Hexagonal Grid Graph as a Basis for Adaptive Sampling of Ocean Gradients using AUVs

Tore Mo-Bjørkelund[1*], Trygve O. Fossum[1], Petter Norgren[1], and Martin Ludvigsen[1,2,3]
[1]Dept. of Marine Technology, Norwegian Univ. of Sci. and Tech. (NTNU), Trondheim, Norway
[2]Center for Autonomous Marine Operations and Systems (AMOS), NTNU, Trondheim, Norway
[3]Arctic Technology Department, University Centre in Svalbard (UNIS), Longyearbyen, Norway

Global Oceans 2020: Singapore - U.S. Gulf Coast | 978-1-7281-5446-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/IEEECONF38699.2020.9389324

***Abstract*—In this paper we present an adaptive sampling strategy for Autonomous Underwater Vehicles (AUVs) seeking to explore and map ocean gradients using a hexagonal grid as its potential path. The laborious and deterministic method of pre-programming trajectories has gradually been replaced by intelligent and adaptive sensing strategies. These both allows more effective use of sampling resources, as well as improved spatiotemporal resolution. We present a method for adaptive and data-driven sampling in the water-column with AUVs that follows the sense-plan-act control paradigm. The method uses a hexagonal grid graph to discretize the survey area into a honeycomb-pattern, which gives both equilateral survey paths and a small branching factor. Using this graph, we present results from a typical environmental sampling application where the goal is to search for and explore temperature gradients. The main use case for this algorithm is to prioritize the strongest gradients within the AUV spatiotemporal envelope. We present results from field trials in Trondheimsfjorden, Norway, where the AUV successfully explored a river front. We find that the algorithm performs as expected, exploring the area and revisiting sections containing gradients.**

***Index Terms*—AUV, adaptive sampling, graph, gradient, path planning.**

I. Introduction

Adaptive behaviour in underwater vehicles has been a field of study steadily increasing in the past decades. The Teleo-Reactive Executor [1], [2] used adaptive behaviour and modelling to identify and sample from stratified, upwelling front and upwelled waters. In [3] they used adaptive behaviour to keep an AUV along the isotherm associated with the Deepsea Chlorophyll-a max, acting as a pseudo-Lagrangian drifter. The authors of [4] present the use of multiple AUVs, informed by aerial drones and remote sensing to track and map the outline of the Douro river plume outside Porto, Portugal.

Path planning algorithms are often limited by an exponential combinatoric increase in dimensionality [5]. Therefore, the problems are often simplified using heuristics and/or greedy algorithms [6]. In underwater adaptive sampling, Gaussian process (GP) modeling [7] has shown to be a suitable tool for modelling the marine environment [8], [9] at high spatiotemporal resolution over submesoscale volumes. Computational complexity grows with model resolution and complexity of the model, in addition, one needs to take into account computational requirements of the path planning algorithm, of which the modelling is only one part. Two suggested solutions for this are anytime algorithms, as presented in [9] and constraining the number of options the algorithm has to evaluate. With limited battery capacity and operating speeds greater than gliders and Remotely Operated Vehicles (ROVs), the AUV is suited for resolving ocean phenomena such as fronts and gradients. These phenomena are often difficult to locate exactly, and a preprogrammed mission would most likely capture more redundant and less interesting data than an adaptive mission where the vehicle can plan its own path. Gradients in the ocean is a feature that is well suited for adaptive mission as they are limited in their extent and dynamic in location and strength. A gradient in the ocean can be a change in the value or distribution of one or more ocean parameter, such as temperature or salinity.

In this paper we present an adaptive sampling strategy for AUVs seeking to explore and map ocean gradients using a hexagonal grid as its potential path. In Section II we present the underlying assumptions, algorithm and experimental setup. Section III outlines the results from the last field trial in the early summer of 2020. A discussion of the method and results is presented in Section IV, while the conclusion and proposed further work is presented in Section V.

II. Method

In this section we present the hexagonal grid graph and natural cubic spline approach to adaptive sampling. We start by presenting the assumptions, continue to present the graph and its properties, and finally we present the algorithm.

A. Assumptions

The adaptive sampling algorithm presented in this paper is based on the following assumptions:

1. The ocean is stratified, and this means the horizontal correlation of ocean parameters are much greater than vertical correlation [10].
2. The AUV speed is greater than the process speed.
3. It is possible to represent an ocean parameter profile with a Natural Cubic Spline (NCS) [11].

