Robocentric map joining: Improving the consistency of EKF-SLAM

J.A. Castellanos, R. Martinez-Cantin, J.D. Tardós, J. Neira*

Dept. Informática e Ingeniería de Sistemas, Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, María de Luna 1, E-50018 Zaragoza, Spain

Abstract

In this paper[1] we study the Extended Kalman Filter approach to simultaneous localization and mapping (EKF-SLAM), describing its known properties and limitations, and concentrate on the filter consistency issue. We show that linearization of the inherent nonlinearities of both the vehicle motion and the sensor models frequently drives the solution of the EKF-SLAM out of consistency, specially in those situations where uncertainty surpasses a certain threshold. We propose a mapping algorithm, *Robocentric Map Joining*, which improves consistency of the EKF-SLAM algorithm by limiting the level of uncertainty in the continuous evolution of the stochastic map: (1) by building a sequence of independent local maps, and (2) by using a robot centered representation of each local map. Simulations and a large-scale indoor/outdoor experiment validate the proposed approach.

1. Introduction

During the last decade, the robotics literature has been populated with scientific work on the problem of simultaneous localization and mapping (SLAM). Basically, SLAM is the problem of determining the position and heading of a vehicle moving through an unknown environment and, concurrently, learning useful information from the surroundings taking into account sensor errors. The most popular approach to SLAM dates back to the seminal work of Smith et al. [1] where the idea of representing the structure of the navigation area in a discrete-time state-space framework was originally presented. They introduced the concept of *stochastic map* and they developed a rigorous solution to the SLAM problem using an Extended Kalman Filter (EKF) perspective [2].

The EKF-SLAM approach is characterized by the existence of a discrete-time augmented state vector, composed of the location of the vehicle and the location of the map elements, recursively estimated from the available sensor observations gathered at time k, and a model of the vehicle motion, between time steps $k-1$ and k. Within this framework, uncertainty is represented by probability density functions (pdfs) associated to the state vector, the motion model and the sensor observations. It is assumed that recursive propagation of the mean and the covariance of those pdfs conveniently approximates the optimal solution of this estimation problem. Many successful implementations of this approach have been reported in indoor [11], outdoor [13], underwater [15] and airborne [14] applications.

The time and memory requirements of the basic EKF-SLAM approach result from the cost of maintaining the full covariance matrix, which is $O(n^2)$ where n is the number of features in the map. Many recent efforts have concentrated on reducing the computational complexity of SLAM in large environments [16,13,10,17]. However, only recently, the consistency issues of the EKF-SLAM algorithm have attracted the attention of the research community. Dissanayake et al. [3] proved three important convergence properties of the EKF-SLAM: (1) the determinant of any submatrix of the map

* Corresponding author.
E-mail addresses: jacaste@unizar.es (J.A. Castellanos), rmcantin@unizar.es (R. Martinez-Cantin), tardos@unizar.es (J.D. Tardós), jneira@unizar.es (J. Neira).
[1] Preliminary work was presented at the 2004 Intelligent Autonomous Vehicles Conference, Lisbon, Portugal.

covariance matrix decreases monotonically as observations are successively made; (2) in the limit as the number of observations increases, the landmark estimates become fully correlated; and (3) in the limit, the covariance associated with any single landmark location estimate reaches a lower bound determined only by the initial covariance in the vehicle location estimate at the time of the first sighting of the first landmark. However, it is important to note that these theoretical results only refer to the evolution of the covariance matrices computed by the EKF in the ideal linear case. They overlook the fact that, given that SLAM is a nonlinear problem, there is no guarantee that the computed covariances will match the actual estimation errors, which is the true SLAM consistency issue first pointed out by Julier and Uhlmann [4] and confirmed experimentally by Castellanos et al. [5].

The classical EKF-SLAM linearizes both the motion and sensor models by using a first-order Taylor series expansion around the best available estimated state-vector, therefore, both the bias and the level of uncertainty in the estimated state-vector influence the accuracy of linearization. In the last few years, some works have been reported which propose either alternative linearization techniques [6,7] or even non-parametric approaches [8,9] to avoid those difficulties.

In this paper we show that linearization errors lead to inconsistent estimates well before the computational problem arises. The main contribution of the paper is the formulation of the *Robocentric Map Joining* approach, whose advantages are threefold:

- It addresses the SLAM problem by building independent local maps of limited size, using the technique first proposed in [10]. This technique was shown to greatly reduce the computational cost of SLAM. Here we show that, as the uncertainty inside a local map is bounded, the linearization errors are also reduced.
- In standard SLAM, the map is built using an absolute representation, i.e. with respect to an external reference frame B. After a certain time, the vehicle and the features currently observed have a growing absolute uncertainty that propagates to the measurement equations introducing errors. Here we build each local map using a *robot centered* representation [5], i.e. relative to a reference frame R attached to the vehicle. Except during loop closing, the features currently observed have an uncertainty in the order of the sensor error, which is much smaller. This also results in a reduction of the linearization errors.
- When the robot moves, its new pose is usually predicted by composing the old pose with the motion measured by odometry, which is frequently the least precise sensor. The new uncertainty is then computed by linearizing the composition around the predicted value. In *Robocentric Map Joining* we delay the composition until the map and the motion have been refined using new observations of the environment. This results in a better linearization point for the composition.

The rest of the paper is structured as follows. In Section 2 we analyze the linearizations which appear in the classical EKF-SLAM algorithm, both in filter prediction and filter update steps. Section 3 introduces a rigorous definition of consistency [2] and discusses the inconsistency of the EKF-SLAM algorithm in certain noisy situations. The reformulation of the classical EKF-SLAM, using a robot centered representation is presented in Section 4. Then, in Section 5 we propose the *Robocentric Map Joining* algorithm, which improves the consistency of the classical EKF-SLAM algorithm. Finally, Section 6 describes both simulation and mixed indoor/outdoor real experiments to validate the proposed algorithm.

2. Linearizations in the classical EKF-SLAM algorithm

In the probabilistic state-space approach to SLAM, the vehicle R and a set of environment features $\mathcal{F} = \{F_1, \ldots, F_n\}$ are represented by a stochastic state vector $\mathbf{x}^B$ with estimated mean $\hat{\mathbf{x}}^B$ and estimated error covariance $\mathbf{P}^B$:

$$\hat{\mathbf{x}}^B = \begin{bmatrix} \hat{\mathbf{x}}_R^B \\ \hat{\mathbf{x}}_{\mathcal{F}}^B \end{bmatrix}; \qquad \mathbf{P}^B = \begin{bmatrix} \mathbf{P}_R^B & \mathbf{P}_{R\mathcal{F}}^B \\ \mathbf{P}_{\mathcal{F}R}^B & \mathbf{P}_{\mathcal{F}}^B \end{bmatrix} \qquad (1)$$

where $\hat{\mathbf{x}}_R^B$ is the estimated location of the vehicle with respect to (wrt) a base reference frame B, $\hat{\mathbf{x}}_{\mathcal{F}}^B$ is the estimated location of the features also wrt B, $\mathbf{P}_R^B$ is the estimated error covariance of the location of R, $\mathbf{P}_{\mathcal{F}}^B$ is the estimated error covariance of the location of the features, and finally, $\mathbf{P}_{R\mathcal{F}}^B$ represents the cross-covariance between the different elements of the state vector.

