Efficient Seabed Coverage Path Planning for ASVs and AUVs*

Enric Galceran[1] and Marc Carreras[1]

Abstract— **Coverage path planning is the problem of moving an effector (e.g. a robot, a sensor) over all points in a given region. In marine robotics, a number of applications require to cover a region on the seafloor while navigating above it at a constant depth. This is the case of Autonomous Surface Vehicles, that always navigate at the water surface level, but also of several Autonomous Underwater Vehicle tasks as well. Most existing coverage algorithms sweep the free space in the target region using lawnmower-like back-and-forth motions, and the inter-lap spacing between these back-and-forth laps is determined by the robot's sensor coverage range. However, while covering the seafloor surface by navigating above it at a constant depth, the sensor's field of view varies depending on the seafloor height. Therefore, to ensure full coverage one would need to use the inter-lap spacing determined by the shallowest point on the target surface, resulting in undesired coverage overlapping among the back-and-forth laps. In this work, we propose a novel method to generate a coverage path that completely covers a surface of interest on the seafloor by navigating in a constant-depth plane above it. The proposed method uses environment information to minimize the coverage overlapping by segmenting the target surface in regions of similar depth features and addressing them as individual coverage path planning problems. A cell decomposition coverage method is applied to each region. The surface gradient is used to determine the best sweep orientation in each cell, and the inter-lap spacing in the lawnmower-like paths used to cover each cell is maximized on a lap-by-lap basis, hence obtaining a shorter, more efficient coverage path. The proposal is validated in simulation experiments conducted with a real-world bathymetric dataset that show a significant increase on path efficiency in comparison with a standard boustrophedon coverage path.**

I. INTRODUCTION

Coverage Path Planning (CPP) is the task of determining a path that passes over all points of an area of interest where there are obstacles that must be avoided. In marine robotics, numerous applications require to cover a region of interest on the seabed, e.g., seabed mapping, resource searching, seabed image mosaicing or mine counter-measures just to name a few. Usually, in these applications, *a priori* information (large-scale bathymetric* chart, size, location, etc.) about the target surface is available. Typically, the vehicle carries some down-looking sensor (e.g. a camera, a sonar) that can perceive the seafloor within a certain field of view (FOV).

*This research was sponsored by the Spanish government (DPI2011-27977-C03-02) and the TRIDENT EU FP7-Project under the Grant agreement No. ICT-248497

[1]E. Galceran and M. Carreras are with the Underwater Robotics Research Center (CIRS), University of Girona, Pic de Peguera s/n, 17071 Girona, Spain `enricgalceran at eia.udg.edu`, `marc.carreras at udg.edu`

*Bathymetry is the study of underwater depth of lake or ocean floors. In other words, bathymetry is the underwater equivalent to hypsometry (the measurement of land elevation relative to sea level).

Basically, a marine vehicle can carry out seafloor coverage by navigating in one of two different modes: at constant depth or at constant altitude from the seafloor. At constant altitude, as the vehicle navigates keeping a certain distance from the seafloor, typically the surface gradient is used to decide whether or not a rise of the ocean floor is steep enough to be considered an obstacle. In this case, as the distance to the seafloor is kept constant, the sensor's FOV over the seafloor does not vary. At constant depth, any single region of the seafloor intersecting the vehicle's horizontal plane is considered to be an obstacle. Also in this latter case, the sensor's FOV varies along the sea surface: the deepest the surface is, the wider the FOV becomes, and vice versa.

In fact, many marine robotics applications require the vehicle to cover a seabed area by navigating at a certain constant depth. This is the case of Autonomous Surface Vehicles (ASVs), that always navigate at the water surface level. Nonetheless, also several AUV tasks require to cover the seabed while navigating at constant depth. Indeed, AUVs with kinematic constraints such as limited heave (motion along the vertical axis) capabilities can also benefit from a constant-depth seabed coverage method.

