Coverage Path Planning for Harbour Seabed Surveys using an Autonomous Underwater Vehicle

Cheng Fang
Engineering Student
University of Sydney
Sydney, NSW, Australia
Cfan6677@uni.sydney.edu.au

Stuart Anstee
Maritime Operations Division
Defence Science & Technology Organisation
Eveleigh, NSW, Australia
stuart.anstee@dsto.defence.gov.au

Abstract—**This paper considers offline algorithms for pre-mission path planning to achieve coverage of the seabed of a complex but well-characterised planar area using an autonomous underwater vehicle fitted with side-looking sonar. The vehicle can swim over terrain at constant altitude, but must follow a path constructed from straight line segments. Imagery collected during turns is not used. This paper discusses the preparation of boundaries for the survey area, the introduction of buffer zones for safety of vehicle navigation, the decomposition of the surveyable area into sub-areas using an approximation to the generalised Voronoi diagram, calculation of paths within sub-areas that allow for incomplete sonar coverage, and connection of sub-area paths with transits to obtain a mission plan.**

Keywords - Path planning; autonomous underwater vehicles; seabed search

I. Introduction

Authorities administering harbours and waterways spend considerable effort conducting frequent and repetitive full-coverage surveys of the seabed in their areas of responsibility, in order to certify safety of navigation and to maintain maritime security. Currently, such surveys are conducted by personnel in boats, using high-resolution towed or hull-mounted sonar. Automation of such surveys using autonomous underwater vehicles (AUVs) fitted with sidescan or multibeam sonar may be an alternative to manned surveys in future, but there are many obstacles to overcome before this will be possible. One of them is preparing mission plans, that is, detailed survey instructions for the AUVs.

Current commercial survey AUVs follow prescriptive mission plans consisting of lists of geographical locations or 'waypoints'. They execute such plans by driving along straight lines from waypoint to waypoint, following the order given in the list. Vendor-supplied mission planning tools require the operator to generate plans line by line, or using a set of pre-defined line patterns. Such tools are adequate to plan missions in open areas, but plans for harbours may be difficult to construct, due to their complex and irregular contours and the presence of isolated obstacles that must accommodated in the mission plan. The idea of a tool that could automatically generate mission plans, given only nautical charts and a description of the required survey area, is therefore attractive.

A. *Vehicle Considerations*

The algorithms in this work are intended to generate mission plans for survey AUVs such as the REMUS and Bluefin families of vehicles [e.g., 1], which are designed to swim long distances at constant speed and altitude, with infrequent turns. They are not designed to follow curved paths and have little or no capacity to sense and avoid obstacles.

This work assumes that:

- Only one vehicle is available, with sufficient endurance to complete the mission,
- The vehicle is fitted with a navigation system that self-monitors its positional uncertainty and that the vehicle surfaces to re-establish its position if the uncertainty exceeds a tolerance,
- Turns occur at waypoints and all turning manoeuvres are contained within a fixed radius of the waypoints at which they occur, and consequently that
- A buffering region of radius b around the planned vehicle path will always contain the vehicle.

B. *Sensor Considerations*

This work assumes that the survey sensor on the AUV is a high-resolution conventional or interferometric sidescan sonar [2] designed for detection of small features. The sound emitted by the sonar is restricted to the neighbourhood of a vertical plane perpendicular to the forward axis of the vehicle. Transducers each side of the vehicle emit separate fan-shaped beams, which sense the seabed up to a maximum range. The maximum range of the sonar is much smaller than the widths of the bodies of water in which the vehicle will typically be deployed. The vehicle is operated at a constant altitude above the seabed, equal to 10-20% of the sonar range setting. Geometrical distortion and transducer sensitivity fluctuations make the data collected in near-vertical directions difficult to analyse and this 'nadir' region is excluded from analysis and coverage calculations. The maximum useful range of the sonar may also be less than its maximum range, due to lack of shadow detail. Fig 1 shows a notional representation of sidescan sonar seabed imaging.

978-1-4244-5222-4/10/$26.00 ©2010 CROWN

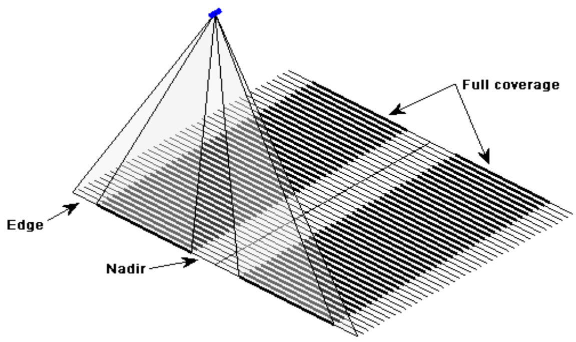

Figure 1. Height-exaggerated view of sidescan sonar seabed coverage, showing degraded coverage around the nadir and at the edges of the swath. Each 'ping' corresponds to one 'scan line' in the imagery.

