Information Gain-guided Online Coverage Path Planning for Side-Scan Sonar Survey Missions

Nadir Kapetanović[1], Nikola Mišković[1], and Adnan Tahirović[2]

***Abstract*— Mapping an unknown large-scale marine area by a side-scan sonar onboard a marine vehicle as quickly as possible is often of great importance. It is also important that a-priori unknown interesting parts of the area are scanned in more detail, i.e. with the removal of sonic shadows. In contrast to the standard overlap-all-sonar-ranges lawnmower pattern, which is an offline static coverage problem solution for side-scan sonar missions, here a novel online side-scan sonar data-driven coverage solution is proposed. The proposed coverage algorithm provides a coverage solution based on local information gain from side-scan sonar data. At the same time, the solution is generated in such a way that coverage path length is minimized while covering the same area as the standard lawnmower. Upper and lower bounds of the proposed algorithm's improvement compared to the overlap-all-sonar-ranges lawnmower method are estimated analytically and validated through extensive mission parameters variation simulations. Simulation results show that our approach can cut down coverage path length significantly compared to the standard lawnmower method in most application cases.**

I. INTRODUCTION

Marine monitoring and exploration missions often include side-scan sonar based seafloor mapping. This includes the exploration of the biosphere, exploration of underwater archaeological sites, marine safety, and many other applications. Currently, side scan sonar missions are executed either by a tethered towfish equipped with a side-scan sonar [1], or by using remotely operated vehicles (ROVs) or autonomous underwater vehicles (AUVs) [2]. Deploying a tethered towfish from a boat requires hiring a boat, its crew, a towfish operator, a side-scan operator, and of course experts in the scientific field for which the survey is being conducted.

Towfish missions are most often executed in three phases. The first phase is a general survey mission at a higher altitude, whose goal is to identify possibly interesting parts of the region being covered [1]. In this phase, the boat tows a towfish in a lawnmower (LM) pattern to cover as much area as possible, without or with a low percent of overlaying side-scan sonar swaths. The second phase has a goal of getting detailed low altitude side-scan sonar images of the manually tagged possibly interesting areas from the first phase with side-scan sonar's swaths overlaid to remove any sonic shadows [1]. The third and final phase includes deploying and ROV and/or an AUV, and recording the identified interesting locations with a high-resolution camera, as proposed in [2]. On the other hand, it would be much more convenient and less costly to deploy an AUV and let it autonomously scan the given area, executing all three above mentioned area survey phases. It should gather more information about parts of the coverage area that it finds interesting for the current exploration mission, and lower resolution general survey data about other parts of the area.

Solutions to problems similar to the one mentioned above can be found in ground and aerial robotics literature, e.g. in [3], [4] and [5]. In marine robotics literature, one of the first papers to mention AUVs used for an online coverage path planning (CPP) is [6]. In [7] an online CPP algorithm generates an adaptive but constant-width LM pattern in order to remove the gaps in measurements. An online information/entropy based CPP approach for AUVs with side-scan sonar is proposed in [8] with application in target localization. It plans paths which reduce the expected entropy of the surrounding environment w.r.t. coverage path length and total turning angles. Target (hydrothermal vents) localization online adaptive CPP approach with temperature spreading model and multi-vehicle cooperation is presented in [9]. An online approach to mapping of ever-changing marine habitat by an AUV equipped with a camera and multi-beam sonar is given in [10]. It covers arbitrarily identified regions of interest (ROIs) from previous mapping missions in such a way to minimize repeated coverage. Another sonar-based survey path planning approach in [11] uses a-priori known bathymetry data of the coverage area in order to find salient points and cover them while minimizing vehicle position and sensing uncertainties. In [12] side-scan sonar data is improved for search missions through adapting the width of LM lanes w.r.t. pose estimation and sonar data uncertainty.

Our research goal is to develop an online side-scan sonar data-driven CPP algorithm for monitoring and surveying large-scale (over $1km^2$) seafloor regions by an AUV, which would replace all three above mentioned phases of survey missions when using a towfish. Lawnmower pattern is one of the most commonly used solutions for the 2D coverage of an area to be scanned [13], [14]. In marine robotics, it is typically used as a reference path for marine vehicles equipped with a side-scan sonar [1], or other sensors [14] for surveying missions.

