

# Differential Evolution approach to the grid-based Localization and Mapping problem

Luis Moreno, Santiago Garrido, Fernando Martín and M. Luisa Muñoz

**Abstract**—A new solution to the Simultaneous Localization and Modelling problem is presented. It is based on the stochastic search of solutions in the state space to the global localization problem by means of a differential evolution algorithm. A non linear evolutive filter, called Evolutive Localization Filter (ELF), searches stochastically along the state space for the best robot pose estimate. The proposed SLAM algorithm operates in two steps: in the first step the ELF filter is used at a local level to re-localize the robot based on the robot odometry, the laser scan at a given position and a local map where only a low number of the last scans have been integrated. In a second step the aligned laser measures together with the corrected robot poses are use to detect when the robot is revisiting a previously crossed area. Once a cycle is detected, the Evolutive Localization Filter is used again to re-estimate the robot poses in order to integrate the sensor measures in the global map of the environment. The algorithm has been tested in different environments to demonstrate the effectiveness, robustness and computational efficiency of the proposed approach.

**Index Terms**—SLAM, Local Positioning Systems, Mobile robots, Differential Evolution Algorithm

## I. INTRODUCTION

Localization and map building are key components in robot navigation and are required to successfully execute a path generated by a global planner. Both problems are closely linked, and learning maps required to solve simultaneously both problems. These simultaneous problems are often referred as *simultaneous localization and mapping*(SLAM). In the SLAM case, uncertainty in measures, uncertainty in robot pose estimates and a partially learned map which contains the residual errors unsolved at integration or re-localization processes makes the SLAM problem complex.

The two main approaches to effectively solve the simultaneous localization and mapping problem have been consolidated in the last decade. The first approach uses a feature-based model of the environment and the extended Kalman filter (EKF) to manage the associated uncertainty. This approach is extremely compact, and its computational cost has been considerably improved in the most recent work. On the other hand, however, the linear nature of the basic method requires linearization of the motion and perception models which causes problems in the long term. Moreover, the technique has difficulties modelling many environment areas due to the limited set of feature models used.

The second group of solutions use particle filters to obtain a solution to the SLAM problem. This group of

solutions can use certainty grid map models or a feature map to represent the environment [12], and the sequential Monte Carlo methods to estimate the posterior probability distribution functions. This approach has proved to be very robust from a statistical point of view in the management of the uncertainties present in the problem. Its disadvantage is that the number of particles required increases the computational cost and the algorithm robustness is heavily dependent on this because each particle has a statistical significance associated with it.

In this work we present a new solution to the grid-based SLAM problem based on the stochastic search of the best pose estimate. This approach uses a differential evolution method [9] to perturb the possible pose estimates contained in a given set until the optimum is obtained. By properly choosing the cost function, a maximum a posteriori estimate is obtained. This method is applied at local level to re-localize the robot and at global level to solve the data association problem. The method proposed integrates sensor information in the map only when cycles are detected and the residual error are eliminated, avoiding a high number of modifications in the map or the existence of multiple maps, thus decreasing the computational cost compared to other solutions.

Our approach has been validate on a set of data obtained from our experiments and data extracted from the Radish repository data. The results show the proposed method is able to satisfactorily close environment cycles to generate accurate metric maps.

## II. STATE OF THE ART

The SLAM problem has been one of the most interesting theoretical problems in mobile robotics since the 90's when the seminal work of Smith, Self and Cheeseman [8] introduced the concept of a stochastic map to establish uncertain spatial relationship between features detected in the environment. The Kalman filter provides the mechanism for integrating and updating the map. Julier and Uhlmann [3] have shown important conclusions derived from the fact that the linearization problem is structural and cannot be avoided. Castellanos et al. have also shown [1] that linearization errors produce inconsistency problems in the standard EKF solution for SLAM.

A widely investigated aspect is the scaling properties of the stochastic map solution to the SLAM problem. The  $O(n^2)$  complexity of the basic solution (where  $n$  is the number of features in the map) has been a serious bottleneck. Several researchers have developed techniques

Luis Moreno, Santiago Garrido, Fernando Martín are with the Robotics Lab., Carlos III University, Madrid, Spain. [moreno@ing.uc3m.es](mailto:moreno@ing.uc3m.es)

M. Luisa Muñoz is with the Facultad de Informática, Polytechnical University of Madrid, Madrid, Spain.

to alleviate the computational burden: the sparse extended information filters [11], the decoupled stochastic mapping [4], the compressed filter [2], the sequential map joining [10] and the constrained local submap filter [13] have improved the scalability problem of the standard EKF-based version of the SLAM problem. However, the linearization problem and the data association problem ambiguities are not completely solved.

