Feature Selection Criteria for Real Time EKF-SLAM Algorithm

Fernando Auat Cheein, Gustavo Scaglia, Fernando di Sciasio and Ricardo Carelli

Instituto de Automatica, National University of San Juan, San Juan, Argentina.
Corresponding author E-mail: fauat@inaut.unsj.edu.ar

Abstract: This paper presents a selection procedure for environment features for the correction stage of a SLAM (Simultaneous Localization and Mapping) algorithm based on an Extended Kalman Filter (EKF). This approach decreases the computational time of the correction stage which allows for real and constant-time implementations of the SLAM. The selection procedure consists in choosing the features the SLAM system state covariance is more sensible to. The entire system is implemented on a mobile robot equipped with a range sensor laser. The features extracted from the environment correspond to lines and corners. Experimental results of the real time SLAM algorithm and an analysis of the processing-time consumed by the SLAM with the feature selection procedure proposed are shown. A comparison between the feature selection approach proposed and the classical sequential EKF-SLAM along with an entropy feature selection approach is also performed.

Keywords: SLAM, Mobile Robots, Feature Selection.

1. Introduction

The mobile robotic research area can be divided into three main lines: Mapping, Localization and Navigation/Exploration. Mapping is related to the way the robot models the environment based on the sensors data (Montemerlo, M. & Thrun, S., 2007; Hähnel, D. et al., 2003). Localization is the knowledge of the robot’s position within the environment through which it is moving or acting with (Guivant, J. E. & Nebot, E. M., 2001; Xi, B. et al., 2008). Navigation (or exploration) is the strategy used to drive the mobile robot motion in the environment (Arkin, R. C., 1998; Thrun, S. et al., 2005).

In order to reach full autonomy, the robot should be capable of localizing itself in the environment through which it is navigating while building a map of such an environment, without human intervention on the mobile robot’s motion commands (Montemerlo, M. & Thrun, S., 2007; Thrun, S. et al., 2005). The SLAM (Simultaneous Localization and Mapping) algorithm comes up as a solution to the concurrent tasks of map building and robot localization (Durrant-Whyte, H. & Bailey, T., 2006; Durrant-Whyte, H. & Bailey, T., 2006a).

From its early beginning, SLAM has been implemented in several solution algorithms, depending on the application of the mobile robot task. The Extended Kalman Filter (EKF) solution is the most widely approach used in the scientific community (Ayache, N. & Faugeras, O., 1989; Chatila, R. & Laumond, J. P., 1985), though it is restricted to modeling every variable as a Gaussian random variable. The Particle Filter (PF) and the Unscented Kalman Filter (UKF) have proven to be better approaches to the SLAM problem. The Particle Filter solves the Gaussian restriction of the variables involved in the SLAM models (Thrun, S. et al., 2005) whereas the UKF has shown a better performance dealing with non-linear models of the vehicle and the measurements (Thrun, S. et al., 2005). Nevertheless, both algorithms are computer-time heavy.

The model of the environment built by the SLAM algorithm represents the way the robot senses and recognizes the environment, depending on the sensor type mounted on the vehicle. The map can be either topological, metrical or hybrid (Ayache, N. & Faugeras, O., 1989; Kouzoubouv, K. & Austin, D., 2004), being the metric feature-based map the most widely used technique (Andrade-Cetto, J. & Sanfeliu, A., 2002). The extracted features usually consist on some geometrical constrain of the environment, such as lines, corners or curves. These features are used to localize the robot within the environment.

Since the SLAM algorithm is a dimensionally dynamic algorithm, its complexity grows as the number of observed features increases. Thus, the sampled time of the entire algorithm is variable. This situation is a hindrance when the SLAM is combined with some planned-based control strategy. The reason for this is that, during the SLAM process, the vehicle keeps operating in an open loop control. Some approaches have been proposed to solve this problem (Castellanos, J. et al., 2007) for real time implementation, usually sacrificing the map’s resolution (Castellanos, J. et al., 2007; Guivant, J. E. & Nebot, E. M., 2001) or, else, they may lack a plan-based control strategy (Andrade-Cetto, J. & Sanfeliu, A., 2002).

