

Near Minimum Time Path Planning for Bearing-Only Localisation and Mapping*

Gu Fang

School of Engineering & Industrial Design
University of Western Sydney, Australia
g.fang@uws.edu.au

Gamini Dissanayake, N.M. Kwok, & Shoudong Huang

ARC Centre of Excellence in Autonomous Systems
Faculty of Engineering
University of Technology, Sydney, Australia
{gdissa,ngai.kwok,sdhuang}@eng.uts.edu.au

Abstract—The main contribution of this paper is an algorithm for integrating motion planning and simultaneous localisation and mapping (SLAM). Accuracy of the maps and the robot locations computed using SLAM is strongly dependent on the characteristics of the environment, for example feature density, as well as the speed and direction of motion of the robot. Appropriate control of the robot motion is particularly important in bearing-only SLAM, where the information from a moving sensor is essential. In this paper a near minimum time path planning algorithm with a finite planning horizon is proposed for bearing-only SLAM. The objective of the algorithm is to achieve a predefined mapping precision while maintaining acceptable vehicle location uncertainty in the minimum time. Simulation results have shown the effectiveness of the proposed method.

Index Terms — *Simultaneous localisation and mapping, bearing only sensing, optimisation.*

I. INTRODUCTION

Simultaneous localisation and mapping (SLAM) is a key requirement in exploration where neither the location of the robot nor the map of the environment is known. In this situation, a robot needs to perform the task of identifying its own position (localisation) as well as identifying the locations of the landmarks in the environment (mapping). SLAM has attracted significant attention in the past few years and a wide range of techniques have been reported for solving this problem [1]-[5]. In much of the literature, the robot is assumed to have sensors that are capable of measuring both the ranges and the bearings of the objects in the environment.

Although range-bearing sensors are widely available and are becoming more affordable in price, bearing-only sensors (such as cameras or radio direction finders) are attractive in many practical applications. Bearing-only sensing has been studied extensively in the past, particularly in the area of target tracking. For example, in [6] the performance of the two well-known algorithms for bearing-only target location, namely the maximum likelihood and Stansfield estimators, is compared. In [7] the discrete-time observability in bearing-only tracking is studied. More importantly, it suggested that an optimal path could be determined for the observer by using

the Fisher information matrix (FIM). Similar work is also reported in [8] where a direct numerical scheme for optimal control is used to control the observer in order to achieve the maximal information, which is defined as the determinant of the FIM. This work has also demonstrated the flexibility of the algorithm to include constraints on the observer trajectories.

The use of bearings-only sensors in SLAM has also been reported. In [9], a constrained initialisation method is introduced for bearing-only SLAM. This method delays the initialisation process until such a stage when the stored sensor information is sufficient to form a well-conditioned estimation of the target. In [10], a modified particle filter is used to perform the bearing-only SLAM. Although this method produces satisfactory results, the computational cost is high. A new initialisation technique is introduced in [11] which utilises the sequential probability ratio test (SPRT) method [12] to determine the initial positions of the objects from a small number of hypotheses. In an airborne application [13], the relative distance to landmarks is computed by measurements obtained from inertia sensors mounted on an aircraft.

To achieve bearing-only SLAM effectively and efficiently, the path that a robot follows needs to be controlled appropriately. This is because that in a bearing-only SLAM, the motion of the robot is essential to improve the quality of landmark initialization as well as the accuracy of the landmark locations estimation as the landmarks must be observed from different viewpoints. As demonstrated in [7] and [8], the paths that the observer takes have a significant effect on the quality of bearing-only observation. In SLAM applications reported in [1] and [2] robots are either following predefined paths or are manually operated. In [14], a hand-held camera is used as the sensor making measurements to landmarks where some control on the sensor path can be exercised. Alternatively, active control on the sensor [15] may be implemented instead of manoeuvring the mobile platform.

There is very little work being reported in the area of designing a robot path for the SLAM applications until recently. In [16] a path planner is used to achieve a maximum information gathering. In this work, Fisher information of the system is used as the objective function in the optimal path

* This work is partially supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government.

planning. This, however, is only based on the immediate measure of the Fisher information. Therefore it is a ‘greedy’ approach that can only achieve a local optimality.

