The kidnapping Problem of Mobile Robots: A Set Membership Approach.

Rémy GUYONNEAU, Sébastien LAGRANGE,
Laurent HARDOUIN and Philippe
LUCIDARME.
Laboratoire d'Ingénierie des Systèmes
Automatisés (LISA),
Université d'Angers,
Angers, France.
{remy.guyonneau, sebastien.lagrange,
laurent.hardouin,
philippe.lucidarme}@univ-angers.fr

***Abstract*— This paper presents a set membership method to deal with the kidnapping problem in mobile robotics. By using a range sensor, the odometry and a discrete map of an indoor environment, a robot has to detect a *kidnapping situation*, while performing a pose tracking, and then perform a global localization to estimate its new pose (position and orientation). In a bounded error context the IAL (Interval Analysis Localization) algorithm searches a small box (interval vector) that includes the robot's pose, using interval analysis and constraint propagation tools. This algorithm allows to perform a pose tracking and a global localization. Using this algorithm, it is possible to deal with the kidnapping problem.**

This method is tested using real data, recorded during the CAROTTE challenge, organized by the French ANR (National Research Agency) and the French Army.

As it is shown in this paper with the IAL algorithm, interval analysis is an efficient tool to solve the kidnapping problem.

I. INTRODUCTION

Robot localization is one of the most important issue of mobile robotics [1], [2]: a robot has to know its location to be able to perform navigation tasks. The localization problem can be divided into two categories: the pose tracking and the global localization. In the pose tracking problem a robot has to find its new pose using the knowledge of its previous pose. In the global localization problem a robot does not have the knowledge of its previous pose. It has to find its new pose globally in the environment. The kidnapping problem is a combination of those two problems: while the robot is performing a pose tracking, it is kidnapped (moved to an other pose, has too many outliers in the measurement set, does too many drifting...). The robot has to detect this kidnapping situation and then perform a global localisation before processing a pose tracking again.

Most of the proposed solutions to localize a robot are based on probabilistic estimation techniques (see [3], [4]). The Kalman Filter [5], [6] and its improvements [7] are used for pose tracking in [8] and more precisely for the SLAM problem (see [9], [10]). Particle filters [11] with for example the Monte Carlo algorithm [12] and its spin-off [13], [14] are used to deal with global localization and kidnapping problems.

In this paper a set membership approach will be considered for the kidnapping problem using the **I**nterval **A**nalysis **L**ocalization (IAL) algorithm.

The paper is organized as follows. First the considered kidnapping problem is presented in Section II. The IAL algorithm and the kidnapping resolution are detailed in Section III and tested in Section IV. Section V presents a way to integrate the IAL module in an mobile robot architecture and Finally VI concludes this paper.

II. THE KIDNAPPING PROBLEM

This section presents the considered problem.

A. The Robot

A mobile wheeled robot (depicted in Figure 1) with a range sensor is considered. This system is characterized by the following discrete time dynamic equations:

$$\begin{cases} \mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)) & (1) \\ \mathbf{y}(k) = g_{\mathbb{E}}(\mathbf{x}(k)) & (2) \end{cases}$$

The robot's pose $\mathbf{x}(k) = (x_1(k), x_2(k), \theta(k))$ is defined by its location $(x_1(k), x_2(k))$ and its orientation $\theta(k)$ in the environment denoted $\mathbb{E}$ at discrete time k. $\mathbb{E}$ in $\mathbb{R}^2$ is a two dimentional domain where the robot moves. The function f characterizes the robot's dynamic and the vector $\mathbf{u}(k)$ is the control vector at time k. The vector $\mathbf{y}(k) = (y_1(k), ..., y_n(k))$ is the vector of measurements. Note that $\mathbf{y}(k)$ depends on the robot pose $\mathbf{x}(k)$ and the environment $\mathbb{E}$. In fact, y_i corresponds to the distance into the direction γ_i between the robot and the first obstacle in $\mathbb{E}$ (Figure 2).