Some approaches concerning a feature selection restriction in the SLAM algorithm can be found in the scientific literature. In (Zhang, D. et al., 2005) a restriction
based on the entropy analysis of the feature is made before updating the SLAM system state. This way, if the entropy is higher than a certain threshold, then it is considered for updating the system state. In the visual SLAM presented in (Fintrop, S. et al., 2006), the algorithm detects regions with strong contrast while rejecting the remaining areas. In (Eade, E. & Drummond, T.W., 2006), a monocular SLAM based on edge detection is presented. The edges are selected according to the order of their geometrical information. Thus, the edges with higher order of geometrical information are used in the SLAM process. In (Lee, Y.J. & Song, J.B., 2007), a visual SLAM algorithm based on object recognition is proposed. In this paper, a certain number of objects are recognized, classified and then used in the SLAM process. Excepting the paper by (Zhang, D. et al., 2005), the remaining works found in (Durrant-Whyte, H. & Bailey, T., 2006; Durrant-Whyte, H. & Bailey, T., 2006a). The mobile robot used in Whyte, H. & Bailey, T., 2006) represents the contribution of this paper—

The paper is organized as follows. Section 2 shows an introduction to the proposed SLAM algorithm, and two feature selection SLAM approaches. One is based on the entropy of the extracted features, while the other approach —representing the contribution of this paper— is based on the sensibility of the covariance matrix of the SLAM system state. Section 3 presents the comparison results between the two feature selection SLAMs and the experimental results with the entire system implemented. Finally, section 4 shows some conclusions of this work.

2. Methods

In this section, the general EKF-SLAM used here is presented, along with two procedures to select the features of the SLAM correction stage. A first approach is based on the entropy analysis of the environment features, and the other one is based on the sensibility of the covariance matrix of the SLAM system state.

2.1. EKF-SLAM Algorithm

The SLAM algorithm solved by an EKF is stated by Eqs. (1) - (5). All variables involved in the estimation process are considered as Gaussian random variables.

**Prediction Stage**

\[ \hat{x}_{t|t-1} = f(\hat{x}_{t-1|t-1}, u_t) \]  

(1)

**Correction Stage**

\[ K_t = P_t H_t^T (H_t P_t H_t^T + R_t)^{-1} \]  

(3)

\[ \hat{z}_t - \hat{x}_{t|t-1} + K_t(z_t - h(\hat{x}_{t|t-1})) \]  

(4)

\[ P_t = (I - K_t H_t) P_t \]  

(5)

In (1) - (2) the expression for the prediction stage of the EKF is presented. \( \hat{x}_{t|t-1} \) is the predicted state of the system at time \( t \); \( u_t \) is the input control commands and \( \hat{x}_{t|t-1} \) is the corrected state at time \( t \); \( f \) describes the motion of the elements of \( \hat{x} \). \( P_t \) and \( P_{t-1} \) are the predicted and corrected covariance matrices respectively, at time \( t \); \( A_t \) is the Jacobian of \( f \) with respect to the SLAM system state and \( Q_t \) is the covariance matrix of the noise associated to the process, whereas \( W_t \) is its Jacobian matrix. Equations (3) - (5) represent the correction stage of the EKF-SLAM. \( K_t \) is the Kalman gain of the algorithm at time \( t \); \( H_t \) is its Jacobian matrix of the measurement model (\( h \) in (4)) and \( R_t \) is the covariance matrix of the actual measurement \( z_t \). In (4), \( (z_t - h(\hat{x}_{t|t-1})) \) is called the innovation vector (Thrun, S. et al., 2005) and takes place when the data association procedure has reached an appropriate matching between the observed feature and the predicted one (\( h(\hat{x}_{t|t-1}) \)). Both, the process model (\( f \) in (1)) and the observation model are non-linear expressions. In this work, the features extracted from the environment correspond to lines and corners, either concave or convex.

