Interval SLAM for underwater robots; a new experiment

Fabrice LE BARS [*] **Alain BERTHOLOM** [**] **Jan SLIWKA** [*]
Luc JAULIN [*]

*[*] ENSIETA (Ecole Nationale Supérieure des Ingénieurs des Etudes et Techniques d'Armement), 2 rue François Verny, 29806 Brest Cédex 09 FRANCE (e-mail: lebarsfa@ensieta.fr, sliwkaja@ensieta.fr, jaulinlu@ensieta.fr)
[**] GESMA (Groupe d'Etudes Sous-Marines de l'Atlantique), 29200 Brest FRANCE (e-mail: alain.bertholom@dga.defense.gouv.fr)*

Abstract: This paper provides a new submarine experiment and shows the results of an interval SLAM (Simultaneous Localization And Mapping) method. The SLAM problem is cast into a constraint satisfaction problem where interval propagation algorithms are particularly efficient. The resulting method is applied to the localization of a submarine robot from the GESMA (Groupe d'Etudes Sous-Marines de l'Atlantique), the *Daurade* during an experiment in the Douarnenez bay, in Brittany (France).

Keywords: SLAM; interval analysis; underwater robot; constraint propagation; set-membership estimation; localization.

1. INTRODUCTION

In this paper, we describe the results of an interval approach developed to deal with a SLAM (Simultaneous Localization And Mapping) problem in a submarine context (see Leonard and Durrant-Whyte [1992] for the general SLAM problem). The SLAM problem for an autonomous robot moving in an unknown environment is to build a map of this environment while simultaneously using this map to compute its location. Most of the approaches presented for the SLAM problem cast the problem into a state estimation problem for a dynamic system, by including the marks locations among the state variables to be estimated (Castellanos and Tardos [1999], Dissanayake et al. [2001], Montemerlo et al. [2003]). See also Williams et al. [2001], Ruiz et al. [2004], Eustice et al. [2005] for the case of autonomous underwater robots. Most of the proposed solutions are based on probabilistic estimation techniques (Kalman filtering, Bayesian estimation, particle filters, see Thrun et al. [2005]) which aim at blending data with some state equations of the robot.

For the experiment to be presented, we will use a set-membership approach for SLAM (see *e.g.*, Di Marco et al. [2004], Di Marco et al. [2001]) based on interval arithmetic initially developed for the *Redermor* submarine (see Jaulin [2009]) through an improved version of GESMI (Guaranteed Estimation of Sea Mines with Intervals), a software developed to build the trajectory of the robot, get the position of the marks... (Jaulin et al. [2006], Jaulin et al. [2007], figure 1).

In this problem, we will try to find an envelope for the trajectory of the robot and to compute sets which contain some detected marks. Note that for our experiment, the detections of the marks are performed by a human opera-
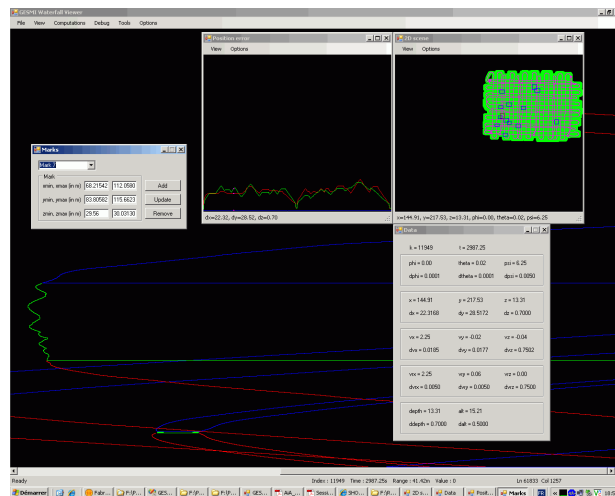

Fig. 1. Screenshot of the new part of GESMI, that computes and shows the position of the marks, the trajectory envelope, the position error and helps for the identification of the marks among the different detections.

tor, after the mission of the robot. Thus, our problem can be considered as an offline SLAM.