Assumption 1 makes it possible for us to regard one dive and ascent with a pitch-constrained vehicle, such as an AUV, as one vertical measurement profile. Additionally, the use of both

The research was funded by the Research Council of Norway through the Nansen Legacy project RCN #276730 and the Center of Excellence Autonomous Marine Operations and Systems (AMOS), RCN #223254.

[*]Corresponding author email: tore.mo-bjorkelund@ntnu.no

978-1-7281-5446-6/20/$31.00 ©2020 IEEE

the descent and ascent profiles enables us to reduce the number of NCS regressions in the model, and thus the computational complexity. Furthermore, if there is a gradient within the envelope of an edge, it will be reflected in the Mean Squared Error (MSE) of the NCS regression. Assumption 2 enables us to regard events such as currents and tides as noise in our model. These effects are accounted for by adding uncertainty to old measurements. Assumption 3 rests on our ability to choose a sufficient number of knots in the regression of the NCS, as a NCS with sufficient number of knots will interpolate all measured points. Our intention is to have a NCS that is as simple as possible without sacrificing too much resolution. For this reason, 6 was chosen as a suitable number of knots.

B. Edge

Each edge, or straight line, of the graph holds the following information:

- Its two endpoints, or nodes, and their position.
- Measured data, if the edge has been explored.
- Assimilated data in the form of a NCS, if the edge has been explored.
- An estimate of the variance of the NCS, or an estimate of the variance of interpolated data.

The uninitialized edges are treated as empty before they are visited and edges that are visited multiple times only use measurements from the last visit. Since empty edges are initialized with a high estimated variance, MSE, they are likely to be visited at least once. It is our intention that this mechanism will drive the algorithm to explore all edges within reasonable time.

C. Graph

We seek to reduce the complexity of the path finding problem by discretizing the survey area using a graph of hexagonal tiles in $\mathbb{R}^2$ space. Given the problem of having computational time bounded by $\mathcal{O}(m^n)$, where m is the branching factor and n is the search depth, we seek to maximize n while still being able to survey an area in $\mathbb{R}^2$ with a reasonable computational complexity. One solution is to use the 2D convex regular polygon which provides a graph with a small branching factor, the hexagon. Higher branching factor polygons, such as squares or triangles, would lead the algorithm to be more greedy, but also to have a higher spatial resolution for the same edge length. The effect is visible in Figure 1. One further advantage of using a grid of regular polygons is that the edges are equilateral. This circumvents the need for tuning a length penalty of a potential path, since all paths will be equally long. In traditional path planning, the goal is often to get from point A to point B, choosing the shortest path. Our goal is not to achieve the shortest path or to move from A to B, but to choose the path that maximizes the gradients traversed and uncertainty reduced along the path.

D. Path Planning Algorithm

1) Initialization: The graph is initialized by choosing an origin, the width and height of the survey area, the max depth of the yoyo maneuver, and the orientation of the area. From these input variables, the parameters of the hexagonal grid are calculated. Edge length is calculated by

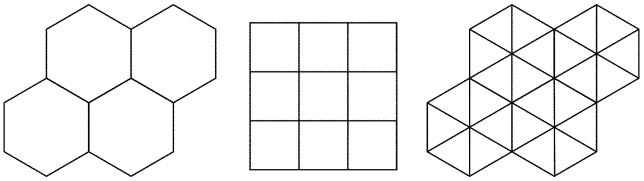

Fig. 1. Grids of the convex regular polygons that can homogeneously fill $\mathbb{R}^2$ space, hexagons, squares and triangles. All polygons have the same edge length.

$$l_e = l_p + 2 \cdot \frac{d}{tan(\theta)}[m] \tag{1}$$

where l_p is the padding length, d is the depth, and θ is the pitch angle. The number of tiles in each direction is calculated by

$$N_k = \left\lfloor \frac{L_k}{C_k \cdot l_e} \right\rfloor \tag{2}$$

where C_k is the width or height of a hexagonal tile in relation unit edge length, and L_k is the width or height of the survey area for $k \in [width, height]$, with $C_{width} = 1.5$ and $C_{height} = \sqrt{3}$.

