Simultaneous Mapping and Planning for Autonomous Underwater Vehicles in Unknown Environments

Juan David Hernández, Eduard Vidal, Guillem Vallicrosa, Èric Pairet and Marc Carreras
Underwater Vision and Robotics Research Centre
Parc Científic i Tecnològic, Universitat de Girona, Pic de Peguera 13, 17003 Girona, Spain
juandhv@eia.udg.edu

***Abstract*— New potential applications of autonomous underwater vehicles (AUVs) involve operations in unknown and cluttered environments, therefore increasing the vehicle exposure to collisions. To cope with these situations, we use an AUV framework for planning collision-free paths in unknown environments, which adapt and replan the paths according to nearby obstacles perceived during the mission execution using different range sensing sonar. We present simulation and real-world results for the SPARUS-II AUV, a torpedo-shaped vehicle, performing autonomous missions.**

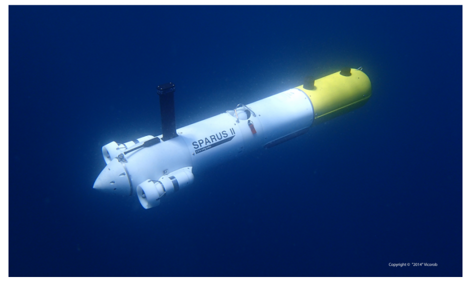


Fig. 1: SPARUS-II, a torpedo-shaped AUV.

I. INTRODUCTION

Originally, a significant number of the applications for autonomous underwater vehicles (AUVs) were devoted to conduct surveys of underwater environments. This meant that vehicles followed a sequence of pre-calculated waypoints at a constant and safe altitude from the seafloor to collect data. Nowadays, the developments in sensors and actuators, but especially the increase of the processing power have fostered new applications in which the AUV has to use environment information gathered with on-board perception sensors. Examples include imaging and inspection of underwater structures in close proximity, in which information obtained from range sensors allow adapting online the survey path [1]. Another example is the exploration of confined and natural structures, *e.g.,* underwater caverns, where acoustic sensors permit to reconstruct a 3-dimensional (3D) model of the explored area [2].

In such scenarios, AUVs operate in unknown and cluttered environments and cannot rely on *a priori* knowledge of the work area. Furthermore, they are also affected by drift effects on the position estimated by their navigation systems, thus exposing them to collisions. Dealing with such constraints requires a framework with online mapping and path planning capabilities that can contribute to overcome global position inaccuracy, especially the one related to nearby obstacles.

A. Motivation

Aiming to cope with the aforementioned requirements, we present an AUV framework for both mapping and planning paths online in unknown environments. The proposed framework incrementally constructs a 3D representation of the environment (*i.e.,* map) using different type of perception sensors, such as multibeam sonars, mechanically scanned profiling sonars, echosounders, etc. At the same time, the framework finds and recalculates collision-free paths insofar that the map is updated.

B. Related Work

Path planning consists in finding a collision-free path from a start configuration to a goal configuration in the configuration space (C-Space), which is the space of possible robot configurations [3]. There are different computational algorithms that attempt to solve this task. A group of them search throughout discretization of the configuration space, such as the A* family of algorithms [4]. However, sampling-based methods [3] have proved effective in problems involving high-dimensional C-Spaces, motion constraints and online computation requirements.

LaValle and Kuffner [5] introduced the rapidly-exploring random tree (RRT), a sampling tree-based algorithm for solving path planning problems, and firstly proposed for systems with kinodynamic (differential) constraints. Rooted at an initial configuration (state), the tree grows by adding collision-free states that are obtained by expanding itself towards random and uniformly distributed samples of the C-Space. Based on the original approach, Karaman and Frazzoli [6] introduced the RRT* and its concept of asymptotic optimality, which states that the total cost of the solution, measured by a user-defined function, decreases as the number of samples increases. In this approach, new configurations are connected to the closest and best configuration, *i.e.,* the one that guarantees a minimum cost. Furthermore, an additional step of sample reconnection allows to improve costs to surrounding states. A typical growth process of an RRT and an RRT* can be clearly observed in Fig. 2. The presented framework utilizes an RRT* variant, which has been adapted to compute collision-free paths online in a anytime fashion.

