

“© 2005 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

Decoupling Localization and Mapping in SLAM Using Compact Relative Maps

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

Abstract—In this paper, we propose a new algorithm for SLAM that makes use of a state vector consisting of quantities that describes the relative locations among features. In contrast to previous relative map strategies, the new state vector is compact and always consists of $2n - 3$ elements (in a 2-D environment) where n is the number of features in the map. It is also shown that the information from observations can be transformed and grouped into two parts: first one containing the information about the map and the second one containing the information about the robot location relative to the features in the map. Therefore the SLAM can be decoupled into two processes where mapping uses the first part of the transformed observation vector and localization becomes a 3-dimensional estimation problem. It is also shown that the information matrix of the map is exactly sparse, resulting in potential computational savings when an information filter is used for mapping. The new decoupled SLAM algorithm is called D-SLAM and is illustrated using simulation.

I. INTRODUCTION

Simultaneous Localization and Mapping (SLAM) is the process of building a map of an environment while concurrently generating an estimate for the location of the robot. In traditional SLAM (for example [1], [2]), the state vector contains both the robot pose and the feature locations and the localization and mapping are performed simultaneously.

In recent years, a few researchers have discussed the possibility of decoupling the mapping and localization processes in SLAM in order to gain computational efficiency. Since the observations made by the robot are about the relative locations between the robot and features, a natural way to decouple mapping and localization is to extract information about the relative locations among the features and then construct a relative map using this part of information. For example, Newman [3] introduced a relative map in which the map state contains the relative locations among the features, which are invariants under shift but are variants under rotation. Two filters were used in the estimation, one is the relative map filter and the other is the geometric projection filter. Csorba et al. [4], Deans and Herbert [5], and Martinelli [6] have made use of relative maps where the map state only contains distances among the features, which are invariants under shift and rotation. However, all the above approaches have redundant elements in the state vector of the relative map. Constraints need to be applied to avoid generating an inconsistent map. However, application of these constraints results in a significant increase in computational complexity of the SLAM algorithm. Moreover, issues of how to extract

the information about the relative map from the original observations has so far not been fully addressed.

In this paper, we provide a novel decoupled SLAM algorithm (D-SLAM) based on a new formulation of 2-D relative map with no redundant elements. The main contributions of this paper are as follows: (1) We show that the minimal dimension of a vector describing a relative map of n features should be $2n - 3$ and provide a natural formulation of such a vector consisting of distances and angles among the features. (2) We recast the observations in a way that the information about the relative map is separated from the other information, making it clear which part of information can be used in the mapping process. (3) We show that the information matrix in Extended Information Filter (EIF) based D-SLAM algorithm is an exactly sparse matrix, resulting in significant potential computational advantages.

The paper is organized as follows. In Section II, we briefly review the SLAM problem. Natural ways of expressing the compact 2-D relative maps are discussed in Section III. In Section IV, the observation vector is recast in to a new form and the key idea of D-SLAM is provided. Section V states the details of the mapping and localization processes in D-SLAM. Simulation results are presented in Section VI. Section VII concludes the paper and addresses future research directions.

II. SLAM PROBLEM REVIEW

The setting for the SLAM problem is that of a robot with a known dynamic model, moving through an environment containing a population of features or landmarks. The robot is equipped with a sensor that can take measurements of the relative location between any individual feature and the robot itself as shown in Figure 1(a). The SLAM problem is to estimate the position and orientation of the robot together with the locations of all the features.

The state vector in traditional SLAM is the combination of the location and orientation of the robot together with all feature locations. The observation contains information about the relative location between robot and features and is used to update both the robot pose estimate and the feature estimates. In the traditional formulation, the mapping and localization therefore need to be performed simultaneously.

Recently, different group of people have tried different ways of decoupling the mapping and localization processes in SLAM in order to significantly reduce the computation burden

and to avoid the effect of the process model error. One commonly used idea is using observation information to construct a relative map of the features [4],[5],[3],[6]. However, the map state used in the literature contains redundant elements which may cause either inconsistency or higher computation load.

In the next two sections, we will (1) provide a natural way of expressing a relative map with no redundant elements; (2) address the issue of extracting information about the map from the observations.