In [17], an integrated exploration technique is proposed where the SLAM quality is included as one of the three utilities in planning exploration. In [18], a time-optimal control method is introduced for SLAM by using the dynamic programming approach. The effectiveness of the method is demonstrated by the SLAM in a corridor-like environment. The necessity of path planning in SLAM is further studied in [19] in terms of information gain/loss and a variant of nonlinear model predictive control is suggested for the multi-step look-ahead path planning of SLAM. However, the above-mentioned methods all use range and bearing sensors for SLAM.

To the best of authors’ knowledge, the path planning for SLAM with bearings-only sensors has only been reported recently in [20] (after the submission of this paper). It is concluded in [20] that maximising the information gain in this type of problems could lead to possible unstable map updates. This paper suggests an exploration policy based on Voronoi graph to enhance the stability. The issues related to the initialization of landmarks, which is a key issue in bearing-only SLAM, is not considered in the proposed path planning technique.

In this paper, a near minimum time path planning method is introduced for bearing-only SLAM. The proposed algorithm generates control signals, i.e., velocities and turn rates of the robot over a finite time horizon. These control signals are computed such that the maximum uncertainty in robot and feature locations is below prescribed limits. The constraints on the robot performance such as maximum velocity and turn rate are also considered in the planning process.

The paper is organised as follows: in section 2, following the review of the EKF based bearing-only SLAM, the near minimum time path planning method is introduced. In section 3, simulation results are provided to demonstrate the effectiveness of the proposed method. The conclusions and discussions are given in section 4.

II. TIME-OPTIMAL PATH PLANNING FOR BEARING- ONLY SLAM

A. The Bearing-Only SLAM Algorithm

This section reviews the extended Kalman filter (EKF) based SLAM algorithm introduced in [1]. In particular, SLAM in the context of bearing-only sensors in a 2-D environment is explained, in order to define the terminology and notations required for later sections.

In the SLAM problem, the state vectors that represent the robot and the i th landmark are denoted by X_v and X_{mi} . In particular, they can be expressed as:

$$X_v = [x_v, y_v, \phi_v]^T \text{ and}$$

$$X_{mi} = [x_{mi}, y_{mi}]^T \quad i = 1, 2, \dots, N$$

where x_v , and y_v are the robot’s location in Cartesian coordinates, ϕ_v is the robot heading with respect to the positive x -direction, x_{mi} , y_{mi} represents the i th landmark’s location, and N is the total number of landmarks contained in the map. Fig 1 depicts this scenario.

The discrete-time dynamic model of the robot at time instance k can be stated as:

$$X_v(k) = F_v(X_v(k-1), u(k), n(k)) \quad (1)$$

where $X_v(k)$ and $X_v(k-1)$ are the robot’s states at time k and $(k-1)$, $F_v(\cdot)$ is the robot nonlinear state transition function, $u(k)$ is the control action applied to the robot at time k , and $n(k)$ is the control noise at time k . It is assumed in this equation that the vehicle modelling noise is only introduced by the noise in the control signal.

The locations of the landmarks, X_{mi} , are assumed to be stationary. Therefore, the state equation of the landmark can be expressed as:

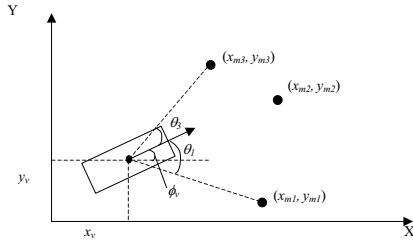


Fig. 1. The robot and the landmarks in the environment.

$$X_{mi}(k) = X_{mi}(k-1) \quad \forall k, \forall i \quad (2)$$

The system’s state that includes states of the robot and N landmarks is:

$$X(k) = [X_v^T(k), X_{m1}^T(k), \dots, X_{mN}^T(k)]^T \quad (3)$$

The state equation of the system can be stated as:

$$X(k) = F(X(k-1), u(k), n(k)) \quad (4)$$

By combining (1) and (2) into (4), it becomes:

$$\begin{bmatrix} X_v(k) \\ X_{m1}(k) \\ \vdots \\ X_{mN}(k) \end{bmatrix} = \begin{bmatrix} F_v(X_v(k-1), u(k), n(k)) \\ X_{m1}(k) \\ \vdots \\ X_{mN}(k) \end{bmatrix} \quad (5)$$

