



ELSEVIER

Int. Journal of Robotics and Autonomous
Systems Vol 40 31 August 2002 pp 79-90

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.

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).

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 fea-

tures. 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

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

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

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

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



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.

134 observation vector will have the form

$$135 \quad Z = \begin{bmatrix} z^1 \\ \vdots \\ z^m \end{bmatrix}. \quad (3)$$

136 Under the SLAM framework the vehicle starts at an
 137 unknown position with given uncertainty and obtains
 138 measurements of the environment relative to its loca-
 139 tion. This information is used to incrementally build
 140 and maintain a navigation map and to localize with
 141 respect to this map. The system will detect new fea-
 142 tures at the beginning of the mission and when the ve-
 143 hicle explores new areas. Once these features become
 144 reliable and stable they are incorporated into the map
 145 becoming part of the state vector. The state vector is

now given by 146

$$147 \quad X = \begin{bmatrix} X_L \\ X_I \end{bmatrix}, \quad X_L = (x_L, y_L, \phi_L)^T \in \mathbb{R}^3, \quad (4)$$

$$148 \quad X_I = (x_1, y_1, \dots, x_N, y_N)^T \in \mathbb{R}^{2N}, \quad 149$$

150 where $(x, y, \phi)_L$ and $(x, y)_i$ are the states correspond-
 151 ing to the vehicle and N features incorporated into the
 152 map, respectively. Since this environment is consider
 153 to be static the dynamic model that includes the fea-
 154 tures is 155

$$156 \quad X_L(k+1) = f(X_L(k)) + \gamma, \quad X_I(k+1) = X_I(k). \quad (5)$$

157 It is important to remark that the landmarks are as-
 158 sumed to be static. Then the Jacobian matrix for the 159

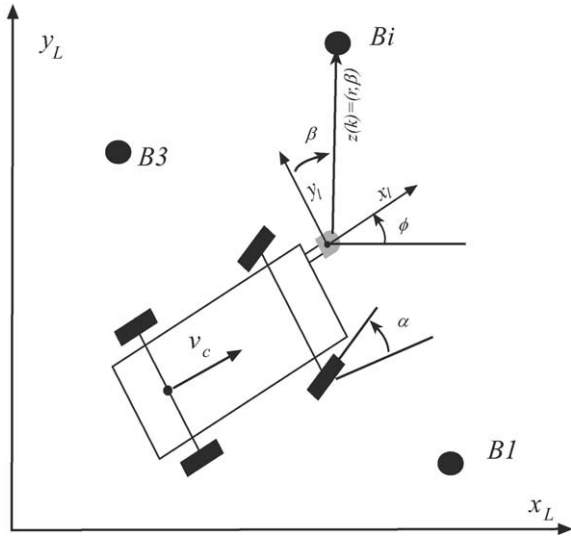


Fig. 2. Vehicle coordinate system.

160 extended system is

$$162 \frac{\partial F}{\partial X} = \begin{bmatrix} \frac{\partial f}{\partial \bar{x}_L} & \emptyset \\ \emptyset^T & I \end{bmatrix} = \begin{bmatrix} J_1 & \emptyset \\ \emptyset^T & I \end{bmatrix},$$

$$163 J_1 \in R^{3 \times 3}, \emptyset \in R^{3 \times N}, I \in R^{2N \times 2N}. \quad (6)$$

164 These models can then be used with a standard EKF
165 algorithm to build and maintain a navigation map of
166 the environment and to track the position of the vehi-
167 cle. The prediction stage is required to obtain the pre-
168 dicted value of the states X and its error covariance
169 P at time k based on the information available up to
170 time $k - 1$,

