HybridSLAM

Macheng Shen
Department of Naval Architecture and Marine Engineering
University of Michigan
Ann Arbor, Michigan 48109
Email: macshen@umich.edu

Shengjia Wu
Department of Mechanical Engineering
University of Michigan
Ann Arbor, Michigan 48109
Email: shengjwu@umich.edu

Yukai Gong
Department of Mechanical Engineering
University of Michigan
Ann Arbor, Michigan 48109
Email: ykgong@umich.edu

Fan Bu
Department of Mechanical Engineering
University of Michigan
Ann Arbor, Michigan 48109
Email: fanbu@umich.edu

***Abstract*—In this paper, the implementation of Hybrid-SLAM (Simultaneous Localization and Mapping) is presented and investigated. Hybrid-SLAM combines the advantage of both EKF and FAST-SLAM to preserve global consistency, reduce the complexity as well as make data association more robust. It has been shown by simulation that linearization and resampling in general leads to over-confidence, which makes loop closure and data association problematic. Combining the particle representation to a Gaussian distribution and incorporating the information into an EKF back-end allows the cross correlation to be remembered over long trajectory as well as minimizing linearization error. By using a sub map approach, the complexity is also reduced compared with pure EKF-SLAM and the number of particles can be reduced compared with FAST-SLAM. In addition, data association becomes more robust as the number of matched features increases significantly.**

I. Introduction

Simultaneous Localization and Mapping (SLAM) has been a popular topic in robotics research for a long time. Several feature-based methods have been proposed to solve this problem, such as EKF, SEIF, ESEIF, which base on the Kalman Filter approach; FAST-SLAM, which bases on the exact Rao-Blackwellized Particle Filter approach and Pose-Graph SLAM, which formulates the SLAM problem as a nonlinear optimization but can only be used offline.

For online SLAM algorithms, complexity and robustness are two crucial aspects to be considered. Algorithms of high complexity may impose strict restriction on the maximum number of features to be located as well as the requirement of expensive computational amount and memory, while those of low robustness may result in inconsistent estimation or divergence, which leads to catastrophic consequences such as collision.

EKF-SLAM is regarded as the golden standard for the solution of SLAM problem as it preserves consistency over long trajectory by remembering the correlation between robot and landmarks. Besides, convergence property has been proved that the estimation of the uncertainty decreases monotonically and converges to a lower bound as the number of observations increases. [1]

However, the cubic complexity of updating the covariance matrix using observation prevents the online application of EKF-SLAM to environment of more than thousands of landmarks. This computational limitation can be overcome in two ways. One is by the development of algorithms of much lower complexity such as SEIF, ESEIF and FAST-SLAM, while all these 'pure' SLAM methods have some drawbacks with respect to consistency, linearization error and loop closing. The other way is by sub-map approaches, which creates local maps and fuses them recursively to generate a consistent global map. The latter approach allows the application of two different algorithms for the local and global maps respectively. In this paper, a hybrid sub-map based approach for the SLAM problem is presented. The key idea is to use FAST-SLAM as a front-end to generate local maps and fuses them to the EKF-SLAM back-ended global map. [2]

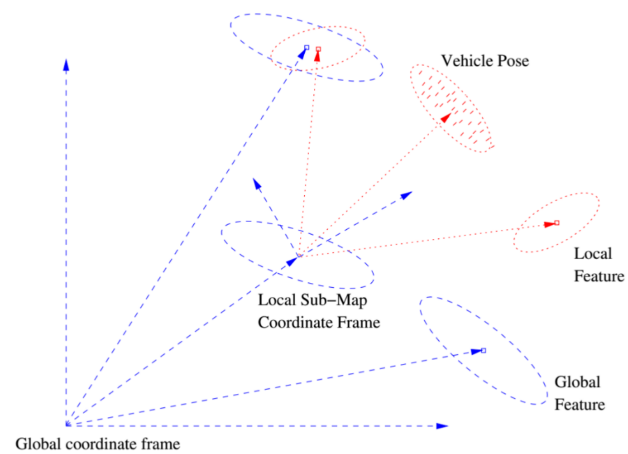


Fig. 1: Illustration of the Hybrid-SLAM based on sub map approach

Fig. 1 illustrates the idea behind Hybrid-SLAM. A local map is created with its local coordinate frame initialized by the robot pose. After several steps of FAST-SLAM on the local map, the particle representation of the state estimation is transformed to a Gaussian distribution and transformed back to the global coordinate frame. Then the corresponding features are associated with the ones in the global map and updated, and a new local map is created.