The observation model is:

$$z(k) = h(X(k)) + n_z(k) \quad (6)$$

where $z(k)$ is the sensor measurement of the system state $X(k)$, $h(\cdot)$ is the transformation function that relates the system state to the sensor output $z(k)$, and $n_z(k)$ is the measurement noise at time k . In particular, when the bearing-only sensors are used to measure the locations of the landmarks, the observation equation is:

$$\begin{bmatrix} \theta_1(k) \\ \vdots \\ \theta_N(k) \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{y_{m1}(k) - y_v(k)}{x_{m1}(k) - x_v(k)}\right) - \phi_v(k) \\ \vdots \\ \tan^{-1}\left(\frac{y_{mN}(k) - y_v(k)}{x_{mN}(k) - x_v(k)}\right) - \phi_v(k) \end{bmatrix} + n_z(k) \quad (7)$$

where θ_i is the angle between the robot heading and the i th landmark when measured from the centre of the robot, as depicted in Fig 1.

If we assume that the system control noise $n(k)$ and $n_z(k)$ are Gaussian noise with zero mean and covariance of Σ and R , respectively, then the extended Kalman filter (EKF) can be used to solve the SLAM problem.

The EKF uses the linearised system and measurement equations to predict the system states. Particularly, the filtering process is achieved in the following steps:

By using (5) the system state can be estimated as:

$$\hat{X}(k) = F(\tilde{X}(k-1), u(k)) \quad (8)$$

In (8), the $\tilde{X}(k-1)$ represents the updated system state at time $k-1$, see (12). For $k=1$, $\tilde{X}(0)$ is the initial system state. In addition, the system covariance matrix $P(k)$ can be estimated using the following equation:

$$\hat{P}(k) = \nabla F_{x_v} P(k-1) \nabla F_{x_v}^T + \nabla F_u \Sigma \nabla F_u^T \quad (9)$$

where, $\nabla F_{x_v} = \partial F / \partial X_v$ and $\nabla F_u = \partial F / \partial u$.

The Kalman gain K can be calculated as:

$$K = \hat{P}(k) \nabla h_x^T S^{-1} \quad (10)$$

where, $\nabla h_x = \partial h / \partial X$, S is the covariance of the innovation (the difference between the observation and the estimated observation) given by:

$$S = \nabla h_x \hat{P}(k) \nabla h_x^T + R \quad (11)$$

The updated system state $\tilde{X}(k)$, is given by:

$$\tilde{X}(k) = \hat{X}(k) + K(z(k) - h(\hat{X}(k))) \quad (12)$$

The updated system covariance matrix P is:

$$P(k) = \hat{P}(k) - KSK^T \quad (13)$$

The detailed description of EKF based SLAM method can be found in a number of recent publications including [1].

One of the difficulties in bearing-only SLAM is the landmark initialisation issue. There are a number of methods available to deal with this problem. In this paper the multi-hypothesis approach introduced in [8] is used. This method is based on the sequential probability ratio test and has been demonstrated to be computationally efficient.

B. Near Minimum Time Path Planning in SLAM

This section deals with the main contribution of this paper; *i.e.*, how the robot should be controlled during the SLAM process in order to achieve a prescribed objective. The quality of the SLAM outcome can be measured by the quality of the map, the overall time used to obtain the map of a certain quality and the maximum uncertainty in the robot location.

In this section an algorithm to obtain the near minimum time paths for the robot such that the robot starts from an initial position and moves such that the uncertainty in the robot location is maintained to be within a prescribed limit and the final uncertainties of all the visible landmarks are within a preset tolerance.

In particular, the optimisation process is carried out in two phases: First, the robot path is only planned by a locally optimal ('greedy') planner that moves the robot in a direction that maximises the reduction of the map uncertainty (the covariance matrix P in (13)) while maintaining the robot localisation uncertainty within a set limit. Once, the map uncertainty is reduced to a preset level, the second phase of the optimal path planning is used to derive an overall time-optimal path. The following describes these two steps in detail.

1) Local Optimal ('Greedy') Path Planning

When a bearing-only sensor is used, there is insufficient information at the beginning on the SLAM process to obtain an estimate of the feature locations. Once the robot starts to move, bearing-only observations from different vantage points can be used to initialise the feature locations. The ability to predict the outcomes of the SLAM process when a certain set of control actions are taken is essential for any optimisation algorithm to succeed. However, this is not possible without a reasonable knowledge about the landmark locations, which is the case in the initial stages of the bearing only SLAM process. In this situation, a locally optimal action that is determined through the gradients of the objective function is the most appropriate. Once the feature locations are known such that the long term predictions are more realistic, the planning horizon of the optimisation process can be increased.

