15th Triennial World Congress, Barcelona, Spain

FIELD DEPLOYMENT OF THE SIMULTANEOUS LOCALISATION AND MAPPING ALGORITHM

Stefan B. Williams [*] **Gamini Dissanayake** [**]
Hugh Durrant-Whyte [*]

[*] *Australian Centre for Field Robotics, J04, University of Sydney, Sydney, NSW, 2006 Australia*
[**] *Faculty of Engineering, University of Technology Sydney, Sydney, NSW, 2006 Australia*

Abstract: Autonomous localisation and mapping requires a vehicle to start in an unknown location in an unknown environment and then to incrementally build a map of landmarks present in this environment while simultaneously using this map to compute absolute vehicle location. The theoretical basis of the solution to this problem, known as Simultaneous Localisation and Mapping (SLAM), is now well understood. A number of approaches to SLAM have appeared in the recent literature. This paper presents results of deployment of the algorithm undertaken at the Australian Centre for Field Robotics in a variety of field applications. The algorithm has been used in indoor environments, using a high speed land vehicle travelling in a park and on a submersible vehicle.

Keywords: Autonomous Mobile Robots, Simultaneous Localisation and Mapping, Extended Kalman Filter, Navigation

1. INTRODUCTION

Simultaneous Localisation and Mapping (SLAM) is the process of concurrently building a feature based map of the environment and using this map to obtain estimates of the location of the vehicle. The theoretical basis of the solution to this problem is now well understood. In essence, the vehicle relies on its ability to extract useful navigation information from the data returned by its sensors. The SLAM algorithm has recently seen a considerable amount of interest from the mobile robotics community as a tool to enable fully autonomous navigation (Castellanos *et al.* 1999)(Dissanayake *et al.* 2001). The prospect of deploying a robotic vehicle that can build a map of its environment while simultaneously using that map to localise itself promises to allow these vehicles to operate autonomously for long periods of time in unknown environments. Much of this work has focused on the use of stochastic estimation techniques to build and maintain estimates of vehicle and map feature locations. In particular, the Extended Kalman Filter (EKF) has been proposed as a mechanism by which the information gathered by the vehicle can be consistently fused to yield bounded estimates of vehicle and landmark locations in a recursive fashion (Dissanayake *et al.* 2001)(Leonard and Durrant-Whyte 1991). Recent work has concentrated on the development of efficient methods for implementing the algorithm using relative maps (Csorba 1997)(Newman 1999) and submaps (Leonard and Feder 1999) (Williams 2001).

While the Kalman Filter approach to the SLAM problem has received considerable interest, alternative philosophies also appear in the literature. A number of research teams have tackled the problem of map building and localisation using batch estimation techniques (Lu and Milios 1997) (Gutmann and Konolige 2000) (Thrun *et al.* 1998). Still other approaches to the problem of map building and localisation have done away with the rigorous mathematical models of the vehicle and sensing properties and have relied instead on more qualitative knowledge of the nature of the environment (Brooks 1986)(Kuipers and Byun 1991)(Levitt and Lawton 1990). While all of these alternative approaches to the problem have their own particular strengths, this paper will be concerned primarily with a recursive, on-line approach to the problem and will rely on the EKF as the primary means of simultaneously building a map while localising the vehicle.

This paper presents results of deployment of the SLAM algorithm undertaken at the Australian Centre for Field Robotics in a variety of fielded applications. Section 2 introduces the basics of the SLAM algorithm. Section 3 presents results of the application of the SLAM algorithm in an indoor, office environment, operating in an outdoor environment on a land vehicle and on a submersible vehicle in an ocean environment. Finally, Section 4 summarises the paper and provides concluding remarks.