A second group of SLAM algorithms are based on the occupancy grid mapping method. Grid-based techniques are computationally expensive and have a larger memory requirement than EKF-based SLAM methods. This group of solutions uses the Rao-Blackwellized particle filters to estimate the posterior probability distribution functions [5]. The main problem of this approach derives from the fact that each particle of the filter represents a different map of the environment. This requires the obtaining of as many maps as particles being used in the filter at each algorithm step. A second problem, common in Bayesian filters, is the difficulty of estimating an adequate number of particles.

The advantage of our approach is twofold. First, the solution proposed builds only one global certainty grip map, thus avoiding unnecessary operations. Second, the computational efficiency of the method reduces the computational cost.

### III. LOCALIZATION PROBLEM FORMULATION AND SOLUTION.

From a Bayesian point of view, the localization problem can be formulated as a probability density estimation problem where the robot seeks to estimate a posterior distribution over the space of its poses conditioned on the available data. The sensor data can be divided into two groups of data  $Y_t \equiv \{z_{0:t}, u_{1:t}\}$  where  $z_{0:t}$  contains the perception sensor measurements and  $u_{1:t}$  contains the odometric information. The recursive determination of the posterior probability density can be computed in two steps:

- *Measurement update.* Applying the Bayes's rule to the last element of the measurement vector  $Y_t$  and assuming that the observation  $z_t$  is conditionally independent of the previous measurements given the state  $x_t$ , yields

$$p(x_t|Y_t) = \frac{p(z_t|x_t)p(x_t|Y_{t-1})}{p(z_t|Y_{t-1})} \quad (1)$$

where the denominator of 1 is obtained by marginalization.

- *Prediction.* The effect of a time step over the state given the observations up to time  $t$  is obtained by observing that

$$p(x_{t+1}|Y_t) = \int_{\mathbb{R}^n} p(x_{t+1}|x_t, u_t)p(x_t|Y_t)dx_t \quad (2)$$

where the assumption that the process  $x_t$  is Markovian, and then  $x_{t+1}$  is independent of  $Y_t$  has been considered.

Each candidate parameter value in  $\mathbb{R}^n$  yields a value of  $p(x|y)$ , reflecting the posterior probability of the robot pose given the data up to time  $t$ . This posterior needs to be weighted according to a given criterion to determine an

estimate  $\hat{x}$  of the true pose value. Since the posterior probability distribution is multi-modal in the global localization problem, the minimum mean-square estimate is inconvenient. The localization algorithm used concentrates on obtaining the better *maximum a posteriori* estimator,

$$\hat{x}^{MAP} = \arg \max_x p(x_t|Y_t) \quad (3)$$

This approach is less dependent on statistical assumptions, has a simpler implementation, is robust from a statistical point of view and has a computational cost lower than Bayesian methods.

#### A. Localization as a MAP optimization problem

The localization problem is basically an optimization problem, where the robot seeks to estimate the pose which maximizes the *a posteriori* probability density.

$$\hat{x}_t^{MAP} = \arg \max_x \prod_{i=1}^t p(z_i|x_i) \prod_{i=1}^t p(x_i|x_{i-1}, u_{i-1})p(x_0) \quad (4)$$

The *maximum a posteriori* (MAP) estimate expression can be easily stated as an optimization problem subject to constraints (the motion and observation models of the robot).

#### B. Recursive formulation of the optimization problem

To implement the global localization algorithm in a robot, a recursive formulation is required. The objective function can be reformulated in a more convenient form:

$$f_0(x_t) = \log p_e(z_t|x_t) + \log p_v(x_t|x_{t-1}, u_{t-1}) + f_0(x_{t-1}) \quad (5)$$

where  $p_e$  and  $p_v$  express the probability density functions for the observation and motion noise, respectively, and the MAP optimization problem can be written as

$$\max_x \log p_e(z_t|x_t) + \log p_v(x_t|x_{t-1}, u_{t-1}) \quad (6)$$

Then, by perturbing and searching new solutions to 6, we obtain a recursive version of the MAP estimate.

