Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-16 (2015) 095–100

Trajectory Generation for Bathymetry based AUV Navigation and Localization

Rodrigo P. França * **Aurélio T. Salton** *,* **Rafael D. S. Castro** * **Bruno N. Green** * **Damian Marelli** **

* *Faculty of Engineering, Pontifical Catholic University of Rio Grande do Sul (PUCRS), Brasil.*
** *School of Electrical Engineering and Computer Science, The University of Newcastle, Australia*

Abstract: This paper presents a path planning algorithm to improve the localization estimate of an autonomous underwater vehicle (AUV) based on bathymetry maps without the aid of external landmarks. A particle filter is used in order to fuse the data from an Inertial Measurement Unit (IMU), a downward pointing sonar and an *a priori* given bathymetry map. Since this method's performance is dependent on the rugosity of the map, a path planning algorithm is proposed in order to avoid such regions and optimize the particle filter performance. By guiding the vehicle to navigate in regions where the filter performance is acceptable, an efficient landmark-free localization algorithm is devised.

Keywords: Autonomous underwater vehicle (AUV), localization, particle filter, trajectory generation, navigation.

1. INTRODUCTION

The growing use of unmanned robotic vehicles is justified through various advantages associated with this technology in aerial, ground or under water environments. Inspection of areas with difficult access, oceanographic research and military use are some of examples of where such vehicles are being used. The significant growth of research on the topic of Autonomous Underwater Vehicles (AUVs) is a natural consequence of the fact that the ocean occupies over 70% of Earth's surface (Hyakudome, 2011).

Recent developments in embedded electronics have allowed a wide range of autonomous vehicles to be designed and manufactured, providing the research community with challenging tasks. One of the most important is associated with the localization of AUVs, since it is not possible to use Global Navigation Satellite Systems (GNSS) underwater. A classical example of underwater vehicle localization comes from the work by Carreras et al. (2003), which presents a method for estimating the position and orientation of the AUV using a camera attached to vehicle. Facing down, the camera seeks mosaics previously placed at the bottom of a tank from which the vehicle position is then determined. This method has proved efficiency in controlled environments, where vision-based sensors' limitations, such as reduced range of cameras in underwater environment, the lack of lighting and turbidity, do not pose serious threats to the localization system (Paull et al., 2014).

Other techniques based on acoustic sensors like Long Base Line (LBL), Short Base Line (SBL), Ultra-Short Base Line (USBL) and GPS Intelligent Buoys (GIB) came as alternatives to avoid the use of underwater cameras. The position of the vehicle in these systems is given by the acoustic return detected by a set of receivers (Caiti et al., 2005). In LBL, SBL and USBL a perimeter is formed with the transponders in the area where the AUV will navigate. The difference between these methods lies in the distances between the transponders and the distance from the transponders to the vehicle (Wolbrecht et al., 2014). For the LBL strategy, for example, the transponders are placed on the seafloor, while for the SBL and USBL the transponders are located in a ship that follow the vehicle. The GIB system uses buoys situated on the surface of the water equipped with DGPS: each buoy emits a ping with its GPS position, from these signals the vehicle is able to compute its own location (Alcocer et al., 2006).

The accuracy of the acoustic sensors are linked to factors such as the choice of the place of attachment of transponders – e.g., fixed on the submarine body, on a support ship or buoy at the water surface – and the depth where the submarine is working (Kinsey et al., 2006). Another important factor to be considered is related to the cost of the peripherals necessary in these localization methods. In this respect it is desirable to have an AUV able to navigate without the aid of a surface ship following its position, and without the necessity of previously fixed transponders and buoys. The above facts have motivated the use of map based techniques, that is, techniques that do not need any peripherals while running the localization algorithm. Of these, the most common form of determining the position and orientation of underwater vehicles is through the use of inertial sensors. However, the use of inertial sensors alone results in an "open-loop" form of estimation, since this technique estimates the location of the vehicle by integrating measurements given by accelerometers and gyroscopes. Naturally, this approach suffers from drift errors generated

⋆ : corresponding author: aurelio.salton@pucrs.br

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2015.10.264

by the integration of small biases over time (Stutters et al., 2008).