$$172 X(k+1, k) = F(X(k, k), u(k)),$$

$$173 P(k+1, k) = J(k) \cdot P(k, k) \cdot J^T(k) + Q(k). \quad (7)$$

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

$$177 S(k+1) = H(k+1) \cdot P(k+1, k) \cdot H^T(k+1)$$

$$178 \quad + R(k+1),$$

$$179 W(k+1) = P(k+1, k) \cdot H^T(k+1) \cdot S^{-1}(k+1),$$

$$180 \vartheta(k+1) = Z(k+1) - h(X(k+1, k)),$$

$$181 X(k+1, k+1) = X(k+1, k) + W(k+1) \cdot \vartheta(k+1),$$

$$182 P(k+1, k+1) = P(k+1, k) - W(k+1)$$

$$183 \quad \cdot S(k+1) \cdot W(k+1)^T, \quad (8)$$

where

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

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

3. Environment description and feature detection 192

193 The navigation map is built with features present in
194 the environment that are detected by external sensors
195 that provide information relative to the position of the
196 vehicle. Recognizable features are essential for SLAM
197 algorithms since they are responsible for bounding the
198 navigation errors. One of the first tasks in the naviga-
199 tion system design is to determine the type of sensor
200 required to obtain a desired localization accuracy in
201 a particular outdoor environment. The most important
202 factor that determines the quality of the map is obvi-
203 ously the accuracy of the relative external sensor. For
204 example, in the case of radar or laser sensors, this is
205 determined by the range and bearing errors obtained
206 when seeing a feature/landmark. These errors are func-
207 tion of the specification of the sensors and the type of
208 feature used. If the shape of the feature is well known
209 a priori, such as the case of artificial landmarks, then
210 the errors can be evaluated and the accuracy of the
211 navigation system can be estimated. A different prob-
212 lem is when the navigation system has to work with
213 natural features. The inspection of the environment
214 can give an idea of the most relevant features that can
215 be detected with a given sensor. The most appropri-
216 ate sensor for the application will depend on the size
217 of the operating area and environmental conditions.
218 Fig. 1 presents an outdoor environment where trees
219 can be considered one of the most relevant features that
220 a laser range sensor can identify. With larger areas or
221 in environment with fog or dust a different sensor such
222 as radar will be a better choice. Once the sensor is se-
223 lected then a model to obtain accurate and consistent
224 feature position estimation is required. For example,
225 if the raw return from the laser is used as a measure of
226 a distance to a tree then a significant error can be in-



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.

227 introduced due to the size, shape and inclination of the
 228 trunk. This problem is shown in Fig. 3 for various type
 229 of trees commonly found in these environments. Any
 230 algorithm designed to extract the location of these fea-
 231 tures needs to consider these problems to increase the
 232 accuracy of the feature location process. In this work,
 233 a Kalman filter was implemented to track the centre
 234 of the trunk by clustering a number of laser observa-
 235 tions as representative of the circular surface of the
 236 trunk.

237 3.1. Feature position determination

238 The landmark’s position estimation can be im-
 239 proved by evaluating the diameter of the tree trunk.
 240 This will also make the observation information more
 241 independent of the sensor viewpoint location. The
 242 first stage of the process consists of determining the
 243 number of consecutive laser returns that belong to
 244 the cluster associated to an object, in this case a tree
 245 trunk. In the case of working with range and bearing
 246 sensors the information returned from a cylindrical
 247 objects is shown in Fig. 4. Depending on the angu-
 248 lar and range resolution and beam angle, the sensor
 249 will return a number of ranges distributed in a semi-
 250 circle. In Fig. 4, the cylindrical object is detected at
 251 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° :

$$\Delta\beta = (i_n - i_i) \cdot \frac{\pi}{NR}, \quad NR = 360, \tag{261}$$

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

The indexes i_n to i_i correspond to the first and last
 beam, respectively, reflected by the object. The mea-
 surement z_D is obtained from range and bearing in-
 formation corrupted by noise. The variance of the ob-
 servation z_D can then be evaluated

$$\sigma_{z_D}^2 = G \cdot \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_{\Delta\beta}^2 \end{bmatrix} \cdot G^T, \tag{269}$$

$$G = \frac{\partial z_D}{\partial(r, \Delta\beta)} = [\Delta\beta \quad r], \tag{270}$$

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

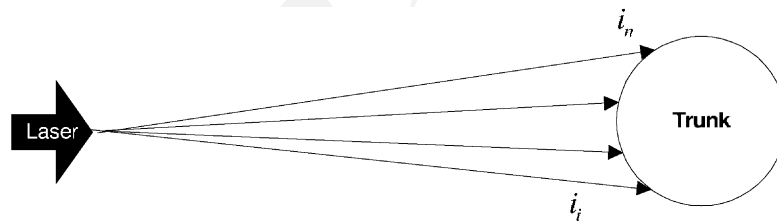


Fig. 4. Laser range finder return information from a cylinder type object.

