Sensor coverage with a heterogeneous fleet of autonomous surface vessels*

P. Mojiri Forooshani and M. Jenkin
York Centre for Field Robotics and Lassonde School of Engineering
York University, Toronto, Ontario, M3J 1P3, Canada
{pmojirif, jenkin}@cse.yorku.ca

***Abstract*—Sensor coverage of large areas is a problem that occurs in a variety of different environments from terrestrial to aerial to aquatic. Here we consider the aquatic version of the problem. Given a known aquatic environment and collection of aquatic surface vehicles with known kinematic and dynamic constraints, how can a fleet of vehicles be deployed to provide sensor coverage of the surface of the body of water? Rather than considering this problem in general, here we consider the problem given a specific fleet consisting of one very well equipped robot capable of global localization aided by a number of smaller, less well equipped devices that rely on the main robot for localization and thus must operate in close proximity to the main robot. A boustrophedon decomposition algorithm is developed that incorporates the motion, sensing and communication constraints imposed by the autonomous fleet. The approach is demonstrated in simulation using a real aquatic environment withe portions of the approach demonstrated using a fleet of real robots operating outdoors.**

***Index Terms*—sensor coverage, aquatic surface vehicles, robot collective**

I. Introduction

Sensor coverage of a large body of water is a task that has wide applicability, but to ground the task in a specific example consider the problem of responding to a major oil spill accident such as the Deepwater Horizon disaster [1]. Dealing with an environmental disaster of this magnitude introduces a range of tasks related to distributed sensing. For example, it becomes critical to be able to answer questions such as where, within this environment, is the contamination and how serious is it at different locations. From a technical point of view answering these and related questions involves providing sensor coverage of the impacted area. Although satellite and airborne sensing can provide some of this information, determining the depth and concentration of contaminants (e.g., oil) at or near the surface requires sensors to actually be deployed over an extremely large surface area.

The performance of coverage algorithms for a given environment is limited by the robot sensor footprint and its speed through its environment. Given the time sensitive nature of providing sensor coverage for tasks such as oil spills and the large potential extent of the disaster, a single autonomous device is unlikely to be able to cover such an environment in a timely manner. One potential solution to this problem is to deploy a collection of autonomous devices rather than a single robot. Introducing multiple robots provides advantages in terms of efficiency and robustness but increases the complexity of the coverage algorithm and also requires that the various elements operate within a common frame of reference. There is also the critical question of the nature of the group of robots being deployed. The group of robots could be homogeneous. That is, each member of the group could have the same capabilities and characteristics. Another approach would be to deploy a heterogeneous group of robots where each robot may have different capabilities. In such a case it becomes important to deploy the group of robots so that each robot can best exploit its own characteristics. For example, a heterogeneous team may consist of some robots equipped with high accuracy localization sensors, others with less effective sensors, and perhaps others with no form of position perception at all. How can all the elements in this group operate within a common frame of reference? Surveying the environment with such a team requires the team to share information allowing the more capable robots to assist those with lower quality instrumentation.

This paper describes a sensor coverage process that has been developed in order to enable a heterogeneous fleet of autonomous surface craft to survey an extended body of water. Although the sensor coverage results here are all of simulations of the fleet, the work presented here is predicated on the eventual deployment using a fleet of robots based on the platforms shown in Figure 1. This fleet consists of one well equipped robot (Eddy) aided by a collection of smaller, less well equipped robots (Minnows). (See [2] and [3] for details on these platforms.) Eddy has access to an accurate DGPS signal and this, coupled with an on board tilt-compensated compass, provides Eddy with an accurate estimate of its pose. On the other hand the Minnow robots must rely on pose update information from Eddy. This pose estimate is based on Eddy monitoring the position of each Minnow relative to itself and then communication of this information to the Minnow robots. One critical constraint imposed by this process is that the Minnow robots must remain in relatively close proximity to Eddy, and thus the entire fleet moves as a single coordinated unit, rather than having the sensor coverage problem addressed by dividing up the world into independent sectors and deploying one robot to each sector.

*The financial support of NSERC Canada and the NSERC Canadian Network in Field Robotics is gratefully acknowledged.