There exist a number of proposed methods in the literature that tackle the generic CPP problem. The most successful proposals can be classified either as approximate or exact cellular decompositions.

In approximate cellular decomposition, the target region is approximated with a grid that covers the region, and the algorithm is applied to the grid. Gabriely and Rimon proposed the the Spiral-STC algorithm [1], which subdivides the workspace into discrete cells and then creates a spanning tree path induced by the cells. The robot is able to cover every point precisely once, and travel an optimal path in a grid-like representation in the workspace. Choi *et al.* proposed an on-line complete coverage path planning solution for mobile robots in which simple spiral paths are traced based on active wall-finding using the history of sensor readings and then linked using the inverse distance transform [2]. However, the problem with these methods is that they are resolution-complete, that is, their ability to achieve completeness (full coverage of the target space) rely on the grain size of the grid.

Exact cellular decomposition methods [3] take the following approach to generating a coverage path: the free space in the target region is divided into subregions called cells that are free of obstacles and hence can be covered by simple motions; then, a traveling-salesman algorithm is used to generate a sequence that visits each cell exactly once; finally, each cell is individually covered by using simple back-and-forth motions. Morse-based exact cellular decompositions are a reliable framework for generating coverage paths [4]. This method guarantees full coverage of the target space and also allows for using different coverage patterns. Moreover, it can be applied to any n-dimensional space.

Efficiency is very important in many coverage applications. Huang presented an optimal line-sweep based method for cellular decomposition algorithms [5]. This approach produces an optimal length coverage path by allowing different sweep directions in the lawnmower paths used to cover each cell in order to minimize the number of turns. However, very few CPP proposals in the literature address the cost of the path generated to cover the given area.

To sweep the area of interest, most existing coverage algorithms produce some sort of lawnmower or boustrophedon —the word coming from Greek, "the way an ox drags a plough". When covering a planar or effectively planar surface, as in a floor cleaning task, the inter-lap spacing of the back-and-forth motions is determined by the robot's sensor coverage range. However, when covering a variable-height surface like the seafloor while navigating above it at constant depth, the sensor's FOV changes depending on the height of the bottom surface. Therefore, to ensure complete coverage of the target area, the inter-lap spacing allowed by the shallowest depth (i.e. the maximum height) in the target surface must be used, producing an undesired, considerable amount of sensor coverage overlapping among the path laps.

Up to date, no research has addressed this issue. It is also worth noticing that most of the research in CPP has been intended for mobile robotics applications, while very few has addressed the particularities of marine environments. In this work, we propose an extension to the cellular decomposition approach to generate a coverage path that completely covers a 2.5D surface of interest on the seafloor by navigating in a constant-depth plane above it. The proposed method minimizes the coverage overlapping by segmenting the target surface into regions of similar depth features and addressing them as individual CPP problems. A cell decomposition coverage method is applied to each region. The surface gradient is used to determine the best sweep orientation in each cell, and the inter-lap spacing in the boustrophedon paths used to cover each cell is maximized on a lap-by-lap basis, hence obtaining a shorter, more efficient coverage path. The proposal is validated in simulation experiments conducted with a real-world bathymetric dataset that show a significant increase on path efficiency.

The remaining sections in this paper are organized as follows. Sec. II discusses the Morse-based cellular decomposition for planar surfaces that will be used by our method to ensure complete coverage. Next, Sec. III describes the proposed seabed coverage method. In Sec. IV we validate the proposed method with simulation experiments conducted with a real-world dataset. Finally, concluding remarks are given in Sec. V.

II. MORSE-BASED CELLULAR DECOMPOSITION COVERAGE

In this work, we use the off-line Morse-based cellular decomposition method to achieve complete coverage [4]. This method, as an exact cellular decomposition method, divides the robot's free space into simple regions (cells) such that the union of the regions fills the free space. Two cells are adjacent if they share a common boundary. An adjacency graph is used to encode the cell decomposition, where a node represents a cell and an edge represents an adjacency relationship between two cells. Given that no obstacles lie inside a cell, to cover each cell is considered a trivial task, as it can be covered using simple motions, such as lawnmower-like back-and-forth motions. Therefore, finding a path that visits each cell once (i.e. finding an exhaustive walk through the adjacency graph) is equivalent to solving the coverage problem.