*This research is sponsored by the Croatian Science Foundation Multi-Year Project under G. A. number IP-2016-06-2082, named *CroMarX*, NATO's Science for Peace and Security Programme as Multi-Year Project under G. A. number 984807, named *MORUS*, and EU Interreg Mediterranean project under ref. number $703/1MED15\_3.1\_M12\_282$, named *BLUEMED*.

[1]Nadir Kapetanović and Nikola Mišković are with the Faculty of Electrical Engineering and Computing, University of Zagreb, 10000 Zagreb, Croatia nadir.kapetanovic@fer.hr, nikola.miskovic@fer.hr

[2]Adnan Tahirović is with the Faculty of Electrical Engineering, University of Sarajevo, 71000 Sarajevo, Bosnia and Herzegovina adnan.tahirovic@etf.unsa.ba

The first step towards the above-mentioned research goal is to design an algorithm for a 2D coverage problem, which is a mix between the first and the second mission with towfish, and which is presented in this paper. The idea is to start the mission without overlapping sonar swaths, i.e. LM lanes twice wider than sonar range. During the mission, in case that the algorithm detects something interesting in the currently traversed LM line, coverage path planning algorithm replans the rest of the mission in order to sonify these interesting objects from the opposite side as well, thus removing their sonic shadows in side-scan sonar images.

Main contributions of this paper are: (1) an online side-scan sonar data-driven coverage path planning algorithm for large-scale unknown terrains presumably containing relatively few interesting parts, which covers the area at best with a twice shorter path, i.e. twice as fast as the statical overlap-all-sonar-swaths LM coverage maneuver, (2) analytical best- and worst-case guarantees validated through extensive mission parameters variation simulations, and (3) statistical performance analysis of the proposed CPP algorithm w.r.t. the coverage path length compared to the nonadaptive static solution of overlap-all-sonar-swaths LM coverage maneuver. Also, a function that estimates the proposed CPP algorithm's performance is proposed. It is designed based on the extensive statistical analysis of the proposed CPP algorithm's performance w.r.t. the coverage area topology and the expected chance of coming across interesting objects in the given area. This can be of good use to system engineers operating survey missions.

The rest of the paper is organized as follows: assumptions and problem definition are given in Section II. Section III describes the proposed algorithm, and models its best- and worst-case scenario performance. Simulation results are presented in Section IV. Key results of this paper and future work are concluded in Section V.

II. ASSUMPTIONS AND PROBLEM DEFINITION

A. Assumptions related to lawnmower pattern

Let us denote LM line length by $L[m]$, width between LM transects by $W[m]$, and a parameter which denotes the ratio between LM segment length and width, denoted by α_{lm}:

$$\alpha_{lm} = \frac{W}{L}. \tag{1}$$

Furthermore, parameter W depends on side-scan sonar's range $w_{sss}[m]$ which an AUV operator sets, i.e. $W = f(w_{sss})$. Generally, LM segments are set in such a way that $\alpha_{lm} \ll 1$ holds, i.e. LM segments are much more elongated than its segments are wide. Without the loss of generality, it is assumed that the LM pattern starts at its local frame origin, and spreads upwards and rightwards.

B. Assumptions related to AUV

Control side of mission execution is out of the scope of this paper and is presented in our previous work [15]. Altitude control is decoupled from $x-y$ position control. It is assumed that the external disturbance to the vehicle is either not present, or properly rejected by our model predictive (MPC) path following controller presented in [15]. Our CPP algorithm assumes the vehicle to be a point mass without any dynamic behavior, so control level has a task to follow paths which the CPP algorithm generates. It is also assumed that localization is either perfect or improved by communication of the AUV with an ASV as in [16], or a fixed beacon(s) as in [17]. Hence the approach presented in this paper does not minimize the number of turns during the mission, as does [18], in order to reduce dead reckoning error which increases the most during turning maneuvers.

It is also assumed that the blind spot of the side-scan sonar (*nadir*), is not present (which holds if nadir is covered with an additional multibeam sonar).

C. Assumptions related to sea floor configuration

The area which the vehicle should cover is assumed to be relatively sparsely populated with objects of interest for a given mission. In this paper information gain is defined as a measure of local side-scan sonar data variation, as in [19]. For each LM lane, a corresponding cost map is computed, based on which the proposed CPP algorithm makes decisions about possible replanning of the rest of the mission. Moreover, it is assumed that the sea floor is approximately flat. In case that the seafloor terrain is significantly slanted dominantly along one direction, then the lawnmower pattern mission should be oriented parallel to terrain's principal gradient. Since we are interested in large-scale open sea areas to survey with side-scan sonar, it is not assumed that obstacles are present in the environment. For an arbitrarily shaped 2D region to cover, it is possible to make a cell decomposition of the problem space, as is described in [13] and [14].

