Proceedings of the 2010 IEEE
International Conference on Robotics and Biomimetics
December 14-18, 2010, Tianjin, China

Fast Large-scale SLAM with Improved Accuracy in Mobile Robot

Bo Zhao, Xiaorui Zhu, *Member, IEEE*

***Abstract*—A Fast Map Joining algorithm (FMJ) is proposed in this paper to achieve the Simultaneous Localization and Mapping (SLAM) of mobile robot in the large-scale environments. The proposed algorithm can efficiently improve the accuracy of the SLAM and reduce the computational load compared with the standard extended Kalman filter (EKF) SLAM. The FMJ SLAM algorithm divides the global map into a sequence of local sub-maps whose sizes are determined according to the density of the features in the environment. The final localization and mapping is achieved once the sub-maps are jointed accordingly. Simulations are conducted to validate the proposed technique.**

Index Terms: Large-scale, EKF, SLAM, Fast Map Joining.

I. INTRODUCTION

THE Simultaneous Localization and Mapping (SLAM) algorithms are widely used in different kinds of robotic systems such as underwater robots, ground-moving robots and space robots. In SLAM, the robot acquires a map of its environment while simultaneously localizing itself relative to this map [1, 2]. The most popular solution to SLAM is to take advantage of extended Kalman filter (EKF) to estimate the current pose of the mobile robot and the environment feature locations together with the corresponding covariance matrices representing the uncertainty of those estimations. Although EKF-SLAM is a good balance of accuracy and computational load in most cases, its application is limited in the large scale environments where the computational cost is significantly increased as the size of the map increases [3]. This paper, therefore, presents a Fast Map Joining algorithm (FMJ) to reduce the computation load of the SLAM with increased accuracy for large scale environments.

Much attention has been paid to SLAM since the late 1980's [4-7] because solving SLAM is central to the effort of conferring real autonomy to robots and vehicles. The applications of SLAM are recently extended to outdoors where large scale is one of main characteristics. Accuracy and computation load are two common factors to evaluate the efficiency of real-time SLAM. For the standard feature-based EKF-SLAM, the computational cost can be represented as $O(n^3)$ where n is the size of the map. In large scale environments, the computational cost would become too large to implement in the real-time. Therefore, how to reduce the computation cost of large scale SLAM has been a hot topic in the field.

The compressed EKF algorithm [8] can reduce the computational requirements only when working in local areas or with high frequency external sensors. The EKF sequential map joining SLAM was first proposed in [9] to deal with large scale SLAM problems by dividing the global map into the local sub-maps and joining those sub-maps sequentially. Overall computational savings, however, can be achieved only if the size of local map can be optimally selected [10]. Some researchers then propose the Sparse Local Sub-map Joining Filter (SLSJF) [11] to combine the advantages of the sequential submap joining algorithm and the extended information filter (EIF) to substantially reduce the computational cost in the global environment. However, SLSJF may still produce inconsistent estimate in very large loop closing, i.e. the approximation can yield overconfident estimations of the state [12]. Although the Unscented Kalman Filter (UKF) SLAM has been shown to have improved consistency properties, the computational complexity was not considered [13]. Divide & Conquer (D&C) SLAM was recently proposed in [3] to reduce the computational load without scarifying the precision of the performance where the local sub-maps are joined hierarchically.

Therefore, this paper proposes the FMJ SLAM algorithm that will be the extension of D&C to greatly improve the accuracy of the SLAM while keeping the computation fast. The FMJ SLAM algorithm adopts a sequence of local sub-maps with the size of each sub-map that is determined according to the density of the features observed in the environment. These local maps are then jointed hierarchically to achieve the global localization and mapping. This proposed algorithm can be thus more flexible and efficient in terms of fast computation and accuracy than the fixed local sub-map size in the above techniques.

The main contribution of this paper involves Fast Map Joining SLAM architecture that is applicable to the large-scale outdoor environments with general distribution

This research is supported by the NSF China Grant No. 60905052 and by Guangdong Province NSF No.8451805707000687.

Xiaorui Zhu is the corresponding author with Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, Guangdong 518055, China (e-mail: xiaoruizhu@hitsz.edu.cn).

978-1-4244-9317-3/10/$26.00 © 2010 IEEE

of landmarks.