B. The Environment

The environment $\mathbb{E}$ where the robot is moving is approximated by an occupancy grid map [3]. Figure 3 represents an example of indoor environment. The grid map, named $\mathbb{G}$, is composed of $n \times m$ cells (i, j) and at each cell (i, j) is associated $g_{i,j} \in \{0, 1\}$:

Fig. 1: Robot used during the CAROTTE challenge.

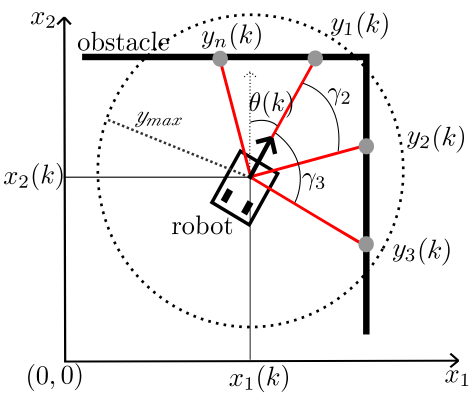


Fig. 2: Sensor measurements $\mathbf{y}(k) = (y_1(k), ..., y_n(k))$. y_{max} denotes the maximal range of the sensor.

$$g_{i,j} = \begin{cases} 0 & \text{if the cell corresponds to a free subspace} \\ & \text{of } \mathbb{E}, \\ 1 & \text{else.} \end{cases}$$

$\mathbb{G}$ is a discrete version of $\mathbb{E}$. Figure 4 represents an example of occupancy grid map with 35×38 cells.

C. The Objective

To solve the kidnapping problem, the objective is to have an algorithm that:

- is able to perform a pose tracking,
- is able to detect a kidnapping situation,
- is able to perform a global localization.

III. Interval Analysis Localization, a Deterministic Approach

The proposed method uses interval analysis and Constraint Satisfaction Problem tools to solve the kidnapping problem. This section presents basics of those tools and then presents the IAL algorithm and the kidnapping resolution.

(a) (b)

Fig. 3: The environment of the CAROTTE challenge.

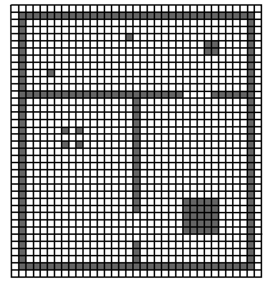


Fig. 4: Example of an occupancy grid map. Each dark cell value is 1 (the white cells represent free spaces).

A. Introduction

An *interval vector* [15], or a *box* $[\mathbf{x}]$ is defined as a closed subset of $\mathbb{R}^n$:$[\mathbf{x}] = ([x_1], [x_2], \cdots) = ([\underline{x_1}, \overline{x_1}], [\underline{x_2}, \overline{x_2}], \cdots)$.

The size of a box is defined as

$$size([\mathbf{x}]) = (\overline{x_1} - \underline{x_1}) \times (\overline{x_2} - \underline{x_2}) \times \cdots,$$

For instance $size([2,5],[1,8],[0,5]) = 105$.

It can be noticed that any arithmetic operators such as $+, -, \times, \div$ and functions such as $exp, sin, sqr, sqrt, ...$ can be easily extended to intervals, [16]. This interval arithmetic can be used to solve constrain satisfaction problems [17]. In the later, the localization problem will be seen as a constraint satisfaction problem.

A CSP[1] is defined by three sets. A set of variables $\mathcal{V}$, a set of domains (intervals) $\mathcal{D}$ for those variables and a set of constraints $\mathcal{C}$ connecting the variables together.