272 In outdoor applications the ranges are in the order of
 273 3–40 m. In this case we have that

$$274 \quad (\Delta\beta)^2 \cdot \sigma_r^2 \ll r^2 \cdot \sigma_{\Delta\beta}^2 \quad (13)$$

275 then

$$276 \quad \sigma_{z_D}^2 \cong r^2 \cdot \sigma_{\Delta\beta}^2. \quad (14)$$

277 This fact indicates that the correlation between z_D and
 278 the range measurement error is weak and can be ne-
 279 glected. Additional noise ω_s is also included to con-
 280 sider the fact that this type of natural features will be
 281 in practice not perfectly circular and will have differ-
 282 ent diameters at different heights. Depending on the
 283 vehicle inclination two scans from the same location
 284 could generate a slightly different shape for the same
 285 object. The complete model with additional noise ω_s
 286 is

$$287 \quad \sigma_{z_D}^2 \cong r^2 \cdot \sigma_{\Delta\beta}^2 + \sigma_{\omega_s}^2. \quad (15)$$

288 Finally, a Kalman filter to track each object is imple-
 289 mented assuming a process model with constant di-
 290 ameter and initial condition generated with the first
 292 observation:

$$293 \quad \dot{D} = 0, \quad D(t_0) = D_0, \quad \sigma_{D_0}^2 = E\{D_0 \cdot D_0\} \neq 0. \quad (16)$$

294

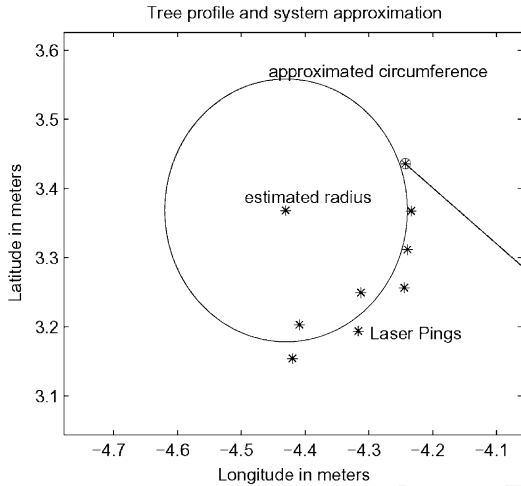


Fig. 5. Tree profile and system approximation. The asterisks indicate the laser range and bearing returns. The filter estimates the radius of the circumference that approximates the trunk of the tree and centre position.

295 The diameter of each feature is updated after each scan
 296 and then used to evaluate the range and bearing to the
 297 centre of the trunk. Fig. 5 presents a set of experimen-
 298 tal data points, the circumference corresponds to the
 299 estimated diameter and the centre of the object esti-
 300 mated by the Kalman filter after a few laser frames.

4. Implementation of SLAM in large environments 301

302 In most SLAM applications, the number of vehicle
 303 states will be insignificant with respect to the number
 304 of landmarks. Under the SLAM framework, the size
 305 of the state vector is equal to the number of the vehi-
 306 cle states plus twice the number of landmarks, that is
 307 $2N + 3 = M$. In [5], a compressed filter and a map
 308 management approach were presented to reduce the
 309 real time computation requirement to $\sim O((2N_a)^2)$,
 310 $N_a = M - N_b$ being the number of landmarks in the
 311 local area, $N_a \ll M$ and N_b the number of landmarks
 312 in the rest of the map. With this approach, the SLAM
 313 algorithm becomes extremely efficient while the vehi-
 314 cle remains navigating in this area since the computa-
 315 tion complexity is independent of the size of the global
 316 map. Still a full update is required when the vehicle
 317 leaves the local area. The next section presents new
 318 algorithms to reduce the computational requirements
 319 of the full update.

