
D-SLAM: Decoupled Localization and Mapping for Autonomous Robots

Zhan Wang, Shoudong Huang, and 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

Summary. The main contribution of this paper is the reformulation of the simultaneous localization and mapping (SLAM) problem for mobile robots such that the mapping and localization can be treated as two concurrent yet separated processes: D-SLAM (decoupled SLAM). It is shown that SLAM can be decoupled into solving a non-linear static estimation problem for mapping and a low-dimensional dynamic estimation problem for localization. The mapping problem can be solved using an Extended Information Filter where the information matrix is shown to be exactly sparse. A significant saving in the computational effort can be achieved for large scale problems by exploiting the special properties of sparse matrices. An important feature of D-SLAM is that the correlation among landmarks are still kept and it is demonstrated that the uncertainty of the map landmarks monotonically decrease. The algorithm is illustrated through computer simulations and experiments.

1 Introduction

Simultaneous localization and mapping (SLAM) has been the subject of extensive research in the past few years with a number of robotics research groups contributing to make substantial progress in this area (see for example, [1], [2], [3], [4], [5], [6], [7] and the references therein). Traditionally, SLAM uses a state vector incorporating the location of the robot, all the landmarks and maintains the associated full covariance matrix. This, however, leads to a heavy computational burden when solving large scale SLAM problems.

Many researchers have exploited the special structure of the covariance matrix in order to reduce the computational effort required in SLAM. One notable result in the recent past has been that of Thrun et al. [7] which uses the Extended Information Filter to exploit the relative sparseness of the information matrix to reduce the computational effort required in SLAM. Frese [8] provided a proof for the approximate sparseness of the information matrix. However, Eustice et al. [9] demonstrated that the process of sparsification proposed in [7] leads to inconsistent estimates.

In a recent development, Eustice et al. [10] show that the inclusion of the robot trajectory in the form of past robot poses in the state vector leads to an exactly sparse information matrix. The resulting Exactly Sparse Delayed State Filter (ESDSF)

provides clear computational advantages when a view-based map representation is used. In the example presented the “map” is not represented within the state vector and is therefore not directly updated.

Another way to reduce the computation complexity is to delete the robot in the map state vector. A variety of attempts have been made to achieve this by constructing relative maps using the observation information. For example, Newman [3] introduced a relative map in which the map state contains the relative locations among the landmarks. Csorba et al. [11] and Martinelli et al. [12] have used relative maps where the map state contains distances (and angles) among the landmarks, which are invariants under shift and rotation. The structure of the covariance matrix is kept sparse by maintaining a state vector with redundant elements. As the relationships between these elements are not enforced, for large scale problems the map becomes complex and difficult to use. However, if the constraints that enforce these relationships are applied, the simple structure of the covariance matrix is destroyed, leading to an increased computational complexity [3].

This paper presents an extension of the decoupled SLAM algorithm, D-SLAM, proposed by the authors in [15] [16], where SLAM is reformulated as a static estimation problem for mapping and a three dimensional dynamic estimation problem for localization. The landmark locations are maintained using either a compact relative map [15] or an absolute Cartesian map [16]. The new formulation retains the significant advantage of being able to improve the location estimates of all the landmarks from one local observation, yet results in an exactly sparse information matrix with the number of nonzero elements related to the range of the sensor on board the robot. The main assumption in [15] [16] is that the robot can observe at least two previously seen landmarks in each observation. This paper provides a strategy to relax the above assumption by merging a set of observations to construct admissible measurements. An improved localization process based on a local SLAM is also presented.

The paper is organized as follows. The mapping and the localization algorithms in D-SLAM are stated in Sections 2 and 3, respectively. The computational complexity is addressed in Section 4. Section 5 provides simulation and experiments results of D-SLAM algorithm. Section 6 concludes the paper by providing a discussion and addressing future research directions.

2 Mapping in D-SLAM

In D-SLAM, the robot location is not included in the state vector in the mapping process. The state vector only contains the Cartesian coordinate of the locations of all the observed landmarks:

$$X = (X_1, \dots, X_n)^T = (x_1, y_1, x_2, y_2, \dots, x_n, y_n)^T. \quad (1)$$

In order to generate estimates of the landmark locations the following two processes are necessary. (1) A method of recasting the observation vector such that the information about the landmarks that is independent of the robot location can be

extracted. (2) A new landmark initialization and update method that does not require the robot location. The following sections provide details of these two processes.