III. COMPACT RELATIVE MAPS

Relative maps describe the relative locations among the features while absolute maps describe the absolute locations of features. For n point features in a 2-D environment, the dimension of the absolute map with no redundant elements is $2n$. Since 3 elements are needed to decide a two dimensional coordinate system (2 for the position of origin and 1 for the orientation), the dimension of a compact relative map should be $2n - 3$.

Suppose there are 5 features f_1, f_2, \dots, f_5 . The location of feature f_i is (x_i, y_i) , $i = 1, \dots, 5$. Below are the state vectors of one absolute map and two examples of relative map.

$$\begin{bmatrix} x_1 \\ y_1 \\ \vdots \\ x_5 \\ y_5 \end{bmatrix}, \begin{bmatrix} d_{12} \\ \alpha_{312} \\ d_{13} \\ \alpha_{412} \\ d_{14} \\ \alpha_{512} \\ d_{15} \end{bmatrix}, \begin{bmatrix} d_{12} \\ \alpha_{312} \\ d_{13} \\ \alpha_{423} \\ d_{24} \\ \alpha_{534} \\ d_{35} \end{bmatrix} \quad (1)$$

The meaning of each elements in the two relative maps is explained in Figures 1(b) and 1(c). In this paper, we will use the first relative map state vector in our decoupled SLAM algorithm. This vector can be naturally explained as the polar coordinates of the features f_2, f_3, f_4, f_5 in the coordinate system decided by $\vec{f_1 f_2}$ (the origin is f_1).

As a comparison, we also list the relative map state vectors used in [3] and [6] (for 5 features) as follows:

$$\begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ \vdots \\ x_5 - x_1 \\ y_5 - y_1 \\ x_3 - x_2 \\ y_3 - y_2 \\ \vdots \\ x_5 - x_4 \\ y_5 - y_4 \end{bmatrix}, \begin{bmatrix} d_{12} \\ d_{13} \\ d_{14} \\ d_{15} \\ d_{23} \\ d_{24} \\ d_{25} \\ d_{34} \\ d_{35} \\ d_{45} \end{bmatrix} \quad (2)$$

The redundancy of these two state vectors is obvious.

IV. EXTRACTING MAP INFORMATION FROM OBSERVATIONS

Since all the observations are made from the robot, the original observations are information on the relative locations

between robot and features. In order to decouple mapping and localization in SLAM, another key step is to extract map information from the original observations. There are two important issues need to be addressed in extracting of this information: (1) we want to extract as much information as possible; (2) we need to make sure that information reuse is avoided.

We will show below that the measurement vector can be recast such that the information about the map contained in the measurements is relatively separated from the information about the robot location. This is achieved by transforming the original measurement vector into one containing distances and angles among features. We assume that the robot observes more than one feature at a time (this assumption can be relaxed but this issue is not addressed in this paper).

A. New formulation of measurements

Suppose the robot observes m features f_1, \dots, f_m at a particular time. Below are different versions of the measurement vectors.

1) *The original measurements*: The original measurements are the measured range and bearing of each observed feature

$$z_{old} = [r_1, \theta_1, \dots, r_m, \theta_m]^T \quad (3)$$

which contain zero-mean noise with covariance matrix R_{old} .

2) *The feature locations in the robot coordinate system*: We use $(\tilde{x}_i, \tilde{y}_i)$ to denote the location of the i -th feature in the robot coordinate system (the origin is the robot location when the observations are made; the x -axis is along the current robot heading). These can be easily computed from r_i, θ_i by

$$\tilde{x}_i = r_i \cos \theta_i, \quad \tilde{y}_i = r_i \sin \theta_i.$$

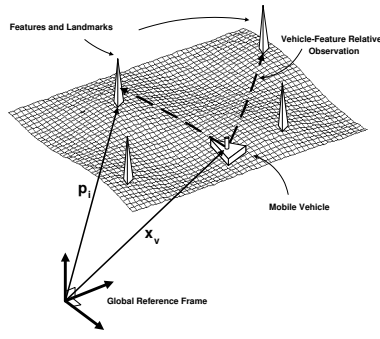
The measurement vector is, therefore

$$z_{\tilde{x}\tilde{y}} = [\tilde{x}_1, \tilde{y}_1, \dots, \tilde{x}_m, \tilde{y}_m]^T. \quad (4)$$