Because the sonar is operated at constant altitude, its lateral coverage is approximately constant. Zones of reduced coverage are assumed at the edges of the swath and around the nadir. The widths of the zones are adjusted to account for degraded image quality and also to allow for the positional uncertainty or 'track-keeping error' of the vehicle.

Two parallel-line survey patterns are used to allow for 'infill' coverage of nadir and edge zones: (i) alternating the line spacing between 50% and 150% of the range setting, and (ii) using a constant line spacing of, say, 67% of the range setting. The patterns and their associated coverage are shown in Fig 2. Constant spacing has smaller large-scale efficiency than alternating spacing, but does not require that survey lines be paired to achieve full coverage; this can be an advantage when the exit point of the pattern must be at the opposite end to the entry point. The same effect can be achieved by adding a single line with 'constant' spacing to an alternating pattern.

Sidescan sonar image analysis depends on scan lines being approximately uniformly-spaced and parallel, i.e., collected when the vehicle is moving along a straight line at uniform speed. Imagery collected during turns is not reliable and is discounted in coverage calculations in this work. Imagery collected where nadir infill is not possible is also ignored.

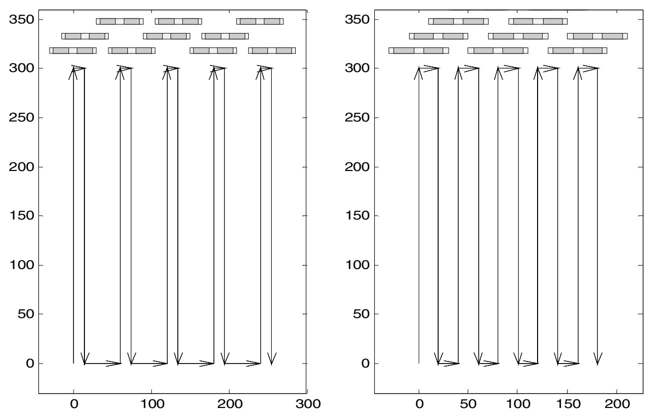

Figure 2. Survey line patterns with alternating spacing (left) and constant spacing (right). Bars indicate the sidescan sonar coverage of individual lines.

C. Environmental Considerations

This work applies to environments that are known in advance and static for the duration of the mission. Moving obstacles such as vessels are ignored.

The AUVs considered here lack the capacity to navigate around extended obstacles; therefore path plans must be restricted to 'safe water'. In this work, the safe water area is expressed as a closed external polygon that may contain 'holes' that are unsafe. Curves are encoded as closely spaced vertices. The safe water area is assumed to be effectively planar.

The safe water boundary is constructed from information available in electronic nautical charts [3]. In areas where the foreshore slopes upward to land, a depth contour is an appropriate safe-water boundary, except where structures such as wharves and jetties extend into deeper water. In such cases, the boundary is refined to include the outline of the structure. Large moored vessels may be included in structure outlines. Islands, shoals, navigational markers, moorings and steep underwater slopes constitute isolated areas where AUVs cannot safely navigate. They are included as 'hole' polygons.

D. Objectives

Harbour surveys are intended to detect changes in the seabed. In terms of safety of navigation and maritime security, the areas of most interest are places frequented by vessels, that is, shipping channels, berths and wharf areas. General foreshore areas are also of interest, but are difficult to survey due to numerous obstructions. Area coverage rate is maximised when overlap of sensor coverage is minimised. However, feature detection along boundaries is optimal when the sensor moves parallel to the boundary. The mission planning process is a compromise between these objectives.

II. PREPARATIONS

Assume that a top-level specification of the area to be surveyed is available as a single closed polygon and that a navigational chart containing it entirely is available. Polygons denoting safe water are calculated by subtracting the union of polygons (denoting land, areas with depth less than the minimum safe operating depth and outlines of structures) from a rectangle denoting the area of the chart. Points denoting the positions of smaller structures within the safe water area are also retained.

A. Buffering

Because the position of the vehicle along its path is only known to within a radius *b* and the path is not yet known, a buffer or 'offset' of thickness *b* is extended from all safe water boundaries and unsafe points within safe water. Several polygon offsetting algorithms are available; e.g. see [4] and references therein. However, an 'exact' offset curve has arcs at reflex (inward-pointing) vertices, sharply increasing the number of vertices necessary to represent the offset curve as a polygon, while a 'mitred' offset [4,5] introduces long, sharp 'peninsulas'. Therefore a 'capped' or 'bevelled' [5] approximation was developed.