The structure of the paper is organized as follows. Section II introduces the motion model and the measurement model of the mobile robot. The standard EKF SLAM algorithm is also briefly introduced here for comparison in simulation. In Section III, the FMJ SLAM algorithm is proposed and analyzed. Simulations and discussions are presented in Section IV for verification. Finally, the concluding remarks of this paper and the future work are described in Section V.

II. BACKGROUND

A. Velocity Motion Model

Our control inputs for SLAM algorithm are $u = [\upsilon \ \omega]^T$ where υ is the translation velocity and ω is the rotational velocity of the mobile robot. The velocity motion model is shown as,

$$\begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix} = \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \theta_{t-1} \end{bmatrix} + \begin{bmatrix} -\frac{\upsilon_{t-1}}{\omega_{t-1}} \sin\theta_{t-1} + \frac{\upsilon_{t-1}}{\omega_{t-1}} \sin(\theta_{t-1} + \omega_{t-1}\Delta t) \\ \frac{\upsilon_{t-1}}{\omega_{t-1}} \cos\theta_{t-1} - \frac{\upsilon_{t-1}}{\omega_{t-1}} \cos(\theta_{t-1} + \omega_{t-1}\Delta t) \\ \omega_{t-1}\Delta t \end{bmatrix} \tag{1}$$

where $\mathrm{x}_{t-1} = \left(x_{t-1} \ y_{t-1} \ \theta_{t-1}\right)^T$ is represented as the robot pose at time interval $t-1$.

B. Feature-based Measurement Model

In this paper, feature-based mapping is used and features are measured by a binocular vision system. We can compute the range r, the bearing ϕ of each observed landmark relative to the mobile robot's local coordinate frame, and the feature of the landmark, s, using the vision sensor. In order to simplify the feature extraction technique, color is treated as the feature of each landmark.

Hence the feature-based measurement model is shown as,

$$\begin{bmatrix} r_t^i \\ \phi_t^i \\ s_t^i \end{bmatrix} = \begin{bmatrix} \sqrt{(m_{j,x} - \hat{x}_t)^2 + (m_{j,y} - \hat{y}_t)^2} \\ \arctan(m_{j,y} - \hat{y}_t, m_{j,x} - \hat{x}_t) \\ s_j \end{bmatrix} + \begin{bmatrix} \varepsilon_{\sigma_r^2} \\ \varepsilon_{\sigma_\phi^2} \\ \varepsilon_{\sigma_s^2} \end{bmatrix} \tag{2}$$

where the location of the landmark in the global coordinate frame is denoted as ($m_{i,x}$, $m_{i,y}$) and the i^{th} observed landmark at time t corresponds to the j^{th} landmark in the map. Here ε_{σ_r} , $\varepsilon_{\sigma_\phi}$, and ε_{σ_s} are zero-mean Gaussian errors with the standard deviations σ_r , σ_ϕ , and σ_s , respectively.

Agorithm1: EKF SLAM

Initialization: $\mathrm{z}_0, \mathrm{R}_0 = get\_measurements$
$\hat{\mathrm{x}}_0, \mathrm{P}_0 = new\_map(\mathrm{z}_0, \mathrm{R}_0)$
EKF Loop: for $k = 1$ to steps do
$\mathrm{x}_{\mathrm{R}_k}^{\mathrm{R}_{k-1}}, \mathrm{Q}_k = get\_odometry$
prediction: $\hat{\mathrm{x}}_{k|k-1}, \mathrm{F}_k, \mathrm{G}_k = f(\hat{\mathrm{x}}_{k-1}, \hat{\mathrm{x}}_{\mathrm{R}_k}^{\mathrm{R}_{k-1}}) + w_k$
$\mathrm{P}_{k|k-1} = \mathrm{F}_k \mathrm{P}_{k-1} \mathrm{F}_k^T + \mathrm{G}_k \mathrm{Q}_k \mathrm{G}_k^T$