It is clear that the use of only one specific sensor does not guarantee the robustness and accuracy of a submerged localization system. Rather, in practical terms it can be said that data fusion from multiple sensors is mandatory for a good estimation. In order to implement data fusion two main classes of nonlinear filters are at the disposal of practitioners: parametric or non-parametric filters. These techniques use probability theory in order to estimate the states of a system, with parametric filters designating those filters that parameterize probability functions using, for example, their mean (μ) and variance (σ). In this classification the most popular filters are part of the Gaussian family, whose most widely used and accepted technique is the famous Kalman Filter (KF) (Kalman, 1960) and its variations. Non parametric filters use numerical approaches to describe the probability function and are particularly suited for nonlinear system estimation since their probability function evolves to better fit the data. In particular, the most known non parametric method is based on Monte Carlo simulations and usually referred to as Particle Filter (PF) (Gordon et al., 1993).

As presented by Maurelli et al. (2008), it is possible to develop a PF based localization algorithm that only fuses an IMU with one or more sonars, provided some form of map of the environment is presented. In fact, methods denominated Terrain Based Localization (TBL) are particularly suitable for AUV localization. These methods use bathymetry maps (a map that gives depth information) as a reference and a PF is used to fuse the inertial sensor with the information coming from the sonar. The advantage of this approach is that no external equipment such as landmarks or transponders are necessary (Nakatani et al., 2009). However, this approach may lead to the filter convergence to incorrect estimates, specially in cases where the terrain does not provide sufficient information for the localization algorithm.

Given this scenario, this paper aims at the development of a low cost localization method for an AUV with no external peripherals. By making use of a Particle Filter in order to fuse the data from the vehicle IMU with one single downward pointing sonar and the terrain map, an efficient localization algorithm is sought. Furthermore, this paper proposes a simple path planing algorithm in order to avoid navigation over flat terrains that do not provide sufficient information to the PF. This work is structured as follows: in Section 2 the problem definition is presented, the particle filter approach and the path planing method are described in Sections 3 and 4, respectively. Section 5 shows simulation results that illustrate the efficacy of the proposed filter, and Section 6 present a brief conclusion.

2. PROBLEM DEFINITION

This paper considers the problem of determining the location of an AUV in a given depth. Given the assumption that the vehicle depth is constant and available online, the AUV of interest may be described by its Cartesian coordinates (x, y) and by its orientation, defined as the angle θ between its heading direction and the abscissa. These three variables determine the so called pose of the vehicle, which is depicted in Fig. 1. At instant k, the pose is denoted by

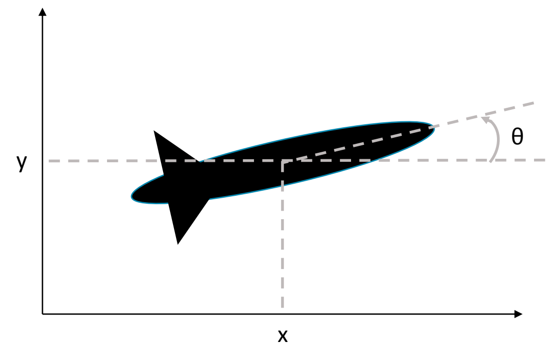

Fig. 1. Top view of a planar AUV depicting its pose (1).

$$p_k := \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} . \tag{1}$$

In order to estimate p_k the proposed algorithm assumes two types of sensors are available: a noisy Inertial Measurement Unit (IMU) from which a rough estimate of the vehicle displacement between different instants of time may be computed, and a noisy sonar that measures the vehicle's distance to the bottom of the sea. It is further assumed that a map of the environment is available for the localization algorithm. As explained below, particle filters provide a systematic form of combining these sensors with the map in order to achieve a good estimate of the pose of the system.

