

## Improving the Precision of AUVs Localization in a Hybrid Interval-Probabilistic Approach Using a Set-Inversion Strategy

Renata Neuland<sup>\*,‡</sup>, Renan Maffei<sup>\*,§</sup>, Luc Jaulin<sup>†,¶</sup>, Edson Prestes<sup>\*,||</sup>, Mariana Kolberg<sup>\*\*\*</sup>

<sup>\*</sup>Institute of Informatics, Federal University of Rio Grande do Sul, Porto Alegre, Brazil

<sup>†</sup>ENSTA Bretagne, Brest, France

One of the fundamental tasks of robotics is to solve the localization problem, in which a robot must determine its true pose without any knowledge on its initial location. In underwater environments, this is specially hard due to sensors restrictions. For instance, many times, the localization process must rely on information from acoustic sensors, such as transponders. We propose a method to deal with this scenario, that consists in a hybridization of probabilistic and interval approaches, aiming to overcome the weaknesses found in each approach and improve the precision of results. In this paper, we use the set inversion via interval analysis (SIVIA) technique to reduce the region of uncertainty about robot localization, and a particle filter to refine the estimates. With the information provided by SIVIA, the distribution of particles can be concentrated in regions of higher interest. We compare this approach with a previous hybrid approach using contractors instead of SIVIA. Experiments with simulated data show that our hybrid method using SIVIA provides more accurate results than the method using contractors.

**Keywords:** Global localization; hybridization; interval analysis; SIVIA; contractors; particle filter.

US

### 1. Introduction

Estimating the precise localization of autonomous underwater vehicles (AUVs) is an important requirement for a wide range of underwater applications, such as area coverage, monitoring, demining, reconnaissance of objects in hostile waters and prevention of algae blooms in drink water reservoirs [1–3]. Furthermore, solving this problem is a very challenging task in robotics, specially given that the sensors availability in underwater environments is quite restricted when compared to terrestrial environments. Many studies about autonomous underwater vehicles were presented in the last years, detailing the variety of navigation techniques, the different sensors used, among other topics [4–6]. In our work, we focus on the development of a strategy for obtaining robust robot localization based on seamounts detection.

The localization problem is one of the three fundamental tasks of robotics along with mapping and navigation, and consists in estimating the robot pose using sensor information [7]. If the initial pose of the robot is known, the problem is called *local localization* or *position tracking*. In this case, the robot, dealing with noisy readings, uses the sensors observations to correct its motion estimate. However, if the initial pose is not known, there is a significant increase in the difficulty of the problem. In this situation, called *global localization*, the goal is to first reduce the uncertainty associated to the multiple hypotheses of robot pose, since the robot can be initially in any place of the environment [7].

The resolution of the global localization problem depends directly on the proper treatment of uncertainties, that are generally associated to the sensor readings, but also to the ambiguities of the environment. Two of the main approaches to handle uncertainties in robotics are the probabilistic and the interval approaches [7,8].

Probabilistic approaches, such as those based on Bayesian filtering, are extensively used to treat problems of high dimensionality in various fields. Among them, the

Received 25 July 2014; Revised 28 September 2014; Accepted 28 September 2014; Published 28 October 2014. This paper was recommended for publication in its revised form by Guest Editor: Jason Gu.

Email Addresses: <sup>‡</sup>[rcneuland@inf.ufrgs.br](mailto:rcneuland@inf.ufrgs.br), <sup>§</sup>[rmaffei@inf.ufrgs.br](mailto:rmaffei@inf.ufrgs.br), <sup>¶</sup>[luc.jaulin@ensta-bretagne.fr](mailto:luc.jaulin@ensta-bretagne.fr), <sup>||</sup>[prestes@inf.ufrgs.br](mailto:prestes@inf.ufrgs.br), <sup>\*\*\*</sup>[mariana.kolberg@inf.ufrgs.br](mailto:mariana.kolberg@inf.ufrgs.br)

particle filter is a method widely used in robotics, mainly due to its capacity of treating nonlinear models and multimodal distributions [9]. However, the quality of the solution found with the particle filter is dependent on the number of samples used. If the uncertainty is too large, the particle filter may require a very large number of particles to cover the solution space, making its use prohibitive. In addition, due to the randomness in the sampling and resampling process, an unfortunate sequence of samples can cause wrong convergence of the method, leading to poor final results in the robot localization.

In contrast, interval approaches deal with high dimensional problems through the reduction of domains. Such reductions are performed based on the constraints of the problem. Given the correct modeling of the problem, we can say that the result obtained with an interval method is mathematically guaranteed, i.e. this method provides a delimited region that contains the correct solution. Nonetheless, the final solution may not be enough significant due to the conservative nature of interval methods, where no feasible solution is discarded. In other words, purely interval results are composed of a set of possibilities, all of which are equally true. When the final result is a big set, it does not provide enough information about the robot localization.

We proposed, in [10], a hybrid method combining probabilistic and interval strategies to solve the global localization problem. The method reduces the uncertainty about the robot pose using contractors from interval analysis, then propagates particles inside the resulting space to enhance the quality of the localization. To summarize, the results have boundaries of uncertainty well defined and mathematically guaranteed, but with a higher precision given by the particles distribution.

In this work, we make a detailed study about the strategy in [10], proposing modifications to improve the quality of its results, in terms of precision. We analyze the use of a more precise method from interval analysis, called set inversion via interval analysis (SIVIA) [11], in comparison with the former strategy using contractors. SIVIA reduces the search space to a region that delimits the robot localization using bisections. This technique presents a computational cost higher than contractors, however, by partitioning the solution space in multiple boxes its accuracy is superior. The metrics of our analysis are the precision in localization and the computation time.

With this research it was possible to improve the precision in the robot localization using the hybrid probabilistic-interval approach. The hybrid approach with SIVIA obtains enhanced information about the robot localization and keeps the benefits presented in the first hybrid approach [10]. Some of the benefits are the higher coverage of the uncertainty region and a fast detection in case of wrong convergence.

The paper is organized as follows. We first present the related work about the localization problem in Sec. 2. We show some of the most relevant methods using probabilistic and interval analysis, and the existent hybrid methods. In Sec. 3, we present a background on interval analysis, detailing both SIVIA and the contractors strategies. In Sec. 4, we present our hybrid methods for global localization: the one from [10]. In Sec. 5 is presented the modification of the hybrid method, now using SIVIA. In Sec. 6, we evaluate and discuss the methods through the analysis of results from simulated experiments. Finally, in Sec. 7, we conclude and discuss future work.

## 2. Related Work

To perform the localization of AUVs, the methods in literature generally rely on inertial sensors, such as accelerometers and gyroscopes; acoustic transponders, i.e. beacons with known positions; and geophysical sensors, such as sonars and cameras, that are able to detect and identify features in the environment [6]. Corke *et al.* present a method for AUV localization based on visual odometry using stereo cameras, along with an acoustic localization system based on geometric intersections of uncertainty regions [12]. Another method based on vision is presented by Kim *et al.* for AUVs in structured environments [13]. The method detects artificial landmarks inserted in the environment through a template matching technique and then uses a particle filter to estimate the localization of the vehicle. In such method, the dead-reckoning information is used for the prediction step of the filter, while the landmark detection is used for the update step.

Probabilistic approaches, such as particle filters, are the core of many methods in robotics. Ko *et al.* present a particle filter strategy for localization of AUVs using dead-reckoning for prediction, and the time difference of arrival (TDOA) of acoustic signals emitted from multiple beacons for correction [14]. Forney *et al.* use a particle filter to perform, from an AUV, the tracking of a ‘tagged’ agent, e.g. a second AUV or a shark [15]. An active control system is used to make the AUV follow the trail of the agent, in order to stay close to the source of the acoustic signals. Some strategies use other Bayesian filters, instead of particle filters. This is the case of Wang *et al.*, that proposes a localization method combining Extended Kalman filter (EKF) and Moving Horizon Estimation (MHE) for AUVs using a single beacon [16]. Maurelli *et al.* propose a sonar-based approach for structured and unstructured environments using a particle filter integrated with an EKF [17]. The localization starts with a particle filter (since there are no knowledge about the initial robot pose), once the filter converges the

localization process passes to an EKF. If the filter diverges, for instance in a situation of kidnapped robot, the particle filter is re-started with uniform samples to regain the diversity of solutions.

On the other hand, there is a recent growth in popularity of interval approaches, in which the measurements inaccuracy is expressed in terms of bounds on the possible errors. An earlier work in this area is the one from Meizel *et al.* which uses set-membership estimation to localize a vehicle using range measurements [18]. It has the advantage of not requiring a large number of data — good to avoid outliers — and can deal with ambiguities by allowing disconnected components of localization estimates. According to Meizel *et al.*, SIVIA was used and provided systematic, efficient and general solutions.