### IV. EVOLUTIVE LOCALIZATION FILTER ALGORITHM

The algorithm proposed to implement the evolvable localization filter is based on the differential evolution method proposed by Storn and Price [9] for global optimization problems over continuous spaces, and Moreno [6] proposed using it to solve the global localization problem by. It utilizes  $N$  parameter vectors as a sub-optimal feasible solutions set (population) for each generation  $t$  of the optimization process.

#### A. Fitness function

According to the recursive optimization problem under consideration, the natural choice for fitness function is the objective function.

The observation probability  $p_e(z_t|x_t)$  can be calculated by predicting the observation value of the noise-free sensor assuming the robot pose estimate is  $\hat{x}_t$ , the sensor relative angle with respect to the robot axis is  $\alpha_i$ , and a given

environment model  $m$ , that in our case is estimated. Let  $\hat{z}_{t,i}$  denote this ideal predicted measurement. Assuming the measurement error  $e_{t,i}$  is Gaussian, centered at  $h(\hat{x}_t, \alpha_i)$  and with a  $\sigma_e$  standard deviation, and conditional independence between the individual measurements, the individual sensor beam probabilities are integrated into a single probability value:

$$p_e(z_t|\hat{x}_t) = \prod_{i=0}^{N_s} \frac{1}{(2\pi\sigma_e^2)^{1/2}} e^{-1/2 \frac{(z_{t,i} - \hat{z}_{t,i})^2}{\sigma_e^2}} \quad (7)$$

where  $N_s$  is the number of sensor observations.

The second probability required to calculate the objective function can be obtained in two steps: the prediction of the noise free value of the robot pose assuming the robot pose estimate is  $\hat{x}_t$  and the motion command at  $t$  is  $u_t$ . Assuming the motion error is Gaussian with covariance matrix  $P$ , then this probability can be expressed as

$$p_v(x_t|x_{t-1}, u_{t-1}) = \frac{1}{\sqrt{|P|(2\pi)^n}} e^{-1/2(x_t - \hat{x}_t)P^{-1}(x_t - \hat{x}_t)^T} \quad (8)$$

Introducing the expressions of  $p_v$  and  $p_e$  into the objective function to optimize at iteration  $t$ , it can be reduced to minimize the following function

$$f'_0(x_t) = \sum_{i=0}^{N_s} \frac{(z_{t,i} - \hat{z}_{t,i})^2}{2\sigma_e^2} + \frac{1}{2}(x_t - \hat{x}_t)P^{-1}(x_t - \hat{x}_t)^T \quad (9)$$

To predict the  $\hat{x}_t$  position, it is used the odometry information and the best estimate of the previous iteration in the algorithm. The odometry information is also used to move the population set for one cycle to the following one.

### B. Evolutive Localization Filter algorithm

The Evolutive Localization Filter (ELF) uses  $n$  dimensional parameter vectors  $x_i^k$  to point each candidate solution  $i$  to the optimization problem at iteration  $k$  for a given time step  $t$ . The filter will generate new set of parameter vectors by perturbing an existing vector through the addition of one or more weighted difference vectors to it. The perturbation scheme generates a variation  $v$  according to the following expression,

$$v_i^k = x_i^k + F(x_{r_2}^k - x_{r_3}^k) \quad (10)$$

where  $x_i^k$  is the parameter vector to be perturbed at iteration  $k$ ,  $x_{r_2}^k$  and  $x_{r_3}^k$  are parameter vectors chosen randomly from the population and are different from running index  $i$ . The  $F$  constant is an amplification factor that, in the experiments, we assigned the value 0,5.

To increase the diversity of the new generation of parameter vectors, a crossover mechanism is introduced. Denoted by  $u_i^k$  the new parameter vector with

$$u_{i,j}^k = \begin{cases} v_{i,j}^k & \text{if } p_{i,j}^k < \delta \\ x_{i,j}^k & \text{otherwise} \end{cases} \quad (11)$$

where  $p_{i,j}^k$  is a randomly chosen value from the interval  $[0, 1]$  for each parameter  $j$  of the population member  $i$  at step  $k$  and  $\delta$  is the crossover probability and constitutes the crossover

control variable. The random values  $p_{i,j}^k$  are made anew for each trial vector  $i$ .