4.1. Full SLAM update 320

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

The nominal global update is

$$327 \quad P_{ab,(k)} = \phi_{(k-1)} \cdot P_{ab,(0)}, \quad 329$$

$$328 \quad P_{bb,(k)} = P_{bb,(0)} - P_{ba,(0)} \cdot \psi_{(k-1)} \cdot P_{ab,(0)}. \quad (17) \quad 330$$

331 The evaluation of P_{bb} is computationally very ex-
 332 pensive. The change in error covariance for this term
 333 is given as

$$334 \quad \Delta P_{bb} = P_{ba,(0)} \cdot \psi_{(k-1)} \cdot P_{ab,(0)} = W \cdot P_{ab,(0)}, \quad 335$$

$$336 \quad P_{ba,(0)} \cdot \psi_{(k-1)} = W \in R^{N_b \times N_a}, \quad P_{ab,(0)} \in R^{N_a \times N_b}. \quad 337$$

$$338 \quad (18) \quad 339$$

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

$$341 \quad X_b = \begin{bmatrix} X_{b1} \\ X_{b2} \end{bmatrix}, \quad X_{b1} \in R^{N_{b1}}, \quad X_{b2} \in R^{N_{b2}},$$

$$342 \quad X_b \in R^{N_b}, \quad N_b = N_{b1} + N_{b2}. \quad (19)$$

343 The associated covariance and the covariance global
 344 update matrices are:

$$346 \quad P_{bb} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21}^T & P_{22} \end{bmatrix},$$

$$347 \quad \Delta P_{bb} = \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21}^T & \Delta P_{22} \end{bmatrix} = W \cdot P_{ab,(0)}. \quad (20)$$

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

$$351 \quad \Delta P_{bb}^* = \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21} & \emptyset \end{bmatrix} = \Delta P_{bb} - \begin{bmatrix} \emptyset & \emptyset \\ \emptyset & \Delta P_{22} \end{bmatrix},$$

$$352 \quad P_{bb}^* = P_{bb} - \Delta P_{bb}^* = P_{bb} - \Delta P_{bb} + \begin{bmatrix} \emptyset & \emptyset \\ \emptyset & \Delta P_{22} \end{bmatrix}.$$

$$353 \quad (21)$$

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

361 The fundamental problem becomes the selection of
 362 the subset X_{b2} . The diagonal of matrix P can be evalu-
 363 ated on-line with low computational cost. By inspect-
 364 ing the diagonal elements of ΔP , we have that many
 365 terms are very small compared to the corresponding
 366 previous covariance value in the matrix P . This indi-
 367 cates that the new observation does not have a signif-
 368 icant information contribution to this particular state.
 369 This is used as an indication to select a particular state
 370 as belonging to the subset $b2$.

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

$$373 \quad I = \{i \mid \Delta P_{bb}(i, i) < c_1 \cdot P_{bb}(i, i)\}. \quad (22)$$

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

$$379 \quad \Delta P_{bb}^*(i, j) = 0 \quad \forall i, j \mid i \in I \text{ and } j \in I,$$

$$380 \quad \Delta P_{bb}^*(i, j) = \Delta P_{bb}(i, j) \quad \forall i, j \mid i \notin I \text{ or } j \notin I. \quad (23)$$

The meaning of the set I is that the gain of informa-
 tion 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 hun-
 dred 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 applica-
 tions, 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 avail-
 able in standard GPS receivers. Another source of ab-
 solute information can be sporadic detection of land-
 marks 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 incor-
 porate new features while bounding the absolute pose
 error with the absolute information.

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

- 419 • The quality of the models and the relativity navigation
420 information used in the SLAM algorithms
421 could lead to very large innovations errors when the
422 absolute information is fused. This will occur after
423 long periods of navigation with relative information
424 only, that is performing pure SLAM. A strong cor-
425 rection will make the linearization of the models not
426 valid generating incorrect update of covariances.
- 427 • The innovations may not be large but can generate
428 strong updates in the covariance matrices. This can
429 potentially introduce serious numerical errors.

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

445 Under the full SLAM formulation, the application
446 of sequential updates results in a notable computa-
447 tional demand increase since each update has cost

448 $M \times M$, M being the number of states. However, this
449 problem does not exist with the compressed filter since
450 these updates are implemented as another group of
451 observations in the local area making the cost propor-
452 tional to the square of the states in the local area.

