3.1. Robot State Estimation. Since it is complex for a centralized approach to handle the changing dimension of the state vectors, UPF approach is used to do the robot state estimation, which is based on the distributed structure. The state is divided into several parts based on the effective landmarks. Each part of the state is estimated by a local UPF filter which is separated from the others. Each estimated result of the states is transmitted to the master filter to make the final results. Even if there is an error in one part, it would not affect the estimated results of the other parts. The distributed algorithm can reduce the effect of error messages on the final estimated result, improve the fault tolerance, respond effectively to the changes in the number of effective landmarks, and have a better accuracy:

$$\begin{aligned} \dot{\mathbf{x}}_r &= f(\mathbf{x}_r) + \boldsymbol{\gamma}, \\ \mathbf{z}_1 &= h(\mathbf{x}_r(k), \mathbf{m}_1) + \boldsymbol{\eta}, \\ \dot{\mathbf{x}}_r &= f(\mathbf{x}_r) + \boldsymbol{\gamma}, \\ \mathbf{z}_2 &= h(\mathbf{x}_r(k), \mathbf{m}_2) + \boldsymbol{\eta}, \\ &\vdots \\ \dot{\mathbf{x}}_r &= f(\mathbf{x}_r) + \boldsymbol{\gamma}, \\ \mathbf{z}_j &= h(\mathbf{x}_r(k), \mathbf{m}_j) + \boldsymbol{\eta}. \end{aligned} \tag{6}$$

Based on the distributed structure, the detail motion model for robot states estimation is derived as (6). The local UPF filters are built based on the effective landmarks m separately, and the process and measurement noises are added to each state vector. Then, each local filter deterministically extracts the sigma points from the Gaussian distribution. After that, the estimated mean $\mathbf{x}_k^i$ and the covariance $\mathbf{P}_k^i$ of each state vector are updated according to the UKF algorithm [12]. The updated mean and covariance are remarked as $\overline{\mathbf{x}}_k^i$ and $\overline{\mathbf{P}}_k^i$, and then they are used to optimize the distribution of the particles in every local filter. From the Gaussian distribution generated by the estimated mean and covariance of the robot, the particles of each local filter are resampled by

$$\mathbf{x}_k^i \sim N\left(\overline{\mathbf{x}}_k^i, \overline{\mathbf{P}}_k^i\right). \tag{7}$$

Then, the next steps are calculating the weight of each local filter, resampling as normal particle filter algorithm and doing results fusion [13]. Finally, the estimation results of the robot state can be obtained accurately.

3.2. Landmarks State Estimation. In the normal FastSLAM, since the measurement is nonlinear as (3), EKF is used to do the landmarks state estimation and approximate the nonlinear process as the linear process. However, the linearization process needs to calculate the Jacobian matrices and has a high linear approximation error, which will increase the computational complexity while reducing the estimation accuracy [14]. In the proposed algorithm, a new observation model is proposed on the basis of FastSLAM algorithm. Because the estimation process of landmarks is separated from the robot state estimation process in FastSLAM, the estimated results of the robot state with high accuracy can be taken as given in the landmarks estimation process. Since the position of landmarks can be calculated by robot pose and the distance between the robot and landmark obtained by the laser sensor and there is an observation error in measurement results, the new observation model can be given by

$$\mathbf{z}(k) = \mathbf{x}_I(k) = \begin{bmatrix} x_r(k) + \Delta x \\ y_r(k) + \Delta y \end{bmatrix} = \mathbf{x}_r(k) + \boldsymbol{\nu}, \tag{8}$$

where Δx and Δy are the distance between the robot and landmarks obtained by the laser sensor. ν is the measurement noise.

Since the landmarks are assumed to be static and the motion model is given by (2), the state estimation model of each landmark is shown as

$$\begin{aligned} x_I(k) &= x_I(k-1), \\ \mathbf{z}(k) &= \mathbf{x}_I(k) + \boldsymbol{\nu}. \end{aligned} \tag{9}$$

From (9), since the motion model and observation model of landmarks are linear in the part of landmarks estimation, then the landmarks estimation process can be seen as a linear optimal estimation problem, and Kalman filter is the best method for this problem. So the Kalman filter is used to estimate the landmarks position to avoid the complex calculation of Jacobian matrices in EKF. However, since the feature points change, it is difficult for the centralized Kalman filter algorithm to deal with it and the centralized approach does not get a good fault tolerance.