To decide whether or not vector  $u_i^k$  should become a member of generation  $i + 1$ , the new vector is compared to  $x_i^k$ . If vector  $u_i^k$  yields a better value for the objective fitness function than  $x_i^k$ , it is replaced by  $u_i^k$  for the new generation; otherwise, the old value  $x_i^k$  is retained for the new generation.

In accordance with the previous ideas, the basic evolutive localization filter algorithm consists of the following steps:

- *Step 1: Initialization.*

The initial set of solutions is calculated and the fitness value associated with each of the points in the state space is evaluated. If no information about initial position is available, the initial set of pose solutions is obtained by drawing the robot poses according to a uniform probability distribution over the state space. The initial robot pose estimate is fixed at an initial value.

- *Step 2: Evolutive search.*

- (a) For each element of the set of robot pose solutions and according to the map, the expected sensor observations are obtained. The expected observations, the sensor observations, the robot pose estimate and the robot pose element are used to evaluate the loss function for each robot pose element in the set of solutions.

- (b) A new generation of perturbed pose solutions are generated. For each new solution, the expected observations are calculated and the loss function evaluated. If the new trial solution results in a better loss function, this solution is selected for the following iteration, otherwise the original is maintained.

- (c) The crossover operator is applied to the resultant population (solutions set).

- (d) The robot pose element of the set with lower value of the loss function is marked as best robot pose estimate. Go to step 2b a given number of iterations.

- *Step 3: Updating.* The best pose of the population is used as the updated state estimate and then used in state transition model to predict the new state according to the odometry information. Next, the displacement is evaluated and the whole population is moved according to this displacement. Then go to step 2.

## V. DIFFERENTIAL EVOLUTION APPROACH TO SLAM PROBLEM

In the simultaneous localization and mapping problem case, the map needs to be estimated concurrently with the pose to generate the expected measurements.

The most extended Bayesian formulation for the SLAM problem is to estimate the posterior probability of the trajectories, the map associated to it given the observations  $z_{1:t}$

and the odometry measurements  $u_{0:t}$ :

$$p(x_{1:t}, m_1^t | u_{0:t}, z_{1:t}) = p(m_1^t | x_{0:t}, z_{1:t}) p(x_{1:t} | z_{1:t}, u_{0:t}) \quad (12)$$

In this expression, the posterior over maps can be easily computed when  $x_{0:t}$  and  $z_{1:t}$  are known by means of a mapping process. Obtaining the posterior over the trajectories is much more complex. At each iteration step the probability associated to augmented states, including a candidate sequence and the map obtained by fusing the sensor data with this sequence, is calculated. The map associated to a given sequence is obtained by fusing the observed data up to time  $t$ ,  $z_{1:t}$  according the estimated robot's pose,  $\hat{x}_{0:t}$ . For notation convenience we express it as  $\hat{m}_0^t$ .

The method proposed is quite different in that it exploits the ELF algorithm capability to operate at local and global level. The algorithm uses a two step approach. The first step exploits the local data coherency idea to partially eliminate the model inconsistency problem by using *local models* to re-localize the robot. This local data coherency is used to estimate the robot's pose precisely but cannot completely eliminate the pose error, and the residual remaining error is accumulated over motion.

The second step of the algorithm exploits a different concept to avoid the global inconsistency problem. The key idea is to delay the sensor data integration in the *global map* until the residual error is eliminated. The residual error can be eliminated when a loop is detected and the global data association done. Both ideas are combined in the solution proposed to obtain an accurate global model based on the application of the Evolutive Localization Filter at local and global level.

#### A. Local data association

One way to eliminate the fast degradation of the estimation process, and consequently of the mapping process, is to re-localize the robot by using a local map which integrates the last  $n$  observations perceived by the robot. This process constitutes a local data association and the pose estimated is referred to as  $\hat{x}_t^L$ .

$$\hat{x}_t^L \approx \arg \max_x p(z_t | x_t, \hat{m}_{t-n}^{t-1}) p(x_t | x_{t-1}, u_{t-1}) \quad (13)$$

This solution allows the robot to be re-localized efficiently and avoids pose estimate degradation. In figures 1 and 2 the re-alignment effect on the robot's poses for the environment used in test 1 (section VI) can be observed.

