Vehicle localization system using offboard range sensor network

J. Ferreira * **A. Vale** * **R. Ventura** **

* *Instituto de Plasmas e Fusão Nuclear, Instituto Superior Técnico,*
Universidade Técnica de Lisboa, Av. Rovisco Pais 1 - 1049-001 Lisboa
- Portugal (`jftferreira@ipfn.ist.utl.pt`*;* `avale@ipfn.ist.utl.pt`*).*
** *Laboratrio de Robótica e Sistemas em Engenharia e Ciência,*
Instituto Superior Técnico, Universidade Técnica de Lisboa,
Av. Rovisco Pais 1 - 1049-001 Lisboa - Portugal
(`rodrigo.ventura@isr.ist.utl.pt`*)*

Abstract: Most mobile vehicle localization approaches rely on onboard sensors. However, whenever the installation of sensors onboard the vehicle is unfeasible, an alternative solution is to install them in the environment. One such environment is the ITER nuclear fusion reactor, where all maintenance operations have to be performed by remote handling, due to the radiation levels. This paper addresses the problem of vehicle localization in a structured environment, using a network of laser range finder sensors. The approach taken is based on: (1) the optimization of the sensor placement in the environment, aiming at the maximization of the area covered by the sensors and the redundancy of the sensor network, and (2) a probabilistic approach for vehicle localization. Two localization methods were evaluated: Extended Kalman Filter and Particle Filter. These two methods are compared, with respect to localization performance and robustness, both in simulation and using a real vehicle in a mock-up scenario.

Keywords: Autonomous vehicles, Localization, Range finders.

1. INTRODUCTION

One of the greatest challenges in the present time is the production of clean energy with small environmental impact, while satisfying an ever growing consumption demand. Nuclear fusion power, the energy of the stars, has potential to be a suitable alternative. The International Thermonuclear Experimental Reactor (ITER) is an experimental fusion reactor, currently under construction, that will be used to test fusion power process [1].

During the ITER lifetime, maintenance operations of inspection and component replacement are required. Given the rad-hard conditions, these operations are remotely handled. The transportation of the heavy equipment required is performed by a Cask and Plug Remote Handling System (CPRHS), with similar dimensions to an autobus, where the entire weight can reach 100 tons. The CPRHS is supported by a Cask Transfer System (CTS), using two wheels in a rhombic profile to move through narrow corridors in the Tokamak Building (TB) and Hot Cell Building (HCB). The challenge of moving heavy loads autonomously, with tight safety margins, requires a precise localization system.

Electronic sensing is a key issue in vehicle localization systems, being commonly carried out using onboard sensors. The CPRHS, which endorses the missions of maintenance in ITER scenarios, transports rad-hard equipment. Electronic sensors have a reduced life time and their measurements are affected when exposed to this level of radiation.

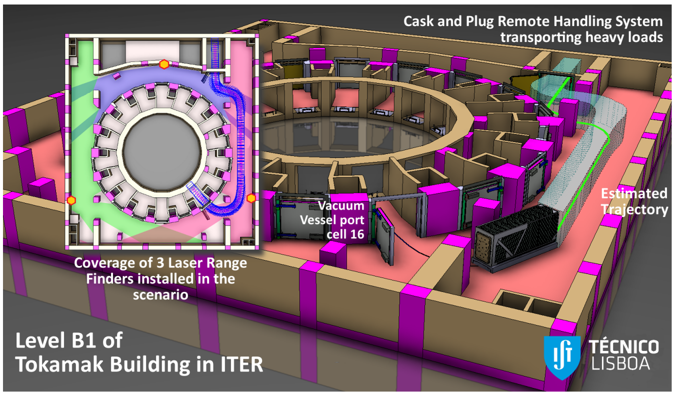


Fig. 1. A vehicle operating in the Tokamak Building and example of coverage using three laser range finders.

Therefore, the electronic sensors should not be installed onboard of the vehicle. Their installation in fixed locations in the building walls is examined instead. In addition, Laser Range Finder (LRF) sensors are considered, since the electronic part can be shielded, while the mirror is exposed, with no interferences or disturbances to the measurements. LRF sensors are precise and accurate from short to long ranges, as described in Surmann et al. [2003], Wahlde et al. [2009], and laser rays are not affected by magnetic fields.

A single LRF sensor is not enough to cover an entire floor of the TB (Fig. 1), given the complexity of the scenario. Hence, a sensor network of LRF units is required to provide a broader coverage. The first challenge in the design of this

---

[1] http://www.iter.org (retrieved 1-Nov-2012)

network is the identification of the best configuration, i.e., the amount of sensors and their respective placements. The second challenges is the use of this network to localize the vehicle.

This paper presents two main novelties over the previous publication, Ferreira et al. [2010]: *(i)* Introduction of a redundancy measure in the sensor placement optimization cost function, and *(ii)* Development of two localization approaches, using the optimized LRF sensor network.

