Mapping large scale environments using relative position information among landmarks

Shoudong Huang, Zhan Wang, Gamini Dissanayake
ARC Centre of Excellence for Autonomous Systems (CAS)
Faculty of Engineering
University of Technology, Sydney, Australia
{*sdhuang,zwang,gdissa*}*@eng.uts.edu.au*

***Abstract*— The main contribution of this paper is a new SLAM algorithm for the mapping of large scale environments by combining local maps. The local maps can be generated by traditional Extended Kalman Filter (EKF) based SLAM. Relationships between the locations of the landmarks in the local map are then extracted and used in an Extended Information Filter (EIF) to build a global map. An important feature is that the information matrix for the global map is exactly sparse, leading to significant computational advantages. This paper thus presents an algorithm that combines the advantages of both the existing local map joining SLAM algorithms, which reduces the linearization error in EKF SLAM and allows computationally demanding global map fusion to be scheduled off-line, and the Decoupled SLAM (D-SLAM) algorithm, which provides an efficient strategy for building large maps using relative location information. The effectiveness of the new algorithm is illustrated through computer simulations.**

I. Introduction

Simultaneous localization and mapping (SLAM) has been studied extensively in the past few years. In the traditional landmark based SLAM algorithm, Extended Kalman Filter (EKF) is used to estimate a state vector containing the locations of the robot and all the landmarks. The uncertainty of the estimation is maintained in an associated covariance matrix (e.g. [1]). When the number of landmarks is large, the computational cost becomes a key issue.

In the recent years, many different SLAM algorithms have been proposed to reduce the computational effort required for solving large scale SLAM problems. For example, Guivant and Nebot [2] proposed a compressed filter which can significantly reduce the computation requirements when working in local areas or with high frequency external sensors. Thrun et al. [3] developed a Sparse Extended Information Filter (SEIF) to exploit the relative sparseness of the information matrix in SLAM. Though Frese [4] provided a proof for the approximate sparseness of the information matrix, it is shown by Eustice et al. [5] that the process of sparsification proposed in [3] leads to inconsistent estimates.

Recently, we developed a decoupled SLAM algorithm, D-SLAM, [6], where SLAM is reformulated as a static estimation problem for mapping and a low dimensional dynamic estimation problem for localization. The two estimators are concurrent but separate. It was demonstrated that the information matrix associated with the map in D-SLAM is exactly sparse resulting in significant computational advantages. D-SLAM, however, does not exploit the information available through the process model and results in some information loss.

Another popular strategy used for solving large scale SLAM problems is to build submaps or local maps. One approach is to maintain a set of submaps and describe the relationship among these maps using a topology (see e.g. [7] [8] and the references therein). While promising, this approach requires more work to completely resolve the questions of when to start a new submap, how to realize that the robot has visited an old submap, and how to treat the common landmarks among local maps. An alternative approach is to first build local maps with a small scale traditional EKF-based SLAM algorithm and then combine these local maps into a large scale global map [10] [9]. Williams [10] provided the Constrained Local Submap Filter (CLSF) where the state of the local maps are first transferred into a global coordinate frame. The transformed local map state is then combined with the global map state using an estimator that enforces constraints to fuse landmarks that are present in both the local and global maps. The resulting map covariance matrix is fully correlated and thus the map fusion process is computationally demanding. CLSF, however, allows the map fusion to be scheduled off-line while the local map is being built in real-time.

In this paper, an algorithm that combines the advantages of the Constrained Local Submap Filter (CLSF) and the Decoupled SLAM (D-SLAM) is presented. It is demonstrated that the computational cost of the global map fusion step in CLSF can be substantially reduced while much of the information lost in D-SLAM as a result of ignoring the process model can be recovered by first combining a sequence of observations to build a submap. The paper is organized as follows. The idea of the new SLAM strategy is introduced in Section II. The strategy for updating a global map using relative location information is stated in Section III. The process for extracting relative information from a local map generated by the traditional EKF based SLAM algorithm is given in Section IV. In Section V, some implementation issues such as robot localization, data association and state vector recovery are discussed. Section VI addresses the computational complexity of the proposed algorithm. Section VII provides simulation and experiments results and Section VIII concludes the paper and addresses future research directions.