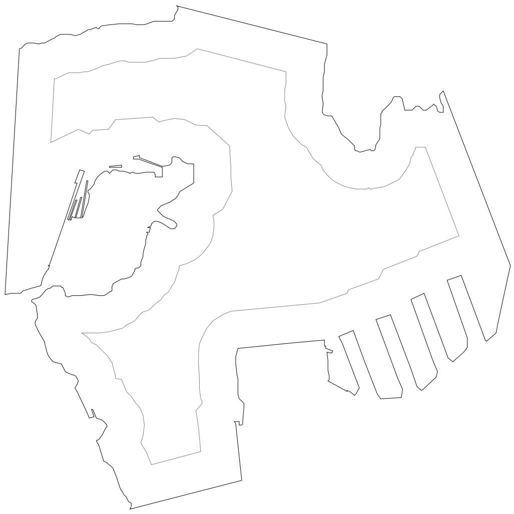

Figure 3. A bevelled polygonal offset curve (grey) and the original boundary curve (black)

The mitred offset curve replaced sharply projecting reflex vertices with two vertices whose connecting segment was tangential to the exact offset. This produced polygonal offset curves with acceptable shapes and numbers of vertices that were not excessive. Fig 3 shows an offset polygon produced using an exaggerated buffer radius.

B. Restriction and Simplification

The navigable part of the survey area is the intersection of the survey area polygon and the buffered safe water polygons. It consists of one or more exterior polygons, each of which may contain a number of holes. For simplicity, we restrict further discussions to a single exterior polygon that may contain holes and refer to the region enclosed between them as the 'working area'.

The number of vertices defining the boundaries of the working area may be very large. In order to control the computational cost of subsequent operations, it is desirable to simplify the boundary by replacing short segments with longer ones, while ensuring that all segments of the simplified polygon are within a tolerance ε of the original polygon – the so-called 'minimum rate' problem. An additional constraint arises from the requirement that, for safety of navigation, the simplified polygon must not pass into unsafe water. A fast algorithm [6] derived from the Douglas-Puecker algorithm [7] was therefore extended with an angular test to exclude segments that passed inside the original safe-water boundary. With heuristically determined parameters, this typically reduces the number of vertices by between 30% and 50%.

III. COVERAGE PATH PLANNING

Methods of coverage path planning [8], also called region filling and area covering, have been developed for robotic floor and window cleaning; mowing, harvesting and ploughing; landmine clearance and marine survey (see, e.g., references cited by [9,10,11]).

In the present case, the environment is known in advance, so an offline, 'global' algorithm can be used.

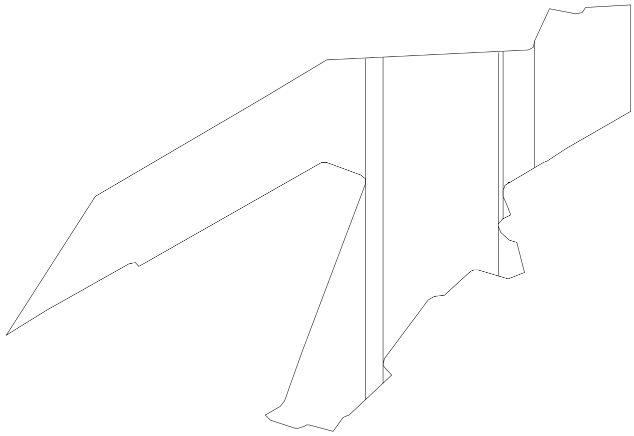

Figure 4. Boustrophedon decomposition of a polygonal area

Indeed, the lack of capacity for high-level feedback in the AUV control systems assumed here requires such an approach. Even with this simplification, computation of an optimal path is NP-hard [12], therefore numerically intractable. Approximate approaches are necessary.

A. Strategies Based on Decomposition

Of the approaches to coverage planning, the boustrophedon cellular decomposition [13,14,15] is the most closely aligned with the way AUV-based seabed surveys are planned by human operators. Fig 4 illustrates the decomposition of a polygonal working area using north-oriented cells. Although an exact cellular decomposition is strictly inappropriate in the presence of steering (navigational) error [16], standard AUV operating procedures incorporate overlap (section I.B) that is adjusted to allow for navigational error.

B. Iterative Decomposition

A single-orientation boustrophedon decomposition such as the one shown in Fig 4 results in paths which guarantee complete coverage, but it is inefficient if data collected during turns must be discarded, as in the present case. However, it can be retained as part of an approach using decomposition of the survey area into polygonal sub-regions that are individually efficient, in the sense that paths within them have low numbers of turns. It has been shown [17,18] that an optimal polygonal decomposition for a system that moves back and forth along parallel lines minimises the sum of the altitudes of the polygonal subregions; that is, it tends to produce polygons that are long and thin. However, the optimal decomposition process may have exponential complexity [17,18]; thus it is infeasible for environments described by polygons with large numbers of vertices, as in the present case.