The Morse-based cell decomposition method uses critical points on the restriction of a Morse function to the obstacle boundaries to determine the cell decomposition. A Morse function is one whose critical points[†] are nondegenerate [6]. Practically speaking, this means that critical points are isolated. Changes on the connectivity of the free space (the space free of obstacles) occur at these critical points on the obstacle boundaries. Thus, critical points can be used to determine the cellular decomposition of the free space.

To determine the cell decomposition, a *slice* is swept through the target space. This slice is defined in terms of the preimage of a real-valued Morse function, $h\colon \mathcal{WS} \to \mathbb{R}$, where $\mathcal{WS}$ is the robot's workspace (i.e. the space that needs to be covered). Choosing different Morse functions produces different slice shapes and hence different cell decomposition patterns. For simplicity, we will describe the Morse-based boustrophedon decomposition [7], which happens in the plane.

In the boustrophedon decomposition, a vertical slice, defined in terms of the Morse function $h(x, y) = x$, is swept from left to right in the workspace. Thus, the vertical slice is determined by the preimage of this Morse function, $\mathcal{WS}_\lambda = h^{-1}(\lambda)$. Increasing the value of the slice parameter, λ, sweeps the slice from left to right through the workspace.

As the slice sweeps the space it intersects (or stops intersecting) obstacles, which sever it into smaller pieces as the slice first encounters an obstacle, that is, the connectivity of the slice in the free space increases. Also, immediately after the slice leaves an obstacle, smaller slice pieces are merged into larger pieces (the connectivity of the slice in the free space decreases). The points where these connectivity changes occur are the critical points. (Notice that critical points are always located on the obstacle boundaries.) Thus, at critical points, the slice is used to determine the cells in the decomposition. One can see that within a cell, the slice connectivity remains constant. Fig. 1 shows how, at the

[†]Recall that in the case of a function of a single real variable, $f(x)$, a critical point is a value x_0 in the domain of f where either the function is not differentiable or its derivative is 0, $f'(x_0) = 0$.

critical point, the connectivity of the slice changes from two to one, and hence two old cells are closed and a new cell is created.

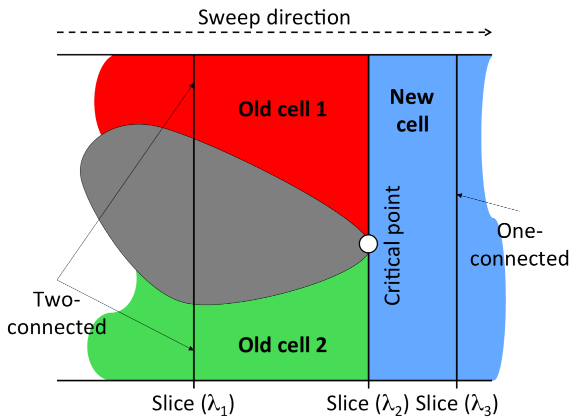


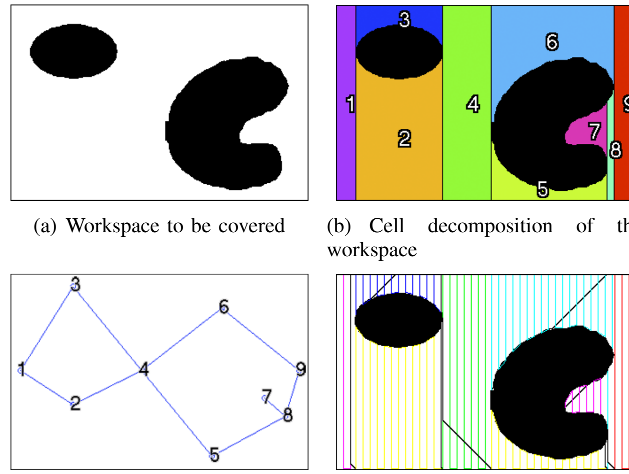
Fig. 1. Cell determination with the Morse-based boustrophedon cell decomposition method

