Acoustic-based techniques for autonomous underwater vehicle localization

Y Petillot[1*], **F Maurelli**[1], **N Valeyrie**[1], **A Mallios**[2], **P Ridao**[2], **J Aulinas**[2], and **J Salvi**[2]
[1]Ocean Systems Laboratory, Herriot–Watt University, Edinburgh, UK
[2]Office 016 - PIV, Escola Politecnica Superior, Campus de Montilivi, Girona, Spain

The manuscript was received on 1 December 2009 and was accepted after revision for publication on 9 June 2010.

DOI: 10.1243/14750902JEME197

Abstract: This paper presents two acoustic-based techniques to solve the localization problem for an autonomous underwater vehicle (AUV). After a brief description of the Bayesian filtering framework, the paper analyses two of the most common underwater scenarios. Each scenario corresponds to one of the two localization techniques. The scenarios are as follows: map-based localization, when the environment is sufficiently distinctive; transponder-based localization for navigation in large and not distinctive environments. An environment is said to be distinctive when there are possible reference points. For example, a completely flat featureless sea floor is not considered as distinctive. The proposed techniques were validated in simulation. The map-based localization technique was also validated in trials with the autonomous underwater vehicles *Ictineu* and *Nessie IV*. The trials with *Ictineu* took place in an abandoned marina with *Ictineu* travelling along a trajectory that was more than 600 m long. The trials with *Nessie IV* took place in a rectangular pool and during an AUV competition.

Keywords: localization problem, autonomous underwater vehicle, acoustic-based techniques

1 INTRODUCTION

Over the past few years, underwater technology has significantly grown and remotely operated vehicles (ROVs) are nowadays safely and routinely used in the offshore industry. ROVs are constantly controlled by a human operator. Other vehicles called autonomous underwater vehicles (AUVs) have received much attention from the underwater robotic community because of their ability to carry out missions without close human intervention and supervision. There are many open research areas related to AUVs such as safe and reliable autonomous navigation. An autonomous vehicle indeed needs to demonstrate high autonomy in order to keep track of its position and orientation and to build a map of this environment while successfully completing autonomous tasks. Many approaches have been developed for land and aerial vehicles [**1–3**]. However, the constraints and peculiarities of the underwater environment prevent the direct application of techniques developed for land or aerial vehicles in the underwater domain. A careful study of each technique with regard to the constraints of the underwater domain is required.

The primary navigation system used in land, aerial, and underwater applications is the inertial measurement unit (IMU), which provides an estimation of the acceleration of the vehicle. In addition to being expensive, this system suffers from drift errors, which, doubly integrated over time, result in the need for some external system to correct the localization errors. In land and aerial robotics, this can be achieved by incorporating Global Positioning System (GPS) measurements. The GPS signal is not available in an underwater environment and other techniques to correct the drift errors of the IMU are needed. The IMU is also not able to give an absolute localization but only an estimation of the inertial movement. There are, then, many cases in which the IMU alone fails to provide accurate localization. This happens when the starting position is unknown or in the

*Corresponding author: Ocean Systems Laboratory, Herriot–Watt University, Earl Mountbatten Building, Riccarton, Edinburgh EH14 4AS, UK.
email: Y.R.Petillot@hw.ac.uk

classical *kidnapped robot* problem [**4**] where the robot starts from an unknown position in a known environment and needs to determine its position. Providing only relative values, the IMU is also not able to recover after an incorrect localization. Similar considerations are valid for the Doppler velocity log (DVL), which provides an estimation of the velocity vector using acoustic measurements.

The absence of a GPS signal and the difficult sensing conditions make underwater localization a complex problem. The lack of visibility and scattering make the use of cameras difficult, if not impossible, at long ranges. Precise laser range finders, widely exploited for land robots, cannot be used efficiently in the water. As a consequence, acoustic techniques are the most widely used and have specific problems such as high noise, poor resolution, and slow frame rate.

This paper investigates solutions to two problems.

1. *AUV localization*. In this case, the environment is known *a priori* and a local map of the environment is captured by an imaging sonar. The algorithm uses the local sonar map to localize itself in the known global map. This is a typical problem for deep-water inspection of structures or port and harbour surveillance where the environment is well known but the vehicle's position is not accurate by the time that it has reached the area of interest for the mission.
2. *Beacon-based simultaneous localization and mapping* (SLAM). In this case, the environment is unknown and feature poor. Classical feature-based mapping and localization techniques cannot be used. In the present approach, the environment is composed of acoustic transponders whose positions are *a priori* unknown. It has been assumed that the transponders broadcast only range information which corresponds to the current state of the art in readily available transponders technology. The algorithm presented here estimates jointly the position of the vehicle and the position of the beacons. This can be seen as a self-calibrating long-baseline (LBL) system.

It is, then, possible to imagine a system built on the two solutions presented here which solves the full localization problem; in open water, the vehicle deploys beacons as it moves and uses the network generated to localize until it reaches a structure that requires closer inspection for which some *a priori* information is known (a local map, for instance). The particle filter localization algorithm is used for this task.

1.1 Related work

Several techniques have been proposed for AUV localization. Caiti *et al.* [**5**] developed an acoustic localization technique using freely floating acoustic buoys equipped with GPS connection. This system requires the buoys to emit a ping at regular time intervals with coded information of its GPS position. The vehicle can locate itself using time-of-flight measurements of acoustic pings from each buoy. The limitations of this approach are the necessity to deploy a sufficient number of floating buoys in the mission area, the need to collect them after the end of the mission, and a non-efficient communication scheme, as the buoys periodically send acoustic messages. Limitations for deep-water missions are also evident. Erol *et al.* [**6**] proposed to use GPS-aided localization. The problem in this approach is that the GPS signal does not propagate in the water. The AUV is therefore forced to acquire the signal at the surface. This approach is not very reliable, since the vehicle has no access to a GPS signal during the submerged period and has to estimate its state with other sensors. To handle this problem, the vehicle dives to a fixed depth and follows a predefined trajectory, which is not suitable for navigation in complex environments.

Other acoustic-based localization techniques include the LBL, short-baseline (SBL), and ultra-short-basline (USBL) systems. In these cases, the vehicle's position is determined on the basis of the acoustic returns detected by a set of receivers. For LBL systems, a set of acoustic transponders is first deployed in the area of operation. The vehicle is then able to locate itself with respect to the transponders (or, in the opposite way, the transponders are able to track the vehicle), as explained in reference [**7**]. For SBL and USBL systems, Storkensen *et al.* [**8**] proposed to use a support ship equipped with a high-frequency directional emitter able to determine the AUV's position accurately with respect to the mother ship. This approach requires, however, the support of a ship. Furthermore, it cannot be used in many situations as it requires the ship and the vehicle to be close to each other. This approach is therefore not suitable for navigation around deep offshore structures. In terms of the computational framework, particle filter techniques have been used for the localization of underwater vehicles although not as widely as for land and aerial vehicles. Karlsson *et al.* [**9**] proposed a particle filter approach for AUV navigation but the focus was more on the mapping part than on the localization. Silver *et al.* [**10**] presented a particle filter merged with scan-matching techniques. Recent work by Ortiz *et al.* [**11**] addressed the use of particle filters for tracking undersea narrow telecommunication cables.

Regarding the SLAM problem, the earliest method put forward uses an extended Kalman filter (EKF) to estimate jointly the state of the AUV and the position of the transponders [**12**]. This approach has been used with success in various cases [**13–15**], but the filter suffers from linearization of non-linear models which can quickly lead to divergence as the covariance estimates become unreliable. In addition, the algorithmic complexity of the EKF algorithm grows quadratically with the number of beacons (in the two-dimensional (2D) case). For example, the prediction step of the EKF not only affects the state of the AUV but also involves the computation of the Jacobian of the motion model which also takes into account the transponders. Several methods have been put forward to reduce the complexity of the EKF algorithm, such as the linear state augmentation principle or the partitioned update approach, both presented in reference [**16**]. These methods have a linear complexity with the number of beacons. Other solutions to the SLAM problem are the extended information filter [**17**], the unscented Kalman filter (UKF) [**18**], and the FastSLAM algorithms [**19**].

1.2 Contribution

The present approach to the localization problem uses the ability of the vehicle to sense the environment and requires a general, but not too precise, *a priori* map of the environment. Most algorithms proposed to date (in air or an land) perform very well in small environments but do not always scale well to large environments and long trajectories. The present approach extends this technique to the underwater domain and demonstrates the robustness of a map-based localization system in a real large-scale scenario. Moreover, most of the approaches for localization assume a perfect knowledge of the environment. This assumption is unrealistic because the ground truth is not available in many situations. Here, however, the map generated [**20**] with a SLAM approach is used. This map does integrate errors and the demonstration that it can still be used validates the use of SLAM-generated maps for other applications. For instance, once the robot has built a coherent map on an unknown environment, it is indeed desirable to use the generated map for further activities in the same portion of environment. The computational complexity of localization decreases and it is easier for the vehicle to perform other tasks in the area.