II. The idea

A. Combine local maps using relative information

Figure 1 illustrates the idea. The local maps are generated one by one. The coordinate system of a local map is decided by the initial robot location when starting the local map. The global map builds on the first local map (their coordinate systems are the same).

The key difference between the algorithm proposed here and the existing map joining methods (e.g. [9], [10]) is we use "relative position information" to update the "absolute map".

When the second local map is completed, the information about the relative location among the landmarks in the second local map is first extracted, and then this information is used to update the global map.

In order that the relative location information in a local map can be fused into the global map, it is necessary that there is some overlap between successive local maps. For two-dimensional mapping, the new local map should contain at least two previously seen landmarks.

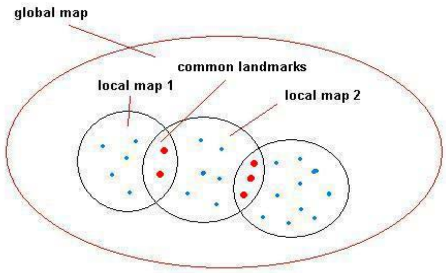


Fig. 1. The global map and local maps

B. State vectors

In this new SLAM algorithm, the final result will be one global map of all the observed landmarks. The state vector for the global map is the absolute location of all the landmarks with respect to the initial robot location. Suppose the total number of landmarks is N, then the state vector is (here the superscript 'G' stands for the global map)

$$\begin{aligned} X^G &= (X_1^G, \cdots, X_N^G) \\ &= (x_1^G, y_1^G, x_2^G, y_2^G, \cdots, x_N^G, y_N^G). \end{aligned} \tag{1}$$

The global map is built based on the information from many local maps. The state vector for each local map contains the robot location and the coordinates of all the local landmarks in the local coordinate frame. Suppose the number of landmarks in a local map is n, then the state vector is (here the superscript 'L' stands for the local map)

$$\begin{aligned} X^L &= (X_r^L, X_1^L, \cdots, X_n^L) \\ &= (x_r^L, y_r^L, \phi_r^L, x_1^L, y_1^L, \cdots, x_n^L, y_n^L). \end{aligned} \tag{2}$$

III. Global map update

When we have an estimate of the absolute locations of landmarks f_1, f_2, f_3 and we have the relative information (distances and angles) among landmarks f_2, f_3, f_4, then it is possible to obtain an initial estimate of the absolute location of the new landmark f_4 and to update the location estimate of landmarks f_1, f_2, f_3. This is the idea used in the D-SLAM mapping process [6] and the global map update in this paper.

A. Relative information relating landmark locations

For landmarks $f_1, f_2, \cdots, f_n$, the relative information are distances to f_1 and angles with respect to vector $\overrightarrow{f_1 f_2}$. These can be regarded as the polar coordinate in the coordinate system defined by origin f_1 and x-axis along $\overrightarrow{f_1 f_2}$ (see Figure 2).

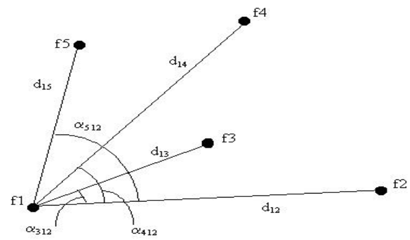


Fig. 2. relative location information – distances and angles

The relative location information can be expressed as a function of the global map state vector (with noise)

$$z_{map} = [d_{12}, \alpha_{312}, d_{13}, \cdots, \alpha_{n12}, d_{1n}]^T = H_{map}(X^G) + w_{map} \tag{3}$$