3. PARTICLE FILTER APPROACH TO TERRAIN BASED LOCALIZATION

Since particle filters are also based on the steps of prediction and correction, they are classified as a particular type of Bayes Filter (Thrun et al., 2005). Furthermore, they are said to be non-parametric because their probability distribution is computed numerically, and is not necessarily parameterizable by, e.g., a mean and a variance. The filter has at its core idea the use of the so-called particles: virtual representations of the posteriori knowledge of the states that must be estimated.

Let us define the set of all particles at instant k by $\chi_k := \{p_k^1,\ p_k^2, ...,\ p_k^M\}$ where M is the total number of particles. Each, p_k^i represents a possible state configuration at instant k, i.e., a possible pose of the vehicle. Through the particle group χ_k the filter generates an estimate $\hat{p}_k$ of the vehicle pose according to,

$$\hat{p}_k = \frac{1}{M} \sum_{i=1}^{M} \hat{p}_k^i , \tag{2}$$

which is also referred to as the *belief* of the states. The above may result in a good estimate provided that all particles are converging to the same location. Advanced forms of estimating $\hat{p}_k$ from χ_k take into account the fact that the denser a subregion of the state space is populated by particles, the more likely it is that the true state falls in that region.

There are many variations of the particle filter, the one used in this paper may be briefly described by the following four steps.

(1) The filter is initialized with M particles with random deviations from a known initial pose (which may be acquired by a GPS signal before the vehicle submerges). Each particle has its own pose $\hat{p}_k^i$ with estimate values $\hat{x}^i$, $\hat{y}^i$ and $\hat{\theta}^i$ at instant k.
(2) For each particle the filter uses the IMU measurements and the past pose to make the prediction of the new pose. These predictions are also perturbed by random variables thus adding variation to the particles in order to take into account uncertainty in IMU measurements.
(3) A value representing the probability w that each particle is at the right position is computed and denominated *importance factor*. In order to compute w_k^i, the filter compares the distance of particle i to the sea floor given by the map $\bar{z}^i$, with the actual measurement z_k acquired by the sonar. In particular, the importance factor w_k^i may be computed by a Gaussian with covariance σ_w (a free parameter):
$$w_k^i = \frac{1}{\sigma_w\sqrt{2\pi}} e^{-(\bar{z}^i - z_k)^2/(2\sigma_w^2)} \cdot w_{k-1}^i \tag{3}$$
The importance factor is attributed for every particle and computed recursively, with $w_0^i = 1$.
(4) The final step is called *importance sampling* where the particles with low probability are replaced by copies of particles with large probability, resulting in a new set χ_k.

Remark 1. While at each instant k the filter iterates steps two and three, the step of *importance sampling* is applied with a lower frequency. This is necessary in order to let the particles propagate through the environment and differentiate among themselves.

4. TRAJECTORY GENERATION

A crucial step for a good performance of the filter is the computation of the importance factor w in step three. In this step the filter compares the distance to the seafloor of each particle given by the map with the actual measurement acquired by the sonar. It is important to point out that all particles that lay in regions of the map with the same depth will have the same value $\bar{z}^i$, being, therefore, indistinguishable from one another. The only possibility of differentiation is if the particles have passed through different depths in previous instants, since w_{k-l}^i for $l > 1$ will be eventually different from w_{k-l}^j for $j \neq i$. This is yet another reason for following the resampling rule stated in Remark 1. Nevertheless, given a terrain that has sufficiently smooth floor, there will eventually come an instant when all particles will be indistinguishable from each other. This will become clear in the simulation results of Section 5, where the path planning algorithm is demonstrated.

Trajectory generation is a common research topic of autonomous and non-autonomous vehicles. For example, Garau et al. (2005) presents a path planing algorithm for underwater obstacle avoidance that seeks to minimizes energy costs using an A* algorithm and the idea of gathering information of the terrain for path planning has been presented by Hausler et al. (2013). Here we propose an algorithm that generates trajectories that avoid regions where the terrain has small variations. A path planning algorithm that defines an optimal action δ for each state (x, y) based on a Markov Decision Process (MDP) is proposed. In order to do so, a value function V is associated with every policy the vehicle may take, representing its *cumulative payoff*. In order to compute V, a *local payoff* function $R(x, y)$ that determines the movement costs for each map position is necessary (Thrun et al., 2005). In order to compute such function we start by computing the maximum terrain variation in each coordinate:

Algorithm 1 Payoff Function

1: **procedure** $(M_I,\ \alpha,\ \lambda_c,\ \lambda_G,\ x_g$ and $y_g)$
2:     **for all** x and y **do**
3:         **if** $x{=}x_g \wedge y{=}y_g$ **then**
4:             $R(x,y){=}\lambda_G$
5:         **else if** $M_I(x,y) > \alpha$ **then**
6:             $R(x,y) = M_I(x,y) - \lambda_c$
7:         **else**
8:             $R(x,y) = -\lambda_c$
9:         **end if**
10:     **end for**
11:     **return** $R(x,y)$
12: **end procedure**

$$M_I(x, y) = \max\left(|\nabla M_B(x)|, |\nabla M_B(y)|\right) \tag{4}$$

where M_B represents a $2D$ matrix containing the x and y coordinates of the environment and their respective depth values $\bar{z}$. Furthermore, $\nabla M_B(i)$, $i = x, y$ represents the gradient of the map in the i direction and the terrain variation matrix $M_I(x, y)$ must then be normalized so that it possesses bounded values between $[0, 1]$ (high values represent regions with greater depth variations). For each position, a movement cost $R(x, y)$ is computed by Algorithm 1. There, x_g and y_g represent the goal position with a respective payoff given by $\lambda_g >> 1$. Also, $0 < \lambda_c \leq 1$ denotes the cost to travel through any other coordinate of the map. Line 5 in the algorithm avoids chatter during the trajectory by considering the terrain importance only when it passes a threshold determined by $0 < \alpha < 1$.

Once $R(x, y)$ has been determined, the value function V, for all $x, y \in M_B$, is given by,

$$V_k(x, y) = \gamma \max_{\delta} \Big[R(x, y) + \kappa \Big] \tag{5}$$

where,

$$\kappa = \sum_{x',\ y'} V_{k-1}(x', y') p(x', y'|\delta, x, y) - \lambda_M(\delta) \tag{6}$$

where $x' \in [1, 2, \ldots x_{max}]$, $y' \in [1, 2, \ldots y_{max}]$, λ_M is an added direction cost and, since this function is computed recursively until $V_k(x, y) = V_{k-1}(x, y)$, $\gamma < 1$ is a constant necessary for convergence. For deterministic cases $p(x', y'|\delta, x, y) = 1$, otherwise this term represents the probability that the action δ will be performed successfully.

Only the deterministic case is studied in this paper, therefore (5) is simplified to:

$$V_k(x,y) = \gamma \max_{\delta} \left[R(x,y) + \sum_{x',\,y'} V_{k-1}(x',y') - \lambda_M \right] \tag{7}$$

The set of movements the robot can follow, denoted by $\pi : (x,y) \to \delta$, are assumed to be: north(N), south(S), east(E) and west(W) and the diagonals northeast (NE), southeast (SE), southwest (SW), and northwest (NW). These motions are represented by the following values:

$$\begin{aligned} \delta_N &= [\ \ 0 \quad 1] \uparrow & \delta_S &= [\ \ 0 \ -1] \downarrow \\ \delta_W &= [-1 \quad 0] \leftarrow & \delta_E &= [\ \ 1 \quad 0] \rightarrow \\ \delta_{NW} &= [-1 \quad 1] \nwarrow & \delta_{NE} &= [\ \ 1 \quad 1] \nearrow \\ \delta_{SE} &= [\ \ 1 \ -1] \searrow & \delta_{SW} &= [-1 \ -1] \swarrow \end{aligned} \tag{8}$$

and define the direction cost as $\lambda_M(\delta) = \|\delta\|_2$.

Once V has converged, the optimal policies may be computed for the whole map, or from any initial condition, by simple hill climbing techniques:

$$\delta(x,y) = \arg\max_{\delta \in \pi} \left[V(x,y) \right] \tag{9}$$

5. SIMULATION RESULTS

This section will present simulation results that show the potentials of the developed path planning algorithm for the aid in AUV localization. The vehicle under consideration is the torpedo shaped Light Autonomous Underwater Vehicle (LAUV) whose model and identification are given by da Silva et al. (2007).