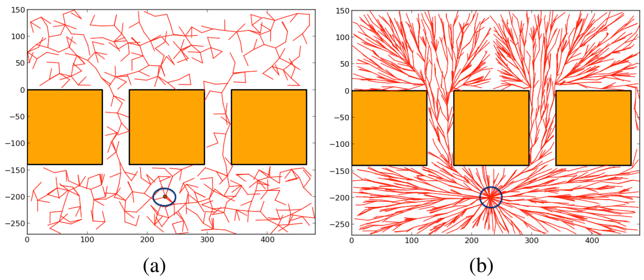


Fig. 2: Growth of (a) an RRT and (b) an RRT* in a 2-dimensional workspace with no goal specified. The starting state of the tree is appreciated enclosed in circle.

1) Anytime Algorithms and Online Planning: An *anytime* algorithm is capable of returning the best partial solution when planning time is over. With this concept, most relevant and known planning algorithms have been extended, including grid-based methods (*e.g.,* A*, D*) [7] and sampling-based ones, such as the probabilistic roadmap (PRM) [8] and the RRT [6].

Most remarkable contributions in terrestrial vehicles have been presented as results of the DARPA Grand Challenge[1]. Likhachev and Ferguson [9] used a lattice as discretization of the configuration space. An anytime dynamic A* (AD*) finds paths over the lattice. Kuwata *et al.* [10] presented an alternative approach using an RRT, which considers the vehicle model and the controller dynamic behavior. Dolgov *et al.* [11] also presented an alternative where the state space is discretized and paths are found by running an A* variant (Hybrid-State A*). In all these cases, a common aspect of our interest is that vehicles cope with unknown environments.

2) Path Planning for AUVs in Unknown Environments: Despite the above-mentioned contributions in terrestrial robotics, available research on path/motion planning for AUVs, especially in unknown environments, is still limited to simulation in most cases. Focusing specifically on those devoted to solve start-to-goal queries there are different approaches. Warren [12] used repelling and attractive potential fields, for obstacles and goals respectively, to compute the obstacle-free paths. Carroll *et al.* [13] discretized and represented the workspace with quadtrees to find minimum-cost paths over them by using an A*. Alvarez *et al.* [14] used genetic algorithm (GA) to find an optimal path while avoiding getting trapped by local minima. Petres *et al.* [15] proposed to use fast marching (FM) method, *i.e.,* a level-set method that permits to obtain a solution of a minimization problem (the distance in path planning queries). However, in all these cases, complete knowledge of the environment was required and assumed as available in datasets such as bathymetric maps.

[1]The DARPA Grand Challenge is a competition of autonomous vehicles, funded by the Defense Advanced Research Projects Agency (DARPA).

There exists, however, other cases that use environment information obtained by on-board sensors. Petillot *et al.* [16] presented a first approach for online obstacle avoidance and path planning for underwater vehicles that uses a real-world dataset of acoustic images obtained by a remotely operated vehicle (ROV) equipped with multibeam forward looking sonar. They demonstrated the validity of their framework by guiding a simulated model of a ROV based on dataset information. However, capacity of simultaneous mapping and planning online was not proven. Another approach proposed an online path planning method that used landmarks to guide the vehicle, though it does not permit replanning. Additionally, the results were obtained in a water tank, *i.e.,* a controlled environment instead of a real-world setting [17].