If a reference external to the vehicle is used as base reference, the vehicle location must be initialized with the corresponding nonzero uncertainty. A common misconception is that this nonzero initial level of uncertainty in the vehicle location may improve map consistency. In contrast, our experiments will show that this quickly results in optimistic covariance values due to linearization errors. For this reason, we use the vehicle location before the first motion (at step $k = 0$) as the base reference ($B = R_0$). Thus, the map can be initialized with zero covariance for the vehicle location: $\hat{\mathbf{x}}_0^B = (0, 0, 0)^T$, $\mathbf{P}_0^B = \mathbf{0}$. Our results show that this improves the consistency of the EKF-SLAM algorithm.

2.1. The prediction step

When the vehicle moves from position at time step $k - 1$ to position at time step k, its location is predicted as follows:

$$\mathbf{x}_{R_{k|k-1}}^B = \mathbf{f}_k(\mathbf{x}_{R_{k-1}}^B, \mathbf{x}_{R_k}^{R_{k-1}})$$
$$= \mathbf{x}_{R_{k-1}}^B \oplus \mathbf{x}_{R_k}^{R_{k-1}} \qquad (2)$$

where the uncertain displacement $\mathbf{x}_{R_k}^{R_{k-1}}$ is estimated by odometry and assumed to be corrupted by zero mean white Gaussian noise, $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$. Note that due to the transformation composition $\oplus$, a nonlinear prediction model is formulated. Thus, a *first* linearization, around the estimated values $\hat{\mathbf{x}}_{R_{k-1}}^B$ and $\hat{\mathbf{x}}_{R_k}^{R_{k-1}}$ using the appropriate Jacobians

(see Appendix), is required:

$$\hat{\mathbf{x}}_{k|k-1}^{B} = \begin{bmatrix} \hat{\mathbf{x}}_{R_{k-1}}^{B} \oplus \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \\ \hat{\mathbf{x}}_{F_1,k-1}^{B} \\ \vdots \\ \hat{\mathbf{x}}_{F_m,k-1}^{B} \end{bmatrix}$$

$$\mathbf{P}_{k|k-1}^{B} \simeq \mathbf{J}_1 \mathbf{P}_{k-1}^{B} \mathbf{J}_1^{\mathrm{T}} + \mathbf{J}_2 \mathbf{Q}_k \mathbf{J}_2^{\mathrm{T}} \qquad (3)$$

where:

$$\mathbf{J}_1 = \begin{bmatrix} \mathbf{J}_{1\oplus}\left\{ \hat{\mathbf{x}}_{R_{k-1}}^{B}, \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \right\} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & & \vdots \\ \vdots & & \ddots & \\ \mathbf{0} & \cdots & & \mathbf{I} \end{bmatrix}$$

$$\mathbf{J}_2 = \begin{bmatrix} \mathbf{J}_{2\oplus}\left\{ \hat{\mathbf{x}}_{R_{k-1}}^{B}, \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \right\} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

where $\mathbf{J}_{1\oplus}$ and $\mathbf{J}_{2\oplus}$ are the Jacobians of the transformation composition (see Appendix).

2.2. The update step

At step k an onboard sensor obtains a partial measurement $\mathbf{z}_k$ of the environment features and is related to the state by a nonlinear function $\mathbf{h}_k$:

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k^{B}, \mathbf{x}_{\mathcal{E}_k}^{R_k}) \qquad (4)$$

where $\mathbf{x}_{\mathcal{E}_k}^{R_k}$ represents the set of uncertain observations with respect to R_k, and corrupted by zero mean white Gaussian noise, $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$.

A *second* linearization, this time around the current map prediction $\hat{\mathbf{x}}_{k|k-1}^{B}$, yields:

$$\mathbf{z}_k \simeq \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}^{B}, \hat{\mathbf{x}}_{\mathcal{E}_k}^{R_k}) + \mathbf{H}_k(\mathbf{x}_k^{B} - \hat{\mathbf{x}}_{k|k-1}^{B}) + \mathbf{G}_k(\mathbf{x}_{\mathcal{E}_k}^{R_k} - \hat{\mathbf{x}}_{\mathcal{E}_k}^{R_k})$$

$$\mathbf{H}_k = \left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_k^{B}} \right|_{(\hat{\mathbf{x}}_{k|k-1}^{B}, \hat{\mathbf{x}}_{\mathcal{E}_k}^{R_k})} = \begin{bmatrix} \mathbf{H}_{R_k} & \mathbf{0} & \cdots & \mathbf{H}_{F_k} \cdots & \mathbf{0} \end{bmatrix}$$

$$\mathbf{H}_{R_k} = \left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_{R_k}^{B}} \right|_{(\hat{\mathbf{x}}_{k|k-1}^{B}, \hat{\mathbf{x}}_{\mathcal{E}_k}^{R_k})} ; \quad \mathbf{H}_{F_k} = \left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_{F_k}^{B}} \right|_{(\hat{\mathbf{x}}_{k|k-1}^{B}, \hat{\mathbf{x}}_{\mathcal{E}_k}^{R_k})}$$

$$\mathbf{G}_k = \left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_{\mathcal{E}_k}^{R_k}} \right|_{(\hat{\mathbf{x}}_{k|k-1}^{B}, \hat{\mathbf{x}}_{\mathcal{E}_k}^{R_k})} . \qquad (5)$$

Measurement $\mathbf{z}_k$ is used to obtain a new estimation of the state using the standard EKF update equations:

$$\hat{\mathbf{x}}_k^{B} = \hat{\mathbf{x}}_{k|k-1}^{B} + \mathbf{K}_k \nu_k$$

$$\mathbf{P}_k^{B} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1}^{B}$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}^{B} \mathbf{H}_k^{\mathrm{T}} (\mathbf{H}_k \mathbf{P}_{k|k-1}^{B} \mathbf{H}_k^{\mathrm{T}} + \mathbf{G}_k \mathbf{R}_k \mathbf{G}_k^{\mathrm{T}})^{-1} \qquad (6)$$

where $\nu_k = \mathbf{z}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}^{B}, \hat{\mathbf{x}}_{\mathcal{E}_k}^{R_k})$ is called the *innovation* of the filter, with covariance matrix $\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1}^{B} \mathbf{H}_k^{\mathrm{T}} + \mathbf{G}_k \mathbf{R}_k \mathbf{G}_k^{\mathrm{T}}$.