To minimise the covariance $P(k)$ in (13), the information matrix, *i.e.*, the covariance inverse $P(k)^{-1}$, is commonly used [16]. The information matrix can be calculated as:

$$P^{-1}(k) = \hat{P}^{-1}(k) + \nabla h_x^T R^{-1} \nabla h_x \quad (14)$$

where the superscript $^{-1}$ indicates the inverse of the matrix or the variable. the locally optimal 'greedy' path planning strategy can now be stated as:

$$\max_{v(k), \gamma(k)} \|P^{-1}(k)\| \quad (15)$$

subject to:

$$0 \leq v(k) \leq v_{\max} \quad \forall k \quad \gamma_{\min} \leq \gamma(k) \leq \gamma_{\max} \quad \forall k$$

and

$$\|P_{robot}(k)\| < \delta_{robot} \quad \forall k$$

where $\| \cdot \|$ is the Frobenius norm of the matrix, $v(k)$ and $\gamma(k)$ are the control inputs that represent the velocity and steering rate of the robot at time k , v_{max} , γ_{min} and γ_{max} are the limits of control velocity and steering rate, $P_{robot}(k)$ is the robot localisation uncertainty at time k , which is the first 3x3 sub-matrix in the $P(k)$ matrix, and δ is the allowed robot localisation uncertainty. This process will continue till the $\|P\|$ of every landmark is below a preset level of δ_{mg} .

It is assumed that the control signals to the robot, *i.e.*, the robot velocity and steering rate, can be changed instantaneously. This assumption can be relaxed at the cost of increased computational complexity.

2) Near Minimum Time Path Planning

Once the ‘greedy’ approach minimises the $\|P\|$ to the predefined level, using a longer planning horizon becomes realistic. Now the objective becomes achieving predefined map accuracy within the shortest possible time. Once again, it is assumed that the robot velocity and the velocity of the steering angle are bounded but can be changed instantaneously. Therefore, during the optimisation process, these control inputs are parameterized using a set of piecewise constants. Under these conditions, the problem of obtaining the time optimal trajectories for SLAM using bearing-only sensors can be formulated as:

Find the vehicle velocity $v(t)$ and the steering velocity $\gamma(t)$, $\forall t \in [0, t_f]$ such that t_f is a minimum, *i.e.*,

$$\min_{v, \gamma, t_f} t_f \quad (16)$$

subject to constraints:

$$0 \leq v(t) \leq v_{max}, \quad \gamma_{min} \leq \gamma(t) \leq \gamma_{max} \quad \forall t \in [0, t_f] \quad (17)$$

and the final conditions:

$$\|P_{robot}(t_f)\| \leq \delta_{robot}, \quad \|P_{mi}(t_f)\| \leq \delta_m \quad i = 1, \dots, N \quad (18)$$

are satisfied.

It is clear that the time-optimal control problem is to find a set of control inputs over a time period of t_f , such that at time t_f the localisation uncertainty of the robot is within a preset value of δ_{robot} and the uncertainties of all landmark mapping are below a preset value of δ_m .

Note that the estimation equations and the final time t_f are implicitly incorporated into the final conditions. The final time t_f can be introduced into the estimation equations using normalised time τ , defined as: $\tau = t/t_f$. In addition, the robot velocity and the steering velocity of the robot, expressed as a function of τ are parameterised using a set of piece-wise constants σ_{j1} and σ_{j2} can be written as:

$$v(\tau) = \sigma_{j1} [f(\tau - (j-1)\delta) - f(\tau - j\delta)],$$

$$\gamma(\tau) = \sigma_{j2} [f(\tau - (j-1)\delta) - f(\tau - j\delta)].$$

where $j = 1, \dots, N_p$, $\delta = 1/N_p$, N_p is the number of partitions used in the approximation, and f denotes the unit step function.

Now the robot’s localisation and mapping uncertainties can be obtained in terms of τ , σ_{j1} and σ_{j2} . The accuracy of the approximation can be improved by increasing the number of partitions N_p , but this will result in an increase in computational time required for obtaining time-optimal trajectories.