Set-membership methods have often been considered for the localization of robots (see, *e.g.*, Meizel et al. [1996], Halbwachs and Meizel [1996], in the case where the problem is linear and also Caiti et al. [2002] when the robot is underwater). Interval analysis has been shown to be efficient in several SLAM applications (see Drocourt et al. [2005] and Porta [2005] where it has been applied to the SLAM of wheeled robots).

In situations where strong nonlinearities are involved, interval analysis has been shown to be useful (see, *e.g.*, Meizel et al. [2002], where the first localization of an actual robot has been solved with interval methods). Our problem will lead us to a huge set of nonlinear equations which can be solved efficiently using interval analysis and constraint propagation (see, *e.g.*, Jaulin et al. [2001] for notions related to interval constraint propagation and applications). There are other robotics applications such as state estimation (see Raissi et al. [2004]), dynamic localization of robots (see Gning and Bonnifait [2006]), control of robots (see Lydoire and Poignet [2003] and Vinas et al. [2006]) or topology analysis of configuration spaces (see Delanoue et al. [2006]) where interval constraint propagation methods have been successful.

In this paper, we first describe the robot to be considered in Section 2. Then in Section 3, the interval approach for SLAM will be presented. The experiment as well as the results will be described in Section 4. Section 5 concludes the paper.

2. THE *DAURADE* AUV (AUTONOMOUS UNDERWATER VEHICLE)

The robot *Daurade* to be considered in our application is an AUV developed by ECA and used by the GESMA (a center of the French defense ministry which supervises most of the research in French underwater robotics) and the SHOM (French Navy Hydrographic and Oceanographic Service). This robot (figure 2) has a length of 5 m, a diameter of 0.70 m and a weight of 1 ton (see Vrignaud and Meyrat [2009]). It can navigate autonomously up to 10 hours at a speed of 4 knots (max speed 8 knots) and up to a depth of 300 m. Its position estimation is expected to be with an accuracy better than 50 m after 10 km of mission thanks to accurate sensors and an embedded Kalman filter (we do not know its parameters).

The *Daurade* is a multi sensor AUV designed for rapid environmental assessment operations. It has to acquire and transmit environment data on a poorly known area. It is one of the successors of the *Redermor* (see Jaulin [2009], Jaulin et al. [2006], Jaulin et al. [2007]). It encloses on board processing capabilities that allow adaptive mission planning.

Several sensors are integrated in the *Daurade*. Here is the list of those that will provide the data used by our SLAM algorithm :

- DGPS(Differential Global Positioning System). Since electromagnetic waves (1.57542 GHz) do not propagate through the water, the GPS is operational only when the robot is at the surface of the ocean.
- Lateral sonar Klein 5500. It provides an image of the bottom of the ocean (often called *waterfall*). The image is of about 75 m large (corresponds to the rangescale parameter of the sonar) for more than 20 km high (corresponding to the length covered by the robot during its mission). Although there is an image from the right and the left of the robot, we have only used the right sonar image. The use of sonars for improving the localization and the navigation has

Fig. 2. The *Daurade* AUV (Courtesy of DGA/GESMA).

already been proven on several robotics applications (see, *e.g.*, Leonard and Durrant-Whyte [1992]).
- Doppler Velocity Loch (DVL) 1200 kHz type Navigator. This sensor makes possible to compute the speed vector of the robot $\mathbf{v}_r$. It emits ultrasonic waves which are reflected on the bottom of the ocean. Since the bottom is immobile, this sensor is able to compute an estimation of its speed using the Doppler effect. It outputs also the distance between the robot and the bottom of the ocean.
- IXSEA U-PHINS INS (Inertial Navigation System). It gets the three Euler angles (*i.e.*, the orientation) of the robot.
- Pressure sensor. It gets the depth of the robot (*i.e.* the distance between the robot and the surface of the ocean).