3. The inconsistency of EKF-SLAM

Let $\hat{\mathbf{x}}_k^{B}$ and $\mathbf{P}_k^{B}$ be the first two moments of the SLAM state estimated at time k. The state estimator is called *consistent* [2] if its state estimation error $\mathbf{x}_k^{B} - \hat{\mathbf{x}}_k^{B}$ is *unbiased*, i.e. $E\left[\mathbf{x}_k^{B} - \hat{\mathbf{x}}_k^{B} \right] = \mathbf{0}$ and the actual Mean Square Error matches the filter calculated covariances:

$$E\left[\left(\mathbf{x}_k^{B} - \hat{\mathbf{x}}_k^{B} \right) \left(\mathbf{x}_k^{B} - \hat{\mathbf{x}}_k^{B} \right)^{\mathrm{T}} \right] = \mathbf{P}_k^{B}. \qquad (7)$$

Whenever ground-truth for the state variables is available, a statistical test for filter consistency can be carried out on the Normalized Estimation Error Squared (NEES):

$$\text{NEES} = \left(\mathbf{x}_k^{B} - \hat{\mathbf{x}}_k^{B} \right)^{\mathrm{T}} \left(\mathbf{P}_k^{B} \right)^{-1} \left(\mathbf{x}_k^{B} - \hat{\mathbf{x}}_k^{B} \right) \leq \chi_{r,1-\alpha}^{2} \qquad (8)$$

where $\chi_{r,1-\alpha}^{2}$ is a threshold obtained from the χ^2 distribution with $r = \dim(\mathbf{x}_k^{B})$ degrees of freedom, and α the desired significance level (usually 0.05).

Unfortunately, for most real-time applications, ground truth for the state variables is not available. However, a statistical test for real-time consistency can still be carried out, in this case, on the Normalized Innovation Squared (NIS):

$$\text{NIS} = \nu_k^{\mathrm{T}} \mathbf{S}_k^{-1} \nu_k \leq \chi_{r,1-\alpha}^{2} \qquad (9)$$

where $r = \dim(\nu_k)$.

In practice, one of the most critical factors that jeopardize the consistency of any SLAM algorithm are the incorrect data associations between observations and map features. To isolate the effects of linearization errors on the consistency of the EKF-based approach to SLAM, we have designed a simulated experiment with known data association. The vehicle travels along a rectangular-shaped trajectory of 100×20 m, i.e. a 240 m loop trajectory, moving 1 m per step. The map of the navigation environment is composed of 2-D point features, located at both sides of the vehicle trajectory with a feature density of 0.5 feature/m. The vehicle is equipped with a range-bearing sensor with a maximum range of 15 m and a 180° frontal field-of-view. Gaussian-distributed synthetic errors were generated for both the sensor measurements (standard deviation of 5 cm per m in range and 0.5° in orientation) and for the odometry model of the vehicle (standard deviations of 0.2 m per m in displacement and 0.5° in orientation). We have run a Monte Carlo simulation with 20 replications.

Fig. 1, top, shows the evolution of angular error and uncertainty (2σ bounds) in the vehicle location along the trajectory for a representative replication of the experiment. For this SLAM simulation, the initial vehicle location is used as base reference, allowing us to set the initial vehicle uncertainty to zero. The theoretical uncertainty level was obtained by simulating the same trajectory linearizing around ground truth (simulated with noise = 0), so that there are no linearization
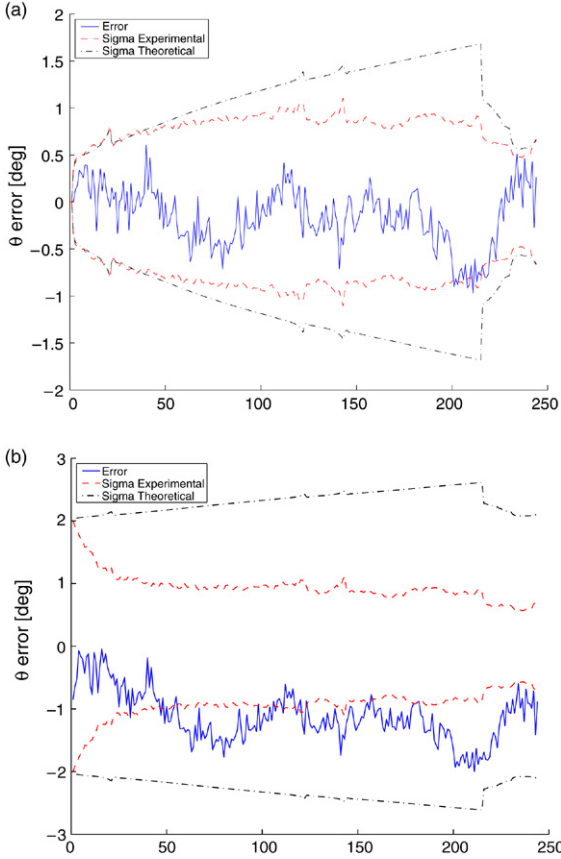
Fig. 1. Angular error and 2σ uncertainty bounds of the vehicle estimated location for the cases of zero (a) and nonzero (b) initial uncertainty.

errors. We can see that, while the theoretical angular uncertainty increases until loop closing, the uncertainty computed by the EKF saturates reaching a maximum level (around $0.5°$ in this case). This results in the vehicle location estimation failing the consistency check of Eq. (8) after only 100 m. Additionally, the average of the 20 replications resulted in biased estimation for frontal, lateral and angular errors.

From the experimental experience gained from EKF-SLAM mapping an important conclusion can be derived: The consistency of the EKF-SLAM algorithm greatly depends on the level of uncertainty of the state vector, the higher the uncertainty, especially vehicle angular uncertainty, the worst the consistency of the estimates.

It is common practice to build a map relative to a fixed base reference, different from the initial vehicle location. This normally requires assignment of an initial level of uncertainty to the vehicle estimated location. As argued in [3], the vehicle uncertainty should always be above this initial level. Surprisingly, our simulations shows that when a non-zero initial uncertainty is used (Fig. 1, bottom), the estimated vehicle uncertainty rapidly drops below its initial value ($1°$) making the estimation inconsistent after only 50 EKF update steps. This corroborates the results of [4], but also shows that the problem arises in practice earlier than they suggested.