Example of CSP: Let $\mathbf{p}_i = (x_{1_i}, x_{2_i}), i = 1, ..., 4$ be four points and $[\mathbf{p}_i] = ([x_{1_i}], [x_{2_i}])$ four domains for those variables. The objective is to find $[a]$ such as $\forall a \in [a], x_{2_i} = a.x_{1_i}, i = 1, ..., 4$. Here is the considered CSP:

$$\left\{ \begin{array}{c} \mathcal{V} = \{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4, a\} \\ \mathcal{D} = \{[\mathbf{p}_1], [\mathbf{p}_2], [\mathbf{p}_3], [\mathbf{p}_4], [a]\} \\ \mathcal{C} = \{x_{2_i} = a \times x_{1_i}, i = 1, ..., 4\} \end{array} \right\}. \quad (3)$$

This kind of problem can be solved by using *constraint propagation* which consists in reducing the variable domains by using *contractors* C_{c_i} associated to each constraints c_i.

Figure 5 represents one example of that CSP. The domain $[a]$ is initialized with $[a] = [-\infty, +\infty]$. On the Figure 5a it can be noticed that there is no solution for this problem with the considered initial domains $[\mathbf{p}_i]$: it does not exist a such as the line $x_2 = a.x_1$ intersects all the boxes $[\mathbf{p}_i]$. The value q represents the number of boxes intersected by the grey line sets.

To solve this CSP it can be considered that one variable is an outlier (a variable that does not satisfied at least one constraint). In this context the solution is $[a]$ such as $\forall a \in [a]$ the line $x_2 = a.x_1$ intersects at least three boxes. This is done by using the relaxed intersection [18] and leads to the dark grey area of the Figure 5b. In the IAL algorithm the relaxed intersection is used to deal with outliers (measurements y_i

[1]Constraint Satisfaction Problem

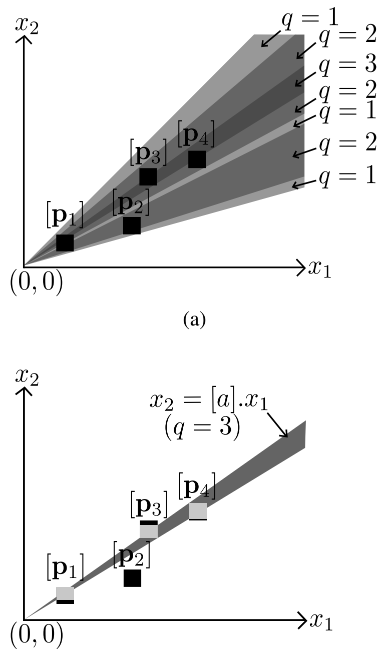


Fig. 5: Example of CSP. The equation $x_2 = [a].x_1$ represents all the lines that intersect at least three points. Note that the light grey boxes in Figure 5b correspond to the contracted points.

that do not correspond to the first obstacle of $\mathbb{E}$ in the direction γ_i).

Note that the CSP resolution can also lead to a reduction of the other domains. When $[a]$ is known it may be possible to reduce the other domains to be consistent with the constraints. In the Figure 5b the light grey boxes correspond to the contracted domains $[\mathbf{p}_1], [\mathbf{p}_3]$ and $[\mathbf{p}_4]$.

B. The IAL algorithm

A bounded error context is considered for the kidnapping problem of section II. Thus an interval $[y_i(k)]$ can be associated to each measurement $y_i(k)$, according to the sensor accuracy, and an interval $[\mathbf{u}(k)]$ can be associated to $\mathbf{u}(k)$.

In this context, the IAL algorithm is able to find a list $\mathcal{L}_k = \{[\mathbf{x}(k)]\}_i$, with $[\mathbf{x}(k)] = ([x_1(k)], [x_2(k)], [\theta(k)])$, that includes the robot pose $\mathbf{x}(k)$. Note that interval analysis leads to a deterministic as opposed to the probabilistic methods (that manipulate probability densities).

The algorithm IAL (see Algorithm 1) has three important steps.

First, line 4, the prediction step. The pose at time k is evaluated from the pose at time $k-1$ by using the equation 1. The calculation $[\mathbf{x}(k)] = f([\mathbf{x}(k-1)], [\mathbf{u}(k-1)])$ is done by using interval arithmetic.

