Map-based localization in structured underwater environment using simulated hydrodynamic maps and an artificial lateral line

Juan Francisco Fuentes-Pérez¹, Naveed Muhammad², Jeffrey A. Tuhtan¹, Ruth Carbonell-Baeza³,
Mark Musall³, Gert Toming¹ and Maarja Kruusmaa¹

Abstract—Flow sensing has recently gained attention of the robotics community, as it can complement the conventional sensing modalities of vision and sonar in underwater robotics. There is increasing literature in flow sensing for robotics focusing on performing tasks such as object detection and positioning, and robot’s orientation estimation, most commonly under idealized laboratory conditions. In this paper, using recent advances and methodologies for bioinspired flow sensing, we propose a map-based localization technique that employs simulated hydrodynamic maps. The proposed conceptual idea could be an interesting complement to perform localization in those underwater environments with structural maps and a heterogeneous hydrodynamic, such as dams, harbour structures, fishways, caves, swers or any other drowned structure. To demonstrate its performance, computational fluid dynamic models are used to generate flow-speed maps of a structured underwater environment. Later, during off-line experiments, pressure data acquired using a flow sensing probe fitted on a Cartesian robot is transformed into speed information, and used inside a particle-filter to perform localization within the simulated flow-speed maps. The proposed technique has been tested using multiple scenarios with varying particle densities and motion command error levels. The results show filter convergence for all studied scenarios, inducing motion errors up to 0.20 m, suggesting that flow based information could be used to improve the navigation and localization abilities of underwater robots.

Index Terms—artificial lateral line, localization, hydrodynamic simulation, particle-filtering, underwater robotics.

I. INTRODUCTION

Localization, in the context of autonomous robotics, can be defined as the ability to estimate one’s location without relying on an external aid such as global positioning systems (GPS). It is one of the basic problems for any autonomous mobile robot [1], as it is a requirement for performing more complex tasks, e.g. navigation or manipulation.

For performing autonomous localization, robots use perception data from their environment, acquired using exteroceptive sensors such as cameras, sonar, lidar or radar. It can be performed with different levels of autonomy depending on the application at hand, and sensing and computational capabilities of a robot. Localization methods range from dead-reckoning (i.e. integration of motion information from wheel encoders [2]), localization in a priori available maps (i.e. map-based localization), to simultaneously mapping an a priori unknown environment and keeping a track of one’s location in it (i.e. the SLAM problem [3], [4]). Map-based localization is especially viable within man-made environments, such as buildings or road networks, where it is possible to equip a robot with an a priori map. One such example is the map-based localization for autonomously driving vehicles. An example of localization using 3D laser scanner data, within a road network created using OpenStreetMap¹, is presented in [5]. Point cloud data from the 3D laser scanner are used inside a particle-filter localization framework in order to perform map-based localization.

Map-based localization, as well as other localization techniques, have also been investigated for underwater robots, but mostly using vision and sonar sensors [6]. For instance, [7] presents a study where a robot localizes itself with respect to a grid-coded floor of a pool, using a downward facing camera. Likewise, underwater map-based localization using particle-filtering and vision sensing is presented in [8]. A sonar-based implementation of SLAM is presented in [9].

Recently it has been shown that flow can also provide information rich cues about a robot’s surroundings [6], [10]. It can be an interesting alternative or complement in dark, turbid or visually homogeneous underwater environments, where vision sensors are of little utility, and sonar sensors might also fail to discriminate between different locations. This is a bioinspired concept since almost all fish species and a large number of aquatic vertebrates possess a flow sensing organ called the lateral line [11], which is used to perform tasks such as orientation towards flow, object detection or build cognitive maps of their environment [12]. Artificially, flow sensing has been achieved in a number of ways (e.g. piezo-resistive materials [13], optical sensors [14] or absolute pressure sensors [6], [15]).

Flow sensing for underwater robotics has only recently garnered the attention of the wider research community. A commonly studied problem is the detection of idealized shapes, by moving the object [16], [17] or by detecting the generated wake [18], [19]. Likewise, some studies have demonstrated the possibility to estimate robot orientation as well as the flow-speed [6], [20], [21].

The localization problem has also been faced taking advantage of flow sensing devices. For instance, 1D localization

¹http://www.openstreetmap.org
has been proposed by [19] and [22] using the velocity change generated by the interaction of flow with a specific obstacle, and [23] simulates a first approach to perform large scale localization using ocean general circulation models (OGCM). However, aforementioned works on flow sensing in robotics have a strong focus on fundamental tasks (object detection, orientation estimation, flow-speed estimation, etc.) in laboratory conditions. Likewise, solved localization problems so far have been limited to 1D [19], [22] or large scale models (OGCM with 3 km spatial resolution) [23].