5.1 Mathematical Models

Since we are only interested in three degrees of freedom, the original model is simplified to a version where only the pose p is represented. Likewise, three velocities $\nu = [u \ v \ r]^\intercal$ with respect to the body frame are considered – respectively, surge, sway and yaw. The dynamic model used in the simulations is fully described by the velocities with respect to the body frame and the pose with respect to the earth-fixed reference frame, that is,

$$p_{k+1} = p_k + T_s J(\theta_k) \cdot \nu_k \ , \tag{10}$$

for some simulation sampling time T_s, and a rotation matrix $J(\theta_k)$ given by,

$$J(\theta_k) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Furthermore, the body fixed equations of motion are determined by,

$$\nu_{k+1} = \nu_k + T_s M^{-1}(\tau_k + D(\nu_k) \cdot \nu_k) \tag{11}$$

where M is the constant inertia mass matrix of the vehicle and $D(\nu_k)$ is the damping matrix, both matrices' parameters are given in da Silva et al. (2007). The input vector $\tau_k := [F \quad 0 \quad \tau_\theta]^\intercal$ comprises the body-fixed thrust force on the u direction and rotation torque around θ.

The IMU sensor provides accelerometers in the body frame directions u and v, and a gyroscope measuring the rotation velocity $\dot\theta = r$. These sensors are simulated by the following equations:

$$\begin{aligned} a_u &= \dot\eta(1) + \varepsilon_{acc} \\ a_v &= \dot\eta(2) + \varepsilon_{acc} \\ g_\theta &= \eta(3) + \varepsilon_{gy} \end{aligned} \tag{12}$$

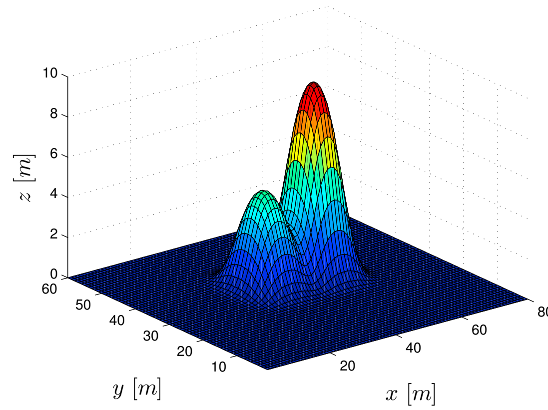

Fig. 2. Synthetic map used to test the proposed localization oriented path generation method.

where ε_{acc} and ε_{gy} represent gaussian noise related to the accelerometer and gyroscope sensors, respectively. In order to simulate the downward pointing sonar a bathymetry map is used in association with the assumed known depth of the vehicle. Thus, the sonar measurement z associated to positions x and y is simulated according to the following equation:

$$z(x,y) = m(x,y) - d + \varepsilon_z \tag{13}$$

where d and ε_z represent the submarine depth (down is positive) and the sonar noise, respectively. The mean and variance of each sensor are detailed in Table 1.

Table 1. Sensor noise parameters.

Noise	Values ($\mathcal{N}(\mu,\sigma)$)
ε_{acc} (accelerometer)	$\mathcal{N}(0.01, 0.001)$
ε_{gy}(gyroscope)	$\mathcal{N}(0.01, 0.1)$
ε_z(sonar)	$\mathcal{N}(0.01, 0.2)$

Two different bathymetry maps are used in order to demonstrate the effectiveness of the proposed method. A synthetic like map that considers a flat region with two central bumps, as depicted in Fig. 2, and a real map depicted in Fig. 5. Both maps are available as matrices whose elements describe the depth at each (x,y) position.