The world model consists of vertical NCS regression models of ocean parameters on each edge and the predicted value at any point within the operational area can be calculated. This is done by GP modelling, using the NCS value along with their estimated variances at the desired depths as inputs to a spatial GP. The world model does not exist until the first measurement is made, i.e. the first edge is explored. In the beginning, there is no model, only a variance estimate on each of the edges.

2) Sensing Step: When arriving at a node, the sensing platform prioritizes the edges of the graph and chooses one. Since the length of an edge is calculated to contain one dive and ascent, the AUV will try to dive to the desired depth halfway between the nodes, and surfacing again at the target node. When the target node is reached, the vertical profile for that edge is generated and a NCS regression profile along the z-dimension (depth) is made.

3) Model Assimilation: The NCS is associated to the respective edge is a model for that edge together with the MSE of the regression. The variance of the edge is estimated as $s^2 = MSE$, and the variance of all edges is updated based on the variance of the current edge and the spatial covariance. In addition, we increase the MSE score of each edge every iteration to account for effects of current, waves, and other noise. The estimated variance of each edge is updated by

$$s_t^k = \alpha \cdot s_{max} + (1 - \alpha) \cdot s_{t-1}^k \tag{3}$$

where $\alpha \in [0, 1]$, and s_{max} is the maximum variance.

4) Planning Step: After the model for the current edge is updated, a search along the graph is performed. The best path, subject to a constrain on maximum path length, is found.

The minimum path length is limited to two edges, while the maximum path length is constrained by the computing time on board the autonomous platform. To account for this, a breadth first search is done over all possible paths up to and including the set maximum path length. The evaluation function is defined as

$$S(p) = \sum_{i=1}^{N-1} \frac{(m_e^i - m_e^{i+1})^2}{N-1} + \sum_{i=1}^{N} \sum_{j=1}^{M} MSE_i \cdot e^{-d_{ij} \cdot I} \quad (4)$$

where m_e^i is the NCS model associated with edge i, d_{ij} is the euclidean distance between edge i and j. I is the horizontal spatial autocorrelation [12], p is the path of length N, and M is the total number of edges in the graph. In (4), N and M are constrained by computing power, a low branching factor enables us to keep M low and thus extend the path lengths that we are able to evaluate. The best path is then found by finding the path p associated with the maximum value of the path score, $S(p)$, from (4).

E. Experimental Setup

1) Simulation: For the simulation setup, a simple Conductivity and Temperature (CT) simulator was developed. It takes the position and depth of the simulated vehicle as input, and outputs values for temperature and conductivity with added Gaussian noise. The simulator enabled testing over a variety of different scenarios. Some base scenarios used in combination in testing the algorithm:

- Stratified and mixed water with no horizontal gradient.
- Varying temperature gradients, $\Delta T \in [0, 2.0]°C$.
- Gradient fronts with varying extents, from $0m$ to $500m$.

Temperature was chosen as the parameter most suited for adaption, as it varies across the entire ocean and is a readily available measurement onboard most AUVs. The simulator enables testing and tuning of the algorithm before the field trials.

2) Field trials: For field trial testing of the algorithm, an area containing horizontal gradients is used. Such areas can be river mouths, upwelling fronts, and fronts between large ocean currents, such as the Arctic front in the Barents sea. The algorithm was implemented using the Robot Operating System (ROS) [13] and a bridge to the onboard control system of the vehicle. The algorithm was successfully implemented on a Light AUV (LAUV) [14]. The vehicle used was LAUV Harald from AURLab at NTNU, Trondheim, Norway, and the algorithm, along with ROS and a bridge to the onboard control system, Dune: Unified Navigation Environment (DUNE) [15], was implemented on the onboard auxiliary CPU, a Jetson TX1 from Nvidia. A max search depth, thus path length, of $N = 6$ was chosen for the experiment, furthermore the top $0.5m$ were discarded in case the Conductivity Temperature Depth (CTD) would be measuring in air so close to the surface.

III. Results

A. Simulation

Simulation results show that the algorithm is able to prioritize paths that have high temperature gradients. Depending on the initialization of the estimated variance of the edges, the rate of variance increase and the amplitude of the gradients observed, the algorithm can be tuned to be more exploratory or exploitative. Where the exploratory tuning leads to more even area coverage, the exploitative tuning will lead to a more concentrated mapping of the edges with the strongest inter-edge gradients. The higher the ratio of variance to gradient, the more exploration will be weighted.