where

$$H_{map}(X^G) = \begin{pmatrix} \sqrt{(x_2^G - x_1^G)^2 + (y_2^G - y_1^G)^2} \\ atan2\left(\frac{y_3^G - y_1^G}{x_3^G - x_1^G}\right) - atan2\left(\frac{y_2^G - y_1^G}{x_2^G - x_1^G}\right) \\ \sqrt{(x_3^G - x_1^G)^2 + (y_3^G - y_1^G)^2} \\ \cdots \\ atan2\left(\frac{y_n^G - y_1^G}{x_n^G - x_1^G}\right) - atan2\left(\frac{y_2^G - y_1^G}{x_2^G - x_1^G}\right) \\ \sqrt{(x_n^G - x_1^G)^2 + (y_n^G - y_1^G)^2} \end{pmatrix} \tag{4}$$

and w_{map} is the information noise whose covariance matrix is R_{map}.

B. Update the global map

Extended Information Filter (EIF) is used to genetate the global map update. Let k denote the step when the k-th local map is fused into the global map. When $k = 1$, the global map is just the same as the local map 1 without the robot pose.

Let $i(k)$ represent information vector and $I(k)$ be the associated information matrix. The estimated state vector and the information vector are related through

$$i(k) = I(k)\hat{X}^G(k). \tag{5}$$

The procedure for using the measurement z_{map} to update the information vector and the information matrix is as follows:

After data association, the correspondence between the landmarks in the local map and the global map are identified. The new landmarks are initialized by the global location estimation of the two common landmarks f_1, f_2 and the α_{l12}, d_{1l} (for new landmark l) in z_{map}, the new estimated state vector is still denoted as $\hat{X}^G(k)$. Then the information vector $i(k)$ and the associated information matrix $I(k)$ are augmented by adding zeros at the corresponding elements of the new landmarks. We still denote them as $i(k)$ and $I(k)$.

Now z_{map} is used to update the information vector and the information matrix:

$$\begin{array}{rcl} I(k+1) & = & I(k) + \nabla H_{map}^T R_{map}^{-1} \nabla H_{map} \\ i(k+1) & = & i(k) + \nabla H_{map}^T R_{map}^{-1} [z_{map}(k+1) \\ & & -H_{map}(\hat{X}^G(k)) + \nabla H_{map} \hat{X}^G(k)] \end{array} \quad (6)$$

where ∇H_{map} is the Jacobian of the function H_{map} with respect to all the states evaluated on the current state estimation $\hat{X}^G(k)$.

C. Sparse information matrix

Since the relative distances and angles only involves a small fraction of the total landmarks in the global map (n out of N), the matrix $\nabla H_{map}^T R_{map}^{-1} \nabla H_{map}$ in (6) is sparse with the elements relating to the landmarks that are not present in the local map being exactly zero. This can be easily seen from the fact

$$\nabla H_{map} = \left[\frac{\partial H_{map}}{\partial X_1^G}, \cdots, \frac{\partial H_{map}}{\partial X_n^G}, 0, \cdots, 0 \right]. \quad (7)$$

Due to the sparse structure of ∇H_{map} in (6), the information matrix $I(k+1)$ is an exactly sparse matrix. This makes it possible to significantly reduce the computation cost of the update step by using the properties of sparse matrix (for example, see [3] or [11]).

IV. OBTAIN RELATIVE INFORMATION FROM LOCAL MAP

A. Generating a local map

To obtain the local map, any SLAM algorithm based on the Bayesian framework (e.g. EKF SLAM algorithm in [1]) can be applied. For the local maps, the initial robot location is set to be $(0, 0, 0)$. The final result of the local EKF SLAM contains the estimation of the state vector containing the final robot pose and the coordinates of local landmarks

$$\hat{X}^L = (\hat{x}_r^L, \hat{y}_r^L, \hat{\phi}_r^L, \hat{x}_1^L, \hat{y}_1^L, \cdots, \hat{x}_n^L, \hat{y}_n^L)^T \quad (8)$$

and the associated covariance matrix

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{rr}^L & \mathbf{P}_{rf}^L \\ (\mathbf{P}_{rf}^L)^T & \mathbf{P}_{ff}^L \end{bmatrix}. \quad (9)$$