Oksanen and Visala [10] describe a 'greedy' algorithm for the incremental decomposition of unstructured agricultural fields, using a process based on trapezoidal decomposition followed by merging. The resulting sub-areas are similar to those produced by boustrophedon decomposition. Their approach made no assumption about a preferred direction for the decomposition; instead, many directions were tested and a bisection search was used to determine the preferred direction at each stage of the decomposition. Oksanen and Visala used a heuristic, path-based cost function to assess the largest sub-area arising in each of six decompositions generated at 30° intervals. They selected the three most favourable directions and repeated the process, with additional decompositions oriented 15° either side of the chosen headings. The bisection search continued until the improvement per step fell below a goal value, which for their application was normally achieved after 5 refinements (about 1° resolution), requiring the evaluation of 36 separate decompositions. The largest sub-region in the minimum-cost decomposition was then subtracted from the working area and the process repeated, until the entire working area had been covered. Oksanen and Visala found that the process was effectively optimal for a convex region and produced high-quality paths for a region whose boundary consisted of long, straight segments. They noted that an algorithm that generated explicitly curved paths was more appropriate for a region incorporating curved boundaries and demonstrated such an algorithm, based on repeated polygonal offsetting. Their results serve as a benchmark for assessment of other approaches.

C. The Generalised Voronoi Diagram

The main contribution of this work is the application of the generalised Voronoi diagram to the decomposition problem. When operators are confronted with a survey area shaped like that in Fig 4, their intuitive approach is to seek long, thin regions into which arrangements of parallel lines fit naturally. If an offset curve is extended at an equal rate from all boundaries, the curve at which the offsets meet tends to follow the alignment of such long, thin regions. This curve is known as the 'skeleton', medial axis or generalised Voronoi diagram (GVD) [19]. Exact methods are available to calculate the GVD of a polygon with holes ([4,5] & references therein), but calculation times increase with the number of edges. An alternative approach used in this work was to discretise the working area on a regular square grid using a spacing less than the positional uncertainty of the AUV.

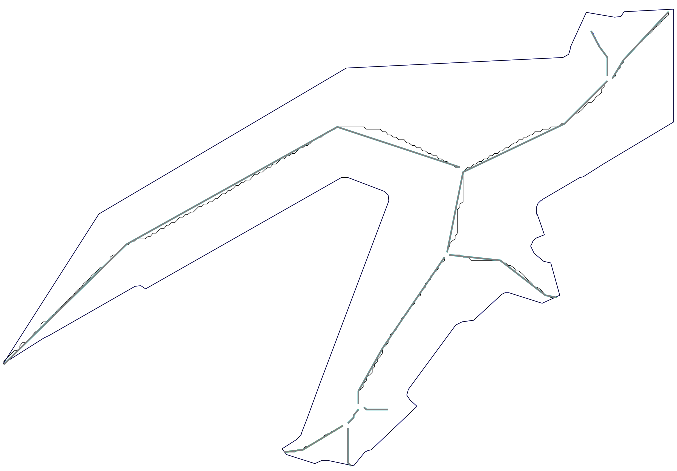

Figure 5. Approximations to the generalised Voronoi diagram or skeleton of a polygonal area. Thin black lines show the skeleton of the binary image, after removal of junction pixels. Thicker grey lines are simplifications of the branches of the binary skeleton.

The binary skeleton was calculated using methods from image processing [20] implemented in the Matlab Imaging Toolbox. Public-domain software [21] was used to locate end-points and junctions. Junction pixels were eliminated and polylines (open polygons) were constructed from the remaining pixels. The polylines were then simplified with an iterative approach based on the Douglas-Puecker algorithm [7], in which the number of segments approximating the original polyline was increased until an accuracy criterion was satisfied. An example of the resulting approximate skeleton is shown in Fig 5. In this example and more generally, long segments in the approximated skeleton tend to align with long, thin structures in the working area. In particular, if segments containing end-points of the branches of the skeleton are excluded, the longest of the remaining lines is frequently well aligned with the least obstructed survey direction in its immediate neighbourhood. An iterative decomposition scheme based on the approximated binary skeleton was devised as follows:

- *Repeat*
    - *Discretise the working area as a binary image*
    - *Calculate the binary skeleton*
    - *Approximate the skeleton by line segments*
    - *Choose the longest internal segment*
    - *Decompose the working area in its direction*
    - *Select the sub-region containing the segment*
    - *Plan the path of the AUV in the sub-region*
    - *Remove the sub-region from the working area*
- *Until {the remaining areas are below a minimum size}*

An earlier decomposition strategy relied on identifying a convex polygon around the line segment chosen for alignment. This was supplanted by boustrophedon decomposition, but the structure of the boundary gave rise to many narrow sub-regions, like those that can be seen in Fig 4. In order to reduce the number of sub-regions, an algorithm was devised to merge sub-regions whose common boundary comprised more than 80% of the partitioning line separating them. This strategy was successful in eliminating most excessively thin sub-regions, while retaining distinctions between widely separated sub-regions.