In this paper, the measurement information provided by the laser sensor may include outliers and the landmarks need to be estimated at the same time; the federated filter [15] is used here to estimate the states of landmarks for its stronger fault tolerance and its capability of dealing with several different estimations simultaneously. And Kalman filter is used as the local filter of the federated filter which can reduce the computation complexity while maintaining the estimation accuracy. According to every estimated result of local UPF filters and observation information, the corresponding landmarks are updated by local Kalman filters separately. The motion model for landmarks state estimation is as follows:

$$\begin{aligned} \mathbf{x}_I^1(k) &= \mathbf{x}_I^1(k-1), \\ \mathbf{z}_1(k) &= \mathbf{x}_I^1(k) + \boldsymbol{\nu}, \\ \mathbf{x}_I^2(k) &= \mathbf{x}_I^2(k-1), \\ \mathbf{z}_2(k) &= \mathbf{x}_I^2(k) + \boldsymbol{\nu}, \\ &\vdots \\ \mathbf{x}_I^j(k) &= \mathbf{x}_I^j(k-1), \\ \mathbf{z}_j(k) &= \mathbf{x}_I^j(k) + \boldsymbol{\nu}, \end{aligned} \tag{10}$$

where j refers to the jth landmark.

The update process of the landmark in each local filter is the same as the normal Kalman filter algorithm [16]. After the update process, all of the local Kalman filter estimation results will be combined to calculate the optimal estimated results of each landmark.

3.3. Importance of Weights Calculation. Generally, in distributed particle filters, the Neff method is used to calculate the importance of weights of every local filter [8]. The Neff method can be defined as follows:

$$\alpha_N = \frac{\mathrm{Neff}_j}{\sum_{j=1}^{S}\left(\mathrm{Neff}_j\right)}, \qquad \mathrm{Neff}_j = \frac{1}{\sum_{i=1}^{N}\left(w_k^{ij}\right)^2}, \tag{11}$$

where Neff_j is the number of effective particles in the jth filter. w_k^{ij} is the weight of each particle in the jth filter.

However, the Neff method distributes the weights through calculating the degree of the particles degeneracy of each filter; it is closely related to the number of particles. If in the case of a small number of particles, it cannot effectively response to the real weights distribution. Therefore, a new method is advanced which combines the Neff method with the innovation method in order to obtain a high precision of real weights distribution. The new method can be calculated by

$$\alpha_{\mathrm{Ne}} \cdot \lambda + \alpha_{\mathrm{In}} \cdot \chi = \alpha_M, \qquad \lambda + \chi = 1, \tag{12}$$

where α_{Ne} is the weight generated by Neff algorithm, α_{In} is the weight generated by innovation algorithm, α_M is the mixed weight, and λ and χ are the proportion parameters, which stand for the proportion of these two methods in the mixed method, and the relationship between these two parameters can be defined as (12).

The weight of each local filter based on the innovation method is calculated by (13), and the innovation is the difference between the observation and the estimation:

$$\alpha_{\mathrm{In}} = \frac{1/\mathrm{innovation}_j}{\sum_{j=1}^{S} 1/\mathrm{innovation}_j}. \tag{13}$$

4. Mean Square Convergence

For the proposed distributed FastSLAM, the convergence of the unscented 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 unscented particle filter to the optimal filter was described. First, the convergence results from particle filters were extended to prove the convergence of distributed particle filter in mean squares as the number particles go to infinity. Second, the mean square convergence of the distributed FastSLAM was proved by considering in turn each of the steps of prediction, measurement, and resampling.

The generic proof of distributed unscented particle filter convergence is as follows.

