ARTICLE IN PRESS

ELSEVIER

Robotics and Autonomous Systems 984 (2002) 1–12

Robotics and Autonomous Systems

www.elsevier.com/locate/robot

Simultaneous localization and map building using natural features and absolute information

Jose Guivant, Favio Masson, Eduardo Nebot*

Department of Mechanical and Mechatronic Engineering, Australian Centre for Field Robotics, The University of Sydney, 2006 Sydney, NSW, Australia

Abstract

This work presents real time implementation algorithms of simultaneous localization and map building (SLAM) with emphasis to outdoor land vehicle applications in large environments. It presents the problematic of outdoors navigation in areas with combination of feature and featureless regions. The aspect of feature detection and validation is investigated to reliably detect the predominant features in the environment. Aided SLAM algorithms are presented that incorporate absolute information in a consistent manner. The SLAM implementation uses the compressed filter algorithm to maintain the map with a cost proportional to number of landmarks in the local area. The information gathered in the local area requires a full SLAM update when the vehicle leaves the local area. Algorithms to reduce the full update computational cost are also presented. Finally, experimental results obtained with a standard vehicle running in unstructured outdoor environment are presented.
© 2002 Published by Elsevier Science B.V.

Keywords: SLAM; Outdoors navigation; Guidance; Mobile robots

1. Introduction

The problem of localization given a map of the environment or estimating the map knowing the vehicle position is known to be a solved problem and in fact applied in many research and industrial applications [1–3]. A much more fundamental problem is when both the map and the vehicle position are not known. This problem is usually referred as simultaneous localization and map building (SLAM) [4]/concurrent map building and localization (CML) [7]. It has been addressed using different techniques such as in [8] where approximation of the probability density functions with samples is used to represent uncertainty. The algorithm is suitable to handle multi-modal distribution. Although it has proven to be robust in many indoor localization applications, due to the high computation requirements this method has not been used for real time SLAM yet, although work is in progress to overcome this limitation.

Kalman filters can also be extended to solve the SLAM problem [6,9–11], once appropriate models for the vehicle and sensors are obtained. This method requires the robot to be localized all the time with a certain accuracy. This is not an issue for many industrial applications [2,3,12,13], where the navigation system has to be designed with enough integrity in order to avoid/detect degradation of localization accuracy. For these applications, the Kalman filter with Gaussian assumptions is the preferred approach to achieve the degree of integrity required in such environments.

One of the main problems with the SLAM algorithm has been the computational requirements that

* Corresponding author.
E-mail addresses: jguivant@acfr.usyd.edu.au (J. Guivant), f.masson@acfr.usyd.edu.au (F. Masson), nebot@acfr.usyd.edu.au (E. Nebot).

0921-8890/02/$ – see front matter © 2002 Published by Elsevier Science B.V.
PII: S0921-8890(02)00233-6

UNCORRECTED PROOF

is of the order $\sim O((2N)^2)$ [14], where N being the number of landmarks in the map. In large environments, the number of landmarks detected will make the computational requirement to be beyond the power capabilities of the computer resources. The computational issues of SLAM have been addressed with a number of sub-optimal simplification, such as [4,7]. In [5], a compressed algorithm was presented that allows the incorporation of the information gathered in a local area with a cost proportional to the number of landmarks in this area. The information can be stored and then transferred to the rest of the map in a single iteration at full SLAM computational cost. This paper presents a sub-optimal simplification to update the covariance matrix of the states with reduced computational cost when full SLAM is required. In several applications, the mobile vehicle needs to navigate in open areas where no features can be detected. In such cases, absolute position information such as GPS can be made available to reduce the navigation error. This paper address the problem of incorporating absolute information under the SLAM framework. The convergence and accuracy of the algorithms are tested in a large outdoor environment with regions where different types of information is available.

This paper is organized as follows. Section 2 presents an introduction to the SLAM problem and the vehicle and sensor models used in this application. Section 3 presents the navigation environment and the algorithms used to detect and validate the most relevant features in the environment. Section 4 presents important implementation issues such as a sub-optimal method to complement the compressed algorithm and a formulation to use the SLAM aided by external absolute information. Section 5 presents experimental results in unstructured outdoor environments. Finally, Section 6 presents conclusions.

2. Simultaneous localization and map building

When absolute position information is not available it is still possible to navigate with small errors for long periods of time. The SLAM algorithm addresses the problem of a vehicle with known kinematics, starting at an unknown position and moving through an unknown environment populated with some type of features. The algorithm uses dead reckoning and relative observation to features in order to estimate the position of the vehicle and to build and maintain a navigation map as shown in Fig. 1. With appropriate planning, the vehicle will be able to simultaneously navigate and build a relative map of the environment. If the initial position is known with respect to a global reference frame or if absolute position information is obtained during the navigation task then the map can be registered to the global frame. If not the vehicle can still navigate in the local while exploring and incorporating new areas to the map. A typical kinematics model of a land vehicle can be obtained from Fig. 2. The steering control α and the speed v_c are used with the kinematics model to predict the position of the vehicle. The external sensor information is processed to extract features of the environment, in this case called $B_{i\,(i=1,\dots,n)}$, and to obtain relative range and bearing, $z(k) = (r, \beta)$, with respect to the vehicle pose. Considering that the vehicle is controlled through a demanded velocity v_c and steering angle α the process model that predicts the trajectory of the centre of the back axle is given by