5.2 Results

In what follows, several simulations considering a particle filter with $M = 200$ particles will be shown in order to illustrate and evaluate the proposed approach. During these simulations, the AUV depth is fixed at $d = 20$ m and the particles are initialized around the AUV assumed known initial position. The importance resampling algorithm occurs every 130 samples. In order to prioritize the regions with suitable terrain variation, the terrain importance parameter was set to $\alpha = 0.3$. The cost to reach the goal position is $\lambda_G = 100$ and to reach any other position is $\lambda_c = -1$. Finally, the value function is initialized with $V(x_{goal}, y_{goal}) = 100$, and $V = 0$ otherwise.

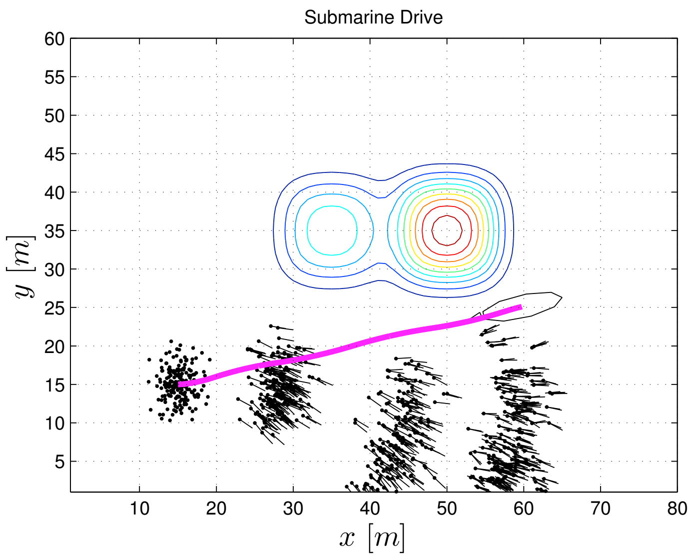

Fig. 3. Simulation with the toy-like map without path planning.

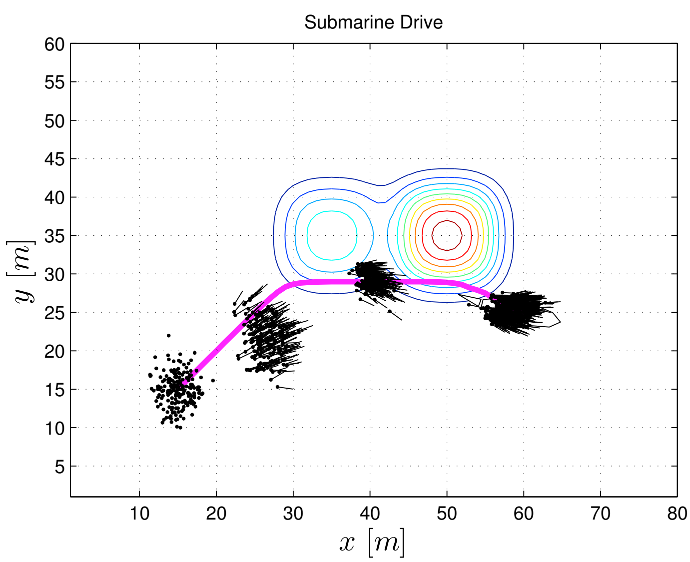

Fig. 4. Simulation with the toy-like map with path planning.

Figure 3 shows a top view of the particle filter localization algorithm being applied to the map in Fig. 2. In this simulation the vehicle is commanded to move from an initial position $(15, 15)$ to the goal at $(60, 25)$. This figure also shows four snapshots of the particle distribution across the map. In order to deal with uncertainty coming from the IMU sensor, the particles disperse as the vehicle moves. However, since the filter cannot differentiate between "good" and "bad" particles, the dispersion continues indefinitely and the filter estimation worsens as time passes. It is clear from this simulation that bathymetry based localization algorithms are helpless in the presence of sea floors with low variation.

The proposed path planning method was then applied to the same map under the same particle filter algorithm, as shown in Fig. 4. The trajectory generation method takes into account the rugosity of the terrain, forcing the vehicle to take a longer, but also safer, path. The particles disperse in a first moment, but as soon as they reach the terrain variations the filter is able to apply the importance resampling algorithm efficiently. As a result, the submarine location is safely estimated throughout the whole trajectory.