Then, line 5 corresponds to the contraction step. The measurements $\mathbf{y}(k)$ and the observation function $g_{\mathbb{E}}$ are used to contract a pose: the idea is to find $\{\mathbf{x}|g_{\mathbb{E}}(\mathbf{x}) = \mathbf{y}(k)\}$.

As $\mathbb{E}$ is approximated by $\mathbb{G}$ IAL algorithm searches $\{[\mathbf{x}]|g_{\mathbb{G}}([\mathbf{x}]) = [\mathbf{y}(k)]\}$. This problem is seen as a CSP

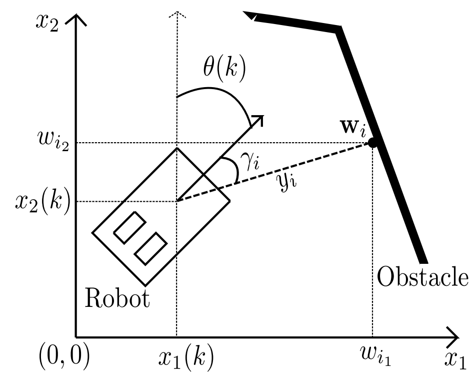


Fig. 6: $\mathbf{w}_i$, the coordinates of the obstacle detected by y_i in the map's frame.

with the variables $\mathbf{x}(k), \mathbf{y}(k)$, the domains $[\mathbf{x}(k)], [\mathbf{y}(k)]$ and constraints built with the measurements $\mathbf{y}(k)$ and the map $\mathbb{G}$. To be compared with the map, the measurements $[y_i]$ are converted to obstacle coordinates $[\mathbf{w}_i] = ([w_{i_1}], [w_{i_2}])$ in $\mathbb{G}$'s frame, according to $[\mathbf{x}(k)]$ and γ_i, as depicted in Figure 6. An example of contraction using one measurement is presented Figure 7.

Finally the last main step of the IAL is the bisection line 7. If a contracted box $[\mathbf{x}(k)]$ is bigger than the minimal size ξ we bisect $[\mathbf{x}(k)]$ into two boxes $[\mathbf{x}'(k)]$ and $[\mathbf{x}''(k)]$ such as $size([\mathbf{x}'(k)]) = size([\mathbf{x}''(k)])$ and $[\mathbf{x}'(k)] \cup [\mathbf{x}''(k)] = [\mathbf{x}(k)]$.

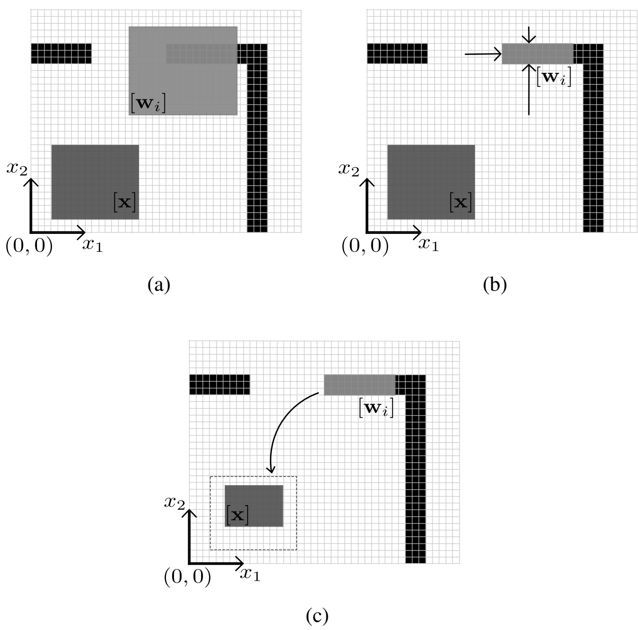
As it can be seen in line 11 a contracted box can be empty. This happens when the box is not consistent with the data set and means that it is impossible for the robot's pose to be included in that box. As said in sub-Section III-A outliers are considered by using relaxed intersection.