B. Obtaining relative information from the local map

Suppose landmarks 1 and 2 in the local map are present in the existing global map. The relative distances and angles from other landmarks with respect to landmarks 1 and 2 can be computed by:

$$\begin{bmatrix} d_{12} \\ \alpha_{312} \\ d_{13} \\ \vdots \\ \alpha_{n12} \\ d_{1n} \end{bmatrix} = \begin{bmatrix} \sqrt{(\hat{x}_2^L - \hat{x}_1^L)^2 + (\hat{y}_2^L - \hat{y}_1^L)^2} \\ atan2\left(\frac{\hat{y}_3^L - \hat{y}_1^L}{\hat{x}_3^L - \hat{x}_1^L}\right) - atan2\left(\frac{\hat{y}_2^L - \hat{y}_1^L}{\hat{x}_2^L - \hat{x}_1^L}\right) \\ \sqrt{(\hat{x}_3^L - \hat{x}_1^L)^2 + (\hat{y}_3^L - \hat{y}_1^L)^2} \\ \vdots \\ atan2\left(\frac{\hat{y}_n^L - \hat{y}_1^L}{\hat{x}_n^L - \hat{x}_1^L}\right) - atan2\left(\frac{\hat{y}_2^L - \hat{y}_1^L}{\hat{x}_2^L - \hat{x}_1^L}\right) \\ \sqrt{(\hat{x}_n^L - \hat{x}_1^L)^2 + (\hat{y}_n^L - \hat{y}_1^L)^2} \end{bmatrix} \quad (10)$$

The physical meaning of the distances and angles is shown in Figure 2. Note the similarity between equations (10) and (4) due to the fact that all the distances and angles are independent to the coordinate system.

The corresponding covariance matrix of the noise on these relative information can be computed by the relationship between the local map and the relative information (10) and the covariance matrix of the local map – $\mathbf{P}_{ff}^L$ in (9).

V. IMPLEMENTATION ISSUES

A. Localization

In this proposed SLAM algorithm, robot location is not included in the state vector of the global map. The localization is only performed in the local EKF SLAM process where the origin of the coordinate system is the robot position at the time of starting the local map. The global robot location is not needed in the update step of the global map, though it can be easily obtained using the global map and the relative location of the robot with respect to the two previously observed landmarks in the local map.

B. Data association

There are two levels of data association in the proposed SLAM algorithm. The lower level data association is for the local EKF SLAM, and the task is to decide whether a newly observed landmark is already included in the local map or not. The upper lever data association is for the update of the global map. The task is to decide whether a landmark in the local map is already included in the global map, and if yes, find the correspondence.

The lower level data association is similar to that in the traditional EKF SLAM, and can be performed either by the Nearest Neighbor method [1] or the more robust Joint Compatibility Test [12].

It should be pointed out that in the local SLAM process, some previously observed landmarks (which are already in the global map) may be treated as new landmarks in the local map.

For the upper level data association in this algorithm, the robot location is not needed. As there are at least two common landmarks in the local map and the existing global map and the correspondence is known, data association is straightforward.

However, if this data association is performed in a naive way, the computation cost will be expensive because the size of the global map is large. Since the local map is built within a limited region, for the data association, it is not necessary to compare the local landmarks with all the landmarks in the global map. We suppose landmarks f_1^L and f_2^L in the local map are the landmarks f_i^G and f_j^G in the global map. We first compute the maximal distance from the landmarks in the local map to f_1^L — d_{max}^L. Then in the global map, we check the distance between each landmark to landmark f_i^G and only keep the landmarks whose distances is less than $d_{max}^L + \Delta$, where Δ is the maximal possible estimation error.