Redundancy in the LRF sensor network is measured as the area covered by more than one sensor. In particular, this area, once being covered by more than one sensor, does not lose coverage after the failure of a single sensor.

The two localization approaches studied are based in Extended Kalman Filter (EKF), mentioned by Arras et al. [2001], Se et al. [2001] and bootstrap Particle Filter (PF) as in Dellaert et al. [1999], Rekleitis [2004], typically used in localization systems with onboard sensors. In this framework, the sensor network observes both the vehicle and the environment. However, there are some systems using offboard sensors, based on video-cameras, like mentioned by Losada et al. [2010] or Radio Frequency Identification (RFID) tags as the work by Bouet and Dos Santos [2009].

This paper is organized as follows: Section 1 introduced the remote handling problem and the proposed localization system. Section 2 describes the optimization of sensor placement with inclusion of redundancy. Section 3 addresses two localization approaches using offboard LRF sensors. Section 4 compares, in simulation, the performance of the two localization approaches with different sensor placements and evaluates the system with real experimental data. Section 5 concludes the paper and suggests future research directions.

2. SENSOR PLACEMENT OPTIMIZATION

The placement of the sensors in the sensor network is crucial for a good localization performance. Thus, an optimization approach is taken to determine the best positioning of the sensors. The best placement evaluation comprises both criteria (i) coverage and (ii) degree of redundancy. (i) minimizes blind spots with respect to the sensor network, while (ii) aims at the mitigation of the effects of arbitrary sensor failures. This section summarizes the previous work by Ferreira et al. [2010], and introduces the redundancy (5) as an optimization criteria, relevant to the sensor placement decision.

Let the configuration of a single sensor be s_i, (1), illustrated on Fig. 2 (top left), composed by x_s^i and y_s^i, the absolute position and θ_s^i, the sensor orientation. Let Φ_s^i be the FoV, δ_s^i the angular resolution, σ_s^i the standard deviation for distance measurement errors, and r_s^i the sensor range. The values of Φ_s^i, δ_s^i, σ_s^i and r_s^i are established depending on the equipment installed. Only x_s^i, y_s^i and θ_s^i are extracted from the optimization process. The LRF sensor network, S, is a set of sensors with different configurations (2), where L is the total number of sensors.

$$s_i = \left[\, x_s^i \;\, y_s^i \;\, \theta_s^i \;\, \Phi_s^i \;\, \delta_s^i \;\, \sigma_s^i \;\, r_s^i \,\right]^T \tag{1}$$

$$S = [s_1 \;\; s_2 \;\; \dots \;\; s_L] \tag{2}$$

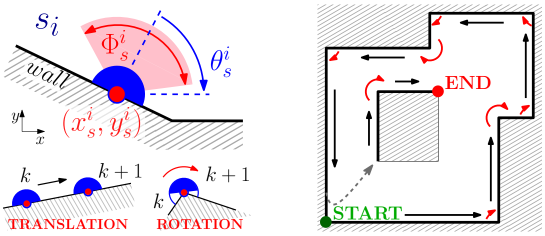


Fig. 2. Sensor configuration (top left), translation and rotation steps on SSS sequence (bottom left) and map with sensor state space sequence (right).

For optimization propose, a map is a 2D simplification of the scenario, and a Sensor State Space (SSS) is defined as the set of all feasible sensor placements. It is obtained as a travelling sequence through map walls where the sensor can be installed, as illustrated in Fig. 2 (right). The sequence, composed by translations and rotations of the sensor, as in Fig. 2 (bottom left), is discrete and parameterized by the index k.

The optimization process computes the optimal placement for L sensors respecting a defined fitness function $F(S)$ (3). Fitness function accounts for two criteria: *(i)* Coverage, $C(S)$ (4), is the most important criteria to guarantee that the network can observe the vehicle in normal operating conditions. *(ii)* Redundancy, $R(S)$ (5), it is the main novelty in sensor placement optimization, important to attain robustness to sensor failure and temporary occlusions of a sensor, it is less important than coverage, for localization system performance, but still critical in ITER scenarios.

$$F(S) = w_C\, C(S) + w_R\, R(S) \tag{3}$$

$$C(S) = Area\left[\bigcup_{j=1}^{L} P(s_j)\right], \; s_j \in S \tag{4}$$

$$R(S) = Area\left[\bigcup_{j=1}^{L-1} \bigcup_{k=j+1}^{L} P(s_j) \cap P(s_k)\right], \; s_{j,k} \in S \tag{5}$$

$P(s_i)$ is a polygon containing the area covered by the sensor with configuration s_i, as illustrated in Fig. 3. *Area* is a function that computes the area of a general polygon as a percentage of the total map area. The function *Area* is computed with a cloud of uniform distributed points over the map, *Area* is the percentage of those points that are inside the polygon. The weights w_C and w_R give the respective importance to coverage and redundancy and, in this case, respect $0 < w_R < w_C$.