2.1 Recasting the measurement vector

Suppose robot observes $m \geq 2$ landmarks f_1, \dots, f_m at a particular time where f_1, f_2 are landmarks that have been previously seen. The raw measurement and the associate Gaussian measurement noise covariance matrix are given by

$$z_{old} = [r_1, \theta_1, \dots, r_m, \theta_m]^T, \quad R_{old} = \text{diag}[R_1, R_2, \dots, R_m]. \quad (2)$$

This measurements can be recast to contain two parts as follows:

$$\begin{bmatrix} z_{rob} \\ \dots \\ z_{map} \end{bmatrix} = \begin{bmatrix} \alpha_{r12} \\ d_{1r} \\ \alpha_{\phi12} \\ \dots \\ d_{12} \\ \alpha_{312} \\ d_{13} \\ \vdots \\ \alpha_{m12} \\ d_{1m} \end{bmatrix} = \begin{bmatrix} \text{atan2}\left(\frac{-\tilde{y}_1}{-\tilde{x}_1}\right) - \text{atan2}\left(\frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}\right) \\ \sqrt{(-\tilde{x}_1)^2 + (-\tilde{y}_1)^2} \\ -\text{atan2}\left(\frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}\right) \\ \dots \\ \sqrt{(\tilde{x}_2 - \tilde{x}_1)^2 + (\tilde{y}_2 - \tilde{y}_1)^2} \\ \text{atan2}\left(\frac{\tilde{y}_3 - \tilde{y}_1}{\tilde{x}_3 - \tilde{x}_1}\right) - \text{atan2}\left(\frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}\right) \\ \sqrt{(\tilde{x}_3 - \tilde{x}_1)^2 + (\tilde{y}_3 - \tilde{y}_1)^2} \\ \vdots \\ \text{atan2}\left(\frac{\tilde{y}_m - \tilde{y}_1}{\tilde{x}_m - \tilde{x}_1}\right) - \text{atan2}\left(\frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}\right) \\ \sqrt{(\tilde{x}_m - \tilde{x}_1)^2 + (\tilde{y}_m - \tilde{y}_1)^2} \end{bmatrix} \quad (3)$$

where

$$\tilde{x}_i = r_i \cos \theta_i, \quad \tilde{y}_i = r_i \sin \theta_i, \quad i = 1, \dots, m. \quad (4)$$

The physical meaning of z_{rob} is the relative angles and distances from the robot to landmarks f_1, f_2 . The physical meaning of z_{map} is the distance between f_1 and f_2 , d_{12} , together with the relative angles and distances from the landmarks f_3, \dots, f_m to landmarks f_1, f_2 .

It is clear that z_{map} contains information about distances and angles among landmarks that are independent of the robot location and the coordinate system. The two measurement vectors z_{rob} and z_{map} are correlated and the associated measurement noise covariance matrices, R_{rob} and R_{map} respectively, are not diagonal although these matrices can be easily computed.

2.2 Mapping using information relating landmark locations

The idea for mapping in D-SLAM is the following. (i) When robot is stationary at the starting point, the raw measurement and the robot starting location are used to initialize and update the landmarks location estimates. (ii) Once the robot moves, two previously seen landmarks and the recast observation z_{map} will be used to initialize and update landmarks.

After the robot moves, the measurement model is (assume f_1, f_2 are previously seen landmarks, recall that the state vector X is given in (1))

$$z_{map} = [d_{12}, \alpha_{312}, d_{13}, \dots, \alpha_{m12}, d_{1m}]^T = H_{map}(X) + w_{map} \quad (5)$$

where $H_{map}(X)$ is given by the last $2m - 3$ formulas in equation (3) by substituting \tilde{x}_i, \tilde{y}_i with the absolute locations of the landmarks x_i, y_i ($i = 1, \dots, m$). w_{map} is the new measurement noise whose covariance matrix R_{map} can be computed by (2), (3), and (4).

The mapping problem can now be solved as a linearized minimum mean square error problem. Let $i(k)$ represent information vector and $I(k)$ be the associated information matrix. The state vector and the information vector are related through

$$i(k) = I(k)\hat{X}(k). \quad (6)$$

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

$$\begin{aligned} I(k+1) &= I(k) + \nabla H_{map}^T R_{map}^{-1} \nabla H_{map} \\ i(k+1) &= i(k) + \nabla H_{map}^T R_{map}^{-1} [z_{map}(k+1) - H_{map}(\hat{X}(k)) + \nabla H_{map} \hat{X}(k)] \end{aligned} \quad (7)$$

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

2.3 Construction of admissible measurements

To be admissible in the mapping algorithm outlined in the previous section, observation vectors need to satisfy the following condition.