C. State vector recovery

Because the global map is maintained by an information vector and an information matrix, the estimation of the true landmark locations is not available all the time. For the evaluation of Jacobian in (6), as well as the upper level data association, the state vector and a submatrix of the covariance matrix need to be recovered from the information vector and the information matrix. The state vector can be recoveed by solving sparse linear equation (5). The desired columns of the covariance matrix can also be obtained by solving a constant number of sparse linear equations.

It is worth mentioning that the global map update and state vector recovery only need to be performed when a local map is fused into the global map. This is at a lower frequency compared with the frequency of the observations and could be scheduled offline [10].

D. When to start a new local map

The only condition imposed on a local map is that there are at least 2 common landmarks in any new local map and the existing global map and that the correspondence is known. Thus it is sufficient if the new local map contains at least 2 landmarks in the previous local map. This can be easily checked by the local EKF SLAM data association.

Certainly, better local maps will result in better relative position information, thus result in better (upper level) data association quality. So it is worth closing small loops to obtain a better local map when performing the local EKF SLAM.

VI. Computational Complexity

Let n be the average size of local maps, k_0 be the total number of local maps, then the size of the global map will be approximately $N = k_0 n$ (the overlap of local maps are neglected).

A. Storage

At each time, the algorithm needs to maintain one local map (including state vector and the corresponding covariance matrix) and one global map (information vector, state vector, sparse information matrix, and $O(n)$ columns of the covariance matrix). The storage requirement is $O(n^2)$ for the local map and $O(nN)$ for the global map.

B. Local map construction

For each local map, because the number of landmarks involved is n, the computation cost only depends on n. When EKF SLAM is used, the computation cost is $O(n^2)$ for each local map [2].

C. Global map update

The size of the current global map depends on how many local maps have been incorporated. Let us consider the time step when the $k+1$-th local map is required to be fused into the global map. At this time, the size of the global map is kn.

To extract relative location information from the local map, it is required to compute the covariance matrix of the relative information noise. This includes the evaluation of the Jacobians of the function in (10) and the multiplication of the Jacobians with the covariance matrix of the local map. The computation cost is $O(n^3)$.

For the global map update in (6), since the size of R_{map} is $2n-3$, the Jacobian of H_{map} is sparse and the information matrix and information vector update is a simple addition, the computation cost is $O(n^3) + O(kn)$.

For the upper level data association using the method suggested in Section V-B, it is first required to decide which landmarks in the global map are within the local map region, and this requires $O(kn)$ computation cost. Then the landmarks in the local map need to be matched with these landmarks in the local map region. The computation cost depends on the particular matching technique applied. If the nearest neighbor approach is used, the cost for the matching is $O(n^2)$. (If the method suggested in Section V-B is not used, the cost would be $O(kn^2)$).

In order to evaluate the Jacobians and do the upper level data association, the state vector and part of the covariance matrix need to be recovered. The state vector can be recovered by solving a sparse linear equation (5). The desired $O(n)$ columns of the covariance matrix can also be obtained by solving $O(n)$ sparse linear equations. Since good initial guesses are available for the linear equations (the previous estimation is a good initial guess for state vector; zero vectors are good initial guesses for the columns of covariance matrix), few iterations are enough for iterative method (for example, Preconditional Conjugate Gradient method) to converge to the solution. Thus the computation cost is $O(kn^2)$, and this is the major computational cost.

D. Total computation cost

Now we consider the total computation cost from starting the first local map to finishing the fusion of the last local map.

Since there are k_0 local maps, the total computation cost of building the local maps is $k_0 O(n^2)$.

The total computation cost for fusing the k_0 local maps into the global map is

$$\sum_{k=1}^{k_0} \left(O(n^3) + O(kn^2) \right) = O(k_0 n^3) + O(k_0^2 n^2). \qquad (11)$$

Since the cost of building local maps can be ignored as compared with the cost of local map fusion when the sizes of the local maps are small. The total cost of the whole SLAM process is also given by (11). On average, the cost of building and fusing of each local map is thus $O(n^3) + O(k_0 n^2)$.