The optimization method employs the simulated annealing (SA) algorithm, using F as fitness function, it returns a network configuration corresponding to a maximum of F. For each evaluation of SA (*run*), the result may vary due to local maxima, since F is non-convex. To mitigate this problem, the resulting network, S^* (6), is obtained after several *runs* of SA, yielding a set of networks, $O = [S_1 ... S_u]$. S^* is the one with the higher fitness value.

$$S^* = arg \max_{O} F(S), \quad S \in O \tag{6}$$

3. VEHICLE LOCALIZATION

The localization problem consists in the estimation, $\bar{x}_t$ of the correct pose, x_t (7), i.e., position and orientation of the vehicle, at time t, in a global reference frame.

$$x_t = \left[\, x_r^t \;\, y_r^t \;\, \theta_r^t \,\right]^T \tag{7}$$

Localization system, with the framework shown in Fig. 3, should estimate the vehicle pose integrating the odome-

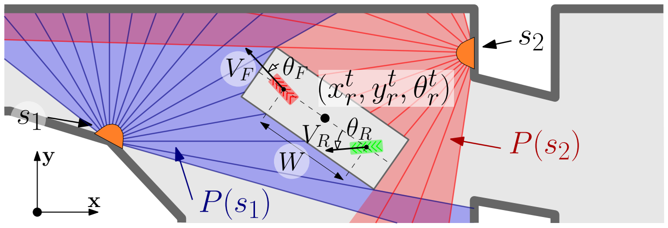


Fig. 3. Localization system framework, with example of LRF sensor network, vehicle pose and odometry.

try commands, $u_{t_{od}}=[\,\theta_F\ V_F\ \theta_R\ V_R\,]^T$ and the measurements coming from the previously optimized LRF sensor network. Integration of odometry is based on the vehicle kinematic model (8), described by Ribeiro et al. [1997], where W is the distance between wheels and t_{od} is the time when the command $u_{t_{od}}$ was applied.

$$\bar{x}_t=\bar{x}_{t_{od}}+\frac{t-t_{od}}{2}\begin{bmatrix} V_F cos(\theta_r^t+\theta_F)+V_R cos(\theta_r^t+\theta_R) \\ V_F sin(\theta_r^t+\theta_F)+V_R sin(\theta_r^t+\theta_R) \\ \frac{V_F sin(\theta_F)-V_R sin(\theta_R)}{W/2} \end{bmatrix} \tag{8}$$

Measurements acquired from the LRF network at time t_{ac}, $Z_{t_{ac}}$ (9), are distances from the corresponding LRF sensor to the nearest obstacle, in each direction. Measurements from each sensor $z(s_i)$, are arranged in an array (10) containing distances and orientations for each laser ray.

$$Z_{t_{ac}}=[z(s_1)\ \ z(s_2)\ \ \ldots\ \ z(s_L)] \tag{9}$$

$$z(s_i)=\begin{bmatrix} d_1^i & \cdots & d_{M_i}^i \\ \varphi_1^i & \cdots & \varphi_{M_i}^i \end{bmatrix}^T \tag{10}$$

φ_j^i is the angle, in the global coordinate frame, of the j-th laser ray emitted from the i-th sensor and d_j^i is the distance to the nearest obstacle measured in that direction. M_i is the number of measurements for each sensor. φ_j^i are considered constants, only d_j^i change according to the surrounding obstacles and measurement noise. The uncertainty in the measured distances can be modelled as zero mean Gaussian noise with σ_s^i standard deviation as suggested by Ye and Borenstein [2002]. Based on this framework, two Bayesian approaches, EKF and PF, are developed to localize the vehicle. The two Bayesian filters are implemented as described by Thrun et al. [2005]. This section is focused on the understanding of the observation models adapted to the current framework, as they are the main innovative contribution.

The observation model $h(\bar{x}_t)$, is a non-linear and non-smooth function that predicts sensor network measurements, $\bar{Z}_t$. It predicts, in direction φ_j^i, the distance from s_i to an obstacle ($\bar{d}_j^i$), given the vehicle pose $\bar{x}_t$. These distances do not correspond to real ones, due to measurement noise or incorrect estimation of vehicle pose.

Let $\chi_i(x_t)$ be the set of directions for which the measurements, from sensor s_i, hit the vehicle with pose x_t, and $\overline{\chi_i}(x_t)$ the measurements from s_i not hitting the vehicle. These sets are illustrated in Fig. 4, for real and predicted poses. The observation models for both Bayesian approaches distinguish these measurement sets.

3.1 Extended Kalman Filter

EKF uses a Jacobian matrix to relate the errors between the real and predicted measurements with the vehicle pose.