Definition. *An observation made at a particular point is called admissible if it contains at least two previously seen landmarks.*

Figure 1 shows an example where robot observes two old landmarks f_1, f_2 and two new landmarks f_3, f_4 at point P_1 , but it only observes one landmark f_5 at point P_2 and one other landmark f_6 at point P_3 . Later on at point P_4 , it observes landmarks f_5, f_6, f_7 . Thus the observations at P_2 and P_3 are not admissible. It is, however, possible to combine the measurements made from different points to generate new admissible measurements as follows. Once it is detected that the observation at point P_2 is not admissible, the update to the map using the observation information from P_2 will be removed. Then a virtual observation from P_2 to f_1, f_2, f_3, f_4 will be constructed using the observation from P_1 to f_1, f_2, f_3, f_4 and an estimate of the relative motion of the robot from P_1 to P_2 (Figure 1). The uncertainty associated with this composite observation can be computed using the relevant observation equations and the process and observation uncertainties. The mapping process will proceed as if landmarks f_1, f_2, f_3, f_4, f_5 are observed from P_2 and no observation is made at P_1 . This process is repeated wherever an inadmissible observation is encountered, for example at P_3 . This strategy allows D-SLAM to function where a cluster of landmarks are separated from another cluster of landmarks by a region of "featureless" terrain.

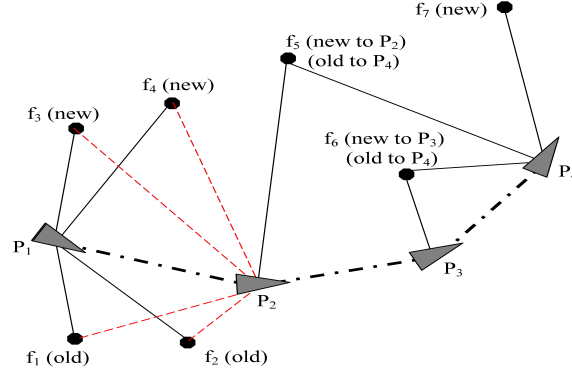


Fig. 1. Construct admissible measurement from the raw measurements

3 Localization in D-SLAM

In the localization process of D-SLAM, estimates from two approaches are combined to obtain an estimate for the robot location (and local landmark locations). One approach is to use a local traditional SLAM. The other is to use the current observation and the map generated in the previous step to solve the kidnapped robot problem. Figure 2 shows a flow-chart illustrating the localization process.

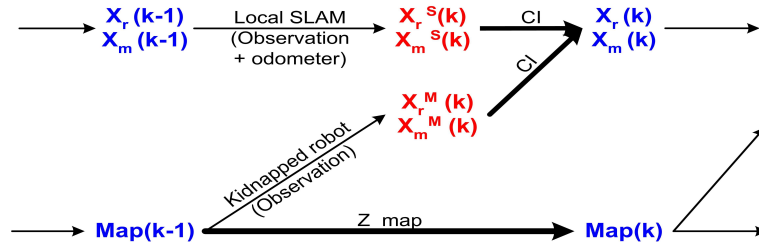


Fig. 2. Flow chat of localization and mapping process in D-SLAM

Suppose that robot observes landmarks f_1, \dots, f_m at time k , among which $f_1, \dots, f_{m_1}, m_1 \leq m$ are landmarks that have been previously seen. The state vector in D-SLAM localization contains the location of the robot and these previously seen landmarks f_1, \dots, f_{m_1} .

$$X_{loc}(k) = (X_r(k), X_1, \dots, X_{m_1})^T. \quad (8)$$

An estimate of X_1, \dots, X_{m_1} and the associated covariance matrix are available from the map obtained at time $k-1$. These together with the part of the measurement vector z_{old} that involves landmarks f_1, \dots, f_{m_1} ,

$$z_{loc} = (r_1, \theta_1, \dots, r_{m_1}, \theta_{m_1})^T = H_{loc}(X_r(k), X_1, \dots, X_{m_1}) + w_{loc}, \quad (9)$$

can be used to estimate $X_{loc}(k)$. Here H_{loc} contains $2m_1$ typical range and bearing observation functions. The estimate of $X_{loc}(k)$ is a low dimensional linearized minimum mean square error estimation problem. This approach does not make use of the process model and therefore is clearly sub-optimal.