4. Robocentric mapping

In this section we formulate the EKF-SLAM problem using the reference frame attached to the vehicle R as base reference of the stochastic map. Thus, the environmental information $\{R, F_1, \ldots, F_n\}$ is represented by a stochastic state vector $\mathbf{x}^R$ with estimated mean $\hat{\mathbf{x}}^R$ and estimated error covariance $\mathbf{P}^R$:

$$\hat{\mathbf{x}}^R = \begin{bmatrix} \hat{\mathbf{x}}_B^R \\ \hat{\mathbf{x}}_{F_1}^R \\ \vdots \\ \hat{\mathbf{x}}_{F_n}^R \end{bmatrix}; \qquad \mathbf{P}^R = \begin{bmatrix} \mathbf{P}_B^R & \cdots & \mathbf{P}_{BF_n}^R \\ \vdots & \ddots & \vdots \\ \mathbf{P}_{F_n B}^R & \cdots & \mathbf{P}_{F_n}^R \end{bmatrix} \tag{10}$$

where the world reference frame B has been included as a non-observable feature in the stochastic state vector (this has the purpose of allowing recovery of the equivalent absolute map if desired). With the purpose of avoiding the inconsistency problem related to non-zero initial uncertainty described above, we take the initial vehicle location as base reference $B = R_0$, and thus at step $k = 0$ the map is initialized with perfect knowledge of the world location: $\hat{\mathbf{x}}_0^R = (0, 0, 0)^T$ and $\mathbf{P}_0^R = \mathbf{0}$.

In robocentric mapping, each filter iteration includes three steps: prediction, update and composition, which are detailed next.

4.1. The prediction step

After the vehicle changes its location from step $k - 1$ to step k, the complete structure of the stochastic map should be affected by the process noise associated with the displacement $\mathbf{x}_{R_k}^{R_{k-1}}$ as estimated by odometry, and with covariance matrix $\mathbf{Q}_k$. Thus, the estimated location of a given map feature F should be updated as:

$$\mathbf{x}_{F_k|k-1}^{R_k} = \ominus \mathbf{x}_{R_k}^{R_{k-1}} \oplus \mathbf{x}_{F_{k-1}}^{R_{k-1}} \tag{11}$$

and therefore, its estimated covariance would be computed from the corresponding linearization around the estimated values $\hat{\mathbf{x}}_{R_k}^{R_{k-1}}$ and $\hat{\mathbf{x}}_{F_{k-1}}^{R_{k-1}}$. As odometry is the least precise component in the system, this linearization can introduce significant errors. Instead, we propose to delay the composition (11) until the estimated vehicle motion has been improved by the update step of the EKF algorithm.

Therefore, in the prediction step the vehicle motion $\hat{\mathbf{x}}_{R_k}^{R_{k-1}}$ obtained by odometry is simply added, as an independent feature, to the previously available stochastic map $\mathbf{x}_{k-1}^{R_{k-1}}$:

$$\hat{\mathbf{x}}_{k|k-1}^{R_{k-1}} = \begin{bmatrix} \hat{\mathbf{x}}_{k-1}^{R_{k-1}} \\ \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \end{bmatrix}; \qquad \mathbf{P}_{k|k-1}^{R_{k-1}} = \begin{bmatrix} \mathbf{P}_{k-1}^{R_{k-1}} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_k \end{bmatrix}. \tag{12}$$

4.2. The update step

Now, linearization of the measurement equation around the estimated values of both the stochastic state vector and the

partial measurement $\mathbf{z}_k$ yields:

$$\mathbf{z}_k \simeq \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}^{R_{k-1}}) + \mathbf{H}_k(\mathbf{x}_k^{R_{k-1}} - \hat{\mathbf{x}}_{k|k-1}^{R_{k-1}})$$

$$\mathbf{H}_k = \left.\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_k^{R_{k-1}}}\right|_{\hat{\mathbf{x}}_{k|k-1}^{R_{k-1}}} = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}_{F_k} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}_{R_k} \end{bmatrix} \qquad (13)$$

where:

$$\mathbf{H}_{F_k} = \left.\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_{F_k}^{R_{k-1}}}\right|_{\hat{\mathbf{x}}_{k|k-1}^{R_{k-1}}} ; \; \mathbf{H}_{R_k} = \left.\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_{R_k}^{R_{k-1}}}\right|_{\hat{\mathbf{x}}_{k|k-1}^{R_{k-1}}} .$$

Equations which are subsequently used to obtain a new estimation of the stochastic state vector $\hat{\mathbf{x}}_{k|k}^{R_{k-1}}$ and its covariance matrix $\mathbf{P}_{k|k}^{R_{k-1}}$, using the previously described EKF update equations. Note that, because the relative displacement of the vehicle from time $k-1$ to time k was included as a feature of the stochastic state vector, it is also refined during the application of the update equations.

The use of a robot centered representation greatly influences the internal structure of the measurement equation (13) in comparison with the measurement equation (5) obtained by using an absolute representation of the stochastic map. Precisely, and except for loop closing, the uncertainty of the filter innovation is greatly reduced down to the level of the observation uncertainty, thus, improving the accuracy of the linearization.

4.3. The composition step

As a final step in the robocentric mapping algorithm, the stochastic state vector of the robocentric map is obtained by affecting each estimated location by the improved vehicle motion:

$$\hat{\mathbf{x}}_k^{R_k} = \begin{bmatrix} \ominus \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \oplus \hat{\mathbf{x}}_{R_0}^{R_{k-1}} \\ \ominus \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \oplus \hat{\mathbf{x}}_{F_1}^{R_{k-1}} \\ \vdots \\ \ominus \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \oplus \hat{\mathbf{x}}_{F_n}^{R_{k-1}} \end{bmatrix} \qquad (14)$$

with corresponding covariance matrix:

$$\mathbf{P}_k^{R_k} \simeq \begin{bmatrix} \mathbf{J}_2 & \mathbf{J}_1 \end{bmatrix} \mathbf{P}_{k|k}^{R_{k-1}} \begin{bmatrix} \mathbf{J}_2^{\mathrm{T}} \\ \mathbf{J}_1^{\mathrm{T}} \end{bmatrix}$$

$$\mathbf{J}_1 = \begin{bmatrix} \mathbf{J}_{1\oplus}\{\ominus \hat{\mathbf{x}}_{R_k}^{R_{k-1}}, \hat{\mathbf{x}}_{R_0}^{R_{k-1}}\} \mathbf{J}_\ominus\{\hat{\mathbf{x}}_{R_k}^{R_{k-1}}\} \\ \vdots \\ \mathbf{J}_{1\oplus}\{\ominus \hat{\mathbf{x}}_{R_k}^{R_{k-1}}, \hat{\mathbf{x}}_{F_n}^{R_{k-1}}\} \mathbf{J}_\ominus\{\hat{\mathbf{x}}_{R_k}^{R_{k-1}}\} \end{bmatrix}$$

$$\mathbf{J}_2 = \begin{bmatrix} \mathbf{J}_{2\oplus}\{\ominus \hat{\mathbf{x}}_{R_k}^{R_{k-1}}, \hat{\mathbf{x}}_{R_0}^{R_{k-1}}\} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{J}_{2\oplus}\{\ominus \hat{\mathbf{x}}_{R_k}^{R_{k-1}}, \hat{\mathbf{x}}_{F_n}^{R_{k-1}}\} \end{bmatrix} . \qquad (15)$$

The computational cost of the update steps in both absolute and robocentric mapping requires updating the covariance

matrix of the estimation and is thus $O(n^2)$, where n is the number of features in the map. The prediction step in absolute mapping requires updating the correlations between the vehicle and the features and is thus $O(n)$, while in robocentric mapping it only requires stacking, $O(1)$. In contrast, robocentric mapping includes an additional composition step in which the full covariance matrix is updated, again with a computational cost of $O(n^2)$.