The SLAM system state is composed by both the vehicle’s degrees of freedom and the parameters that define all features, considered as stationary. According to this, the motion computed in (1) corresponds to the robot’s movements. More information about this topic can be found in (Durrant-Whyte, H. & Bailey, T., 2006; Durrant-Whyte, H. & Bailey, T., 2006a). The mobile robot used in this work is a non-holonomic unicycle type -Pioneer 3DX, built by ActivMedia®. Figure 1 shows a scheme of the vehicle and (6) represents their kinematic equations. The robot has a range sensor laser -built by SICK®- which acquires 181 measurements, from 0 to 180 degrees, within a 30-meter range. The models of the features are shown in equations (7) - (8), and Fig. 2 shows the geometrical meanings of each variable used in (7) and (8). More information about features extraction and the SLAM

![Fig. 1. Schematic of the mobile robot used.](image_url)
Let $N_t$ be the set of observed features

Let $M_t \subseteq N_t$ be the set of features with correct matching

for $i = 1 : \# M_t$

(iii)

$$K_i = P_t^{-1} H_t^T (H_t P_t^{-1} H_t^T + R_t)^{-1}$$

(iv)

$$\hat{z}_t^i = \hat{z}_t^i + K_i (z_t - h(\hat{z}_t^i))$$

(v)

$$P_t = (I - K_t H_t) P_t^{-1}$$

(vi)

$$M_t = M_t - \{ z_t \}$$

(vii)

$$P_t^- := P_t, \quad \hat{z}_t^- := \hat{z}_t$$

(viii)

end for

(ix)

Algorithm 1. Sequential EKF-SLAM without Feature Selection.

2.2. SLAM with feature selection: Entropy Approach

The SLAM algorithm with feature selection based on the observation of the entropy of the measurements was previously presented by (Zhang, D. et al., 2005). This algorithm is considered as related to the one proposed in this paper and will be used for comparison purposes.

This method is based on the calculation of the entropy attached to each observed feature. If the entropy is below a certain threshold value, then the observation will be computed in the correction stage of the EKF.

Considering that all variables involved in the EKF-SLAM estimation process are Gaussian random variables, the entropy value associated with a single observation can be represented as in (9).

$$\Sigma_t = E[-\ln p(\hat{z}_t^i | z_t)] = 0.5 \ln \left( \frac{2 \pi e}{| P_t |} \right)$$

(9)

Here $\Sigma$ is the entropy of the observation $z$. The a priori and posteriori information metric can be defined as the inverse of the entropy value, shown in (9).

Thus,

$$\im_t = -\Sigma_t = -0.5 \ln \left( \frac{2 \pi e}{| P_t |} \right)$$

(10)

$$\im_t^- = -\Sigma_t^- = -0.5 \ln \left( \frac{2 \pi e}{| P_t^- |} \right).$$

(11)

The information difference can be calculated as in (12), where the absolute incremental information is obtained.

$$\Delta_t = \im_t - \im_t^-$$

(12)

Thus, when the absolute information of a feature exceeds a certain threshold ($\delta$), that feature will be used in the correction stage of the EKF-SLAM algorithm. The complete algorithm can be summarized as follows.

Let $N_t$ be the set of all $z$ at time $t$

Let $M_t \subseteq N_t$ be the set of all $z$ with appropriate matching at time $t$

$\forall z_t \in M_t$, $P_t | z_t$ is calculated according to (5)

If $\Delta_t \geq \delta$, then the update (Eq. (4)) takes place.

Algorithm 2. Entropy feature selection procedure.

---

Fig. 2. Feature extraction.
As Algorithm 2 shows, the calculation of the entropy associated with a single observation -and its information metric- is related to the determinant of the complete covariance matrix of the SLAM system state. Thus, the complexity of the calculation of the entropy is \( O(n^2) \), where \( n \) is the dimension of the SLAM system state. Since this dimension varies, the complexity of the algorithm varies as well.