The alternative is to use a local traditional SLAM process to estimate $X_{loc}(k)$, where only the landmarks in the vicinity of the robot are retained in the state vector. Landmarks are removed from the state vector once they are not visible from the robot. When a previously deleted landmark is re-observed, the landmark is reinitialised and is treated as a new landmark. This is effectively a SLAM-aided dead reckoning process which provides a much better robot location estimate than that obtained using dead-reckoning alone.

Which of the two estimates is more accurate depends on the prevailing circumstances. Local SLAM estimate is optimal, until the robot closes a loop by revisiting a previously traversed region of the map. The kidnapped robot solution will be superior when loop closures are present. Fusing the outcomes of the two localization processes will result in a better estimate. However, these two estimates for the robot location are correlated. Therefore, it is necessary to combine these estimates using a strategy, for example covariance intersection (CI) [14], that facilitates combining two correlated pieces of information, when the extent of correlation itself is unknown (see Figure 2).

The robot location computed is sub-optimal and is correlated to the map. These correlations do not affect the mapping process as the observation used for mapping, z_{map} , is independent of the robot location. However, as information about the robot location is not exploited in the mapping process, estimate of the map will also be suboptimal.

4 Computational complexity

A key feature of D-SLAM is that the information matrix in the mapping process is exactly sparse, and this reduces the computation cost significantly.

Since the measurement z_{map} only involves a small fraction of the total number of landmarks, the matrix $\nabla H_{map}^T R_{map}^{-1} \nabla H_{map}$ in (7) is sparse with the elements relating to the landmarks that are not present in the measurement vector being exactly zero. This can be easily seen by the fact $\nabla H_{map} = \left[\frac{\partial H_{map}}{\partial X_1}, \dots, \frac{\partial H_{map}}{\partial X_m}, 0, \dots, 0 \right]$.

In a typical sensor where the sensor range is finite, the observations only relate landmarks that are close to each other. Therefore, if landmark i and landmark j are not visible simultaneously from the robot, the measurement z_{map} will never contain both f_i and f_j . As the information matrix update involves a simple addition, the elements relating to i and j in the information matrix will remain exactly zero. Thus, in a large scale map, a significant portion of the information matrix will be exactly zero, resulting in an exactly sparse information matrix.

Let N be the size of the map. The storage requirement is $O(N)$ because the information matrix is sparse with non-zero elements $O(N)$. Localization step in D-SLAM requires updating a state vector containing only constant number of elements, thus computational cost is $O(1)$. Mapping in D-SLAM is formulated in the information form where the update step is a $O(1)$ time process and the prediction step, the computationally demanding stage of an information filter, does not exist. For data association, locations as well as the uncertainty of the landmarks in the vicinity of the robot are required. The vicinity here is defined in terms of the range of the sensor used and contains only $O(1)$ landmarks.

The major computational cost of D-SLAM is due to the need for recovering the state vector containing the landmark locations and the related parts of the covariance matrix. The state vector can be recovered by solving a sparse linear equation (6). The desired columns of the covariance matrix can also be obtained by solving a constant number of 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 solutions. Thus the computation cost for the recovery is $O(N)$. The multigrid algorithm proposed in [13] may also be an efficient method for the recovery. Overall cost of D-SLAM is, therefore, $O(N)$.

5 Evaluation of D-SLAM

5.1 Experimental Evaluation with a Pioneer robot in an office environment

The Pioneer 2 DX robot was used for the experimental implementation. This robot is equipped with a laser range finder with a field of view of 180 degrees and an angular resolution of 0.5 degree. Twelve laser reflectors were placed in a $8 \times 8m^2$ area and the Player software was used to control the robot and collect sensor data.

Matlab implementation of D-SLAM was used to process the data and compute the robot and landmark locations. Nearest neighbour algorithm was used for data association and for comparison, robot and landmark locations were also obtained using the traditional full SLAM algorithm. The results are presented in Figure 3. Although the robot localization estimates are conservative compared to traditional SLAM, the new algorithm provided a much superior estimate to that presented in [16].