Algorithm 1 Interval Analysis Localization (IAL)

Require: $\mathcal{L}_{k-1}, [\mathbf{y}(k)], [\mathbf{u}(k-1)]$
1: $\mathcal{L}_k = \emptyset$
2: **while** $\mathcal{L}_{k-1} \neq \emptyset$ **do**
3: $[\mathbf{x}(k-1)] = \mathcal{L}_{k-1}.pop\_back()$
4: update $[\mathbf{x}(k-1)]$ to $[\mathbf{x}(k)]$ according to $[\mathbf{u}(k-1)]$
5: contract $[\mathbf{x}(k)]$ by using $[\mathbf{y}(k)]$ and $\mathbb{G}$
6: **if** $size([\mathbf{x}(k)]) > \xi$ **then**
7: bisect $[\mathbf{x}(k)]$ into $[\mathbf{x}_1(k)]$ and $[\mathbf{x}_2(k)]$
8: $\mathcal{L}_k.push\_back([\mathbf{x}_1(k)])$
9: $\mathcal{L}_k.push\_back([\mathbf{x}_2](k))$
10: **else**
11: **if** $[\mathbf{x}(k)] \neq \emptyset$ **then**
12: $\mathcal{L}_k.push\_back([\mathbf{x}(k)])$
13: **end if**
14: **end if**
15: **end while**
16: **return** $\mathcal{L}_k$

C. The kidnapping Resolution algorithm

Algorithm 2 represents the set membership resolution of the kidnapping problem using the IAL algorithm. It requires the initial pose of the robot $[\mathbf{x}(0)]$, such as $\mathbf{x}(0) \in [\mathbf{x}(0)]$. Lines 2 to 9, at each time k the robot's pose and the measurement set are updated, and the new robot's pose is



(a) (b) (c)

Fig. 7: Example of contraction. In Figure 7a there is a grid map $\mathbb{G}$ (each black cell value is 1), a pose $\mathbf{x}$ and a measurement y_i. The measurement is defined by $\mathbf{w}_i = (w_{i_1}, w_{i_2})$, the coordinates (in $\mathbb{G}$'s frame) of the detected obstacle. The box $[\mathbf{w}_i]$ (light grey) is a guaranteed evaluation of $\mathbf{w}_i$ according to $[\mathbf{x}]$ (dark grey) and $[y_i]$. By using the constraint *a measurement has to intersect with an obstacle in the map* the domain $[\mathbf{w}_i]$ is contracted (Figure 7b) and then by using an other constrain (*the distance between the robot and the detected obstacle is* $y_i \in [y_i]$) it is possible to contract the domain $[\mathbf{x}]$ (Figure 7b).

evaluated using the IAL algorithm. If the evaluated pose corresponds to an empty set, Line 6, it means that the robot has been kidnapped. The current pose is so initialised by all the possible poses, Line 8 ($x_{1_{max}}$ and $x_{2_{max}}$ corresponding to the limits of the environment), in order to perform a global localisation.

IV. Experimental Results

The following data (map and sensor measurements) were recorded during the CAROTTE challenge in June 2011, by the team CARTOMATIC.

The CAROTTE 20×20 meter environment is considered, Figure 3. By using the data sets of two different robots A and B (exploring this environment simultaneously) a grip map is built, Figure 8. This grid map has 2×2 centimetre grid cells. Because of the SLAM method used to fuse the data sets and build the map, some obstacles seen by the robot A and not by the robot B do not appear in the map, and vice-versa.