3. SET-MEMBERSHIP APPROACH USED FOR SLAM

3.1 Interval arithmetic

An interval is a closed and connected subset of $\mathbb{R}$. $\mathbb{IR}$ is the set of the intervals of $\mathbb{R}$. Consider two intervals $[x]$ and $[y]$ and an operator $\diamond \in \{+, -, ., /\}$, we define $[x] \diamond [y]$ as the smallest interval which contains all feasible values for $x \diamond y$, if $x \in [x]$ and $y \in [y]$ (see Moore [1979]). For instance

$$\begin{aligned} [-1,3] + [2,5] &= [1,8], \\ [-1,3].[2,5] &= [-5,15], \\ [-1,3]/[2,5] &= [-\frac{1}{2}, \frac{3}{2}]. \end{aligned} \quad (1)$$

If f is an elementary function such as sin,cos,... we define $f([x])$ as the smallest interval which contains all feasible values for $f(x)$, if $x \in [x]$. A box or interval vector $[\mathbf{x}]$ is a vector whose components are intervals:

$$[\mathbf{x}] = [x_1^-, x_1^+] \times ... \times [x_n^-, x_n^+] = [x_1] \times ... \times [x_n]. \quad (2)$$

3.2 Contraction and propagation

Consider a constraint $\mathcal{C}$ (*i.e.*, an equation or an inequality), some variables $x_1, x_2, ...$ involved in $\mathcal{C}$ and prior interval domains $[x_i]$ that contain all feasible values for the x_i's. Interval arithmetic makes possible to contract the domains $[x_i]$ without removing any feasible values for the x_i's. For

instance, consider the equation $x_3 = x_1 + x_2$ where the domains for x_1, x_2, x_3 are given by $[x_1] = [-\infty, 5], [x_2] = [-\infty, 4]$ and $[x_3] = [6, \infty]$. These domains can be contracted to $[x_1'] = [2, 5], [x_2'] = [1, 4]$ and $[x_3'] = [6, 9]$. The resulting interval calculus is as follows:

$$\begin{aligned}
x_3 = x_1 + x_2 \Rightarrow x_3 &\in [6, \infty] \cap ([-\infty, 5] + [-\infty, 4]) \\
&= [6, \infty] \cap [-\infty, 9] = [6, 9]. \\
x_1 = x_3 - x_2 \Rightarrow x_1 &\in [-\infty, 5] \cap ([6, \infty] - [-\infty, 4]) \\
&= [-\infty, 5] \cap [2, \infty] = [2, 5]. \\
x_2 = x_3 - x_1 \Rightarrow x_2 &\in [-\infty, 4] \cap ([6, \infty] - [-\infty, 5]) \\
&= [-\infty, 4] \cap [1, \infty] = [1, 4].
\end{aligned} \quad (3)$$

This contraction procedure can be performed with much more complex constraints. A contraction operator is called a *contractor* (Chabert and Jaulin [2009]). When more than one constraint are involved, the contractions are performed sequentially several times, until no more significant contractions can be observed. It can be shown that the box to which the method converges does not depend on the order to which the contractors are applied (Jaulin et al. [2001]), but the computing time is highly sensitive to this order. There is no optimal order in general, but in practice, one of the most efficient is called *forward-backward propagation*. It consists in writing the whole set of equations under the form $\mathbf{f}(\mathbf{x}) = \mathbf{y}$ where $\mathbf{x}$ and $\mathbf{y}$ correspond to quantities that can be measured (*i.e.*, some prior interval domains are given for them). Then, using interval arithmetic, the intervals are propagated from $\mathbf{x}$ to $\mathbf{y}$ in a first step (*forward propagation*) and, in a second step, the intervals are propagated from $\mathbf{y}$ to $\mathbf{x}$ (*backward propagation*).