D. Treatment of Sub-regions

Individual sub-regions were treated separately for the purposes of path-planning. Paths within each cell were aligned with the edges of the cell produced by the decomposition process. Parallel lines were extended with uniform or alternating spacing, as shown in Figure 2, such that each individual survey leg was defined by the intersection of the extended line with the boundaries of the cell. On encountering the boundary, the path followed the boundary of the cell, adding vertices until the intersection with the next line was encountered. Where the width of the cell allowed, four different paths were calculated, corresponding to even and odd numbers of lines, with the first survey leg either parallel or anti-parallel with the alignment of the cell-generating line segment. Line separations were scaled to allow the appropriate numbers of lines in each case.

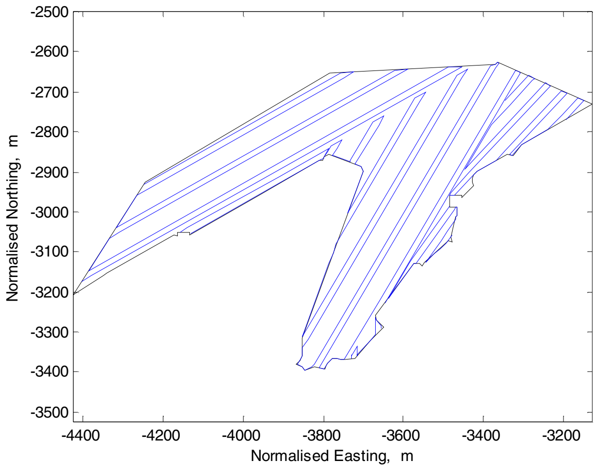

Figure 6. Decomposition and path generation using alternating line-spacing in a container terminal in Sydney Harbour. In this case, paths with 'even' numbers of survey legs are shown. Sub-region boundaries are implicit in the distinctions between the paths, which are not connected.

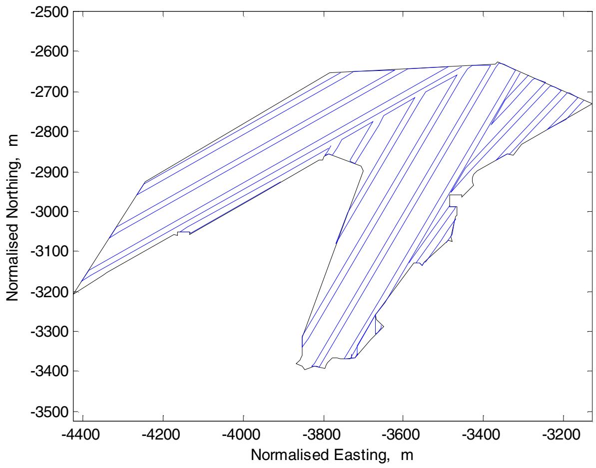

Figure 7. As per the previous figure, but with survey leg orientations reversed.

For missions with alternating line spacing, 'even' paths included only pairs of survey lines, but 'odd' paths included an extra line spaced two-thirds of the sonar range setting from the outermost paired survey line. The advantage of this approach is that it allows flexibility as to which pair of end-points is selected in order to minimise the transit distance involved in moving between sub-regions. Figs 6 and 7 demonstrate two of the four sets of paths associated with a particular decomposition, using line spacing alternating between 15 m and 45 m. The survey legs in each case are identical, but the positions of short and long connecting sections have been reversed, as have the termination points of each path. The decomposition shown in Fig 6 consisted of six sub-regions. The calculation was stopped when any remaining uncovered area was less than one-thousandth of the original survey area.

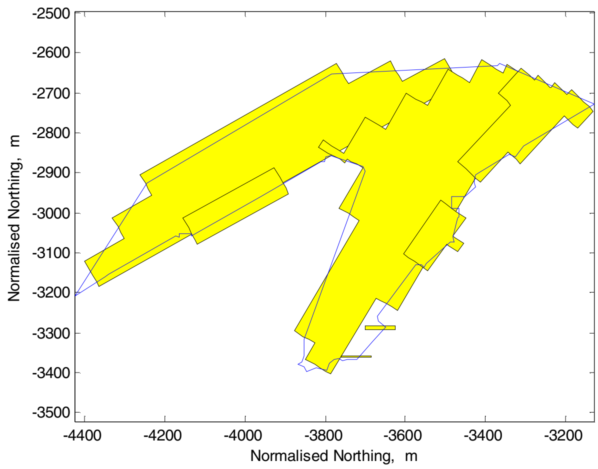

Figure 8. Sonar coverage associated with the survey lines shown in Fig 6.

Coverage calculations were conservative, neglecting coverage during transits and those parts of survey lines where overlap was insufficient to cover the nadir gap. Fig 8 shows the coverage estimate for the survey lines shown in Fig 6 and Fig 7 with a sonar range setting of 30 m per side. In this case, 94.7% of the survey area is covered with full nadir infill, although actual coverage is higher due to unpaired lines not included in the calculation. All significant coverage 'holes' occur near the boundary.