D. Problem definition

The problem addressed in this paper can be defined as: In order to cover a given unknown area of interest of dimensions $A \times B[m]$ with a given side-scan sonar at a constant surge speed u, and a constant altitude h_{ref}, generate an initial LM coverage pattern of length L and width $W = 2w_{sss}$. During the mission adapt the width of LM lanes as a multiple of sonar range w_{sss} depending on the detection of mission-specific interesting objects in sonar data. Here covering means getting the data from all informative objects (by sonifying them from both sides) as in case of the overlap-all-sonar-swaths LM pattern, while not sonifying uninteresting parts of the area from both sides.

III. BASIC ACCORDION COVERAGE PATH PLANNING ALGORITHM

In this Section, the proposed coverage algorithm which we named Basic Accordion Coverage Path Planner (BA-CPP) is described. It is a solution to the problem defined in Subsection II-D. The name "accordion" was inspired by the way this algorithm rearranges the LM pattern, expands and contracts it in some parts, just as the accordion bellows do while accordion is played. Also, from here on, the classical

overlap-all-sonar-swaths LM CPP method will be referred to as the CL-CPP algorithm.

A. Behavior of the proposed algorithm

The initial solution of the BA-CPP algorithm is an LM pattern with lanes twice as wide as the sonar range. This is, of course, a good solution only if there is nothing of interest for the mission in the area to be covered. In case that the vehicle detects some interesting object(s) in its sonar data while traversing the current LM line, the coverage path planning algorithm should replan the rest of the mission in such a way to sonify the interesting object(s) from the opposite side, if that has not been already done. Also, it should again (optimistically) assume that during the rest of the mission it will not encounter any interesting objects, and thus it will again generate the "stretched" LM pattern for the remaining part of the mission. Moreover, in the limiting case when the sea floor area of interest is densely covered with interesting objects, the proposed coverage path planning algorithm should behave as the CL-CPP.

Since the proposed CPP solution is a regular pattern, at least in the initial moment of the mission, elementary periods of the CL-CPP and the BA-CPP approach are given in Fig. 1h and 1a, respectively. This is needed to analytically predict the behavior of the CPP algorithm proposed in this paper on a spatial period of width $4W$, which ends with a horizontal line. This way every LM pattern spatial period starts with a vertical line and is of the same width. Heuristics behind the area covering of CL-CPP, ensure that complete coverage is achieved by the end of the mission, under the assumptions stated in Section II-D. This is why it is not insisted that the algorithm achieves a full coverage of the current pattern spatial period, but instead to achieve globally complete coverage in terms of acquiring all informative data present within the whole coverage area.

The characteristic behavior of the proposed algorithm in specific cases of interesting objects' distribution is formally described in Algorithm 1, and graphically represented in Fig. 1 (Case 0). If the vehicle encounters no interesting objects in the sonar image of the current LM line, which is the best-case scenario, it will follow the initial LM pattern, see Fig. 1a, line 1 in Algorithm 1. Method *add_case_0(pose, W, L, A)* generates LM waypoints (WPs) from the starting position *pose*, with LM lanes of width $2W$ and length L until LM lanes reach coverage area's width A.

If the vehicle detects something interesting to its right in the cost map while moving "downwards" along the line 3 as in Fig. 1b (Case 1), it will generate new waypoints in order to perform a loopback to its right side (LM lines 4, 5, and 6), and thus sonify interesting objects' sonic shadow(s) from the opposite side, given in lines 11 and 12 in Algorithm 1.

It is important to note that methods *add_case_1-6(pose, W, L, A)* generate LM waypoints (WPs) from the starting position *pose*, with LM lanes of width W and length L as graphically represented in Fig. 1 for each corresponding case analyzed. This replans the coverage path based on local sonar information gain. Calling the method *add_case_0(pose, W, L, A)* after this step generates LM lanes $2W$ apart until coverage area's edge, assuming no further interesting object detection.