2. THE ESTIMATION PROCESS

The localisation and map building process consists of generating the best estimate for the system states given the information available to the system. This can be accomplished using a recursive, three-stage procedure comprising prediction, observation and update steps known as the Extended Kalman Filter (EKF) (Dissanayake *et al.* 2001). The Kalman filter is a recursive, least squares estimator and produces at time i a minimum mean-squared error estimate $\hat{\mathbf{x}}(i|j)$ of the state $\mathbf{x}(i)$ given a sequence of observations up to time j, $\mathbf{Z}^j = \{\mathbf{z}(1)...\mathbf{z}(j)\}$ (Gelb 1996)(Maybeck 1982)

$$\hat{\mathbf{x}}(i|j) = \mathsf{E}[\mathbf{x}(i)|\mathbf{Z}^j] \tag{1}$$

For the Simultaneous Localisation and Mapping algorithm, the EKF is used to estimate the pose of the vehicle $\hat{\mathbf{x}}_v^+(k)$ along with the positions of the n_f observed features $\hat{\mathbf{x}}_i^+(k), i = 1...n_f$. The augmented state estimate consists of the current vehicle state estimates as well as those associated with the observed features

$$\hat{\mathbf{x}}^+(k) = \begin{bmatrix} \hat{\mathbf{x}}_v^+(k) \\ \hat{\mathbf{x}}_1^+(k) \\ \vdots \\ \hat{\mathbf{x}}_{n_f}^+(k) \end{bmatrix} \tag{2}$$

The covariance matrix for this state estimate is defined through

$$\mathbf{P}^+(k) = \mathsf{E}[(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k))(\mathbf{x}(k) - \hat{\mathbf{x}}^+(k))^T|\mathbf{Z}^k]. \tag{3}$$

This defines the mean squared error and error correlations in each of the state estimates. For the case of the SLAM filter, the covariance matrix takes on the following form using $\mathbf{P}_{vv}^+(k)$ to represent the vehicle covariances, $\mathbf{P}_{mm}^+(k)$ to represent the map covariances and $\mathbf{P}_{vm}^+(k)$ to represent the cross-covariance between the vehicle and the map.

$$\mathbf{P}^+(k) = \begin{bmatrix} \mathbf{P}_{vv}^+(k) & \mathbf{P}_{vm}^+(k) \\ \mathbf{P}_{vm}^{+T}(k) & \mathbf{P}_{mm}^+(k) \end{bmatrix} \tag{4}$$

2.1 *Prediction*

The prediction stage of the filter uses a model of the motion of the vehicle, $\mathbf{f}(\hat{\mathbf{x}}_v^+(k-1), \mathbf{u}(k))$, to generate an estimate of the vehicle position, $\hat{\mathbf{x}}_v^-(k)$, at instant k given the information available to instant $k-1$. The landmarks are generally assumed to be stationary. Together, these two models result in the propagation of the augmented state matrix during the prediction cycle of the filter.

$$\begin{bmatrix} \hat{\mathbf{x}}_v^-(k) \\ \hat{\mathbf{x}}_m^-(k) \end{bmatrix} = \begin{bmatrix} \mathbf{f}(\hat{\mathbf{x}}_v^+(k-1), \mathbf{u}(k)) \\ \hat{\mathbf{x}}_m^+(k-1) \end{bmatrix} \tag{5}$$

The covariance matrix must also be propagated through the vehicle model as part of the prediction. The Extended Kalman Filter linearises the propagation of uncertainty about the current state estimate $\hat{\mathbf{x}}^+(k-1)$ using the the Jacobian $\nabla_x \mathbf{f}(k)$ of $\mathbf{f}$ evaluated at $\hat{\mathbf{x}}^+(k-1)$ as

$$\mathbf{P}^-(k) = \nabla_x \mathbf{f}(k)\mathbf{P}^+(k-1)\nabla_x \mathbf{f}^T(k) + \mathbf{Q}(k). \tag{6}$$

For the SLAM algorithm, this step in the filter can be simplified because of the assumption that the feature states are stationary. This allows the complexity of computing the predicted covariance to be reduced by requiring that only the variances associated with the vehicle and the cross covariance terms between the vehicle and the map are updated during the prediction step.