Once the cell decomposition is constructed, an exhaustive walk through its associated adjacency graph is determined by the planner. Then, explicit boustrophedon paths are generated in each cell in the following way. First, the robot moves along the current slice until it encounters the cell boundary. Second, the robot moves orthogonally to the slice (i.e. following the sweep direction) for a distance equivalent to one robot sensor range. Then, the robot restarts the process by moving along the next slice until the cell is completely covered.

An example of the application of the Morse-based boustrophedon cell decomposition method is shown in Fig. 2. A target workspace is shown in Fig. 2(a) (free space is shown in white, obstacles in black). Fig. 2(b) shows its corresponding Morse-based boustrophedon cell decomposition. The cell decomposition is encoded by the adjacency graph shown in Fig. 2(c). Then, a traveling-salesman algorithm is used to determine an exhaustive walk that visits all the nodes (cells) in the graph. For the adjacency graph shown in Fig. 2(c), an exhaustive walk is 1, 2, 4, 3, 5, 8, 7, 9, 6. Finally, simple boustrophedon paths are generated inside each cell. These paths are linked in the order determined by the exhaustive walk to form the complete coverage path shown in Fig. 2(d). A simple start-to-goal path planner, like the Bug1 and Bug2 algorithms [3], can be used to determine the cell-to-cell paths.

III. PROPOSED COVERAGE PATH PLANNING METHOD

The overall objective of this work is to create a method that will generate an efficient path to cover a 2.5D surface of interest on the seafloor, i.e. a projectively planar surface, by means of an autonomous marine vehicle (namely an ASV or AUV) that navigates in a planar surface laying at a constant depth and above the surface of interest. We assume the vehicle carries a down-looking sensor with a certain FOV, able to image the seafloor from the navigating



(a) Workspace to be covered

(b) Cell decomposition of the workspace

(c) Cell adjacency graph

(d) Specific coverage paths in each cell linked to form the final coverage path (the diagonal paths are ”links between individual cell paths)

Fig. 2. Morse-based cell decomposition coverage path generation process for an example workspace

depth. Thus, a coverage path in the plane in which the vehicle navigates will be determined. Naturally, protruding parts of the underlying seafloor raising above the navigating plane represent obstacles that must be avoided.

The Morse-based boustrophedon cell decomposition method described in Sec. II will be used to ensure complete coverage of the target area. As described, in our scenario the robot's sensor FOV varies depending on the target surface height, as shown in Fig. 3.

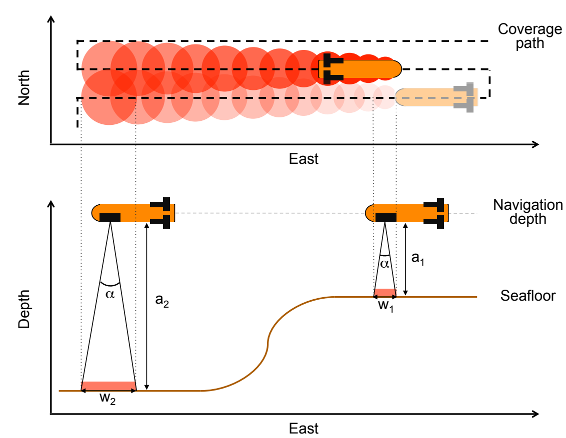


Fig. 3. Coverage overlapping. As the robot navigates from right to left with a sensor FOV angle α, vehicle altitude varies from a_1 to a_2. The FOV corresponding to a_2, w_2, is considerably wider than the FOV corresponding to a_1, w_1. As the inter-lap spacing is fixed, undesired coverage overlapping appears.