453 The sequential updates can also be applied to pre-
454 vent a large update when about to close a loop, that
455 is, when returning or revisiting a known location. In
456 general terms, when working with nonlinear systems a
457 gradual equivalent update will work better than a sin-
458 gle large update. The reason for that is that the Jaco-
459 bians used to linearize the system are recalculated in
460 each update and used closer to the linearization point.

461 5. Experimental results

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

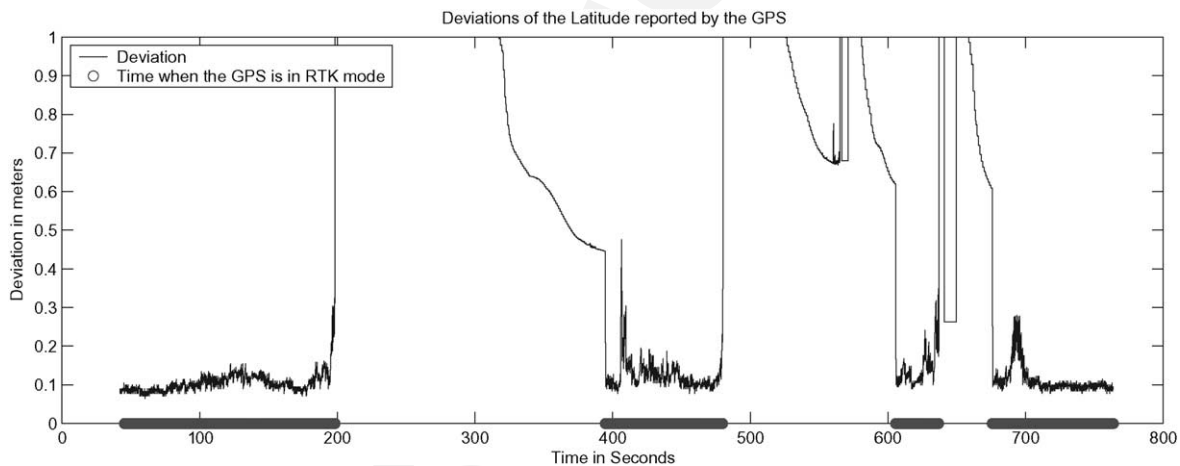


Fig. 6. Deviation reported by the GPS in an open field. The precision is degraded by the presence of trees.