In this context, the objective is to track the pose of the robot A in the map. The Kidnapping Resolution algorithm is initialised by $[\mathbf{x}(0)] = ([0, x_{1_{max}}], [0, x_{2_{max}}], [0, 2\pi])$. During the pose tracking step, it appends that the IAL algorithm leads to an empty set $\mathcal{L}_k = \emptyset$, since too many

Algorithm 2 Kidnapping Resolution

Require: $[\mathbf{x}(0)]$
1: $\mathcal{L}_0 = \{[\mathbf{x}(0)]\}$
2: **for** $k = 1$ **to end do**
3: $\mathbf{x}(k) = f(\mathbf{x}(k-1), \mathbf{u}(k-1)), \mathbf{u}(k) \in [\mathbf{u}(k)]$
4: $\mathbf{y}(k) = g_{\mathbb{E}}(\mathbf{x}(k)), \mathbf{y}(k) \in [\mathbf{y}(k)]$
5: $\mathcal{L}_k = IAL(\mathcal{L}_{k-1}, [\mathbf{y}(k)], [\mathbf{u}(k)])$
6: **if** $\mathcal{L}_k = \emptyset$ **then**
7: *% kidnapping situation*
8: $\mathcal{L}_k = \{([0, x_{1_{max}}], [0, x_{2_{max}}], [0, 2\pi])\}$
9: **end if**
10: **end for**
11: **return** $\mathcal{L}_k$

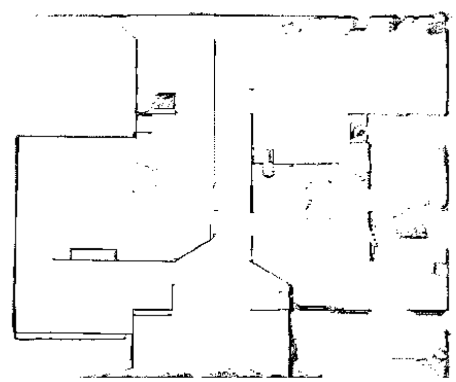


Fig. 8: The map of the CAROTTE arena, obtained by mixing the data sets of two robots

data $y_i(k)$ do not correspond to an obstacle in the map (too many outliers that lead to a kidnapping situation). Figure 9 represents a successful localization recovery performed after a kidnapping situation.

V. The CAROTTE robots' architecture

In this section the presented platform named MiniRex[2] was designed at the university of Angers by the LISA and the IUT[3] of Angers in order to participate at the third round of the CAROTTE contest (June 2012). The materiel architecture is presented Figure 10.

A simplified (for legibility reasons) software architecture of the MiniRex Robots is presented Figure 11. The actual architecture is composed of 22 modules (C++ library), each one with a specific functionality. Functions are used for the communication between the modules. Those modules have been developed with the LORIA[4] [19], [20]. The presented architecture has three levels:

- The main program which is a state machine dealing with the different steps of the exploration: startup, navigation, acquisition...,
- a set of four blocks detailed later,

[2]MINI Robot for EXPloration
[3]Institute of Technology
[4]Lorraine Research Laboratory in Computer Science and its Applications

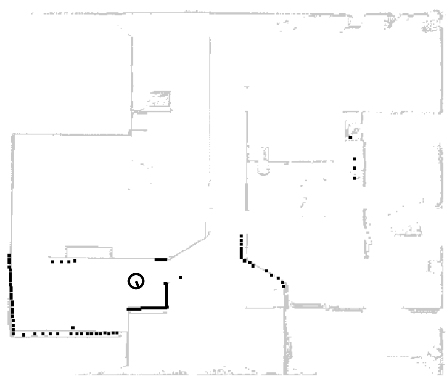


Fig. 9: A result of a localization recovery after a kidnapping situation. The grey cells correspond to the obstacles in the map and the black points correspond to the data set used to perform the localization. The black circle corresponds to a solution $\mathbf{x}(k) \in [\mathbf{x}(k)]$ of the global localization.