$\mathrm{z}_k, \mathrm{R}_k = get\_measurements$
$H_k, \mathrm{H}_k, \mathrm{z}_{H_k}, \mathrm{R}_{H_k} = data\_asso-$
$ciation(\hat{\mathrm{x}}_{k|k-1}, \mathrm{P}_{k|k-1}, \mathrm{z}_k, \mathrm{R}_k)$
$\tilde{y}_k = z_k - h(\hat{\mathrm{x}}_{k|k-1})$
$\mathrm{S}_k = \mathrm{H}_k \mathrm{P}_{k|k-1} \mathrm{H}_k^T + \mathrm{R}_k$
$\mathrm{K}_k = \mathrm{P}_{k|k-1} \mathrm{H}_k^T \mathrm{S}_k^{-1}$
Correction: $\hat{\mathrm{x}}_{k|k} = \hat{\mathrm{x}}_{k|k-1}, +\mathrm{K}_k \tilde{y}_k$
$\mathrm{P}_{k|k} = (I - \mathrm{K}_k \mathrm{H}_k) \mathrm{P}_{k|k-1}$
End loop: end for
return m=($\hat{\mathrm{x}}_{k|k}$, $\mathrm{P}_{k|k}$)

C. Standard EKF SLAM Algorithm

In robotics, standard EKF SLAM is a traditional technique where only one global map is used. It will not only be used in this paper for comparing with the proposed FMJ algorithm, but also a core algorithm inside each local map (see Algorithm 1[14]).

The matrices F_k, G_k in Algorithm 1 are estimated by a First-Estimate Jacobian EKF to improve the estimation's consistency [15]. The computational cost per step of standard EKF SLAM is quadratic on the size of the map, i.e., $O(n^2)$. The total cost of standard EKF SLAM is cubic on the size of the map, i.e., $O(n^3)$.

III. FAST MAP JOINING ALGORITHM

A. Fast Map Joining Algorithm

In this section, we will introduce the new FMJ SLAM algorithm in details, Fig. 1. In FMJ SLAM, the global map is divided into a few sub-maps, m_k^p , k=1,2,…M, p=0,1,2,…M-1. M is the total number of the sub-maps. The subscript k represents the order of the sub-map. The superscript p means the k^{th} map is defined with respect to the posture of the robot at the end of the p^{th} map. The map with the superscript 0 is with respect to the inertial coordinate of the global map. In most outdoor environments, landmarks may not be distributed uniformly. In the proposed technique, the sub-map will be

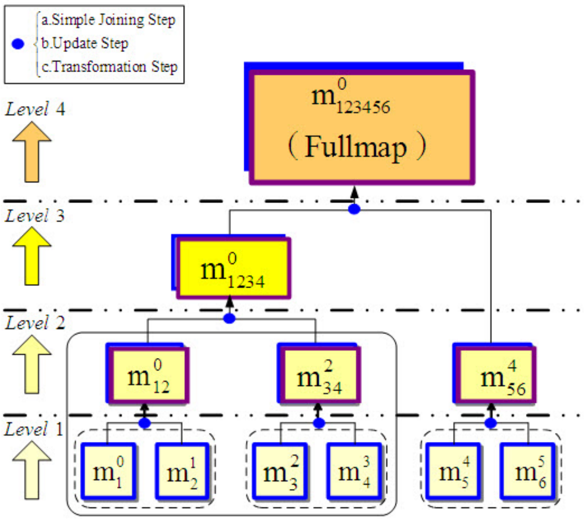


Fig.1. Architecture of FMJ SLAM.

generated until the local area with dense landmarks is observed. Thus the two neighboring sub-maps will have more common landmarks, which is critical for accurate joining operations.

As shown in Fig.1, every two locally defined sub-maps on the level 1 will carry out the joining process until all the joining process on this level are finished. Then the sub-maps on the level 2 are obtained such as m_{12}^0, m_{34}^2 and m_{56}^4. These new sub-maps on the level 2 are made with the joining operations again until a global map m_{123456}^0 is obtained finally. All these local sub-maps are joined to a global map in a hierarchical fashion.