While the first approach is very useful to navigate around underwater structures, the environment surrounding the robot might not be sufficiently distinctive to allow reliable localization (e.g. large areas with a flat featureless seabed). Acoustic transponders deployed in the area of interest help to localize the vehicle. The position of the transponders is usually unknown, in which case the vehicle has to localize itself and the transponders simultaneously. A range-only SLAM algorithm inspired by reference [**21**] is proposed; this is based on a particle-filtering implementation of the SLAM problem and is coupled with a mixture-of-Gaussians representation of the posterior distribution of the beacon's position. This is demonstrated in the simulation and shows the potential of the technique as a self-calibrating LBL system.

The paper is divided into four parts: the first part presents the Bayesian filtering techniques for robot localization. The next two parts address the two common scenarios in underwater localization: in *closed* and *open* environments, with the use of particle filter and Gaussian sum filter techniques. The fourth part presents the experimental results, both in the simulation and in real conditions. Finally, conclusions and future work are detailed.

2 THE BAYESIAN FILTERING FRAMEWORK

2.1 Optimal Bayesian filtering

The state of the AUV at time t is a random vector denoted by $\boldsymbol{X}_t$ that usually includes the position and the speed of the AUV. The evolution of the AUV's state is provided by the motion model. Usually, the motion model assumes that $(\boldsymbol{X}_k)$ is a Markov process entirely defined by the transition kernel $p(x_k|x_{k-1})$ and the distribution of the initial state $\boldsymbol{X}_0$. The transition kernel is related to the measurements provided by the IMU or the DVL that predict the future state of the AUV, given its past state. The state $\boldsymbol{X}_t$ is not directly observed but is known through measurements regularly gathered by the AUV. The measurements are, for example, the scan of a rotating sonar or range of measurements with respect to acoustic transponders. The measurement at time t is a random vector denoted by $\boldsymbol{Y}_t$ that only depends on $\boldsymbol{X}_t$. The measurement model provides the conditional density $p(y_t|x_t)$ that statistically links the observation $\boldsymbol{Y}_t$ to the state $\boldsymbol{X}_t$. These assumptions are those of a hidden Markov model.

The objective in Bayesian filtering is to calculate the conditional density $p(x_t|y_{0:t})$ so that an estimate of the state $\boldsymbol{X}_t$ can be obtained through the conditional expectation

$$\mathbb{E}[\boldsymbol{X}_t|\boldsymbol{Y}_{0:t}] = \int x_t p(x_t|y_{0:t})\,\mathrm{d}x_t \tag{1}$$

This conditional expectation is the estimate of $\boldsymbol{X}_t$, given $\boldsymbol{Y}_{0:t}$, which minimizes the mean square error, namely

$$\mathbb{E}\left[|\boldsymbol{X}_t - \mathbb{E}[\boldsymbol{X}_t|\boldsymbol{Y}_{0:t}]|^2\right] \leqslant \mathbb{E}\left[|\boldsymbol{X}_t - \psi(\boldsymbol{Y}_{0:t})|^2\right] \tag{2}$$

for any other estimate $\psi(\boldsymbol{Y}_{0:t})$ of $\boldsymbol{X}_t$, given the observations $\boldsymbol{Y}_{0:t}$. The conditional density $p(x_t|y_{0:t})$ may be recursively calculated from $p(x_{t-1}|y_{0:t-1})$ through the exact prediction and correction steps

$$p(x_t|y_{0:t-1}) = \int p(x_{t-1}|y_{0:t-1}) \underbrace{p(x_t|x_{t-1})}_{\text{transition kernel}} \mathrm{d}x_{t-1} \tag{3}$$

and

$$p(x_t|y_{0:t}) = c_t p(x_t|y_{0:t-1}) \quad \underbrace{p(y_t|x_t)}_{\text{measurement model}} \tag{4}$$

The normalization constant c_t is unknown and given by

$$\begin{aligned}\frac{1}{c_t} &= \int p(x_t|y_{0:t})\,\mathrm{d}x_t \\ &= \int p(x_t|y_{0:t-1})p(y_t|x_t)\,\mathrm{d}x_t\end{aligned} \tag{5}$$

Assuming that the motion and the measurement models are both linear and Gaussian, the distribution of $\boldsymbol{X}_t|\boldsymbol{Y}_{0:t}$ is also Gaussian. The mean and the covariance matrix of $\boldsymbol{X}_t|\boldsymbol{Y}_{0:t}$ may then be calculated using the recurrence relations of the Kalman filter [**22**, **23**]. The calculation of the conditional expectation $\mathbb{E}[\boldsymbol{X}_t|\boldsymbol{Y}_{0:t}]$ is exact so that the Kalman filter is optimal; i.e. it provides the estimate of $\boldsymbol{X}_t$, given $\boldsymbol{Y}_{0:t}$, which minimizes the mean square error mentioned previously.

The motion and the measurement models of an AUV are seldom both linear and Gaussian. Consequently, approximate and thereby suboptimal solutions to the non-linear non-Gaussian Bayesian filtering problem have to be provided. Various approximation schemes have been put forward, among which are the Monte Carlo and the sum of Gaussian approximations. The former approximation is at the core of any particle filter and the latter is known as the Gaussian sum filter.

2.2 Particle filters

Any particle filter relies on a Monte Carlo approximation of the conditional density $p(x_t|y_{0:t})$ using a finite set of N points ξ_t^i in the state space called *particles*. The approximation is of the form

$$p(x_t|y_{0:t}) \approx \sum_{i=1}^{N} w_t^i \delta_{\xi_t^i}(x_t)$$

where

$$\sum_{i=1}^{N} w_t^i = 1 \tag{6}$$

and where $\delta_{\xi_t^i}$ denotes the usual Dirac function, namely

$$\delta_{\xi_t^i}(x) = \begin{cases} 1 & \text{if} \quad x = \xi_t^i \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Equation (6) may be interpreted in the following way: the denser the particles in a region of the state space and the higher their weights, the higher is the probability that the state lies in this region. Assuming that the way in which samples can be obtained from $p(x_t|y_{0:t})$ is known, the Monte Carlo approximation becomes

$$p(x_t|y_{0:t}) \approx \sum_{i=1}^{N} \frac{1}{N} \delta_{\xi_t^i}(x_t)$$

with

$$\xi_t^i \sim p(x_t|y_{0:t}) \tag{8}$$

where $\sim$ means 'follows the distribution of'. The previous assumption is unrealistic, since it implies that the Bayesian filtering problem is solved. Any particle filter relies on the importance sampling principle which provides a mechanism to build a Monte Carlo approximation of $p(x_t|y_{0:t})$.

Suppose that it is required to sample from a distribution whose density f is of the form

$$f(x) = cr(x)g(x)$$

where

g = density of a distribution from which it is easy to sample and which is called the proposal distribution
r = weighting function which is easy to evaluate
$c = \int f(x)\,\mathrm{d}x$ is an unknown normalization constant

The importance sampling principle provides the approximation for f as

$$f(x) \approx \sum_{i=1}^{N} w_i \delta_{\xi_i}(x)$$

with

$$\xi_t^i \sim g(x) \tag{9}$$

where the weights $w_i = f(\xi_i) \Big/ \sum_{j=1}^{N} f(\xi_j)$ are not equal to $1/N$, since they account for the particles being generated using a distribution other than f.

The first particle filter ever proposed was the *bootstrap filter* introduced by Gordon *et al.* [**24**] but the most commonly used particle filter is the *sampling with importance resampling* (SIR) filter. Other kinds of particle filter such as the auxiliary particle filter or the kernel filter have been introduced in order to improve the performance of the SIR algorithm. The interested reader can find an overview of the existing sequential Monte Carlo methods for Bayesian filtering in references [**25**] and [**26**].

The SIR filter intends to reproduce the optimal prediction and correction steps of the optimal Bayesian filter. The three steps of iteration $t \geqslant 1$ of the SIR algorithm are as follows.

Step 1: selection: Generate $\tau_t^i \sim (w_{t-1}^1, \ldots, w_{t-1}^N)$.

Step 2: propagation: Generate $\xi_t^i \sim p\left(x_t | \xi_{t-1}^{\tau_t^i}\right)$.

Step 3: correction: Set $w_t^i \propto p(y_t | \xi_t^i)$.

In the SIR particle filter, the propagation step of iteration t uses the transition kernel to simulate the new set of particles $(\xi_t^1, \ldots, \xi_t^N)$, as in the bootstrap filter. However, only the most likely particles at time $t-1$ are selected to generate the particles at time t. The selection is made according to the weights, since the higher a weight the more adequate are the observation of the corresponding particle. This selection step makes the SIR particle filter more efficient than the bootstrap filter.