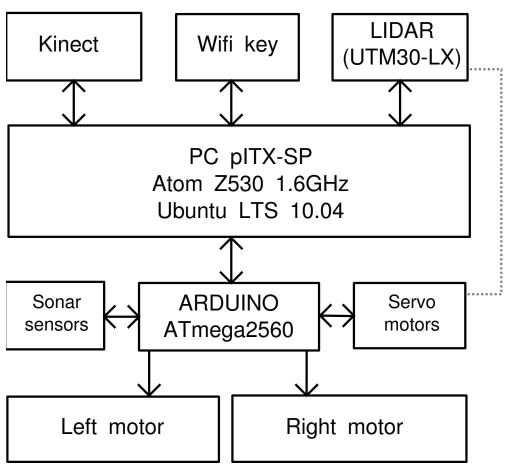


Fig. 10: The materiel architecture of the MiniRex robots. Note that the servomotors command the orientation of the LIDAR.

- and an interface between the actuators/sensors and higher levels.

Note that the four presented blocks are composed of several modules. Here is a presentation of those blocks:

- The CheckUp block enables to verify the performance of all the part (materiel and software) of the robot. This check up is done before the beginning of an exploration mission.
- The Expl-O-matic block computes in a distributive way the exploration strategy and the dispersion of the robots in the environment.
- The Plan-O-matic block enables the path planning and the building of a path free map.
- The Slam-O-matic block deals with the simultaneous localization and the map building during the moving of the robots.

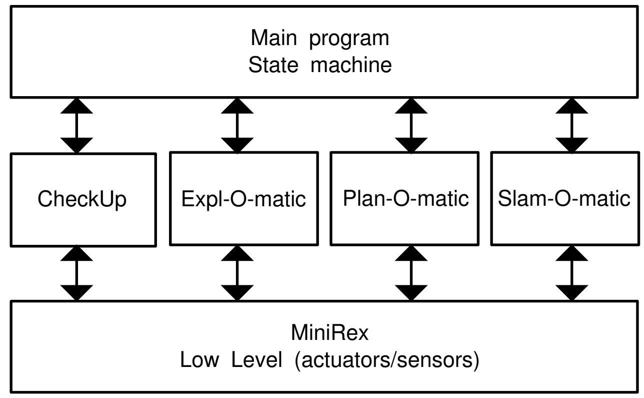


Fig. 11: A simplified software architecture presentation of the MiniRex robots.

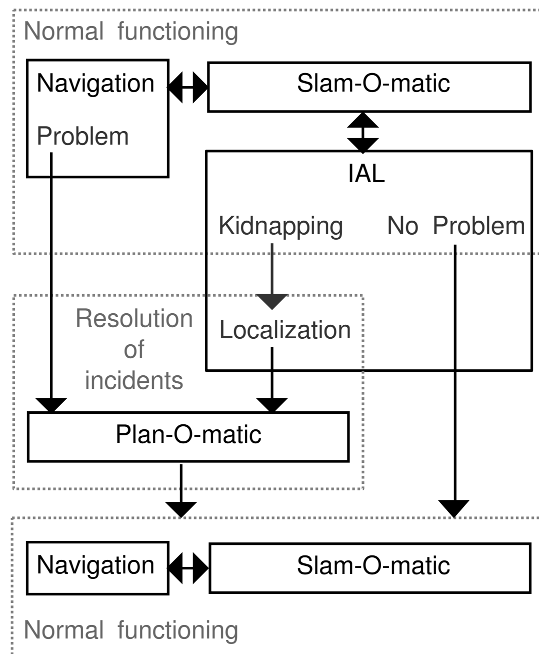


Fig. 12: Integration of the IAL module.

The objective of the CAROTTE challenge is to build a map of an unknown indoor environment. It is the module Slam-0-matic that builds this map (it is based on minima search). In order to avoid possible inconsistencies during the mission a kidnapping algorithm can be implemented. Figure 12 shows how to use the IAL module for this purpose. The IAL algorithm will detect an inconsistency pose and will re-localize the robot if necessary (by using a LIDAR measurement set and the map already built).

VI. Conclusion

In this paper a set membership approach has been presented to deal with the kidnapping problem. This approach allows to localize in a guaranteed way a robot in its environment even in a kidnapped situation. The IAL algorithm shows that interval analysis is an interesting tool for the localization problems. Thanks to experimental results this approach appears to be efficient in a real context.