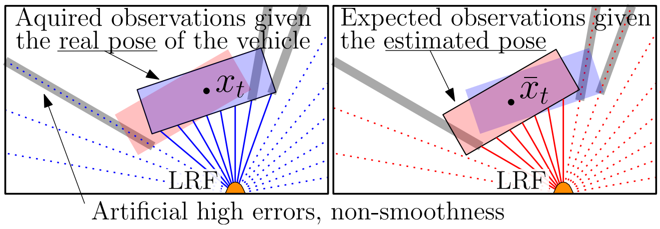


Fig. 4. $\chi_i(x_t)$ in solid and $\overline{\chi_i}(x_t)$ in dashed lines (left), $\chi_i(\bar{x}_t)$ in solid and $\overline{\chi_i}(\bar{x}_t)$ in dashed lines (right).

If $\varphi_j^i \in \overline{\chi_i}(\bar{x}_t)$, the distance $\bar{d}_j^i$ does not depend directly on vehicle pose, this measurement should not be integrated because the Jacobian entrance is zero. If $\varphi_j^i \in \chi_i(\bar{x}_t) \cap \overline{\chi_i}(x_t)$, the measurement hits the vehicle in predicted pose but not in reality, the corresponding measurement should not be included as well. This case represents a non-smoothness of measurement model function. EKF confuses a vehicle edge with a map wall and introduces an artificial high error, illustrated in Fig. 4 (shaded rays).

Only measurements hitting the vehicle in both predicted and real poses, $\varphi_j^i \in \chi_i(\bar{x}_t) \cap \chi_i(x_t)$, are integrated. EKF predicted pose must be always near the real pose, otherwise there will be no data integration and the update step of EKF is ineffective.

To avoid that EKF neglects all data, the number of measurements integrated on each iteration, is monitored, and, every time it drops below a certain threshold, EKF is restarted. The restarting iteration of EKF assumes that the vehicle position is the center of mass of measurements hitting the vehicle, *mmc*,

$$mmc=\frac{1}{K}\sum_i\sum_j p2c(d_j^i,\varphi_j^i)+\begin{bmatrix} x_s^i \\ y_s^i \end{bmatrix},\varphi_j^i\in\chi_i(x_t) \tag{11}$$

where $p2c(d_j^i, \varphi_j^i)$ is the transformation from polar to Cartesian coordinates, and K the number of measurements that hit the vehicle. The EKF uncertainty in restarting iteration is high, allowing the convergence to the correct pose of the vehicle. This heuristic allows EKF to globally localize the vehicle, thus effectively solving the kidnapped robot problem described by Thrun et al. [2005].

3.2 Particle Filter

PF uses a set of particles (X_t) to represent a probability distribution of the vehicle pose, each particle, with index n, is an hypothetical poses of the vehicle, $x_t^{[n]}$, with a distinct weight $w_t^{[n]}$, assigned by the observation model.

The observation model is a likelihood function, $p(Z|x_t^{[n]})$, that assumes conditionally independent but not identically distributed measurements, (12); it distinguish two different distributions if the predicted measurements hit the hypothetical vehicle or not.

$$w_t^{[n]}=p(Z|x_t^{[n]})\propto\prod_{i=1}^{L}\prod_{j=1}^{P_i} p(d_j^i|x_t^{[n]}). \tag{12}$$

$p(d_j^i|x_t^{[n]})$ is the likelihood function of a single measurement that follows the distribution $p(x_t^{[n]}|d_j^i)$ (13) depending on whether the measurement hits the vehicle in the hypothetical pose – case (i), or not – case (ii).

$$p(x_t^{[n]}|d_j^i)=m{*}p^T=[\,m_1\ \ldots\ m_6\,][\,p_1\ \ldots\ p_6\,]^T \tag{13}$$

(i) If $\varphi_j^i \in \chi_i(x_t^{[n]})$: $p_1 = \mathcal{N}(\bar{d}_j^i, \sigma^2)$, modelling measurements that hit the vehicle in reality; $p_2 = \mathcal{N}(D_j^i, \sigma^2)$, modelling measurements that hit the walls in reality; $p_3 = \mathcal{U}(0, r_s^i)$, modelling outliers; $p_4 = p_5 = p_6 = 0$; $m_1 > m_2$, $m_1 > m_3$, to reinforce particles with correct prediction.

(ii) If $\varphi_j^i \in \overline{\chi_i}(x_t^{[n]})$: $p_4 = \mathcal{N}(D_j^i, \sigma^2)$, modelling measurements that hit the walls in reality; $p_5 = \mathcal{U}(0, D_j^i)$, modelling measurements that hit the vehicle in reality; $p_6 = \mathcal{U}(0, r_s^i)$, modelling outliers; $p_1 = p_2 = p_3 = 0$; $m_4 > m_5$, $m_4 > m_6$, to ensure that correct predictions have greater likelihood.