Theorem 1. *If the importance of weight $w_t \propto p(\mathbf{y}_t \mid \mathbf{x}_t)p(\mathbf{x}_t \mid \mathbf{x}_{t-1})/q(\mathbf{x}_t \mid \mathbf{x}_{0:t-1}, \mathbf{y}_{1:t})$ is an upper bounded for any $(\mathbf{x}_{t-1}, \mathbf{y}_t)$ and if one uses one of the selection schemes described previously, then, for all $t \geq 0$, there exists c_t independent of $> N$ such that for any $f_t \in B(\mathbb{R}^{n_x \times (t+1)})$*

$$\mathbf{E}\left[\left(\frac{1}{N}\sum_{i=1}^{N} f_t(\mathbf{x}_{0:t}^{(i)}) - \int f_t(\mathbf{x}_{0:t})p(d\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t})\right)^2\right] \leq c_t \frac{\|f_t\|^2}{N}, \tag{14}$$

where $B(\mathbb{R}^n)$ is the space of bounded, Borel measurable functions on $\mathbb{R}^n$. We denote $\|f\| \triangleq \sup|f(x)|$, $x \in \mathbb{R}^n$. Equation (14) is the mean square convergence of the filters. This convergence result shows that, under very loose assumptions, convergence of the unscented particle filter is ensured and the only crucial assumption is to ensure that w_t is upper bounded. Hence, the weight of the distributed FastSLAM proposed must prove that it is upper bounded firstly.

Proof. The weight w_t' of the distributed FastSLAM is defined by

$$w_t' = \sum_{j=1}^{S} w_{tj} \cdot \alpha_{Mj}, \tag{15}$$

where w_{tj} is the weight to the jth local filter, α_{Mj} is the weights distribution for the new method in (12) to the jth local filter, and S is the number of local filters.

Because of the principle of information fusion, the weight distribution sets to individual local filters should be described by $\sum_{j=1}^{S} \alpha_{Mj} = 1$ and then $\alpha_{Mj} < 1$. Because each local filter is UPF which has the same structure and the literature has proved that the weight of UPF is an upper bounded [8], so we can get $\{w_{tj}\}_{j=1}^{s} \le C_1$, where C_1 is constant. Because S is also the number of features matching, which must be upper bounded, we suppose $S \le C_2$, where C_2 is constant. Then we have $w_t' = \sum_{j=1}^{S} w_{tj} \cdot \alpha_{Mj} \le \sum_{j=1}^{S} w_{tj} \le S \cdot w_{t\max} \le C_1 \cdot C_2 \le C$. Thus, the weight of the distributed FastSLAM is an upper bounded, so that the mean square error is convergent.

Then, the mean square convergence of the distributed FastSLAM will be proved by considering in turn each of the steps of prediction, measurement, and resampling. This proof is a modification of [17, 18]. The main difference from that proof is that the effect of UKF must be considered and an upper bound can be obtained.

The prediction update step will begin with the following lemma. The mean square error can be supposed by $\mathbf{E}[|(\pi_t^N, \varphi) - (\pi_t, \varphi)|^2]$ for any $\varphi \in B(\mathbb{R}^n)$. So the mean square error in the prediction update step can be defined as $\mathbf{E}[|(\pi_{t|t-1}^N, \varphi) - (\pi_{t|t-1}, \varphi)|^2]$. □

Proof. We have

$$\begin{aligned}
&\left|\left(\pi_{t|t-1}^N, \varphi\right) - \left(\pi_{t|t-1}, \varphi\right)\right| \\
&\quad = \left|\left(\pi_{t|t-1}^N, \varphi\right)\left(-\left(\pi_{t-1|t-1}^N, \kappa\varphi\right) + \left(\pi_{t-1|t-1}^N, \kappa\varphi\right)\right)\right. \\
&\qquad \left. - \left(\pi_{t-1|t-1}, \kappa\varphi\right)\right| \\
&\quad \le \left|\left(\pi_{t|t-1}^N, \varphi\right) - \left(\pi_{t-1|t-1}^N, \kappa\varphi\right)\right| \\
&\qquad + \left|\left(\pi_{t-1|t-1}^N, \kappa\varphi\right) - \left(\pi_{t-1|t-1}, \kappa\varphi\right)\right|.
\end{aligned} \tag{16}$$

If $\mathbf{G}_{t-1}$ is the σ-algebra generated by $\{x_{t-1}^i\}_{i=1}^N$, then $(\pi_{t-1|t-1}^N, \kappa\varphi) = \mathbf{E}[(\pi_{t|t-1}^N, \varphi) \mid \mathbf{G}_{t-1}]$.