Jaulin propose a similar interval method more robust to outliers, reliable in respect to nonlinearities and tested it in an AUV localization [19]. The method based on relaxed set inversions (RSIVIA) is an extension of SIVIA [11], and consists in performing the intersection of all intervals of uncertainty except a small number  $q$  of intervals, where  $q$  is an estimate of the number of outliers. The tests were performed considering a 2D underwater environment.

Based on such technique, Langerwisch and Wagner present a method for localization using laser sensors, which is able to detect and mark outliers in the laser scans [20]. Another method of localization using range sensors is the one from Guyonneau *et al.* [21]. They define localization as a constraint satisfaction problem (CSP) and combine bisections and contractors techniques to solve the problem. Their experimental results showed the efficiency of the method in a real context.

Seddik *et al.* [22] deal with localization of underwater robots using an acoustic signal. The proposed algorithm is based on contractors and bisections and the authors introduce the time constraint satisfaction problem (TCSP). A set of measurements is used to compute the robot position. This set is stored in a buffer and used according to a time window. In this way, the solution is not computed from a single measurement, but from a set of measurements collected during this time window.

Finally, merging probabilistic and interval approaches to improve the quality of the estimation process has been the focus of recent studies. Abdallah *et al.* propose the box particle filter (BPF), which is a method that allows the reduction of the number of particles in comparison to traditional particle filters, by defining particles using boxes (cartesian products of intervals — one interval for each space dimension) [23]. In the traditional particle filter, efficiency and precision are mainly dependent on the number of particles used, which can be significantly large for some applications. In that aspect, the use of interval data in BPF makes it more efficient, reducing the computational time.

On the other hand, the BPF does not show improvements in terms of results precision.

In the combination of particle filter and intervals, a box may represent the uncertainty about the localization of a single particle, but may also represent a set of particles distributed in the area covered by the box. It was based on the idea of particles contained inside boxes, that we proposed, in [10], our hybrid method to solve the global localization problem.

### 3. Background on Interval Analysis

The study of interval techniques began about 50 years ago, and since then, interval approaches have been applied to a large number of problems from different areas, including robotics [8]. The main feature of such approaches is that by applying interval operations over data properly modeled using intervals, like an estimate of a robot pose, there is a guarantee that the correct solution will be contained in the resulting interval.

This section presents some fundamental concepts of interval computation, and also two approaches, used in this work, for reducing the size of intervals without discarding viable solutions: contractors and SIVIA.

#### 3.1. Interval computation

A real interval  $[x]$  is considered a connected subset of  $\mathbb{R}$ , and it is composed of a lower bound  $\underline{x}$  and an upper bound  $\bar{x}$ . We define a real interval to represent one-dimensional data as

$$[x] = [\underline{x}; \bar{x}] = \{x \in \mathbb{R} | \underline{x} \leq x \leq \bar{x}\}.$$

If a multi-dimensional representation is required, we can model data using boxes [8]. A box is a subset of  $\mathbb{R}^n$  that can be described by a Cartesian product of intervals,

$$[\mathbf{x}] = [x_1] \times [x_2] \times \cdots \times [x_n],$$

where  $[x_i] = [\underline{x}_i; \bar{x}_i]$  to  $i = 1, \dots, n$ , and  $n$  represents the number of dimensions of the box  $[\mathbf{x}]$ . Thus, each interval component  $[x_i]$  is a projection in one of the Cartesian axes.

If we want to refine the representation of some data to increase its precision, then a single box may not be appropriate. A possible alternative is to use a *subpaving*, which is a set of nonoverlapped boxes that together represent a solution set [8].

Classic operations used in real computations  $\diamond = (+; -; *; /)$  can be naturally extended to interval computations [24]. Considering the real intervals  $[x] = [\underline{x}; \bar{x}]$  and  $[y] = [\underline{y}; \bar{y}]$ , interval computations are defined by

$$[x] \diamond [y] = \{w \diamond z \in \mathbb{R} | w \in [x], z \in [y]\}.$$

Interval computations using functions are also possible. An interval image  $\mathcal{I}$  of a real function  $f([x])$  can be defined by

$$\begin{aligned}\mathcal{I}(f, [x]) &= [\underline{I}; \bar{I}] \\ &= [\min\{f(x) | x \in [x]\}; \max\{f(x) | x \in [x]\}].\end{aligned}$$

A function  $f$  composed of arithmetic operators and elementary functions can be an inclusion function. An interval function is an inclusion function  $[f]$  if it satisfies the property [8,25]

$$f([x]) \subset [f]([x]).$$

When the inclusion function result is the smallest possible interval or box that contains the result, the function is called minimal  $[f]^*([x])$  [8].

One of the main goals in interval approaches is to represent information with the smallest possible representation. Nevertheless, interval computations do not always generate so precise results. In this sense, some techniques, such as contractors and SIVIA, can be used to reduce the size of intervals based on constraints.

### 3.2. Contractors

Contractors are used to reduce domains from a set of constraints. An operator  $\mathcal{C}$  is a contractor if given a constraint  $c$  and a domain  $[x]$  it satisfies the following properties [26]:

- **Completeness:**  $(c \cap [x]) \subset \mathcal{C}([x])$   
All values in the interval  $[x]$  that satisfy the constraint  $c$  are contained in the result of the operator  $\mathcal{C}([x])$ , in other words, no feasible solution is discarded.
- **Contractance:**  $\mathcal{C}([x]) \subset [x]$   
The resulting box of the contraction  $\mathcal{C}([x])$  is contained in the initial domain  $[x]$ .

There are different kinds of contractors in the literature, but one of the most used in approaches for robotics is the *forward-backward* contractor [8]. Since a constraint can be written like a function  $y = f(x)$  or in the inverse form  $x = f^{-1}(y)$ , forward-backward works in two steps:

- (i) **Forward:** Contract  $y$  using  $[y] \cap [f]([x])$
- (ii) **Backward:** Contract  $x$  using  $[x] \cap [f^{-1}]( [y] )$

For example, considering the equation  $x_3 = x_1 + x_2$ , and the initial domains  $[x_1] = [-\infty; 5]$ ,  $[x_2] = [-\infty; 4]$  and  $[x_3] = [6; \infty]$ .

- $x_3 = x_1 + x_2 \Rightarrow z \in [6; \infty] \cap ([-\infty; 5] + [-\infty; 4]) = [6; \infty] \cap [-\infty; 9] = [6; 9]$ .
- $x_1 = x_3 - x_2 \Rightarrow x \in [-\infty; 5] \cap ([6; 9] - [-\infty; 4]) = [-\infty; 5] \cap [2; \infty] = [2; 5]$ .
- $x_2 = x_3 - x_1 \Rightarrow y \in [-\infty; 4] \cap ([6; 9] - [-\infty; 5]) = [-\infty; 4] \cap [1; \infty] = [1; 4]$ .

After contraction the new domains are:  $[x_1] = [2; 5]$ ,  $[x_2] = [1; 4]$  and  $[x_3] = [6; 9]$ .

Contractors can be used to increase the precision of the results of inclusion functions. Thus, they can be useful to deal with high-dimensional spaces, due to their polynomial complexity in the number of constraints [8].

One disadvantage is that, despite ensuring the inclusion of the correct result in the final solution, its precision may not be as high as expected. Also, by always maintaining a single box, the representation of disconnected sets of solutions may lead to high uncertainty, specially when they are too far apart.

### 3.3. SIVIA

SIVIA algorithm, presented by Jaulin and Walter, is a method that solves the set inversion problem [11], defined by

$$\mathbb{X} = \{\mathbf{x} \in \mathbb{R}^n | f(\mathbf{x}) \in \mathbb{Y}\} = f^{-1}(\mathbb{Y}).$$

In other words, we want to define  $\mathbb{X}$  given a function  $f$  and the image function  $\mathbb{Y}$ . The main idea of the SIVIA algorithm is to generate *subpavings* based on the bisection of the solution space. The algorithm can be summarized in four iterative steps, given an initial search space modeled using a box  $[x]$ :

- (i) If  $f([x])$  does not have intersection with  $\mathbb{Y}$ ,  $[x]$  is discarded.
- (ii) If  $f([x])$  is contained in  $\mathbb{Y}$ ,  $[x]$  is considered part of the solution.
- (iii) If  $f([x])$  has intersection, but is not contained in  $\mathbb{Y}$ ,  $[x]$  is considered undetermined. If the box is undetermined and its width  $w([x])$  is bigger than a predefined limit, then the box is bisected, and the resulting boxes will be tested.
- (iv) If the box  $[x]$  has a width smaller than the predefined limit and it is undetermined,  $[x]$  is considered part of the solution.