5. Robocentric map joining

In [10] Tardós et al. proposed a map building technique in which, instead of building one global map from the beginning of the exploration task, a sequence of local maps of limited size is built, and later joined together, to obtain the global map. Here we show that, not only is map joining computationally more efficient than building one global map from the beginning, as is it shown in [10], but it also allows one to attain better consistency in the stochastic map.

Robocentric map joining is carried out as follows: given two consecutive robocentric local maps:

$$\mathcal{M}_{\mathcal{F}}^{R_l} = (\hat{\mathbf{x}}_{\mathcal{F}}^{R_l}, \mathbf{P}_{\mathcal{F}}^{R_l}); \qquad \mathcal{F} = \{R_l, B_l, F_1, \ldots, F_m\}$$
$$\mathcal{M}_{\mathcal{E}}^{R_{l-1}} = (\hat{\mathbf{x}}_{\mathcal{E}}^{R_{l-1}}, \mathbf{P}_{\mathcal{E}}^{R_{l-1}}); \qquad \mathcal{E} = \{R_{l-1}, B_{l-1}, E_1, \ldots, E_n\}.$$

Because, there exists a link between the two maps $B_l \equiv R_{l-1}$ a full stochastic map can obtained by map joining:

$$\mathcal{M}_{\mathcal{F}+E}^{R_l} = (\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}}^{R_l}, \mathbf{P}_{\mathcal{F}+\mathcal{E}}^{R_l})$$

which contains the estimations of the features from both maps, relative to the reference frame R_l of the current robocentric local map. We proceed as follows:

5.1. Stacking together the local maps

Because the robocentric local maps $\mathcal{M}_{\mathcal{F}}^{R_l}$ and $\mathcal{M}_{\mathcal{E}}^{R_{l-1}}$ are built using independent information, they are uncorrelated [10]. Thus, we form a stacked state vector:

$$\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}} = \begin{bmatrix} \hat{\mathbf{x}}_{\mathcal{F}}^{R_l} \\ \hat{\mathbf{x}}_{\mathcal{E}}^{R_{l-1}} \end{bmatrix}; \qquad \mathbf{P}_{\mathcal{F}+\mathcal{E}} = \begin{bmatrix} \mathbf{P}_{\mathcal{F}}^{R_l} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{\mathcal{E}}^{R_{l-1}} \end{bmatrix} \qquad (16)$$

which stores all the available information about the previous uncorrelated local maps.

5.2. The update step

Data association is carried out to match the features of the local map $\mathcal{M}_{\mathcal{F}}^{R_l}$ with those of the local map $\mathcal{M}_{\mathcal{E}}^{R_{l-1}}$. We use the Joint Compatibility test [12], which obtains the largest set of pairings which are jointly compatible, a consensus criteria that reduces the possibility of accepting a spurious pairing. Let F_i and E_{ij_i} be two matched features, thus, a nonlinear measurement equation of the form:

$$\mathbf{z}_{ij_i} = \mathbf{h}_{ij_i}(\mathbf{x}_{\mathcal{F}+\mathcal{E}}) = \ominus \mathbf{x}_{F_i}^{R_l} \oplus \mathbf{x}_{B_l}^{R_l} \oplus \mathbf{x}_{E_{j_i}}^{R_{l-1}} = \mathbf{0} \qquad (17)$$

constrains their relative location vector, where $B_l \equiv R_{l-1}$ as discussed above. Linearization of Eq. (17) around the best available estimation $\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}}$ gives:

$$\mathbf{z}_{ij_i} \simeq \mathbf{h}_{ij_i}(\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}}) + \mathbf{H}_{ij_i}(\mathbf{x}_{\mathcal{F}+\mathcal{E}} - \hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}}) \tag{18}$$

where, the linearization coefficient results from:

$$\mathbf{H}_{ij_i} = \left. \frac{\partial \mathbf{h}_{ij_i}}{\partial \mathbf{x}_{\mathcal{F}+\mathcal{E}}} \right|_{\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}}}$$
$$= [\mathbf{H}_{B_l}\mathbf{0}\dots\mathbf{0}\,\mathbf{H}_{F_i}\,\mathbf{0}\dots\mathbf{0}\,\mathbf{H}_{E_{j_i}}\,\mathbf{0}\dots\mathbf{0}] \tag{19}$$

and,

$$\mathbf{H}_{B_l} = \left. \frac{\partial \mathbf{h}_{ij_i}}{\partial \mathbf{x}_{B_l}^{R_l}} \right|_{\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}}} ; \qquad \mathbf{H}_{F_i} = \left. \frac{\partial \mathbf{h}_{ij_i}}{\partial \mathbf{x}_{F_i}^{R_l}} \right|_{\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}}} ;$$
$$\mathbf{H}_{E_{j_i}} = \left. \frac{\partial \mathbf{h}_{ij_i}}{\partial \mathbf{x}_{E_{j_i}}^{R_{l-1}}} \right|_{\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}}} . \tag{20}$$

The update of the stacked state vector by using the EKF equations, would therefore improve not only the structure of both local maps but also the link between them. This strategy increases the accuracy of the map joining technique over the direct change of reference between local map proposed in [10]. After updating the map, matched features of the local map $\mathcal{M}_{\mathcal{E}}^{R_{l-1}}$ are removed from the stacked state vector and only a subset $\mathcal{E}^* \subset \mathcal{E}$ of features remains.

5.3. The composition step

Given that the features from the current local map are expressed relative to reference R_l and features from the previous local map are expressed relative to reference R_{l-1}, to form the full stochastic map $\mathcal{M}_{\mathcal{F}+\mathcal{E}^*}^{R_l}$ we only need to transform the features of the previous map to the reference R_l using their common link. Thus,

$$\mathbf{x}_{\mathcal{F}+\mathcal{E}^*}^{R_l} = \begin{bmatrix} \mathbf{x}_{\mathcal{F}}^{R_l} \\ \mathbf{x}_{\mathcal{E}^*}^{R_l} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{\mathcal{F}}^{R_l} \\ \mathbf{x}_{R_{l-1}}^{R_l} \oplus \mathbf{x}_{\mathcal{E}^*}^{R_{l-1}} \end{bmatrix} \tag{21}$$

which directly provides the estimation $\hat{\mathbf{x}}_{\mathcal{F}+\mathcal{E}^*}^{R_l}$ of the full stochastic map. Its covariance matrix $\mathbf{P}_{\mathcal{F}+\mathcal{E}^*}^{R_l}$ derives from the linearization of Eq. (21) as:

$$\mathbf{P}_{\mathcal{F}+\mathcal{E}^*}^{R_l} = \begin{bmatrix} \mathbf{P}_{\mathcal{F}}^{R_l} & \mathbf{P}_{\mathcal{F}\mathcal{E}^*}^{R_l} \\ \mathbf{P}_{\mathcal{E}^*\mathcal{F}}^{R_l} & \mathbf{P}_{\mathcal{E}^*}^{R_l} \end{bmatrix}$$
$$\simeq \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{J}_1 & \mathbf{J}_2 \end{bmatrix} \begin{bmatrix} \mathbf{P}_{\mathcal{F}}^{R_l} & \mathbf{P}_{\mathcal{F}\mathcal{E}^*} \\ \mathbf{P}_{\mathcal{E}^*\mathcal{F}} & \mathbf{P}_{\mathcal{E}^*}^{R_{l-1}} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{J}_1^{\mathrm{T}} \\ \mathbf{0} & \mathbf{J}_2^{\mathrm{T}} \end{bmatrix} \tag{22}$$

where the matrices of the Jacobians (see Appendix) are:

$$\mathbf{J}_1 = \begin{bmatrix} \mathbf{J}_{1\oplus}\{\hat{\mathbf{x}}_{R_{l-1}}^{R_l}, \hat{\mathbf{x}}_{B_{l-1}}^{R_{l-1}}\} & \dots & \mathbf{0} \\ \vdots & & \vdots \\ \mathbf{J}_{1\oplus}\{\hat{\mathbf{x}}_{R_{l-1}}^{R_l}, \hat{\mathbf{x}}_{E_n}^{R_{l-1}}\} & \dots & \mathbf{0} \end{bmatrix}$$

and

$$\mathbf{J}_2 = \begin{bmatrix} \mathbf{J}_{2\oplus}\{\hat{\mathbf{x}}_{R_{l-1}}^{R_l}, \hat{\mathbf{x}}_{B_{l-1}}^{R_{l-1}}\} & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{J}_{2\oplus}\{\hat{\mathbf{x}}_{R_{l-1}}^{R_l}, \hat{\mathbf{x}}_{E_n}^{R_{l-1}}\} \end{bmatrix} .$$

As in absolute map joining [10], the computational cost of robocentric map joining requires updating the full stochastic map covariance at each join and thus is $O(n^2)$. Note however that most of the updates take place in a local map of bounded size (with a $O(1)$ cost), and thus you can expect robocentric map joining to cut processing time by a large constant factor.

6. Experiments

In this section, we carry out a series of simulated and indoor/outdoor experiments to validate the proposed algorithm.

6.1. Simulation

In our controlled simulation environment we have compared the performance of the different algorithms presented in the previous discussion. Two main categories of experiments have been conducted, namely, those with an absolute representation, and those with a robot centered representation.

Fig. 2(a) compares the vehicle heading uncertainty computed using an absolute map representation by an ideal error-free EKF (simulated with noise = 0), the standard EKF and the map joining algorithm proposed in [10]. Note that all mapping algorithms in error-free simulations should produce identical correct results. In this case, the heading uncertainty computed by the standard EKF presents a saturation effect which makes the algorithm more and more optimistic as the number of updates increases, and thus it undermines consistency. The map joining algorithm performs much better but it is still slightly optimistic in the second-half of the trajectory.

Similarly, Fig. 2(b) compares the vehicle heading uncertainty, using a robot centered representation, of the ideal error-free EKF against the robocentric mapping approach reported in [5] and the new robocentric map joining algorithm. As observed from the figure, both algorithms obtain a non-optimistic estimation for the vehicle heading uncertainty along the vehicle trajectory, which makes loop closing detection possible. In this experiment, robocentric map joining behaves slightly pessimistically as compared to the basic robocentric approach. It provides a more efficient solution in terms of computing time. Finally, we can see that in these robot centered approaches, saturation effects have disappeared.

6.2. Indoor/outdoor mapping experiment

To validate the new robocentric mapping algorithm, we have conducted an experiment in one of the buildings at our campus using a robotized wheelchair equipped with a SICK laser scanner. The vehicle was hand-driven along a mixed indoor/outdoor path of about 250 m, at a mean speed of 0.45 m/s. The scans were processed to obtain line features using a robust segmentation algorithm.
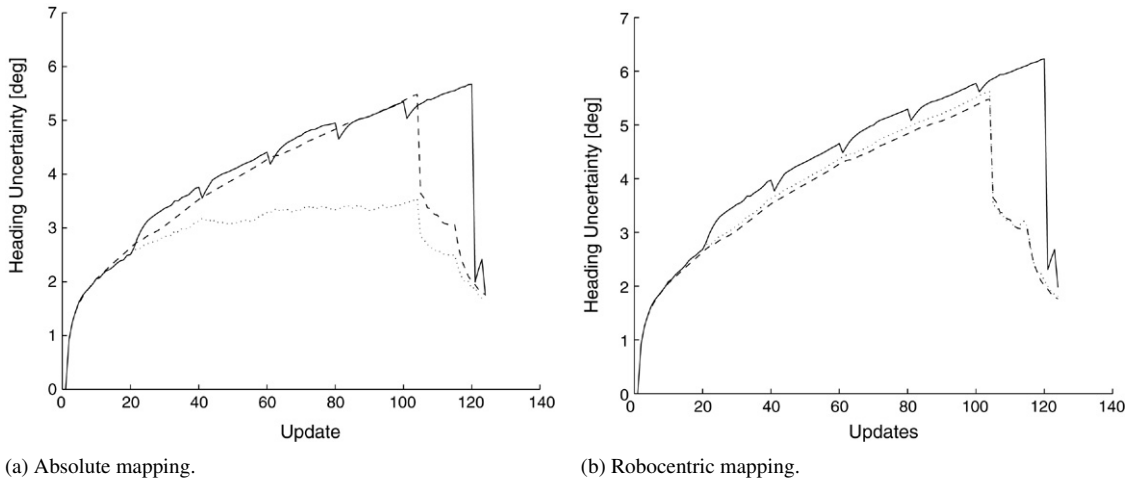
(a) Absolute mapping.

(b) Robocentric mapping.

Fig. 2. Vehicle heading uncertainty (1σ) computed using absolute and robocentric mapping by: the ideal error-free EKF (dashed line), the standard EKF (dotted line) and the map joining algorithm (solid line).