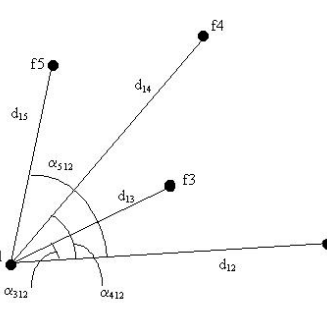
3) *Distances and angles*: The measurement vector can also be rewritten as

$$z_{new} = \begin{bmatrix} \alpha_{r12} \\ d_{1r} \\ \alpha_{\phi12} \\ \vdots \\ d_{12} \\ \alpha_{312} \\ d_{13} \\ \vdots \\ \alpha_{m12} \\ d_{1m} \end{bmatrix} = \begin{bmatrix} \text{atan2}\left(\frac{-\tilde{y}_1}{-\tilde{x}_1}\right) - \text{atan2}\left(\frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}\right) \\ \sqrt{(-\tilde{x}_1)^2 + (-\tilde{y}_1)^2} \\ -\text{atan2}\left(\frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}\right) \\ \vdots \\ \sqrt{(\tilde{x}_2 - \tilde{x}_1)^2 + (\tilde{y}_2 - \tilde{y}_1)^2} \\ \text{atan2}\left(\frac{\tilde{y}_3 - \tilde{y}_1}{\tilde{x}_3 - \tilde{x}_1}\right) - \text{atan2}\left(\frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}\right) \\ \sqrt{(\tilde{x}_3 - \tilde{x}_1)^2 + (\tilde{y}_3 - \tilde{y}_1)^2} \\ \vdots \\ \text{atan2}\left(\frac{\tilde{y}_m - \tilde{y}_1}{\tilde{x}_m - \tilde{x}_1}\right) - \text{atan2}\left(\frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}\right) \\ \sqrt{(\tilde{x}_m - \tilde{x}_1)^2 + (\tilde{y}_m - \tilde{y}_1)^2} \end{bmatrix} \quad (5)$$

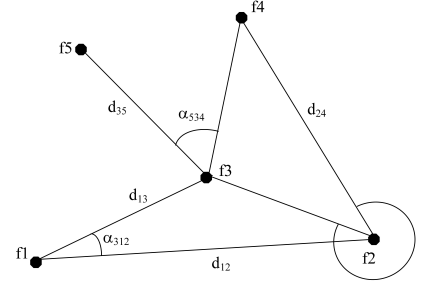
The physical meaning of the new measurement vector is shown in Figure 2(b) while that of the original measurements shown in Figure 2(a). (*In this paper the notation should be read in context. For example, d_{12} may mean the real, the estimated or the measured distance from feature 1 to feature*



(a) SLAM problem

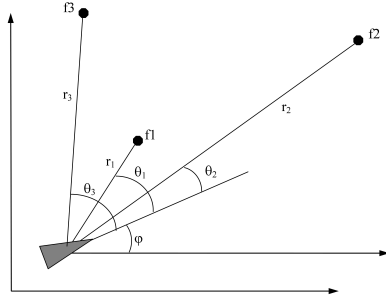


(b) Relative map version 1

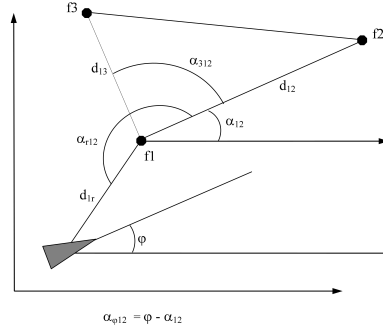


(c) Relative map version 2

Fig. 1. SLAM problem and different versions of 2-D relative maps



(a) Original measurements used in traditional SLAM



(b) New measurements used in D-SLAM

Fig. 2. Idea of D-SLAM

2. The reader should be able to tell the exact meaning from the context.)

It can now be seen that the measurement vector shown in (5) can be naturally partitioned into two vectors. The last $2m - 3$ measurements contain information about distances and angles among features that are independent of the coordinate system (see the similarity with the second vector in (1)). The first three measurements depend on the robot pose and features f_1, f_2 . This part carries information about the robot.