2.3 Gaussian sum filter

Particle filters rely on a Monte Carlo approximation of the distribution of $\boldsymbol{X}_t | \boldsymbol{Y}_{0:t}$. They require a large number of particles so that the approximation is sufficiently accurate and thereby have a high computational cost. A long time before the emergence of particle filters, an approximation scheme based on a mixture of Gaussian distributions was suggested by Alspach and Sorenson [**27**]. This Gaussian sum approximation was shown to perform better than the EKF while being compatible with the computational capabilities of that time. The conditional density $p(x_t|y_{0:t})$ is approximated by

$$p(x_t | y_{0:t}) \approx \sum_{i=1}^{N} \alpha_t^i \Gamma(x_t;\, m_t^i, \mathbf{P}_t^i)$$

with

$$\sum_{i=1}^{N} \alpha_t^i = 1 \tag{10}$$

where $x \mapsto \Gamma(x;\, \mathbf{m}, \mathbf{P})$ denotes the probability density function (PDF) of a Gaussian multi-variate distribution of mean m and covariance matrix $\mathbf{P}$. The individual means m_t^i and individual covariance matrices and $\mathbf{P}_t^i$ are updated according to the recurrence relations of the EKF. The interested reader can find a complete description of the Gaussian sum filter in reference [**27**].

2.4 Bayesian approach to the SLAM problem

The Bayesian filtering framework is suitable for estimating the state $\boldsymbol{X}_t$ of an AUV. The SLAM problem also requires the estimation of the state $\boldsymbol{X}_t$ together with the estimation of the map of the environment. In the SLAM problem considered in this paper, the environment is composed of acoustic transponders whose positions are to be found. The position of each transponder is denoted by B_i and B denotes the set of all the transponders.

The Bayesian approach to the SLAM problem requires the calculation of the conditional density $p(x_t, b|y_{0:t})$ so that an estimate of $(\boldsymbol{X}_t, B)$ can be obtained through the conditional expectation

$$\mathbb{E}[\boldsymbol{X}_t, B | \boldsymbol{Y}_{0:t}] = \int (x_t, b) p(x_t, b | y_{0:t}) \, \mathrm{d}(x_t, b) \tag{11}$$

3 PARTICLE FILTER FOR LOCALIZATION IN A DISTINCTIVE ENVIRONMENT

The first scenario is when the vehicle can sense a sufficiently distinctive environment. By distinctive is meant that the sensor's measures vary significantly with position. A typical example is the navigation in man-made environments, such as marinas, or navigation close to underwater structures, both natural and artificial, such as an off-shore underwater oil infrastructure.

The chosen approach uses particle filter techniques for a variety of reasons. First, it can handle the estimation of non-Gaussian and non-linear processes. This is very important because non-linearities are very frequent in AUVs, both in the motion model specification and in the observation process. Additionally, the noise cannot be modelled as Gaussian in many situations. Another advantage of using particle filters is that it does not require any assumption on the initial position and orientation of the vehicle. However, the standard approach presents two main problems. The first is the large number of particles required in order to explore the state space, resulting in an increase in the computational power needed. The other major issue in particle filter approaches is the *sample impoverishment problem*, i.e. the loss of diversity for the particles to represent the solution space adequately [**28**]. In the present approach, both these problems are addressed, as shown next.

3.1 Particle resampling

The standard SIR algorithm for particle resampling lets the particles with high weights reproduce, while the particle with low weights are more unlikely to survive. However, the resulting PDF at time t depends only on the PDF at time $t-1$. In time, this means that only a small part of the state space is represented by the particles and the system cannot recover from an incorrect estimation of the vehicle's position (owing to sensor noise, for instance). In the present system, at each step, a portion of the particles is instantiated randomly in the state space. Thus, the resampling algorithm is built with two modules. The first is a standard SIR module, returning $N-k$ particles. The second module returns k particles, created randomly. The combination of these two modules constitutes the resampling step in the proposed system. The algorithm is then able to recover in the case of a wrong convergence, as shown in section 3.4. The benefits from the computational point of view are also relevant, since there is no need to instantiate a large number of particles. Even if in the initial step there are no particles near the real position of the vehicle, the proposed solution is still able to find the correct position, after some time, because of the partial random resampling.

3.2 Sensor model and the likelihood calculation

The sensor used here is a forward-looking sonar profiler which returns range profiles (Fig. 1). The general idea to compute the likelihood is to compare two arrays of distances. The first is the array generated by the real sonar system, while the second is simulated on the basis of the possible position of the vehicle (given by the particle) and knowledge of the environment (*a priori* map). For the simulated set-up, a ray-tracing algorithm was used to compute the intersection between the sonar beams and the environment. In order to determine the array of distances for each particle, an alternative solution to ray tracing has been used. As the map given by Ribas *et al.* [**20**] is a set of lines, a geometrical approach based on line intersection is much faster than and as accurate as ray tracing. For the real tests, sonar data processing is required in order to obtain the array of distances. The imaging sonar returns for each beam an intensity array. To transform this array to a distance value, a threshold is applied in order to separate the acoustic imprint left

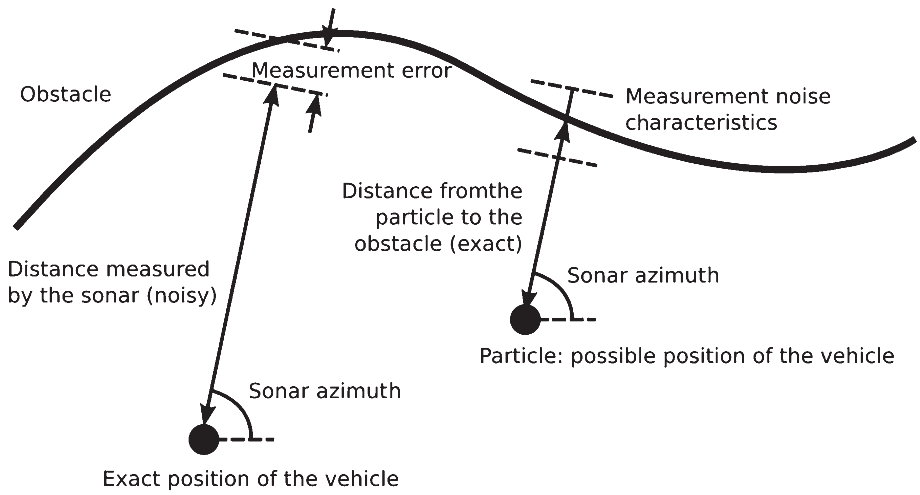


Fig. 1 Illustration of the calculation of the likelihood for the particle filter

by an object in the image from the noisy background data. Once the two arrays of distances have been computed, the next problem is the likelihood calculation. The likelihood value should reflect the similarity of the two arrays. To provide better understanding of the proposed approach, the case of arrays composed by a single element each (r and s) will now be described. In this case, the likelihood is given by the equation

$$L(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}x^2\right) \tag{12}$$

where $x = r - s$, r represents the real value of distance given by the vehicle, and s represents the simulated value given by the particles. In the case of arrays with more than one value, the same procedure can be applied to all indices of the array, producing a likelihood array. In order to calculate a single likelihood value, a mean-value solution was chosen. The likelihood function calculated in this way is therefore a mixture of Gaussians. A pure Gaussian likelihood was also tested, multiplying the likelihoods of the single indices, like a joint probability of independent variables. Both methods proved to be valid and reliable. However, the product method is more selective but also more sensitive to noise, as a few bad-index likelihoods have a great impact on the overall likelihood of the particle. On the other hand, the average method seems to preserve better the diversity of the particles in the solution state. Thus, it is to be preferred when few particles are used, while the product method is to be preferred when there are many particles in the same area, to discriminate better between them.

3.3 Motion model

For the simulated set-up, the motion model is simply the difference between the ground truth positions at time t and at time $t - 1$, disturbed with some process noise. The particle state is thus updated considering that value, plus a different noise for each particle, in order to explore more effectively the solution space (Fig. 2). For the real set-up, the motion estimation is given by the sensors. The *Ictineu* vehicle is equipped with a SonTek Argonaut DVL unit which provides bottom tracking and water velocity measurements at a frequency of 1.5 Hz. Additionally, an MTi sensor, a low-cost motion reference unit (MRU), provides attitude data at a 0.1 Hz rate. These values are integrated in an EKF. A six-degrees-of-freedom constant-velocity kinematic model is used to predict the state of the vehicle. Since AUVs are commonly operated describing rectilinear transects at a constant speed during survey missions, such a model, although simple, represents a realistic way to describe the motion.