More recently, in [24], we have covered 2D localization using flow features and within the same scenario studied in this work. For this, we used a priori scan of the environment made with the same sensing technology used after in the localization stage and an abstract flow feature (i.e. without physical meaning) to successfully achieve 2D localization. However, two main problems were detected for the successful applicability of this conceptual method: (1) the abstract nature of the defined flow feature makes its application impossible in other environments without the pre-calibration of features and (2) the method required, previous to the localization and for each possible hydrodynamic scenario, a priori scan with the same flow sensing technology that is going to be used for localization. Both problems limit its use in difficult access environments, increase operation costs and make difficult to generalize it.

A. Contribution

In this paper we solve aforementioned problems by taking into account (1) the speed estimation techniques proposed by the specialized references (e.g. [19], [21], [22], [25], among others), which will allow to estimate standard flow features (i.e. flow speed and turbulence metrics) and, thus, skip pre-calibration processes, and (2) the advances in 3D computational fluid dynamics (CFD), which will facilitate a flow 3D map without the need of a priori measurement with the flow-probe in any desired hydrodynamic scenario.

The combination of flow sensing systems and standard variables together with simulated 3D maps, spreads the possible applicability of the conceptual method to any scenario where structural drawings are available. For instance, target scenarios and tasks could be inspections of harbour structures, sewers, fishway and pipe inspections, or navigation in caves and drowned structures, where this technique could be used as a complement to other sensing modalities that may be limited due to the constrains of visibility and dimensions of these environments.

In order to demonstrate the utility of the proposed idea we present a localization example using simulated maps as a priori information and flow sensing as the only perception modality. Flow sensing is achieved using an array of absolute pressure sensors (an artificial lateral line) fitted within a fish-shaped probe which is moved using a Cartesian robot. Later, during off-line tests, pressure data from the sensors is used to estimate flow-speed around the probe, and particle-filter is used to perform localization within a flow-speed map simulated by CFD techniques. The environment used for the experimentation is a vertical slot fishway model.

The remainder of the paper is organized as follows: Section II describes the proposed map-based localization approach and the experimental setup, results and their discussion are presented in Section III, and the paper concludes in Section IV.

II. MATERIAL AND METHODS

A. Map-based localization using particle-filtering

Particle-filtering was first proposed as a solution to the robot localization problem in the beginning of the last decade by [26]. Since then, the method has been successfully used for localization of robots with different sensing capabilities, for instance in [1], [5], [27].

Fig. 1 shows the general algorithm used to perform map-based localization using particle-filtering. The 2D position of the robot at a time \( t \) is defined by a set of \( m \) particles \( X_i \); the particles are a representation of the probability distribution of the robot’s pose. In the beginning, the particle set is randomly scattered through the a priori available map (simulated map in our case). Each particle \( i \) has an associated probability or weight \( (w_{t,i}) \) of being the current position of the robot. Until the first observation \( (v_t) \), current flow speed), weights are equally distributed across particles. After a motion step \( (d_t) \) and an observation, \( w_i \) is updated, resampling the \( m \) particles according to the new \( w_i \), i.e. the probability of a particle being sampled will be proportional to its weight, thus, particles with higher \( w_i \) will have more probability to be selected. At any given time, the mean value of the resampled particles is taken to be the estimated position of the robot.

One of the main steps in is the weighting or probability assignment for each particle after the motion step and observation. In our implementation, a normal probability distribution was used, with a mean value equal to the current observation \( (\mu = v_t) \) and a span equal to the possible range in the observation (variance \( (\sigma) \) equal to the one observed in the simulated map).

During the application of the algorithm first state was defined as the preliminary position before updating, second state as the first state after an update and a resample step

2Fishways or fish passes are structures built to enable fish migration in transversal obstacles to the rivers.
and successive states included a motion, an update and a resample step.

B. Flow sensing and flow-speed estimation

To perform localization within the simulated map it is necessary to measure simulable flow features. In this sense, the proposed conceptual idea could be developed using different sensing technologies. In this paper, a robotic probe that mimics the body shape of a 0.45 m length rainbow trout equipped with an artificial lateral line (ALL) has been used (Fig. 2). The ALL consists of an array of 16 absolute pressure sensors distributed over the robotic probe (Fig. 2).