Now the near minimum time path planning problem becomes one of finding a suitable set of values for σ_{j1} and σ_{j2} such that constraints are satisfied and t_f is minimised. This can be achieved by using one of many techniques described in the literature for solving constrained nonlinear optimisation problems. In this paper the constrained minimisation routine (function *fmincon*) based on sequential quadratic programming available in MATLAB was used [21]. This type of optimisation techniques has also been used effectively in time-optimal control of robot manipulators [22] [23].

III. SIMULATION RESULTS

To examine the effectiveness of the proposed method, a robot that has a differential driving mechanism, performing bearing-only SLAM is simulated. By using the velocity, v , of the robot and the turning rate γ of the steering as the control for the robot with a sampling rate of Δt , the robot’s equation of motion can be expressed as:

$$\begin{bmatrix} x_v(k) \\ y_v(k) \\ \phi_v(k) \end{bmatrix} = \begin{bmatrix} x_v(k-1) + v(k)\Delta t \cos(\phi_v(k-1)) \\ y_v(k-1) + v(k)\Delta t \sin(\phi_v(k-1)) \\ \phi_v(k-1) + \gamma(k)\Delta t \end{bmatrix} + n(k) \quad (19)$$

To carry out the simulation, it is assumed that the robot has a bearing-only sensor that has a 180° view. Furthermore, the robot’s maximum velocity and the steering rate limits v_{max} , γ_{min} and γ_{max} are set to be 0.1m/sec and $\pm 5^\circ$ /sec respectively. In the environment where the SLAM is to be carried out, the robot is assumed to be able to observe all 5 landmarks. It is also assumed that the environment is free of obstacles.

As stated in section II B, the ‘greedy’ path planning is carried out at the beginning of the SLAM process until the uncertainty of the landmarks is below $0.4 m^2$, *i.e.*, δ_{mg} is set to be $0.4 m^2$. At the completion of the ‘greedy’ only control the SLAM results are shown in Fig 3.

Then the proposed near minimum time path planning strategy is used to obtain the overall control signals to achieve acceptable SLAM quality in a minimum time. During the optimisation process, the partition number N_p is set to 20. The δ_{robot} and δ_m in (18) are set to $0.01 m^2$ and $0.04 m^2$ respectively. The final map obtained by using the planned optimal control signals to control the robot can be seen in Fig 4. The time taken for the robot to achieve such a map is 82.8 seconds. This time includes the 13.1 seconds used in the ‘greedy’ only control phase. The overall control signals generated in the two optimal control phases are shown in Fig 5, while in Figs 6 and 7, the estimates of the errors for location of the robot and the landmarks are given, respectively.

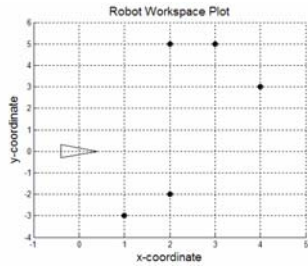


Fig. 2 The robot workspace that includes landmarks.

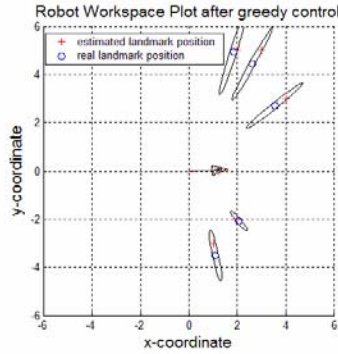


Fig. 3. The SLAM results after greedy control

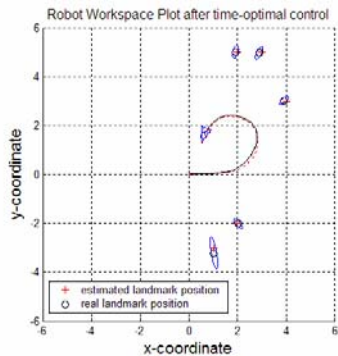


Fig. 4. The SLAM results at the completion of the time-optimal control.

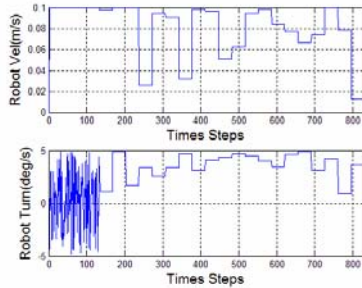


Fig. 5. The total control action of the proposed optimal controller.