According to those results, interval analysis is an interesting alternative to the classic probabilistic approaches.

VII. Acknowledgements

This work has been partially supported by the French National Research Agency (Agence Nationale de la Recherche, ANR) and Defence Procurement Agency (Direction Générale de l'Armement, DGA) through the Cartomatic project in ANR Carotte Challenge.

References

[1] I. J. Cox and G. T. Wilfong, *Autonomous robot vehicles*, Springer-Verlag, Ed., 1990.

[2] J. Borenstein, H. R. Everett, and L. Feng, *Navigating Mobile Robots: Systems and Techniques*. Natick, MA, USA: A. K. Peters, Ltd., 1996.

[3] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*, ser. Intelligent robotics and autonomous agents. MIT Press, 2005.

[4] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations*. Cambridge, MA: MIT Press, June 2005.

[5] R. Smith, M. Self, and P. Cheeseman, *Estimating uncertain spatial relationships in robotics*. New York, NY, USA: Springer-Verlag New York, Inc., 1990, pp. 167–193.

[6] K. O. Arrast and S. J. Vestli, "Hybrid, high-precision localisation for the mail distributing mobile robot system mops," in *IEEE International Conference on Robotics and Automation*, 1998.

[7] P. Jensfelt and S. Kristensen, "Active global localization for a mobile robot using multiple hypothesis tracking." *IEEE Transactions on Robotics*, vol. 17, no. 5, pp. 748–760, 2001.

[8] J. Leonard and H. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 376 –382, 1991.

[9] R. Smith and P. Cheeseman, "On the Representation and Estimation of Spatial Uncertainty," 1986.

[10] H. Durrant-Whyte, S. Majumder, S. Thrun, M. de Battista, and S. Scheding, "A bayesian algorithm for simultaneous localisation and map building," in *Robotics Research*, ser. Springer Tracts in Advanced Robotics, R. Jarvis and A. Zelinsky, Eds. Springer Berlin / Heidelberg, 2003.

[11] G. Cen, N. Matsuhira, J. Hirokawa, H. Ogawa, and I. Hagiwara, "Mobile robot global localization using particle filters," in *International Conference on Control, Automation and Systems (ICCAS)*, 2008, pp. 710–713.

[12] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte Carlo localization for mobile robots," *Proceedings 1999 IEEE International Conference on Robotics and Automation (ICRA)*, vol. 128, no. February, pp. 1322–1328, 1999.

[13] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust monte carlo localization for mobile robots," *Artificial Intelligence Journal*, 2001.

[14] L. Zhang, R. Zapata, and P. Lépinay, "Self-adaptive Monte Carlo localization for mobile robots using range sensors," in *Proceedings of the International Conference on Intelligent robots and systems (IROS)*, 2009, pp. 1541–1546.

[15] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis*, 1st ed. Springer, Sept. 2001.

[16] A. Neumaier, *Interval Methods for Systems of Equations (Encyclopedia of Mathematics and its Applications)*. Cambridge University Press, Jan. 1991.

[17] F. Rossi, P. v. Beek, and T. Walsh, *Handbook of Constraint Programming (Foundations of Artificial Intelligence)*. New York, NY, USA: Elsevier Science Inc., 2006.

[18] M. Kieffer, L. Jaulin, . Walter, and D. Meizel, "Robust autonomous robot localization using interval analysis," *Reliable Computing*, vol. 6, pp. 337–362, 2000.

[19] A. Bautin, O. Simonin, and F. Charpillet, "Stratégie d'exploration multi-robot fondée sur les champs de potentiels artificiels," in *Journées Francophones sur les Systèmes Multi-Agents (JFSMA)*, Valenciennes, France, Oct. 2011. [Online]. Available: http://hal.inria.fr/inria-00598608

[20] ——, "Towards a communication free coordination for multi-robot exploration," in *6th National Conference on Control Architectures of Robots (CAR 2011)*.