Thus, to ensure complete coverage of the target area, the constant inter-lap spacing among the back-and-forth laps used by the method to sweep the space must be the one allowed by the robot's sensor FOV on the shallowest depth (i.e. the maximum height) in the target surface. However, a lawnmower-like path with constant inter-lap spacing would produce a considerable undesired amount of coverage overhead, as shown in Fig. 3.

In this work, we seek to use the available environment information to produce an efficient coverage path by minimizing the coverage overlapping among the back-and-forth motions used to sweep the workspace. The strategy to achieve this relies in three key points:

- Target surface segmentation. The target surface on the seafloor is segmented in regions of similar depth features and each region is tackled as an individual CPP problem. As the inter-lap spacing is constrained by the surface's shallowest point, the similar-depth regions allow for minimizing the coverage overlap.
- Sweep orientation. The orientation of the sweeping path will be adapted to the seafloor surface gradient. By sweeping perpendicularly to the underlying surface gradient, the difference between the shallowest and the deepest point along a lap is minimized, and hence the coverage overlapping is minimized as well.
- Variable inter-lap spacing. The inter-lap spacing is maximized on a lap-by-lap basis by determining the spacing between the current lap and the next one with the shallowest surface point under the current lap.

The following sections will describe each step of our proposed algorithm. The algorithm description will be accompanied by figures illustrating the effect of every step using a real-world bathymetric dataset we introduce next.

A. Real-World Dataset

In this work, we use a real-world dataset to demonstrate the successive steps of our method. This dataset is a bathymetric map constructed with data recorded near the Formigues islands by members of the Underwater Robotics Research Center (CIRS) at the University of Girona in July 2009. The Formigues islands are an archipelago of sixteen little islets located about 1300 m off the Canet cape in the Costa Brava in Girona, Spain.

A 253 m width, 148 m height rectangular area was mapped, with its lower-left corner located at 41° 51' 34.35" N, 3° 10' 38.30" E. Depth in the mapped area ranges from about 5.5 m down to 16 m. The bathymetric data were obtained by means of a system composed of a down-looking multi-beam sonar, a motion reference unit and a GPS receiver. The whole system was attached to a boat's hull. For the demonstration purposes of this paper, basic image processing techniques were used to filter out noise originally present in the map and missing data points were filled using neighboring pixels information. Fig. 4 shows the obtained bathymetric map.

B. Environment Modeling

The environment modeling is composed of two elements: the 2.5D surface of interest (namely the seabed) and the planar surface in which the vehicle navigates (the vehicle's workspace). We model the 2.5D surface of interest as a heighmap, $\mathcal{H}$, where the elevation values are negative downwards, as shown in Fig. 4. The planar navigation surface, i.e.

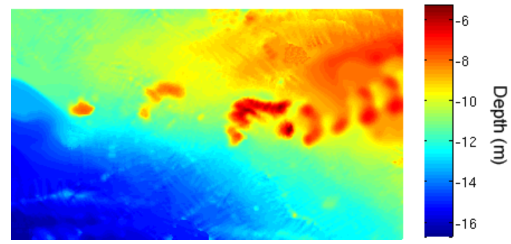


Fig. 4. Bathymetric map obtained near the Formigues islands

the vehicle's workspace, $\mathcal{WS}$, is modeled as an occupancy grid corresponding to a horizontal slice of the heightmap laying at the constant vehicle depth, D, such that for a location (x, y) in the heightmap

$$\mathcal{WS}(x,y) = \begin{cases} 1 & \text{if } \mathcal{H}(x,y) \geq D \\ 0 & \text{if } \mathcal{H}(x,y) < D \end{cases}$$

In other words, a point in the workspace is an obstacle (i.e. it is occupied) if the elevation of the surface of interest is above the vehicle's constant depth. Fig. 5 shows the obtained workspace by slicing the target surface at $D = -8.5$ m.