When the size of the local map is significantly small as compared with the size of the global map, n can be regarded as a constant number and the cost for each iteration is just $O(N)$. This is the "Linear Update Cost" for an ideal SLAM solution as proposed by [13].

E. Rescheduling the computation effort

Note that the update of the global map only needs to be done at a very low frequency as compared to the observation frequency. Thus the computationally intensive fusion of the local map into the global map can be managed in such a way that it will not interfere with the update of the local maps. This is another advantages of the local map strategy [10].

VII. Simulation results

A simulation experiment with large number of landmarks was conducted to evaluate the proposed SLAM algorithm. The environment used is a 40 meter square with 196 landmarks arranged in uniformly spaced rows and columns. The robot starts from the left bottom corner of the square and follows a random trajectory, revisiting many landmarks and closing many loops. A sensor with a field of view of 180 degrees and a range of 5 meters is simulated to generate relative range and bearing measurements between the robot and the landmarks.

Figures 3(a) and 3(b) show two of the five local maps obtained by EKF SLAM. Figure 3(c) shows the global map generated by fusing all the five local maps using the proposed SLAM algorithms.

In Figure 4(a)-4(c), we compare the uncertainty of the estimates of three landmarks obtained using traditional EKF SLAM [1], D-SLAM [6] and the proposed SLAM algorithm. It can be seen the result of the proposed SLAM algorithm is slightly worse than traditional SLAM but better than D-SLAM. In most cases, we found the result to be very close to traditional SLAM. This is expected, since most of the information from robot process model is used in the proposed algorithm, resulting in very little information loss. But it is worth noticing the effect of the delay of the fusion. In Figure 4(b), the result of the proposed SLAM algorithm is worse than D-SLAM before the local map fusion at loop 1000, but after that the result is very close to traditional SLAM.

VIII. Conclusions

In this paper, a novel SLAM strategy using local maps is proposed. The idea is to use traditional EKF SLAM to construct local maps, then extract the relative information among the local landmarks and fuse the information into the global map using the D-SLAM mapping procedure.

The features of this algorithms include: (1) localization is only performed in the local SLAM, and the robot location is not included in the global map (although it can be easily calculated when needed); (2) EIF is used in the global map update and the information matrix associated with the global map is exactly sparse and only the landmarks that are within the same local map are linked through the information matrix. This results in significant saving as compared with the Constrained Local Submap Filter (CLSF) proposed by Williams [10]; (3) the information from the process model is exploited in the local SLAM, resulting in more accurate global map than that obtained using D-SLAM [6]; (4) the quality of data association (for the global map) is improved by allowing the association decisions to be deferred until an improved local picture of the environment is available; (5) the algorithm places very little restriction on how and when local maps are spawned; as the local maps are used to summerise a sequence of observations gathered by the robot, the only requirement is there are at least two previously seen landmarks in each new local map.

In fact, the proposed SLAM algorithm can be regarded as a unifying framework of SLAM algorithms which includes both EKF SLAM and D-SLAM as special cases. When there is only one local map, the algorithm is just the traditional SLAM. When the local map is constructed from a set of observations acquired at one instant of time, the algorithm proposed here is directly equivalent to D-SLAM. Moreover, the proposed global map update method can also be applied to bearing-only or range-only SLAM, as long as accurate and consistent local maps can be built.

In the proposed SLAM algorithm, as the robot location is deleted when fusing the local map into the global map, there is some information loss compared to traditional EKF SLAM. This information loss can be recovered by adding a number of robot locations (the robot starting location and the robot end location for each local maps) into the global map. This is a subject of current research. We are also in the process of conducting large scale experiments to further test the proposed SLAM algorithm.

Acknowledgment

This work is supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government. The authors would also like to thank Dr. Stefan B. Willams for helpful discussions.

References

[1] G. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. on Robotics and Automation*, vol. 17, pp. 229–241, 2001.