D_j^i is the distance, in direction φ_j^i, to the nearest wall. $\mathcal{U}(a, b)$ is an uniform distribution with limits a and b and $\mathcal{N}(\mu, \sigma)$ is a normal distribution with mean μ and the standard deviation σ, assumed for the measurements and always greater than σ_s^i. The condition $\sigma > \sigma_s^i$ allows a smoother variation of particle weights.

PF have the possibility for global localization, using the same principle explained for EKF. When the measurement likelihood drops, the probability of generating particles around *mmc* rises and $\bar{x}_t$ converges to real vehicle pose.

4. RESULTS

This section presents the obtained results from (i) the sensor placement method, comparing it with previous work of Ferreira et al. [2010], and (ii) the localization performance, both in simulation and experimentally using the two approaches, based on EKF and PF.

4.1 Sensor Placement Optimization

The algorithm to optimize sensors distribution was tested in a level of the TB, represented in Fig. 6 (left), with 150 corners and a total wall length of 680 m. The sensor state space includes all walls and was obtained with a translation step of 0.2 m and 2.25^o step for rotation. The discretization step was chosen considering a trade-off between precision and computational effort.

Fig. 5 shows the optimization results for 25 *runs* of the SA, for networks of up to 10 sensors, with and without accounting for redundancy. Red (lighter) and blue (darker) lines represent, for the given number of sensors, the coverage and redundancy of optimal network, S^*, respectively, while the red (lighter) and blue (darker) areas represent the intervals of coverage and redundancy achieved during all *runs*. The thickness of the blue (darker) area in Fig. 5 (left) shows that, for different *runs*, redundancy have large variations while coverage is steady, suggesting that, including redundancy in the fitness function would not affect the coverage. In Fig. 5 (right) redundancy is considered and the resulting optimal networks converges to values quite close to the maximum redundancy achieved for all *runs*. The computational effort of adding redundancy to the optimization method was found not to be relevant.

4.2 Simulated Vehicle Localization

Both localization approaches were tested in several trajectories in the TB using the Trajectory Evaluator and Simulator (TES), a software tool developed in the MATLAB platform under the grants F4E-2008-GRT-016 and F4E-GRT-276-01. A PC with Intel Core 2 Duo at 1.66GHz with 1Gb RAM was used to obtain the numerical results.

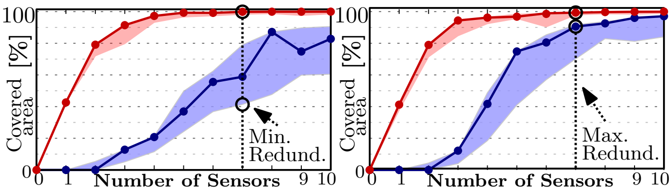


Fig. 5. Optimization for multiple *runs* in TB, considering only coverage $(w_C, w_R) = (1, 0)$ (left) and considering also redundancy $(w_C, w_R) = (0.9, 0.1)$ (right).

For the following results, the optimized LRF sensor network with 4 sensors was employed, as depicted in Fig. 6 (left). Four sensors is enough to achieve $C(S) = 97\%$ while keeping a low computer effort. The standard deviation for simulated noise in distance measurements is $\sigma_s^i = 0.1m$ and the one considered in the measurement model is $\sigma = 0.5m$, (empirical value explained in section 3). The PF uses 100 particles.

The weights used in the distribution $p(x_t^{[n]}|d_j^i)$ are experimentally tuned. For the following experiments the weights are $m = [0.9\ 0.09\ 0.01\ 0.6\ 0.39\ 0.01]$. The vehicle follows a trajectory depicted in green (lighter) in Figs. 6 and 8 (left). Pose estimation along trajectory is shown in blue (darker). The resulting position error, $l_{err}(t)$, and orientation error, $\theta_{err}(t)$ (14), are plotted on the right of Figs. 6 and 8.

The initial larger error values are due to global localization. In this experiment the localization belief was initialized with a random pose, taking a few iterations to converge to the vicinity of the correct pose of the vehicle. The average number of iterations needed to converge from random pose to a steady, accurate pose, in a kidnapped vehicle situation are 48 for EKF and 30 for PF.

$$\begin{gathered} l_{err}(t)=\sqrt{(\bar{x}_r^t-x_r^t)^2+(\bar{y}_r^t-x_r^t)^2} \\ \theta_{err}(t)=\bar{\theta}_r^t-\theta_r^t \qquad e=\tfrac{1}{T}\sum\nolimits_t^T l_{err}(t) \end{gathered} \tag{14}$$

Zoomed areas with a red/lighter rectangle show a segment of the trajectory where the EKF loses stability while the PF keeps an accurate estimation. This is a consequence of the non-smoothness of the observation model, depicted in gray on Fig. 4. This effect is minimized by the distributions p_2 and p_5, included in PF observation model (section 3).