Fig. 3. Classical EKF-SLAM algorithm with an absolute representation. Observe that the vehicle location uncertainty (extremely small ellipses) is incompatible with the real error, especially in the top left part of the figure, where clearly, multiple hypotheses for the same feature appear.
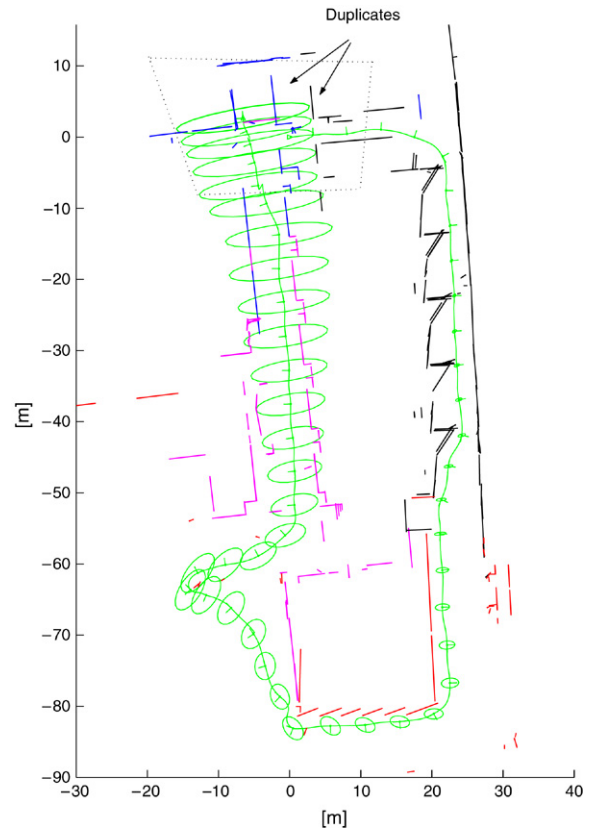

Fig. 4. Robocentric map joining before loop closing. In this case, the vehicle location uncertainty is consistent with the real error, as observed in the top left part of the figure (Note that results have been transformed back to the absolute representation).

Fig. 3 shows the map obtained along the commanded trajectory by the classical EKF-SLAM algorithm using an absolute representation. The saturation effect in the map uncertainty makes the result inconsistent as observed in the top-left part of the figure, where clearly the loop could not be closed by simple data association strategies. We processed the same data by dividing the full map into 50 robocentric maps (each for a trajectory of 5 m with approximate 10 line features) and using

robocentic map joining to compute the full stochastic map. As shown in Fig. 4, this change of representation, from absolute to robot centered using map joining performs adequately in this case. Due to the increased accuracy in linearization and the reduced level of uncertainty of this local representation, the mapping of the 250 m trajectory was accurately performed, closing the loop when the vehicle homed.
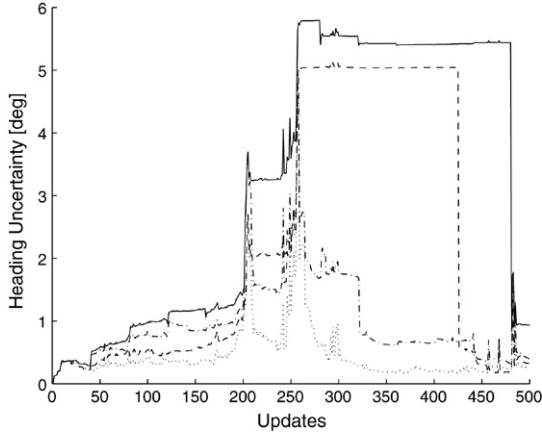
Fig. 5. Experimental vehicle heading uncertainty: EKF-SLAM algorithm (dotted line), map joining algorithm (dash-dot), robocentric (dashed) and robocentric map joining (solid).

Finally, Fig. 5 depicts the evolution of the vehicle angular uncertainty along the 250 m trajectory. Again, the EKF-SLAM estimated uncertainty present the previously discussed saturation effects driving the solution of the mapping algorithm out of consistency. The Map Joining algorithm which also computes the estimated location of features in an absolute representation present a similar, although clearly improved, optimistic behavior. As observed in simulation, and now in a real experiment, the robot centered representation surpasses those inconsistency problems, at least at the scale of the reported experiments. Although the Robocentric mapping algorithm performed consistently, the Robocentric Map Joining algorithm provided the more efficient solution from the computational time point-of-view with an accurate estimation of the uncertainty.

7. Conclusion

In this work we have shown that in the standard extended Kalman Filter approach to SLAM, linearization errors produce inconsistency problems that show up long before computational problems arise. We follow a precise definition of filter consistency that considers both the accuracy of the estimation and of its covariances.

We have proposed the Robocentric Map Joining algorithm which improves consistency of the mapping scheme by: (1) bounding the uncertainty along the exploration trajectory using a sequence of local maps, and (2) improving linearization of the model equations due to the reduced level of uncertainty provided by the robot centered representation. As described both in simulation and in real mixed indoor/outdoor experiments, the combination of local map joining and robocentric mapping allows one to apply the EKF-based solution to SLAM at a larger scale.

It is however likely that nonlinearity problems will arise again as larger environments are tackled. We feel that to overcome these limitations it is important to investigate the use of alternative formulations to SLAM, nonlinear and non Gaussian methods. Making these methods computationally efficient to be used in real time is the next important challenge in SLAM.

Appendix. Transformation and Jacobians in 2D

Two basic operations used in stochastic mapping are transformation inversion and composition, which were represented by [1] using operators $\ominus$ and $\oplus$:

$$\hat{\mathbf{x}}_A^B = \ominus \hat{\mathbf{x}}_B^A$$
$$\hat{\mathbf{x}}_C^A = \hat{\mathbf{x}}_B^A \oplus \hat{\mathbf{x}}_C^B.$$

The Jacobians of these operations are defined as:

$$\mathbf{J}_\ominus\left\{\hat{\mathbf{x}}_B^A\right\} = \left.\frac{\partial\left(\ominus\mathbf{x}_B^A\right)}{\partial\mathbf{x}_B^A}\right|_{(\hat{\mathbf{x}}_B^A)}$$

$$\mathbf{J}_{1\oplus}\left\{\hat{\mathbf{x}}_B^A, \hat{\mathbf{x}}_C^B\right\} = \left.\frac{\partial\left(\mathbf{x}_B^A \oplus \mathbf{x}_C^B\right)}{\partial\mathbf{x}_B^A}\right|_{(\hat{\mathbf{x}}_B^A, \hat{\mathbf{x}}_C^B)}$$

$$\mathbf{J}_{2\oplus}\left\{\hat{\mathbf{x}}_B^A, \hat{\mathbf{x}}_C^B\right\} = \left.\frac{\partial\left(\mathbf{x}_B^A \oplus \mathbf{x}_C^B\right)}{\partial\mathbf{x}_C^B}\right|_{(\hat{\mathbf{x}}_B^A, \hat{\mathbf{x}}_C^B)}.$$

In 2D, the location of a reference B relative to a reference A (or transformation from A to B) can be expressed using a vector with three d.o.f.: $\mathbf{x}_B^A = [x_1, y_1, \phi_1]^T$. The location of A relative to B is computed using the inversion operation:

$$\mathbf{x}_A^B = \ominus\mathbf{x}_B^A = \begin{bmatrix} -x_1\cos\phi_1 - y_1\sin\phi_1 \\ x_1\sin\phi_1 - y_1\cos\phi_1 \\ -\phi_1 \end{bmatrix}.$$