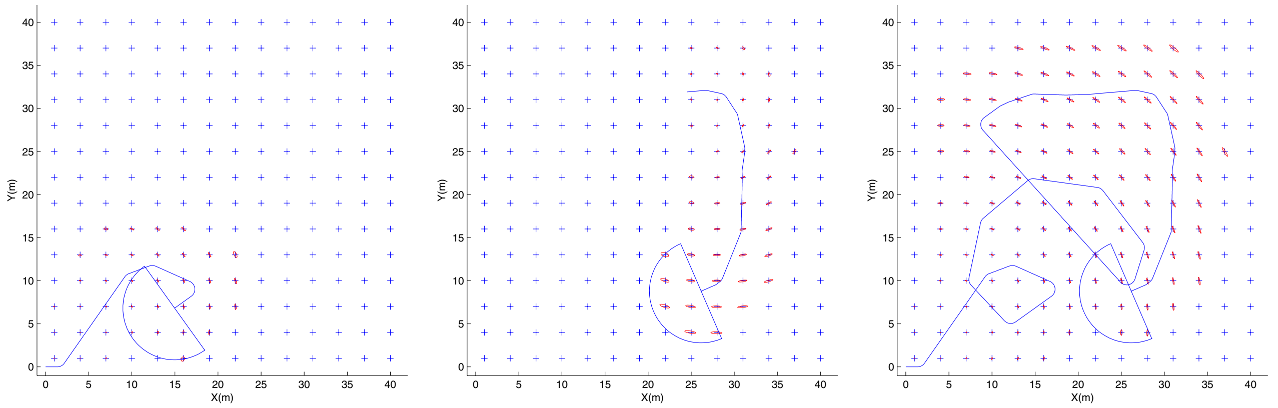
[2] J.E. Guivant, E.M. Nebot, "Optimization of the simultaneous localization and map building (SLAM) algorithm for real time implementation," *IEEE Trans. on Robotics and Automation*, vol. 17, pp. 242–257, 2001.

[3] S. Thrun, Y. Liu, D. Koller,A.Y. Ng, Z. Ghahramani, H. Durrant-Whyte, "Simultaneous Localization and Mapping with Sparse Extended Information Filters," *International J. of Robotics Research*, vol. 23, pp. 693–716, 2004.

[4] U. Frese, "A Proof for the Approximate Sparsity of SLAM Information Matrices," *In Proceedings IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, April 2005 , pp. 331–337, 2005.

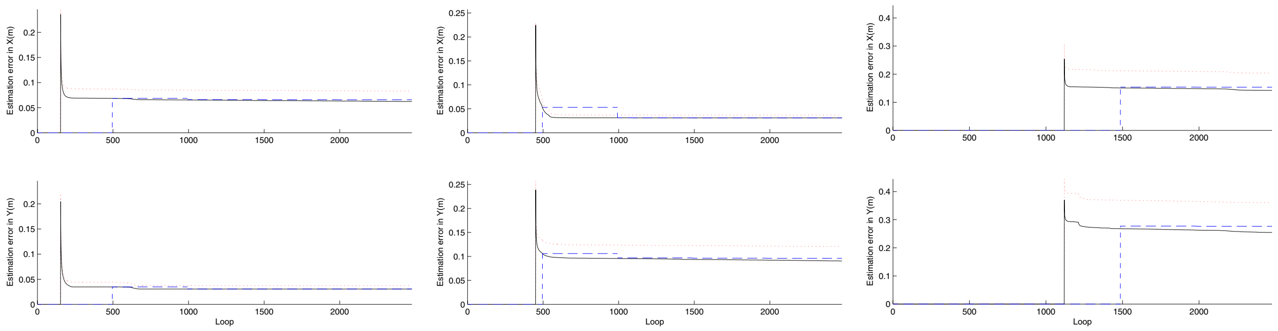
[5] R. M. Eustice, M. Walter, J. J. Leonard, "Sparse Extended Information Filters: Insights into Sparsification," *In Proceedings of 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Edmonton, Alberta, Canada, August 2-6, 2005. pp.641-648.