The subsequent sections are arranged as follow: In Section II, previous works are reviewed and the key contribution of this paper is summarized. In Section III, technical details of the implementation are presented. Focus has been put on the transformation of particle representation to a single Gaussian distribution, covariance propagation, map fusion and data association. In Section IV, simulation results on the simulator and the Victoria Park data set are presented. The performance of the hybrid method concerning complexity and consistency compared with EKF-SLAM and FAST-SLAM is investigated. In Section V, conclusions are drawn.

II.

A. Review of Related Previous works

The theoretical background of sub-map building, matching and fusion with the global map was presented in Tard's et.al.[3], which serves as a foundation of this work. A compressed filter approach was implemented by Guivant et.al. [4], which essentially applied the sub-map approach on a EKF front and back-end algorithm for efficient SLAM. This significantly reduced the complexity, while it was subject to linearization error and prone to wrong data association due to the EKF front-end. The idea of Hybrid SLAM using FAST-SLAM front-end and EKF-SLAM back-end was proposed by Brooks and Bailey [2], which acclaimed a superior performance over both EKF and FAST-SLAM. Nevertheless, the technical details were not presented with sufficient details.

B. Key contributions of this work

The work presented in this paper formulates the missing technical details as well as addresses important corner cases and adopt an adaptive scheme to determine the fusion time. The advantages of Hybrid SLAM can be summarized as follow:

(1) Linearization errors are eliminated by the FAST-SLAM front-end.

(2) Correlation and consistency are preserved by the EKF-SLAM back-end.

(3) Complexity is drastically reduced due to the much fewer number of Kalman update and much fewer number of particles required because of the limited size of the local map.

(4) Robustness for data association and loop closure is significantly increased due to the increased number of matched features between local and global maps.

III. Technical part

A. The whole picture of the technical solution

FastSLAM2.0 is used to get the Gaussian Mixture Model. The Gaussian Mixture Model is then converted to a Single Gaussian. Finally using this Single Gaussian Distribution to do the EkfSLAM update and thus finish a cycle of the HybridSLAM.

B. Gaussian Mixture Model from FastSLAM2.0

The FastSLAM algorithm factors distribution is:

$$\begin{aligned} p(x_{1:t}, &M|z_{1:t}, u_{1:t}, x_t) \\ &= p(x_{1:t}|z_{1:t}, u_{1:t}, x_t) \prod_n p(m_n|x_{1:t}, z_{1:t}, u_{1:t}) \end{aligned}$$

This factored distribution is represented as a set of P samples, which can be represented as:

$$S_t^p = \{w_t^p, y_t^p, P_t^p\}$$

w_t^p is the weight of the p^{th} particles, and y_t^p, P_t^p are the mean and covariance for the robot state and all landmarks.

$$y_t^p = \left[x_t^p, \mu_{1,t}^p, ..., \mu_{N,t}^p\right]$$

C. Single Gaussian from Gaussian Mixture Model

The parameters of a Single Gaussian, the mean xt and covariance Pt, can be computed from Gaussian Mixture Model by using formula (1) and (2) known as Moment Matching:

$$y_t = \sum_p w_t^p y_t^p \tag{1}$$

$$P_t = \sum_p w_t \left[P_t^p + (x_t^p - x_t)(x_t^p - x_t)^T\right] \tag{2}$$

In Equation (2), the first term in the square brackets is the covariance of the particles individual map. The second term is from the variation between particles maps.