C. Outline

Finally, to overcome some of the limitations of the existing approaches and to validate our alternative approach, we used our framework for solving and conducting a start-to-goal task in an unknown and challenging scenario, in which online mapping and planning (replanning) is required, thus demonstrating not only its functionality in this particular case, but also its suitability for the applications mentioned at the beginning. Results include simulations with different perception sensors commonly used in underwater environments. To corroborate its applicability, we also present real-world tests with the SPARUS-II [18], a torpedo-shaped AUV (see Fig. 1) equipped with a set of echosounders.

II. SIMULTANEOUS MAPPING AND PLANNING

The framework is composed of three main modules (see Fig. 3). Firstly, a *mapping* module that incrementally builds an occupancy map using Octomaps, an octree-based volumetric representation of the environment [19]. Secondly, a *planning* module that generates online collision-free paths with our modified version of a sampling-based method [3]. Finally, a *mission handler* module that works as a high-level coordinator of the planner and the AUV controllers.

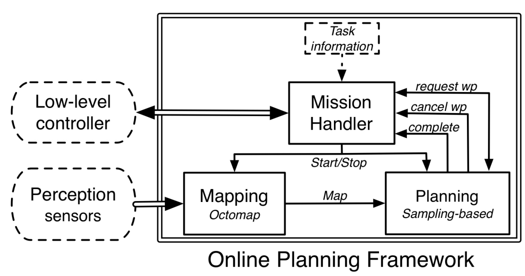


Fig. 3: Main modules of the online planning framework.

A. Mapping Online Underwater Environments

In mobile robotics, exteroceptive sensors are used to gather environment information for different purposes, such as obstacle detection, environment mapping, robot localization or even a combination of them, as occurs in a typical simultaneous localization and mapping (SLAM) application.

In what path/motion planning concerns, sensors must provide data to build a representation of the environment (map) over which the planner has to find collision-free and feasible paths. In contrast to what occurs in ground and aerial applications, where laser-based range sensors and cameras provide reliable and accurate information of nearby obstacles, their applicability is very limited in underwater environments, where visibility is a highly variable condition and is also correlated with the light attenuation. Thus, acoustic sensors provide a more reliable solution in such scenarios. Active sonars transmit and receive acoustic signals to estimate the distance to reflecting objects (obstacles).

Most common acoustic perception sensors can be classified into two main groups, range sensing sonars and imaging sonars. When dealing with online computation constraints, it is necessary to define a trade-off between the amount of detail and the computation time required to process the data. For this reason, we decided to use range sensing sonars within our framework, which provide us with distance information relative to nearby obstacles. This group includes echosounders, profilers and multibeam sonars. In this section, we present their most relevant characteristics and the octree-based structure selected to represent their data. In Section III, we present simulation and real-world results in a simultaneous mapping and planning task.

1) Single-beam echosounders: are devices typically mounted pointing downwards on different maritime systems to estimate the ocean depth. They use one emitting and receiving transducer that releases acoustic signals in the form of a narrow beam (with a typical aperture of $\sim 10°$). The approximate distance to the seabed is calculated using the time required to receive the echo signal and the known speed of sound in water. They can also be used to detect obstacles if they are mounted pointing towards the vehicle's motion direction (see Fig. 4).

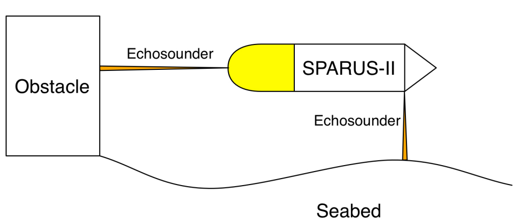


Fig. 4: Typical setup for a single-beam sonar in an AUV. One of the echosounder is mounted looking downward to distance to the seabed, while a second one is oriented towards the vehicle's motion direction.

2) Mechanically-scanning profilers: are devices that work on the same principle of single-beam sonars, but are mechanically actuated in a way that permits to sequentially reorient the single beam to cover a predefined sector, thus producing a series of range data (see Fig. 5).

3) Multibeam sonars: are devices that, similarly to scanning profilers, produce a series of range data of a scan sector. The main difference relies on the fact that all beams are triggered simultaneously, which permits to cover completely the scan sector in each period of time.