Although this algorithm has the advantage of restricting the number of features to be updated, the benefits in processing time are shown only in the updating part of the EKF-SLAM (Eq. (4)). The reason for this is that the calculation of the entropy requires a previous calculation of both the covariance matrix (Eq. (5)) and its determinant. Furthermore, when the entropy is calculated, the dimensions of the covariance matrix and its determinant are equal to \( n \) (see (2)).

Another convergence property states that, at the limit, all elements of \( P_t \) become fully correlated (Castellanos, J. et al., 2004). This last statement is equal to say that,

\[
\lim_{t \to \infty} |P_t| = 0. \tag{16}
\]

Statement (15) defines the sensibility of the covariance matrix of the SLAM system state measured from the determinant of such a matrix. Thus, according to (15) and (16), given a set of observed features with correct matching, the feature that causes the highest decrease of \( |P_t| \), is the feature to which the EKF-SLAM is more sensitive and will cause the fastest convergence of (16). This latter point can be regarded as an optimization problem. Let \( N_t \) be the set of observed features at time \( t \); let \( M_t \subseteq N_t \) be the set of features with correct matching. Then, \( \forall z \in M_t \subseteq N_t \):

\[
z^{\text{opt}} = \arg \min \{||K_t H_t|| \} = \arg \min \{|(I - KH_t)|| \}. \tag{17}
\]

Thus, according to (17), finding the observation \( z \) that minimizes \( |I - KH_t| \) is equivalent to finding the observed feature that causes the highest decrease of \( |P_t| \). This is so because \( |P_t| \) is independent of the current observation (see (2)).

Considering now that the EKF-SLAM implemented in this work is a sequential algorithm (Thrun, S. et al., 2005), the Jacobian of the observation model has the form shown in (18), where \( H_{z,t} \) is the Jacobian of the observation model with respect to the vehicle's variables and \( H_{v,t} \) is the Jacobian of the observation model with respect to the parameters of the observed feature. The Kalman gain -(19)- is also defined according to (18).

\[
H_t = \begin{bmatrix} H_{v,t} & \Theta_1 & H_{z,t} & \Theta_2 \end{bmatrix} \tag{18}
\]

\[
K_t = \begin{bmatrix} k_{v,t}^T & k_{v,1}^T & k_{z,t}^T & k_{z,1}^T \end{bmatrix}^T \tag{19}
\]

In (18), \( \Theta \) is a null matrix. Thus, the Jacobian of the observation is only calculated on the Jacobian entries that correspond to the vehicle and to the feature with correct matching. More information about this topic can be found in (Durrant-Whyte, H. & Bailey, T., 2006; Durrant-Whyte, H. & Bailey, T., 2006a). Using (18) and (19), the expression \( |I - KH_t| \) can be calculated as:

\[
|I - KH_t| = \begin{bmatrix} I_v & I_{v,1}^T & k_{v,1}^T \Theta_1 & k_{v,1}^T H_{z,t} & \Theta_2 \\ I_{z,1} & I_{z,1,1} & I_{z,1,1} & I_{z,1,1} \end{bmatrix} \begin{bmatrix} I_v - K_{v,t} H_{z,t} & 0 & -K_{v,1} H_{z,t} & \Theta_1 \\ -K_{v,1} H_{z,t} & I_{v,1} & -K_{v,1} H_{z,t} & \Theta_2 \\ I_{z,1} - K_{v,1} H_{z,t} & 0 & I_{z,1} & \Theta_1 \\ I_{z,1} - K_{v,1} H_{z,t} & 0 & I_{z,1} & \Theta_1 \\ -K_{v,1} H_{z,t} & 0 & -K_{v,1} H_{z,t} & I_{v,1} \\ I_{z,1} - K_{v,1} H_{z,t} & 0 & -K_{v,1} H_{z,t} & I_{v,1} \end{bmatrix} \tag{20}
\]