Fig. 5. Obtained planar workspace by slicing the target heightmap at depth $D = -8.5$ m

C. Vehicle Modeling

In this work, we modeled the vehicle as a point, i.e. having no volume and weight, so it is considered as a fully actuated vehicle. Nonetheless, the generated path could theoretically be followed by a vehicle with only the surge (motion along the X axis) and heading (rotation about the Z axis) degrees of freedom.

The vehicle has a down-looking sensor that covers a FOV beneath the vehicle determined by an angle of view, α. It is assumed that, from a given point in the planar navigation surface, the sensor range can reach the underlying surface of interest.

D. Surface Segmentation

The first step of the algorithm consists in segmenting the heightmap in n regions of similar depth. We use the well-known K-means clustering algorithm to obtain an initial segmentation. The number of regions, n, is a parameter provided by the user. Then, we post-process the initial segmentation using morphological operations (namely *dilate* and *erode* operations) to smooth the region borders and to ensure that the regions are simply connected. We do this because, rather than an accurate segmentation that would produce regions with abrupt borders hence difficult to cover, we foster the division of the terrain into smooth regions that allow to minimize the path's coverage overlapping. Smooth regions are obtained thanks to the dilate and erode operations. Fig. 6 shows the segmentation of the bathymetric map shown in Fig. 4 with $n = 3$ regions.

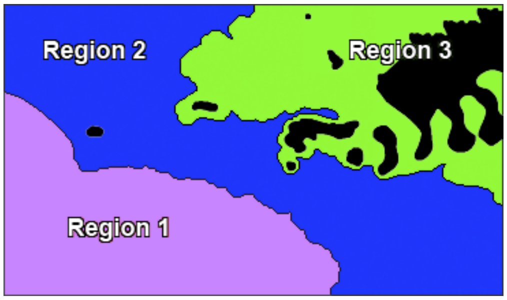


Fig. 6. Segmentation of the bathymetric map shown in Fig. 4 with $n = 3$ regions, with the workspace obstacles shown in black

E. Coverage Path Generation for Every Region

Once the surface segmentation is obtained, we actually tackle n different CPP problems, one for each region. As we mentioned, this contributes to minimize the coverage overlapping along the generated path. Then, what we do in this step is to plan a coverage path for every surface region obtained in the segmentation.

Each individual coverage path in a region is planned by adhering to the following steps. First, the Morse-based boustrophedon cell decomposition method discussed in Sec. II is applied to each region to obtain its cellular decomposition. Second, the sweep orientation in each cell is determined by the gradient of the underlying surface. Third, using the determined sweep directions, a boustrophedon path is generated to cover each cell where the inter-lap spacing is maximized on a lap-by-lap basis. Fourth and last, the individual paths in each region are concatenated to obtain the final coverage path.

1) Region Cell Decomposition: The cellular decomposition of each region is obtained by applying the Morse-based boustrophedon cell decomposition method. Fig. 7 shows the cellular decomposition for region 2 of the surface segmentation shown in Fig. 6. An exhaustive walk through the adjacency graph associated to the decomposition is computed. The exhaustive walk determines the order in which the cells are covered.

2) Sweep Orientation: Once the cell decomposition of the region is obtained, we compute the sweep orientation of the individual boustrophedon paths used to cover each cell to be perpendicular to the main seafloor surface gradient under the cell. We do this because navigating perpendicularly to the surface gradient allows for maximizing the inter-lap spacing, as steep surface ascends or descends under a lap are avoided and hence the difference between the lowest