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\phi}_c \end{bmatrix} = \begin{bmatrix} v_c \cos(\phi) \\ v_c \sin(\phi) \\ \dfrac{v_c}{L} \tan(\alpha) \end{bmatrix} + \gamma, \tag{1}$$

where L is the distance between wheel axles and γ the zero mean Gaussian white noise. The observation equation relating the vehicle states to the observations is

$$z = h(X, x_i, y_i) = \begin{bmatrix} z_r^i \\ z_\beta^i \end{bmatrix} = \begin{bmatrix} \sqrt{(x_i - x_L)^2 + (y_i - y_L)^2} \\ \phi_L - a\tan\left(-\dfrac{y_i - y_L}{x_i - x_L}\right) + \dfrac{\pi}{2} \end{bmatrix} + \gamma_h, \tag{2}$$

where z is the observation vector, (x_i, y_i) the coordinates of the landmarks, x_L, y_L and ϕ_L are the vehicle states defined at the external sensor location and γ_h the zero mean Gaussian white noise. In the case where multiple observations are obtained the

UNCORRECTED PROOF

Fig. 1. Navigation using SLAM. The vehicle builds a relative local map and localizes within this map using dead reckoning information and relative observations of features in the environment. The accuracy of the map is a function of the accuracy of the local map origin and the quality of the kinematics model and relative observations. The local map can be registered to the global map if absolute information becomes available, such as the observation of a beacon at a known position or GPS position information.

observation vector will have the form

$$Z = \begin{bmatrix} z^1 \\ \vdots \\ z^m \end{bmatrix}. \quad (3)$$

Under the SLAM framework the vehicle starts at an unknown position with given uncertainty and obtains measurements of the environment relative to its location. This information is used to incrementally build and maintain a navigation map and to localize with respect to this map. The system will detect new features at the beginning of the mission and when the vehicle explores new areas. Once these features become reliable and stable they are incorporated into the map becoming part of the state vector. The state vector is now given by

$$X = \begin{bmatrix} X_L \\ X_I \end{bmatrix}, \quad X_L = (x_L, y_L, \phi_L)^{\mathrm{T}} \in R^3,$$

$$X_I = (x_1, y_1, \dots, x_N, y_N)^{\mathrm{T}} \in R^{2N}, \quad (4)$$

where $(x, y, \phi)_L$ and $(x, y)_i$ are the states corresponding to the vehicle and N features incorporated into the map, respectively. Since this environment is consider to be static the dynamic model that includes the features is

$$X_L(k+1) = f(X_L(k)) + \gamma, \quad X_I(k+1) = X_I(k). \quad (5)$$

It is important to remark that the landmarks are assumed to be static. Then the Jacobian matrix for the

UNCORRECTED PROOF

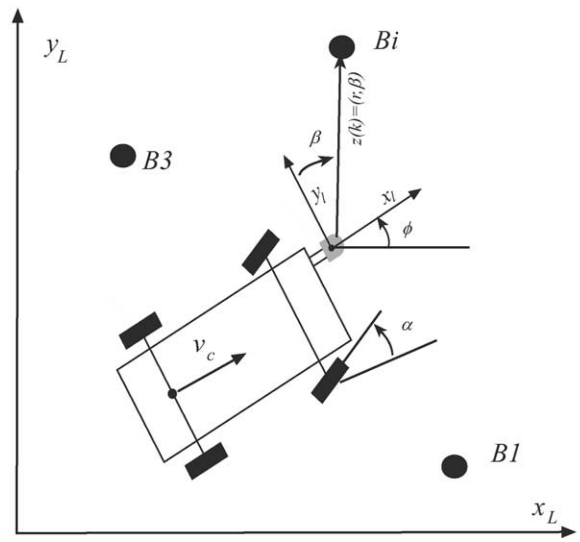

Fig. 2. Vehicle coordinate system.

extended system is

$$\frac{\partial F}{\partial X} = \begin{bmatrix} \dfrac{\partial f}{\partial \tilde{x}_L} & \emptyset \\ \emptyset^{\mathrm{T}} & I \end{bmatrix} = \begin{bmatrix} J_1 & \emptyset \\ \emptyset^{\mathrm{T}} & I \end{bmatrix},$$

$$J_1 \in R^{3\times 3},\quad \emptyset \in R^{3\times N},\quad I \in R^{2N\times 2N}. \tag{6}$$

These models can then be used with a standard EKF algorithm to build and maintain a navigation map of the environment and to track the position of the vehicle. The prediction stage is required to obtain the predicted value of the states X and its error covariance P at time k based on the information available up to time $k-1$,

$$X(k+1,k) = F(X(k,k), u(k)),$$
$$P(k+1,k) = J(k)\cdot P(k,k)\cdot J^{\mathrm{T}}(k) + Q(k). \tag{7}$$

The update stage is a function of the observation model and the error covariance:

$$\begin{aligned} S(k+1) &= H(k+1)\cdot P(k+1,k)\cdot H^{\mathrm{T}}(k+1) \\ &\quad + R(k+1), \\ W(k+1) &= P(k+1,k)\cdot H^{\mathrm{T}}(k+1)\cdot S^{-1}(k+1), \\ \vartheta(k+1) &= Z(k+1) - h(X(k+1,k)), \\ X(k+1,k+1) &= X(k+1,k) + W(k+1)\cdot\vartheta(k+1), \\ P(k+1,k+1) &= P(k+1,k) - W(k+1) \\ &\quad \cdot S(k+1)\cdot W(k+1)^{\mathrm{T}}, \end{aligned} \tag{8}$$

where

$$J(k) = \left.\frac{\partial F}{\partial X}\right|_{(X,u)=(X(k),u(k))},\qquad H(k) = \left.\frac{\partial h}{\partial X}\right|_{X=X(k)} \tag{9}$$

are the Jacobian matrices derived from vectorial functions $F(x,u)$ and $h(x)$ with respect to the state X. R and Q are the error covariance matrices characterizing the noise in the observations and model, respectively.