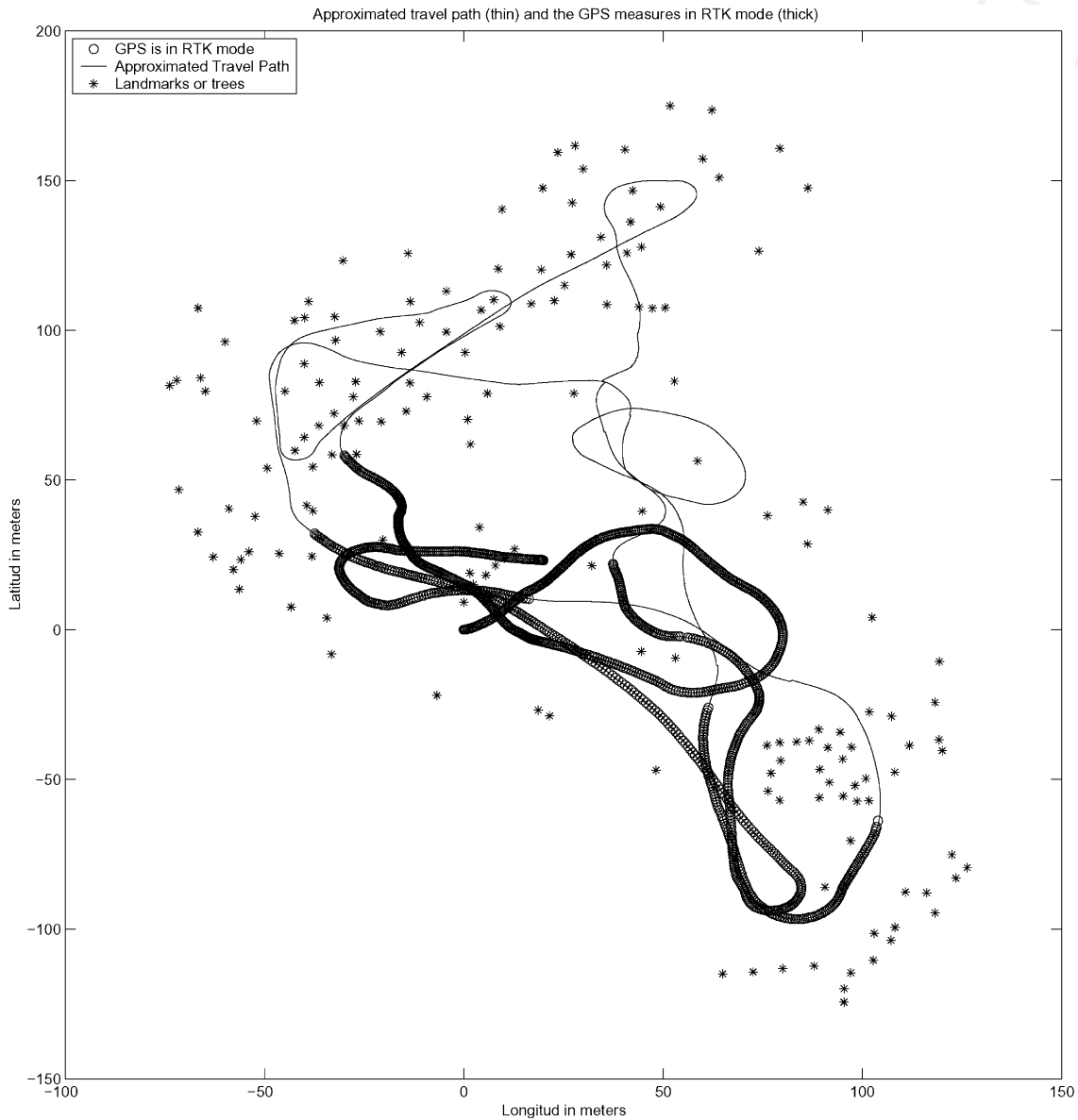


Fig. 7. The approximated travel path and the landmarks or trees present in the field. Thick line shows the moment where the GPS reported an RTK measure.

474 The regions where GPS reports good quality infor-
 475 mation are also shown in Fig. 7. This figure presents
 476 the estimated trajectory of the vehicle obtained us-
 477 ing aided SLAM and the natural features incorporated
 478 into the map. It can be seen that GPS information is
 479 not available in areas highly populated with features

and become more reliable in open areas where fea- 480
 tures are scarce. This is expected since GPS will be af- 481
 fected by dense foliage. At the same time these areas 482
 are usually rich in natural features and can be used in 483
 the SLAM framework to reduce the navigation errors. 484
 Fig. 8 presents the estimated error of the vehicle states 485

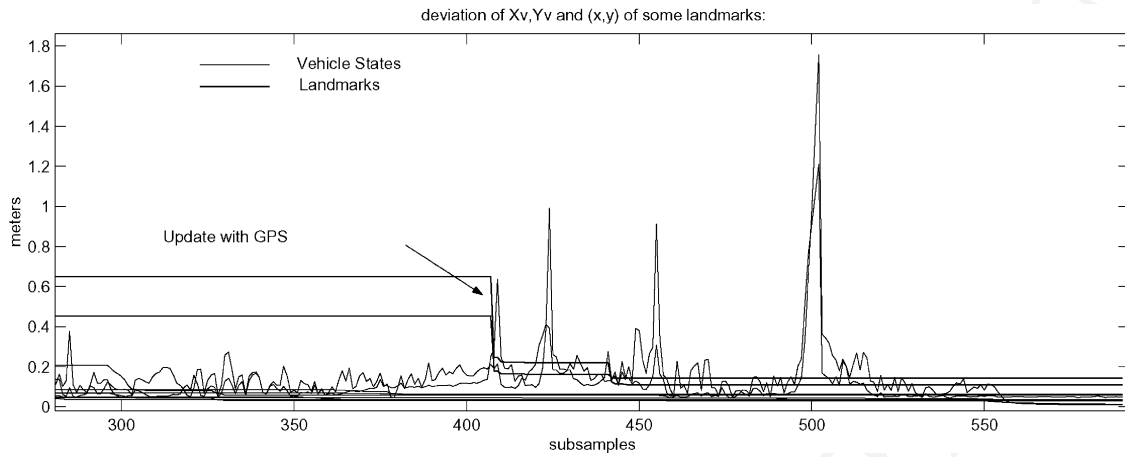


Fig. 8. Deviations of the states of the car and selected landmarks. The vehicle navigates with dead reckoning and relative observations most of the time. When GPS position is incorporated a significant reduction in the covariance of vehicle and landmark is obtained as shown at sample 410.