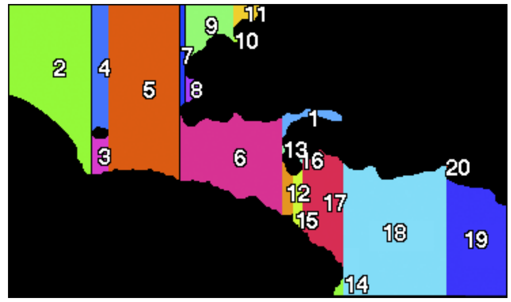


Fig. 7. Cell decomposition of region 2 of the segmentation shown in Fig. 6

and highest surface elevation under a lap is smaller. Thus, we compute the mean surface gradient under the cell as a measure of its main incline orientation. Then, we make the laps perpendicular to the angle determined by the mean gradient.

3) Variable Inter-Lap Spacing Boustrophedon Paths: Next, we generate the individual boustrophedon paths to cover each cell where the inter-lap spacing varies according to the minimum distance to the bottom surface under each lap. That is, the spacing between the current lap and the next lap is determined by the highest point on the seafloor surface under the current lap, where the minimum sensor FOV width occurs. By adhering to this variable inter-lap spacing strategy we guarantee full coverage of the cell while minimizing the coverage overlapping among laps, hence obtaining a shorter, more efficient path. Fig. 8 shows the generated coverage paths for each cell in the decomposition shown in Fig. 7.

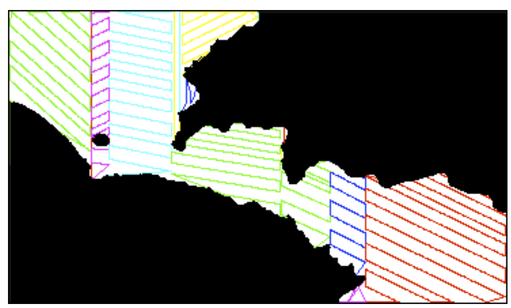


Fig. 8. Coverage paths for each cell of the cellular decomposition shown in Fig. 7

4) Final Coverage Path: The three steps we just described are applied to all the regions in the surface segmentation. Finally, we concatenate the individual coverage paths computed for each region in the surface segmentation. The well-known start-to-goal path planner A*[8] is used to compute a path going from the last point of a cell coverage path to the first point of the next cell's coverage path. Fig. 9 shows the generated final coverage path.

IV. RESULTS

To validate the method proposed in this work, we compare our adaptive inter-lap spacing approach to the Morse-based boustrophedon decomposition for planar spaces. It is worth

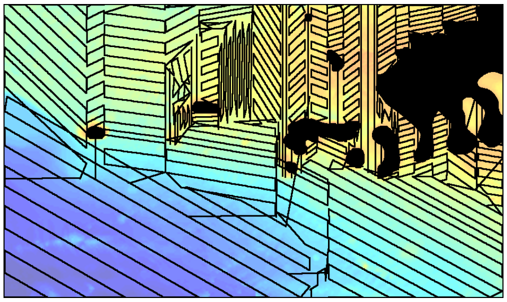


Fig. 9. Final coverage path of the workspace shown in Fig. 5 overlapped on the target bathymetric map

noticing that the latter method is intended for covering planar spaces by applying both methods to cover the seabed area near the Formigues islands described above in III-A. Seeking a fair comparison, we use the minimum possible sensor FOV width in each cell to determine the inter-lap spacing (that is, the FOV width when the vehicle is located over the highest point on the seafloor surface).

Our comparison is twofold. On one hand, we quantitatively measure both paths in terms of path length and running time. On the other hand, we provide coverage density maps generated by following the paths computed by both methods and taking into account the FOV of the vehicle's sensor in every point. In other words, a coverage density map shows, for a given path on the workspace, how many times has every point on the workspace been covered.

Table I shows the quantitative comparison of both methods. The vehicle navigates at depth $D = -8.5$ m and the target seafloor surface is segmented in $n = 3$ regions. The FOV angle is set to $\alpha = 50°$ and the vehicle speed is 2 m/s.