Therefore we denote

$$z_{rob} = \begin{bmatrix} \alpha_{r12} \\ d_{1r} \\ \alpha_{\phi12} \end{bmatrix}, \quad z_{map} = \begin{bmatrix} d_{12} \\ \alpha_{312} \\ d_{13} \\ \vdots \\ \alpha_{m12} \\ d_{1m} \end{bmatrix}. \quad (6)$$

There is a one to one correspondence between z_{old} and $z_{new} = [z_{rob}^T, z_{map}^T]^T$ and there is no dependence among the elements in z_{new} , therefore z_{old} and z_{new} contain the same information. But we note here that the two measurement

vectors z_{rob} and z_{map} are not independent, so we can not say that the information has been completely divided into two parts.

B. Measurement noise covariances

In all the Extended Kalman Filter (or Extended Information Filter) related literature, a basic assumption is that: if x is a Gaussian random (vector) variable with mean \bar{x} and variance matrix R_x , then any (vector) function of x , $g(x)$, is also Gaussian provided x is near \bar{x} . Its mean value is $g(\bar{x})$ and covariance matrix is $\nabla g R_x \nabla g^T$ where ∇g is the Jacobian of g evaluated at \bar{x} . This relationship can be used to compute the covariance matrix of the noise on the new measurements vector z_{map} .

C. The key idea of D-SLAM

In D-SLAM, the key idea is to use only z_{map} in the mapping.

Since z_{rob} and z_{map} are not independent, z_{rob} also contains some information about the map, thus the D-SLAM algorithm has some information loss. Also due to this correlation, the localization process needs to be formulated carefully in order

that the same observation information is not reused. In the next section, we describe how this issue is dealt with and state the details of the mapping and localization algorithms in D-SLAM.

V. D-SLAM ALGORITHM USING RELATIVE MAPS

In this section, we describe how we can decouple the mapping and localization using the 2-D relative map version 1 in (1) (Figure 1(b)).

A. Mapping in D-SLAM

1) *State vector*: Suppose the total number of features is n . The state vector for mapping is

$$X = (d_{12}, \alpha_{312}, d_{13}, \alpha_{412}, d_{14}, \dots, \alpha_{n12}, d_{1n})^T. \quad (7)$$

As explained in Section III, this vector expresses the polar coordinates of the features f_2, \dots, f_n in the coordinate system decided by f_1, f_2 .

Since all the features are assumed to be stationary, the mapping problem is a non-linear static estimation problem. We use the Extended Information Filter (EIF) to compute the map (for details about EIF equations, refer to [13] or [2]). At time step k (before the observation), the estimation of the state vector is $\hat{X}(k)$ and the information matrix is $I(k)(I(0) = 0)$. The relation between information vector and the estimated state vector is

$$i(k) = I(k)\hat{X}(k). \quad (8)$$

So initially, $i(0) = 0$.

2) *Measurement model*: Suppose at time k , the robot observes m features $f_{k_1}, f_{k_2}, \dots, f_{k_m}$. The model of the new measurement for mapping is

$$\begin{aligned} z_{map}(k) &= [d_{k_1 k_2}, \alpha_{k_3 k_1 k_2}, d_{k_1 k_3}, \dots, \alpha_{k_m k_1 k_2}, d_{k_1 k_m}]^T \\ &= H_{map}(X) + w_{map} \end{aligned} \quad (9)$$

where $H_{map}(X)$ can be obtained by the relationship between $[d_{k_1 k_2}, \alpha_{k_3 k_1 k_2}, d_{k_1 k_3}, \dots, \alpha_{k_m k_1 k_2}, d_{k_1 k_m}]^T$ and $[\alpha_{k_1 12}, d_{1k_1}, \dots, \alpha_{k_m 12}, d_{1k_m}]^T$. w_{map} is the new measurement noise whose covariance matrix R_{map} can be computed according to Section IV-B.