3.3 Principle of the method used

In our set membership context, we can describe our SLAM problem as follows (see Jaulin [2009])

$$\begin{cases}
\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) & \text{(evolution equation)} \\
\mathbf{y} = \mathbf{g}(\mathbf{x}) & \text{(observation equation)} \\
\mathbf{z}_i = \mathbf{h}(\mathbf{x}, \mathbf{u}, \mathbf{m}_i) & \text{(mark equation)}
\end{cases} \quad (4)$$

where $\mathbf{x}(t)$ is the state vector of the robot, $\mathbf{u}(t)$ is its input vector, $\mathbf{y}(t)$ is its output vector, $\mathbf{m}_i$ is the location of the ith mark and $\mathbf{z}_i$ is the measurement vector related to the ith mark. We consider that

- At some time t, we have a box $[\mathbf{x}](t)$ containing the state vector:
$$\mathbf{x}(t) \in [\mathbf{x}](t) \quad (5)$$
- For all $t \in [t_0, t_f]$, we have boxes enclosing the $\mathbf{u}(t)$ and $\mathbf{y}(t)$:
$$\forall t \in [t_0, t_f], \mathbf{u}(t) \in [\mathbf{u}](t) \text{ and } \mathbf{y}(t) \in [\mathbf{y}](t) \quad (6)$$
- We have a finite subset $\mathcal{M} \subset [t_0, t_f] \times \{1, \ldots, i_{\max}\}$ such that
$$\mathbf{z}_i(t) \in [\mathbf{z}_i](t). \quad (7)$$
If $(t, i) \in \mathcal{M}$, the mark i has been detected at time t.

In a submarine context, the robot motion can be described by the following state equation:
$$\dot{\mathbf{p}} = \mathbf{R}_{Euler}(\varphi, \theta, \psi).\mathbf{v}_r \quad (8)$$
where

- $\mathbf{p} = (p_x, p_y, p_z)$ are the coordinates of the robot expressed in the frame $(\mathbf{O}, \mathbf{i}, \mathbf{j}, \mathbf{k})$ where $\mathbf{O}$ is the location of the robot at time $t_0 = 0$, the vector

$\mathbf{i}$ indicates the north, $\mathbf{j}$ indicates the east and $\mathbf{k}$ is oriented toward the center of the earth
- $\mathbf{R}_{Euler}(\varphi, \theta, \psi)$ represent the Euler rotation matrix (the heading ψ, the pitch θ, the roll φ) of the robot measured by the INS
- $\mathbf{v}_r$ represents the speed of the robot measured by the DVL sensor.

The observation equation is
$$\mathbf{y} = \mathbf{p} \quad (9)$$
because p_x, p_y can be measured sometimes by the GPS and p_z is obtained from the pressure sensor and the altimeter.

Using the sonar waterfall data, a human operator can detect marks and get the distance mark-robot. The z position of the mark detected can be deduced from the submarine depth p_z and altitude a. Therefore the mark equation for the ith mark is
$$\mathbf{z}_i = \begin{pmatrix} \mathbf{r}(\mathbf{p}, \varphi, \theta, \psi, \mathbf{m}_i) \\ \mathbf{m}_i^z \end{pmatrix} \quad (10)$$
where $\mathbf{r}(\mathbf{p}, \varphi, \theta, \psi, \mathbf{m}_i)$ is the distance mark-robot and $\mathbf{m}_i^z$ is its z position.

As a consequence, our set-membership SLAM using interval arithmetic in a submarine context can be written as follows:
$$\begin{cases}
\dot{\mathbf{p}} = \mathbf{R}_{Euler}(\varphi, \theta, \psi).\mathbf{v}_r \\
\mathbf{y} = \mathbf{p} \\
\mathbf{z}_i = \begin{pmatrix} \mathbf{r}(\mathbf{p}, \varphi, \theta, \psi, \mathbf{m}_i) \\ \mathbf{m}_i^z \end{pmatrix}
\end{cases}. \quad (11)$$

4. APPLICATION

4.1 Experiment and data collected

The *Daurade* AUV was launched in the Douarnenez bay for an experiment of about 5 hours (from $t = 0$ to $t = 18445.75$) of autonomous scanning of an area making vertical and horizontal trajectories above a limited zone. During its mission, the robot localizes itself using all its sensors combined through a Kalman filter. The robot tries to follow a trajectory defined by a list of predefined waypoints to be reached. These waypoints are chosen by a human operator before the mission. The precision of the embedded localization algorithm is estimated around 50 m for 10 km which is sufficient to control the robot for the mission in the ocean without being lost.