Algorithm 1 Basic Accordion Coverage Path Planner

```
1:  WPs = add_case_0(pose,W,α_lm,L,A)
2:  k = 0
3:  while x_auv ≤ A do
4:      pose = WPs(k);
5:      pose = move_to_next_wp(pose, WPs)
6:      C = cost_map(sss_image(WPs(k):WPs(k+1)))
7:      if contains_int_obj(C) then
8:          if (obj_to_the_right&right_not_resonified) then
9:              WPs(k + 1 : end) = [ ]
10:             if (direction = 'down') then
11:                 WPs.add_case_1(pose,W,L,A)
12:                 WPs.add_case_0(WPs(end),W,L,A)
13:             else if (direction = 'up') then
14:                 WPs.add_case_2(pose,W,L,A)
15:                 WPs.add_case_0(WPs(end),W,L,A)
16:         else if (obj_to_the_left&left_not_resonified) then
17:             WPs(k + 1 : end) = [ ]
18:             if (direction = 'down') then
19:                 WPs.add_case_3(pose,W,L,A)
20:                 WPs.add_case_0(WPs(end),W,L,A)
21:             else if (direction = 'up') then
22:                 WPs.add_case_4(pose,W,L,A)
23:                 WPs.add_case_0(WPs(end),W,L,A)
24:         else if (obj_to_the_right&obj_to_the_left&
25: &right_not_resonified&left_not_resonified) then
26:             WPs(k + 1 : end) = [ ]
27:             if (direction = 'down') then
28:                 WPs.add_case_5(pose,W,L,A)
29:                 WPs.add_case_0(WPs(end),W,L,A)
30:             else if (direction = 'up') then
31:                 WPs.add_case_6(pose,W,L,A)
32:                 WPs.add_case_0(WPs(end),W,L,A)
33:     k ← k + 1
```

If there is something in the current sonar range to the right of the vehicle, as is the case while the vehicle moves "upwards" along line 1 in Fig. 1c (Case 2), the BA-CPP algorithm will generate new waypoints of LM line 3, W meters to the right of the LM line 1, to ensure sonar shadows sonification from the opposite side, and continue generating $2W$-wide LM pattern, see lines 14 and 15 in Algorithm 1.

In case that the vehicle detects interesting objects to its left side, e.g. while traversing "downwards" along the LM line 3 in Fig. 1d (Case 3), BA-CPP will generate a new LM line 5, which is W meters to the left of the vehicle. After this line, it sets the line 6, again assuming no further LM lanes containing interesting objects, see lines 19 and 20 in Algorithm 1. If on the other hand, the vehicle encounters an interesting object to its left side while traversing "upwards" along the LM line 3 in Fig. 1e (Case 4), BA-CPP will generate a new loopback LM line 5, which is W meters to the left of the vehicle, and add the line 6, again assuming

no further LM lanes containing interesting objects, see lines 22 and 23 in Algorithm 1.

If the vehicle detects interesting objects both to its left and right side, e.g. while following LM line 3 "downwards" in Fig. 1f (Case 5), it will first do a loopback maneuver to its right side (LM lines 4, 5, and 6), and follow a new LM line 7, which is W meters to the right of the LM line 3. After this line, it sets the LM line 8, again assuming no further LM lanes containing interesting objects. This is given in pseudocode in lines 28 and 29 in Algorithm 1. In the opposite case, when the vehicle moves "up" the LM line 3 as in Fig. 1g (Case 6), BA-CPP will generate a loopback to the left of the vehicle (LM lines 4, 5, and 6), and line 7 in order to sonify acoustic shadows of the objects to the right of its current LM line 3. This process is described by pseudocode lines 31 and 32 in Algorithm 1.

In the limiting case, in which each LM lane contains interesting objects, the BA-CPP algorithm behaves as the CL-CPP, see Fig. 1h (Case 7). The worst-case scenario occurs when the vehicle encounters the characteristic layout of LM lanes containing interesting objects, as shown in Fig. 1i (Case 8). The coverage path length for the current LM spatial period stays the same as in Cases 5 and 6, but in this case, the vehicle has to start the next LM spatial period at a line which is W away from line 7, making its cumulative coverage path significantly longer. Table I shows path lengths for all the above mentioned cases in one LM pattern spatial period, which are based on graphs depicted in Fig. 1a-1i.