486 and selected landmarks. As expected, the vehicle state
 487 covariance grows when the vehicle is exploring new
 488 regions and decrease when revisiting a known place
 489 or when GPS information become available. Fig. 9

shows the behaviour of the algorithm while closing a
 490 loop, that is revisiting a known place. It can be seen
 491 that at coordinate (37.5, 22), the vehicle start receiv-
 492 ing good quality GPS information. At this point, there
 493

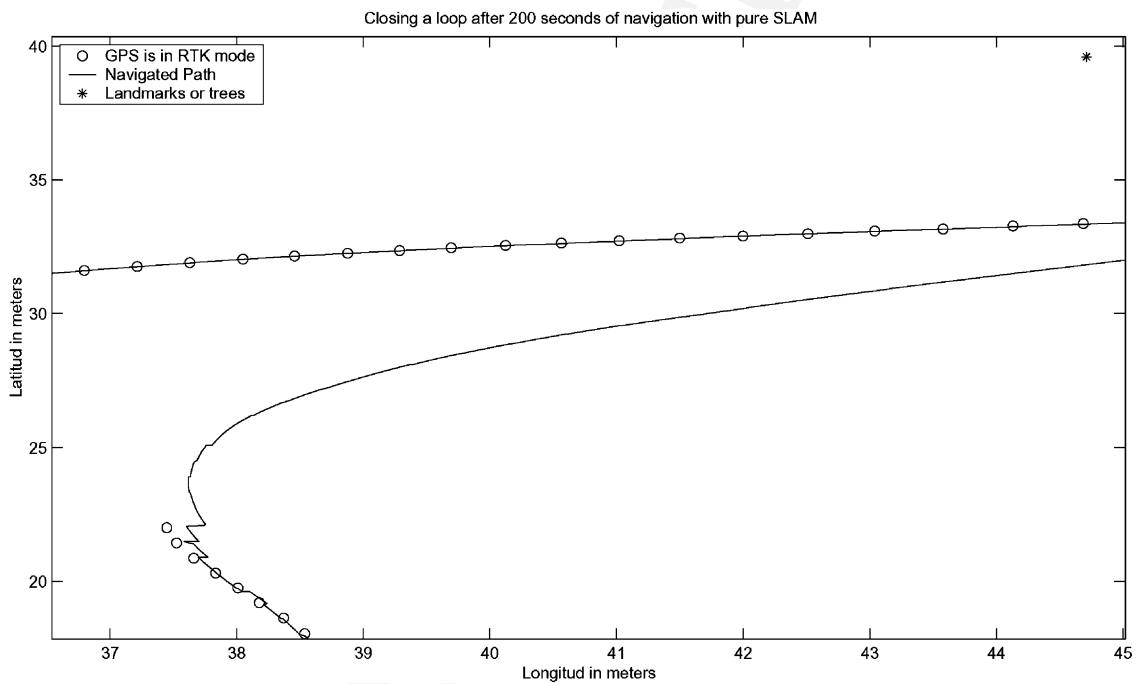


Fig. 9. Closing a loop after 200s navigating with pure SLAM.

494 is a significant difference between the SLAM naviga-
 495 tion estimation and the absolute GPS information. The
 496 GPS information is then incorporated as sequence of
 497 updates as shown in the figure. These results demon-
 498 strated that the navigation algorithm has enough in-
 499 tegrity to work in large areas incorporating relative
 500 landmark information and absolute position informa-
 501 tion.