The first term on the right hand side of (16) is bounded above as follows:

$$\begin{aligned}
&\mathbf{E}\left[\left|\left(\pi_{t|t-1}^N, \varphi\right) - \left(\pi_{t-1|t-1}^N, \kappa\varphi\right)\right|^2\right] \\
&\quad = \frac{1}{N^2}\sum_{i=1}^{N}\mathbf{E}\left[\left|\varphi(x_{t-1}^i) - \mathbf{E}\left[\varphi\left(x_{t-1}^i\right) \mid \mathbf{G}_{t-1}\right]\right|^2\right] \\
&\quad \le \frac{4}{N^2}\sum_{i=1}^{N}\mathbf{E}\left[\varphi^2\left(x_{t-1}^i\right) \mid \mathbf{G}_{t-1}\right] \\
&\quad = \frac{4}{N}\left(\pi_{t-1|t-1}^N, \kappa\varphi^2\right).
\end{aligned} \tag{17}$$

Since Markov operators are contractions $\|\kappa\varphi\| \le \|\varphi\|$ [19], so

$$\mathbf{E}\left[\left|\left(\pi_{t|t-1}^N, \varphi\right) - \left(\pi_{t-1|t-1}^N, \kappa\varphi\right)\right|^2\right] \le \frac{4}{N}\|\varphi\|^2. \tag{18}$$

The upper bound of the second term (16) is

$$\mathbf{E}\left[\left|\left(\pi_{t-1|t-1}^N, \varphi\right) - \left(\pi_{t-1|t-1}, \varphi\right)\right|^2\right] \le \|\varphi\|^2\left(C\|f\|^2 + \frac{c_{t-1|t-1}}{N}\right), \tag{19}$$

so that $\mathbf{E}[|(\pi_{t|t-1}^N, \varphi) - (\pi_{t|t-1}, \varphi)|^2] \le \|\varphi\|^2(C\|f\|^2 + c_{t|t-1}/N)$, where $c_{t|t-1} = c_{t-1|t-1} + 4$.

Assume that, for any $\varphi \in B(\mathbb{R}^n)$, the mean square error for the UKF step can be defined as $\mathbf{E}[|(\pi_{t|t-1}^N, f_{t,s}\varphi) - (\pi_{t|t-1}, f\varphi)|^2]$. □

Proof. From Minkowski's inequality,

$$\begin{aligned}
&\mathbf{E}\left[\left|\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right) - \left(\pi_{t|t-1}, f\varphi\right)\right|^2\right] \\
&\quad \le \mathbf{E}\left[\left|\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right) - \left(\pi_{t-1|t-1}^N, f\varphi\right)\right|^2\right] \\
&\qquad + \mathbf{E}\left[\left|\left(\pi_{t-1|t-1}^N, f\varphi\right) - \left(\pi_{t-1|t-1}, f\varphi\right)\right|^2\right].
\end{aligned} \tag{20}$$

The first term (20) on the right hand side is bounded as follows:

$$\begin{aligned}
&\mathbf{E}\left[\left|\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right) - \left(\pi_{t|t-1}^N, f\varphi\right)\right|^2\right] \\
&\quad = \mathbf{E}\left[\frac{1}{N^2}\left|\sum_{i=1}^{N} f_{t,s}\left(x_t^i\right)\varphi\left(x_t^i\right) - f\left(x_t^i\right)\varphi\left(x_t^i\right)\right|^2\right] \\
&\quad \le \mathbf{E}\left[\frac{\|\varphi\|^2}{N}\sum_{i=1}^{N}\left(f_{t,s}\left(x_t^i\right) - f\left(x_t^i\right)\right)^2\right] \\
&\quad \le \widetilde{C}\|\varphi\|^2\|f\|^2.
\end{aligned} \tag{21}$$

Using the result of the prediction step, we can get

$$\begin{aligned}
&\mathbf{E}\left[\left|\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right) - \left(\pi_{t|t-1}, f\varphi\right)\right|^2\right] \\
&\quad \le \widetilde{C}\|\varphi\|^2\|f\|^2 + \|\varphi\|^2\left(C\|f\|^2 + \frac{c_{t|t-1}}{N}\right) \\
&\quad = \|\varphi\|^2\left(C_1\|f\|^2 + \frac{c_{t|t-1}}{N}\right),
\end{aligned} \tag{22}$$

where $C_1 = \widetilde{C} + C$.