3. Environment description and feature detection

The navigation map is built with features present in the environment that are detected by external sensors that provide information relative to the position of the vehicle. Recognizable features are essential for SLAM algorithms since they are responsible for bounding the navigation errors. One of the first tasks in the navigation system design is to determine the type of sensor required to obtain a desired localization accuracy in a particular outdoor environment. The most important factor that determines the quality of the map is obviously the accuracy of the relative external sensor. For example, in the case of radar or laser sensors, this is determined by the range and bearing errors obtained when seeing a feature/landmark. These errors are function of the specification of the sensors and the type of feature used. If the shape of the feature is well known a priori, such as the case of artificial landmarks, then the errors can be evaluated and the accuracy of the navigation system can be estimated. A different problem is when the navigation system has to work with natural features. The inspection of the environment can give an idea of the most relevant features that can be detected with a given sensor. The most appropriate sensor for the application will depend on the size of the operating area and environmental conditions. Fig. 1 presents an outdoor environment where trees can be consider one of the most relevant features that a laser range sensor can identify. With larger areas or in environment with fog or dust a different sensor such as radar will be a better choice. Once the sensor is selected then a model to obtain accurate and consistent feature position estimation is required. For example, if the raw return from the laser is used as a measure of a distance to a tree then a significant error can be in-

UNCORRECTED PROOF

Fig. 3. Trees with different shape, size and inclination. The feature detection algorithm needs to consider these type of different trees to accurately determine the position of the feature.

troduced due to the size, shape and inclination of the trunk. This problem is shown in Fig. 3 for various type of trees commonly found in these environments. Any algorithm designed to extract the location of these features needs to consider these problems to increase the accuracy of the feature location process. In this work, a Kalman filter was implemented to track the centre of the trunk by clustering a number of laser observations as representative of the circular surface of the trunk.

3.1. *Feature position determination*

The landmark's position estimation can be improved by evaluating the diameter of the tree trunk. This will also make the observation information more independent of the sensor viewpoint location. The first stage of the process consists of determining the number of consecutive laser returns that belong to the cluster associated to an object, in this case a tree trunk. In the case of working with range and bearing sensors the information returned from a cylindrical objects is shown in Fig. 4. Depending on the angular and range resolution and beam angle, the sensor will return a number of ranges distributed in a semicircle. In Fig. 4, the cylindrical object is detected at four different bearing angles. An observation of the diameter of the feature can be generated using the average range and bearing angle enclosing the cluster of points representing the object:

$$z_D = \Delta\beta \cdot r, \tag{10}$$

where $\Delta\beta$ and r are the angular width and average distance to the object obtained from the laser location, respectively. For the case of a laser returning 361 range and bearing observations distributed in 180°:

$$\Delta\beta = (i_n - i_i) \cdot \frac{\pi}{NR}, \quad NR = 360,$$

$$r = \frac{1}{i_n - i_i + 1} \cdot \sum_{i=i_i}^{i_n} r(i). \tag{11}$$

The indexes i_n to i_i correspond to the first and last beam, respectively, reflected by the object. The measurement z_D is obtained from range and bearing information corrupted by noise. The variance of the observation z_D can then be evaluated

$$\sigma_{z_D}^2 = G \cdot \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_{\Delta\beta}^2 \end{bmatrix} \cdot G^{\mathrm{T}},$$

$$G = \frac{\partial z_D}{\partial (r, \Delta\beta)} = [\,\Delta\beta \quad r\,],$$

$$\sigma_{z_D}^2 = (\Delta\beta)^2 \cdot \sigma_r^2 + r^2 \cdot \sigma_{\Delta\beta}^2. \tag{12}$$

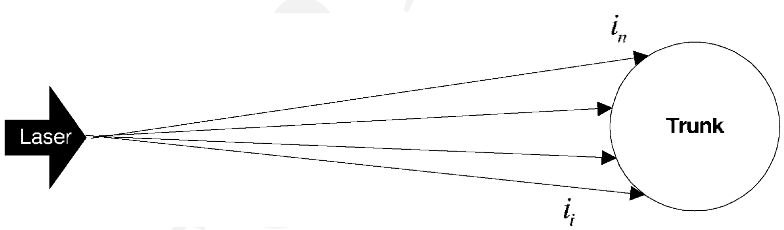

Fig. 4. Laser range finder return information from a cylinder type object.