B. Analysis of the upper and lower performance bounds

In this subsubsection the best- and the worst-case scenarios for the BA-CPP performance are analyzed (based on the coverage path lengths of the characteristic cases given in Table I) in order to predict the upper and the lower bounds of its performance. Performance measure, denoted by e_{CL-CPP}^{BA-CPP}, is chosen to represent the relative improvement of the BA-CPP generated coverage path length l_{BA-CPP} w.r.t. the CL-CPP path length l_{CL-CPP}. In case that the sea floor area which is being explored is devoid of any interesting objects (see Fig. 1a), then *the best-case scenario improvement* which the BA-CPP has over the CL-CPP algorithm, i.e. the upper bound on e_{CL-CPP}^{BA-CPP}, is given by:

$$e_{CL-CPP}^{best\,BA-CPP} = \frac{1}{2(1+\alpha_{lm})} 100[\%] \tag{2}$$

If the parameter $\alpha_{lm} \to 0$, the improvement of the BA-CPP method tends to $e_{CL-CPP}^{BA-CPP} \to E_{best} = 50\%$, ensuring a twice shorter coverage path than in case of the CL-CPP. On the other hand, if $\alpha_{lm} \to \infty$, the relative improvement tends to go to $e_{CL-CPP}^{BA-CPP} \to 0\%$.

In case that the interesting objects are positioned in such a way that the vehicle has do to a maneuver shown in Fig. 1f in each LM pattern spatial period, then the worst-case scenario improvement which the BA-CPP has over the CL-CPP method, i.e. the lower bound on e_{CL-CPP}^{BA-CPP}, is given by:

$$e_{CL-CPP}^{worst\,BA-CPP} = \frac{-\alpha_{lm}}{2(1+\alpha_{lm})} 100[\%] \tag{3}$$

TABLE I: Path lengths for different specific cases of the BA-CPP algorithm behavior. For reference to the BA-CPP algorithm's special behavior cases see Fig. 1.

Case No.	Path length of the CL-CPP method	Path length of the BA-CPP method
0	$4L + 4W$	$2L + 4W$
1	$4L + 4W$	$3L + 6W$
2	$4L + 4W$	$3L + 4W$
3	$4L + 4W$	$3L + 4W$
4	$4L + 4W$	$3L + 6W$
5	$4L + 4W$	$4L + 6W$
6	$4L + 4W$	$4L + 6W$
7	$4L + 4W$	$4L + 4W$
8	$4L + 4W$	$4L + 6W$

This means that for $\alpha_{lm} \to 0$, the BA-CPP is never worse than the CL-CPP, i.e. $e_{CL-CPP}^{BA-CPP} \geq 0\%$. On the other hand, when $\alpha_{lm} \to \infty$, the relative improvement tends to go to $e_{CL-CPP}^{BA-CPP} \to E_{worst} = -50\%$.

IV. Simulation Results

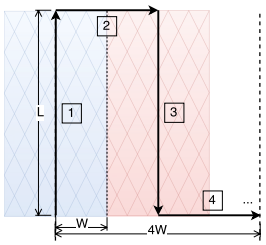
A. Statistical analysis of algorithm performance

In order to gain further insight into the behavior of the BA-CPP algorithms, its performance has been evaluated for various values of mission dependent parameters α_{lm} and p_{obj}. Parameter α_{lm} have been varied as $\alpha_{lm} \in \{0.01, 0.1, 0.25, 0.5, 0.75, 1, 2\}$, while p_{obj} took values from 0% to 100% with a 5% step. For each possible tuple (α_{lm}, p_{obj}), 100 tests with appropriate information gain cost maps have been generated, where the interesting objects have been dispersed randomly. Coverage area C had values $C \in \{5.5, 3.8, 3.3, 2.73, 2.34, 2.05, 1.36\}[km^2]$ for each α_{lm}, respectively. Cost maps values were binarised in a way that the value of $c = 0.1$ meant noninteresting areas, while $c = 1$ meant interesting areas. Relative coverage path length improvement of the BA-CPP algorithm over the CL-CPP pattern was averaged over 100 tests per each (α_{lm}, p_{obj}) parameter tuple.

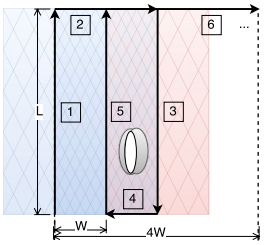
The results of this analysis are shown in Fig. 2a. It can be noted that the average improvement of the BA-CPP w.r.t. the CL-CPP is most significant in the area defined by low α_{lm}, $\alpha_{lm} \leq 0.25$, with $p_{obj} \leq 20\%$. In these cases, the BA-CPP generates coverage paths which are on average $25 - 50\%$ shorter than the CL-CPP paths. For $\alpha \leq 0.1$, BA-CPP coverage path is mostly shorter than the CL-CPP counterpart.