The AUV took GPS data at the surface 3 times during the mission : (i) during the first seconds, (ii) in the middle of the mission, before changing the survey direction from horizontal to vertical and (iii) at the end of the mission:

$$\begin{aligned}
t = 0s & \quad \begin{pmatrix} p_x \\ p_y \end{pmatrix} = \begin{pmatrix} 0.332443 \\ -0.091429 \end{pmatrix} \\
t = 9210s & \quad \begin{pmatrix} p_x \\ p_y \end{pmatrix} = \begin{pmatrix} 10.277683 \\ 663.195976 \end{pmatrix} \\
t = 18419s & \quad \begin{pmatrix} p_x \\ p_y \end{pmatrix} = \begin{pmatrix} 875.402443 \\ 724.556278 \end{pmatrix}
\end{aligned}$$

A file containing the navigation data and error estimation for the orientation and the speed is produced during the mission. Thirteen different marks $\mathbf{m}_1, \ldots, \mathbf{m}_{13}$ have been seen through 33 detections on the waterfall produced by the sonar. For example, the marks selected can be brighter
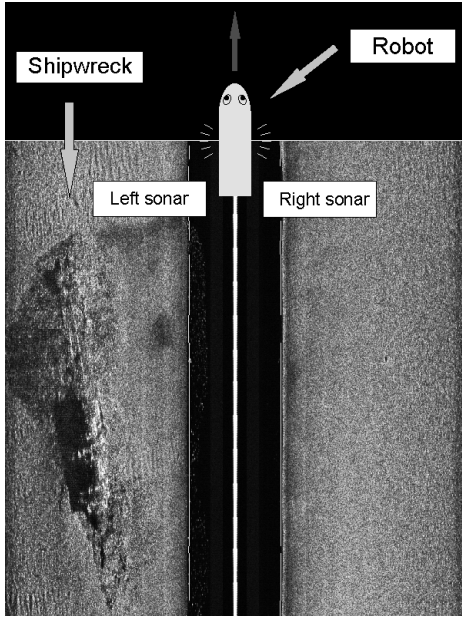
Fig. 3. Part of the sonar image taken by the robot (screenshot from the SONARPRO software).

points on the sonar image due to rocks, shipwrecks parts... (figure 3). A scrolling of the waterfall was performed after the mission by a human operator using the SONARPRO software to get the list of marks detections. He was able to get an estimation $\tilde{r}(t)$ of the distance $r(t)$ from the robot to the mark detected at time t. The actual distance $r_i(t)$ between the robot and the mark $\mathbf{m}_i$ is supposed to satisfy the relation

$$r_i(t) \in [\tilde{r}_i(t) - 1, \tilde{r}_i(t) + 1] \qquad (12)$$

(Jaulin [2009]), assuming that the precision of the sonar is of 1 m. Each mark detection adds constraints on the position of the robot at the time of the detection as well as on the position of the mark in the experiment zone.

Although the detection of a mark and the matching between marks detected several times were performed manually once the robot has accomplished its mission, the new part of GESMI helps the human operator to correct wrong mark matching (see figure 4). GESMI enables to see sonar data, to add a detection, to see the envelopes enclosing the trajectory of the robot and to generate the predicted waterfall for the detected marks. It also makes possible to see the positions in the waterfall where the marks should have been seen and confirm these predictions by adding a new detection of the mark. The human operator should first detect each mark only once. The time and distance robot-mark of other possible detections of these marks will be predicted thanks to the navigation data and the operator will just have to confirm the new detections provided by GESMI and make them more accurate.