3.4 Results

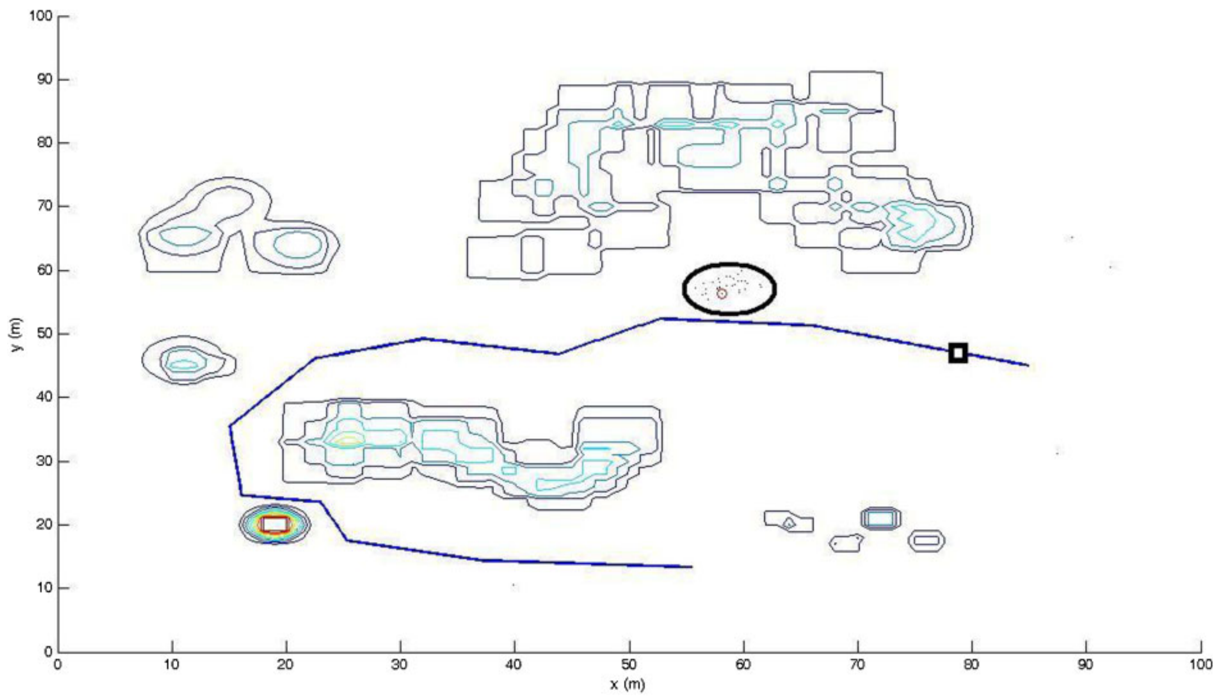
3.4.1 Results on simulated data

The first step in the validation of the proposed system is by simulation. The present system can model a vehicle with six degrees of freedom. In this particular set-up it has been assumed that the pitch and roll of the vehicle are neglected. Additionally, at this point, the sensor's orientation in relation to the vehicle is fixed. A simulated gyroscope is used to have a noisy estimation of the orientation of the vehicle's heading (*yaw*). A simulated depth sensor provides a noisy estimation of the vehicle's depth. Finally, a simulated sonar is modelled to acquire range profiles. It is assumed that an *a priori* map of the vehicle's surroundings is known. No assumptions are made on the initial position of the vehicle within the map. The particle state is represented by six variables (three for the orientation and three for the position of the vehicle), plus an additional variable representing the weight of the particle.

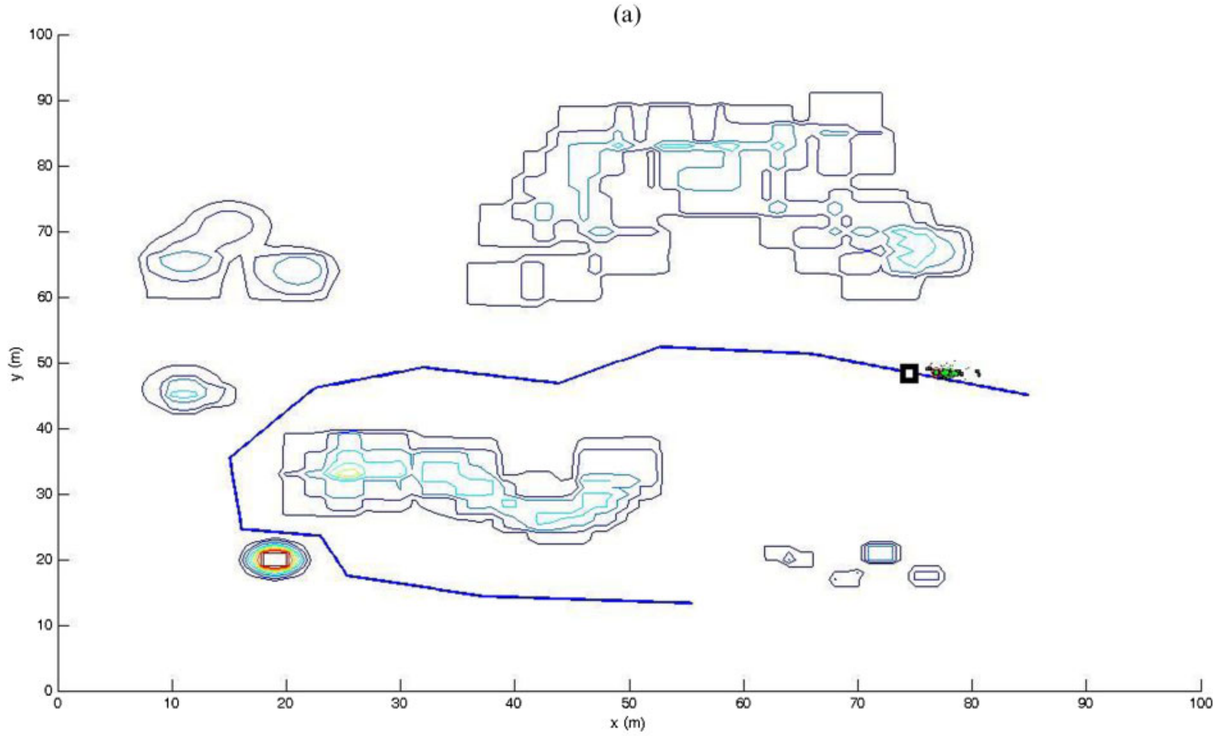
A synthetic environment was created to validate the approach. Different types of scenario were considered in order to analyse the algorithm performances. The system has proved to be valid for both structured and unstructured scenarios. In more than 95 per cent of cases, the algorithm converges to the real trajectory before the end of the experiment. Two methods have been explored in order to infer the trajectory, given the particles' state. The first method considers the mean of the Monte Carlo approximation (i.e. the weighted sum of the particles), and the second method considers the best particle as an estimate of the state of the AUV (a crude estimate of the mode of the distribution). In the simulated set-up, the best particle trajectory always gives better results, minimizing both the time needed for convergence and the overall error.

3.4.2 Results on real data

The localization system was tested on the same data set used in reference [**20**] to perform underwater SLAM. The data were gathered during an extensive survey of an abandoned marina on the Costa Brava (Spain). The *Ictineu* AUV gathered a data set along a 600 m trajectory which included a small loop around the principal water tank and a 200 m straight path through an outgoing canal. The data set included

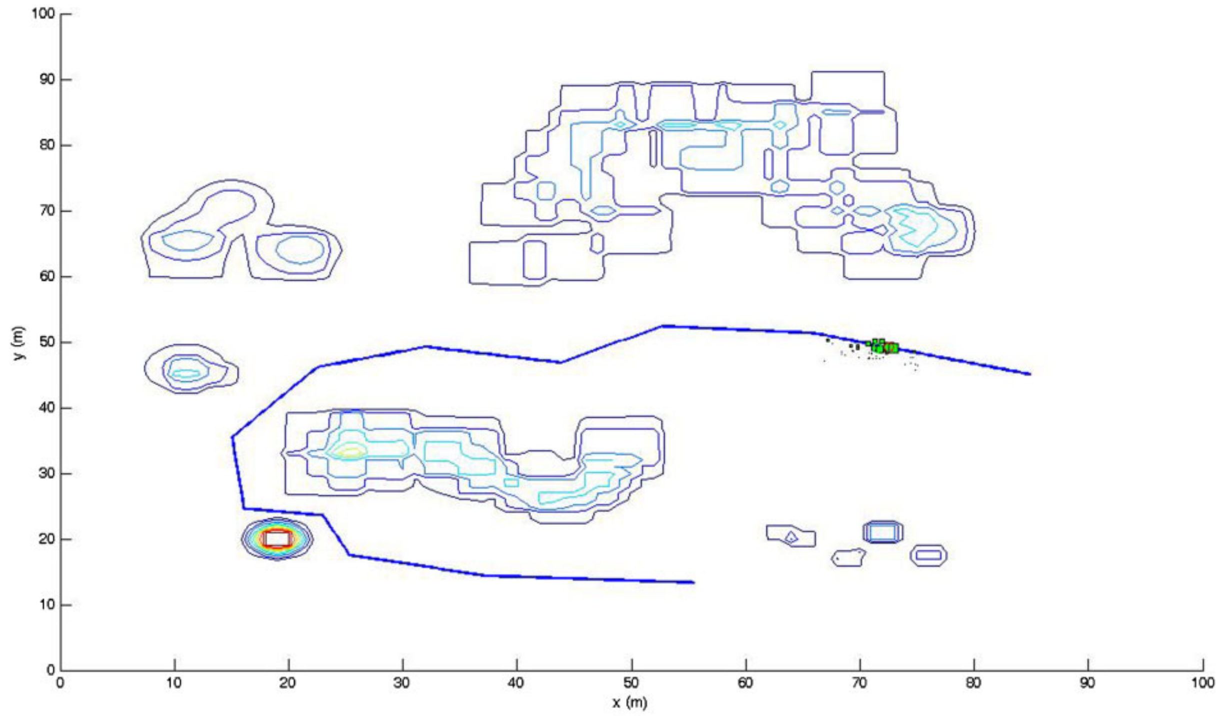


(a)