Zoomed areas with a blue/darker rectangle emphasize a situation where the LRF measurements hit only one side of the vehicle and the EKF has a greater uncertainty than PF along the vehicle longitudinal direction, $\overrightarrow{L}$. EKF only integrates measurements hitting the vehicle while PF also integrates measurements hitting the walls, as illustrated in the zoomed area by the red rays. PF has information that the vehicle is not shifted to front or rear which decreases uncertainty in that direction.

The average error, e, (14), along the trajectory is used to evaluate the localization performance for the two approaches using different LRF sensor network configurations. Fig. 7 shows e depending on the number of sensor installed. The bars show the standard deviation of e.

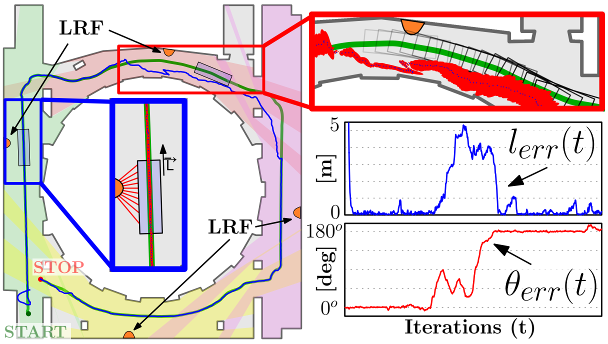


Fig. 6. EKF approach – Real (green/lighter) and estimated (blue/darker) trajectories (left), Zoomed areas with estimated poses (blue dots) and 99% Gaussian beliefs (red); Error along the trajectory (right)

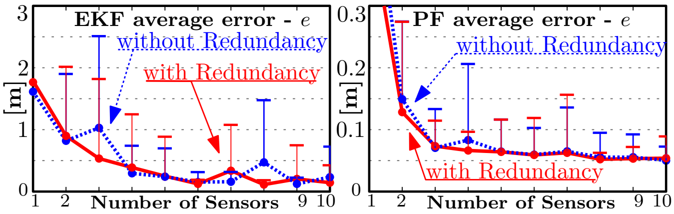


Fig. 7. Average error e depending on the LRF sensor network, EKF approach (left) and PF approach (right).

The used LRF networks were the ones optimized in the previous section, with coverage and redundancy depicted in Fig. 5. Blue doted lines correspond to e achieved with networks optimized with coverage only while red solid lines is for networks with optimized redundancy. For networks with a single sensor, the vehicle is occluded for great part of the trajectory, resulting in high average error but, while vehicle is observed, the e is 0.7m for EKF and 0.07m for PF. PF is generally more accurate and stable but, as the number of sensors (L) installed grows, EKF accuracy improves significantly. Furthermore, the computational time is a noticeable advantage for EKF, when L is high, as shown in Fig. 9 (left).

Results in Fig. 7 show that the localization performance of both approaches does not change significantly with LRF networks obtained with or without redundancy optimization. Redundancy becomes important when there are sensor failures. The failure of one sensor is simulated for two LRF networks with 7 sensors: *(Min. Redund.)* – $(C(S), R(S)) = (99.9\%, 41.4\%)$ has the lowest redundancy achieved in optimization process, and *(Max. Redund.)* – $(C(S), R(S)) = (99.3\%, 90.6\%)$ is the network with optimized redundancy. Values of networks used are marked in Fig. 5. The error e is shown in Fig. 9 (right) for both approaches depending on which sensor is failing. The error does not vary much depending on the sensor that fails, but it is visible that the network with higher redundancy achieves better accuracy when faced with a failure.

4.3 Experimental Vehicle Localization

The experimental setup, shown in Fig. 10, is a 1:25 scaled mock-up of half of a floor of the TB. It includes four Hokuyo LRF sensors with $\Phi_s^i = 240^o$ and $\delta_s^i = 0.36^o$ and a vehicle prototype with $330mm$x$100mm$ that follows a trajectory, in Fig. 10 (right) previously optimized resorting to the work of Fonte et al. [2011].

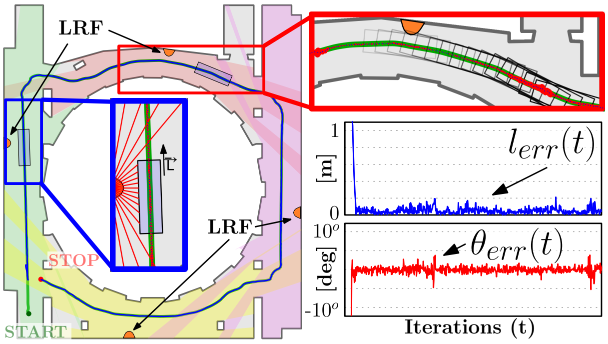


Fig. 8. PF approach – Real (green/lighter) and estimated (blue/darker) trajectories (left), Zoomed areas with estimated poses (blue dots) and 99% Gaussian beliefs (red); Error along the trajectory (right)

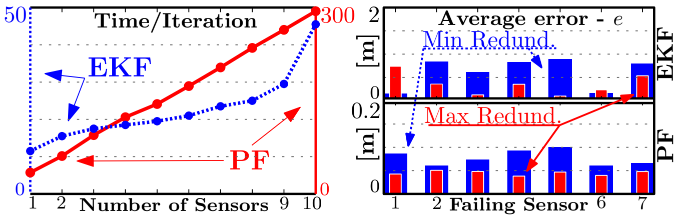


Fig. 9. Average computational time per iteration (left), Average error e depending on which sensor fails (right), for EKF and PF approaches.