The probe measures the pressure \( p \) distribution over the body of the robot. This pressure signals can be used to estimate the depth as well the flow-speed \( v \). There are different techniques for \( v \) estimation. For instance, directly by means of the difference in pressure measurements between sensors [19], [25], or by taking into account the pressure fluctuations produced by different velocities over the body [21]. The latter method was used to test the proposed localization method and estimate flow-speed in this study.

Fig. 3 summarizes the algorithm used for calculating \( \bar{v} \) and \( v_{signal} \) (speed at original sampling rate of the probe). Once the pressure data \( (p_{signal}) \) is registered by a sensor, its fluctuations \( p' \) are calculated by subtracting the observed mean value \( (\bar{p}) \) from \( p_{signal} \). Afterwards \( p' \) is transformed to the frequency domain using fast-Fourier-transform, and then a band-pass filter is applied (to remove high frequency acoustic noise and low frequency fluctuations in the water surface). Then \( \bar{v} \) is calculated through its relation with the mean amplitudes of the remaining frequencies \( (A) \). Finally, resampling is done by taking advantage of the specific relation between both the difference in pressure between sensors, and \( \bar{v} \), as well as the observed \( p' \). However \( \bar{v} \) is enough to perform localization. Further description of this method as the analysis of the expected performance can be found in [21].

C. Experimental setup and validation

To test the proposed approach, a 1:1.6 scale fishway (pool dimensions: 1.70 m by 2.39 m), located at Theodor Rehbock Hydraulic Engineering Laboratory (Karlsruhe Institute of Technology), Germany, was selected (for a complete geometrical description see [28]). A fish pass was chosen as the experimental environment because it is a well-known structure with diverse and complex flow pattern [29], providing the opportunity to exploit simulated maps for map-based localization. The experimental setup along with the flow sensing probe is shown in Fig. 4.

The simulated flow-speed maps of the environment were obtained using OpenFOAM®, a standard OpenSource CFD tool. As the problem consists in two phases (air and water) interFOAM solver was used. Likewise, natural scenarios, as in our test environment, are highly turbulent. In order to solve this two main methods can be used for practical (computationally manageable) turbulence simulation: Reynolds averaged Navier-Stokes equations (RANS), and
large eddy simulations (LES). For this study 3D RANS (k-epsilon model) simulations under various experimental conditions were selected to report the results. Fig. 5 shows the simulated flow-speed map for flow discharges of 0.17 m$^3$/s and 0.13 m$^3$/s at 40% of the water column height.

The localization was performed off-line on flow data that was acquired by attaching the probe to a Cartesian robot with three degrees of freedom (Fig. 4 (a)). Two different flow scenarios were studied, 0.13 m$^3$/s and 0.17 m$^3$/s, each with three different heights within the water column ($h_0$): 0.25$h_0$, 0.40$h_0$, and 0.60$h_0$ (i.e. at 25, 40, and 60% of the water column height).

Measurements were carried out following the main trajectory shown in Fig. 4 (b). However, due to its off-line implementation, any measured point combination can be tested using the localization algorithm. The algorithm and methodology were evaluated comparing the evolution of the position error according to different particle densities (500, 1500 and 2500), noise levels in robot movements and flow speed readings, and different simulated scenarios (different discharges and depths), over a fixed trajectory.

Normally distributed random errors were added to artificially distort the motion of the robot ($\mu = 0$ and $\sigma = 0.05, 0.10$ and 0.20 m), and hence to check the performance of the proposed method. Likewise, the robot needs to face the current to estimate the flow speed accurately. This can be controlled by monitoring pressure differences in both sides [30]. However, until the orientation correction, small distortions may occur which are translated into flow speed estimation errors. Thus, the same error levels were added into the estimated speeds (normal distribution, $\mu = 0$ and $\sigma = 0.05, 0.10$ and 0.20 m/s).

Regarding the studied scenario, one of the main characteristics of this kind of fishway is that the velocity distribution remains almost constant for different depths [29], thus, only 2D localization was tested. However, absolute pressure readings in pressure sensors could be used as depth sensors, providing a full 3D estimate of the position.

### III. RESULTS AND DISCUSSION

#### A. Results

A exemplary result of one of the 2D localization experiments is shown in Fig. 6 ($\sigma = 0.20$ m and 0.20 m/s, 500 particles, 0.40$h_0$ and 0.17 m$^3$/s). The real position of the robot is shown in green, whereas the trajectory based on control inputs is shown in blue. In the first state (Fig. 6 (a)) the particles are randomly distributed over the simulated map. The second state updates the localization estimate only taking into account the observation in the first position. Finally (Fig. 6 (b and c)), the localization in each state is updated considering the control inputs and the new observations. Fig. 6 (d) shows the evolution of the absolute localization error over the different states.