\[
|I - KH_t| = \begin{bmatrix} I_v - K_{v,1} H_{z,t} & 0 & -K_{v,1} H_{z,t} & \Theta_1 \\ -K_{v,1} H_{z,t} & I_{v,1} & -K_{v,1} H_{z,t} & \Theta_2 \\ I_{z,1} - K_{v,1} H_{z,t} & 0 & I_{z,1} & \Theta_1 \\ I_{z,1} - K_{v,1} H_{z,t} & 0 & I_{z,1} & \Theta_1 \\ -K_{v,1} H_{z,t} & 0 & -K_{v,1} H_{z,t} & I_{v,1} \\ I_{z,1} - K_{v,1} H_{z,t} & 0 & -K_{v,1} H_{z,t} & I_{v,1} \end{bmatrix} \]

Equation set (15) implicitly defines the convergence property of the EKF-SLAM, which is: the determinant of the correction covariance is always smaller than or equal to the determinant of the predicted covariance. The determinant of the covariance matrix, in this case, is used as a measure of the volume of the uncertainty ellipse associated with that matrix (Castellanos, J. et al., 2004).
In (20), I is the identity matrix, \( I_{x}, I_{y}, I_{z} \) and \( I_{o2} \) are identity block matrices with the dimensions of \( K_{x}H_{x,t}, K_{y}H_{y,t}, K_{z}H_{z,t} \) and \( K_{o2}H_{o2,t} \) respectively. Thus, considering that the vehicle has three degrees of freedom and a feature -corner or line- is determined within two parameters -see (7) and (8)-, the final calculation in (20) is a determinant of a \( 5 \times 5 \) matrix.

The final correction stage algorithm implemented in this work is formalized in Algorithm 3.

Let \( N_{i} \) be the set of observed features

Let \( M_{i} \subseteq N_{i} \) be the set of features with correct matching

Let \( LIM \) be the maximum number of features to be used in the correction stage

for \( i = 1: \min(LIM, \# M_{i}) \)

\[
\text{find } z_{i}^{opt} : \arg \min \{ P_{i} | \# M_{i} \} \]

\[
K_{i} = P_{i} H_{i}^{T} (H_{i} P_{i} H_{i}^{T} + R_{i})^{-1} \]

\[
\hat{\xi}_{i}^{opt} = \hat{\xi}_{i} + K_{i} \left( z_{i}^{opt} - H_{i} \hat{\xi}_{i} \right) \]

\[
P_{r} = (I - K_{i} H_{i}) P_{r} \]

\[
M_{i} = M_{i} - \{ z_{i}^{opt} \} \]

\[
P_{r}^{c} = P_{t} \]

end for

Algorithm 3. Feature Selection based on Covariance Sensibility

In Algorithm 3, sentences (i) - (ii) are the declaration of the domain that is going to be used in the correction stage; sentence (iii) determines -if possible- the maximum number of features that will be used for correcting the SLAM. If the number of features in \( M_{i} \) is smaller than \( LIM \), then the complete set of features in \( M_{i} \) will be used in the correction loop. Sentences (iv) - (xi) show the for-loop of the correction stage. Given \( M_{i} \), the algorithm searches for a first \( z_{i}^{opt} \). When it is found, the correction stage takes place -(v) to (viii)- and this feature is removed from \( M_{i} \). In the second iteration of the for-loop, the \( z_{i}^{opt} \) is searched inside the new \( M_{i} \) and the actual predicted covariance is the last corrected system state covariance matrix as shown in sentence (x). This situation ensures that sequentiality of the EKF-SLAM is not lost.

Comparing Algorithm 2 with Algorithm 3 it is possible to see that Algorithm 3 does not compute the update of the SLAM covariance system state until a \( z_{i}^{opt} \) is found and only for a limited number of \( z_{i}^{opt} \). On the other hand, Algorithm 2 must calculate the correction of the covariance matrix in order to obtain the entropy value. This situation requires the system to execute \( O(n^{2}) \) iterations for every feature -besides the \( O(n^{2}) \) computational cost required for the calculation of the determinant of \( P_{t} \).