502 6. Conclusion

503 This work presents real time algorithms and imple-
 504 mentation issues of SLAM with emphasis to outdoor
 505 land vehicle applications. The aspect of feature detec-
 506 tion is investigated to reliably detect the predominant
 507 features in the environment. Algorithms to simplify the
 508 full SLAM update and to incorporate absolute infor-
 509 mation are also presented. Finally, the algorithms are
 510 validated with experimental results obtained in large
 511 unstructured outdoor environment.

512 References

- 513 [1] A. Elfes, Occupancy grids: a probabilistic framework for
 514 robot perception and navigation, Ph.D. Thesis, Department
 515 of Electrical Engineering, Carnegie Mellon University,
 516 1989.
- 517 [2] A. Stentz, M. Ollis, S. Scheding, H. Herman, C. Fromme,
 518 J. Pedersen, T. Hegardorn, R. McCall, J. Bares, R. Moore,
 519 Position measurement for automated mining machinery, in:
 520 Proceedings of the International Conference on Field and
 521 Service Robotics, August 1999, pp. 293–304.
- 522 [3] H.F. Durrant-Whyte, An autonomous guided vehicle for
 523 cargo handling applications, *International Journal of Robotics
 524 Research* 15 (5) (1996) 407–441.
- 525 [4] J. Guivant, E. Nebot, H.F. Durrant-Whyte, Simultaneous
 526 localization and map building using natural features in
 527 outdoor environments, in: Proceedings of the Sixth Intelligent
 528 Autonomous Systems (IAS), Italy, July 25–28, 2000,
 529 pp. 581–586.
- 530 [5] J. Guivant, E. Nebot, Optimization of the simultaneous
 531 localization and map building algorithm for real time
 532 implementation, in: Proceedings of the IEEE Transactions of
 533 Robotics and Automation, Vol. 17, 2001.
- 534 [6] J.A. Castellanos, J.D. Tardos, *Mobile Robot Localization
 535 and Map Building: A Multisensor Fusion Approach*, Kluwer
 536 Academic Publishers, Boston, 1999.
- 537 [7] J.J. Leonard, H.J.S. Feder, A computationally efficient
 538 method for large-scale concurrent mapping and localization,
 539 in: Proceedings of the Ninth International Symposium on
 540 Robotics Research, Utah, USA, October 1999, pp. 316–321.
- [8] S. Thrun, D. Fox, W. Burgard, Probabilistic mapping of
 541 environment by a mobile robot, in: Proceedings of the 1998
 542 IEEE, Belgium, pp. 1546–1551. 543
- [9] J. Leonard, H.F. Durrant-Whyte, Simultaneous map building
 544 and localization for an autonomous mobile robot, in:
 545 Proceedings of the IEEE International Workshop on
 546 Intelligent Robots and Systems, Osaka, Japan, pp. 1442–1447. 547
- [10] S.B. Williams, P. Newman, M.W.M.G. Dissanayake, J.
 548 Rosenblatt, H.F. Durrant-Whyte, A decoupled, distributed
 549 AUV control architecture, in: Proceedings of the 31st
 550 International Symposium on Robotics, Vol. 1, Canadian
 551 Federation for Robotics, Ottawa, Ont., 1999, pp. 246–251. 552
- [11] J. Guivant, E. Nebot, S. Baiker, Localization and map building
 553 using laser range sensors in outdoor applications, *Journal of
 554 Robotic Systems* 17 (10) (2000) 565–583. 555
- [12] S. Sukkarieh, E. Nebot, H.F. Durrant-Whyte, A high integrity
 556 IMU/GPS navigation loop for autonomous land vehicle
 557 applications, in: Proceedings of the IEEE Transaction on
 558 Robotics and Automation, 1999, pp. 572–578. 559
- [13] S. Scheding, E.M. Nebot, H.F. Durrant-Whyte, High
 560 integrity navigation using frequency domain techniques, in:
 561 Proceedings of the IEEE Transaction of System Control
 562 Technology, 2000, pp. 676–694. 563
- [14] P. Motalier, R. Chatila, Stochastic multi-sensory data fusion
 564 for mobile robot location and environmental modelling, in:
 565 Proceedings of the Fifth Symposium on Robotics Research,
 566 Tokyo, 1989. 567



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.



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.

568
569
570
571
572
573
574
575

UNCORRECTED PROOF