4.2 Results

From the navigation data and the 3 GPS positions measured at the beginning, the middle and the end of the mission, we can build an envelope for the trajectory using a forward propagation performed on the evolution equations provided by Equations (11). Figure 5 represents the esti-
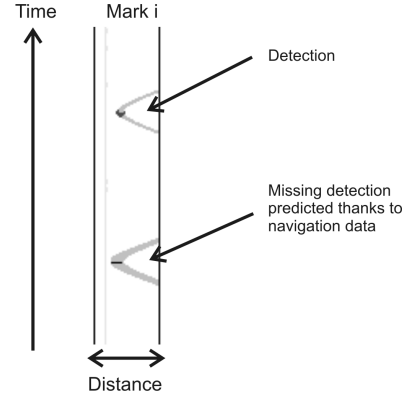

Fig. 4. Reconstructed waterfall showing a detected mark and the prediction of the distance robot-mark at another time using navigation data.
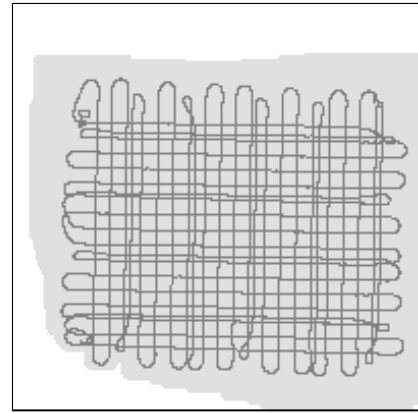

Fig. 5. Trajectory envelope after a forward propagation (less than 1 s of computing time), painted grey. The center of the corresponding boxes is painted dark grey. The frame box is $[-200, 1000]$m$\times[-200, 1000]$m (in the (x,y) plane).

mated envelope with the center of the corresponding boxes. Note that due to the nature of the mission of the robot, the trajectory overlaps and it is difficult to interpret the accuracy of the envelope without figure 6 which represents the errors (half of the box width containing the estimated positions) on the positions x, y and z with respect to t.

A backward propagation improves the position estimation and produces the results shown on figures 7 and 8. We see that the position errors in x and y have a threshold. This is due to the difference between the estimated position and the measured position at the GPS point. If the estimated position had been equal to the measured position, there would have been no contraction of the trajectory and we would have a triangular shape, without any threshold.

More measurements and more information on the experiment imply a better accuracy on the results. Therefore, if we take into account the detections made by the human operator on the sonar image, the estimation of the trajectory of the robot can be improved with the estimation of the position of the marks (see figures 9 and 10). After a global interval propagation taking into account the marks and all other sensors we have a localization of the AUV and the marks with an accuracy of less than 40 m in less than 10 s of computing time. The accuracy of the
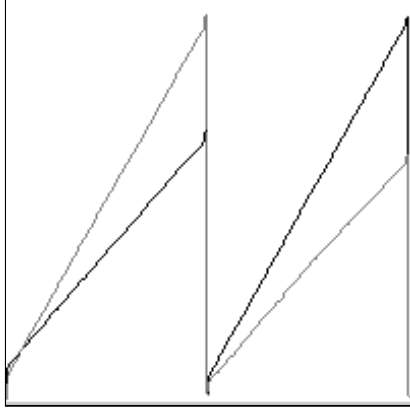
Fig. 6. Position errors on x (black), y (dark grey) and z (light grey) with respect to t obtained after a forward propagation. The frame box is $[0, 18445.75]\mathrm{s} \times [0, 200]\mathrm{m}$.
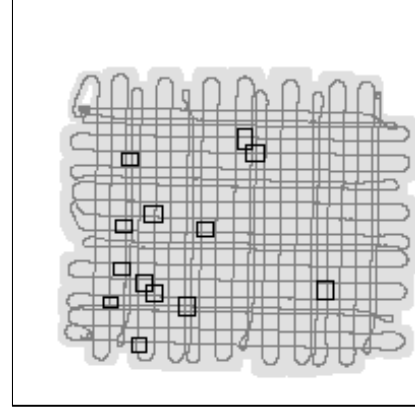

Fig. 7. Trajectory envelope after a forward/backward propagation (less than 2 s of computing time).

Fig. 8. Position errors after a forward/backward propagation.