An example of solutions obtained with the SIVIA algorithm can be seen in Fig. 1. These figures show two scenarios where a robot observes landmarks in the environment. On the left, the robot observes the landmark L1 at some given distance (modeled as an interval  $[\underline{d}; \bar{d}]$ ). Therefore, the robot can be anywhere inside a shade region around L1. This circular region represents the solution of the problem and it is composed of a large set of small adjacent boxes. On the right, the robot observes two landmarks, L1 and L2. Now the solution is the intersection of the region constrained by the observation of L1 and the region constrained by the observation of L2.

By using a set of boxes to represent the result, SIVIA can be much more precise than contractors. It can also



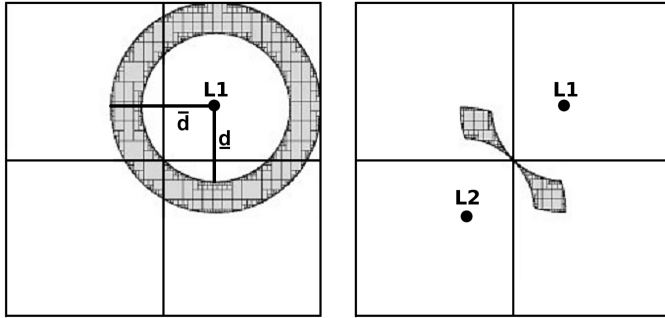


Fig. 1. Example of *subpavings* obtained with SIVIA, in a scenario with one constraint (left) and two constraints (right).

represent with high precision disconnected sets of solutions. However, the bisection process has high computational cost. In fact, it has an exponential complexity according to the number of dimensions of the space [8].

#### 4. Solving the Localization Problem with a Hybridization of Probabilistic and Interval Approaches

Particle filter is one of the most prominent approaches to solve the localization problem [9]. Its main idea consists in approximating a target probability distribution using a set

of particles (weighted samples), and updating those particles in an iterative process of weighting and resampling. In localization, each particle represents a hypothesis about the robot localization, and its weight is given by the similarity between the robot observations and the virtual observations of the particle.

Figure 2 shows an example of the evolution of a particle filter, where the robot is depicted in black, the landmarks in blue, the particles in shades of yellow and red, and the weighted average of the particles in green. The color scale of the particles represents the importance factor assigned to each particle, i.e. the closer to yellow, the lower the importance factor, while the closer to red, the higher the important factor.

Among the main advantages of particle filters are the capacity to represent multi-modal distributions (they do not require parametric assumptions, such as Kalman filter-based approaches), the ease of implementation, and the possibility of increasing precision given the available computational resources. On the other hand, particle filters suffer with the particle depletion problem, associated to the differences between proposal and target distributions and the randomness in the resampling step, that discard good particles, eventually losing diversity [7].

In [10], we proposed a hybrid method for localization using a particle filter and contractors. The basic idea is to reduce the uncertainties in the robot localization by performing interval computations while applying the particle

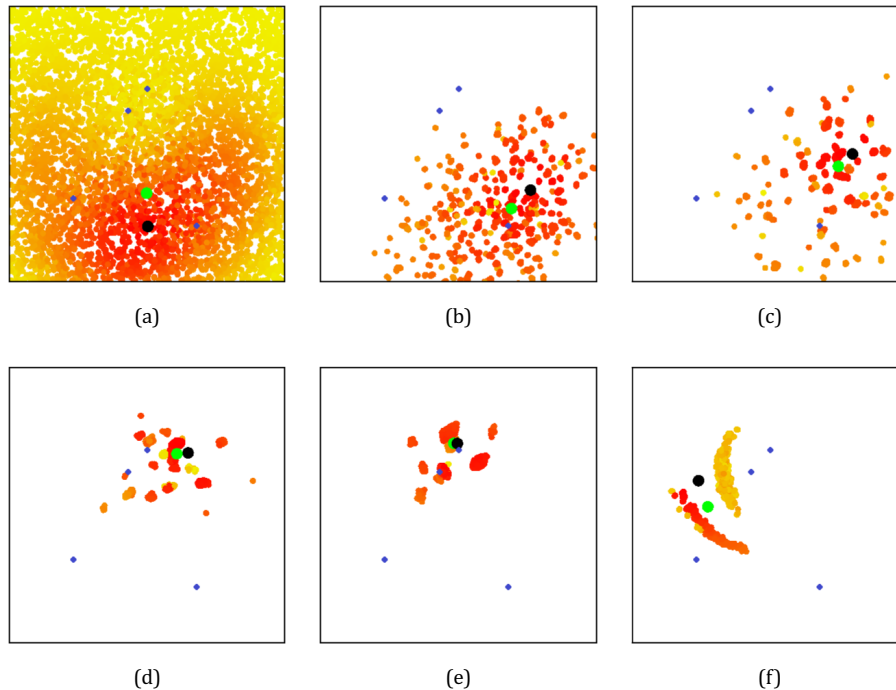


Fig. 2. (Color online) Example of evolution of the traditional particle filter. Robot (black circle), landmarks (blue points), particles (yellow to red points), weighted average of the particles (green circle).

**Algorithm 1** Hybrid Method using Contractors

---

```

1: Data:  $f, [\mathbf{x}], u_{1:n}, z_{0:n}$ 
2: Result:  $\mathbb{S}$ 

3:  $\mathbb{S} = \emptyset$ 
4:  $\mathbb{F} =$  set of constraints  $f$  using  $z_0$ ;
5:  $[\mathbf{x}] = \text{Contract}([\mathbf{x}], \mathbb{F})$ ;
6:  $\mathbb{P} = \text{createParticles}(m, [\mathbf{x}])$ ;
7: for  $t = 1 : n$  do
8:    $\text{moveI}([\mathbf{x}], [\mathbf{u}])$ ;
9:    $\mathbb{F} =$  set of constraints  $f$  using  $z_t$ ;
10:   $[\mathbf{x}] = \text{Contract}([\mathbf{x}], \mathbb{F})$ ;
11:   $\text{move}(\mathbb{P}, u)$ ;
12:  for  $j = 0 : m$  do
13:    if  $p^j \cap [\mathbf{x}] = \emptyset$  then
14:       $\mathbb{P} = \mathbb{P} \setminus p^j$ ;
15:       $\mathbb{P} = \mathbb{P} \cup \text{createParticles}(1, [\mathbf{x}])$ ;
16:    end if
17:  end for
18:   $\text{weight}(\mathbb{P})$ ;
19:   $\text{resampling}(\mathbb{P})$ ;
20:   $\mathbb{S} = \text{avgParticle}(\mathbb{P})$ ;
21: end for
22: return  $\mathbb{S}$ 

```

---

filter. With contractors, the uncertainty about the robot localization is reduced to a box. As this box is computed using interval rules, we can assume that the true robot pose is contained in the box. Nonetheless, even with the reduction of uncertainty, the size of the resulting intervals may be larger than the expected. Thus, to refine the robot localization we apply particle filter over the interval result.

The hybrid method is described in Algorithm 1. Initially, the algorithm does not have any information about the robot localization, only about the initial search space, i.e. the whole environment. The boundaries of this space can be defined or not. Undefined limits are modeled through infinity intervals (e.g.  $[-\infty; +\infty]$ ). Then, the first set of constraints is defined using the initial sensors observations  $z_0$  (line 4) and passed to the contractors (line 5). After, particles are generated inside the current solution space  $[\mathbf{x}]$  (line 6).