(b)

Fig. 2 Three consecutive states of a mission (2D projection of a three-dimensional simulation). This tests shows the ability to recover after a wrong state estimation. The real trajectory is a solid black curve, where the rectangle on top of the line represents the actual position of the AUV at that time. (a) Wrong particle convergence; 90 per cent of the particles are in the circle, quite far from the real AUV position. (b) Recovering from the wrong convergence; the particles are now close to the real position (with increased likelihood). (c) The actual AUV state has been correctly estimated



(c)

Fig. 2 (continued)

measurements from the imaging sonar (a Tritech Miniking), DVL, and MRU sensors. For validation purposes, the vehicle was operated close to the surface attached to a buoy equipped with a Differential Global Positioning System (DGPS) and used for registering the real trajectory (ground truth). Figures 3 and 4 show the results of the localization algorithm, in two different settings. In Fig. 3, 100 particles are used and they are spread over an area of 1 848 m^2. In Fig. 4, the initial area where the particles are spread increases to 10 368 m^2 and the number of particles is consequently increased to 600.

As can be seen, the dead-reckoning trajectory obtained by merging DVL and MRU data suffers from an appreciable drift (even causing it to go outside the caral). On the other hand, during all the mission, the trajectory computed by the localization system is very close to the trajectory given by the DGPS. Figure 5 shows a zoom in a specific area of the marina (part of the large trapezoid). The particles have different shapes according to their weights. It is clear that the error given by the dead-reckoning trajectory is continuously increasing, while the performance of the trajectory given by the localization is very good during all the mission.

The localization system has also been integrated into the navigation system of the *Nessie IV* robot of the Ocean Systems Laboratory, the winning entry in the Student Autonomous Underwater Competition – Europe (SAUC-E) [**29**]. The sensor measures were provided by a Tritech Micron sonar, scanning continuously. The motion estimation was provided by an optical flow detection algorithm, using a camera looking downwards, as the use of DVL was forbidden. Figures 6(a) and 7(a) show the raw sonar images, Figs 6(b) and 7(b) the segmented image, and Figs 6(c) and 7(c) the vehicle state estimation in the environment, a rectangular tank. In spite of the high interference noise in the sonar image in Fig. 7(a) (due to another active sonar operating nearby), the state of the AUV is accurately estimated, which shows the robustness of the algorithm.

4 RANGE-ONLY SLAM USING THE GAUSSIAN SUM FILTER

In the case where the environment is unknown or not sufficiently distinctive to be recognized, SLAM-based solutions need to be explored. The solution to the range-only SLAM problem presented in this paper is based on a Rao–Blackwellized particle filter. It is thereby an extension of the two FastSLAM algorithms presented in reference [**19**]. The underlying idea is to take advantage of the statistical dependences between the observations, the state of the AUV, and the

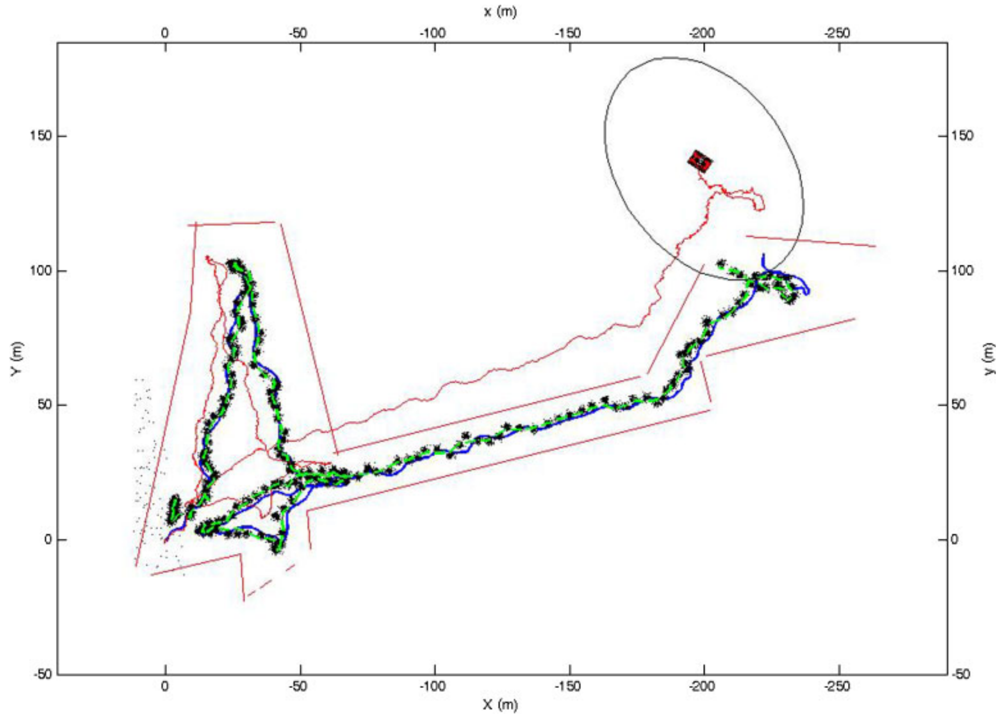


Fig. 3 2D plot of the environment, with the particles, plotted for all the time stamps, the DGPS trajectory (blue), the dead-reckoning trajectory (red), the uncertainty ellipse from the dead reckoning, and the trajectory inferred by the particles (green). 100 particles are spread over an area of 1 848 m^2

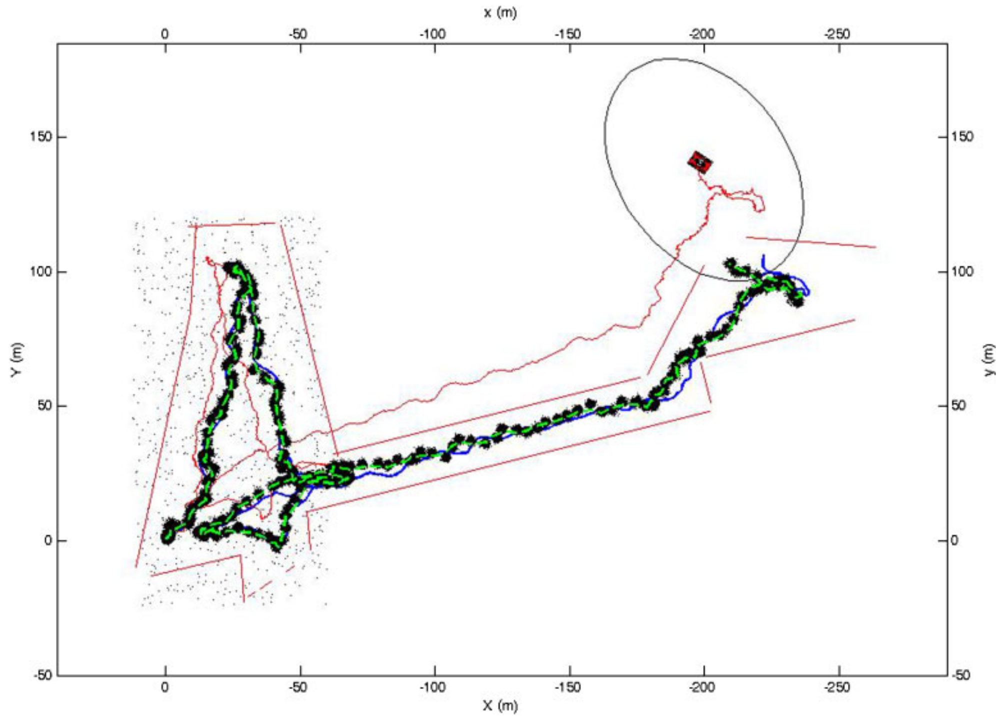


Fig. 4 2D plot of the environment, with the particles, plotted for all the time stamps, the DGPS trajectory (blue), the dead-reckoning trajectory (red), the uncertainty ellipse from the dead reckoning, and the trajectory inferred by the particles (green). 600 particles are spread over an area of 10 368 m^2