In each sub-map, EKF SLAM is applied as the core algorithm. EKF is also applied for each joining operation. Assume the state vector in the first local map m_1^0 is estimated as

$$\hat{X}_1^0 = \begin{bmatrix} \hat{x}_r^0 & \hat{X}_{m_1,1}^0 & \hat{X}_{m_1,2}^0 & \dots & \hat{X}_{m_1,p_1}^0 \end{bmatrix}^T. \quad (3)$$

The p_k represents the number of landmarks in the k^{th} map. The corresponding covariance matrix is P_1^0. The state vector of the second local map m_2^1 is with respect to the final posture of the mobile robot at the end of the first map, which is represented as,

$$\hat{X}_2^1 = \begin{bmatrix} \hat{x}_r^1 & \hat{X}_{m_2,1}^1 & \hat{X}_{m_2,2}^1 & \dots & \hat{X}_{m_2,p_2}^1 \end{bmatrix}^T. \quad (4)$$

The corresponding covariance matrix is P_2^1. The state vector of the second map is then transformed to obtain the vector $\hat{X}_2^0$ and the corresponding P_2^0 in the reference coordinate of the first map. This vector generated by transformation is combined with the state vector of the first map to form a new state vector of this local system as

$$\hat{X}_{1-2}^0 = \begin{bmatrix} \hat{X}_1^0 & \hat{X}_2^0 \end{bmatrix}^T, \quad (5)$$

and the corresponding P_{1-2}^0. As discussed before, the first map contains the observed landmarks from 1 to p_1 while the second one includes the observed landmarks from 1 to p_2. However, there are common landmarks c_i, i=1, 2,… l_1, between two adjacent local maps. The positions of these common landmarks should be ideally the same in two local sub-maps, i.e.,

$$h = \hat{X}_{m_1,c_i}^0 - \hat{X}_{m_2,c_i}^0 = 0. \quad (6)$$

Eq. (6) can be treated as the measurement model of the joined system. Then EKF can be applied to get the estimation of the postures of the robot and the positions of all the landmarks observed in these two local maps. The detailed derivation of this joining operation can be found in [3].

According to the FMJ architecture, more common landmarks observed between two local maps have added more constraints to the measurement model during the joining operation. Hence the local map generations according to the density of the surrounding landmarks have greatly improved the precision of the SLAM. The computation complexity is analyzed in the next sub-section.

B. *Computation Complexity Analysis*

Assume that all the landmarks are distributed in more natural ways throughout the whole environment. The k^{th} local sub-map on the level 1, Fig. 1, has the size of p_k where p_k may not be equal to p_j for $k \neq j$. The computation complexity of each is then obtained as $O(p_k^3)$ because the standard EKF SLAM is used in each sub-map. These sub-maps on the level 1 are joined pair-wise to the sub-maps on the level 2. Assume that the number of the overlap landmarks between m_k and m_{k+1} is represented as l_k. Then the computation complexity of joining each pair of sub-maps on the level 1 can be expressed as,

$$C_{joining} = O(l_k (p_{2k-1} + p_{2k})^2 + l_k^2 (p_{2k-1} + p_{2k})). \quad (7)$$

In order to keep the computation complexity less than the cubic of $(p_{2k-1} + p_{2k})$, Eq. (7) can be simplified to $O((p_{2k-1} + p_{2k})^2)$ if l_k is less than

$0.62(p_{2k-1}+p_{2k})$. This condition could be satisfied in most situations. Hence the computation complexity of the global map with the size n (for instance, Fig. 1) is obtained finally as,

$$C_{FMJ}=O(\sum_{k=1}^{6}p_k^3+\sum_{k=1}^{3}(p_{2k-1}+p_{2k})^2+ \\ (\sum_{k=1}^{4}p_k-l_1-l_3)^2+(n-l_4)^2) \quad (8)$$

Because $(n-l_4)^2$ is less than n^2 and $(\sum_{k=1}^{4}p_k-l_1-l_3)^2$ is less than $(\sum_{k=1}^{4}p_k)^2$, the computation complexity of the FMJ SLAM becomes,

$$C_{FMJ}=O(\sum_{k=1}^{6}p_k^3+\sum_{k=1}^{3}(p_{2k-1}+p_{2k})^2+(\sum_{k=1}^{4}p_k)^2+n^2) \quad (9)$$

Given $p_k \ll n$ for each k, it is easy to prove that,

$$\sum_{k=1}^{6}p_k^3+\sum_{k=1}^{3}(p_{2k-1}+p_{2k})^2+\sum_{k=1}^{4}p_k^2<n^3. \quad (10)$$

Then we can deduce $C_{FMJ}=O(n^2)$. In the proposed FMJ SLAM algorithm, therefore, the increased common landmarks between two neighboring local sub-maps can also achieve fast computation.