The constraints are generated according to the  $k$  observed markers at instant  $t$  and they are described by

$$\{\sqrt{(x - x_k)^2 + (y - y_k)^2 + (z - z_k)^2}\} \subset [d_k],$$

where  $x_k \in [x_k]$ ,  $y_k \in [y_k]$  and  $z_k \in [z_k]$  define the localization of marker  $k$ , and the interval  $[d_k]$  is defined from the robot observation  $z_t^k$  associated to the marker  $k$ , considering the sensor uncertainties  $s_e$ ,

$$[d_k] = [\underline{d}_k; \overline{d}_k] = [z_t^k - s_e; z_t^k + s_e].$$

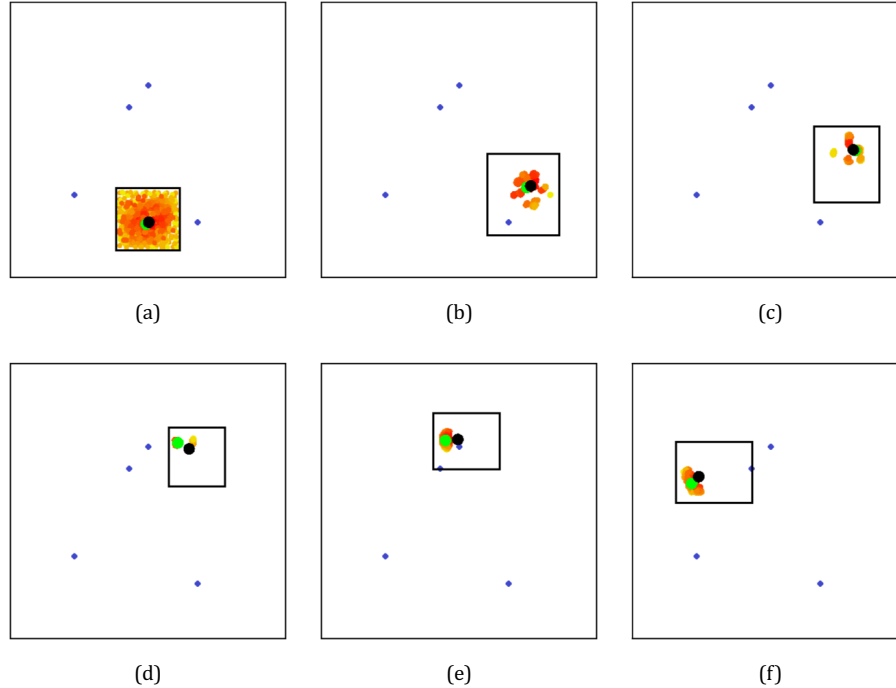


Fig. 3. (Color online) Example of evolution of the hybrid approach using contractors. Robot (black circle), current search space (black box), landmarks (blue points), particles (yellow to red points), weighted average of the particles (green circle).

If the exact marker location is known, the resulting intervals are punctual.

For each iteration, the solution box moves according to the robot motion  $u_t$  (line 8). Then, the box is contracted using the constraints associated to the current observations (lines 9–10). The box movement is given by the inclusion function of the motion model,

$$[\mathbf{x}]_{t+1} = [f]([\mathbf{x}]_t) = [\mathbf{x}]_t + [\mathbf{R}]_{([\phi], [\theta], [\psi])} \times [\mathbf{v}], \quad (1)$$

where  $[\mathbf{v}]$  is the linear speed of the vehicle and  $[\mathbf{R}]_{([\phi], [\theta], [\psi])}$  is a 3D rotation matrix using the Euler angles representing its orientation.

The particle filter is updated by moving the particles with the robot motion model (line 11). Then, each particle is tested against the box representing the search space (lines 12–17). Particles that fall outside the box are discarded, and, in their place, new particles are randomly created inside the box. In the next step, the current set of particles is weighted and resampled (lines 18–19). During the weighting process, all particles receive an importance factor according to the robot observation. Particles with observations similar to the robot observations receive higher importance factors. During the resampling process, a new set of particles is generated from the current set. We can use, for instance, the roulette wheel method, in which particles with higher importance factor have more probability to survive [7]. Finally, we choose the particles weighted average as the best estimate for the robot pose (line 20).

Figure 3 shows an example of the evolution of the hybrid method combining particle filter with contractors. We can see in comparison to Fig. 2, that the particles distribution becomes much more condensed using the hybrid approach, and closer to the real robot pose (black point).

## 5. Improving the Precision of Localization with a Set-Inversion Strategy

In [10], we showed that our hybrid method, described in last section, obtained considerable improvements in terms of precision when compared to the traditional particle filter alone. And, more important, this gain in precision does not degrade the system performance. Based on those results, we decided to examine if it is possible to obtain higher precision using methods more precise, such as SIVIA, while keeping low computational cost.

The method proposed in this work is a hybridization of SIVIA with a particle filter. Instead of using contractors, SIVIA is now used to reduce the region representing the robot localization, which is defined as search space for the particle filter. This approach enables a better spread of particles in regions of real interest, since, by using multiple

---

### Algorithm 2 Hybrid Method using SIVIA

---

```

1: Data:  $f, [\mathbf{x}], \varepsilon, u_{1:n}, z_{0:n}$ 
2: Result:  $\mathbb{S}$ 

3:  $\mathbb{S} = \emptyset$ 
4:  $\mathbb{F}$  = set of constraints  $f$ ;
5:  $\mathbb{D}$  = set of measurements  $z_0$ 
6:  $siviaResult = SIVIA(\mathbb{F}, \mathbb{D}, [\mathbf{x}], \varepsilon)$ ;
7:  $\mathbb{P} = createParticle(m, siviaResult)$ ;
8: for  $t = 1 : n$  do
9:    $[\mathbf{bBox}] = hull(siviaResult)$ ;
10:   $moveI([\mathbf{bBox}], [\mathbf{u}])$ ;
11:   $[\mathbf{x}] = [\mathbf{bBox}]$ ;
12:   $\mathbb{F}$  = set of constraints  $f$ ;
13:   $\mathbb{D}$  = set of measurements  $z_t$ ;
14:   $siviaResult = SIVIA(\mathbb{F}, \mathbb{D}, [\mathbf{x}], \varepsilon)$ ;
15:   $move(\mathbb{P}, u)$ ;
16:  for  $j = 0 : m$  do
17:    if  $p^j \cap siviaResult = \emptyset$  then
18:       $\mathbb{P} = \mathbb{P} \setminus p^j$ ;
19:       $\mathbb{P} = \mathbb{P} \cup createParticle(1, siviaResult)$ ;
20:    end if
21:  end for
22:   $weight(\mathbb{P})$ ;
23:   $resampling(\mathbb{P})$ ;
24:   $\mathbb{S} = avgParticle(\mathbb{P})$ ;
25: end for
26: return  $\mathbb{S}$ 

```

---

boxes, SIVIA approximates the space of possible robot poses in a much better way than with a single box.

Algorithm 2 shows the proposed method step-by-step. Once again, the initial search space is the whole environment, because there is no prior information about the robot pose. SIVIA reduces the space to a set of boxes (line 6) using the set of constraints defined by the markers positions (line 4) and the set of initial measurements (line 5). Then, the set of particles is generated over the reduced search space (line 7).

Just like before, for each iteration of the process, the solution space must move according to the robot motion. However, the estimation of robot motion, as defined in Eq. (1), for each one of the small boxes can be an extremely costly task. Not only that, but, by growing and moving each box independently, boxes overlapping is very likely to happen, which is not a proper representation of a subpaving. The solution to this problem is to bound all the boxes by the smallest possible box that contains all boxes, i.e. the hull of the boxes (line 9), and move this bounding box according

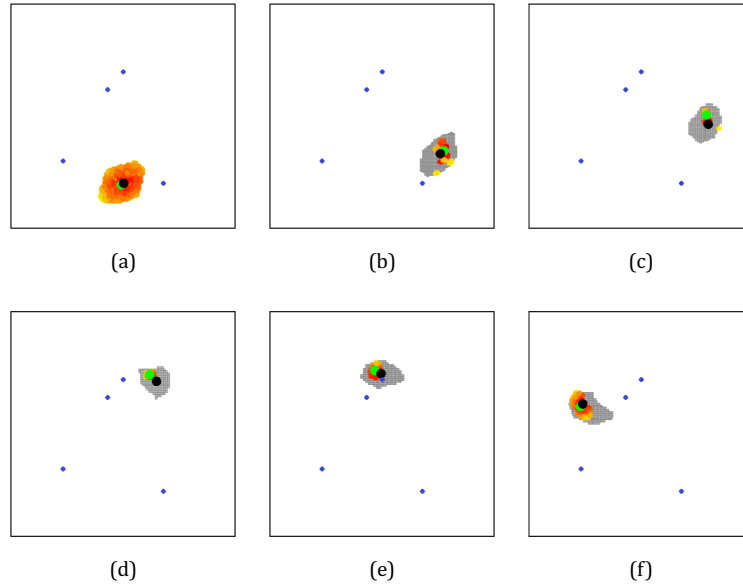


Fig. 4. (Color online) Example of evolution of the hybrid approach using SIVIA. Robot (black circle), current search space (gray region), landmarks (blue points), particles (yellow to red points), weighted average of the particles (green circle).

to the inclusion function defined in Eq. (1) (line 10). The resulting box is used as search space to the next measurement update (line 11). Using the set of constraints  $f$  (line 12) and the current set of measurements  $z_t$  (line 13), the algorithm performs the SIVIA method (line 14).

The remaining of the process remains the same of Algorithm 1. First, the set of particles is updated according to the robot motion (line 15). Then, all particles are tested against the interval results. Particles that fall outside the search space are discarded, and to each discarded particle, a new particle is randomly created inside the current search space (lines 16–21). The particles are weighted (line 22), resampled (line 23), and the weighted average of the particles is selected to represent the robot pose (line 24).