The changes of the error covariance for the robot and the landmarks are shown in Figs 8 and 9. As shown in Fig 9, with the exception of one, all landmarks' final covariance at t_f are within the set value of $0.04 m^2$. The reason for the final

mapping uncertainty to exceed the preset value is possibly the fact that during the computation of the optimal control actions, the exact locations of the landmarks are not known. Thus the exact values of the constraints can not be computed at the beginning of the optimisation process. In fact, this highlights the need for the gradient based solution with effectively zero planning horizon in the initial stages of the optimisation process, as it is clearly essential to recompute the optimal control actions once it is apparent that the conditions under which they were computed have changed.

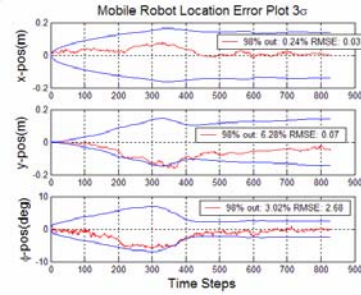


Fig. 6. Errors in the estimated robot location and the associated covariances..

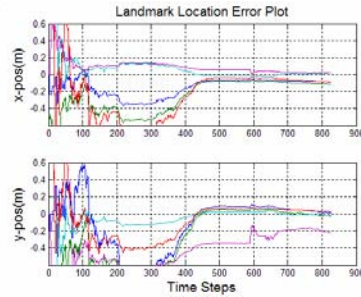


Fig. 7. The estimation errors of the landmarks locations.

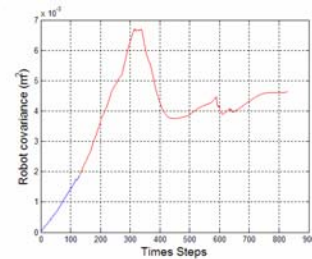


Fig. 8. The robot location accuracy $\|P_{robot}\|$.

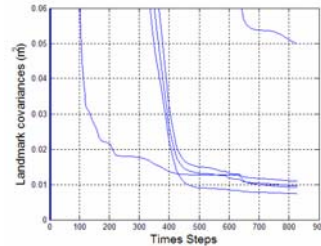


Fig. 9. Landmark location accuracy $\|P_{mi}\|$.

IV. CONCLUSIONS

In a real world scenario, it is important to control the actions of the robot in order to achieve good quality maps. This is particularly important in the case of SLAM using bearing only sensors. In this paper a near minimum time path planning method for SLAM is introduced. The effectiveness of the proposed method is demonstrated by computer simulations. It is shown that the proposed method gives a reasonable result, although further work is required to verify its operation in a real-life environment. Current simulations are conducted using MATLAB. The next step of the project will be to migrate this algorithm to the C++ so that an indoor Pioneer robot equipped with a camera can be used for experiments.

In addition, the proposed method is implemented under the assumption that all landmarks (features) are visible once the robot starts moving. It is also assumed that when the robot is moving along the planned near minimum time path no new features will be discovered. These situations refer to the scenario when the map of a given local region needs to be improved before the robot moves on. For example, when the robot visits an office room it may be desirable for it to leave the room only after it has obtained a sufficiently accurate map. Therefore, the proposed algorithm deals only with the first step in active SLAM. Clearly, as the robot explores an environment new landmarks will appear at unknown locations, thus any plans generated using the landmarks already observed become suboptimal. One possible way to overcome this shortcoming is to use the model predictive control strategy as suggested in [19]. Another strategy is to restart planning when the environment is "sufficiently" different from the map used for generating the path. Moreover, the need for localisation accuracy needs to be balanced against the desire for exploration. This issue is not considered in this paper and needs further investigation.

Furthermore, occasional violations of the final state constraints stated in (18) were seen during the simulations. This can be overcome by either replanning or by setting the constraints to a tighter level, *i.e.*, setting δ_m to a value that is smaller than the acceptable one. While the number of landmarks used in this paper is small, the method does provide the fundamental base on which a large scaled active SLAM can be built. We are currently in the process of implementing a large scale active SLAM strategy by incorporating the "replanning" method mentioned above.