IV. SIMULATIONS AND DISCUSSIONS

A. Procedures

The simulation block diagram of the wheeled mobile robot is shown in Fig. 2 where white noise is added as the disturbance. In this paper, the reference trajectory $p_r(t)$ is set to be an ellipse that can be represented as,

$$p_r(t)=\begin{bmatrix}x_r(t)\\ y_r(t)\\ \theta_r(t)\end{bmatrix}=\begin{cases}x_r=30\sin(0.1t)\\ y_r=20\cos(0.1t)-20\\ \theta_r=\tan^{-1}(\frac{\dot{y}_r}{\dot{x}_r})\end{cases}. \quad (11)$$

A stable tracking controller [16] is applied as,

$$u_c(t)=\begin{bmatrix}\upsilon_c(t)\\ \omega_c(t)\end{bmatrix}=\begin{bmatrix}\upsilon_r\cos\theta_e+K_x x_e\\ \omega_r+\upsilon_r(K_y y_e+K_\theta\sin\theta_e)\end{bmatrix} \quad (12)$$

where $u_r=[\upsilon_r\ \ \omega_r]^T$ is the reference input and $K_x=32$, $K_y=20$, $K_\theta=0.01$. Good tracking of the reference ellipse is therefore guaranteed. The same control input

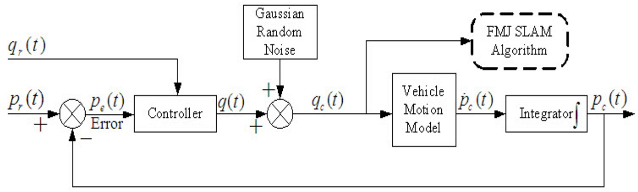


Fig. 2. The simulation block diagram..

$u_c(t)$ is used for FMJ SLAM algorithm.

Two cases are considered to analyze the performance of the mobile robot under the proposed SLAM algorithm. Case 1 is a special environment where the landmarks are distributed almost uniformly along the trajectory of the robot. Case 2 is more general surrounding environment with the different density of the landmarks in the local areas.

Consistency is one of the important criterions of a SLAM algorithm in large scale environments, especially for EKF-based SLAM [2, 7, 17, 18]. Hence chi-square test is used to check the consistency of the robot pose here. The consistency index is shown as, [3]:

$$\mathrm{CI}=\frac{D^2}{\chi^2_{r,1-\alpha}} \quad (13)$$

where $D^2=(\mathrm{x}-\hat{\mathrm{x}})^T\mathrm{P}^{-1}(\mathrm{x}-\hat{\mathrm{x}})$, $r=\dim(\mathrm{x})$ and α is the desired significance level (usually $\alpha=0.05$). It is desired for good consistency if $\mathrm{CI}\le 1$. Computation load is the other factor to evaluate the performance of the proposed SLAM algorithm especially in large scale environments.

B. Results and Discussion

1) Case 1

In Case 1, the final global map using FMJ SLAM algorithm is shown in Figure 3 where 98 landmarks are included in the whole environment. Since the landmarks