The evolution of the hybrid method using SIVIA is presented in Fig. 4. It is possible to observe that the set of boxes resulting of the interval operations (depicted by the small adjacent gray boxes) may strongly differ from a squared region.

Our goal in this work is to see if such quality improvement is enough to justify the high computational cost associated with SIVIA. Therefore, we present and discuss, in next section, the results of experiments comparing the hybrid approaches using contractors and SIVIA.

## 6. Experiments

Here we present and compare the results obtained by our hybrid approaches. The hybrid method with SIVIA was compared with the method presented in [10] and with

particle filter, one of the most popular approaches to deal with this kind of problem. The results were compared in terms of precision and computational cost. The graphs and tables presented in this section show the comparison among the three methods.

The experiments were performed in a simulated underwater environment containing distinguishable markers, with positions that are known *a priori*. We used MORSE Simulator [27] with a generic submarine, available in MORSE, equipped with loch-doppler sensor and gyroscope. The robot is able to detect and distinguish the markers, and to obtain information about its own linear velocity and orientation. To simulate the markers, transponders were spread in the environment.

We performed tests using three different environments, with the same size, defined by the box:

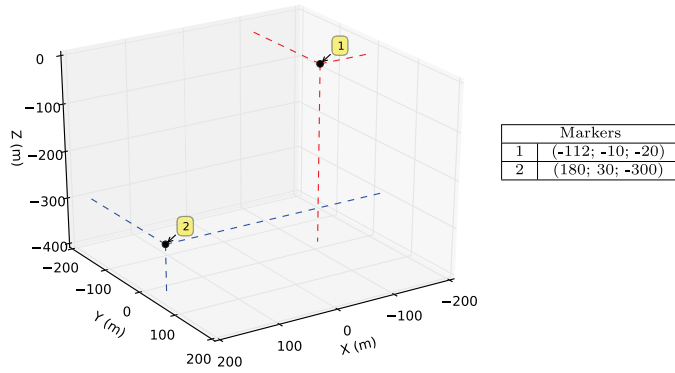
$$[\mathbf{x}] = [-200; 200] \times [-200; 200] \times [-400; 0].$$

The differences among the environments are in the number of markers used. The environments 1, 2 and 3 have respectively 2, 4 and 8 markers, positioned as shown in Fig. 5. Two different trajectories were tested, as shown in Fig. 6.

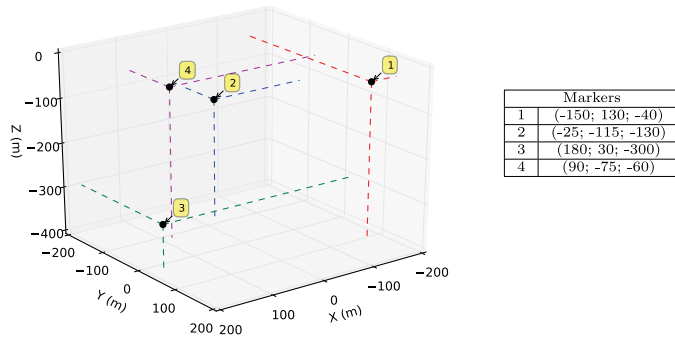
Table 1 shows the parameters of the tests, where  $\sigma$  is the standard deviation representing the error of the sensors and  $w$  is the parameter used to define the width of the interval.

All configurations were tested 10 times with each method, using 5000 particles in each test. Figures 7–10 show the behavior of the average error and standard deviation of error over time.

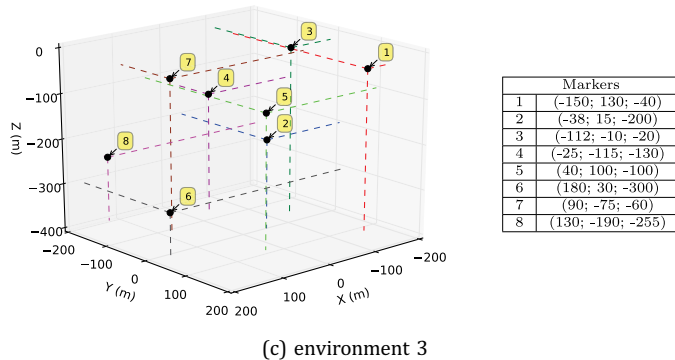




(a) environment 1



(b) environment 2

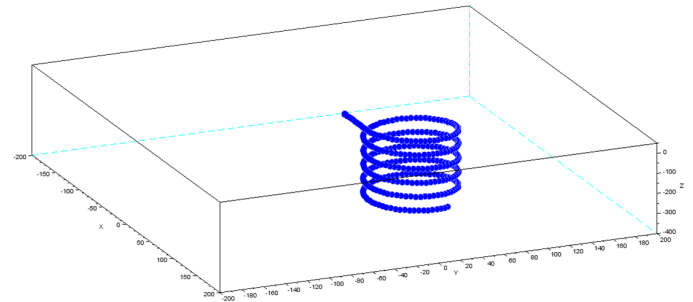


(c) environment 3

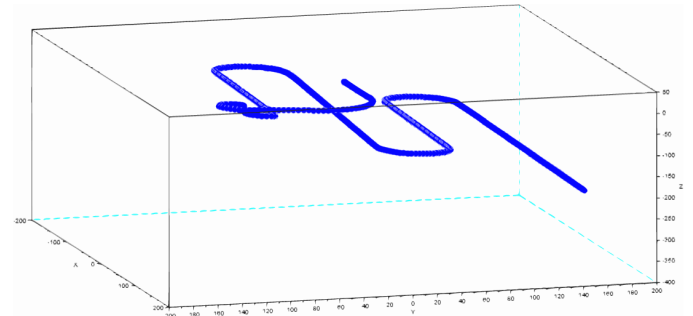
Fig. 5. Markers configuration in the environments.

Figure 7 shows the error behavior in tests using the trajectory 1. We can see that, in all environments, the errors obtained with the hybrid methods are smaller than the error obtained with the traditional particles filter. Paying attention to the difference between the scales is possible to see that the errors become smaller when more markers are used.

Figure 8 shows the errors obtained with the hybrid methods in the first trajectory using a different scale for a better view. We can see that the method using SIVIA has more precise results, with the average particles error consistently smaller than 0.5 m.



(a) Trajectories 1



(b) Trajectories 2

Fig. 6. Trajectories tested in the experiments.

Table 1. Parameters used during the experiments.

Measurement	Noise ( $\sigma$ = standard deviation)	$w$
Loch-doppler	0.04 m	$w_v = 3$
Gyroscope	0.02°	$w_o = 3$
Markers distance	0.3 m	$w_d = 3$

The results of the methods using trajectory 2 are presented in Fig. 9. Once again, both hybrid methods show better results than the traditional particle filter, as in the results presented in Fig. 7.

In Fig. 10, the results of the hybrid methods are presented in different scale. As before, the method using SIVIA obtained smaller errors than the method using contractors, specially in the environment with less markers.

In the graphs representing the average error over time (Figs. 7–10) there is an expressive difference among the standard deviation of the methods. The hybrid method using SIVIA presents the smallest standard deviation, followed by the hybrid method using contractors. This is related to the particles dispersion, which is limited by the size of the solution space. Given that the region defined by SIVIA is smaller than the region defined by contractors, the particles dispersion is smaller, as the standard deviation. In this context, the traditional particle filter distributes particles in

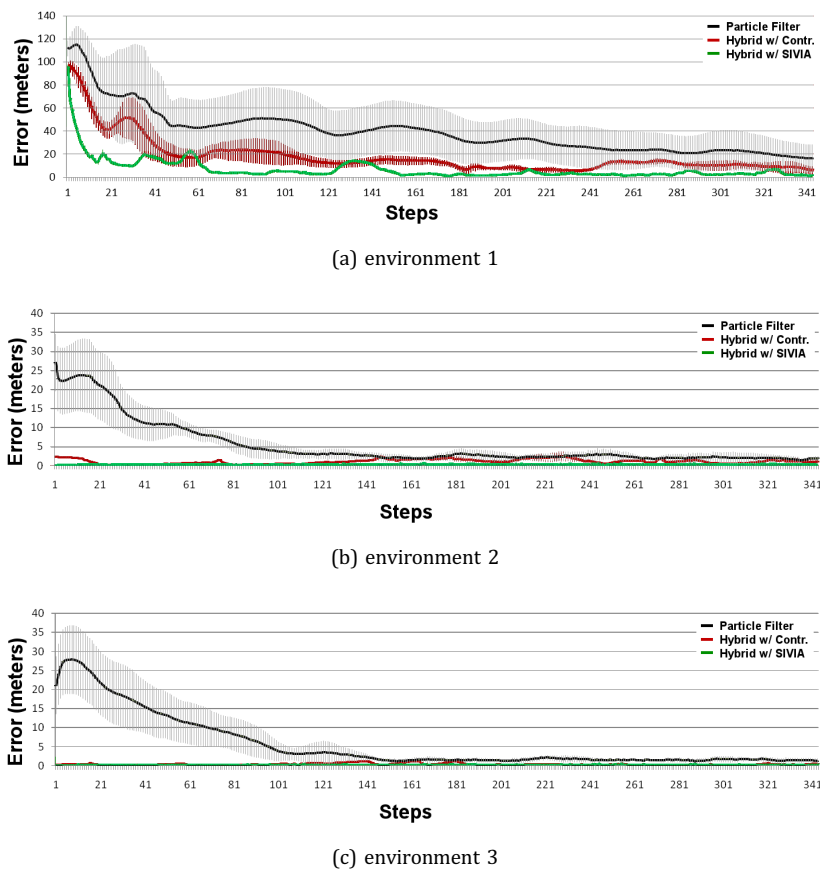


Fig. 7. Comparison of the average particles error for trajectory 1.