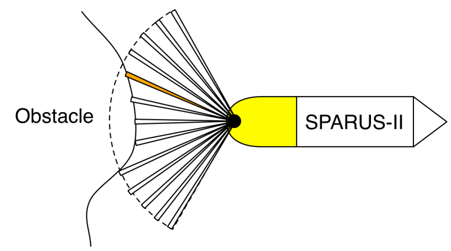


Fig. 5: Setup for profiling sonar in an AUV. The profiler can cover a predefined scan sector by sequentially reorienting a single-beam sonar signal. The figure indicates that only one position is being evaluated at the same time.

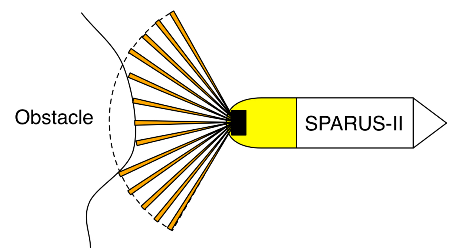


Fig. 6: Setup for multibeam sonar in an AUV. The multibeam can cover a scan sector. The figure indicates that all beams are being evaluated at the same time.

4) Octree-based representation of environment data: In our mapping and path planning context, an AUV captures environment distance data through any of the aforementioned range-based sonars. That information is used to create a 3-dimensional (3D) model that must be employed to plan collision-free and feasible paths. There are different alternatives to represent 3D workspaces such as point clouds, elevation maps, and multi-level surface maps. Such approaches have features that do not fulfill some relevant aspects of our research. For instance, point clouds store large amounts of information, making it a memory-inefficient option for an on-line application. On the other hand, together with point clouds, elevation and multi-level maps do not permit to differentiate between obstacle-free areas and unexplored areas, which can be critical when performing missions in environments where no previous information is available, as is our case. Considering these facts, the proposed framework, and specifically its *mapping* module, uses Octomap [19], an octree-based framework for modeling volumetric information. The module incrementally builds a representation of the environment using information received from range sensing sonars, thus defining the free and occupied space with respect an inertial coordinate frame.

Octomaps [19] have three main characteristics that contribute directly to our online mapping and path planning application. The first of them is the probabilistic state representation that considers previous information when updating states, which calculates new state values according to probabilistic functions, thus not only updating map information, but also protecting it from noisy measurements. The second is the capacity of representing unexplored areas, which is the particular interest for exploration tasks. Finally, Octomap offers an efficient method for modeling volumetric information, since its computational costs in terms of time of access and memory consumption are less that other alternatives, for instance, it does not have to be initialized with a predefined size, but can be enlarged or extended as demanded.

B. Incremental Path Planning

The *planning* module receives a query to be solved, specified as a start and goal vehicle configuration, as well as additional planning parameters, such as the available computing time, minimum distance to the goal, and workspace boundaries. One of the most challenging aspects presented on this work lies on the fact that no previous information of the environment is assumed. In order to find collision-free paths under this assumption, this module periodically requests an updated version of the map (Octomap) to validate if the current path to the goal is still feasible. If it is not, the module discards the path and reuses existing information to find a new valid solution.

To incrementally solve the query, *i.e.,* by updating the solution as the vehicle moves towards the goal, this module contains our modified version of the sampling-based method RRT [5], [6], which has been extended with the concept of *anytime* algorithms, thus enabling online and fast planning of collision-free paths. The main advantage of using a tree-based method is that discarding a colliding part implies removing the affected branch of tree, but without recalculating completely the solution. Section III presents some scenarios where this characteristic can be clearly observed.