UNCORRECTED PROOF

In outdoor applications the ranges are in the order of 3–40 m. In this case we have that

$$(\Delta\beta)^2 \cdot \sigma_r^2 \ll r^2 \cdot \sigma_{\Delta\beta}^2 \tag{13}$$

then

$$\sigma_{z_D}^2 \cong r^2 \cdot \sigma_{\Delta\beta}^2. \tag{14}$$

This fact indicates that the correlation between z_D and the range measurement error is weak and can be neglected. Additional noise ω_s is also included to consider the fact that this type of natural features will be in practice not perfectly circular and will have different diameters at different heights. Depending on the vehicle inclination two scans from the same location could generate a slightly different shape for the same object. The complete model with additional noise ω_s is

$$\sigma_{z_D}^2 \cong r^2 \cdot \sigma_{\Delta\beta}^2 + \sigma_{\omega s}^2. \tag{15}$$

Finally, a Kalman filter to track each object is implemented assuming a process model with constant diameter and initial condition generated with the first observation:

$$\dot{D} = 0, \qquad D(t_0) = D_0, \qquad \sigma_{D_0}^2 = E\{D_0 \cdot D_0\} \neq 0. \tag{16}$$

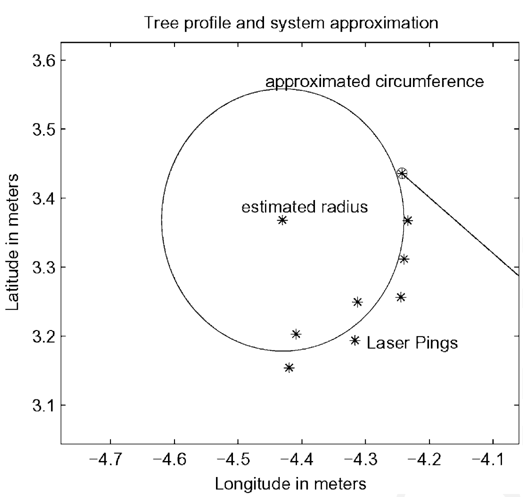

Fig. 5. Tree profile and system approximation. The asterisks indicate the laser range and bearing returns. The filter estimates the radius of the circumference that approximates the trunk of the tree and centre position.

The diameter of each feature is updated after each scan and then used to evaluate the range and bearing to the centre of the trunk. Fig. 5 presents a set of experimental data points, the circumference corresponds to the estimated diameter and the centre of the object estimated by the Kalman filter after a few laser frames.

4. Implementation of SLAM in large environments

In most SLAM applications, the number of vehicle states will be insignificant with respect to the number of landmarks. Under the SLAM framework, the size of the state vector is equal to the number of the vehicle states plus twice the number of landmarks, that is $2N + 3 = M$. In [5], a compressed filter and a map management approach were presented to reduce the real time computation requirement to $\sim \mathrm{O}((2N_a)^2)$, $N_a = M - N_b$ being the number of landmarks in the local area, $N_a \ll M$ and N_b the number of landmarks in the rest of the map. With this approach, the SLAM algorithm becomes extremely efficient while the vehicle remains navigating in this area since the computation complexity is independent of the size of the global map. Still a full update is required when the vehicle leaves the local area. The next section presents new algorithms to reduce the computational requirements of the full update.

4.1. Full SLAM update

The most computational expensive stage of the compressed filter is the global update that needs to be performed after a transition to a new region [5]. This update has a cost of $\sim \mathrm{O}(N_b^2)$. A sub-optimal approach can be used to reduce the computation required for this step.

The nominal global update is

$$\begin{aligned} P_{ab,(k)} &= \phi_{(k-1)} \cdot P_{ab,(0)}, \\ P_{bb,(k)} &= P_{bb,(0)} - P_{ba,(0)} \cdot \psi_{(k-1)} \cdot P_{ab,(0)}. \end{aligned} \tag{17}$$

The evaluation of P_{bb} is computationally very expensive. The change in error covariance for this term is given as

$$\begin{aligned} \Delta P_{bb} &= P_{ba,(0)} \cdot \psi_{(k-1)} \cdot P_{ab,(0)} = W \cdot P_{ab,(0)}, \\ P_{ba,(0)} \cdot \psi_{(k-1)} &= W \in R^{N_b \times N_a}, \quad P_{ab,(0)} \in R^{N_a \times N_b}. \end{aligned} \tag{18}$$

UNCORRECTED PROOF

In order to address this problem the sub-state X_b can be divided into two sub-groups

$$X_b = \begin{bmatrix} X_{b1} \\ X_{b2} \end{bmatrix}, \quad X_{b1} \in R^{N_{b1}}, \quad X_{b2} \in R^{N_{b2}},$$
$$X_b \in R^{N_b}, \quad N_b = N_{b1} + N_{b2}. \tag{19}$$

The associated covariance and the covariance global update matrices are:

$$P_{bb} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21}^{\mathrm{T}} & P_{22} \end{bmatrix},$$
$$\Delta P_{bb} = \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21}^{\mathrm{T}} & \Delta P_{22} \end{bmatrix} = W \cdot P_{ab,(0)}. \tag{20}$$

A conservative global update can be done replacing the matrix ΔP_{bb} by the sub-optimal ΔP_{bb}^*. Now

$$\Delta P_{bb}^* = \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21} & \emptyset \end{bmatrix} = \Delta P_{bb} - \begin{bmatrix} \emptyset & \emptyset \\ \emptyset & \Delta P_{22} \end{bmatrix},$$
$$P_{bb}^* = P_{bb} - \Delta P_{bb}^* = P_{bb} - \Delta P_{bb} + \begin{bmatrix} \emptyset & \emptyset \\ \emptyset & \Delta P_{22} \end{bmatrix}. \tag{21}$$

It can be proved that this update is consistent and does not generate over-confident results [4]. Finally, the sub-matrices that need to be evaluated are P_{11}, P_{12} and P_{21}. The significance of this result is that P_{22} is not evaluated. In general, this matrix will be of high order since it includes the states corresponding to most of the landmarks.