Let \( n \) be the number of features with correct matching in Algorithm 1 (\( \# M_{i} = n \)). Then, the determinant of the final state covariance matrix can be calculated as

\[
\begin{align*}
| P_{r} | & = \prod | P_{i} | | I - K_{i} H_{i} | \prod P_{i}^{r} \\
& = \prod | P_{i} | | I - K_{i} H_{i} | \prod P_{r}^{c} \\
& = \prod | P_{i} | | I - K_{i} H_{i} | \prod P_{r}^{c} \\
& = \prod | P_{i} | | I - K_{i} H_{i} | \prod P_{r}^{c} \\
& \Rightarrow | P_{r} | = | I - K_{i} H_{i} | \times \cdots | I - K_{i} H_{i} | \prod P_{r}^{c} .
\end{align*}
\]

In addition, in Algorithm 3, let us suppose that the maximum number of features to be used in the correction stage is \( LIM = 2 \) and that \( \# M_{i} > 2 \). Then, the determinant of the final system state covariance matrix can be calculated as

\[
\begin{align*}
| P_{r} | & = \prod | P_{i} | | I - K_{i} H_{i} | \prod P_{r}^{c} \\
& = \prod | P_{i} | | I - K_{i} H_{i} | \prod P_{r}^{c} \\
& \Rightarrow | P_{r} | = | I - K_{i} H_{i} | \times | I - K_{i} H_{i} | \prod P_{r}^{c} .
\end{align*}
\]

Then, considering that \( | I - K_{i} H_{i} | \leq 1, i = 1 \ldots n \), from Eqs. (21) and (22),

\[
\frac{| P_{r} |}{| P_{r} |_{\text{Algorithm 3}}} \leq | P_{r} |_{\text{Algorithm 3}}
\]

Then, from Eqs. (21) to (23), the following expression holds,

\[
\frac{| P_{r} |_{\text{Algorithm 1}}}{| P_{r} |_{\text{Algorithm 3}}} \leq | I - K_{i} H_{i} | \times \cdots | I - K_{i} H_{i} | \leq 1 .
\]

Equation (24) can be used as a measurement means for the accuracy of the system state estimation using the EKF-SLAM with feature selection based on covariance sensibility, with respect to the classical sequential EKF-SLAM. Then, the higher the ratio shown in (24), the better the system state estimation made by the EKF-SLAM with the feature selection procedure proposed in this work.

3. Implementation Results

Several experiments were carried out in order to show the performance and benefits of the EKF-SLAM with feature selection criteria based on the covariance matrix sensibility proposed in this work. The computational cost of this algorithm was also compared to the sequential EKF-SLAM and the entropy approach EKF-SLAM shown in sections 2.2 and 2.3 respectively. All SLAM algorithms worked in real time. The mobile robot motion strategy used in the experiences was a local navigation based on frontier points determination (Tao, T. et al., 2007). Considering that the SLAM provides an estimate of the robot’s pose to the control algorithm, only one SLAM algorithm can be executed at a time. The following sections show the adopted control strategy in detail, the SLAM consistence analysis and the computational costs of each SLAM algorithm.

3.1. Mobile Robot Motion Strategy

The motion strategy used in this work obeys the concept of Active SLAM introduced by (Tao, T. et al., 2007). Thus, the robot’s pose estimation is used to generate the controller references in order to drive the robot. In this
The mobile robot first searches for an unoccupied space at the frontier of its range sensor measurements (frontier points). Once a frontier point is selected, a straight trajectory is generated from the robot’s position to the frontier point. A trajectory controller is then executed to reach a neighborhood of the frontier point. This controller is an hybrid dynamic adaptive controller presented in (De la Cruz, C. et al., 2008).

Once the robot reaches a neighborhood of the destination point, a next frontier point is searched. If no frontier point is detected, then the robot turns around until it founds a frontier point. If during the trajectory an obstacle is detected, the robot avoids it using a tangencial avoidance criteria (Santos Brandao, A. et al., 2008) without losing its destination objective.

The SLAM algorithm keeps being active during the control execution.