Figure 2: Trees with different shape, size, and inclination.

The next distributed FastSLAM step is a measurement update, and the mean square error can be provided as $\mathbf{E}[|(\pi_{t|t}^N, \varphi) - (\pi_{t|t}, \varphi)|^2]$. □

Proof. We have

$$\begin{aligned}
&\left(\tilde{\pi}_{t|t}^N, \varphi\right) - \left(\pi_{t|t}, \varphi\right) \\
&\quad = \frac{\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right)}{\left(\pi_{t|t-1}^N, f_{t,s}\right)} - \frac{\left(\pi_{t|t-1}, f\varphi\right)}{\left(\pi_{t|t-1}, f\right)} \\
&\quad = \frac{\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right)\left(\left(\pi_{t|t-1}, f\right) - \left(\pi_{t|t-1}^N, f_{t,s}\right)\right)}{\left(\pi_{t|t-1}^N, f_{t,s}\right)\left(\pi_{t|t-1}, f\right)} \\
&\qquad + \frac{\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right) - \left(\pi_{t|t-1}, f\varphi\right)}{\left(\pi_{t|t-1}, f\right)}.
\end{aligned} \tag{23}$$

From the result of the UKF step, we can get

$$\begin{aligned}
&\mathbf{E}\left[\left|\left(\tilde{\pi}_{t|t}^N, \varphi\right) - \left(\pi_{t|t}, \varphi\right)\right|^2\right] \\
&\quad \le \mathbf{E}\left[\left(\frac{\|\varphi\|\left|\left(\pi_{t|t-1}, f\right) - \left(\pi_{t|t-1}^N, f_{t,s}\right)\right|}{\left(\pi_{t|t-1}, f\right)}\right.\right. \\
&\qquad\qquad \left.\left. + \frac{\left|\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right) - \left(\pi_{t|t-1}, f\varphi\right)\right|}{\left(\pi_{t|t-1}, f\right)}\right)^2\right] \\
&\quad \le 2\mathbf{E}\left[\frac{\|\varphi\|^2}{\left(\pi_{t|t-1}, f\right)^2}\left|\left(\pi_{t|t-1}^N, f_{t,s}\varphi\right) - \left(\pi_{t|t-1}, f\varphi\right)\right|^2\right] \\
&\quad \le 2\frac{\|\varphi\|^2}{\left(\pi_{t|t-1}, f\right)^2}\left(C_1\|f\|^2 + \frac{c_{t|t-1}}{N}\right) \\
&\quad = \|\varphi\|^2\left(C_2\|f\|^2 + \frac{\tilde{c}_{t|t-1}}{N}\right),
\end{aligned} \tag{24}$$

where $C_2 = 4C_1/(\pi_{t|t-1}, f)^2$, $\tilde{c}_{t|t-1} = 4c_{t|t-1}/(\pi_{t|t-1}, f)^2$.

The last step is the resampling step, and the mean square error can be provided as $\mathbf{E}[|(\pi_{t|t}^N, \varphi) - (\pi_{t|t}, \varphi)|^2]$. □

Proof. We have

$$\begin{aligned}
&\left(\pi_{t|t}^N, \varphi\right) - \left(\pi_{t|t}, \varphi\right) \\
&\quad = \left(\pi_{t|t}^N, \varphi\right) - \left(\tilde{\pi}_{t|t}^N, \varphi\right) + \left(\tilde{\pi}_{t|t}^N, \varphi\right) - \left(\pi_{t|t}, \varphi\right).
\end{aligned} \tag{25}$$

If F_t is a σ-algebra generated by $\{\tilde{x}_t^i\}_{i=1}^N$, then $(\tilde{\pi}_{t|t}^N, \varphi) = \mathrm{E}[(\tilde{\pi}_{t|t}^N, \varphi) \mid F_t]$.

Thus, by the same procedure from (17), for some constant $\overline{C}$, finally we have

$$\mathbf{E}\left[\left|\left(\pi_{t|t}^N, \varphi\right) - \left(\pi_{t|t}, \varphi\right)\right|^2\right] \le \|\varphi\|^2\left(C_2\|f\|^2 + \frac{c_{t|t}}{N}\right), \tag{26}$$

where $c_{t|t} = \tilde{c}_{t|t-1} + \overline{C}$.