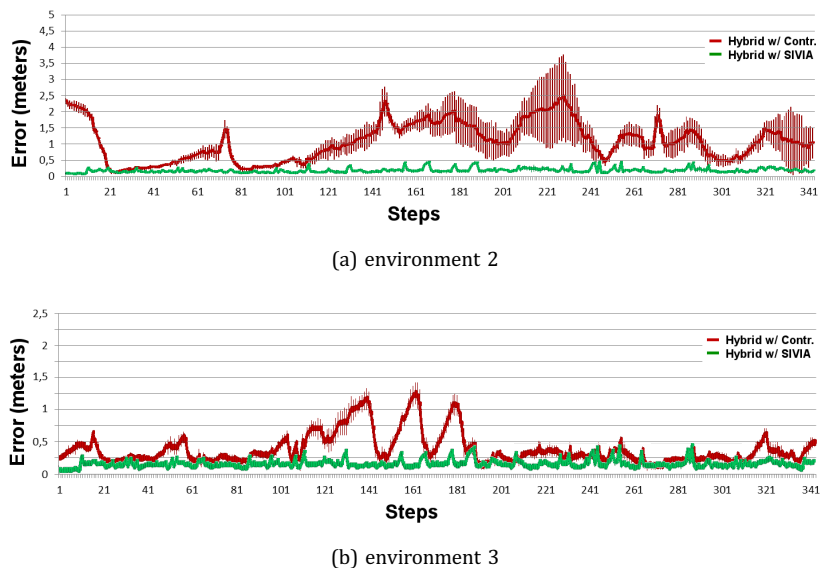


Fig. 8. Different scale (magnified) of the average particles error of the hybrid approaches for trajectory 1 in environments 2 and 3.

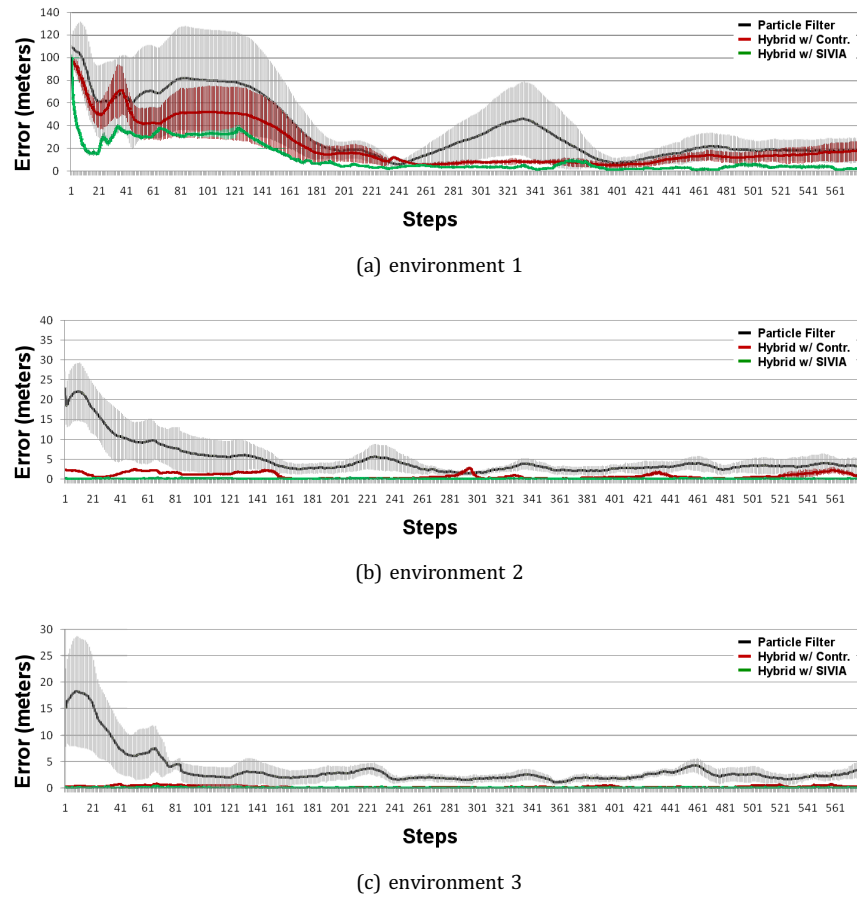


Fig. 9. Comparison of the average particles error for trajectory 2.

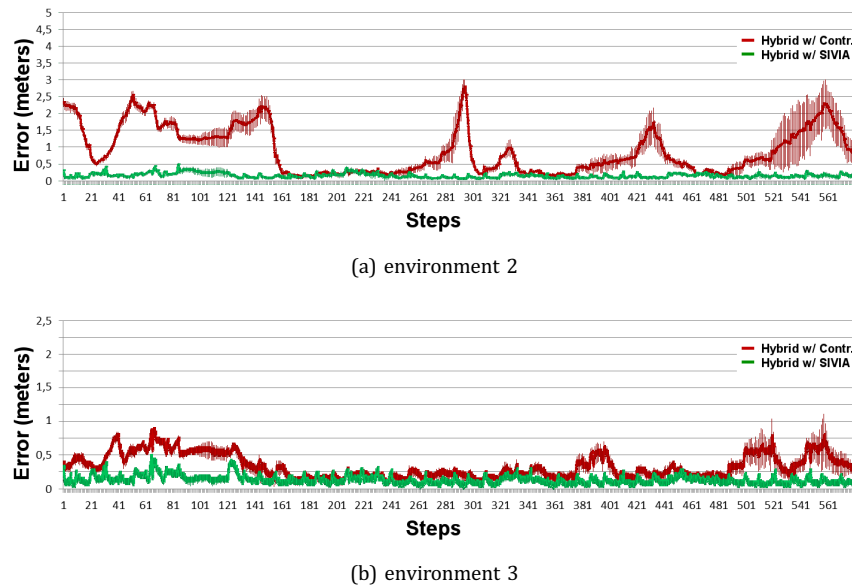


Fig. 10. Different scale (magnified) of the average particles error of the hybrid approaches for trajectory 2 in environments 2 and 3.

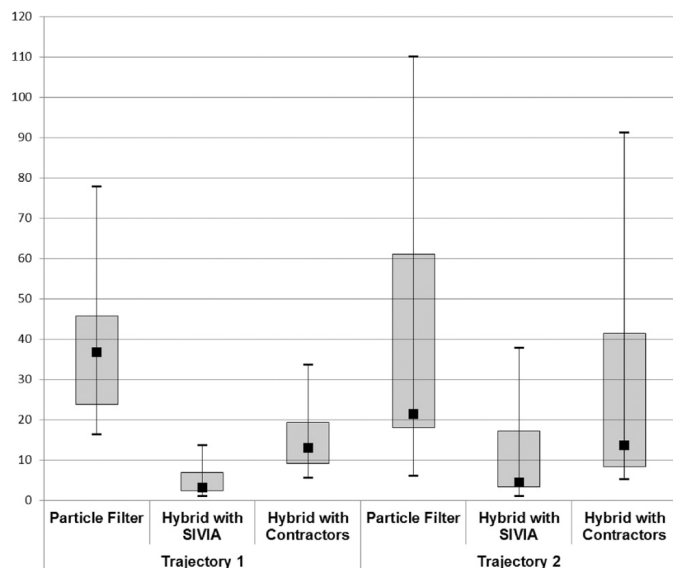


Fig. 11. Boxplot with the errors concentration of experiments in environment 1.

the whole environment, thus it has the largest standard deviation.