The fundamental problem becomes the selection of the subset X_{b2}. The diagonal of matrix P can be evaluated on-line with low computational cost. By inspecting the diagonal elements of ΔP, we have that many terms are very small compared to the corresponding previous covariance value in the matrix P. This indicates that the new observation does not have a significant information contribution to this particular state. This is used as an indication to select a particular state as belonging to the subset $b2$.

A selection criteria to obtain the partition of the state vector is given by the following set I:

$$I = \{i \backslash \Delta P_{bb}(i,i) < c_1 \cdot P_{bb}(i,i)\}. \tag{22}$$

The evaluation of $\Delta P_{bb}(i,i)$ has a computational cost $\sim \mathrm{O}(N_b)$ (instead of $\sim \mathrm{O}(N_b^2)$ for the evaluation of the complete ΔP_{bb} matrix). Then ΔP^* is evaluated as follows:

$$\Delta P_{bb}^*(i,j) = 0 \quad \forall i,j \backslash i \in I \ \text{ and } \ j \in I,$$
$$\Delta P_{bb}^*(i,j) = \Delta P_{bb}(i,j) \quad \forall i,j \backslash i \notin I \ \text{ or } \ j \notin I. \tag{23}$$

The meaning of the set I is that the gain of information for this group of states is very small. For example, in the case of c_1=1/10,000, it is required about 100 global updates of this 'quality' to be able to obtain a 1% reduction in covariance value. It has to be noted that a global update occurs approximately every hundred or thousands of local updates. With appropriate selection of the constant c_1 the difference between the nominal global full update and the sub-optimal global update will be negligible for practical purposes. Then the update of the sub-matrix ΔP_{22} can be ignored. The total covariance matrix is still consistent since the cross-covariance matrices are updated. The magnitude of the computation saving factor depends on the size of the set I. With appropriate exploration policies, real time mission planning, the computation requirements can be maintained within the bounds of the on-board resources.

4.2. Aided SLAM

One of the fundamental issues in navigation is to be able to use all available information in an optimal manner. Although the SLAM algorithms presented can work in large areas it can also benefit from absolute position information such as GPS. In many applications, it is not possible to obtain GPS information for long periods of time. Nevertheless, at some locations this sensor will be able to report navigation data with an estimated error. This information is usually available in standard GPS receivers. Another source of absolute information can be sporadic detection of landmarks whose position and uncertainty are known. It is important to be able to incorporate this information to improve the localization estimates and at the same time enable the SLAM algorithm to explore and incorporate new features while bounding the absolute pose error with the absolute information.

In order to add this information in a consistent manner some important issues need to be considered:

UNCORRECTED PROOF

- The quality of the models and the relativity navigation information used in the SLAM algorithms could lead to very large innovations errors when the absolute information is fussed. This will occur after long periods of navigation with relative information only, that is performing pure SLAM. A strong correction will make the linearization of the models not valid generating incorrect update of covariances.
- The innovations may not be large but can generate strong updates in the covariance matrices. This can potentially introduce serious numerical errors.

In order to prevent these problems, it is possible to treat a new absolute information as L observations such that the total information introduced become equivalent to a single update. In this case, the filter will perform L updates with the observation value and modified noise covariance. For example, if the quality of the observation is modelled with a noise covariance R, then L sequential observations of quality $K_i \times R$ will be used with $\sum_{i=1}^{L} 1/K_i = 1$. The constant K_i is selected such that the updated position does not violate the linearization assumptions. This will obviously be a function of the particular application. The sequential updates not only generate the same results as the single update but alleviate numerical problems arising from large covariance updates.

Under the full SLAM formulation, the application of sequential updates results in a notable computational demand increase since each update has cost $M \times M$, M being the number of states. However, this problem does not exist with the compressed filter since these updates are implemented as another group of observations in the local area making the cost proportional to the square of the states in the local area.

The sequential updates can also be applied to prevent a large update when about to close a loop, that is, when returning or revisiting a known location. In general terms, when working with nonlinear systems a gradual equivalent update will work better than a single large update. The reason for that is that the Jacobians used to linearize the system are recalculated in each update and used closer to the linearization point.

5. Experimental results

The navigation algorithms presented were tested in the outdoor environment of the type shown in Fig. 1. The vehicle was retrofitted with a laser range sensor and dead reckoning capabilities. A differential real time kinematic (RTK) GPS system capable of delivering position with up to 2 cm accuracy was used. In this experiment, the vehicle traversed different types or terrains and large areas covered by dense foliage where the GPS reported position with different levels of accuracy. This can be seen in Fig. 6 where the GPS were able to report position with 10 cm accuracy for short periods of time.