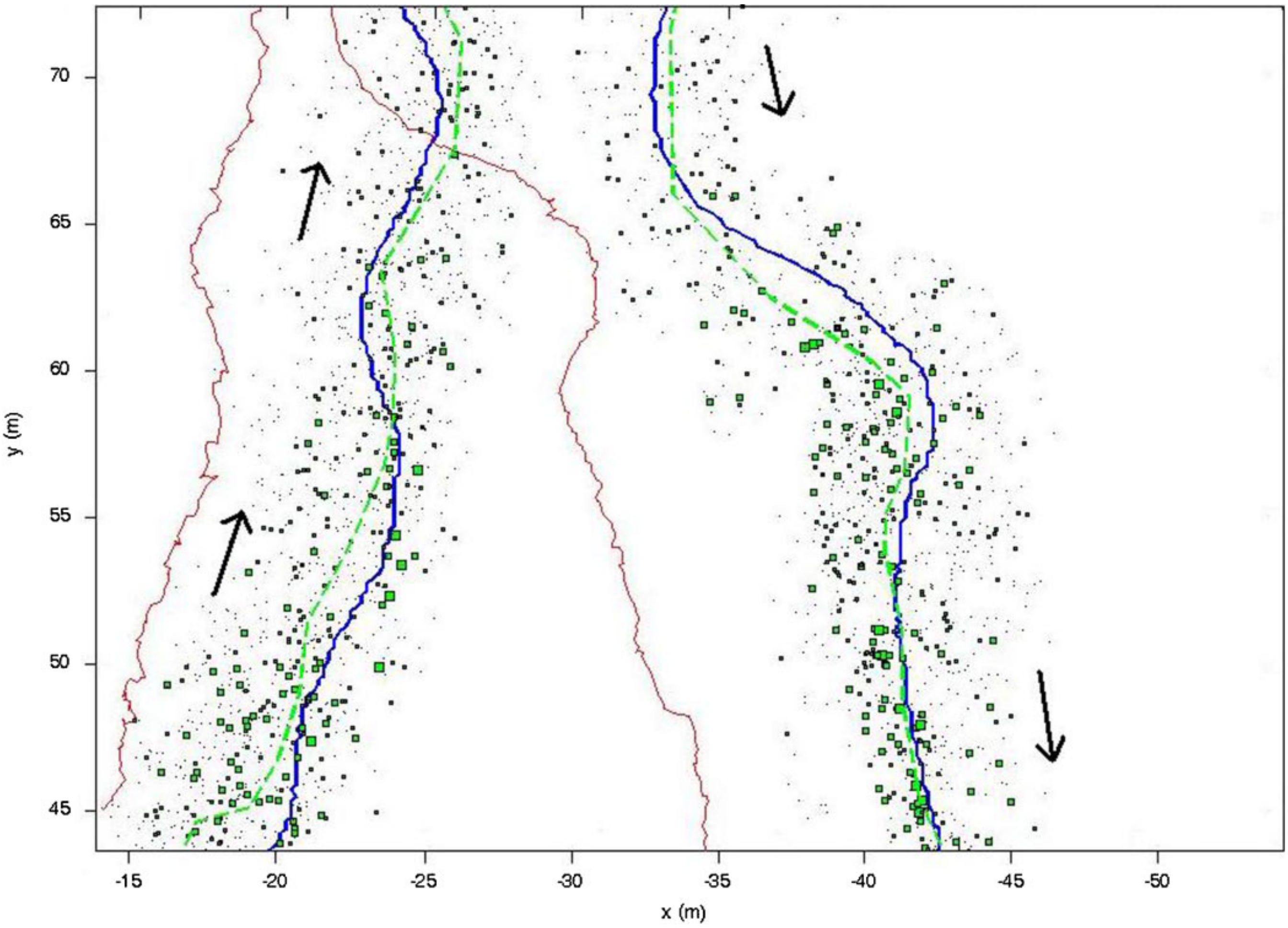


Fig. 5 Zoom of the area to show how close the inferred trajectory (green) is to the real trajectory (blue) in comparison with the dead reckoning (red)

transponders. The Rao–Blackwellized particle filter takes advantage of the fact that the transponders are mutually independent, given the trajectory and the observations, according to

$$p(b|x_{0:t}, y_{0:t}) = \prod_i p(b_i|x_{0:t}, y_{0:t})$$

The conditional density of $(\boldsymbol{X}_{0:t}, B)$, given $\boldsymbol{Y}_{0:t}$, may consequently be written as

$$\begin{aligned} p(x_{0:t}, b|y_{0:t}) &= p(x_{0:t}|y_{0:t})p(b|x_{0:t}, y_{0:t}) \\ &= p(x_{0:t}|y_{0:t}) \prod_i p(b_i|x_{0:t}, y_{0:t}) \end{aligned}$$

The objective is to approximate the conditional distribution of $(\boldsymbol{X}_{0:k}, B)$, given $\boldsymbol{Y}_{0:k}$. The trajectory of the AUV is approximated using a set of particles given by

$$p(x_{0:t}|y_{0:t}) \approx \sum_{i=1}^{N} w_t^i \delta_{\xi_{0:t}^i} \tag{13}$$

In the FastSLAM algorithms, the conditional distribution of $B_j|(\boldsymbol{X}_{0:t}, \boldsymbol{Y}_{0:t})$ is assumed to be Gaussian. It is updated using the classical EKF recurrence relations once a transponder is observed. In this paper, the distribution of $B_j|(\boldsymbol{X}_{0:t}, \boldsymbol{Y}_{0:t})$ is approximated using a sum of Gaussian distribution, which was first suggested in reference [**21**], since a sum of Gaussian distributions provides a better approximation of the inverse sensor model than a single Gaussian does. The recursive calculation of the means and the covariance matrices is made according to the recurrence relations of the Gaussian sum filter.

A Rao–Blackwellized particle filter is usually much more efficient than the EKF and the UKF algorithms [**17**]. It is also more flexible regarding the number of beacons, since the distributions of the beacons, given the trajectory and the observations, are independent. A newly observed beacon is simply added to the map of each particle. If a beacon has not been observed yet, the sum of Gaussian distributions is initialized. If

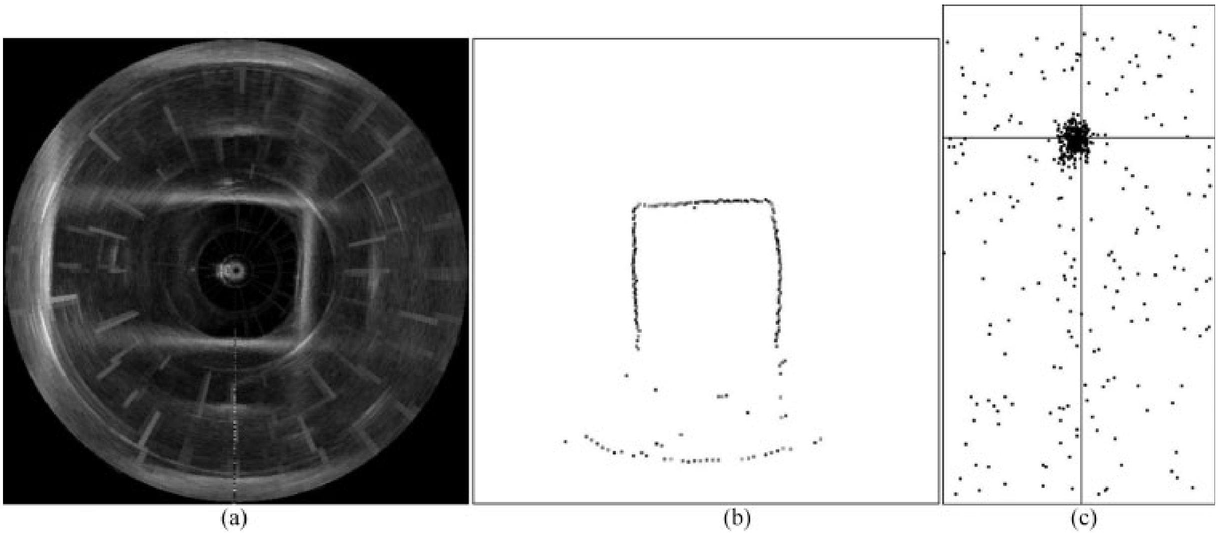


Fig. 6 (a) Raw sonar image, with range 10 m; (b) segmented image; (c) vehicle state estimation in the environment, 6 m × 11 m. Note that the raw sonar image is in the sonar reference frame, with the sonar mounted with the head looking down and with a rotation of 90°, while the segmented image is already transformed in the vehicle reference frame. The crossing lines in the image in (c) show the particle with greater weight. The real position of the vehicle is not known, as there is no ground-truth sensor available underwater. However, it can be inferred by looking at the raw sonar image

a beacon has already been observed, the sum of Gaussian distributions is updated.