D. Setting Correspondences from Voting Mechanism

It is necessary to set correspondences for each observation. For a single observation, each particle can make following vote decision:

(1) Correspond the observation to an existing map feature.

(2) Ignore the observation(see the observation as spurious).

(3) Correspond the observation with a new map feature.

Each particle votes in proportion to its weight. The voting mechanism considers the number of particles and their weights to determine the winner and set it as the correspondence for this single observation. In the end, this voting mechanism forms a consensus about the common set of features.

E. Forming a Gaussian Given Correspondences

Using the correspondences to map the features in each particles individual map to the common set of features. The function δ is used as the reverse mapping:$\delta_t(n, p) = i$, which indicates that the n^{th} feature in the common set is represented by the i^{th} feature in the p^{th} particles map. Thus each particle can be represented by its weight, mean and covariance. The mean and covariance matrix can be simply written as:

$$y_t^p = [x_t^p, \mu_{\delta_t(1,p),t}^p, ...\mu_{\delta_t(N,p),t}^p]$$

$$P_t^p = \begin{bmatrix} P_{xx,t}^p & & & \\ & \Sigma_{\delta_{t(1,p),t}}^p & & \\ & & ... & \\ & & & \Sigma_{\delta_{t(N,p),t}}^p \end{bmatrix}$$

Thus each particle's individual map can be represented by a single multidimensional mean and covariance.

The difficulties are the two cases. Firstly, a particles individual map may contain multiple features correspond to the same feature in the common set. Secondly, a feature in the common set may have no corresponding features in a particles individual map. For case one, the algorithm simply pick one feature as random to compute the mean and covariance for the corresponding common feature. For case two, the algorithm ignores those particles when computing the mean and covariance of that common feature.

F. Map Fusion

During the HybridSLAM process, the filter consists of two maps, the Gaussian global map and the Gaussian local map. By fusing the local map to the global map periodically, the required number of particles can be reduced compared with pure FAST-SLAM. The filter takes two steps to do the Map Fusion:

(1) Initialize the local features in the global map.

(2) Features are associated and fused.

The initialization step requires initialization of both the local features and the robot pose because the robot pose in the local map is obviously different from that in the global map during local map initialization. The mean and covariance are computing as follows:

$$y^+ = \begin{bmatrix} y^- \\ g(x^-, x_L) \end{bmatrix}$$

$$P^+ = \begin{bmatrix} P^-_{xx} & P^-_{xm} & P^{-\,T}_{xx} \bigtriangledown^T_{xg} \\ P^-_{xm} & P^-_{mm} & P^{-\,T}_{xm} \bigtriangledown^T_{xg} \\ \bigtriangledown_{xg} P^-_{xx} & \bigtriangledown_{xg} P^-_{xm} & \bigtriangledown_{xg} P^-_{xx} \bigtriangledown^T_{xg} + \bigtriangledown_{zg} P^-_L \bigtriangledown^T_{zg} \end{bmatrix}$$

Where x^-, x^+, P^-, P^+ represents the mean and covariance of the global map before and after fusion, x_L and P_L represents the mean and covariance of the local map, and $g(x^-, x_L)$ transform the local map into the global coordinates by using the Head-to-Tail [5] and x^-: the robots global pose at the time of the previous fusion.

The association step:

The condition that features E_i from the local map and F_{ji} from the global map coincide can be expressed using an ideal measurement equation which does not consider noise. This is taken as the difference of the coordinates in the global map.

$$z_i = h_{ij_i}(x) = 0$$

The measurement equation can be expanded around the mean, which is actually precise here because the measurement function is linear.

$$h_{ij_i}(x) \simeq h_{ij_i}(\hat{x}) + H_{ij_i}(x - \hat{x})$$

with

$$H_{ij_i} = \left.\frac{\partial h_{ij_i}}{\partial x}\right|_{\hat{x}} = \begin{bmatrix} 0 & \dots & H_{Fj_i} & \dots & H_{Ej_i} & \dots & 0 \end{bmatrix}$$

$$H_{Fj_i} = \left.\frac{\partial h_{ij_i}}{\partial x_{Fj_i}}\right|_{\hat{x}}$$

$$H_{Ej_i} = \left.\frac{\partial h_{ij_i}}{\partial x_{Ej_i}}\right|_{\hat{x}}$$