B. Best- and worst-case scenario analysis validation

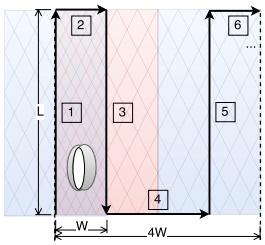
The goal of the aforementioned parameter variation analysis was to validate analytical best- and worst-case performance given in Subsection III-B. Aggregated distributions of relative coverage path improvements of the BA-CPP algorithm w.r.t. the CL-CPP for each fixed α_{lm}, and varied p_{obj}, are shown in Fig. 2b. Each of the boxplots contains the respective improvement values obtained from 2100 simulations ran per each α_{lm} value. It can be noted that the simulation results of the best-case scenarios match the predicted algorithm performance given by Eq. 2. For each (α_{lm}, p_{obj}) tuple 2 worst-case scenarios, shown in Fig. 1f,



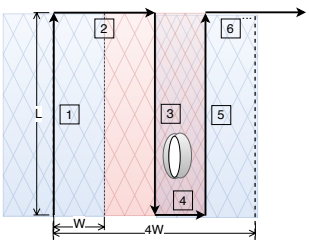
(a) Case 0: Best case, sea floor area to be covered is devoid of interesting objects, i.e. the initial solution of the proposed BA-CPP algorithm.



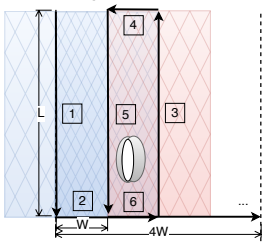
(b) Case 1: something was detected to the right of the vehicle in the cost map, while moving downwards along the line 3



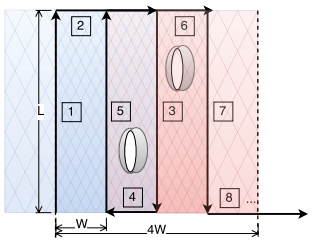
(c) Case 2: something was detected to the right of the vehicle in the cost map, while moving upwards along the line 1



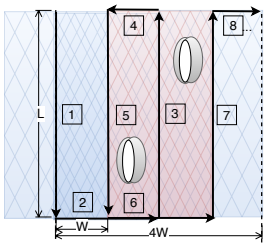
(d) Case 3: something was detected to the left of the vehicle in the cost map, while moving downwards along the line 3



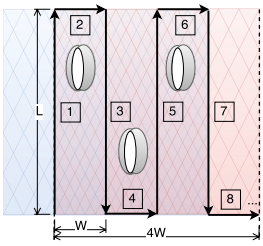
(e) Case 4: something was detected to the left of the vehicle in the cost map, while moving upwards along the line 3



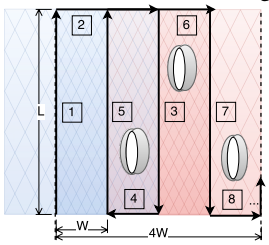
(f) Case 5: something was detected both to the left and to the right of the vehicle in the cost map, while moving downwards along the line 3



(g) Case 6: something was detected both to the left and to the right of the vehicle in the cost map, while moving upwards along the line 3



(h) Case 7: whole area to be covered is fully covered by interesting objects, i.e. CL-CPP pattern.



(i) Case 8: Worst case, added third LM lane in a row to have interesting objects

Fig. 1: Basic accordion coverage path planning algorithm: Characteristic cases of interesting objects (white ellipses) placement in current LM pattern period left and/or right of the current vehicle path in the cost map. Sonic shadows in side-scan sonar data are denoted by gray areas.

have been generated, and the performance of the BA-CPP algorithm in these cases also matches the predictions given by Eq. 3.