Combing the above results, the following main conclusion can be gotten where the mean square error of the distributed FastSLAM is the upper bounded, and the conclusion is as follows:

$$\mathbf{E}\left[\left|\left(\pi_{t|t}^N, \varphi\right) - \left(\pi_{t|t}, \varphi\right)\right|^2\right] \le \|\varphi\|^2\left(C\|f\|^2 + \frac{c_{t|t-1}}{N}\right). \tag{27}$$

□

5. Experiment Results

To verify the accuracy of the improved algorithm, a simulation experiment was finished in this section. We compared the estimated results of the normal FastSLAM algorithm, the distribute SLAM (DPF-SLAM) algorithm, and the distributed FastSLAM algorithm proposed in this paper.

These experiments were tested using the experiment data come from an experiment which was taken at Victoria Park in Sydney, Australia [20, 21]. The trees can be considered one of the most relevant features that a laser sensor can identify with this outdoor environment. The algorithm implemented in [20] was used in this paper, which tracks the center of the trunk by clustering a number of laser observations. The extraction algorithm of the center was shown in Figures 2 and 3.

The vehicle was started at a location with known uncertainty and driven in this area for more than 300 sec. The

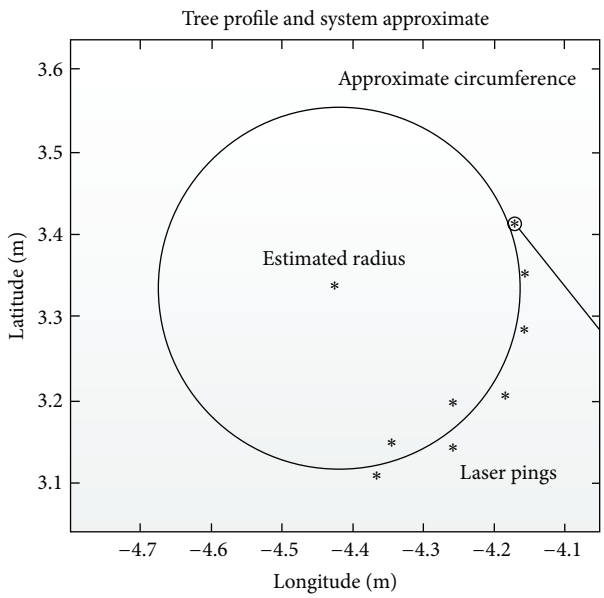

Figure 3: Tree information and trunk approximation.

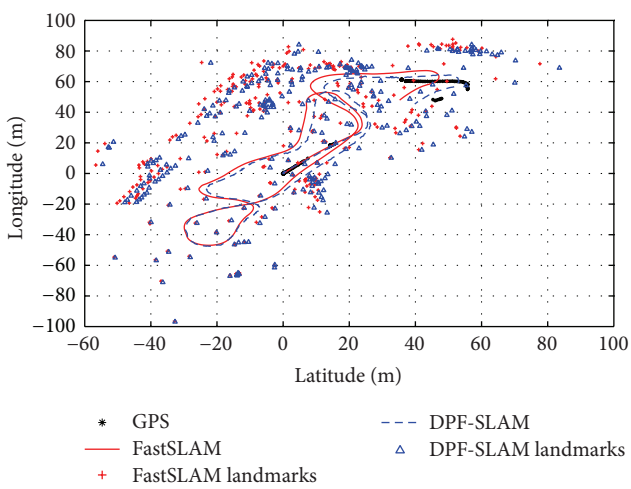

Figure 4: Vehicle trajectory and landmarks estimated position under three different algorithms.

estimated results of the vehicle trajectory and navigation landmarks were given by three algorithms as shown in Figures 4 and 5. We used the GPS information on good quality of the standard data compared with the results estimated by the three different algorithms separately, so as to compare the accuracy of these algorithms directly. From Figures 4 and 5 we can see that all of the three algorithms can match the GPS standard data very well at the beginning of the estimated trajectory. However, as time went on the normal FastSLAM algorithm and the normal DPF-SLAM algorithm could not match the GPS standard data, especially after 148 sec, while the improved distributed FastSLAM algorithm can make better estimation results, although the accuracy of the normal DPF-SLAM algorithm is better than that of the FastSLAM algorithm. Since GPS was in the shade, it could be seen