3) *Initialization of new features and the update of old features*: Suppose at time 0, the robot observes features f_1, f_2, \dots, f_m , then the (recast) measurement $z_{map}(0) = [d_{12}, \alpha_{312}, d_{13}, \dots, \alpha_{m12}, d_{1m}]^T$ is used to initialize the first $2m - 3$ elements in the state vector X .

We assume that f_{k_1}, f_{k_2} are old features in the observation made at time k . For any new feature f_{k_i} , the corresponding state $\alpha_{k_i 12}, d_{1k_i}$ can be initialized by $\alpha_{k_1 12}, d_{1k_1}, \alpha_{k_2 12}, d_{1k_2}, \alpha_{k_i k_1 k_2}, d_{k_1 k_i}$, where $\alpha_{k_i k_1 k_2}, d_{k_1 k_i}$ are from the (recast) measurement $z_{map}(k)$.

The formulas for updating the information vector and the information matrix are as follows:

$$\begin{aligned} I(k+1) &= I(k) + \nabla H_{map}^T R_{map}^{-1} \nabla H_{map} \\ i(k+1) &= i(k) + \nabla H_{map}^T R_{map}^{-1} [z_{map}(k) - H_{map}(\hat{X}(k)) + \nabla H_{map} \hat{X}(k)] \end{aligned} \quad (10)$$

where H_{map} is the function in the observation model (9) and ∇H_{map} is the Jacobian of H_{map} evaluated on the current state estimation $\hat{X}(k)$.

As we can see, using the information vector, it is not necessary to compute the inverse of the high dimensional matrix $I(k+1)$ in the update step. We only need part of the current state estimation to compute ∇H_{map} and $\nabla H_{map} \hat{X}(k)$ (similar to that in [2]).

4) *Sparse information matrix*: Due to the sparse structure of ∇H_{map} , the information matrix $I(k+1)$ is an exactly sparse matrix. This makes it possible to significantly reduce the computation cost of the recovery of the current state estimation by using the properties of sparse matrix (for example, see [11] or [2] that uses amortized approximate map recovery).

In the above information filter, the initialization of new features and the update of old features are performed in the same step. This makes the filter structure more straightforward.

B. Localization in D-SLAM

We want to estimate the relative location of the robot with respect to the coordinate system with origin at f_1 and x -axis along the vector $\overrightarrow{f_1 f_2}$.

1) *State vector*: The state vector is three dimensional

$$X_r = (d_{1r}, \alpha_{r12}, \alpha_{\phi12})^T. \quad (11)$$

We have two pieces of information on the robot location. One is from the observations. Another is from the process model plus the priori knowledge of the robot location.

2) *Process Model*: The process model can be obtained by the dynamic model of the robot. It is a (nonlinear) recursion

$$X_r(k+1) = F(X_r(k), u(k), d_r) \quad (12)$$

where $u(k)$ is the control input at time k , and d_r is the Gaussian process noise with covariance Σ . The exact formula of function F depends on the type of the robot.

3) *Measurement Model*: Suppose at time k , the robot observes m old features f_{k_1}, \dots, f_{k_m} . Their corresponding state vector is denoted as $X_s = [\alpha_{k_1 12}, d_{1k_1}, \dots, \alpha_{k_m 12}, d_{1k_m}]^T$. The measurement model used for localization is

$$\begin{aligned} z_{loc}(k) &= [r_{k_1}, \theta_{k_1}, \dots, r_{k_m}, \theta_{k_m}]^T \\ &= H_{loc}(X_r, X_s) + w_{loc} \end{aligned} \quad (13)$$

where $H_{loc}(X_r, X_s)$ can be obtained from the relationship between $[r_{k_1}, \theta_{k_1}, \dots, r_{k_m}, \theta_{k_m}]^T$ and $[d_{1r}, \alpha_{r12}, \alpha_{\phi12}, \alpha_{k_1 12}, d_{1k_1}, \dots, \alpha_{k_m 12}, d_{1k_m}]^T$. w_{loc} is the new measurement noise. Its covariance matrix R_{loc} can be obtained by R_{old} (covariance matrix of noise on z_{old}).