Figure 3 shows the Active SLAM architecture implemented in this work. The localization and mapping procedures are carried out at the global reference frame, i.e., the system state variables are defined at this global frame. Nevertheless, the EKF-SLAM is executed at a local reference frame level. Thus, the data association and the features’ prediction and extraction are taken into account at the local reference frame attached to the robot. The robot avoids it using a tangencial avoidance criteria (Santos Brandao, A. et al., 2008) without losing its destination objective.

The SLAM algorithm keeps being active during the control execution.

Figure 3 shows the Active SLAM architecture implemented in this work. The localization and mapping procedures are carried out at the global reference frame, i.e., the system state variables are defined at this global frame. Nevertheless, the EKF-SLAM is executed at a local reference frame level. Thus, the data association and the features’ prediction and extraction are taken into account at the local reference frame attached to the robot. The robot avoids it using a tangencial avoidance criteria (Santos Brandao, A. et al., 2008) without losing its destination objective.

The SLAM algorithm keeps being active during the control execution.

3.2. Consistence Analysis

Although the EKF-SLAM with the feature selection criteria proposed in this work does not imply a modification of the EKF, a consistence analysis was carried out for this SLAM. The consistence analysis shows that, when the EKF correction stage is restricted to those features that contribute the most for SLAM system state covariance matrix convergence (see Eq. (14)), the map of the environment does not become inconsistent.

In order to show consistence, the true location of the robot in the environment is needed for comparison purposes. Figure 4.a shows the simulated environment with the expected trajectory. The environment was created using the MobileSim® tools (provided by ActivMedia®). Figure 4.b shows the map reconstruction and the actual path travelled by the robot. The mobile robot drives according to the control strategy presented in the previous section. The green circles (○) correspond to corners of the environment; the blue points represent the travelled path by the mobile robot; the red crosses (×) are the beginning and ending points of the segments associated with the lines of the environment. Although the SLAM algorithm presented in this work has the parameters of lines and corners in its system state, the beginning and ending points of each line are stored in a secondary map. This map is updated according the SLAM system state updating. Finally, the solid black lines in Fig. 4.b are the segments associated to walls of the environment. Figures 4.c and 4.d show that the robot’s position error remains bounded by 2(two times the standard deviation).

3.3. Experimental Results

According to section 3.1, the mobile robot trajectory controller uses the pose estimation of the mobile robot for generating the motion references. This scenario implies that only one SLAM algorithm is able to perform its estimation at a time. Therefore, three similar experiments were made in the facilities of the Instituto de Automatica. For each experience, the robot was placed in the same initial position in the building. Figure 5.a-c shows the map reconstruction of the places traversed of the Instituto deAutomatica according to the three EKF-SLAM considered in this paper (sequential EKF-SLAM, entropy feature selection approach and the covariance sensibility approach). In Fig. 5, the green circles represent corners of the environment, the blue points are the path travelled by the robot, the red crosses are the beginning and ending points of the segments of the environment, whereas solid black lines are the segments associated with lines and the yellow points are raw laser data. The accumulated processing time, as can be seen in Fig. 6, shows that the EKF-SLAM with feature selection based on the SLAM system state with the covariance sensibility approach has shorter computational processing time when compared with the sequential EKF-SLAM and the entropy feature selection approach. The accumulated
Fig. 4. Consistence analysis of the EKF-SLAM with feature selection criteria based on SLAM system state covariance matrix sensibility. a) Simulated environment with expected path; b) map reconstruction based on the SLAM system state. It can be seen that the map is coherent with its model; c) the \( x \)-robot position in the environment is bounded by two times its variance; d) idem for \( y \)-robot position.

Fig. 5. Real time experimental results. a) Map reconstruction of the facilities at the Instituto de Automatica using the sequential EKF-SLAM system state; b) map reconstruction using the EKF-SLAM with entropy feature selection system state, and c) map reconstruction using the EKF-SLAM with covariance sensibility feature selection procedure system state. In the three experiments the robot was left at the same position inside the Instituto de Automatica, although its orientation varied.
Fig. 6. Accumulated SLAM processing time. The EKF-SLAM with the covariance sensibility approach demands lesser processing time per iteration than the classical sequential EKF-SLAM or the entropy feature selection approach.