The Jacobian of transformation inversion is:

$$\mathbf{J}_\ominus\{\mathbf{x}_B^A\} = \begin{bmatrix} -\cos\phi_1 & -\sin\phi_1 & -x_1\sin\phi_1 - y_1\cos\phi_1 \\ \sin\phi_1 & -\cos\phi_1 & x_1\cos\phi_1 + y_1\sin\phi_1 \\ 0 & 0 & -1 \end{bmatrix}.$$

Let $\mathbf{x}_C^B = [x_2, y_2, \phi_2]^T$ be a second transformation. The location of reference C relative to A is obtained by the composition of transformations $\mathbf{x}_B^A$ and $\mathbf{x}_C^B$:

$$\mathbf{x}_C^A = \mathbf{x}_B^A \oplus \mathbf{x}_C^B = \begin{bmatrix} x_1 + x_2\cos\phi_1 - y_2\sin\phi_1 \\ y_1 + x_2\sin\phi_1 + y_2\cos\phi_1 \\ \phi_1 + \phi_2 \end{bmatrix}.$$

The Jacobians of transformation composition are:

$$\mathbf{J}_{1\oplus}\{\mathbf{x}_B^A, \mathbf{x}_C^B\} = \begin{bmatrix} 1 & 0 & -x_2\sin\phi_1 - y_2\cos\phi_1 \\ 0 & 1 & x_2\cos\phi_1 - y_2\sin\phi_1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{J}_{2\oplus}\{\mathbf{x}_B^A, \mathbf{x}_C^B\} = \begin{bmatrix} \cos\phi_1 & -\sin\phi_1 & 0 \\ \sin\phi_1 & \cos\phi_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

A generalization of the $\oplus$ operator to also represent the composition of transformations with feature location vectors, which results in the change of base reference of the feature, can be found in [10].

References

[1] R. Smith, M. Self, P. Cheeseman, A stochastic map for uncertain spatial relationships, in: O. Faugeras, G. Giralt (Eds.), Robotics Research: The Fourth Int. Symposium, The MIT Press, 1988, pp. 467–474.

[2] Y. Bar-Shalom, X. Rong Li, T. Kirubarajan, Estimation with Applications to Tracking and Navigation, Wiley InterScience, 2001.

[3] M.W.M.G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, M. Csorba, A solution to the simultaneous localization and map building (SLAM) problem, IEEE Transactions on Robotics and Automation 17 (3) (2001) 229–241.

[4] S. Julier, J.K. Uhlmann, A counter example to the theory of simultaneous localization and map building, in: Proceedings of the 2001 IEEE International Confeerence on Robotics and Automation, Seoul, Korea, 2001, pp. 4238–4243.

[5] J.A. Castellanos, J. Neira, J.D. Tardós, Limits to the consistency of EKF-based SLAM, in: 5th IFAC Symposium on Intelligent Autonomous Vehicles, Lisbon, July 2004.

[6] M.A. Paskin, Thin junction tree filters for simultaneous localization and mapping, in: International Joint Conference on Artificial Intelligence, 2003, pp. 1157–1164.

[7] R. Martinez-Cantin, J.A. Castellanos, Unscented SLAM for large-scale outdoor environments, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Canada, August 2005, pp. 328–333.

[8] A. Doucet, N. de Freitas, K. Murphy, S. Russell, Rao-blackwellised particle filtering for dynamic bayesian networks, in: Uncertainty in Artificial Intelligence, 2000.

[9] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, E. Nebot, FastSLAM: An efficient solution to the simultaneous localization and mapping problem with unknown data association, Journal of Machine Learning Research (2004).

[10] J.D. Tardós, J. Neira, P.M. Newman, J.J. Leonard, Robust mapping and localization in indoor environments using sonar data, International Journal of Robotics Research 21 (4) (2002) 311–330.

[11] J.A. Castellanos, J.D. Tardós, Mobile Robot Localization and Map Building. A Multisensor Fusion Approach, Kluwer Academic Publishers, 1999.

[12] J. Neira, J.D. Tardós, Data association in stochastic mapping using the joint compatibility test, IEEE Transactions on Robotics and Automation 17 (6) (2001) 890–897.

[13] J. Guivant, E. Nebot, Optimization of the simultaneous localization and map-building algorithm for real-time implementation, IEEE Transactions on Robotics and Automation 17 (3) (2001) 242–257.

[14] J.H. Kim, S. Sukkarieh, Airborne simultaneous localisation and map building, in: Proceedings of the IEEE International Conference on Robotics and Automation, Taipei, Taiwan, September 2003.

[15] J.J. Leonard, R. Carpenter, H.J.S. Feder, Stochastic mapping using forward look sonar, Robotica 19 (2001) 467–480.

[16] J.J. Leonard, H.J.S. Feder, A computationally efficient method for large-scale concurrent mapping and localization, in: D. Koditschek, J. Hollerbach (Eds.), Robotics Research: The Ninth International Symposium, Springer Verlag, 2000, pp. 169–176.

[17] J.J. Leonard, P.M. Newman, Consistent, convergent and constant-time SLAM, in: International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 2003.

J.A. Castellanos was born in Zaragoza, Spain, in 1969. He earned the M.S. and Ph.D. degrees in industrial–electrical engineering from the University of Zaragoza, Spain, in 1994 and 1998, respectively. He is associate professor with the Departamento de Informática e Ingeniería de Sistemas, University of Zaragoza, where he is in charge of courses in SLAM, automatic control systems, and computer modelling and simulation. His current research interests include multisensor fusion and integration, Bayesian estimation in nonlinear systems, and simultaneous localization and mapping.

R. Martinez-Cantin was born in Zaragoza, Spain, in 1977. He earned the M.S. degree in industrial-electrical engineering from the University of Zaragoza, Spain, in 2003. He pursues the Ph.D. degree in the Robotics, Perception and Real Time Group of the University of Zaragoza. His research interests include mobile robotics, computer vision, sensor data fusion, probabilistic inference and learning, and behaviour models.

J.D. Tardós was born in Huesca, Spain, in 1961. He earned the M.S. and Ph.D. degrees in industrial-electrical engineering from the University of Zaragoza, Spain, in 1985 and 1991, respectively. He is associate professor with the Departamento de Informática e Ingeniería de Sistemas, University of Zaragoza, where he is in charge of courses in real time systems, computer vision, and artificial intelligence. His current research interests include perception and mobile robotics.

J. Neira was born in Bogotá, Colombia, in 1963. He earned the M.S. degree in computer science for the Universidad de los Andes, Colombia, in 1986, and the Ph.D. degree in computer science from the University of Zaragoza, Spain, in 1993. He is associate professor with the Departamento de Informática e Ingeniería de Sistemas, University of Zaragoza, where he is in charge of courses in compiler theory, computer vision, and mobile robotics. His current research interests include autonomous robots, data association, and environment modelling.