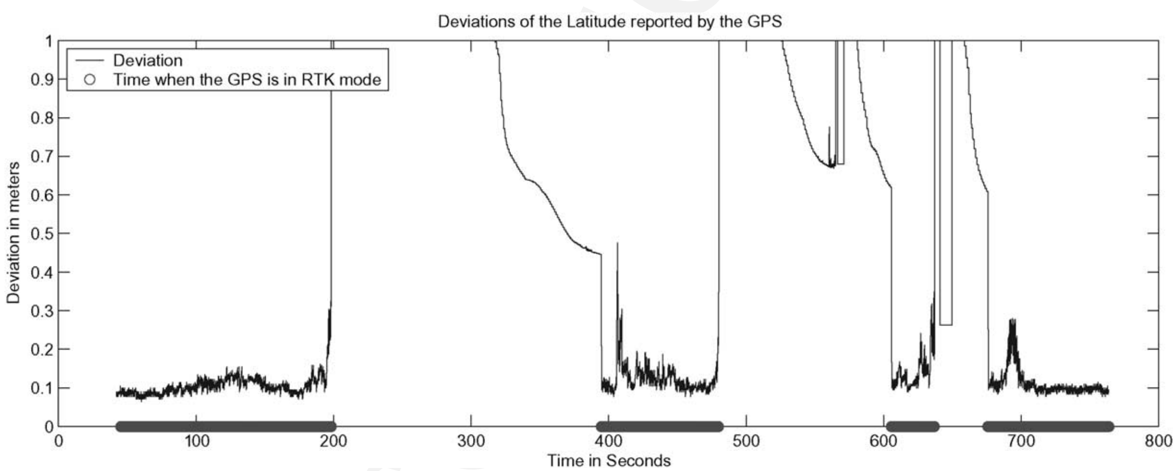

Fig. 6. Deviation reported by the GPS in an open field. The precision is degraded by the presence of trees.

UNCORRECTED PROOF

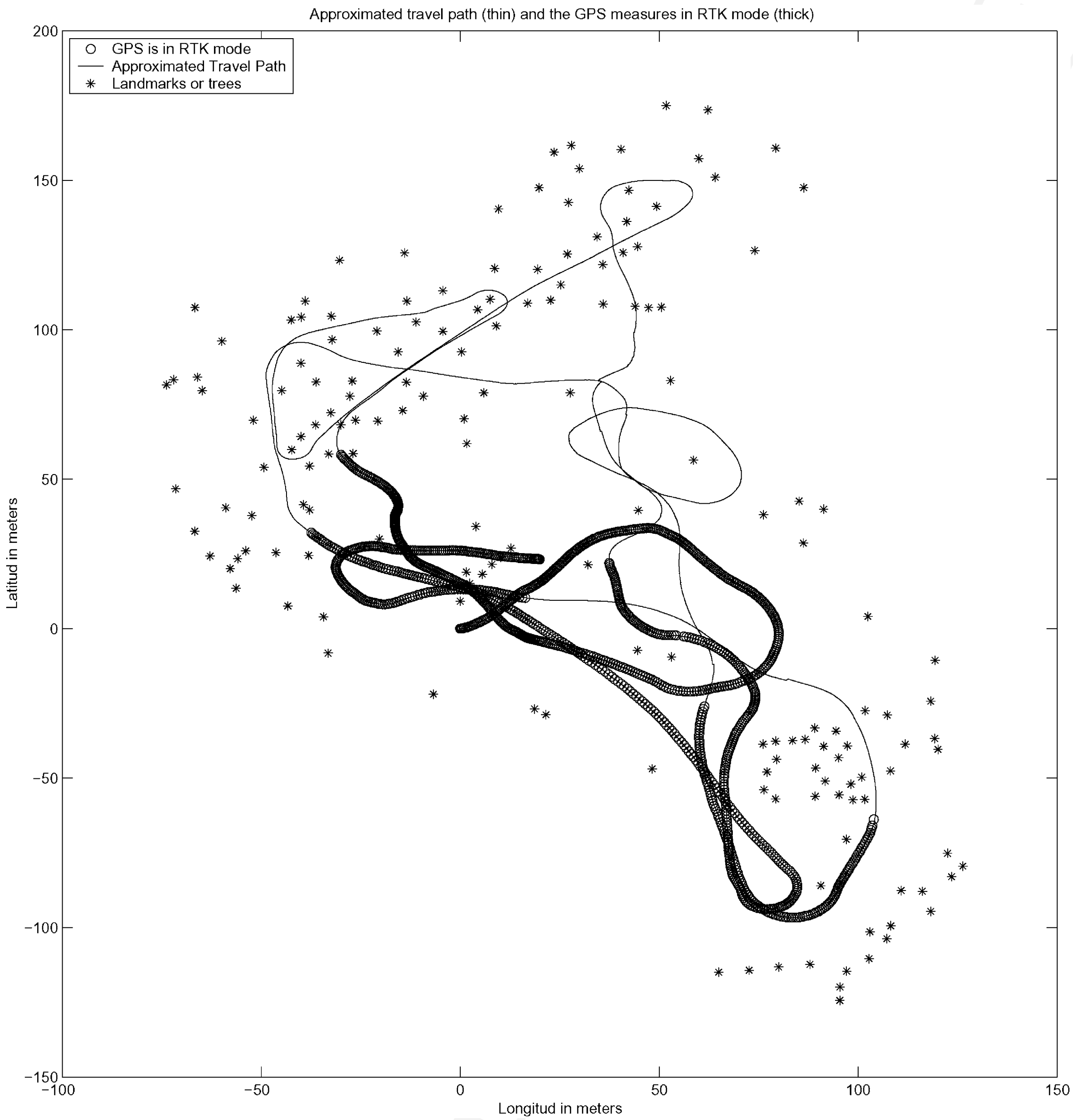

Fig. 7. The approximated travel path and the landmarks or trees present in the field. Thick line shows the moment where the GPS reported an RTK measure.