2.2 *Observation*

The fusion of the observation into the state estimate is accomplished by first calculating a predicted observation, $\hat{\mathbf{z}}^-(k)$, using the observation model, $\mathbf{h}$ as

$$\hat{\mathbf{z}}^-(k) = \mathbf{h}(\hat{\mathbf{x}}^-(k)) \tag{7}$$

When observations are received from the vehicle's on-board sensors they must be associated with particular features in the environment. The difference between the actual observation, $\mathbf{z}(k)$, received from the system's sensors and the predicted observation, $\hat{\mathbf{z}}^-(k)$, is termed the innovation $\nu(k)$,

$$\nu(k) = \mathbf{z}(k) - \hat{\mathbf{z}}^-(k) \tag{8}$$

The innovation covariance, $\mathbf{S}(k)$, is computed from the current state covariance estimate, $\mathbf{P}^-(k)$, the Jacobian of the observation model, $_{x}\mathbf{h}(k)$, and the covariance of the observation model $\mathbf{R}(k)$.

$$\mathbf{S}(k) = \ _{x}\mathbf{h}(k)\mathbf{P}^{-}(k) \ _{x}\mathbf{h}^{T}(k) + \mathbf{R}(k) \tag{9}$$

The innovations and their associated covariances can be used to validate measurements before they are incorporated into the filtered estimates. The calculation of the innovation covariance can be simplified by noting that each observation is only a function of the feature being observed.

2.3 *Update*

Once the observation has been associated with a particular feature in the map, the state estimate can be updated using the optimal gain matrix $\mathbf{W}(k)$. This gain matrix provides a weighted sum of the prediction and observation and is computed using the innovation covariance, $\mathbf{S}(k)$ and the predicted state covariance, $\mathbf{P}^{-}(k)$. The weighting factor is proportional to $\mathbf{P}^{-}(k)$ and inversely proportional to the innovation covariance (Smith *et al.* 1990). This is used to compute the state update $\hat{\mathbf{x}}^{+}(k)$ as well as the updated state covariance $\mathbf{P}^{+}(k)$.

$$\hat{\mathbf{x}}^{+}(k) = \hat{\mathbf{x}}^{-}(k) + \mathbf{W}(k) \ (k) \tag{10}$$

$$\mathbf{P}^{+}(k) = \mathbf{P}^{-}(k) \check{S} \ \mathbf{W}(k)\mathbf{S}(k)\mathbf{W}^{T}(k) \tag{11}$$

where

$$\mathbf{W}(k) = \mathbf{P}^{-}(k) \ _{x}\mathbf{h}^{T}(k)\mathbf{S}^{-1}(k) \tag{12}$$

2.4 *Feature Initialisation*

When a new feature is observed its estimate must be properly initialised and added to the state vector. Given a current state estimate, $\hat{\mathbf{x}}^{-}(k)$, comprised of the vehicle state, $\hat{\mathbf{x}}_{v}^{-}(k)$, and the map states, $\hat{\mathbf{x}}_{m}^{-}(k)$, a relative observation between the vehicle and the new feature, $\mathbf{z}(k)$, and a feature initialisation model, $\mathbf{g}_{i}(\cdot,\cdot)$, that maps the current vehicle state estimate and observation to a new feature estimate, the initial estimate of the feature state is

$$\hat{\mathbf{x}}_{i}^{+}(k) = \mathbf{g}_{i}(\hat{\mathbf{x}}_{v}^{-}(k), \mathbf{z}(k)). \tag{13}$$

These new state estimates are then appended to the state vector as new map feature elements.

The covariances of the new feature estimates must also be properly initialised since the initial estimate depends on the current vehicle estimate and is therefore correlated with the rest of the vehicle and other map state estimates. Ignoring the correlation between the new state estimates and the remainder of the map can lead to inconsistency in the filtering process (Csorba 1997). The SLAM covariance matrix is first augmented with the observation covariance and the cross-covariance terms between the existing state elements and the new state estimates are computed.

$$\mathbf{P}^{*-}(k) = \begin{matrix} \mathbf{P}_{vv}^{-}(k) & \mathbf{P}_{vm}^{-}(k) & 0 \\ \mathbf{P}_{vm}^{-T}(k) & \mathbf{P}_{mm}^{-}(k) & 0 \\ 0 & 0 & \mathbf{R}(k) \end{matrix} \tag{14}$$