Table I summarizes the performance of all the experiments conducted (i.e. three different heights for each of the two flow discharges of 0.13 m$^3$/s and 0.17 m$^3$/s, six simulated maps in total). The values shown for each type of experiment in the table are mean values calculated for fifty random trials for each depth studied. The first parameter (state ($\varepsilon < 0.2$ m)) corresponds to the state where a localization absolute error smaller than 0.20 m is reached (localization convergence, Fig. 6 (d)), while the second error ($\varepsilon_{\text{final}}$) represents the localization absolute error in the final state and the third is standard deviation of this error ($\sigma_{\varepsilon}$). These parameters provide a general overview of the localization algorithm performance under different configurations.

Using an off-line implementation, it is possible to simulate robot trajectories, regardless of the sequence in which the data acquisition was done. The last three rows of the Table I show the performance of an alternative path during 0.170 m$^3$/s scenario (Fig.4 (b)).

#### B. Discussion

In all cases exposed (Table I), the estimated localization converges to the real location of the robot. No significant difference was observed between the depths and flow discharges investigated during the study ($t$-test, $p$-value $> 0.05$). As expected, in all the experiments it was observed that a larger error in motion commands and flow estimates usually is translated into a larger localization error, and thus more...
move-and-observe steps are required to converge to real robot location. For added speed errors larger than 0.3 m/s, the convergence was not always produced as it is translated into an error larger than 100% in the 80% of the scenario for 0.17 m³/s (larger in the case of 0.13 m³/s). Likewise, as all the errors are considered positives, whatever the direction, \( \epsilon_{\text{final}} \) after convergence is near to the introduced error in the motion command. Thus if positive and negative direction had been considered \( \epsilon_{\text{final}} \) (average of all trials) would have been close to 0.

The number of particles in many cases was found to be positively proportional to the accuracy of the estimation of the location. However, the observed difference was not significant (t-test, \( p \)-value > 0.05), concluding that the use of the lower number of particles is the most interesting alternative, as the lower the number of particle the lower

TABLE I

<table>
<thead>
<tr>
<th>Discharge (m³/s)</th>
<th>Variables</th>
<th>Particle density</th>
<th>Introduction error (( \sigma ))</th>
<th>500</th>
<th>1500</th>
<th>2500</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.170</td>
<td>State (( \epsilon &lt; 0.2 ) m)</td>
<td>[6 7] [6 7] [7 10] [6 7] [6 7] [8 10] [6 7] [6 7] [7]</td>
<td>0.066</td>
<td>0.091</td>
<td>0.204</td>
<td>0.061</td>
</tr>
<tr>
<td></td>
<td>( \epsilon_{\text{final}} ) (m)</td>
<td>0.037</td>
<td>0.049</td>
<td>0.104</td>
<td>0.027</td>
<td>0.037</td>
</tr>
<tr>
<td></td>
<td>( \sigma_{\epsilon} ) (m)</td>
<td>0.061</td>
<td>0.081</td>
<td>0.188</td>
<td>0.052</td>
<td>0.081</td>
</tr>
<tr>
<td>0.130</td>
<td>State (( \epsilon &lt; 0.2 ) m)</td>
<td>[6 7] [6 7] [6 10] [6 7] [6 7] [6 10] [6 7] [6 10] [6]</td>
<td>0.039</td>
<td>0.046</td>
<td>0.103</td>
<td>0.025</td>
</tr>
<tr>
<td></td>
<td>( \epsilon_{\text{final}} ) (m)</td>
<td>0.039</td>
<td>0.061</td>
<td>0.188</td>
<td>0.052</td>
<td>0.081</td>
</tr>
<tr>
<td></td>
<td>( \sigma_{\epsilon} ) (m)</td>
<td>0.067</td>
<td>0.088</td>
<td>0.213</td>
<td>0.059</td>
<td>0.087</td>
</tr>
<tr>
<td>0.170</td>
<td>State (( \epsilon &lt; 0.2 ) m)</td>
<td>[4] [4] [4] [4] [4] [4] [4] [4] [4]</td>
<td>0.067</td>
<td>0.088</td>
<td>0.213</td>
<td>0.059</td>
</tr>
<tr>
<td>Alternative trajectory</td>
<td>( \epsilon_{\text{final}} ) (m)</td>
<td>0.043</td>
<td>0.046</td>
<td>0.115</td>
<td>0.029</td>
<td>0.044</td>
</tr>
</tbody>
</table>
the computational cost.