E. Quality of Survey Paths

Implemented in Matlab on a generic personal computer, the procedure outlined in preceding sections decomposes a survey area and generates four alternate sets of survey paths in seconds. The paths are not optimal for any but rectangular survey areas, but they are navigationally feasible and provide adequate coverage of navigable channels.

The paths are ineffective when there is a requirement to survey foreshore facilities, since survey lines do not usually run parallel to such structures. A survey along the wharves (the long, straight lines in Fig 6) may be better handled as a separate process, for which paths could be generated using a combination of the offsetting and line simplification algorithms discussed previously. This problem is not discussed further in this work.

A second problem arises where the process selects an inappropriate survey direction inside a cell. This usually occurs in small sub-regions, where it is associated with a disproportionate reduction in efficiency. The effect is visible in Fig 9, in a triangular sub-region northeast of an island.

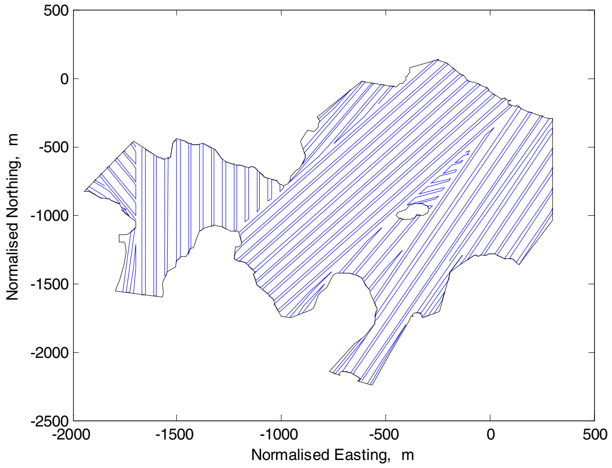

Figure 9. Decomposition and path generation in an irregular region including an island, using alternating line

Because the distances between survey lines are approximately fixed, the efficiency of the survey process falls as survey lines become shorter, either because sub-regions become progressively smaller as the process evolves, or because the algorithm selects a line orientation that is divergent from the longest dimension of the sub-region.

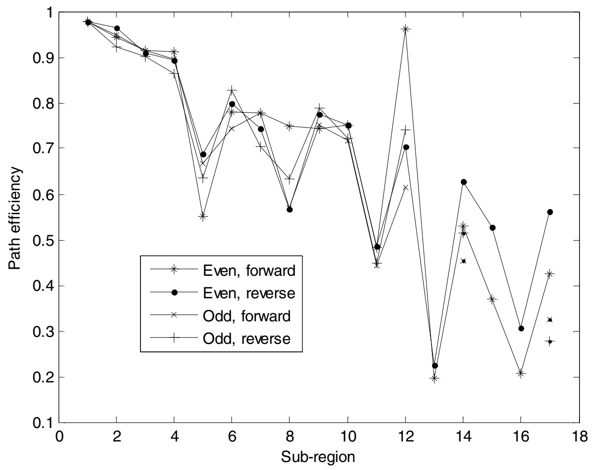

Figure 10. Efficiency of automatically generated paths in each of 17 sub-regions generated during the decomposition of the survey area shown in Fig 9.

Fig 10 shows the efficiency (survey length to total path length ratio) in each of the sub-regions generated in Fig 9. In this example, the paths in the first few sub-regions covering the largest areas are more than 90% efficient. The others are less efficient, but they are so much shorter that the total efficiency of paths generated in all sub-regions is approximately 88-89%. This suggests that the survey planning process will generate a result with useful efficiency, so long as the interconnections between sub-regions are arranged efficiently. Because the efficiency for cases considered to date has generally been found to be high, the more thorough approach of Oksanen and Visala [10] has not been implemented, as the additional benefit is unlikely to justify much larger computation times (potentially, 30 times greater) associated with a binary search for the most efficient path in each sub-region.

IV. INTERCONNECTING REGIONS

The main remaining requirement for construction of a survey path is efficient interconnection of sub-regions. This has two elements: efficient transit routes between exit and entry points located in different sub-regions, and efficient solution of a variant of the travelling salesman problem.

A. Transits between Sub-regions

A transit between an exit point from one sub-region and an entry-point to another should be the shortest safely navigable path between the two points, allowing for obstacles. Authors have considered this problem in connection with the efficiency of ship routing given vector chart information [22,23] and raster chart information [24,25]. The vector method of Viegas and Hansen [22] suffices for the present case, but because its computation time scales as the cube of the number of the number of nodes (points) describing the limits of the survey area and the entry and exit points, a raster method may be more appropriate for larger or more complicated areas. In the present case, there are typically hundreds of nodes describing the boundary of the survey area and a few tens of nodes describing entry and exit points. This means that the bulk of the execution time associated with the algorithm is spent building a network of paths between different points on the boundary and the penalty associated with calculating the distances between all possible pairs of entry and exit nodes is relatively small.