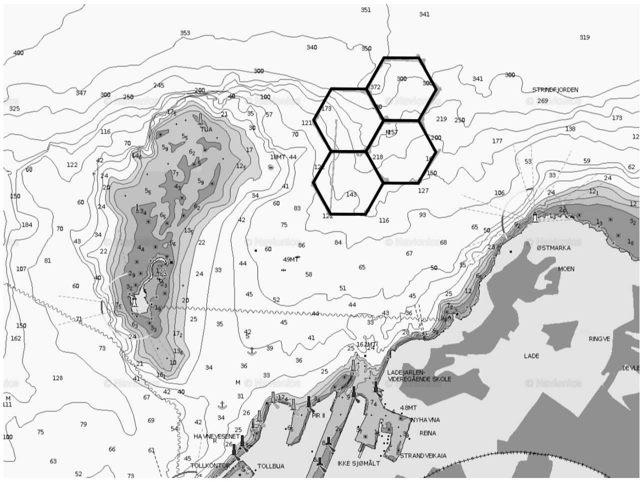

Fig. 2. Operational area, indicated by hexagonal tiles, for final field trials in Trondheimsfjorden outside Trondheim, Norway. The river outlet of the Nidelva river is visible in the lower center.

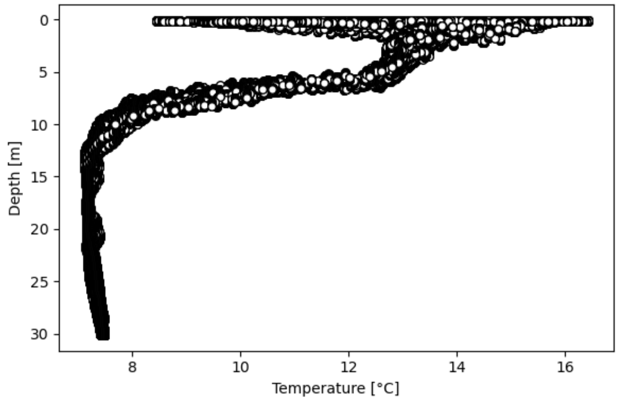

Fig. 3. Temperature profile over the entire mission.

B. Field Trials

Final field trials were carried out in Trondheimsfjorden, Norway on the 11th of June 2020. The adaptive mission ran from 12:40 to 15:45 local time, covering a grid of four hexagonal tiles covering an area of about $1km^2$ and exploring the upper water column between $0m$ and $30m$ depth. The area covered, indicated in Figure 2, was chosen to be close enough to the river mouth of Nidelva as to provide a temperature gradient for the vehicle to explore, while still being relatively safe from traffic in the area.

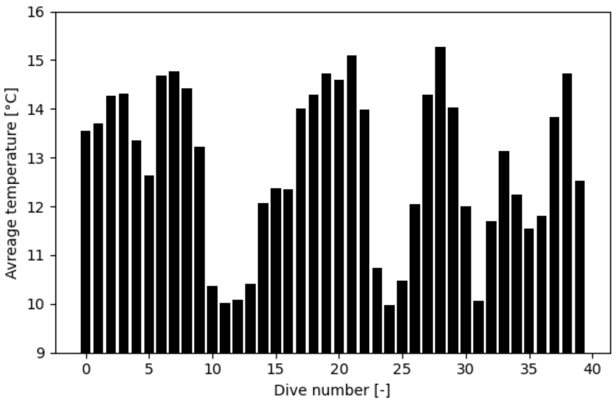

Fig. 4. Average temperature of each of the dives performed during the mission in consecutive order.