5.2 Evaluation of D-SLAM in simulation with a large number of landmarks

A more complex simulation experiment with larger number of landmarks was conducted to further evaluate D-SLAM and demonstrate its properties. 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 as seen in

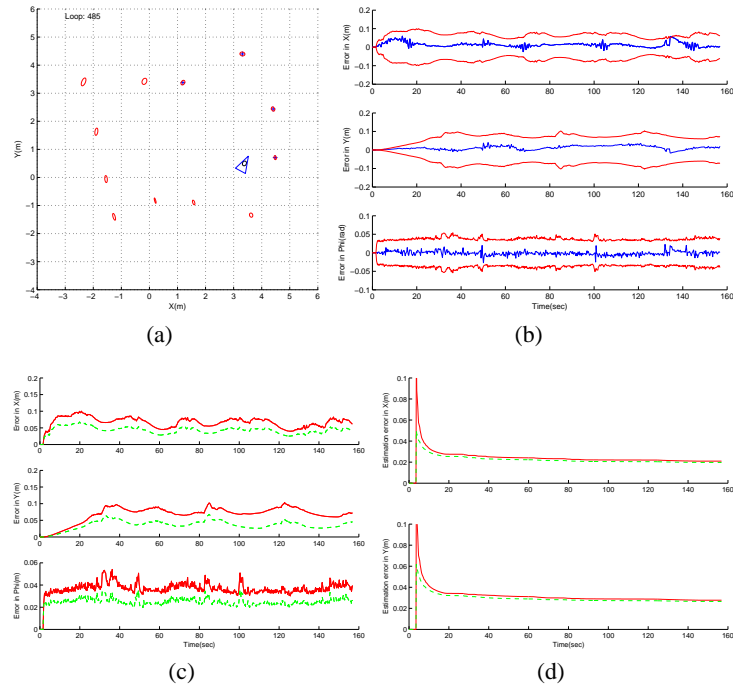


Fig. 3. D-SLAM implementation: (a) Map obtained by D-SLAM; (b) Robot location estimation error; (c) 2σ bounds of robot location estimation (solid line is from D-SLAM; dashed line is from traditional SLAM); (d) 2σ bounds of landmark 9 estimation (solid line is from D-SLAM; dashed line is from traditional SLAM).

Figure 4(a). 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.

Figure 4(b) shows the estimation error and the associated 95% confidence levels for one landmark far away from the robot initial location. It is clear that the estimates are consistent. Figure 4(c) shows all the non-zero elements of the information matrix in black after reordering. It is clear that this matrix is sparse as there are 7312 non-zero elements and 68864 exactly zero elements. The blocks diagonal terms are due to landmarks in close vicinity observed together and the off diagonal terms are due to loop closures where a previously seen landmark is re-observed some time later. Reordering the information matrix, so that indices of the nearby landmarks are adjacent, results in the banded matrix. This matrix demonstrates the fact that only the nearby landmarks are linked in the information matrix.

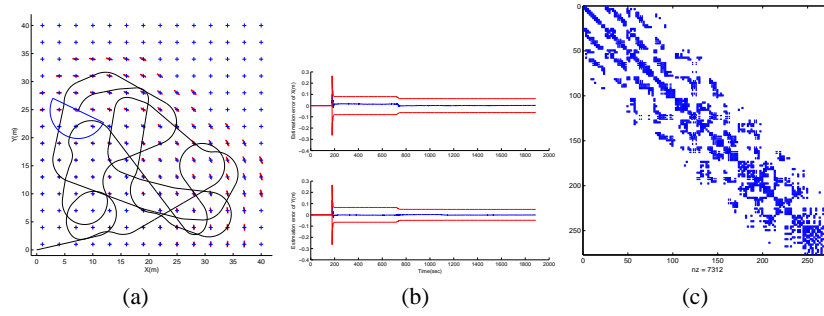


Fig. 4. D-SLAM simulations: (a) Map obtained by D-SLAM; (b) Estimation error of a landmark far away from robot starting location, and its 95% confidence limit; (c) Sparse information matrix obtained by D-SLAM after reordering (7312 non-zero elements and 68864 exactly zero elements).

6 Discussion and Conclusions

In this paper, a new decoupled SLAM algorithm: D-SLAM, is described. While the localization and mapping are performed simultaneously, mapping and localization are separated processes. The significant advantages gained are that there is no prediction step for the mapping, the information matrix associated with mapping is exactly sparse and only the landmarks that are in the close vicinity are linked through the information matrix. This results in an $O(N)$ SLAM algorithm where N is the number of landmarks.