The final covariance is computed by projecting the augmented covariance matrix through the Jacobian $_{x}\mathbf{g}(k)$ of the initialisation function, $\mathbf{g}_{i}$, with respect to the augmented states,

$$\mathbf{P}^{+}(k) = \ _{x}\mathbf{g}(k)\mathbf{P}^{*-}(k) \ _{x}\mathbf{g}^{T}(k) \tag{15}$$

Proper initialisation of the feature estimates is necessary to maintain their consistency and to generate the correct cross-covariances between the feature and vehicle estimates.

2.5 *Computational Complexity*

One significant obstacle on the road to the implementation and deployment of large scale SLAM algorithms is the computational effort required to maintain the correlation information between features in the map and between the features and the vehicle. Performing the update of the covariance matrix is of $O(n^3)$ for a straightforward implementation of the Kalman Filter. In the case of the SLAM algorithm, this complexity can be reduced to $O(n^2)$ given the sparse nature of typical observations. Even so, this implies that the computational effort will grow with the square of the number of features maintained in the map. For maps containing more than a few tens of features, this computational burden will quickly make the update intractable - especially if the observation rates are high. An effective map-management technique is therefore required in order to help manage this complexity.

A number of potential methods by which the growth of the computational burden imposed by the covariance update can be regulated have been described in the literature. Some approaches rely on reducing the number of features incorporated into the map (Davison 1998)(Dissanayake *et al.* 2000) while others use a suboptimal update step (Uhlmann *et al.* 1997)(Guivant *et al.* 2000). Still other approaches have examined the effect of changing the map representation on the computational complexity of the update; using relative maps (Csorba and Durrant-Whyte 1997)(Newman 1999) or submaps (Chong and Kleeman 1998)(Leonard and Feder 1999)(Williams 2001) recent work has shown that the algorithm is tractable for larger maps.

Fig. 1. One of the indoor vehicles used for testing the SLAM algorithm. This vehicle is equipped with a scanning laser range finder, sonar and wireless communications.

Fig. 2. A map of an indoor, office environment generated using a scanning laser range finder.

3. FIELD DEPLOYMENT

The goal of any Field Robotics project should ultimately be the deployment of algorithms using vehicles operating in natural environments. Recent work undertaken at the Australian Centre for Field Robotics has shown that the SLAM techniques can be adopted in real world settings. This section provides a brief summary of results of these deployments and provides references to more detailed descriptions of some of the systems.

Fig. 3. The utility vehicle, equipped with laser range finder and wheel encoders, is driven in a park. The trunks of trees observed by the laser are tracked as features in the SLAM map.

Fig. 4. The path of the vehicle and the SLAM map of features (image courtesy of Jose Guivant).

3.1 *Indoor SLAM*

Indoor mobile robotic platforms, an example of which is shown in Figure 1, provide an ideal means with which to test novel navigation algorithms prior to their deployment in field environments. The ACFR has recently established an indoor robotics laboratory to facilitate this work. Figure 2 shows an example of results of the application of the SLAM algorithm in an indoor environment. The SLAM map consists of estimates of the position of retro-reflective strips that have been mounted in an office environment. Observations of the positions of these strips are used to correct errors in odometry. The laser returns are stored relative to the vehicle location estimates and plotted in this view, yielding the walls apparent in the map. Research into autonomous exploration as well as multiple vehicle SLAM is currently underway.

3.2 *Land vehicle SLAM*

Experiments have been conducted using a land vehicle operating in a park-like setting (Guivant *et* *al.* 2000). These experimental runs are performed in a totally unstructured environment. Figure 3 shows the experimental car in the testing environment. The vehicle is equipped with a scanning laser range finder and wheel encoders. The laser scans are processed to extract circular cross-sections returned from the trunks of trees in the park. The position of the trees, together with trunk diameter, are estimated as part of the SLAM algorithm. The estimate of trunk diameter is used to facilitate the data association process.