Figures 11–13 show boxplots, from descriptive statistics, that are used to compare the errors concentration. In boxplot, the error is represented in the vertical axis, and we can see the maximum and the minimum errors (black trace), the quartiles and the median (black square). In these plots, the outliers are discarded. Values 1.5 times higher or smaller than interquartile region (gray rectangle) are considered outliers.

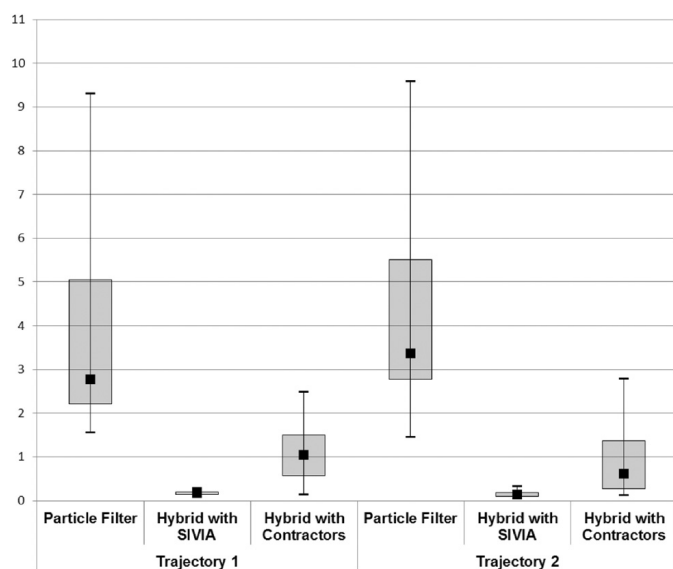


Fig. 12. Boxplot with the errors concentration of experiments in environment 2.

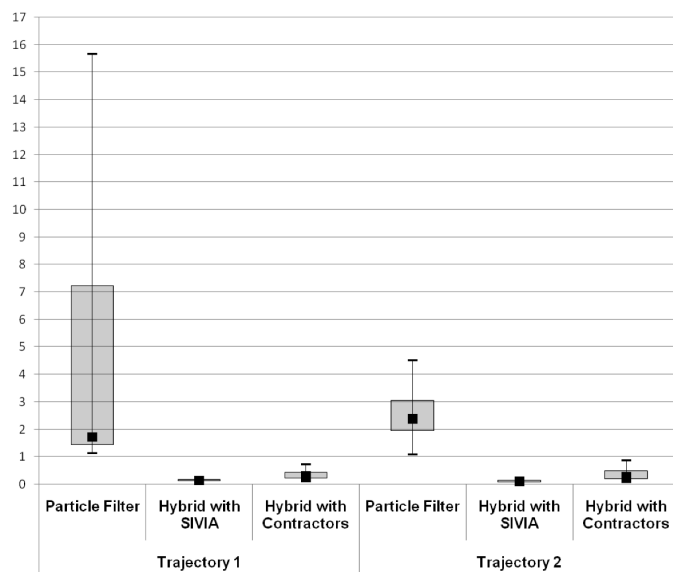


Fig. 13. Boxplot with the errors concentration of experiments in environment 3.

Figure 11 shows the concentration of the errors for the tests in the first environment. We can see the precision gain obtained by hybrid approaches, specially using SIVIA. Considering the maximum error, the particle filter, presents values of 77.7 and 110 m for trajectories 1 and 2, respectively. On the other hand, hybrid approaches have maximum errors of 13.5 and 37.8 m, when SIVIA is used, and 33.7 and 91.2 m, when contractors are used. With respect to the minimum errors, the particle filter shows 16.3 m in the first trajectory and 6 m in the second, while the method using SIVIA has values of 0.98 and 1.08 m, and the method using contractors, 5.6 and 5.3 m.

In the second environment, the errors from all methods notably drop, as shown in Fig. 12, but we still can see that hybrid methods have better results. The maximum errors with the traditional particle filter are 9.3 and 9.5 m, with the hybrid approach using SIVIA are 0.2 and 0.3 m and with the hybrid method using contractors are 2.4 and 2.7 m. Considering the minimum errors, the particle filter obtained 1.5 and 1.4 m, the method using SIVIA obtained 0.07 and 0.05 m, and the method using contractors obtained 0.13 and 0.12 m.

Finally, Fig. 13 shows the boxplot related to the experiments in the third environment. The particle filter obtained maximum errors of 15.6 and 4.51 m, the hybrid method using SIVIA obtained maximum errors of 0.24 m for both trajectories, and the hybrid method using contractors obtained maximum errors of 0.7 and 0.8 m. Regarding the minimum errors, the particle filter has values of 1.14 and 1.08 m, while the method using SIVIA has values of 0.05 and 0.04 m, and the method using contractors, 0.11 and 0.09 m.

Table 2. Average computational time per iteration (in seconds).

Env/Traj	Particle filter	Hybrid method using contractors	Hybrid method using SIVIA
1/1	$67 \times 10^{-4}$	$70 \times 10^{-4}$	0.2404
1/2	$67 \times 10^{-4}$	$70 \times 10^{-4}$	0.1664
2/1	$68 \times 10^{-4}$	$73 \times 10^{-4}$	0.0144
2/2	$68 \times 10^{-4}$	$72 \times 10^{-4}$	0.0142
3/1	$71 \times 10^{-4}$	$78 \times 10^{-4}$	0.0132
3/2	$71 \times 10^{-4}$	$77 \times 10^{-4}$	0.0134

As shown by the results of the experiments, both hybrid methods are more precise than the traditional particle filter for localization in the tested scenarios. The gain in precision is notorious in both cases, and becomes more apparent when there is more information available in environment. In addition to the precision gain, the standard deviation is minimized due to the constraints in the spread of particles.

If we analyze the results in the different environments, it is possible to perceive that the increase in the number of markers improves the results of the methods. Nevertheless, there is a limit to increase the number of markers and still improve the results. On the other hand, insufficient markers reduce the capacity of improvement, because the resulting intervals after applying the constraints may be too large.

Considering only the hybrid approaches, the method using SIVIA shows more precise results than the method using contractors. This means that, no matter how good a contractor is, its results will usually be worse than the ones obtained with SIVIA, due to the fact that SIVIA does not have this limitation of using a single box. However, we need to consider the computational cost associated with the use of interval analysis. Table 2 shows the average time per iteration (in seconds) of each tested method. As expected, both hybrid methods have computational cost higher than the particle filter. But, while the overhead of the method using contractors is small (in general, less than 10% if compared with particle filter results), the method using SIVIA has an overhead varying from 180% to 3500%. However, since the average time per iteration of the particle filter is small, all methods tend to generate results in less than 1 s. Thus, all methods are feasible to deal with the localization problem.

## 7. Conclusions

In this paper, we described a variation of the hybrid probabilistic-interval approach for localization, presented in [10]. By using the SIVIA technique, instead of contractors, we obtained improvements in terms of quality of localization, in expense of increasing the computational

time. Nevertheless, the final time per iteration measured in our experiments using SIVIA is still suitable for real applications.

Although susceptible to errors (associated with usage of the particle filter), both hybrid methods showed increases in accuracy of localization. In such hybridizations, the uncertainty is limited to an area represented by a box or set of boxes. These regions — mathematically guaranteed by interval rules — limit the action of the particle filter. Nonetheless, the estimate obtained by the particles is probabilistic, thus, it is still possible to occur the common errors associated to particle filters (see Sec. 4). But the great advantage is that, even in the worst case, the errors do not exceed the limits imposed by the interval analysis.

It should be noted that the methods presented here can be used in a broad range of applications without any substantial changes. This includes applications not only in underwater scenarios, but also in aerial or terrestrial environments. For instance, the detection of the markers can be done using acoustic waves, using visual characteristics, or even using odor sensors. There is the need to know the location of the markers during the whole process, however, it is not mandatory that the markers are stationary.

As future work, there are some modifications that could make the method more generic. One of the possible improvements is to use indistinguishable makers. Another opportunity is to use the method in real environments. Given the uncertainties existent in such environments, the treatment of outliers becomes necessary. The challenge, in this case, lies in the implementation of a good method to discard outliers without eliminating feasible solutions. Finally, a natural evolution is the extension of the hybrid approaches to the SLAM problem, in which, the robot must localize itself and create the map of the environment, at the same time, using the robot motion and observations.

## Acknowledgments

This work has been partially supported by Brazilian agencies CAPES and CNPq. Special thanks to Vitor Jorge and Jeremy Nicola for the support during our work.