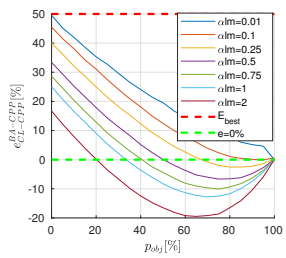
Fig. 2c shows the percentage of simulations p_{better}, for each α_{lm}, in which the BA-CPP generated shorter complete coverage paths than the CL-CPP. It can be noted that even for $\alpha_{lm} \leq 0.25$ over 70% of simulations gave better results using the BA-CPP instead of the CL-CPP. This percentage jumps significantly to 90% and even 100% as α_{lm} decreases to values of 0.1, and 0.01, respectively. This fact serves as a good example that shows how the BA-CPP generates shorter complete coverage paths than the CL-CPP in a wide range of parameter α_{lm} values that mostly used in practice.

C. BA-CPP performance function extrapolation

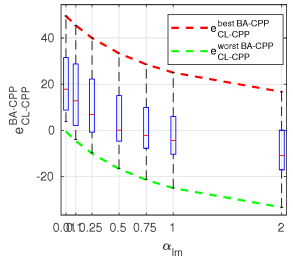
To generalize the conclusion about the BA-CPP algorithm performance, sample values of improvement percentages e_{CL-CPP}^{BA-CPP} resulting from the statistical analysis of the BA-CPP performance, have been interpolated by using a polynomial function of the third order of parameters α_{lm} and p_{obj} by using Levenberg-Marquardt method. This function is given by the following equation:

$$e_{CL-CPP}^{BA-CPP}(\alpha_{lm}, p_{obj}) = \sum_{i=0}^{3}\sum_{j=0}^{3} k_{ij}\alpha_{lm}^{i}p_{obj}^{j} \qquad (4)$$

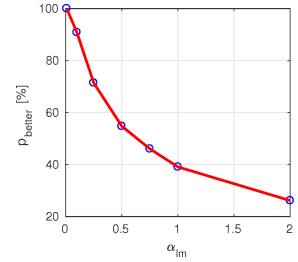
where $k_{00} = 50.2$, $k_{10} = -52.42$, $k_{01} = -0.6427$, $k_{20} = 34.86$, $k_{11} = -0.09411$, $k_{02} = -0.002369$, $k_{30} = -8.262$, $k_{21} = -0.05325$, $k_{12} = 0.00327$,and $k_{03} = 4.38e-05$ are nonzero coefficients estimated within 95% confidence bounds. This results is important from a system engineer's perspective, since it allows an estimate of possible survey mission average speed-up based on an a-priori known value of α_{lm}, and a rough estimate of p_{obj}.



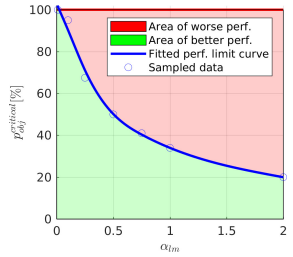
(a) BA-CPP algorithm averaged improvements relative to CL-CPP method.



(b) Coverage path lenth improvements distributions of the BA-CPP relative to the CL-CPP.



(c) Percentage $p_{better}[\%]$ of 630 simulations ran for each α_{lm} in which performance of the BA-CPP was better than the CL-CPP algorithm.



(d) Critical percentage of coverage area populated with interesting objects $p_{obj}^{critical}$ for which performance of the BA-CPP deteriorates to performance of the CL-CPP.

Fig. 2: BA-CPP algorithm performance analysis.

Curve $e_{CL-CPP}^{BA-CPP}(\alpha_{lm}, p_{obj}) = 0 = p_{obj}^{critical}(\alpha_{lm})$ is shown in Fig. 2d. This curve represents the limit case in which performance of the BA-CPP algorithm cannot be guaranteed to be on average better than performance of the CL-CPP. The main conclusion that needs to be stressed is that even when $\alpha_{lm} = 1$, the coverage area can contain even 35% of LM lanes populated with interesting objects in order for the BA-CPP algorithm to have on average the same performance as the CL-CPP method. For $\alpha_{lm} \leq 0.25$, the percentage of LM lanes of interest needs to be more than 70% for the BA-CPP to become equally inefficient as the CL-CPP.