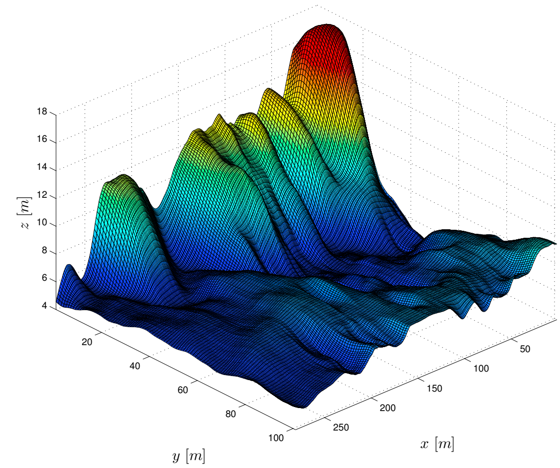

Fig. 5. Real world map used to validate the proposed localization oriented path generation method.

The second set of simulations was applied to the real world bathymetry map depicted in Fig. 5. The vehicle was commanded to move from an initial pose of $p_k = [90 \quad 110 \quad 0]^{\intercal}$ to the location at $(250, 25)$. The results comparing the shortest path trajectory to the proposed path planning algorithm are depicted in Figures 6 and 7, respectively. These plots show once again the advantages of a localization oriented path planing algorithm. Note that the direct route taken in Fig. 6 passes through long flat regions of the map, and for a long time the localization algorithm must rely solely on dead-reckoning, since no useful sonar information is available. The same is not true in the case of the trajectory depicted in Fig. 7, which is clearly longer, but also safer, in the perspective of the particle filter algorithm. Once again the vehicle position is well estimated throughout the whole trajectory.

6. CONCLUSION

This work proposed a path planning algorithm to aid the localization of AUVs under Particle Filters (PF) and bathymetry maps. The resulting method presents itself as a cost effective alternative to the use of triangularization based methods that require external apparatus to the UAV, such as a ship following the vehicle throughout the mission, buoys previously deployed with DGPS sensors, underwater anchors, etc. It has been noted that the PF bathymetry approach is inefficient in the presence of terrains with small variations, effectively turning the algorithm in a dead-reckoning form of estimation. In order to avoid these situations the present work has developed a path planning algorithm that encourages the vehicles to reach the goal target through trajectories that explore the terrain variation. Simulation results including artificial and real bathymetry maps have shown the efficiency of the proposed method.

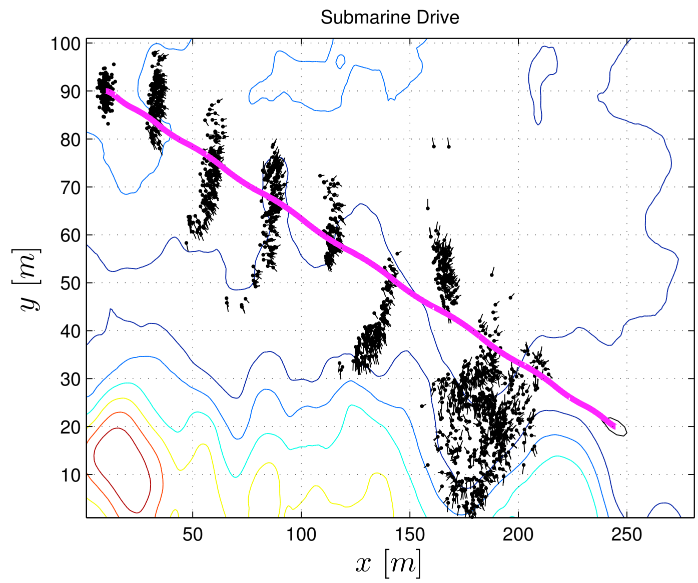

Fig. 6. Real map simulation without path planning.