The regions where GPS reports good quality information are also shown in Fig. 7. This figure presents the estimated trajectory of the vehicle obtained using aided SLAM and the natural features incorporated into the map. It can be seen that GPS information is not available in areas highly populated with features and become more reliable in open areas where features are scare. This is expected since GPS will be affected by dense foliage. At the same time these areas are usually rich in natural features and can be used in the SLAM framework to reduce the navigation errors. Fig. 8 presents the estimated error of the vehicle states

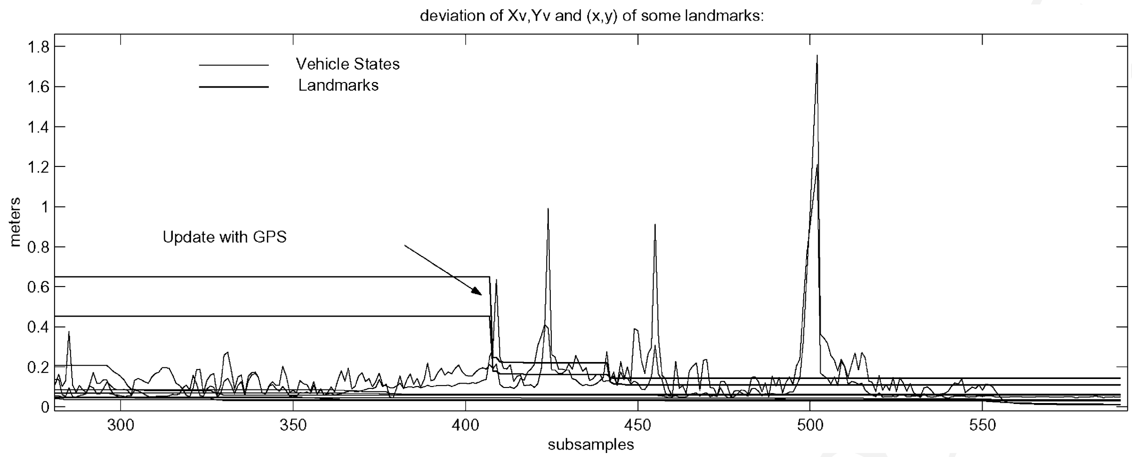

Fig. 8. Deviations of the states of the car and selected landmarks. The vehicle navigates with dead reckoning and relative observations most of the time. When GPS position is incorporated a significant reduction in the covariance of vehicle and landmark is obtained as shown at sample 410.

and selected landmarks. As expected, the vehicle state covariance grows when the vehicle is exploring new regions and decrease when revisiting a known place or when GPS information become available. Fig. 9 shows the behaviour of the algorithm while closing a loop, that is revisiting a known place. It can be seen that at coordinate (37.5, 22), the vehicle start receiving good quality GPS information. At this point, there

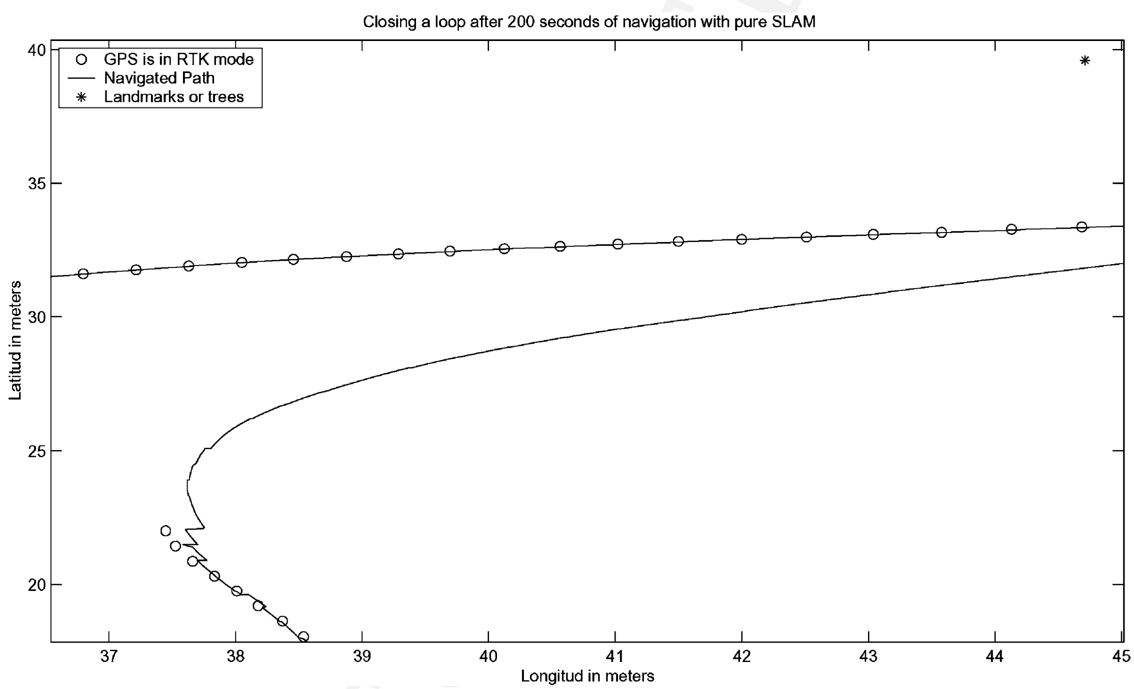

Fig. 9. Closing a loop after 200 s navigating with pure SLAM.

is a significant difference between the SLAM navigation estimation and the absolute GPS information. The GPS information is then incorporated as sequence of updates as shown in the figure. These results demonstrated that the navigation algorithm has enough integrity to work in large areas incorporating relative landmark information and absolute position information.