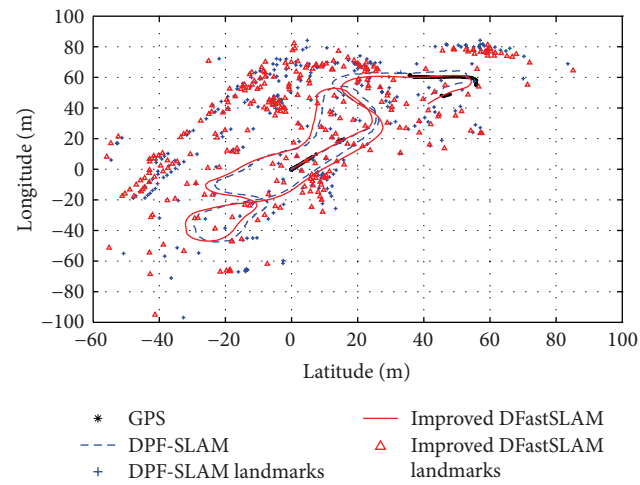

Figure 5: Vehicle trajectory and landmarks estimated position under three different algorithms.

Table 1: Statistical data for experiment results.

SLAM algorithm	RMSE/m	
	X	Y
FastSLAM	7.1278	3.5215
DPF-SLAM	5.4257	2.4136
Distributed FastSLAM	2.2704	0.8313

that GPS information was not available in some areas. The landmarks were used here to reduce the navigation errors.

The root mean square error (RMSE) of the vehicle position of three algorithms between 148 sec and 160 sec in the directions x and y, compared with GPS standard data, is shown in Figure 6, and Table 1 shows the statistical data of RMSE in Figure 6. The RMSE of the normal FastSLAM algorithm is higher than the normal DPF-SLAM algorithm, and the improved distributed FastSLAM algorithm has a little estimated error compared to the DPF-SLAM algorithm. Thus, we know that the performance of the improved distributed FastSLAM system is better than the others.

These results demonstrate that the improved distributed FastSLAM algorithm is workable and stable and has a higher accuracy at the same time.

6. Conclusion

This paper proposed a distributed FastSLAM algorithm as an effective solution to simultaneous localization and mapping. In the localization process, the distributed structure was used to reduce the effect of error messages on the final estimated results and improved the fault tolerance of the SLAM. The unscented Kalman filter was implemented in the prediction step of the robot state and provided a better proposal distribution. In the landmarks estimation process, the nonlinear landmark observation model was changed into linear models, and federated Kalman filter was implemented in this process to estimate the state of landmarks for its