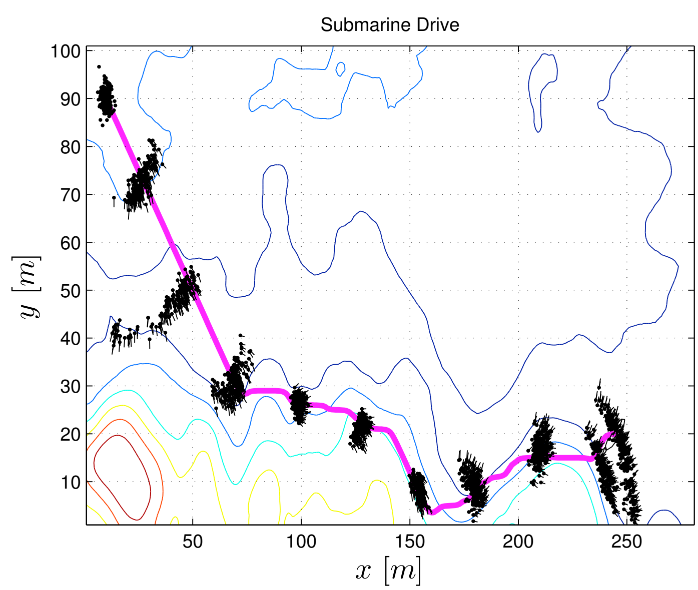

Fig. 7. Real map simulation with path planning.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support by FAPERGS 13/1896-1, CAPES and HP Brasil Ltda.

REFERENCES

Alcocer, A., Oliveira, P., and Pascoal, A. (2006). Underwater acoustic positioning systems based on buoys with gps. In *Proceedings of the Eighth European Conference on Underwater Acoustics*, volume 8, 1–8.

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

Carreras, M., Ridao, P., García, R., and Nicosevici, T. (2003). Vision-based localization of an underwater robot in a structured environment. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 1, 971–976. IEEE.

da Silva, J.E., Terra, B., Martins, R., and de Sousa, J.B. (2007). Modeling and simulation of the lauv autonomous underwater vehicle. In *13th IEEE IFAC International Conference on Methods and Models in Automation and Robotics*.

Garau, B., Alvarez, A., and Oliver, G. (2005). Path planning of autonomous underwater vehicles in current fields with complex spatial variability: an a* approach. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, 194–198. IEEE.

Gordon, N.J., Salmond, D.J., and Smith, A.F. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F (Radar and Signal Processing)*, volume 140, 107–113. IET.

Hausler, A.J., Saccon, A., Pascoal, A.M., Hauser, J., et al. (2013). Cooperative auv motion planning using terrain information. In *OCEANS-Bergen, 2013 MTS/IEEE*, 1–10. IEEE.

Hyakudome, T. (2011). Design of autonomous underwater vehicle. *International Journal of Advanced Robotic Systems*, 8(1), 131–139.

Kalman, R.E. (1960). A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1), 35–45.

Kinsey, J.C., Eustice, R.M., and Whitcomb, L.L. (2006). A survey of underwater vehicle navigation: Recent advances and new challenges. In *IFAC Conference of Manoeuvering and Control of Marine Craft*.

Maurelli, F., Krupinski, S., Petillot, Y., and Salvi, J. (2008). A particle filter approach for auv localization. In *OCEANS 2008*, 1–7. IEEE.

Nakatani, T., Ura, T., Sakamaki, T., and Kojima, J. (2009). Terrain based localization for pinpoint observation of deep seafloors. In *OCEANS 2009-EUROPE*, 1–6. IEEE.

Paull, L., Saeedi, S., Seto, M., and Li, H. (2014). Auv navigation and localization: A review. *Oceanic Engineering, IEEE Journal of*, 39(1), 131–149.

Stutters, L., Liu, H., Tiltman, C., and Brown, D.J. (2008). Navigation technologies for autonomous underwater vehicles. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 38(4), 581–589.

Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic robotics*. MIT press.

Wolbrecht, E., Gill, B., Borth, R., Canning, J., Anderson, M., and Edwards, D. (2014). Hybrid baseline localization for autonomous underwater vehicles. *Journal of Intelligent & Robotic Systems*, 1–19.