4.1 Results

The proposed solution to the range-only SLAM problem has been tested in the simulation only. The scenario is that of an AUV navigating in a 2D environment and regularly collecting range measurements from a set of transponders. The AUV performs a trajectory typically used for survey missions. The trajectory is made of long straight lines and short sharp turns (Fig. 8). The AUV's state includes only its position and orientation. The vehicle collects dead-

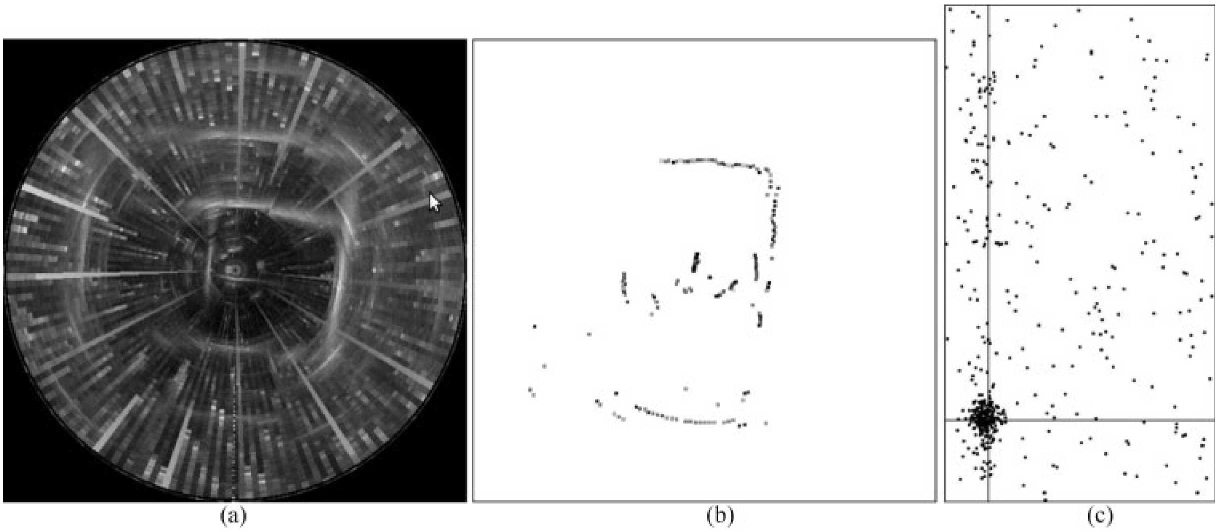


Fig. 7 (a) Raw sonar image, with range 20 m and with high interference noise; (b) segmented image; (c) vehicle state estimation in the environment, 6 m × 11 m. Similar considerations to those made for Fig. 6 are valid here. Although it might be hard to understand from the raw sonar image, the pool is a rectangle, with one long side starting near the centre, on the left, slightly down. The other is above it. In addition to the noise, the image presents multi-path reflections, which should not be confused with the pool borders

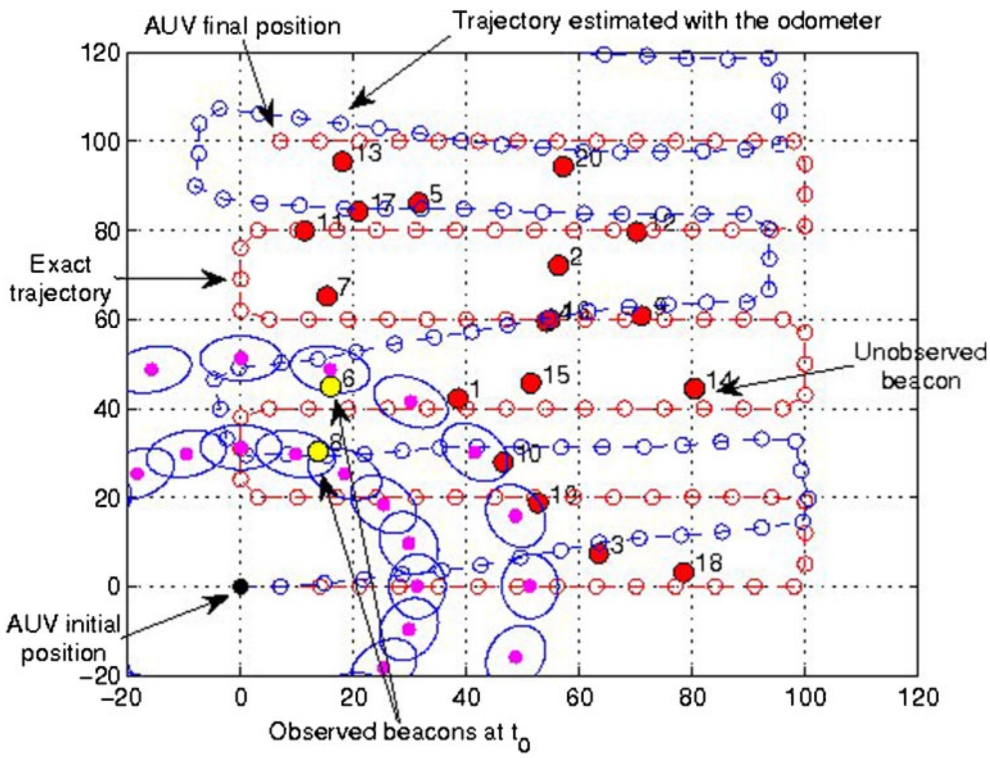


Fig. 8 Initialization of the sum of Gaussian distributions for the first two observed beacons. The red and yellow full circles are the beacons deployed around the AUV. The yellow full circles correspond to the beacons that have been observed and the red full circles to those that have not. The small pink full circles and the blue ellipses correspond to the means and the covariance matrices of the mixture of Gaussians respectively. Two mixtures are shown in the figure, one for each beacon that has been observed

reckoning measurements from its internal sensors and has thereby an estimate of the distance travelled and of the change in the orientation. In the scenario, the AUV receives potentially more than one range measurement depending on the range to the transponders. The further away from the AUV a transponder is, the less likely is the AUV to receive a reliable measurement. The number of observed transponders influences the behaviour of the particle filter. It was observed that, if at some point the AUV observes only one transponder, the particle filter is likely to diverge as it cannot track the vehicle orientation reliably with only one landmark. The observation at time t is, indeed, used to select among the set $\{\xi_t^1, \ldots, \xi_t^N\}$ the particles that are adequately most represented by the observation. If this observation consists of only one range measurement, the particles that lie on the circle centred on the corresponding transponder are those with a high weight. If the observation consists of more than one range measurement, the particles that are at the intersection of a set of circles are assigned a high weight. The position of the transponders is estimated using a mixture of Gaussian distributions. The mixture is initialized using the first measurement so that a sufficiently good approximation of the measurement model is attained. The details of the initialization procedure have been given in reference [**21**]. When a transponder is observed more than once, the mixture is updated according to the Gaussian sum filter, which changes the means and the covariance matrices of each component (Fig. 9).

5 CONCLUSION AND FUTURE WORK

This paper has presented two solutions to the critical problem of underwater navigation and localization. One is designed to work in structured environments where some of the structures are known. The other is designed to work in featureless environments where man-made beacons whose location might not be known have to be relied on. The first problem is solved using a particle filter approach for localization of AUVs. The theoretical approach has first been presented, followed by results of the simulation and of a real large-scale data set. The experimental results show the high performances of this algorithm, which is robust to noisy measurements and suitable for long

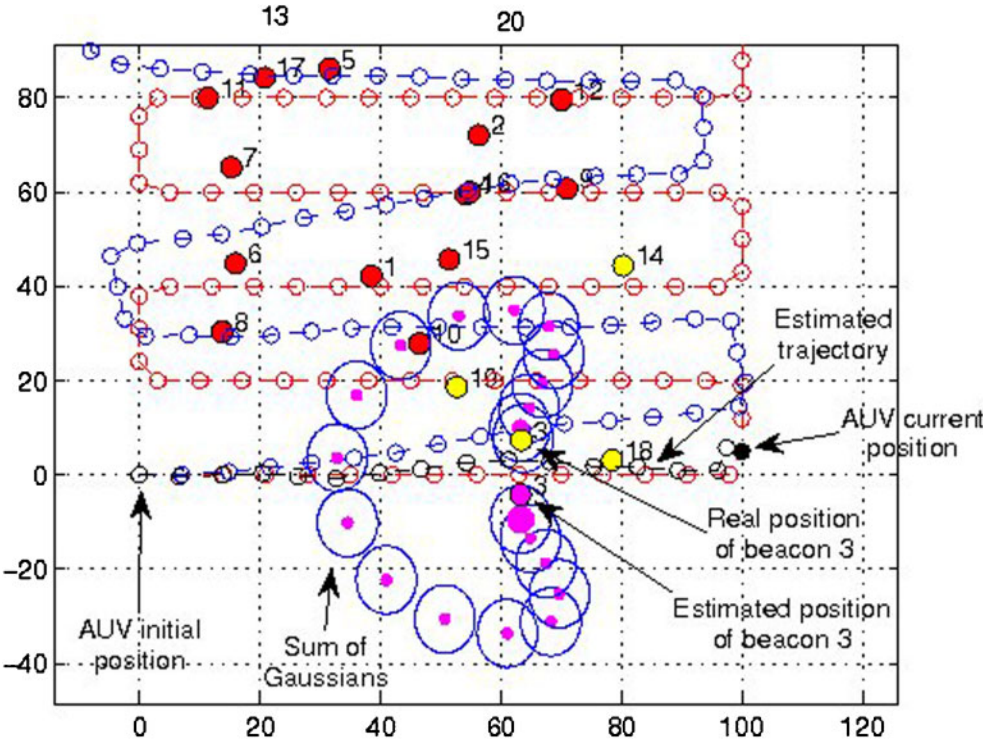


Fig. 9 This is very similar to Fig. 8. The main difference is that beacon 3 has already been observed and the mixture of Gaussians has already been updated. The pink full circles, corresponding to the mean of the mixture, have different sizes. The larger the dot, the higher is the weight attached to the Gaussian