In this case the car is running on a grassy surface. Modelling errors are expected due to wheel slip and the 3-D nature of the environment. In spite of these modelling errors, the SLAM algorithm is able to accurately track the position of the vehicle as it travels through the park. The tracking accuracy has been verified against a GPS/INS navigation system (Guivant *et al.* 2000). Another objective of these tests was to determine the convergence characteristics of the algorithm when running in large areas for long periods of time. In this case the vehicle was running for more than 20 minutes and the resulting map and vehicle path are shown in Figure 4. The point features represent the trunks of the trees seen by the vehicle.

3.3 *Subsea SLAM*

The SLAM algorithms have also been demonstrated in an underwater environment off the coast of Sydney, Australia (Williams *et al.* 2001). The submersible was deployed in a natural inlet with a number of sonar targets positioned in a straight line at intervals of 10m. The vehicle controls were set to maintain a constant heading and altitude during the run. Once the vehicle had reached the end of its tether (approximately 50m) it was turned around and returned along the line of targets. The slope of the inlet in which the vehicle was deployed meant that the depth of the vehicle varied between approximately 1m and 5m over the course of the run.

Figure 6 shows a plot of the final map obtained by the SLAM algorithm. The position of the sonar targets are clearly visible. The absolute location of all the potential point targets identified based on the sonar principal returns are also shown in this map. These locations were computed using the estimated vehicle location at the instant of the corresponding sonar return. The returns seen near the top and bottom of the map are from the reef walls. As can be seen, large clusters of returns have been successfully identified as landmarks.

Fig. 5. Oberon at Sea

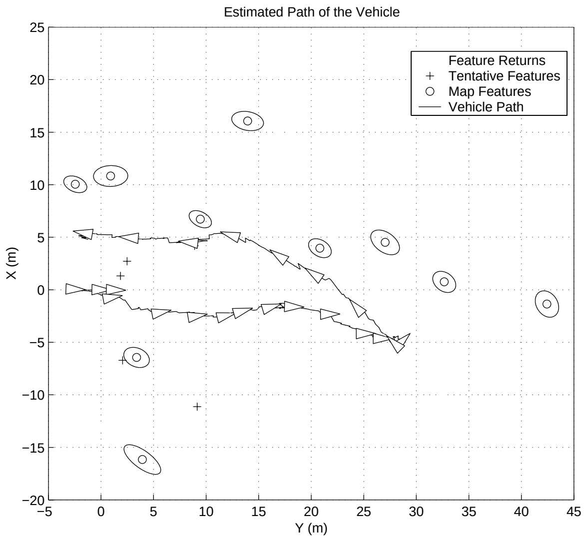

Fig. 6. The path of the vehicle computed using the beacon returns. In addition to the identified target returns, strong energy returns from the reef walls and the sea floor can also be extracted from the sonar pings.

4. CONCLUSIONS

This paper has presented recent field deployment of the Simultaneous Localisation and Mapping algorithm undertaken at the Australian Centre for Field Robotics. The approach has been shown to perform well in a number of environments. The algorithms have been tested in indoor, office environments, on land vehicles driving in a park-like setting and on a submersible vehicle operating in a natural environment off the coast of Sydney, Australia.

Work currently underway is concentrating on the deployment of the algorithm in additional field applications, such as using airborne vehicles. This work is complicated by the complexity of implementing the algorithm in a true three dimensional space on a platform moving at high speeds. Multiple vehicle SLAM as well as incorporating localisation into the task of exploration are also areas of active research.

5. REFERENCES

Brooks, R.A. (1986). A robust, layered control system for a mobile robot. *IEEE Transactions on Robotics and Automation* **2**(1), 14–23.

Castellanos, J.A., J.M.M. Montiel, J. Neira and J.D. Tardos (1999). The SpMap: A probabilistic framework for simultaneous localization and map building. *IEEE Transactions on Robotics and Automation* **15**(5), 948–952.