H represents the innovation of the pairing and H is the associated Jacobian

$$h_{\mathcal{H}}(x) = \begin{bmatrix} h_{ij_i}(x) \\ \dots \\ h_{mj_m}(x) \end{bmatrix} \simeq h_{\mathcal{H}}(\hat{x}) + H_{\mathcal{H}}(x - \hat{x})$$

$$H_{\mathcal{H}} = \partial h_{\mathcal{H}} / \partial x|_{(\hat{x})} = \begin{bmatrix} H_{1j_1} \\ \dots \\ h_{mj_m} \end{bmatrix}$$

The validity of the hypothesis $\mathcal{H}$ can be determined using an innovation test on the joint innovation $h_{\mathcal{H}}(\hat{x})$ as follows:

$$D^2_{\mathcal{H}} = h_{\mathcal{H}}(\hat{x})^T (H_{\mathcal{H}} P H_{\mathcal{H}})^{-1} h_{\mathcal{H}}(\hat{x}) < \chi^2_{d,\alpha}$$

Once this hypothesis has been determined, a new estimate $\hat{x}_0$ of the state vector and its covariance P_0 can be obtained by applying the modified EKF update equations:

$$\hat{x}' = \hat{x} - K h_{\mathcal{H}}(\hat{x})$$
$$P' = (I - K H_{\mathcal{H}}) P$$
$$K = P H^T_{\mathcal{H}} (H_{\mathcal{H}} P H^T_{\mathcal{H}})^{-1}$$

After the modified EKF update, the associated features become fully correlated with the same mean and covariance, and one of the duplicate can be eliminated.

G. Adaptive Step Number

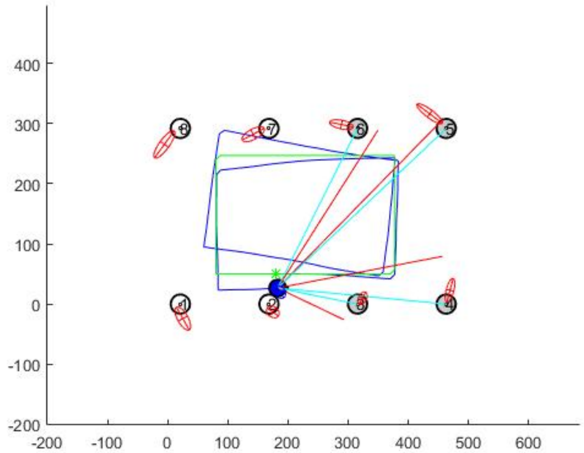
Another difficulty to get a good SLAM result is to determine when to conduct map fusion. Map fusion right after a sharp turn can be susceptible to linearization error of covariance propagation, resulting in wrong data association. adaptive number of step for local SLAM is applied to avoid fusion under large uncertainty, delaying the fusion process until sufficient number of pairings appears for robust JCBB.

IV. Simulation results and discussion

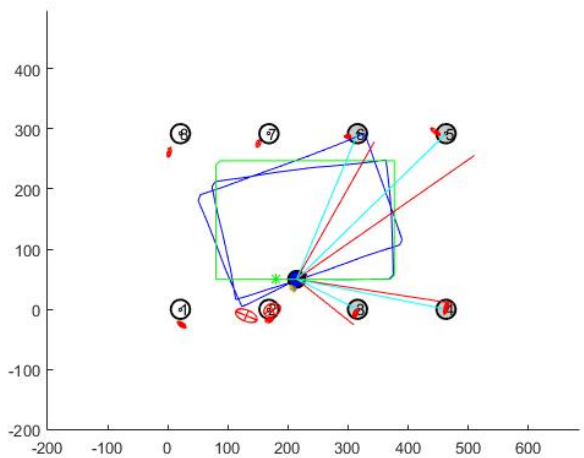
A. Consistency

One of the important criterions for robust SLAM algorithms is the capability of preserving consistency. Loss of consistency can result in failure of data association, loop closure and even more severe consequences such as collision if the results of SLAM are to be used for planning.