V. CONCLUSION AND FUTURE WORK

The BA-CPP is an online side-scan sonar data-driven complete coverage path planning algorithm for unknown large-scale marine areas. It is designed bearing in mind that in most exploration and survey missions, LM segments are significantly longer than wider, and that only a small part of the coverage area is interesting for the survey mission. The BA-CPP algorithm overlaps neighboring side-scan sonar swaths only if they contain high information gain defined specifically for the current mission. The proposed algorithm replans the rest of the coverage mission taking estimated local sonar data information gain into consideration. Algorithm's upper and lower performance bounds are estimated analytically. Its performance is tested through extensive mission parameters variation simulations, which validate the modeled performance bounds. Simulation results show significantly shorter coverage paths obtained by the BA-CPP algorithm compared to the CL-CPP approach, which in a limiting case, results in a twice shorter coverage path.

Our next goal is to improve the BA-CPP algorithm by tightening its lower bounds of performance when it becomes worse than the CL-CPP algorithm and also to minimize coverage path length even further.

REFERENCES

[1] R. Plets, J. Dix, and R. Bates, "Marine geophysics data acquisition, processing and interpretation," *English Heritage*, May 2013.

[2] M. Ludvigsen, G. Johnsen, A. J. Sørensen, P. A. Lågstad, and Ø. Ødegård, "Scientific operations combining ROV and AUV in the Trondheim Fjord," *Marine Technology Society Journal*, vol. 48, no. 2, pp. 59–71, 2014.

[3] J. Ousingsawat and M. G. Earl, "Modified lawn-mower search pattern for areas comprised of weighted regions," in *2007 American Control Conference*, July 2007, pp. 918–923.

[4] A. Tahirovic and A. Astolfi, "A convergent solution to the multi-vehicle coverage problem," in *2013 American Control Conference*, June 2013, pp. 4635–4641.

[5] A. Tahirovic, M. Brkic, A. Bostan, and B. Seferagic, "A receding horizon scheme for constrained multi-vehicle coverage problems," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct 2016.

[6] S. Hert, S. Tiwari, and V. Lumelsky, "A terrain-covering algorithm for an AUV," *Autonomous Robots*, vol. 3, no. 2, pp. 91–119, Jun 1996.

[7] Y.-S. Jung, K.-W. Lee, S.-Y. Lee, M. H. Choi, and B.-H. Lee, "An efficient underwater coverage method for multi-AUV with sea current disturbances," *International Journal of Control, Automation and Systems*, vol. 7, no. 4, pp. 615–629, Aug 2009.

[8] L. Paull, S. SaeediGharahbolagh, M. Seto, and H. Li, "Sensor driven online coverage planning for autonomous underwater vehicles," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2012.

[9] A. Belbachir, F. Ingrand, and S. Lacroix, "A cooperative architecture for target localization using multiple AUVs," *Intelligent Service Robotics*, vol. 5, no. 2, pp. 119–132, Apr 2012.

[10] E. Galceran and M. Carreras, "Coverage path planning for marine habitat mapping," in *2012 Oceans*, Oct 2012, pp. 1–8.

[11] E. Galceran, S. Nagappa, M. Carreras, P. Ridao, and A. Palomer, "Uncertainty-driven survey path planning for bathymetric mapping," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 6006–6012.

[12] N. Abreu, N. Cruz, and A. Matos, "Accounting for uncertainty in search operations using AUVs," in *2017 IEEE Underwater Technology (UT)*, Feb 2017, pp. 1–8.

[13] H. Choset, "Coverage for robotics – A survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1, pp. 113–126, Oct 2001.

[14] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous systems*, vol. 61, no. 12, pp. 1258–1276, 2013.

[15] N. Kapetanović, M. Bibuli, N. Mišković, and M. Caccia, "Real-time model predictive line following control for underactuated marine vehicles," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12 374–12 379, 2017.

[16] F. Mandić, N. Mišković, and Z. Vukić, "Range–only navigation – maximizing system observability by using extremum seeking," *IFAC-PapersOnLine*, vol. 48, no. 16, pp. 101 – 106, 2015, 10th IFAC Conference on Manoeuvring and Control of Marine Craft MCMC 2015.

[17] M. Bayat and A. P. Aguiar, "AUV range-only localization and mapping: Observer design and experimental results," in *2013 European Control Conference (ECC)*, July 2013, pp. 4394–4399.

[18] W. H. Huang, "Optimal line-sweep-based decompositions for coverage algorithms," in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 1, 02 2001, pp. 27 – 32.

[19] K. Iagnemma and S. Dubowsky, *Mobile Robots in Rough Terrain: Estimation, Motion Planning, and Control with Application to Planetary Rovers*, 1st ed. Springer Publishing Company, Incorporated, 2010.