Chong, K.S. and L. Kleeman (1998). Large scale sonarray mapping using multiple connected local maps. In: *Field and Service Robotics* (A. Zelinsky, Ed.). pp. 507–514. Springer-Verlag.

Csorba, M. (1997). Simultaneous Localisation and Map Building. PhD thesis. University of Oxford.

Csorba, M. and H.F. Durrant-Whyte (1997). New approach to map building using relative position estimates. In: *Proc. of SPIE : Navigation and Control Technologies for Unmanned Systems II*. Vol. 3087. The International Society for Optical Engineering. pp. 115–125.

Davison, A. (1998). Mobile Robot Navigation Using Active Vision. PhD thesis. University of Oxford.

Dissanayake, M.W.M.G., H.F. Durrant-Whyte and T. Bailey (2000). A computationally efficient solution to the simultaneous localisation and map building (SLAM) problem. In: *Proc. IEEE Int. Conf. on Robotics and Automation*. Vol. 2. IEEE. pp. 1009–14.

Dissanayake, M.W.M.G., P. Newman, S. Clark, H.F. Durrant-Whyte and M. Csobra (2001). A solution to the simultaneous localization and map building (slam) problem.. In: *IEEE Transactions on Robotics and Automation*. Vol. 17(3). Sydney, Australia. pp. 229–241.

Gelb, A. (1996). *Applied Optimal Estimation*. 14th ed.. MIT Press.

Guivant, J., E.M. Nebot and H.F. Durrant-Whyte (2000). Simultaneous localization and map building using natural features in outdoor environments. In: *Proc. 6th Int. Conference on Intelligent Autonomous Systems*. Vol. 1. pp. 581–588.

Gutmann, J.S. and K. Konolige (2000). Incremental mapping of large cyclic environments. In: *Proc. IEEE International Symposium on Computational Intelligence in Robotics and Automation*. IEEE. pp. 318–325.

Kuipers, B.J. and Y.T. Byun (1991). A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and Autonomous Systmes* **8**(1-2), 47–63.

Leonard, J.J. and H.F. Durrant-Whyte (1991). Simultaneous map building and localisation for an autonomous mobile robot. In: *IEEE/RSJ International Workshop on Intelligent Robots and Systems*. Vol. 3. IEEE/RSJ. pp. 1442–1447.

Leonard, J.J. and H.J.S. Feder (1999). A computationally efficient method for large-scale concurrent mapping and localization. In: *Proc. Ninth International Symposium on Robotics Research*. International Foundation of Robotics Research. pp. 169–176.

Levitt, T.S. and D.T. Lawton (1990). Qualitative navigation for mobile robots. *Artificial Intelligence Journal* **44**(3), 305–360.

Lu, F. and E. Milios (1997). Robot pose estimation in unknown environments by matching 2d range scans. *Journal of Intelligent and Robotic Systems* **18**(3), 249275.

Maybeck, P. (1982). *Stochastic Models Estimation and Control*. Vol. 1. Academic Press.

Newman, P. (1999). On The Structure and Solution of the Simultaneous Localisation and Map Building Problem. PhD thesis. University of Sydney, Australian Centre for Field Robotics.

Smith, R., M. Self and P. Cheeseman (1990). Estimating uncertain spatial relationships in robotics. *Autonomous Robot Vehicles* pp. 167–193.

Thrun, S., D. Fox and W. Burgard (1998). A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning and Autonomous Robots (joint issue)*.

Uhlmann, J., S. Julier and M. Csorba (1997). Nondivergent simultaneous map building and localization using covariance intersection. In: *Proc. of SPIE : Navigation and Control Technologies for Unmanned Systems II*. Vol. 3087. The International Society for Optical Engineering. pp. 2–11.

Williams, S.B. (2001). Efficient Solutions to Autonomous Mapping and Navigation Problems. PhD thesis. University of Sydney, Australian Centre for Field Robotics.

Williams, S.B., G. Dissanayake and H.F. Durrant-Whyte (2001). Towards terrain-aided navigation for underwater robotics. *Advanced Robotics* **15**(5), 533–550.