#### B. Global data association

According to the previous notation, we can also estimate the robot's pose by using the global map up to a given time to obtain the *maximum a posteriori* estimate over the full pose sequence. This we refer to as  $\hat{x}_t^G$ .

$$\hat{x}_t^G = \arg \max_x p(z_t | x_{0:t}, \hat{m}_0^{t-1}) p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | Y_{t-1}, \hat{m}_0^{k-1}) \quad (14)$$

The objective of the global data association is to estimate the residual error accumulated in a loop. Figure 3 shows

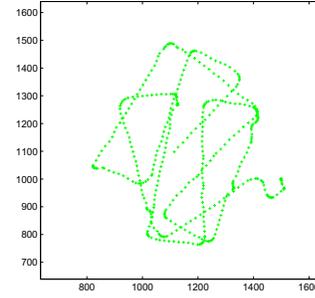


Fig. 1. Initial odometry poses for test environment 1

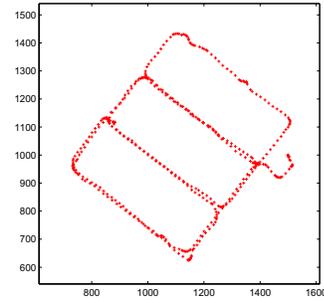


Fig. 2. Re-aligned poses for test environment 1

the robot's pose at the end of a detected cycle (test example number 1) and the new pose estimate according the observed data and the global map modelled up to this moment.

#### C. Residual error correction

Assuming the modelled global map up to a given time  $t$  is  $\hat{m}_0^{k-1}$ , a combined test based on the distance between  $x_t$  and points in the pose sequence, and the overlapping area between sensor scanned area and the modelled global map is used. The error is proportionally attributed to poses in the loop, and a new re-localization is done for each robot's pose using the global map up to that moment instead of the local environment.

Using these new estimates, to integrate observations in the map, it is possible to maintain the robot properly localized and to maintain a globally coherent representation of the environment observed along the robot motion up to time  $t$ .

#### D. ELF based SLAM algorithm

According to previously explained ideas, the ELF grid map based SLAM algorithm operates as follows: if the sensor data has been obtained directly from a laser scanner and using only the odometry, a first re-alignment step is done.

Once the robot's poses have been realigned, a second step is executed to obtain the global map and the globally corrected pose sequence. The global mapping algorithm steps are the following:

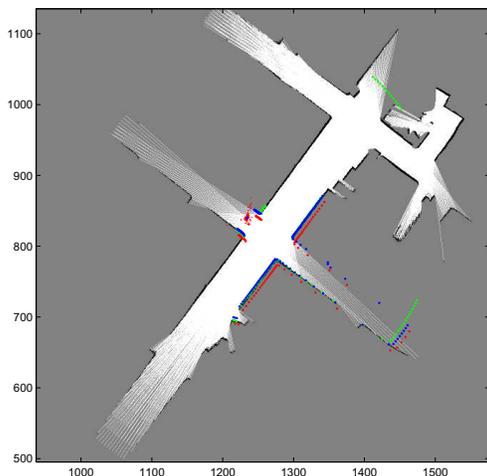


Fig. 3. Global pose estimate at the end of the first detected loop. (red points indicates the population set, and the observed measures before the global pose correction. The blue ones are the observed measures after the global pose correction and the green ones are the estimated measures in the global pose estimate.

- 1) Index is initialized to 1, and the first scan is integrated into the global map.
- 2) The following poses in the sequence are compared with the current pose. If the distance between them is below a given threshold or if there is an overlap between current scan area and the global map, a potential cycle exists. Otherwise, the next pose is explored until a cycle is detected.
- 3) A global localization is started around this pose, and the number of elements in the solution set is related to the area to be explored.
  - If the fitness value of the new pose estimate is below an acceptance threshold, the new pose is accepted.
  - The accumulated residual error is distributed proportionally between all poses along the cycle to correct the accumulated error at the end of the detected cycle.
  - For all poses in the loop:
    - The ELF algorithm is started, limiting its search area to a small area around the corrected pose. This time the global map is used. This readjusts the estimate around the corrected pose and the updated global map.
    - With the corrected pose the new scan data of the detected cycle are integrated into the global map.
- 4) If no cycle is detected but the area has been previously modelled, a re-localization is done, the error corrected and the scan integrated in the global existent map.
- 5) Increment current pose and go to step 2.