III. RESULTS

As a test scenario to evaluate our approach we used the harbour of Sant Feliu de Guíxols (see Fig. 7) in Catalonia, Spain. Experiments were conducted in the external and open area of the harbour, in a breakwater structure (marked with a red ellipse) that is composed by a series of concrete blocks of $14.5m$ long and $12m$ width, separated by a $4m$ gap with an average depth of $7m$. We used the SPARUS-II AUV (see Fig. 1), a torpedo-shaped vehicle with hovering capabilities, rated for depths up to $200m$. The robot has three thrusters (two horizontal and one vertical) and can be actuated in surge, heave and yaw degrees of freedoms (DOFs). The vehicle is equipped with a navigation sensor suite including a pressure sensor, a doppler velocity log (DVL), an inertial measurement unit (IMU) and a GPS to receive fixes while at surface.

Before conducting in-water trials, we created an equivalent virtual environment using underwater simulator (UWSim) [20], in which we could validate our mapping and planning approach. We defined three different sensor configurations to perceive the environment. In the first of them, a set of four echosounders are located within the vehicle payload (front) area pointing in the horizontal plane, three are separated by $45°$, with the central one looking forward and parallel to the vehicle's direction of motion, while the fourth one is perpendicular to the central one (see Figs. 8, 9). In the second scenario, we simulated a mechanically-scanning profiler with an aperture of $120°$ (see Fig. 10). The last evaluated scenario used a multibeam sonar also with an aperture of $120°$ (see Fig. 11).

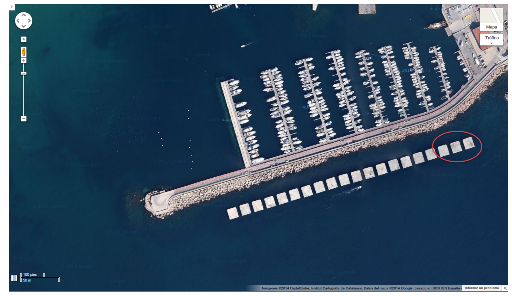


Fig. 7: Experiments scenario. Harbor of Sant Feliu de Guíxols in Catalonia, Spain, where a breakwater structure composed of concrete blocks is demarcated.

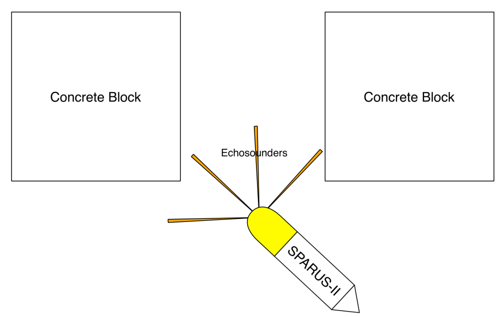
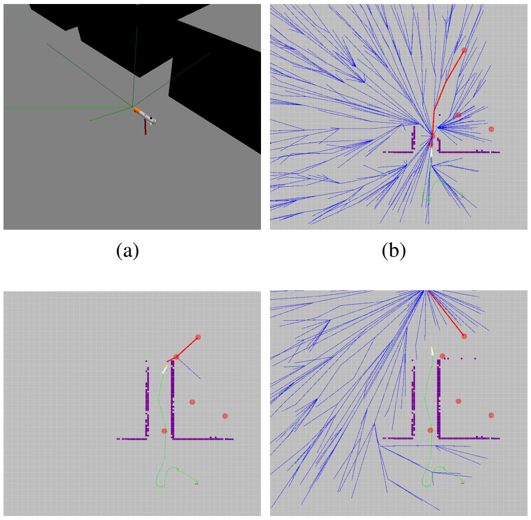


Fig. 8: Perception sensors configuration. Top view of the echosounders beams direction in the horizontal plane the scenario of series of blocks (not drawn to scale).