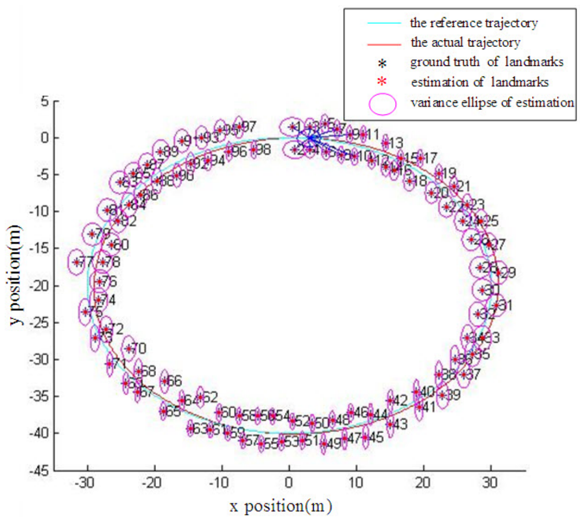
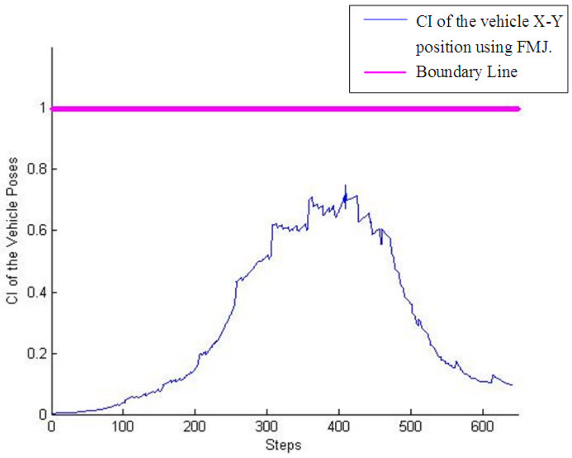
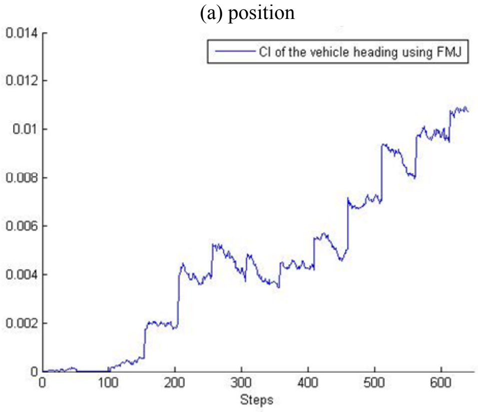


Fig. 3. The global map after carrying out FMJ SLAM in Case 1.



(a) position



(b) heading

Fig. 4. CI about posture of the mobile robot with FMJ SLAM in Case 1.

are distributed uniformly, each local sub-map has the same size with 15 landmarks including the overlap landmarks and totally 10 generated sub-maps as an example. As shown in this figure, loop closing can be realized.

The consistency index CI about the vehicle position is shown in Fig. 4 (a) using the FMJ SLAM. The horizontal axis represents time step. The index CI increases before step 400 rapidly. However, the robot revisits some initial landmarks after finishing the last local sub-map that results in lower CI. Under certain observation noisy and control noisy, CI is always lower than 1 and the final value of CI is 0.125. The index CI about the vehicle heading is shown in Fig.4 (b) using the FMJ SLAM. The results indicate that CI is lower than 0.02 throughout the process. Therefore, if we can ensure the quality of tracking control, FMJ SLAM can achieve good consistency that indicates good precision of location and mapping [18].

The comparison of total elapsed time between standard EKF SLAM and FMJ SLAM is shown in Fig.5. According to Fig.5, the total elapsed time (92.2 seconds) of the FMJ SLAM is one-fourth of the total elapsed time (368 seconds) of the standard EKF SLAM.

The next question to answer is how large scale of the environment the proposed SLAM algorithm is more

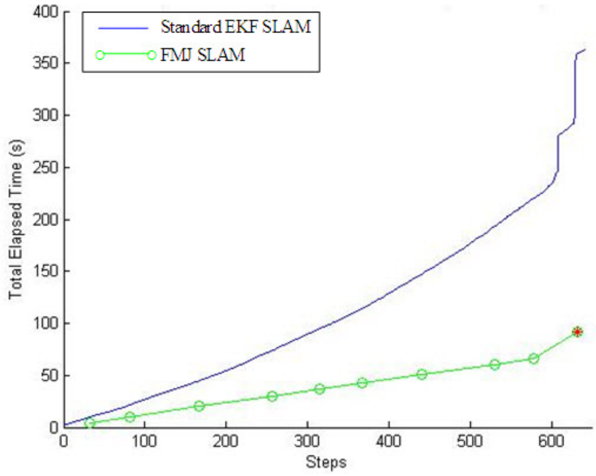


Fig. 5. The comparison of total elapsed time between the standard EKF SLAM and FMJ SLAM in Case 1.