## VI. EXPERIMENTAL RESULTS

To demonstrate the algorithm, two different test environments have been used. The data for the first test environment have been obtained from the Robotics Data Set Repository (Radish) [7]. They are part of the Intel Jones Farms Campus (Oregon) and were provided for M. Batalin (we thank him).

In Figure 3, the first cycle is detected in the mapping process, the cycle starts in pose 44 and finishes in pose 149. Figure 3 shows the pose at the end of the cycle, the new pose estimate in the global data association, together with the laser measures at original and re-estimated poses and the measures prediction at re-estimated pose. After the accumulated error is distributed between all poses included in the cycle and re-estimated around this new position according to the global map, the global data integration process in the global map is done, and the result is shown in figure 4, together with the next scan.

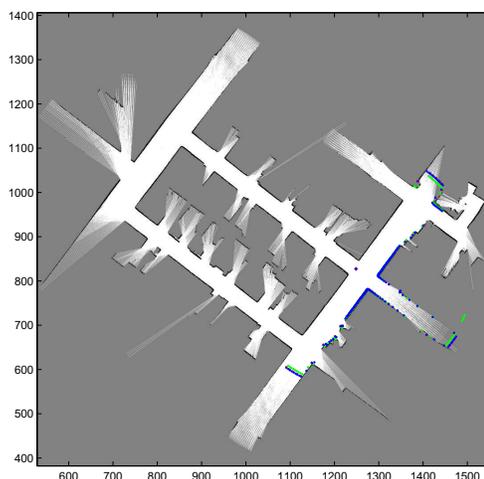


Fig. 4. Global map after error attribution and data fusion at the end of the first detected loop

Figure 5 shows the state of the global map after integrating 451 scans. Four cycle situations have been detected in addition to the revisiting of previously crossed areas.

A second test has been done with data obtained from experiments developed at our University laboratories, offices and corridors. The test site is around 60 meters length. The data scans have been obtained each 80cm approximately using a stop and go method. In this test example there are no cycles present, but there is a high degree of symmetry.

Figure 6 shows the global map up to motion 104.

An interesting aspect of the algorithm is the low number of pose solutions required in the population to achieve satisfactory results. A more detailed consideration of the accuracy and convergence properties of the localization algorithm can be found in [6].

The complexity of the ELF algorithm is  $O(N.M)$ , where  $N$  is the population size and  $M$  is the iteration number.

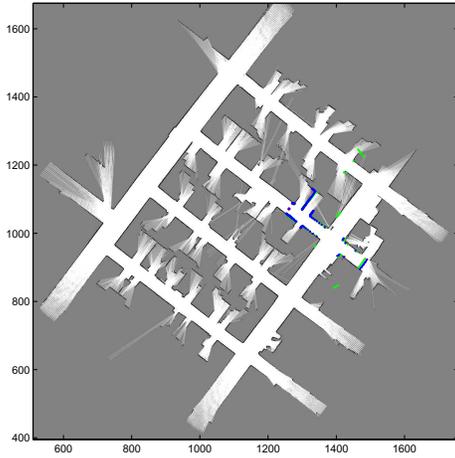


Fig. 5. Global map after 451 scans fusion in environment test 1

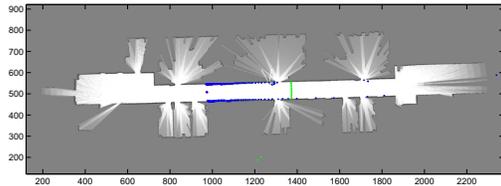


Fig. 6. Global map after 103 scans fusion in environment test 2

In spite of this complexity appearing to be substantial, in practical terms it is very moderate. This is because only in the case of a loop detection is the global data association started, and in case of local correction, the population set is very low.

The initial realignment done left an estimated residual error lower than 0.25 degrees in orientation and lower than 1 centimeter in position (on average). This error is eliminated in the global data association process.