Localization results from the alternative robot trajectory (last two rows of the Table I) show the use of the proposed technique for different trajectories. As it is apparent from the table, a similar performance to the original robot trajectory was observed, with a faster convergence rate.

The faster convergence is explained by the speed distribution in the studied scenario (Fig. 5). This is characterized by a small jet region with high speeds surrounded by two bigger areas of lower speeds. Thus, when high speeds are perceived the probable localization area is reduced faster, converging into a solution. This suggests that localization in highly heterogeneous and non-symmetric scenarios will be faster. Same conclusions can be deduced from other studies where a strong change in the perception is used to perform localization [19], [22], [24]. In the same way, the localization can be limited in homogeneous regions or areas with velocities lower than the sensitivity of the device under use.

One of the main advantages of the proposed conceptual method, is that it allows localization in environments where only structural maps (in the form of CAD drawings, for instance) are available and using standard CFD simulation techniques, in contrast to other methods that use not simulable features [24]. In the same way, despite the localization convergence is comparable to [24] (in most cases state 6-7 for the main trajectory) the final error in some studied scenarios are slightly higher. This is due to two reasons: (1) the maps used in [24] are measured a priori with the same technology (which limits its use in other scenarios but reduces the matching error) and (2) no errors are considered in the flow feature readings (which likely occurs in real setups). Likewise, the used CFD techniques compared to other possible simulations techniques (e.g. potential flow [22]) can have a higher computational cost (cell side of 0.04 m). However, it is possible to model accurately a wide array of complex geometric scenarios and, furthermore, simulations can always be done a priori to application.

In the case under study the results indicate the possible use of flow sensing for localization in hydrodynamically diverse structured environments, but also show the potential of flow sensing to be incorporated into underwater navigation in conjunction with other sensing modalities such as sonar and vision. Such multi-modal perception and navigation techniques (demonstrated by works such as [31], [32]) are attractive because each sensing modality, be it vision, sonar, flow, lidar or radar, has its inherent pros and cons.

It is worth mentioning that it is possible to replace the speed estimate method used by any of the techniques proposed by the specialized references (e.g. using ALL [19], [22], [25] or acoustic Doppler profilers (ADP) [33]) expecting similar behavior with more or less accuracy depending on the errors. Likewise, ideally, the combination of multiple technologies will provide a better estimate.

During the experiments presented in this paper, the pressure sensing probe was moved using a Cartesian robot and it was held stationary while the pressure data were logged. When an underwater robot is in motion, it perceives the external flow stimuli together with the effect of its own movement. This can be considered with techniques such as the one presented in [34], which relates the pressure data sensed by a robot to its speed and acceleration using a second-order polynomial model allowing the estimation of the flow-speed in moving platforms. Another alternative could be the combination of multiple technologies, for instance the use of acoustic Doppler velocimetry together with an ALL or an ADP. The online implementation requires the fastest possible velocity estimation. In this sense, latest advances in velocity estimation with ALL can provide 10Hz of stable velocity estimation and an accuracy of 0.024 m/s [25].

In the future we plan to investigate the proposed map-based localization technique for a continuously moving platform. Another research avenue that we plan to investigate is the combination of flow-speed estimation together with turbulence metrics (e.g. turbulence kinetic energy or turbulence intensity), as their simulation is also possible with CFD techniques and their heterogeneity is high within a scenario, which makes them interesting for robot localization.

IV. Conclusion

This study presents a conceptual method for underwater map-based localization using simulated maps of the flow-speed profile of an underwater scenario. To perform the localization, particle-filtering is used. The study shows that simulated flow-speed maps of structured underwater environments could enable map-based localization using flow and demonstrates an improvement of previously proposed techniques by extending and standardizing its application. Likewise, the results also indicate the potential of flow sensing to complement the conventional underwater sensing modalities of vision and sonar.

ACKNOWLEDGMENTS

The research leading to these results has received funding from BONUS, the joint Baltic research and development programme, cofinanced by the European Union’s Seventh Framework Programme (2007-2013) under the BONUS Implementation Agreement. National funding for this work has been provided by the Academy of Finland (under the grant 280715), the German Federal Ministry for Education and Research (BMBF FKZ:03F0687A), and the Estonian Environmental Investment Centre (KIK P.7254 C.3255). The work has also been partly financed by the EU FP7 project ROBOCADEMY (No. 608096) and by institutional research funding IUT (IUT33-9) of the Estonian Ministry of Education and Research.

REFERENCES