B. Travelling Salesman Problem

As noted previously, there are four possible paths traversing each sub-region, corresponding to eight choices of entry and exit point. In general, the lengths of the four paths are all different. An algorithm is required to minimise the length of a path surveying each of the sub-regions and transiting between exit and entry points in different sub-regions. This is a variant of the Travelling Salesman Problem (TSP) [26]; more accurately, the problem here is to find an optimal choice among 8^N individual networks, each of which constitutes a TSP with N nodes corresponding to the N sub-regions that must be visited. The different networks are distinguished by the entry and exit points at each node, and therefore the path lengths between the nodes, and the length of the path internal to each node. An exact solution of the TSP is numerically intractable [26] and the covering salesman problem [27-29], which is similar to the present problem, is still less tractable. Many sub-optimal methods have been devised for solution of the TSP and its variants, but the authors are not aware of an approach that is particularly suited to the current problem. However, the problem is easily formulated in terms of a genetic algorithm (GA) [30]. This approach is known to have limitations when applied to TSPs [31] and may require improvement, but paths generated with it have proven feasible to date. Fig 11 shows an example of a path generated this way.

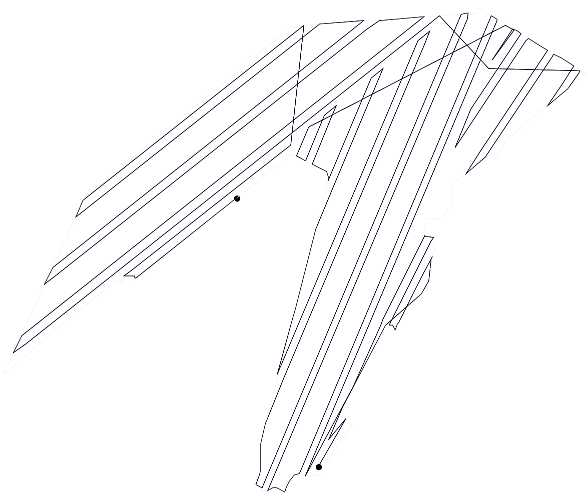

Figure 11. A completed survey path generated using a genetic algorithm approach to the solution of the modified Travelling Salesman Problem. Black dots indicate the beginning and ending points of the path. The borders of the survey area are shown in light grey.

In the conventional approach to solving the TSP with a GA, the algorithm generates a 'population' of random sequences of node numbers representing potential paths. Swapping, shifting and reordering operations are applied to evolve the population. In the present case, a parallel population designating the 'internal' path chosen to survey each sub-region was generated, and at each evolutionary cycle, identical operations were applied to both populations. The elements of the second population were random integers from 1 to 8, representing the four paths and the two directions in which they can be traversed. The genetic algorithm was found to converge within a few iterations, implying that the quality of the solution may be more dependent on the size of the initial population than is desirable. In the case shown in Fig 11, the efficiency of the completed path, in terms of distance surveying versus total distance, is approximately 80%. Visual inspection suggests that further improvements, while possible, would not be large.

V. Concluding Remarks

This work has demonstrated a process allowing automatic generation of a path that could be used to survey an area with complex shape, although further simplification of the path will be required in order to implement it as a mission program for an autonomous underwater vehicle. The current Matlab implementation generates paths for limited areas in a few minutes on a personal computer and the skeleton-based heuristic approach to the decomposition of the survey area appears to be effective, although incorporation of the search-based decomposition approach of Oksanen and Visala [10] may be useful when the skeleton-based approach produces large numbers of sub-regions with poor efficiency. It is likely that more efficient offsetting and route-finding algorithms will be necessary for larger and more complex problems than those so far considered. The option of manual intervention may also be valuable at each stage of the process.

Acknowledgment

The authors thank R. Winter for useful discussions about the coverage path planning literature. Matlab code contributed to the Matlab Central website by E. A. Johnson, B. Obara, W. Schwanghart and J. Kirk was used in the work.

References

[1] J. W. Nicholson and A. J. Healey, "The present state of autonomous underwater vehicle (AUV) applications and technologies," Marine Tech. Soc. J., vol. 42, pp. 44-51, Spring 2008.

[2] A. J. Kenny, I. Cato, M. Desprez, G. Fader, R. T. E. Schuettenhelm, and J. Side, "An overview of seabed-mapping technologies in the context of marine habitat classification," ICES J. of Marine Sci., vol. 60, pp 411-418, 2003.

[3] IHO transfer standard for digital hydrographic data, Edition 3.1, International Hydrographic Bureau, Monaco, November 2000.