## VII. CONCLUSION

The differential evolution-based solution to the grid-based SLAM problem presented introduces a new possibility to accurately solve the SLAM problem. At a local scale, the ELF localization algorithm provides a fast and accurate maximum likelihood estimate with results equivalent to other re-localization methods like scan matching approaches. And at a global scale, the algorithm only incurs substantial cost at cycle closing detection time and the cost increases linearly with the number of poses included in the detected loop, remaining constant for the re-localization and integration of the scans in the global map which is lower than half

second second for the re-location and the mapping of the scan data into the map. The low computational cost allows online operation even in relatively large areas.

The search for this solution is done stochastically, employing an evolutive search technique. The algorithm has been tested with data acquired with different robots equipped with laser range scanners. Tests performed with our algorithm have demonstrated the algorithm's robustness and accuracy generating maps.

The method can accommodate arbitrary non-linear system dynamics, sensor characteristics and non-Gaussian noise, avoiding unnecessary linearizations. It does not require any assumptions about the shape of the posterior density, unlike parametric approaches and due to the stochastic nature of the algorithm search of the best robot pose estimate, the algorithm is able to cope with a high level of sensor noise with low degradation of the estimation results.

The delayed mapping in case of loops eliminates the necessity of maintaining multiple maps or re-mapping each time a significant global error is detected.

## VIII. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the funds provided by the Spanish Government through the MCYT project DPI2004-0594 .

## REFERENCES

- [1] Castellanos, J.A., Neira, J., and Tardos, J.D., Limits to the consistency of EKF-based SLAM, in: *Proc. of the 5th IFAC/EURON Symposium on Intelligent and Autonomous Vehicles IAV-2004*, Lisboa, Portugal, 2004.
- [2] Guivant, J. and Nebot, E., Optimization of the simultaneous localization and map building algorithm for real time implementation, in: *EEE Trans. on Robotics and Automation*, 17 (3), pp 242-257, 2001.
- [3] Julier, S.J. and Uhlmann, J.K., A counter example to the theory of simultaneous localization and map building, in: *Proc. of the Int. Conference on Robotics and Automation ICRA-01*, Seoul, Korea, pp 4238-4243, 2001.
- [4] Leonard, J. and Feder, H., Decoupled stochastic mapping, in: *IEEE J. Ocean Engineering*, 26 (4), pp 561-571, 2002.
- [5] Montemerlo, M., and Thrun, S., Simultaneous localization and mapping with unknown data association using FastSLAM, in: *Proc. of the Int. Conference on Intelligent Robotics and Automation ICRA-03*, Taipei, Taiwan, vol 2, pp 1985-1991, 2003.
- [6] Moreno, L., Garrido, S., and Muñoz, M.L., Evolutionary filter for robust mobile robot global localization, in: *Robotics and Autonomous Systems*, 54, pp 590-600, 2006.
- [7] Batalin, M., The Robotics Data Set Repository (Radish), in: <http://radish.sourceforge.net>.
- [8] Smith, R., Self, M., and Cheeseman, P., Estimating uncertain spatial relationships in robotics, in: I. Cox and G. Wilfong (Eds.), *Autonomous Robots Vehicles*, pp 167-193, Springer-Verlag, 1990.
- [9] Storn, R. and Price, K., Differential Evolution- A simple and efficient adaptive heuristic for global optimization over continuous spaces, in: *Journal of Global Optimization*, 11, pp 341-359, 1997.
- [10] Tardos, J., Neira, J., Newman, P., and Leonard, J., Robust mapping and localization in indoor environments using sonar data, in: *Int. Jou. Robotics Research*, 21 (4), pp 311-330, 200.
- [11] Thrun, S., Simultaneous mpping and localization with sparse extended information filters: theory and initial results. in: CMU-CS-02-112, 2002.
- [12] Thrun, S., Burgard, W. and Fox, D., in: *Probabilistic Robotics*, The MIT Press, 2005.
- [13] Williams, S., Dissanayake, G. and Durrant-Whyte, H., An efficient approach to the simultaneous localization and mapping problem, in: *Proc. of IEEE Int. Conf on Robotics and Automation ICRA-02*, pp 406-411, 2002.