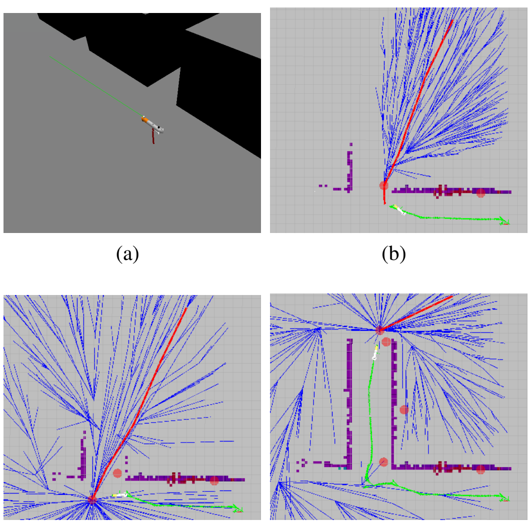
In all cases, the proposed framework succeeded in solving start-to-goal queries, which were defined in way that the vehicle had to move through the four-meter gap between the blocks. The main difference in each case lies on the environment information gathered by the sensors. In the case of the mechanically-scanning profiler and multibeam sonars, the replanning maneuvers are less than in the case of using echosounders, since a more accurate representation of the environment can be achieved faster.

Finally, we conducted in-water trials to validate our approach in a real-world scenario. The SPARUS-II performed different autonomous missions at a constant surge speed $u = 0.5m/s$ and a maximum turning rate $r_{max} = 0.3rad/s$ in a mission that is equivalent to the one defined in the simulated environment (see Fig 12).



(a) (b) (c) (d)

Fig. 9: Planning with echosounders.



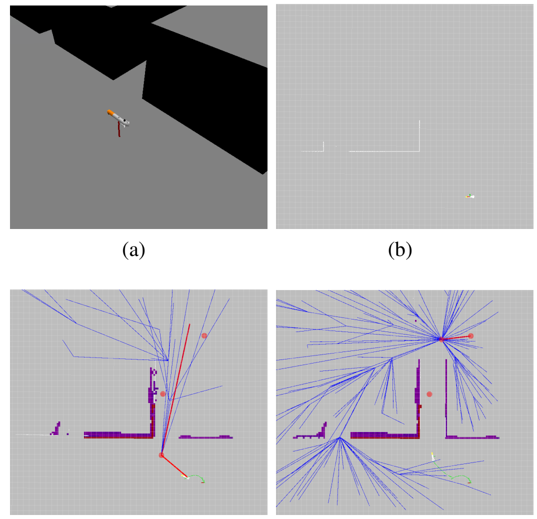
(a) (b) (c) (d)

Fig. 10: Planning with profiler.

IV. CONCLUSIONS AND FURTHER WORK

We presented a framework for mapping and planning online collision-free paths for AUVs in an unknown environment. The framework gathers and processes data obtained from range sensing sonars to represent the environment. Simulation and real-world results not only validated our approach, but also exposed different alternatives for future work.

Replanning maneuvers can be considerably reduced considering the vehicle's motion constraints. Working along this line, we plan to implement a different controller that allows the AUV to follow resulting trajectories generated by a planner with motion constraints. Finally, we aim to extend our approach to 3D motion and to consider external sources of uncertainty such as ocean currents.



(a) (b) (c) (d)

Fig. 11: Planning with multibeam.

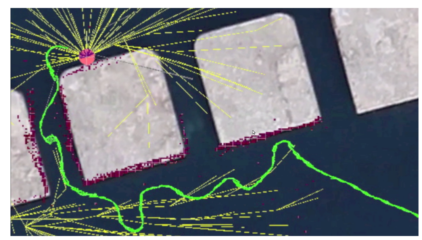


Fig. 12: Real world result of the SPARUS-II AUV solving and executing a start-to-goal query. Result is overlapping a satellite image of the test scenario.

REFERENCES

[1] E. Galceran, R. Campos, N. Palomeras, M. Carreras, and P. Ridao, "Coverage path planning with realtime replanning for inspection of 3D underwater structures," in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6586–6591, 2014.

[2] A. Mallios, P. Ridao, M. Carreras, and E. Hernàndez, "Navigating and mapping with the SPARUS AUV in a natural and unstructured underwater environment," in *MTS/IEEE OCEANS*, (Waikoloa), 2011.