trajectories. Additionally, it was demonstrated that the algorithm can deal with uncertainty in the *a priori* map by using the map produced by the SLAM algorithm proposed in reference [**20**], instead of the ground truth. The second problem is approached by an algorithm that solves the range-only SLAM problem. The algorithm is based on the FastSLAM algorithm, which is also particle filter based. The localization of the transponders is enhanced with respect to the original FastSLAM algorithms as the distribution of the individual transponders is approximated by a sum of Gaussians. The convergence of the particle filter is highly sensitive to the number of received range measurements. The higher this number is, the more likely it is that the filter will converge.

Future work will focus on active localization. Currently, the localization algorithm does not take control of the vehicle's trajectory. In the future, the algorithm will move the vehicle in order to localize the vehicle better. This could be achieved by clustering the particles and actively deciding the best motion to reduce the entropy in the particle's distribution, in order to minimize the time needed for convergence.

REFERENCES

1 **Borenstein, J., Everett, H. R., Feng, L.,** and **Wehe, D.** Mobile robot positioning: sensors and techniques. *J. Robotic Systems*, 1997, **14**, 231–249.

2 **Mouaddib, E. M.** and **Marhic, B.** Geometrical matching for mobile robot localization. *IEEE Trans. Robotics Automn*, 2000, **16**(5), 542–552.

3 **Sutherland, K. T.** and **Thompson, W. B.** Localizing in unstructured environments: dealing with the errors. *IEEE Trans. Robotics Automn*, 1994, **10**(6), 740–754.

4 **Filliat, D.** Map-based navigation in mobile robots: I. A review of localization strategies. *Cognitive Systems Res.*, 2003, **4**(4), 243–282.

5 **Caiti, A., Garulli, A., Livide, F.,** and **Prattichizzo, D.** Localization of autonomous underwater vehicles by floating acoustic buoys: a set-membership approach. *IEEE J. Oceanic Engng*, 2005, **30**(1), 140–152.

6 **Erol, M., Vieira, L. F. M.,** and **Gerla, M.** AUV-aided localization for underwater sensor networks. In Proceedings of the International Conference on *Wireless algorithms, systems and applications* (*WASA 2007*), Chicago, Illinois, USA, 1–3 August 2007, pp. 44–54 (IEEE, New York).

7 **Collin, L., Azou, S., Yao, K.,** and **Burel, G.** On spatial uncertainty in a surface long base-line positioning system. In Proceedings of the Fifth European Con-

ference on *Underwater acoustics* (*ECUA 2000*), Lyon, France, 10–13 July 2000, pp. 607–612 (École Supérieure de Chimie Physique Electronique de Lyon, Lyon).

8 **Storkensen, N., Kristensen, J., Indreeide, A., Seim, J.,** and **Glancy, T.** Hugin UUV for seabed surveying. *Sea Technol.*, February 1998.

9 **Karlsson, R., Gusfafsson, F.,** and **Karlsson, T.** Particle filtering and Cramer–Rao lower bound for underwater navigation. In Proceedings of the IEEE International Conference on *Acoustics, speech and signal processing* (*ICASSP '03*), April 2003, vol. 6, pp. 65–68 (IEEE, New York).

10 **Silver, D., Bradley, D.,** and **Thayer, S.** Scan matching for flooded subterranean voids. In Proceedings of the IEEE Conference on *Robotics, automation and mechatronics*, December 2004, vol. 1, pp. 422–427 (IEEE, New York).

11 **Ortiz, A., Antich, J.,** and **Oliver, G.** A Bayesian approach for tracking undersea narrow telecommunication cables. In Proceedings of Oceans 2009 – Europe, Bremen, Germany, 11–14 May 2009, pp. 1–10 (IEEE, New York).

12 **Smith, R.** and **Cheeseman, P.** On the representation and estimation of spatial uncertainty. *Int. J. Robotics Res.*, 1986, **5**(4), 56–68.

13 **Kantor, G.** and **Singh, S.** Preliminary results in range-only localization and mapping. In Proceedings of the IEEE International Conference on *Robotics and automation*, (*ICRA '02*). vol. 2, pp. 1818–1823.

14 **Kurth, D., Kantor, G.,** and **Singh, S.** Experimental results in range-only localization with radio. In Proceedings of the 2003 IEEE/RJS International Conference on *Intelligent robots and systems* (*IROS 2003*), vol. 1, pp. 974–979.

15 **Djugash, J., Singh, S.,** and **Corke, P. I.** Further results with localization and mapping using range from radio. In International Conference on *Field and service robotics* (*FSR '05*), July 2005.

16 **Bailey, T.** and **Durrant-Whyte, H.** Simultaneous localization and mapping (SLAM): part II. *Robotics & Automation Magazine*, 2006, **13**(3), 108–117.

17 **Thrun, S., Burgard, W.,** and **Fox, D.** *Probabilistic robotics,* 2005 (MIT Press, Cambridge, London).

18 **Julier, S. J.** and **Uhlmann, J. K.** A new extension of the Kalman filter to nonlinear systems. In Proceedings of the International Symposium on *Aerospace/defence sensing, simululation and controls*, Orlando, Florida, 1997, pp. 182–193.

19 **Montemerlo, M., Thrun, S.,** and **Whittaker, W.** Conditional particle filters for simultaneous mobile robot localization and people-tracking. In *Robotics and automation 2002*, Proceedings of ICRA '02, the IEEE International Conference, 2002, vol. 1, pp. 695–701.

20 **Ribas, D., Ridao, P., Tardós, J. D.,** and **Neira, J.** Underwater slam in man-made structured environments. *J. Field Robotics*, 2008, **25**(11–12), 898–921.

21 **Blanco, J.-L., Fernandez-Madrigal, J.-A.,** and **Gonzalez, J.** Efficient probabilistic range-only SLAM. In Proceedings of the IEEE/RJS International Conference on *Intelligent robots and systems*, (*IROS 2008*), September 2008, pp. 1017–1022 (IEEE, New York).

22 **Kalman, R. E.** A new approach to linear filtering and prediction problems. *Trans. ASME, J. Basic Engng*, 1960, **82**, 35–45.

23 **Kalman, R. E.** and **Bucy, R. S.** New results in linear filtering and prediction theory. *Trans. ASME, J. Basic Engng*, 1961, **83**, 95–107.

24 **Gordon, N., Salmond, D.,** and **Ewing, C.** A novel approach to nonlinear/non-Gaussian Bayesian estimation. *Proc. IEE Part F, Radar Signal Processing*, 1993, **40**, 107–113.

25 **Cappe, O., Godsill, S. J.,** and **Moulines, E.** An overview of existing methods and recent advances in sequential Monte Carlo. *Proc. IEEE*, 2007, **95**(5), 899–924.

26 **Doucet, A., Godsill, S.,** and **Andrieu, C.** On sequential Monte Carlo sampling methods for Bayesian filtering. *Statist. Comput.*, 2000, **10**(3), 197–208.

27 **Alspach, D.** and **Sorenson, H.** Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Trans. Autom. Control*, 1972, **17**(4), 439–448.

28 **Carpenter, J., Clifford, P.,** and **Fearnhead, P.** Improved particle filter for nonlinear problems. *Proc. IEE, Part F, Radar, Sonar Navig.*, 1999, **146**(1), 2–7.

29 **Maurelli, F., Cartwright, J., Johnson, N., Bossant, G., Garmier, P. L., Regis, P., Sawas, J.,** and **Petillot, Y.** *Nessie II* autonomous underwater vehicle. In Proceedings of the Tenth Unmanned Underwater Vehicles Showcase 2009 (*UUVS 2009)*, Southampton, UK, 22–23 September 2009.