Fig. 9. Trajectory envelope after several contractions taking into account marks detected in the waterfall (less than 10 s of computing time). The envelopes of the marks position are painted black.
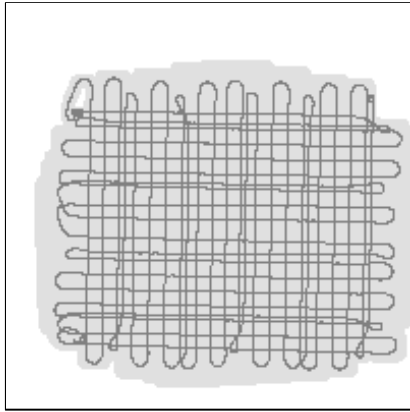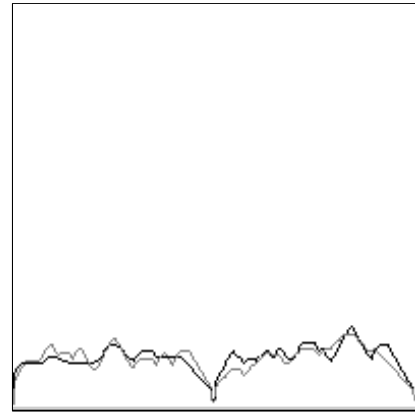

Fig. 10. Position errors after several contractions taking into account the marks detected in the waterfall.

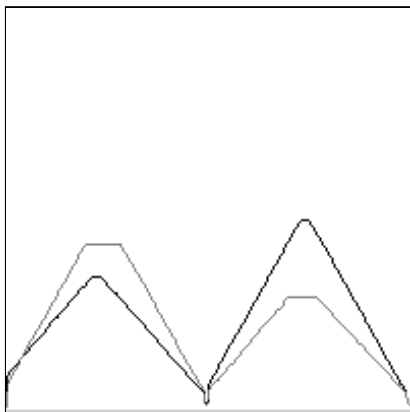positions provided after the interval SLAM compared to that obtained by the Kalman filter of the *Daurade* is poor (the improvement is less than 10 m). This poor improvement is due to the fact that (i) the data provided by the INS and by the Loch Doppler are very accurate and make possible to localize the robot with good accuracy (less than 50 m) without any SLAM (ii) A GPS point has been taken at the middle of the mission which makes the SLAM less useful and (iii) the Kalman filter assumes that the state and measurement noises are white, Gaussian and centered which makes possible to have an error which increases with the square root of t (which is often optimistic in practice). The error computed by the interval method considers the worst case scenario and only assumes bounds on the errors. As a consequence, the interval method provides more conservative errors. However, since no linearization is performed and since no assumption about the Gaussian nature of the noise is done,... the errors provided by the interval approach can be considered as reliable. In the context of our experiment, no inconsistencies have been detected, i.e., all data with all errors provided by the navigation system are consistent with the marks detections and with the envelope detected by the interval propagation. As a consequence, the interval propagation can be used to validate the navigation system. Of course, a light validation of the navigation system is already done when we compare the predicted position with the GPS position when the AUV surfaces. But here, with the interval method, we check more: all marks detections, the envelope provided by the navigation system, the GPS points,... are all consistent together.

5. CONCLUSION

In this paper, we have shown a new experiment and the results of an interval constraint propagation used to

solve an offline SLAM problem. We described the *Daurade AUV* and in particular its sensors. Then, we reminded the main principles of the interval propagation method used, described the context of the experiment and provided an analysis of the results of our interval method on the data collected.

REFERENCES

A. Caiti, A. Garulli, F. Livide, and D. Prattichizzo. Set-membership acoustic tracking of autonomous underwater vehicles. *Acta Acustica united with Acustica*, 5(88): 648–652, 2002.

J. A. Castellanos and J. D. Tardos. Mobile robot localization and map building: A multisensor fusion approach. *Kluwer*, 1999.

G. Chabert and L. Jaulin. QUIMPER, A Language for Quick Interval Modelling and Programming in a Bounded-Error Context. *Artificial Intelligence*, 173: 1079–1100, 2009.

N. Delanoue, L. Jaulin, and B. Cottenceau. Using interval arithmetic to prove that a set is path-connected. *Theoretical Computer Science, Special issue: Real Numbers and Computers*, 351(1):119–128, 2006.

M. Di Marco, A. Garulli, A. Giannitrapani, and A. Vicino. A set theoretic approach to dynamic robot localization and mapping. *Autonomous Robots*, 16(1):23–47, 2004.