Although the robot location is not incorporated in the state vector used in mapping, correlations between the landmarks are still preserved. Thus the location estimates of all the landmarks are improved using information from one local observation.

In D-SLAM, however, the knowledge about the robot location is not exploited in the mapping process and this results in some information loss. An analysis based on a linear one-dimensional model as well as 2-D simulations demonstrated that the information loss depends on the ratio between the sensor noise and the process noise. The smaller the ratio, the less amount of information lost. Further analytical work to quantify the extent of information loss is currently underway.

Additional work is necessary to further reduce the computation effort by exploring the possibilities of using D-SLAM in conjunction with the submap idea (e.g. [5]). Investigations in these directions together with a large scale experiment using Victoria Park data set [17] are currently in progress. Further work is required to compare D-SLAM with the recent developments in view-based SLAM [10]. In view-based SLAM the state vector consists of robot poses whereas the map is obtained through registration of successive observation sets. In D-SLAM, the map is represented in the state vector and one localization estimate is generated by registering the robot in the map. Both approaches result in significant computational advantages at the expense of some information loss. Examination of the relationship between D-SLAM with

the FastSLAM algorithm where particles are used to represent the possible robot trajectories will also be interesting.

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. Dissanayake G, Newman P, Clark S, Durrant-Whyte H, Csorba M (2001) "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. on Robotics and Automation*, vol. 17, pp. 229–241
2. Guivant JE, Nebot EM (2001) "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
3. Newman P (2000) On the Structure and Solution of the Simultaneous Localization and Map Building Problem, PhD thesis, Australian Centre of Field Robotics, University of Sydney, Sydney
4. Castellanos JA, Neira J, Tardos JD (2001) "Multisensor fusion for simultaneous localization and map building," *IEEE Trans. on Robotics and Automation*, vol. 17, pp. 908–914
5. Bosse M, Newman P, Leonard JJ, and Teller S (2004) "SLAM in Large-scale Cyclic Environments using the Atlas Framework", *International Journal on Robotics Research*, vol. 23 (12), pp. 1113–1139
6. Folkesson J, Christensen HI (2004) "Graphical SLAM - A Self-Correcting Map," *In Proceedings IEEE International Conference on Robotics and Automation (ICRA)*, LA, New Orleans, pp. 383–390
7. Thrun S, Liu Y, Koller D, Ng AY, Ghahramani Z, Durrant-Whyte H (2004) "Simultaneous Localization and Mapping with Sparse Extended Information Filters," *International J. of Robotics Research*, vol. 23, pp. 693–716
8. Frese U (2005) "A Proof for the Approximate Sparsity of SLAM Information Matrices," *In Proceedings IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, pp. 331–337
9. Eustice RM, Walter M, Leonard JJ (2005) "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, pp. 641–648
10. Eustice RM, Singh H, Leonard JJ (2005) "Exactly sparse delayed-state filters," *In Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, pp. 2428–2435
11. Csorba M, Uhlmann JK, Durrant-Whyte H (1997) "A suboptimal algorithm for automatic map building," *In Proceedings of 1997 American Control Conference*, USA, pp. 537–541
12. Martinelli A, Tomatis N, Siegwart R (2004) "Open challenges in SLAM: An optimal solution based on shift and rotation invariants," *In Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, LA, New Orleans, pp. 1327–1332
13. Frese U, Larsson P, Duckett T (2005) "A Multigrid Algorithm for Simultaneous Localization and Mapping," *IEEE Transactions on Robotics*, vol. 21 (2), pp. 1–12

14. Chen L, Arambel PO, Mehra RK (2002) "Estimation under unknown correlation: covariance intersection revisited," *IEEE Transactions on Automatic Control*, 47(11), pp. 1879–1882
15. Wang Z, Huang S, Dissanayake G (2005) "Decoupling Localization and Mapping in SLAM Using Compact Relative Maps," *In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Edmonton, Alberta, Canada, pp. 1041–1046
16. Wang Z, Huang S, Dissanayake G (2005) "Implementation Issues and Experimental Evaluation of D-SLAM," *In Proceedings of the 5th International Conference on Field and Service Robotics (FSR)*, Port Douglas, Australia. pp. 153–164
17. Nebot EM, UTE Experimental Data from Victoria Park, available online http://www.acfr.usyd.edu.au/homepages/academic/enebot/experimental_data_ute.htm