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 Transactions on Robotics and Automation*. Vol 17 No 3 2001, pp 229-241.
- [2] J. Guivant, and E. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation." *IEEE Transactions on Robotics and Automation*. Vol 17 No 3 2001, pp 242-257.
- [3] S. Thrun, "Robotic Mapping: A Survey", Technical report, CMU-CS-02-111, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, Feb 2002.
- [4] J.-D. Tardos, 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.
- [5] J.-A. Perez, J.-A. Castellanos, J.-M. Montiel, J. Neira and J.-D. Tardos, "Continuous Mobile Robot Localization: Vision vs. Laser", *Proc. 1999 IEEE Intl. Conf. on Robotics and Automation*, Detroit, Michigan, May 1999, pp. 2917-2923.
- [6] M. Gavish and A. Weiss, "Performance analysis of bearing-only target location algorithms." *IEEE Transactions on Aerospace and Electronic Systems*. Vol. 28, No. 3, 1992. pp. 817-828.
- [7] J.-P. Le Cadre, and C. Jauffret, "Discrete-time observability and estimability analysis for bearings-only target motion analysis." *IEEE Transactions on Aerospace and Electronic Systems*. Vol. 33, No. 1, 1997. pp. 178-201.
- [8] Y. Oshman, and P. Davidson, "Optimization of observer trajectories for bearings-only target localization." *IEEE Transactions on Aerospace and Electronic Systems*. Vol. 35, No. 3, 1999. pp. 892-902.
- [9] T. Bailey, "Constrained initialisation for bearing-only SLAM." *Proceedings of 2003 IEEE International Conference on Robotics & Automation*. Taipei, 2003. pp. 1966-1971.
- [10] N.-M. Kwok, and G. Dissanayake, "Bearing-only SLAM in indoor environments using a modified particle filter." *Proceedings on Australasian Conference on Robotics and Automation*. CD-ROM, Brisbane, 2003
- [11] N.-M. Kwok, and G. Dissanayake, "An efficient multiple hypothesis filter for bearing-only SLAM." *International Conference on Intelligent Robots and Systems*. 2004.
- [12] A. Wald, *Sequential analysis*. Dover Publications Inc, NY. 1973.
- [13] J.-H. Kim and S. Sukkarieh, "Airborne Simultaneous Localisation and Map Building", *Proc. 2003 IEEE Intl. Conf. on Robotics and Automation*, Taipei, Taiwan, September 2003, pp. 406-411.
- [14] A.-J. Davison, Y.-G. Cid and N. Kita, "Real-time 3D SLAM with Wide-Angle Vision", *Proc. IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, July 2004.
- [15] A.-J. Davison, "Mobile Robot Navigation Using Active Vision", Ph.D. Thesis, University of Oxford, 1998.
- [16] H. Feder, J. Leonard, and C. Smith, "Adaptive mobile robot navigation and mapping." *International Journal of Robotics Research*. Vol 18, No 7, pp. 650-668, 1999.
- [17] A. Makarenko, S. Williams, F. Bourgault, and H. Durrant-Whyte, "An experiment in integrated exploration," *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems, Oct 2002*, pp. 534 – 539.
- [18] S. Huang, Z. Wang, and G. Dissanayake, "Time optimal robot motion control in simultaneous localization and map building (SLAM) problem." *International Conference on Intelligent Robots and Systems*. 2004, *Sendai, Japan*, 3110-3115.
- [19] S. Huang, N.-M. Kwok, G. Dissanayake, Q.-P. Ha and G. Fang, "Multi-Step Look-Ahead Trajectory Planning in SLAM: Possibility and Necessity." *Proceedings of the 2005 IEEE International Conference on Robots and Automation, April 2005*, pp. 1103 – 1108.
- [20] R. Sim, "Stable exploration for bearings-only SLAM". *Proceedings of the 2005 IEEE International Conference on Robots and Automation, April 2005*, pp. 2422 – 2427.
- [21] The Mathworks, "Optimisation toolbox – user's manual," 2003
- [22] G. Dissanayake, "High-speed positioning of robot manipulators," *Journal of Systems Engineering*, Vol 1, pp. 103 – 113, 1991
- [23] G. Fang and G. Dissanayake, "Experiments on a neural network based method for time-optimal trajectory planning," *Robotica*, Vol. 16, pp. 143 – 158, 1998.