[3] S. M. LaValle, *Planning Algorithms*. Cambridge: Cambridge University Press, 2006.

[4] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, Dec. 1959.

[5] S. M. LaValle and J. J. Kuffner, "Randomized Kinodynamic Planning," *The International Journal of Robotics Research*, vol. 20, pp. 378–400, May 2001.

[6] S. Karaman and E. Frazzoli, "Sampling-based Algorithms for Optimal Motion Planning," *The International Journal of Robotics Research*, vol. 30, pp. 846–894, June 2011.

[7] M. Likhachev, D. Ferguson, G. Gordon, A. Stentz, and S. Thrun, "Anytime Dynamic A*: An Anytime, Replanning Algorithm.," in *International Conference on Automated Planning and Scheduling (ICAPS)*, 2005.

[8] K. Belghith, F. Kabanza, L. Hartman, and R. Nkambou, "Anytime dynamic path-planning with flexible probabilistic roadmaps," in *IEEE International Conference on Robotics and Automation (ICRA)*, no. May, pp. 2372–2377, 2006.

[9] M. Likhachev and D. Ferguson, "Planning Long Dynamically Feasible Maneuvers for Autonomous Vehicles," *The International Journal of Robotics Research*, vol. 28, pp. 933–945, June 2009.

[10] Y. Kuwata, S. Karaman, J. Teo, E. Frazzoli, J. P. How, and G. Fiore, "Real-Time Motion Planning With Applications to Autonomous Urban

Driving," *IEEE Transactions on Control Systems Technology*, vol. 17, pp. 1105–1118, Sept. 2009.

[11] D. Dolgov, S. Thrun, M. Montemerlo, and J. Diebel, "Path Planning for Autonomous Vehicles in Unknown Semi-structured Environments," *The International Journal of Robotics Research*, vol. 29, pp. 485–501, Jan. 2010.

[12] C. W. Warren, "A technique for autonomous underwater vehicle route planning," in *Symposium on Autonomous Underwater Vehicle Technology*, vol. 15, pp. 201–205, 1990.

[13] K. P. Carroll, S. R. McClaran, E. L. Nelson, D. M. Barnett, D. K. Friesen, and G. N. William, "AUV path planning: an A* approach to path planning with consideration of variable vehicle speeds and multiple, overlapping, time-dependent exclusion zones," in *Symposium on Autonomous Underwater Vehicle Technology*, pp. 79–84, 1992.

[14] A. Alvarez, A. Caiti, and R. Onken, "Evolutionary Path Planning for Autonomous Underwater Vehicles in a Variable Ocean," *IEEE Journal of Oceanic Engineering*, vol. 29, pp. 418–429, Apr. 2004.

[15] C. Petres, Y. Pailhas, P. Patron, Y. Petillot, J. Evans, and D. Lane, "Path Planning for Autonomous Underwater Vehicles," *IEEE Transactions on Robotics*, vol. 23, pp. 331–341, Apr. 2007.

[16] Y. Petillot, I. T. Ruiz, and D. M. Lane, "Underwater vehicle obstacle avoidance and path planning using a multi-beam forward looking sonar," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 2, pp. 240–251, 2001.

[17] T. Maki, H. Mizushima, H. Kondo, T. Ura, T. Sakamaki, and M. Yanagisawa, "Real time path-planning of an AUV based on characteristics of passive acoustic landmarks for visual mapping of shallow vent fields," in *MTS/IEEE OCEANS*, (Vancouver), 2007.

[18] M. Carreras, C. Candela, and D. Ribas, "Sparus II, design of a lightweight hovering AUV," in *5th International Workshop on Marine Technology (MARTECH)*, pp. 163–164, 2013.

[19] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: an efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, pp. 189–206, Feb. 2013.

[20] M. Prats, J. Perez, J. J. Fernandez, and P. J. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2577–2582, Oct. 2012.