[4] R. Wein, "Exact and approximate construction of offset polygons," Computer-aided Design, vol. 39, pp 518-527, 2007.

[5] D. Eppstein, "Raising roofs, crashing cycles and playing pool: Applications of a data structure for finding pairwise interactions," Discrete and Computational Geometry, vol. 22, pp. 569-592, December 1999.

[6] A. Kolesnikov and P. Franti, "Polygonal approximation of closed curves," Pattern Recognition, vol. 40, pp. 1282-1293, (2007).

[7] D. H. Douglas, T. K. Peucker, "Algorithm for the reduction of the number of points required to represent a line or its caricature," The Canadian Cartographer, vol. 10, pp. 112-122 (1973).

[8] H. Choset, "Coverage for robotics – A survey of recent results," Annals of Mathematics and Artificial Intelligence, vol. 31, pp. 113-126, 2001.

[9] X. Wang and V. L. Syrmos, "Coverage path planning for multiple robotic agent-based inspection of an unknown 2D environment," 17th International Conference on Control and Automation, Makedonia Palace, Thessaloniki, Greece. June 24-26, 2009.

[10] T. Oksanen and A. Visala, "Coverage path planning algorithms for agricultural field machines," J. Field Robotics, vol. 26, pp. 651-558, 2009.

[11] M. Ozkan, A. Yazici, M. Kapanoglu and O. Parlaktuna, "Hierarchical oriented genetic algorithms for coverage path planning of multi-robot teams with load balancing," Proceedings of the First ACM/SIGEVO Summit on Genetic and Evolutionary Computation, pp. 451-458, 2009.

[12] E. M. Arkin and R. Hassin, "Approximation algorithms for the geometric covering salesman problem," Disc. Appl. Math., vol. 55, pp. 197-218, December 1994.

[13] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," in Int. Conf. on Field and Service Robotics, Canberra, Australia, December 1997.

[14] H. Choset, "Coverage of known spaces: The boustrophedon cellular decomposition," Autonomous Robots, vol. 9, pp. 247-253, 2000.

[15] E. Garcia and P. Gonzalez de Santos, "Mobile-robot navigation with complete coverage of unstructured environments," Robotics and Autonomous Systems, vol. 46, pp. 195-204, 2004.

[16] M. Mazo Jr., K. H. Johansson, "Robust area coverage using hybrid control," TELEC, Santiago de Cuba, Cuba, 2004.

[17] W. H. Huang, "The minimal sum of altitudes decomposition for coverage algorithms," Rensselaer Polytechnic Institute Computer Science Tech. Rep. 00-3. June 2000.

[18] W. H. Huang, "Optimal line-sweep-based decompositions for coverage algorithms," Proc. of the 2001 ICRA IEEE Conf. on Robotics and Automation, 2001.

[19] M. de Berg, O. Cheong, M. v. Kreveld and M. Overmars, Computational Geometry, 3rd Ed., Springer, 2008.

[20] A. K. Jain, Fundamentals of digital image processing, Prentice-Hall, 1989.

[21] C. d. Ruberto, "Recognition of shapes by attributed skeletal graphs," Pattern Recognition, vol. 37, pp. 21-31, 2004. As implemented by B. Obara and made available at http://www.mathworks.com/matlabcentral.

[22] J. Viegas and P. Hansen, "Finding shortest paths in the plane in the presence of barriers to travel (for any l_p norm)," European J. of Operational Res., vol. 20, pp. 373-381, 1985.

[23] K. Fagerholt, S. I. Heimdal and A. Loktu, "Shortest path in the presence of obstacles: An application to ocean shipping," J. of the Operational Res. Soc., vol. 51, pp. 683-688,

[24] K.-Y. Chang and G. E. Jan, "A method for searching optimal routes with collision avoidance on raster charts," J. of Navigation, vol. 56, pp. 371-384, 2003.

[25] R. Szlapczynski, "A new method of ship routing on raster grids, with turn penalties and collision avoidance," J. of Navigation, vol. 59, pp. 27-42, 2006.

[26] W. L. Wilson, Operations research, Applications and Algorithms, Thomson, 2004.

[27] J. R. Current and D. A. Schilling, "The covering salesman problem," Transportation Science, vol. 23, pp. 208-213, August 1989.

[28] J. R. Current and D. A. Schilling, "The median tour and maximal covering tour problems: Formulations and heuristics," European J. of Operational Res., vol. 73, pp. 114-126, 1994.

[29] E. M. Arkin and R. Hassin, "Approximation algorithms for the geometric covering salesman problem," Discrete and Applied Math., vol. 55, pp. 197-218, 1994.

[30] A. E. Eiben, Introduction to evolutionary computing, Springer, 2003.

[31] L. V. Snyder and M. S. Daskin, "A random-key genetic algorithm for the Generalised Travelling Salesman Problem," Technical Report #04T-018, Dept. of Industrial and Systems Engineering, Lehigh University.