The processing time shown in Fig. 6 corresponds to the experiments of Fig. 5. After several long-time trials, the accumulated time of the EKF-SLAM with covariance sensibility approach was 33% shorter than the accumulative time of the sequential EKF-SLAM and 47% shorter than the EKF-SLAM with entropy approach. In addition, the accuracy of the map -see Eq. (24)- for the experiment shown in Fig. 5 was 97.2%.

During the SLAM execution, the mobile robot remains in an open loop state. This period depends on the SLAM processing charges, i.e., on the number of features to be corrected (Thrun, S. et al., 2005). Considering that the sampling time of the mobile robot used in this work (Pioneer 3DX) must be a multiple of 0.1 seconds, if the SLAM requires, e.g., 0.45 seconds of processing, then the sampling time for this iteration will be at least 0.5 seconds. During this time of 0.5 second, the command control of the robot does not change, thus being unable to react if an obstacle appears on its path. This crucial navigation problem can be solved in part by adopting a constant time SLAM. In the sequential EKF-SLAM, a constant processing time is not feasible due to the variation of the number of features to be associated. On the other hand, the EKF-SLAM with feature selection based on SLAM system state covariance matrix sensibility is a constant time algorithm due to the restriction on the number of features to be associated. Figure 7 shows the open-loop accumulated time for the three experiments shown in Fig. 5. As noted, the EKF-SLAM with feature selection based on covariance sensibility proposed in this work presents the shortest robot’s open loop time.

After several trials, the EKF-SLAM with feature selection based on covariance sensibility have shown a 45% shorter accumulated open loop time than the sequential EKF-SLAM, and 56% shorter time than the EKF-SLAM with feature selection based on the feature selection entropy approach.

Since the experiments concerning the EKF-SLAM with feature selection based on the covariance matrix shown in this work had a restricted number of correction iterations (see Algorithm 3), the mobile robot sampling time turned to be 0.2 seconds. For the sequential EKF-SLAM it rose to 0.5 seconds (its maximum registered value), and for the EKF-SLAM with feature selection based on the entropy approach it had a maximum sampling time of 0.7 seconds.

Figure 8 shows the map reconstruction of the entire Instituto de Automatica when the robot navigates within its premises using the control strategy presented in this work along with the EKF-SLAM with feature selection based on SLAM system state covariance matrix sensibility. Figure 9 shows the accumulated time when a similar navigation is performed using the classical sequential EKF-SLAM.

Fig. 8. Map reconstruction of the Instituto de Automatica using the proposed feature selection procedure. Green circles (o) represent corners; red crosses (x) are the beginning and ending points of the segments associated with lines; solid black lines represent walls and yellow points are raw laser data.
corridors within the Instituto de Automatica. In both cases, the robot had performed similar paths.

4. Conclusions

A new solution to the sequential EKF-SLAM algorithm based on a feature selection procedure has been proposed in this work. The feature selection method consisted in correcting the EKF according to those features to which the SLAM covariance matrix shows more sensibility, allowing, in that way, EKF convergence. The experiments were made in real time realizations and within a real semi-structured environment. A comparison was also made between the approach presented in this work and a feature selection procedure based on entropy analysis along with the classical sequential EKF-SLAM algorithm. The processing times of the different realizations have shown the advantages of using the approach presented in this paper with respect to the entropy analysis and the classical sequential EKF-SLAM. The EKF-SLAM with feature selection based on the covariance sensibility approach has shown an average accumulated processing time 45% shorter than the sequential EKF-SLAM and 56% shorter than the EKF-SLAM with feature selection based on the entropy approach. The experiments have also shown that the map kept being consistent.

The EKF-SLAM with feature selection based on the covariance sensibility approach has allowed to execute a constant time SLAM in real time realizations.

5. References