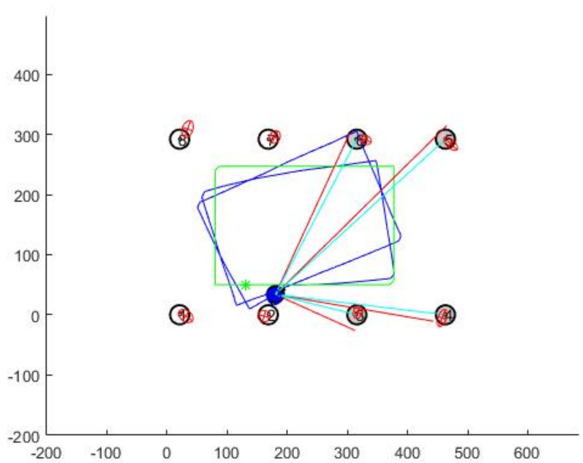
Fig. 2 shows the SLAM results by EKF, FAST and Hybrid-SLAM after 200 time steps. All of the three results have reached a steady state, i.e., the mean and covariance of the landmarks does not significantly change with time any more. The three sigma ellipsoids generated by EKF and Hybrid-SLAM are larger than those generated by FAST-SLAM, which indicates that the effect of particle degeneracy in general leads to over-confidence. This problem can be well handled by Hybrid-SLAM because the local map generated by FAST-SLAM front-end is transformed into a Gaussian representation and fused to the global map every several time steps before the particles degenerate severely. The cross correlation between the robot and landmarks are preserved by taking the variance of the estimation of landmarks by each particles into consideration.



(a) generated by EKF-SLAM



(b) generated by FAST-SLAM (50 particles)



(c) generated by Hybrid-SLAM (50 particles, fusion every 10 steps)

Fig. 2: Consistency of the resulting map

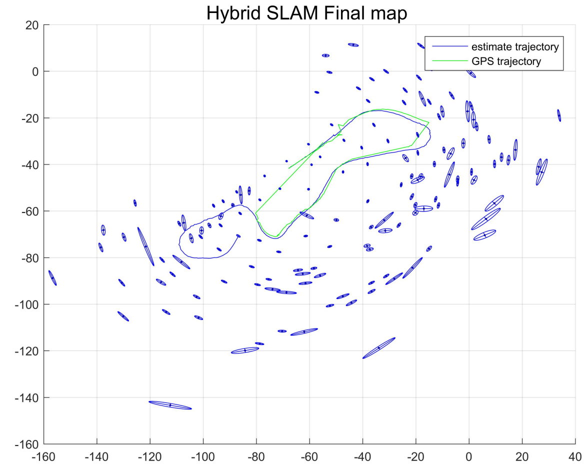


Fig. 3: Hybrid-SLAM solution to the Victoria Park data set

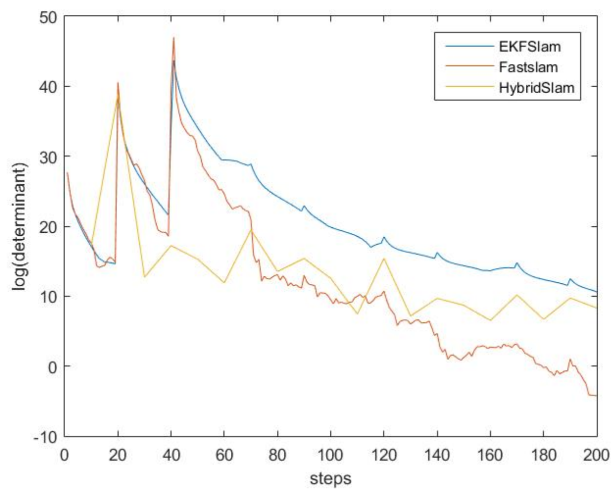


Fig. 4: Evolution of determinant of the covariance matrix using EKF-SLAM, FAST-SLAM and Hybrid-SLAM

Fig. 3 shows the SLAM solution by Hybrid-SLAM on the Victoria Park data set. It is very similar to that by EKF-SLAM, indicating nice consistency preserving property of the hybrid algorithm.