6. Conclusion

This work presents real time algorithms and implementation issues of SLAM with emphasis to outdoor land vehicle applications. The aspect of feature detection is investigated to reliably detect the predominant features in the environment. Algorithms to simplify the full SLAM update and to incorporate absolute information are also presented. Finally, the algorithms are validated with experimental results obtained in large unstructured outdoor environment.

References

[1] A. Elfes, Occupancy grids: a probabilistic framework for robot perception and navigation, Ph.D. Thesis, Department of Electrical Engineering, Carnegie Mellon University, 1989.

[2] A. Stentz, M. Ollis, S. Scheding, H. Herman, C. Fromme, J. Pedersen, T. Hegardorn, R. McCall, J. Bares, R. Moore, Position measurement for automated mining machinery, in: Proceedings of the International Conference on Field and Service Robotics, August 1999, pp. 293–304.

[3] H.F. Durrant-Whyte, An autonomous guided vehicle for cargo handling applications, International Journal of Robotics Research 15 (5) (1996) 407–441.

[4] J. Guivant, E. Nebot, H.F. Durrant-Whyte, Simultaneous localization and map building using natural features in outdoor environments, in: Proceedings of the Sixth Intelligent Autonomous Systems (IAS), Italy, July 25–28, 2000, pp. 581–586.

[5] J. Guivant, E. Nebot, Optimization of the simultaneous localization and map building algorithm for real time implementation, in: Proceedings of the IEEE Transactions of Robotics and Automation, Vol. 17, 2001.

[6] J.A. Castellanos, J.D. Tardos, Mobile Robot Localization and Map Building: A Multisensor Fusion Approach, Kluwer Academic Publishers, Boston, 1999.

[7] J.J. Leonard, H.J.S. Feder, A computationally efficient method for large-scale concurrent mapping and localization, in: Proceedings of the Ninth International Symposium on Robotics Research, Utah, USA, October 1999, pp. 316–321.

[8] S. Thrun, D. Fox, W. Bugard, Probabilistic mapping of environment by a mobile robot, in: Proceedings of the 1998 IEEE, Belgium, pp. 1546–1551.

[9] J. Leonard, H.F. Durrant-Whyte, Simultaneous map building and localization for an autonomous mobile robot, in: Proceedings of the IEEE International Workshop on Intelligent Robots and Systems, Osaka, Japan, pp. 1442–1447.

[10] S.B. Williams, P. Newman, M.W.M.G. Dissanayake, J. Rosenblatt, H.F. Durrant-Whyte, A decoupled, distributed AUV control architecture, in: Proceedings of the 31st International Symposium on Robotics, Vol. 1, Canadian Federation for Robotics, Ottawa, Ont., 1999, pp. 246–251.

[11] J. Guivant, E. Nebot, S. Baiker, Localization and map building using laser range sensors in outdoor applications, Journal of Robotic Systems 17 (10) (2000) 565–583.

[12] S. Sukkarieh, E. Nebot, H.F. Durrant-Whyte, A high integrity IMU/GPS navigation loop for autonomous land vehicle applications, in: Proceedings of the IEEE Transaction on Robotics and Automation, 1999, pp. 572–578.

[13] S. Scheding, E.M. Nebot, H.F. Durrant-Whyte, High integrity navigation using frequency domain techniques, in: Proceedings of the IEEE Transaction of System Control Technology, 2000, pp. 676–694.

[14] P. Motalier, R. Chatila, Stochastic multi-sensory data fusion for mobile robot location and environmental modelling, in: Proceedings of the Fifth Symposium on Robotics Research, Tokyo, 1989.

Jose Guivant received his Electronic Engineering degree from the Universidad Nacional del Sur (UNS), Bahía Blanca, Argentina, in 1993. Since 1995, he has been with the control group of the Departamento de Ingeniería Eléctrica at the UNS and since 1999, he has been at Australian Centre of Field Robotics at the University of Sydney where at present he is a Ph.D. student. His research interests include nonlinear control systems, autonomous navigation, localization and navigation algorithms, sensor-based map building.

Favio Masson received his Electronic Engineering degree from the Universidad Nacional del Sur (UNS), Bahía Blanca, Argentina, in 1993. Since 1994, he has been with the Control Group of the Departamento de Ingeniería Eléctrica, UNS. Since 1999, he has pursuing the Ph.D. degree in the UNS. He is a Professor in the Departamento de Ingeniería Eléctrica, UNS. His research interests include autonomous navigation, localization algorithms, sensor-based map building and Monte Carlo filters.

UNCORRECTED PROOF

Eduardo Nebot received his Bachelor's degree in Electrical Engineering from the Universidad Nacional del Sur (Argentina) and M.S. and Ph.D. from Colorado State University, USA. He is Associate Professor at the University of Sydney in the Department of Mechanical and Mechatronic Engineering. He has been a faculty member since 1992. His current research contents inertial sensors, GPS and inertial sensing in land vehicle applications, navigation and guidance algorithms; decentralized and distributed navigation, map building and terrain-aiding. He has also been involved in different automation projects in the stevedoring, mining and cargo handling industries. He is currently the Automation Program Manager for the Centre for Mining, Technology and Equipment (CMTE). He is a Senior Member of the Institute of Electrical and Electronic Engineering.

UNCORRECTED PROOF