4) *Location estimation 1*: At time k , one estimation of X_r can be obtained by (13) together with the previous estimation of the map $\hat{X}(k-1)$ and its covariance matrix $P(k-1)$.

5) *Location estimation 2*: We can obtain another estimation of X_r at time k using the process model as follows

$$\begin{aligned} \hat{X}_r(k) &= F(\hat{X}_r(k-1), u(k-1), 0), \\ P_r(k) &= \nabla F_{X_r} P_r(k-1) \nabla F_{X_r}^T + \nabla F_{d_r} \Sigma \nabla F_{d_r}^T, \end{aligned} \quad (14)$$

where $\nabla F_{X_r}, \nabla F_{d_r}$ are the Jacobians of F with respect to X_r, d_r evaluated at $(\hat{X}_r(k-1), u(k-1), 0)$, respectively.

6) *Combining the two estimation*: Close examination of the estimation process reveals that the two estimates generated above are not independent. In this scenario, Covariance Intersection (CI) [8], which facilitates combining two correlated pieces of information when the extent of correlation itself is unknown, can be used to combine the two estimates. This is the technique we used in the simulation. But in the some cases, for example in an indoor robot equipped with a laser sensor, estimate 1 itself may provide a sufficiently accurate robot location.

Note that the information about the robot location obtained from the localization process will never be used in the mapping process, so we are not reusing the same information although z_{map} in (9) and z_{loc} in (13) are not independent.

VI. RESULTS

In this section the relative map based D-SLAM algorithm is evaluated using the results from a computer simulation and the outcome is compared with traditional SLAM.

The environment used is a square with the width of 40 meters. We put 196 features arranged in uniformly spaced rows and columns. The interval between two adjacent features is 3 meters. The robot starts from the left bottom corner and follows a circular trajectory, then in the end returns to somewhere near its starting location. Robot speed is 20cm/s and turnrate is 0.2rad/s. A sensor with a field of view of 180 degrees and a range of 6 meters is simulated to generate relative range and bearing measurements between the robot and the features.

In order to make a comparison, we need to convert the relative map to an absolute map. In our relative map, all the distances are with respect to the first feature and all the angles are with respect to the direction decided by the first two features the robot observes. So we specify the absolute location of the first feature and the direction from the first feature to the second feature, then the absolute locations of all the features can be easily computed. Thus the absolute map is recovered. We also transfer the map from the traditional SLAM algorithm into the same coordinate system to make the comparison.

Figure 3(a) and Figure 3(b) show the maps obtained from the two algorithms. Figure 3(e) and Figure 3(d) demonstrate that the uncertainty of the features monotonically decrease. However, it can also be seen that the uncertainty of the feature location estimates are more conservative in D-SLAM, compared to the optimal (traditional) SLAM estimator. As stated previously, this information loss is expected. Figure 3(c) shows the sparse information matrix containing 5073 non-zero elements out of 48841 obtained by the D-SLAM algorithm. It is seen that the features that are not in close proximity do not have any links in the information matrix. If the robot closes more loops, there will be more off-diagonal non-zero elements. But this does not affect the exact sparseness.

VII. CONCLUSIONS AND FURTHER REMARKS

The main contribution of this paper is the reformulation of the simultaneous localization and mapping (SLAM) algorithm

for mobile robots such that the mapping and localization can be treated as two independent processes – D-SLAM using relative map. This new algorithm is based on (1) a novel minimal dimensional description of a relative map which contains certain amount of distances and angles among the features, and (2) a method to recast the observation vector such that the information about the relative map can be extracted.

The most significant advantage of this algorithm is that there is no prediction step for mapping, and thus the information matrix associated with the map estimation is exactly sparse and only the features that are in the close vicinity are linked through the information matrix. This has significant implications in terms of the computational complexity of the proposed D-SLAM algorithm. Some recent results have also shown that large errors in the robot orientation introduce significant errors to the Jacobian matrices used in traditional SLAM, leading to erroneous estimates [14] [15]. D-SLAM does not have the robot location in the state vector used for mapping thus may be more robust than traditional SLAM.