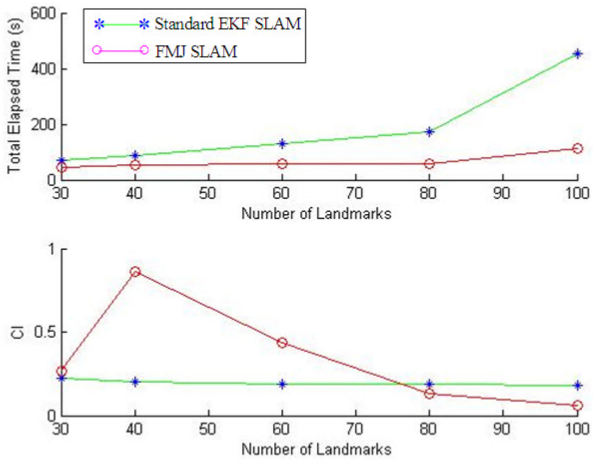


Fig. 6. The performance comparsion for different size of the global map.

suitable to apply for. The Fig.6 shows the comparison between the standard EKF SLAM and the FMJ SLAM for different scales of the environments in terms of total elapsed time and CI. It is conclude that the advantages are not obvious when the total landmarks in the environment is lower than 80. In the case of 100 landmarks in the whole environment, the total elapsed time of FMJ SLAM is one-forth of the elapsed time of the standard FMJ SLAM while the index CI of landmarks is very small.

2) Case 2

More general situation, Case 2, is discussed here where 100 landmarks are not distributed uniformly. The Fig. 7 shows the final global map with the FMJ SLAM in Case 2. Fig. 8 (a)-(d) shows the FMJ SLAM process at four different time steps. The above results indicate that the FMJ SLAM can achieve good performance. There are 100 landmarks totally in the whole environment. The total number of landmarks observed during the process is 130 including the overlap ones. The total elapsed time of the FMJ SLAM is 68.6 seconds.

In order to further illustrate the advantages of the FMJ

SLAM in Case 2, two other algorithms are introduced for comparison: the standard EKF SLAM and the modified FMJ SLAM without considering the density of the landmarks in the environment. The results in Table 1 indicate the FMJ SLAM can greatly reduce the joining time and improve the consistency index.

V. CONCLUSIONS

This paper introduces a new Fast Map Joining SLAM algorithm that can be applied for mobile robot in the large scale environment. The proposed method can be adaptive to the large-scale environment with certain density of landmark distributions. Simulations are conducted to conclude that this strategy can greatly reduce computational cost and improve the precision. The future work will focus on the real implementation for a mobile robot equipped with vision sensors.

REFERENCE

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robotics and Automation Magazine,* vol. 13, pp. 99-108, 2006.

[2] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II," *IEEE Robotics and Automation Magazine,* vol. 13, pp. 108-117, 2006.

[3] L. M. Paz, J. D. Tardos, and J. Neira, "Divide and conquer: EKF SLAM in O(n)," *IEEE Transactions on Robotics,* vol. 24, pp. 1107-1120, 2008.

[4] R. Chatila and J. Laumond, "Position referencing and consistent world modeling for mobile robots," *IEEE Robotics and Automation Magazine,* vol. 2, pp. 138-145, 1985.

[5] R. C. Smith and P. Cheeseman, "On The Representation and Estimation of Spatial Uncertainty," *International Journal of Robotics Research,* vol. 5, pp. 56-68, 1986.

[6] M. S. R. Smith and P. Cheeseman, "A Stochastic map for uncertain Spatial relationships," *Robotics Res., 4th Int,* pp. 467-474, 1988.

[7] J. Leonard and H. Durrant-Whyte, "Simultaneous map building and localization for an autonomous mobile robot," *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst., Osaka, Japan,* pp. 1442-1447, 1991.

[8] J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *IEEE Trans. Robot. Autom.,* vol. 17, pp. 242-257, May 2001.

[9] J. D. T. os, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mappingand localization in indoor environments using Sonar data," *Int. J. Robot.Res,* vol. 21, pp. 311-330, 2002.

[10] L. M. Paz and J. Neira, "Optimal localmap size for EKF-based SLAM," *in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., Beijing, China, ,* pp. 5019-5025, Oct. 2006.