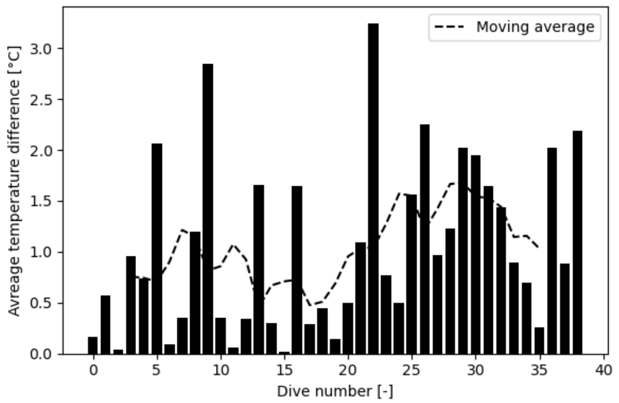

Fig. 5. Absolute value of difference of average temperatures of each of the 40 dives performed during the mission in consecutive order. Dashed line indicates a 6-unit wide (reflecting the search depth) rolling average.

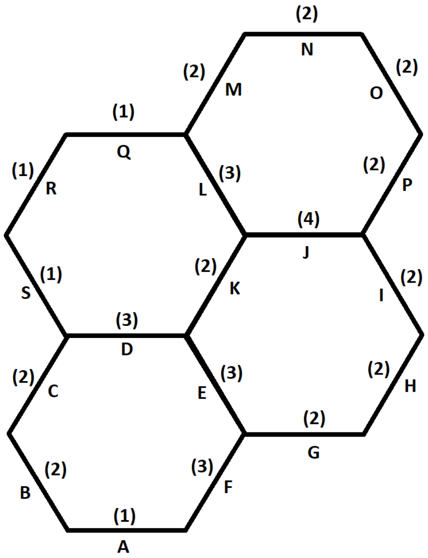

Fig. 6. Naming and numbering of the edges in the hexagonal grid, name in capital letters, and the number of edge visits by the AUV in parenthesis (n).

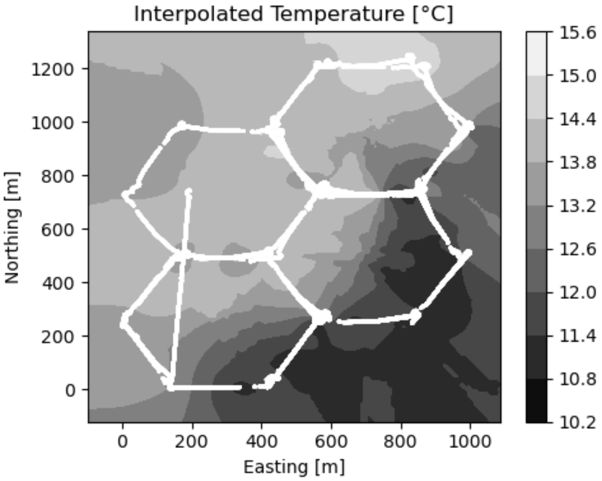

Fig. 7. An interpolated map of the temperature in the operational area between $0.5m$ and $2.0m$ based on the average temperatures from each ascent and descent along with the AUV path.

The temperature profile for the entire mission is presented in Figure 3, here, the lateral variance seems to be most prominent in the upper $5m$. Further, we present an interpolated plot of the lateral temperature distribution in Figure 7.

The AUV successfully explored all of the 19 edges in the graph, and displayed the expected behavior: first exploring, then choosing paths with strong inter-edge gradients. This behavior can be shown by plotting the average temperature for each edge in the order they were visited, as shown in Figure 4. Plotting the absolute value of the difference of the average temperature of consecutive edges in Figure 5 gives an indication that edges with stronger inter-edge gradients were favored during the last half of the mission. It is worth noting that the averages are taken over the entire dive, as well as time spent in the surface and that the data is averaged over time, not depth. Thus, these averages are skewed towards the surface temperature.

The edges were visited in the following order, using Figure 6 as legend: B, C, D, K, J, J, L, M, N, O, P, I, H, G, F, F, E, D, S, R, Q, L, J, I, H, G, E, D, C, B, A, F, E, K, J, P, O, N, M, L. By comparing the temperature field in Figure 7 and the number of edge visits in Figure 6 in we see that edges with stronger gradients have a tendency to be visited more often. This is visible by looking at the position of edges that were visited once (Q, R and S) in Figure 7, this area is relatively homogeneous. Edge J was visited 4 times, this area has a strong gradient showing in Figure 7. Edge A was visited once as well, despite it apparently sitting atop a strong gradient. On average internal edges (edges that are within the outline of the graph) were visited more often than the external edges (edges along the border of the graph) with 1.786 and 3.0 average respectively.