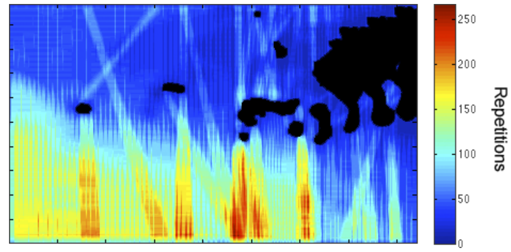
TABLE I

PATH LENGTH COMPARISON

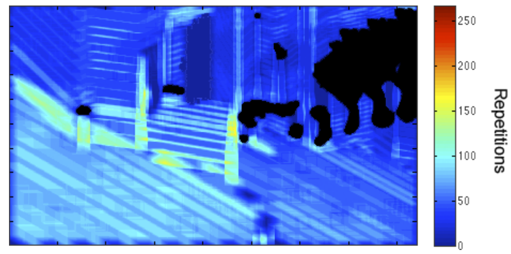
Method	**Path Length**
Constant inter-lap spacing	15846.08 m
Variable inter-lap spacing	10349.63 m

These results show that, for the real-world environment presented in this paper, our method performs approximately 34% better than the boustrophedon approach for planar spaces in terms of path length and running time.

Fig. 10 shows the comparison of the coverage density maps obtained using the "naive" boustrophedon decomposition method and the novel approach proposed in this paper. Ideally, each point in the target space should be covered once and hence be represented in a dark blue color. The coverage density map corresponding to the "naive" boustrophedon method (Fig. 10(a)) shows a high amount of overlapping, denoted by yellow and red colors. On the other hand, the coverage overlapping is highly reduced with our method, as denoted by the blue colors on Fig. 10(b).



(a) "Naive" boustrophedon method



(b) Our method

Fig. 10. Coverage density map comparison

V. CONCLUSIONS

A novel algorithm for coverage of seabed surfaces from an overlying planar surface has been presented. The proposed method minimizes the coverage overlapping by segmenting the target surface in regions of similar depth features and addressing them as individual CPP problems. A cell decomposition coverage method is applied to each region. The surface gradient is used to determine the best sweep orientation in each cell, and the inter-lap spacing in the boustrophedon paths used to cover each cell is maximized on a lap-by-lap basis, hence obtaining a shorter, more efficient coverage path. The validity of the algorithm has been demonstrated in simulation experiments conducted using a real-world bathymetric dataset, where a significant increase in the coverage path efficiency has been shown.

REFERENCES

[1] Y. Gabriely and E. Rimon, "Spiral-stc: an on-line coverage algorithm of grid environments by a mobile robot," in *Proc. IEEE Int. Conf. Robotics and Automation ICRA '02*, vol. 1, 2002, pp. 954–960.

[2] Y.-H. Choi, T.-K. Lee, S.-H. Baek, and S.-Y. Oh, "Online complete coverage path planning for mobile robots based on linked spiral paths using constrained inverse distance transform," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems IROS 2009*, 2009, pp. 5788–5793.

[3] H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementation*. The MIT Press, 2005.

[4] E. U. Acar, H. Choset, A. A. Rizzi, P. N. Atkar, and D. Hull, "Morse decompositions for coverage tasks," *International Journal of Robotics Research*, vol. 21, no. 4, pp. 331–344, 2002.

[5] W. H. Huang, "Optimal line-sweep-based decompositions for coverage algorithms," in *Proc. ICRA Robotics and Automation IEEE Int. Conf*, vol. 1, 2001, pp. 27–32.

[6] J. Milnor, *Morse Theory*. Princeton University Press, 1963.

[7] H. Choset, "Coverage of known spaces: the boustrophedon cellular decomposition," *Autonomous Robots*, vol. 9, no. 3, pp. 247–253, 2000.

[8] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson Education, 2003. [Online]. Available: http://portal.acm.org/citation.cfm?id=773294