Although the robot location is not incorporated in the state vector used in mapping, correlations among the features are still preserved in the information matrix. It is, therefore, not surprising to see a large reduction in the uncertainty of feature locations that are far away from the initial location of the robot as the loop is closed (see Figure 3(d) around 500 second mark, the time when the robot comes back to the vicinity of its starting position).

D-SLAM, however, results in some information loss. This is because we recast the observation into two parts and do not use all the observation information in mapping and localization. Simulation results show that a key factor that influences the extent of this information loss is the ratio between process noise and observation noise. We believe that in many practical scenarios, with the availability of high frequency scanners such as the SICK laser, the information loss is not a significant drawback.

In this work, relative map is used in the D-SLAM algorithm. Recently, we have also demonstrated that absolute map can be used in D-SLAM algorithm [16]. Some further research is required to analyse the computational complexity and the data association process. Work is in progress on a detailed analysis of information loss and the implementation of the algorithm using data available in the Robotics Data Set Repository: Radish and some outdoor data collected at our centre.

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.

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, no.3, pp. 229-241, 2001.

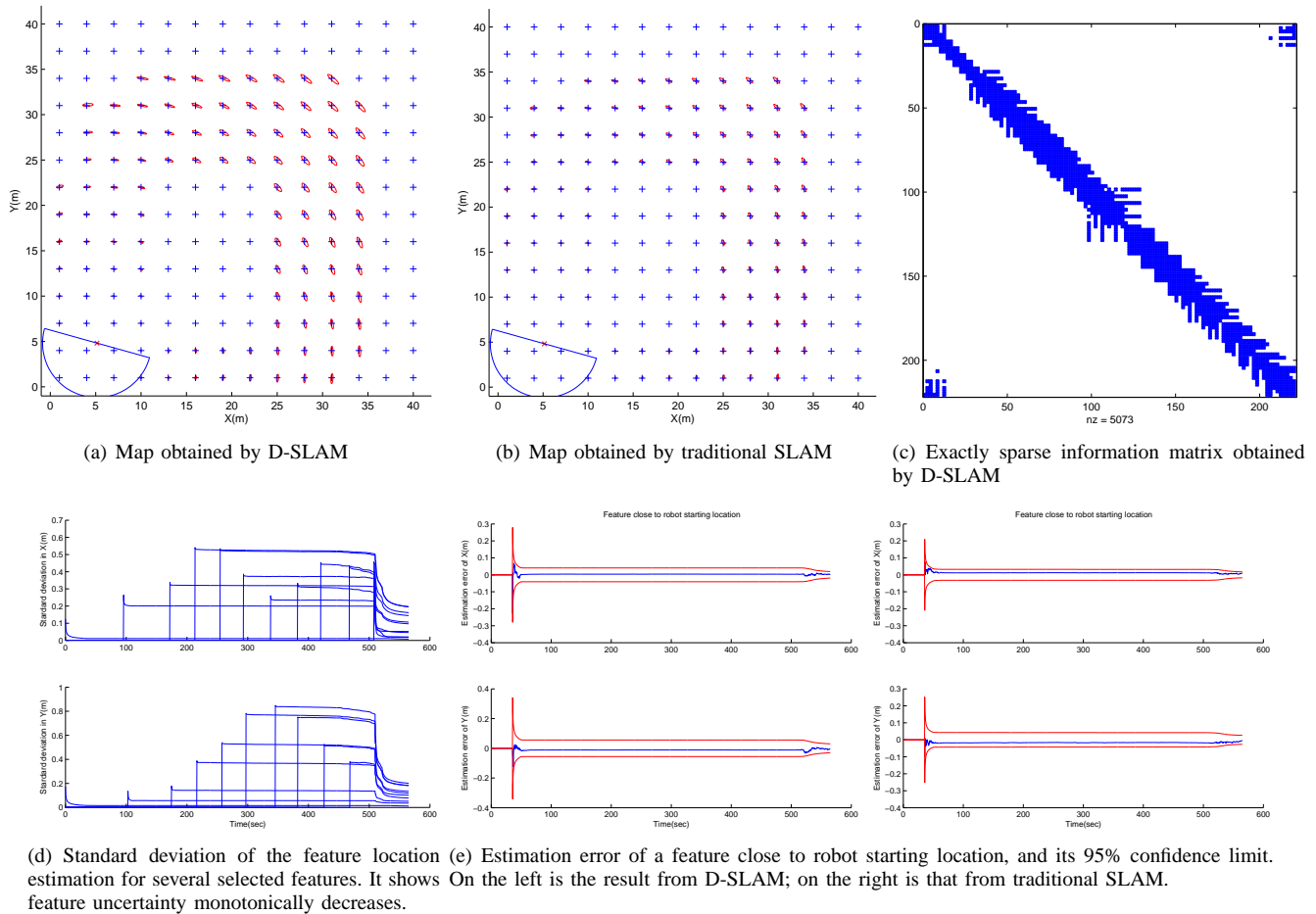


Fig. 3. Simulation results of D-SLAM and tradition SLAM

- [2] S. Thrun, Y. Liu, D. Koller, A.Y. Ng, Z. Ghahramani and H. Durrant-Whyte, "Simultaneous Localization and Mapping with Sparse Extended Information Filters", *International J. of Robotics Research*, vol. 23, no. 7-8, pp. 693-716, 2004.
- [3] P. Newman, On the Structure and Solution of the Simultaneous Localization and Map Building Problem. PhD thesis, Australian Centre of Field Robotics, University of Sydney, Sydney, Australia. 2000
- [4] M. Csorba, J. K. Uhlmann and H. F. Durrant-Whyte, "A suboptimal algorithm for automatic map building", in *Proceedings of 1997 American Control Conference*. 537-541, USA 1997.