[6] Z. Wang, S. Huang, G. Dissanayake, D-SLAM: Decoupled Localization and Mapping for Autonomous Robot *In Proceedings of the International Symposium on Robotics Research (ISRR)*, October 2005, US. available online http://robot.cc/program.html



(a) Local Map 1 (b) Local Map 5 (c) Global Map obtained by combining the 5 local maps

Fig. 3. The 2 local maps and the global map: each local map is created by EKF SLAM, the coordinate system of the local map is decided by the robot starting position (and orientation) when starting the local map, the initial uncertainty of the robot is set to 0 in the local map. The coordinate system of the Global Map is the same as that of Local Map 1. Local Maps 1-5 are fused into the global map in sequence. Local Map 1 is fused into the global map directly, for Local Maps 2-5, the relative position information is first extracted, then the information is fused into the existing global map. It is quite flexible in when to start a local map, the only requirement is there are at least 2 common landmarks between two adjacent local maps – this guarantees that the relative position information with respect to these two common landmarks in the local map can be fused into the global map.



(a) landmark 11 (b) landmark 36 (c) landmark 71

Fig. 4. The 2σ bound for three landmarks in the global map. Solid line (black) is from tradition SLAM; dotted line (red) is from D-SLAM; dashed line (blue) is from the proposed SLAM algorithm. Landmark 11 is very near to the robot starting point and is observed from the first local map. It can be seen from Figure 4(a) that the results of the three methods are very similar. Figure 4(b) shows the effect of the fusion delay. Before the fusion of the second local map (at loop 1000) the result of the proposed SLAM algorithm is worse than D-SLAM because D-SLAM had already fused some of the observation information in the second local map but the results of the proposed SLAM only contains information from the first local map. It can be seen that after the fusion of the second local map, the estimate in the proposed SLAM algorithm becomes very close to that of tradition SLAM. The fusion delay also leads to the delay of its initialization. Figure 4(c) shows that Landmark 71, which is far away from the robot starting point, is initialized at loop 1500 (when the third local map is fused) in the proposed SLAM algorithm. But it was observed at around loop 1100 when it was initialized by the other two SLAM algorithms.

[7] M. Bosse, P. M. Newman, J. J. Leonard, and S. Teller, "SLAM in Large-scale Cyclic Environments using the Atlas Framework", *International Journal on Robotics Research*. 23(12):1113 – 1139, 2004.

[8] C. Estrada, J. Neira and J.D. Tards. Hierarchical SLAM: real-time accurate mapping of large environments. *IEEE Transactions on Robotics*, Volume 21, No 4 588- 596, 2005.

[9] J. D. Tards, J. Neira, P. M. Newman and J. J. Leonard, Robust Mapping and Localization in Indoor Environments Using Sonar Data, The International Journal of Robotics Research Vol. 21, No. 4, April 2002, pp. 311-330.

[10] S. B. Williams, Efficient Solutions to Autonomous Mapping and Navigation Problems, PhD thesis, Australian Centre of Field Robotics, University of Sydney, Sydney, 2001. available online http://www.acfr.usyd.edu.au/

[11] U. Frese, P. Larsson, T. Duckett, "A multilevel relaxation algorithm for simultaneous localization and mapping," *IEEE Transactions on Robotics*, vol. 21 (2), pp. 196- 207, 2005.

[12] J. Neira and J.D. Tardos. Data Association in Stochastic Mapping Using the Joint Compatibility Test, *IEEE Trans. Robotics and Automation, vol. 17, no. 6, pp. 890-897, Dec 2001.*

[13] U. Frese. A Discussion of Simultaneous Localization and Mapping. *Autonomous Robots* (to appear). available online http://www.informatik.uni-bremen.de/ũfrese/publications_e.html