M. Di Marco, A. Garulli, S. Lacroix, and A. Vicino. Set membership localization and mapping for autonomous navigation. *International Journal of Robust and Nonlinear Control*, 7(11):709–734, 2001.

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

C. Drocourt, L. Delahoche, B. Marhic E. Brassart, and A. Clerentin. Incremental construction of the robot's environmental map using interval analysis. *Global Optimization and Constraint Satisfaction: Second International Workshop, COCOS 2003*, 3478:127–141, 2005.

R. Eustice, H. Singh, J. Leonard, M. Walter, and R. Ballard. Visually navigating the rms titanic with slam information filters. In *Proceedings of Robotics: Science and Systems (RSS)*, Cambridge, MA, USA, 2005.

A. Gning and P. Bonnifait. Constraints propagation techniques on intervals for a guaranteed localization using redundant data. *Automatica*, 42(7):1167–1175, 2006.

E. Halbwachs and D. Meizel. Bounded-error estimation for mobile vehicle localization. *CESA'96 IMACS Multiconference (Symposium on Modelling, Analysis and Simulation)*, pages 1005–1010, 1996.

L. Jaulin. A Nonlinear Set-membership Approach for the Localization and Map Building of an Underwater Robot using Interval Constraint Propagation. *IEEE Transaction on Robotics*, 25(1):88–98, 2009.

L. Jaulin, A. Bertholom, F. Dabe, and M. Legris. A set approach to the simultaneous localization and map building; application to underwater robots. *ICINCO 2007*, 2007.

L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics.* Springer-Verlag, London, 2001.

L. Jaulin, M. Legris, and F. Dabe. GESMI, un logiciel pour l'aide à localisation de mines sous-marines. In *JIME 2006 (Journées Identification et Modélisation Expérimentale)*, Poitiers (France), 2006.

J. J. Leonard and H. F. Durrant-Whyte. *Directed Sonar Sensing for Mobile Robot Navigation.* Kluwer, Boston, 1992.

F. Lydoire and P. Poignet. Nonlinear predictive control using constraint satisfaction. In *In 2nd International Workshop on Global Constrained Optimization and Constraint Satisfaction (COCOS)*, pages 179–188, 2003.

D. Meizel, O. Lévêque, L. Jaulin, and E. Walter. Initial localization by set inversion. *IEEE Transactions on Robotics and Automation*, 18(6):966–971, 2002.

D. Meizel, A. Preciado-Ruiz, and E. Halbwachs. Estimation of mobile robot localization: geometric approaches. In M. Milanese, J. Norton, H. Piet-Lahanier, and E. Walter, editors, *Bounding Approaches to System Identification*, pages 463–489. Plenum Press, New York, NY, 1996.

M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003. IJCAI.

R. E. Moore. *Methods and Applications of Interval Analysis.* SIAM, Philadelphia, PA, 1979.

J.M. Porta. Cuikslam: A kinematics-based approach to slam. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2436–2442, Barcelona (Spain), 2005.

T. Raissi, N. Ramdani, and Y. Candau. Set membership state and parameter estimation for systems described by nonlinear differential equations. *Automatica*, 40:1771–1777, 2004.

I.T. Ruiz, S. de Raucourt, Y. Petillot, and D. Lane. Concurent mapping and localization using sidescan sonar. *IEEE Journal of Oceanic Engineering*, 39(2):442–456, 2004.

S. Thrun, W. Bugard, and D. Fox. *Probabilistic Robotics.* MIT Press, Cambridge, M.A., 2005.

P. Herrero Vinas, M. A. Sainz, J. Vehi, and L. Jaulin. Quantified set inversion algorithm with applications to control. *Reliable computing*, 11(5):369–382, 2006.

C. Vrignaud and J. Meyrat. Use of a dvl in a autonomous underwater vehicle for a rapid environmental assessment. In *ADCP in Action*, San Diego, 2009.

S. Williams, G. Dissanayake, and H. Durrant-Whyte. Towards terrain-aided navigation for underwater robotics. *Advanced Robotics*, 15(5):533–549, 2001.