- [5] M. C. Deans and M. Hebert, "Invariant filtering for simultaneous localization and map building", in *Proceedings IEEE International Conference on Robotics and Automation*. 1042-1047, 2000.
- [6] A. Martinelli, N. Tomatis, and R. Siegwart, "Open challenges in SLAM: An optimal solution based on shift and rotation invariants", in *Proceedings IEEE International Conference on Robotics and Automation*. 1327-1332, 2004.
- [7] J.A. Castellanos, J. Neira, J.D. Tardos, "Multisensor fusion for simultaneous localization and map building", *IEEE Trans. on Robotics and Automation*, vol. 17, no.6, pp. 908-914, 2001.
- [8] S. J. Julier and J. K. Uhlmann, "Simultaneous localization and map building using split covariance intersection", in *Proceedings of IROS 2001*, SPIE, November 2001.
- [9] J.E. Guivant and 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, no.3, pp. 242-257, 2001.
- [10] M. Bosse, P. Newman, J. Leonard, and S. Teller, "Simultaneous Localization and Map Building in Large-Scale Cyclic Environments Using the Atlas Framework". *International J. of Robotics Research*, vol. 23, no. 12, pp. 1113 - 1139, 2004
- [11] S. Pissanetzky, Sparse Matrix Technology. Academic Press, Inc. (LONDON) LTD. 1984.
- [12] Y. Bar-Shalom, X. Rong Li, T. Kirubarajan, Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software. John Wiley & Sons, Inc. 2001.(Electronic Version)
- [13] P. Maybeck, "Stochastic Models, Estimation, and Control", Vol.1, Academic, New York, 1979.
- [14] U. Frese, A Discussion of Simultaneous Localization and Mapping. *Autonomous Robots* (to appear). Available online <http://www.informatik.uni-bremen.de/~ufrese>
- [15] J.A. Castellanos, J. Neira, J.D. Tardos, "Limits to the consistency of EKF-based SLAM", *5th IFAC Symp. on Intelligent Autonomous Vehicles, IAV'04*, Lisbon, Portugal, July 2004. Available online <http://webdiis.unizar.es/~jdtardos/>
- [16] Z. Wang, S. Huang, G. Dissanayake, "Implementation Issues and Experimental Evaluation of D-SLAM", in *International Conference on Field and Service Robotics*. Accepted, To Appear. Available online <http://services.eng.uts.edu.au/~sdhuang/publications.html>