[11] S. Huang, Z. Wang, G. Dissanayake, and U. Frese, "Iterated SLSJF: A Sparse Local Submap Joining Algorithm with Improved Consistency," *2008 Australasian Conference on Robotics and Automation. Canberra, Australia,* Dec.2008.

[12] S. J. Julier, "A Sparse weight kalman filter approach to simultaneous localization andmap building," *in Proc. IEEE/RSJ Int. Conf. Intell. RobotsSyst., Maui, HI,* vol. 1, pp. 1251-1256, Oct. 2001.

[13] S. Julier and J.Uhlmann, "A new extension of the kalman filter to nonlinearsystems," *in Proc. Int. Symp. Aerosp./Defense Sens., Simulate Controls,Orlando, FL, ,* pp. 182-193, 1997.

[14] http://en.wikipedia.org/wiki/EKF_SLAM.

[15] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "A First-Estimates Jacobian EKF for Improving SLAM Consistency."

[16] Y. Knayama, Y. Kimura, F. Miyazaki, and T. Noguchi, "A Stable Tracking Control Method for an Autonomous Mobile Robot," *IEEE Robotics and Automation Magazine,* pp. 384-389, 1990.

[17] J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *IEEE Trans. Robot. Autom,* vol. 17, pp. 242-257, May 2001.

[18] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the EKF-SLAM Algorithm."

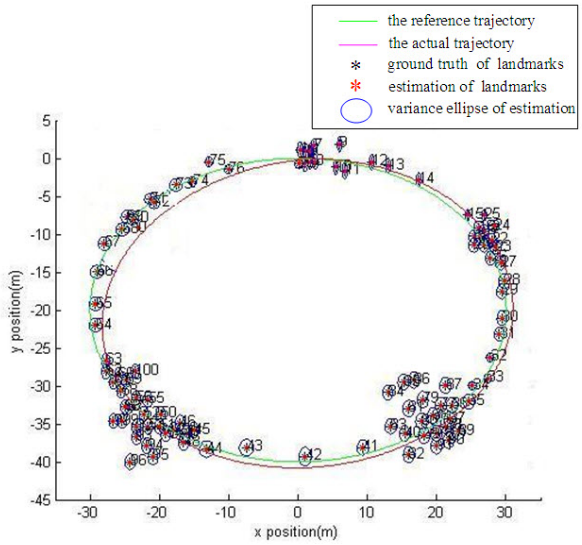
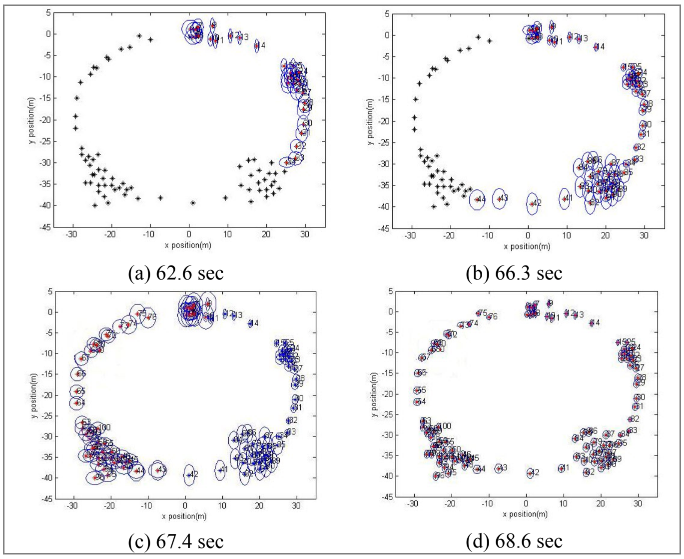


Fig. 7. The global map after FMJ SLAM in Case 2.



(a) 62.6 sec (b) 66.3 sec

(c) 67.4 sec (d) 68.6 sec

Fig. 8. Fast Map Joining process in different time intervals for Case 2.

TABLE I COMPARISON OF THREE SLAM ALGORITHMS

Algorithm	Number of local sub-maps	Joining Time(s)	Total Elapsed Time (s)	CI
Standard EKF SLAM	1 global map	None	289.1	0.1777
Modified FMJ SLAM with fixed local map size	31	29	92.2	0.5511
FMJ SLAM	7	7.1	68.6	0.0612