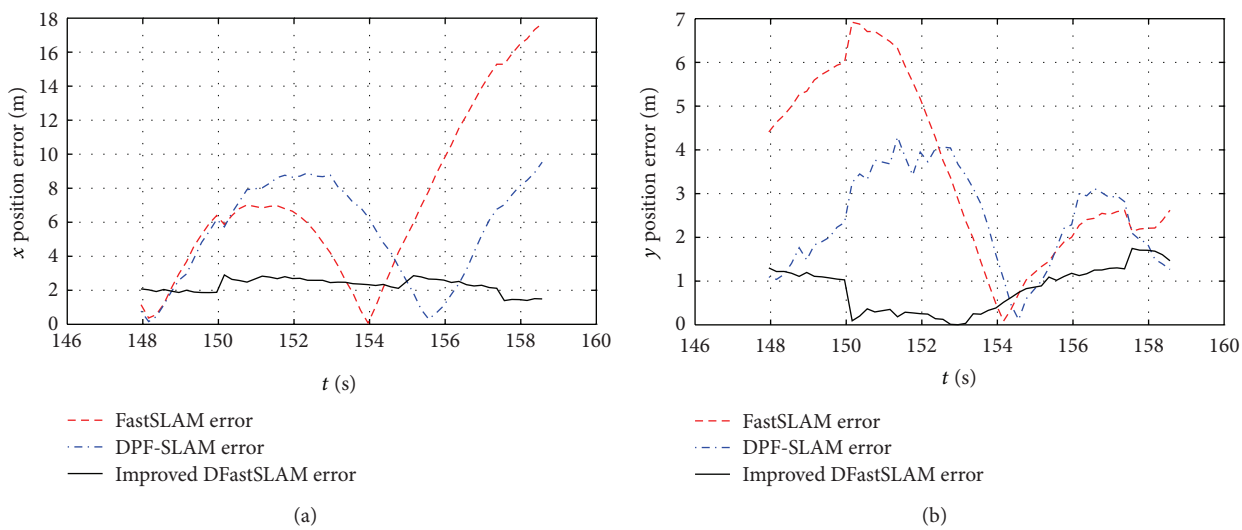

Figure 6: RMSE of the vehicle state estimation under the three algorithms.

stronger fault tolerance and its capability of dealing with several different estimations at the same time. The convergence of the proposed distributed FastSLAM was proved in the sense of mean squares. The simulation results showed that the proposed distributed FastSLAM algorithm has better estimation accuracy than the previous algorithms, and more accurate estimation could be achieved consistently for longer time periods. The simulation results also proved that the proposed algorithm can reduce the computation complexity and maintain the estimation accuracy.

Conflict of Interests

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

Acknowledgments

The authors would like to thank the University of Sydney for the use of the Victoria Park dataset. This work was supported by the National Science Foundation of China under Grant 60975065 and supported by Program for the Top Young Innovative Talents of Beijing CITTCD201304046.

References

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping. Part I," *IEEE Robotics and 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 and 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] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering methods for practitioners," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 736–746, 2002.

[5] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, and E. Nebot, "FastSLAM: an efficient solution to the simultaneous localization and mapping problem with unknown data association," *Journal of Machine Learning Research*, vol. 4, no. 3, pp. 380–407, 2004.

[6] C. Kim, R. Sakthivel, and W. K. Chung, "Unscented FastSLAM: a robust and efficient solution to the SLAM problem," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 808–820, 2008.

[7] 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.

[8] R. Merwe, A. Doucet, N. Freitas, and E. Wan, "The unscented particle filter," Tech. Rep. CUED/F-INFENG/TR 380, Cambridge University, Engineering Department, Cambridge, UK, 2000.

[9] D. A. Castanon and D. Teneketzis, "Distributed estimation algorithms for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 30, no. 5, pp. 418–425, 1985.

[10] M. W. M. Gamini 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.

[11] M. Montemerlo, S. Thrun, and D. Koller, "FastSLAM: a factored solution to the simultaneous localization and mapping problem," in *Proceedings of the National Conference on Artificial Intelligence (AAAI '02)*, pp. 593–598, 2002.

[12] S. A. Holmes, G. Klein, and D. W. Murray, "An $O(N^2)$ square root unscented kalman filter for visual simultaneous localization and mapping," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 7, pp. 1251–1263, 2009.

[13] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[14] C. Kim, R. Sakthivel, and W. K. Chung, "Unscented FastSLAM: a robust and efficient solution to the SLAM problem," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 808–820, 2008.

[15] Y. H. Hu and Z. K. Shi, "New method for correcting location error of unmanned aerial vehicle (UAV) using federated filter

data fusion," *Journal of Northwestern Polytechnical University*, vol. 21, no. 4, pp. 422–426, 2003.

[16] E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME—Journal of Basic Engineering, Series D*, vol. 82, pp. 35–45, 1960.

[17] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering methods for practitioners," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 736–746, 2002.

[18] D. Steinsaltz, "Convergence of moments in a Markov-chain central limit theorem," *Indagationes Mathematicae*, vol. 12, no. 4, pp. 533–555, 2001.

[19] A. Lasota and M. C. Mackey, *Chaos, Fractals, and Noise: Stochastic Aspects of Dynamics*, Springer, New York, NY, USA, 1994.

[20] J. Guivant, E. Nebot, and H. Durrant-Whyte, "Simultaneous localization and map building using natural features in outdoor environments," in *Proceedings of the IAS-6 Intelligent Autonomous Systems*, pp. 581–586, Venice, Italy, 2000.

[21] 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.