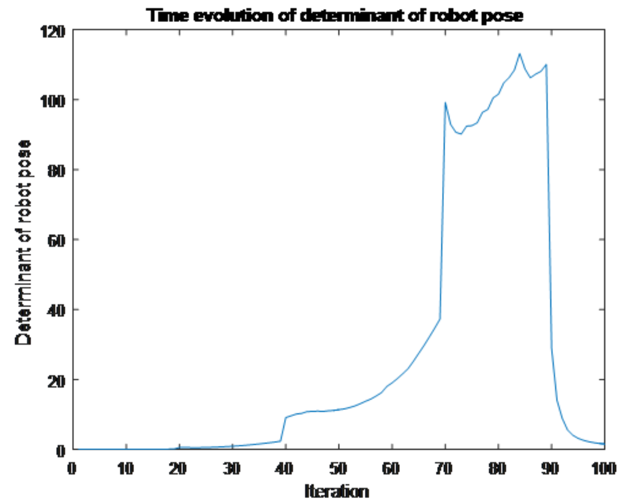
To further investigate the consistency property, time evolution of the determinants of the covariance matrix as a measurement of the uncertainty are compared as shown in Fig. 4. The sub-map is fused to the global map every 10 time steps for Hybrid-SLAM, so the determinant is shown only for these discrete time steps. The determinant of FAST-SLAM is calculated by first transforming the particle representation to a single Gaussian distribution as described in the technical part, taking into consideration the cross covariance between particles. As shown in the figure, initially FAST-SLAM can preserve consistency because particles diversity is sufficient, while after the second peak, the determinant decrease rapidly before loop closure. Finally, the determinant of FAST-SLAM is several orders smaller than those of EKF-SLAM and Hybrid-SLAM. Although Hybrid-SLAM inherits some degeneracy because of its FAST-SLAM front-end, the effect can be reduced by either increasing the particle number or reduce the local map time steps.

Moreover, Hybrid-SLAM also inherits the advantage of FAST-SLAM that no linearization is made in the prediction step. Therefore, Hybrid-SLAM outperforms EKF-SLAM where nonlinearity is significant such as a sharp turn. Also noted that the aspect ratios of the ellipsoids generated by Hybrid-SLAM are smaller than those generated by EKF-SLAM. This is because that the front-end makes no linearization to initialize the landmarks using observation, which results in a banana shaped estimation for landmarks uncertainty. While linearization error is also a source of over-confidence, the effect is much less significant than that of diversity loss. Nonetheless, large linearization error can result in large uncertainty of the current state estimation and make it prone to wrong data association.

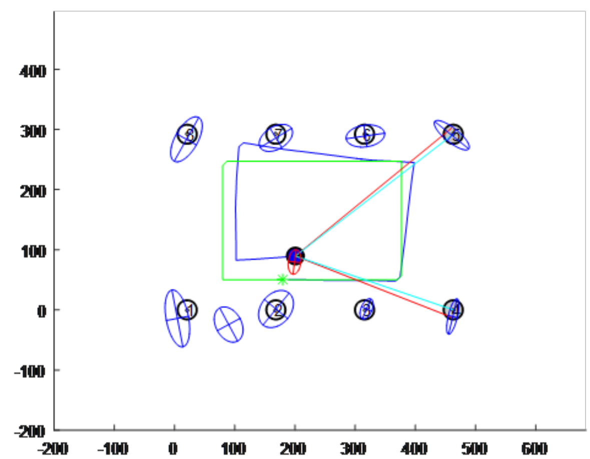
B. Data Association

Bad data association decisions can be made with high possibility when the uncertainty caused by linearization error becomes large. Wrong data association can make the state estimation deviate from the ground truth and even cause filter divergence when it happens during loop closure. This problem cannot be solved by JCBB if a spurious observation occurs without the true feature being observed. Fig.5 illustrates the typical scenario where a sharp turn results in large transient uncertainty of the robot pose and consequently wrong data association caused by inaccurate sensing.