Only the PF approach is used in this context, since it proved to be more robust in simulation. Lasers and odometry readings are acquired approximately at $4Hz$. PF uses 100 particles and $\sigma = 7mm$.

Due to de-synchronization between odometry and laser readings the localization algorithm was adapted as shown in Fig. 11 (right). It has the possibility to buffer several odometry readings in a queue Q, and only integrates them when a laser reading arrives. In the mean time, the pose estimation is based only in odometry. The instruction $Q_{t_1}^{t_2}$ retrieves the odometry commands from the queue between times t_1 and t_2 ($t_1 < t_2$). $X_{t_2}{=}\mathcal{F}(X_{t_1}, Q_{t_1}^{t_2})$ applies the commands retrieved by $Q_{t_1}^{t_2}$ to all particles in X_{t_1} following the model in (8) which generates the predicted distribution X_{t_2} . $\bar{x}_{t_2}{=}\mathcal{F}(\bar{x}_{t_1}, Q_{t_1}^{t_2})$ is similar but for one pose only. $\mathcal{H}(X_t, Z_t)$ assigns weights to all particles in X_t as in the observation model (12). **resample()** implements a typical resample step similar as explained by Thrun et al. [2005] and **estimation()** retrieves one pose estimation from the particle distribution using a weighted average of all particle poses. t_{queue} is the t_{od} of the second older command in the queue Q. $t(k)$ and $T(k)$ are the times of update, at iteration k, of the pose estimation $\bar{x}_{t(k)}$ and distribution $X_{T(k)}$, respectively.

Fig. 11 (left) shows the results compared with the original trajectory. The vehicle is guided through the trajectory with position feedback from a tracking system that is outside our scope. The localization error l_{err} is below $4cm$ and θ_{err} is typically below 10^o. Since the pose from tracking algorithm is the best estimation of the real pose, the localization estimations are compared with those poses and the error presented includes also errors related to the tracking system as well. The performance is also affected by two other factors: (1) the vehicle travels very close

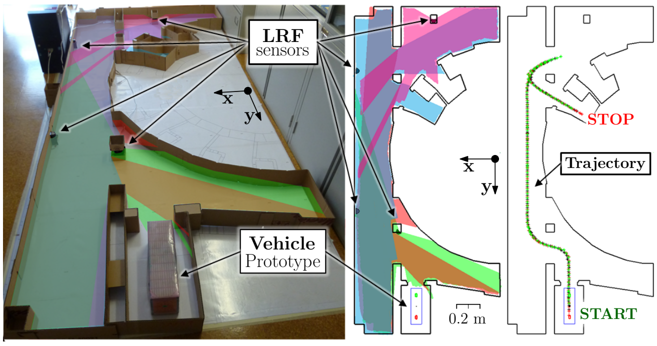


Fig. 10. Experimental setup – Mock-up (left) and 2D Map (middle) with areas covered by sensors; 2D Map with trajectory followed by the vehicle prototype (right).

to the sensors and, for Hokuyo sensors, the precision for ranges below 20cm is severely affected; (2) the mock-up is not a perfect match with his virtual model, used in measurement prediction. Even though the algorithm shows high robustness and it is still capable of localizing the vehicle with acceptable precision. For a real size implementation the building construction is preciser and the distance between sensors and vehicle greater.

5. CONCLUSIONS AND FUTURE DEVELOPMENTS

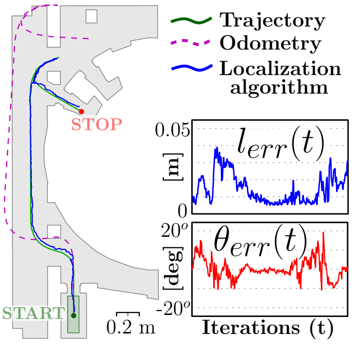
This paper describes a vehicle localization system using offboard range sensor network. The LRF sensor network is installed in the map and its design is based on an optimization method that yields the sensors placement maximizing both network coverage and redundancy.

Based on the data acquired by the LRF sensors in the optimized network, an algorithm estimates the vehicle position and orientation. Two approaches are evaluated and compared: EKF and PF. The PF has shown better results for the vehicle localization since it is more accurate, stable, and robust. However EKF shows better improvement when the number of sensors grow, moreover it requires less computer effort. Based on these results, the PF approach was tested in a experimental environment and the results confirm the feasibility of the system and the correct functionality inside safety margins. The experiment is completely scalable for a larger vehicle as the number of measurements hitting the vehicle is the same and the building dimensions are inside the typical LRF sensor ranges, moreover the the building construction is preciser and there are not close range measurements that affect sensor precision.