IV. Discussion

Presenting the time averaged data for each dive enables us to emphasize the gradient found at the surface. Since there is not a great variation below $5m$, as seen in Figure 3, we chose to present the data this way. This reveals a weakness in the approach presented in the paper, the vehicle does not adapt its depth as a response to either horizontal or vertical variance, thus more time is spent exploring homogeneous areas. In this case, it was the depths below approximately $12m$, as seen in Figure 3. As the onboard CPU performed its task without fail, there is room for trying other and more complex evaluation functions, search depths or geometries.

Looking at the number of visits for each edge, there is a discrepancy between the number of visits to internal and external edges. We believe this stems from the second term in (4), where internal edges, being closer to all other edges than external edges, score higher for the same MSE.

V. Conclusion

The vehicle and algorithm performed its task of mapping the operational area with more emphasis on areas with strong gradients. Using an hexagonal grid, we can increase the search depth N, constrained by the computing power of the onboard computer. Thus enabling search for gradients along large parts of the graph. Future work includes adding a more refined model, multivariate gradient tracking, and exploring other objective functions.

References

[1] Y. Zhang, J. P. Ryan, J. G. Bellingham, J. B. Harvey, and R. S. McEwen, "Autonomous detection and sampling of water types and fronts in a coastal upwelling system by an autonomous underwater vehicle," *Limnology and Oceanography: Methods*, vol. 10, no. 11, pp. 934–951, 2012.

[2] Y. Zhang, J. G. Bellingham, J. P. Ryan, B. Kieft, and M. J. Stanway, "Two-dimensional mapping and tracking of a coastal upwelling front by an autonomous underwater vehicle," in *2013 OCEANS-San Diego*. IEEE, 2013, pp. 1–4.

[3] Y. Zhang, B. Kieft, B. W. Hobson, J. P. Ryan, B. Barone, C. M. Preston, B. Roman, B.-Y. Raanan, R. Marin III, T. C. O'Reilly *et al.*, "Autonomous tracking and sampling of the deep chlorophyll maximum layer in an open-ocean eddy by a long-range autonomous underwater vehicle," *IEEE Journal of Oceanic Engineering*, 2019.

[4] J. Pinto, R. Mendes, J. C. da Silva, J. M. Dias, and J. B. de Sousa, "Multiple autonomous vehicles applied to plume detection and tracking," in *2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*. IEEE, 2018, pp. 1–6.

[5] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*. MIT press, 2009.

[6] T. O. Fossum, "Adaptive sampling for marine robotics," Ph.D. dissertation, Norwegian University of Science and Technology, 2019.

[7] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*. Springer, 2003, pp. 63–71.

[8] T. O. Fossum, J. Eidsvik, I. Ellingsen, M. O. Alver, G. M. Fragoso, G. Johnsen, R. Mendes, M. Ludvigsen, and K. Rajan, "Information-driven robotic sampling in the coastal ocean," *Journal of Field Robotics*, vol. 35, no. 7, pp. 1101–1121, 2018.

[9] S. Kemna, "Multi-robot strategies for adaptive sampling with autonomous underwater vehicles," Ph.D. dissertation, thesis, University of Southern California, 2018.

[10] R. H. Stewart, *Introduction to physical oceanography*. Texas A & M University College Station, 2008.

[11] J. H. Ahlberg, E. N. Nilson, and J. L. Walsh, "The theory of splines and their applications," *Mathematics in Science and Engineering, New York: Academic Press, 1967*, 1967.

[12] J. K. Ord and A. Getis, "Local spatial autocorrelation statistics: distributional issues and an application," *Geographical analysis*, vol. 27, no. 4, pp. 286–306, 1995.

[13] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[14] A. Sousa, L. Madureira, J. Coelho, J. Pinto, J. Pereira, J. B. Sousa, and P. Dias, "Lauv: The man-portable autonomous underwater vehicle," *IFAC Proceedings Volumes*, vol. 45, no. 5, pp. 268–274, 2012.

[15] J. Pinto, P. S. Dias, R. Martins, J. Fortuna, E. Marques, and J. Sousa, "The lsts toolchain for networked vehicle systems," in *2013 MTS/IEEE OCEANS-Bergen*. IEEE, 2013, pp. 1–9.