Possible remedies can be rejecting observation if it is hard to make the data association decision or delaying the association decision and using the current observation to update future state. In either case, the state estimation process is delayed which is undesirable. Besides, it is not straightforward to update the current state using a previous observation in either EKF or FAST-SLAM frames, which forces the filter to make hard data association decisions. This problem can be solved by Hybrid-SLAM by adopting an adaptive scheme to determine the time for map fusion. When the data association becomes hard due to large uncertainty, the map fusion process is delayed and the observations are used to update the local map without any loss of information. Wrong data association in the local map can be overturned by the voting mechanism and particles that make bad data association have a large possibility of being eliminated by resampling. Map fusion is made when the uncertainty of the local robot pose and landmarks decreases below a certain level. As the size of the local map increases, the number of matched pairings between the local and global maps also increases. This also boosts the robustness of JCBB as the possibility of incorporating spurious observation becomes exponentially small with the increasing number of matched pairings.



(a) Evolution of robot pose uncertainty



(b) wrong data association caused by spurious observation

Fig. 5: Illustration of the effect of the linearization error on data association

C. Complexity

Fig. 6 shows the computational time of Hybrid-SLAM on the Victoria Park data set. The spikes appearing every 10 time steps are caused by the map fusion, which includes the transformation from particle representation, data association and the modified EKF update. The height of the spikes increases over time as the associated complexity increases with the number of landmarks in the global map. However, except for the map fusion, the average complexity does not increase with time as shown in the figure. This is because the size of the local map is limited which mostly depends on the time steps between which two successive map fusion process takes place. A small number of time steps between successive map fusion allows a choice of fewer number of particles in the local map since the size of the local map will be small and degeneracy will not be a problem; while a large number of

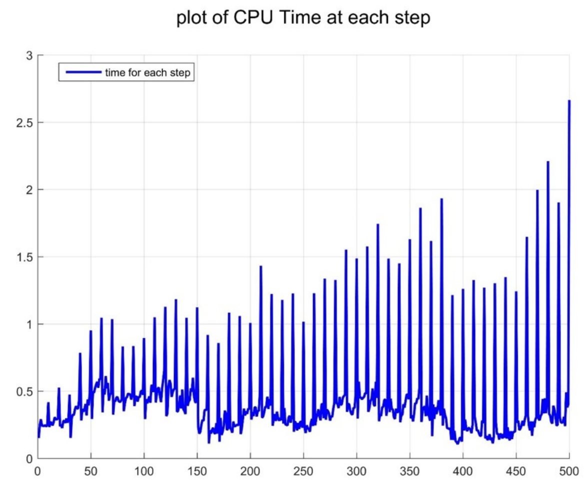
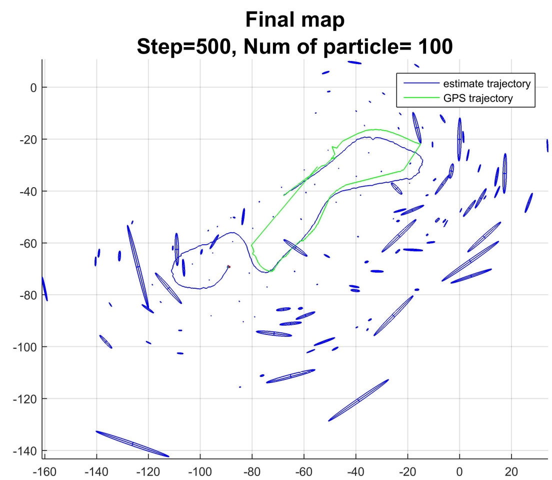
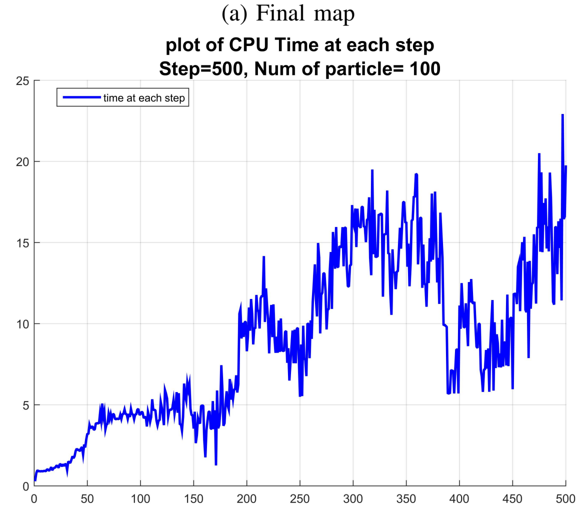