Further work will focus on the following issues: *(i)* reduce PF computational effort, for instance, using only measurements in the vicinity of the vehicle; *(ii)* devise a calibration system to correct LRF sensors installation miss placement; *(iii)* multiple vehicle support (e.g. vehicle with different dimensions, operating at same time). It is necessary to assess the occlusions created by the vehicle between them when operating simultaneously.

ACKNOWLEDGEMENTS

The work was supported by FCT in the frame of the Contract of Associate Laboratories of Instituto de Plasmas e Fusão Nuclear/IST (PEst-OE/SADG/LA0010/2011) and Laboratório de Robótica e Sistemas em Engenharia e Ciências/IST (PEst-OE/EEI/LA0009/2011).



```
$\bar{x}_{t(k)}$=**Localization**($t$, $Z_{t_{ac}}$, $u_{t_{od}}$)
    $t(k) = t(k-1) \quad T(k) = T(k-1)$
    **if** $u_{t_{od}}$ not empty, **push** $u_{t_{od}}$ in $Q$
    **if** $Z_{t_{ac}}$ not empty,
        $t(k) = T(k) = t_{ac}$
        $X_{T(k)} = \mathcal{F}(X_{T(k-1)}, Q_{T(k-1)}^{T(k)})$
        $X_{T(k)} = \mathcal{H}(X_{T(k)}, Z_{t_{ac}})$
        $X_{T(k)} = \mathbf{resample}(X_{T(k)})$
        $\bar{x}_{t(k)} = \mathbf{estimation}(X_{T(k)})$
    **elseif** $Q$ is full,
        $t(k) = T(k) = t_{queue}$
        $X_{T(k)} = \mathcal{F}(X_{T(k-1)}, Q_{T(k-1)}^{T(k)})$
        $\bar{x}_{t(k)} = \mathbf{estimation}(X_{T(k)})$
    **if** $t > t(k)$,
        $\bar{x}_{t(k)} = \mathcal{F}(\bar{x}_{t(k)}, Q_{t(k)}^{t})$
        $t(k) = t$
```

Fig. 11. Localization results, Real trajectory (green/ lighter), Trajectory estimated only with odometry (purple/ dashed) and Trajectory estimated using PF approach (blue/ darker) (left); Localization algorithm (right)

REFERENCES

K. Arras, N. Tomatis, B.T. Jensen, and R. Siegwart. Multisensor on-the-fly localization: Precision and reliability for applications. *Rob. and Autonomous Sys.*, pages 131–143, 2001.

M. Bouet and A.L. Dos Santos. RFID tags: Positioning principles and localization techniques. In *Wireless Days*, pages 1–5, 2009.

F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte Carlo localization for mobile robots. In *IEEE Intern. Conf. on Rob. and Autom.*, pages 1322–1328, 1999.

J. Ferreira, A. Vale, and R. Ventura. Optimizing range finder sensor network coverage in indoor environment. In *IFAC Intelligent Autonomous Vehicles*, 2010.

D. Fonte, F. Valente, A. Vale, and I. Ribeiro. Path Optimization of Rhombic-Like Vehicles: An Approach Based on Rigid Body Dynamic. pages 106–111, 2011.

C. Losada, M. Mazo, S. Palazuelos, D. Pizarro, and M. Marrón. Multi-Camera Sensor System for 3D Segmentation and Localization of Multiple Mobile Robots. *Sensors*, pages 3261–3279, 2010.

I.M. Rekleitis. A particle filter tutorial for mobile robot localization. In *Raport, Univerity Montreal*, 2004.

I. Ribeiro, P. Lima, P. Aparcio, and Ferreira R. Conceptual Study on Flexible Guidance and Navigation for ITER Remote Handling Transport Casks. pages 969–972, 1997.

S. Se, D. Lowe, and J. Little. Local and global localization for mobile robots using visual landmarks. In *Proc. of the IEEE/RSJ Intern. Conf. on Intelligent Rob. and Sys.*, pages 414–420, 2001.

H. Surmann, A. Nuchter, and J. Hertzberg. An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. *Rob. and Autonomous Sys.*, pages 181–198, 2003.

S. Thrun, Burgard W., and D. Fox. *Probabilistic Robotics.* The MIT Press, 2005.

R. Wahlde, N. Wiedenman, W.A. Brown, and C. Viqueira. An Open-Path Obstacle Avoidance Algorithm Using Scanning Laser Range Data, 2009.

C. Ye and J. Borenstein. Characterization of a 2D laser scanner for mobile robot obstacle negotiation. In *IEEE Intern. Conf. on Rob. and Autom.*, 2002.