## References

- [1] M. Dunbabin, P. Corke, I. Vasilescu and D. Rus, Data muling over underwater wireless sensor networks using an autonomous underwater vehicle, *Robotics and Automation, in Proc. 2006 IEEE Int. Conf. ICRA 2006*, May 2006, pp. 2091–2098.
- [2] S. Shatara, X. Tan, E. Mhemmo, N. Gingery and S. Henneberger, Experimental investigation on underwater acoustic ranging for small robotic fish, *Robotics and Automation, IEEE Int. Conf. ICRA 2008*, May 2008, pp. 712–717.



- [3] L. Jaulin, A nonlinear set membership approach for the localization and map building of underwater robots, *IEEE Trans. Robot.* **25** (2009) 88–98.
- [4] J. J. Leonard, A. A. Bennett, C. M. Smith and H. Feder, Autonomous underwater vehicle navigation, *IEEE ICRA Workshop on Navigation of Outdoor Autonomous Vehicles* (1998).
- [5] J. C. Kinsey, R. M. Eustice and L. L. Whitcomb, A survey of underwater vehicle navigation: Recent advances and new challenges, *IFAC Conf. Manoeuvring and Control of Marine Craft*, Lisbon, Portugal, September 2006. Invited paper.
- [6] L. Paull, S. Saeedi, M. Seto and H. Li, Auv navigation and localization: A review, *IEEE J. Ocean. Eng.* **39** (2014) 131–149.
- [7] S. Thrun, W. Burgard and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents series)* (MIT Press, Cambridge, MA, USA, 2005).
- [8] L. Jaulin, M. Kieffer, O. Didrit and E. Walter, *Applied Interval Analysis: With Examples in Parameter and State Estimation, Robust Control and Robotics* (Springer, 2001).
- [9] F. Dellaert, D. Fox, W. Burgard and S. Thrun, Monte carlo localization for mobile robots, in *Proc. 1999 IEEE Int. Conf. Robotics and Automation (ICRA)* (IEEE Press, Piscataway, NJ, USA, 1999).
- [10] R. Neuland, J. Nicola, R. Maffei, L. Jaulin, E. Prestes and M. Kolberg, Hybridization of monte carlo and set-membership methods for the global localization of underwater robots, *2014 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)* (2014).
- [11] L. Jaulin and E. Walter, Set inversion via interval analysis for nonlinear bounded-error estimation, *Automatica* **29**(4) (1993) 1053–1064.
- [12] P. Corke, C. Detweiler, M. Dunbabin, M. Hamilton, D. Rus and I. Vasilescu, Experiments with underwater robot localization and tracking, *2007 IEEE Int. Conf. Robotics and Automation*, April 2007, pp. 4556–4561.
- [13] D. Kim, D. Lee, H. Myung and H.-T. Choi, Artificial landmark-based underwater localization for auvs using weighted template matching, *Intel. Service Robot.* (2014) 1–10.
- [14] N.-Y. Ko, T. G. Kim and Y. S. Moon, Particle filter approach for localization of an underwater robot using time difference of arrival, *OCEANS, 2012 — Yeosu*, May 2012, pp. 1–7.
- [15] C. Forney, E. Manii, M. Farris, M. A. Moline, C. G. Lowe and C. M. Clark, Tracking of a tagged leopard shark Sensor calibration and state estimation, *2012 IEEE Int. Conf. Robotics and Automation (ICRA)*, May 2012, pp. 5315–5321.
- [16] S. Wang, L. Chen, H. Hu and D. Gu, Single beacon based localization of auvs using moving horizon estimation, *2013 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, November 2013, pp. 885–890.
- [17] F. Maurelli, Y. Petillot, A. Mallios, P. Ridao and S. Krupinski, Sonar-based auv localization using an improved particle filter approach, *OCEANS 2009 — EUROPE*, May 2009, pp. 1–9.
- [18] D. Meizel, O. Lévêque, L. Jaulin and E. Walter, Initial localization by set inversion, *IEEE Trans. Robot. Autom.* **18**(6) (2002) 966–971.
- [19] L. Jaulin, Robust set membership state estimation; application to underwater robotics, *Automatica* **45**(1) (2009) 202–206.
- [20] M. Langerwisch and B. Wagner, Guaranteed mobile robot tracking using robust interval constraint propagation, *Intelligent Robotics and Applications*, Vol. 7507, eds. C.-Y. Su, S. Rakheja and H. Liu (Springer, Berlin Heidelberg, 2012), pp. 354–365.
- [21] R. Guyonneau, S. Lagrange, L. Hardouin and P. Lucidarme, The kidnapping problem of mobile robots: A set membership approach, *7th National Conf. Control Architectures of Robots* (2012).
- [22] M. S. I. Seddik, L. Jaulin and J. Grimsdale, Phase based localization for underwater vehicles using interval analysis, *Math. Comput. Sci.* **8**(3–4) (2014) 495–502.
- [23] F. Abdallah, A. Gning and P. Bonnifait, Box particle filtering for nonlinear state estimation using interval analysis, *Automatica* **44**(3) (2008) 807–815.
- [24] L. Jaulin, Nonlinear bounded-error state estimation of continuous-time systems, *Automatica* **38**(6) (2002) 1079–1082.
- [25] I. Kueviakoe, A. Lambert and P. Tarroux, Comparison of interval constraint propagation algorithms for vehicle localization, *J. Softw. Eng. Appl.* **5** (2012) 157–162.
- [26] L. Jaulin, Range-only slam with occupancy maps: A set-membership approach, *IEEE Trans. Robotics* **27**(5) (2011) 1004–1010.
- [27] G. Echeverria, N. Lassabe, A. Degroote and S. Lemaignan, Modular open robots simulation engine: Morse, *ICRA* (IEEE, 2011), pp. 46–51.



**Renata Neuland** received her M.S. degree in Computer Science from the Federal University of Rio Grande do Sul (UFRGS), Brazil, in 2014. She is currently a Ph.D. student in Computer Science in UFRGS. Her research interests include underwater robotics and interval analysis.



**Luc Jaulin** received his Ph.D. degree in Automatic Control from the University of Orsay, France in 1993. He is currently a Professor of Robotics at the ENSIETA engineering school in Brest, France, since 2004. He does his research on underwater robotics using interval methods and constraint propagation. Furthermore, he has several publications in high-ranked journals and conferences in topics that include interval analysis, simultaneous localization and mapping, localization and underwater robotics.



**Renan Maffei** received his M.S. degree in Computer Science from the Federal University of Rio Grande do Sul (UFRGS), Brazil, in 2013. He is currently a Ph.D. student in Computer Science in UFRGS. His research interests include localization, SLAM and integrated exploration.



**Edson Prestes** received M.Sc. and Ph.D. in CS from the Federal University of Rio Grande do Sul (UFRGS), Brazil, in 1999 and 2003, respectively. He is a CNPq Productivity Fellow for his contribution to Brazilian Scientific Progress. In 2005, he became Professor at UFRGS and nowadays, he is the Head of the Theoretical Informatics Department at Informatics Institute (UFRGS). He has active participation in the community, as a member of IEEE, IEEE RAS and IEEE SA. He is currently co-chair of the IEEE RAS WG ORA and Advisory Board member

of IEEE RAS SIGHT. Among his recent achievements, he was co-chair of the special session entitled “Knowledge Representation and Ontologies for Robotics and Automation” at IROS 2012, co-chair of workshop on Smart Gadgets Meet Ubiquitous and Social Robots on the Web, held in ACM Ubicomp 2012, and guest editor of special issue “Ubiquitous Robotics” in Robotics and Autonomous Systems Journal/Elsevier (published late 2013). He also has been serving as member of program committee and reviewer boards in several conferences and journals.



**Mariana Kolberg** received the B.Sc. (2002), M.Sc. (2005) and Ph.D. (2009) degrees in Computer Science (CS) from the Pontifical Catholic University of Rio Grande do Sul (PUCRS), Brazil. She did half of her Ph.D. at Karlsruhe Institute of Technology (KIT), Germany. In 2011 she became Adjunct Professor at the Theoretical Informatics Department at the Federal University of Rio Grande do Sul (UFRGS). Since 2011 Mariana is focusing in applying interval analysis in robotics problems. Mariana is also a member of IEEE RAS SIGHT. She is a

regular reviewer for Applied Mathematics and Computation (Elsevier), TEMA and RBCA. She is also reviewer in conferences like CLCAR, ERAD, WSCAD e SBPO. In addition, she has been acting as *ad hoc* consultant to Research Support Foundation of Rio Grande do Sul (FAPERGS). Furthermore, she has several publications in high-ranked journals and conferences in topics that include interval analysis, simultaneous localization and mapping, global localization and reliable computing.