Fig. 6: Complexity of Hybrid-SLAM



(a) Final map



(b) Complexity of Fast-SLAM

Fig. 7: Fast-SLAM solution to the Victoria Park data set

time steps between successive map fusion can significantly save the computational complexity of the map fusion since the total number of map fusion process is reduced by a factor of the number of local map time steps.

The simulation results on the Victoria Park data set and the complexity using FAST-SLAM is also shown in Fig. 7 for comparison. It should be noted that although both FAST-SLAM and Hybrid-SLAM seems to be very inefficient as shown by the needed CPU time, it is actually caused by the graphics and inevitable iteration loops used in the Matlab. If the codes were written in other language, e.g., Java, it would be much more efficient. Besides, the observation noise used for data association and update by the filter has been artificially amplified so that the particle filter can correctly associate observation to the known landmarks instead of creating excessive number of new landmarks.

V. Conclusion

In this paper, the performance of Hybrid-SLAM is evaluated. It has been shown that Hybrid-SLAM outperforms both EKF-SLAM and FAST-SLAM in both loop closure and data association. The reason for that is analyzed in details and the following conclusions can be made:

(1) Linearization and resampling in general leads to overconfidence. By combining the particle representation to a Gaussian distribution, the cross correlation between robust and landmarks is allowed to be remembered, which makes it easier to close a large loop.

(2) Sub map approach allows the required number of particles in the local map to be reduced and makes JCBB data association more robust as the number of matched features increases.

(3) Hybrid-SLAM also inherits to some extent overconfidence from its FAST-SLAM front-end, while the severity is much less than that of FAST-SLAM, which is expected to be addressed by either increasing the number of particles or reducing the steps on the local map before fusion.

Reference

[1] MWMG Dissanayake, Paul Newman, Steven Clark, Hugh F Durrant-Whyte, and Michael Csorba. A solution to the simultaneous localization and map building (slam) problem. *Robotics and Automation, IEEE Transactions on*, 17(3):229–241, 2001.

[2] Alex Brooks and Tim Bailey. Hybridslam: Combining fastslam and ekf-slam for reliable mapping. In *Algorithmic Foundation of Robotics VIII*, pages 647–661. Springer, 2009.

[3] Juan D Tardós, José Neira, Paul M Newman, and John J Leonard. Robust mapping and localization in indoor environments using sonar data. *The International Journal of Robotics Research*, 21(4):311–330, 2002.

[4] Jose E Guivant and Eduardo Mario Nebot. Optimization of the simultaneous localization and map-building algorithm for real-time implementation. *Robotics and Automation, IEEE Transactions on*, 17(3):242–257, 2001.

[5] Randall Smith, Matthew Self, and Peter Cheeseman. Estimating uncertain spatial relationships in robotics. In *Autonomous robot vehicles*, pages 167–193. Springer, 1990.