(a) Eddy

(b) Minnow

Fig. 1. The robot classes that make up the fleet. (a) Eddy is a differential drive platform with two propellers. This allows it to change orientation independently of changes in position. (b) The Minnow robots utilize a propeller-rudder mechanism. This arrangement requires that the robot move in order change orientation, much like a tricycle ground contact robot.

We begin by reviewing the capabilities of the Eddy and Minnow robots and the process of obtaining good localization estimates of the entire fleet including communicating this information throughout the fleet. We then describe how this fleet can be used to provide sensor-based coverage, including providing sensor coverage using a simulated fleet on a real body of water.

II. Eddy and the Minnows

The fleet is composed of vehicles drawn from two robot classes. These robots have been deployed in a number of different aquatic environments, from the waters of the Caribbean to ponds and lakes in Ontario, to indoor pools. Eddy is a differential drive vehicle. The Minnows are essentially modified radio-controlled (RC) motorboats.

a) **Eddy:** Eddy is a modified version of the Kingfisher M100 platform developed by Clearpath Robotics [4]. The Kingfisher M100 is a differential-drive surface vessel powered by two outboard motors. Eddy is equipped with a tilt-compensated compass, differential GPS, an omnidirectional camera system, a wireless network hub for local communication as well as a long-range radio communication channel for off-board command and control. The vehicle has been augmented with additional onboard computation and sensors as well as through additional floatation support.

b) **The Minnow's:** Each Minnow robot is a re-purposed radio controlled (RC) hobby boat. The vessel is powered by a single DC motor driving a prop while a RC servo motor is used to provide control over the rudder. Each Minnow robot is equipped with a tilt-compensated compass, a front-facing video camera, and a commercial GPS system although the latter is not used here. Onboard computation is provided via a Beagleboard which communicates to the outside world via standard WIFI technology.

Each of the robots runs ROS [5] each with its own ROSCORE. A heartbeat strategy is used to monitor liveliness among the robot fleet and to communicate localization and other information among the fleet members. See [2] for details.

III. Collaborative localization

We assume that Eddy is able to solve the localization problem via access to high resolution external sensors (e.g., DGPS, a tilt-compensated three axis digital compass and the like) but other solutions are required for robot localization for the Minnow robots. One potential inexpensive solution to the localization problem for the Minnow robots would be to equip them with commercial GPS systems (as they are). Unfortunately such systems do not provide the necessary accuracy for localization of the Minnow robots. Notwithstanding the accuracy and precision reported by the manufactures of such devices, positional errors in the range of 10's of meters are not uncommon, with considerable variability in the reported position. A more practical solution is to instrument Eddy with sufficient sensing capability to determine the position (dx, dy) of each Minnow robot relative to the Eddy robot and to utilize a compass on board each of the Minnow robots to compute their absolute orientation (θ).

Many solutions are possible for obtaining the position of each Minnow robot relative to Eddy. Here a visual system is employed. A 360 degree panoramic camera is mounted on Eddy to track the movement of the Minnow robots using coloured targets mounted on them. Although commercial panoramic sensors exist, in order to deal with the wide range of visual conditions and in order to harden the sensor against environmental concerns, a custom sensor was constructed from a collection of eight networked web cameras. These images could be pasted together in order to provide a panoramic view of the robot's environment although such processing is not required here, rather, each camera's image stream is processed independently and only the locations obtained by the individual cameras combined.

Localizing the individual Minnow robots from the omnidirectional camera mounted on Eddy could be accomplished in

(a) (b)

Fig. 2. Blob detection on the water.

a number of different ways as well. Perhaps the simplest is to assume that the water's surface is flat, that the minnow's themselves are flat, and then to augment the Minnow robots so that they are easily identified and localized in the visual scene. Under these assumptions localizing the individual robots is straightforward although it then becomes necessary to evaluate the impact of these assumptions on the accuracy of the localization process.

Let us assume that an individual Minnow robot lies on the surface of the water and that it is easily localizable in a camera image. Then under the pinhole camera model the relationship between a point $(X, Y)^T$ on the surface of the water and its corresponding image point $(x, y)^T$ in a camera is given by the Homography H. Individual cameras can be easily calibrated using a calibration target prior to deployment to determine H_j for each camera j between points on the water's surface given in an Eddy-centric coordinate frame and each individual camera.

Localization of an individual Minnow robot is simplified by providing each robot with a uniquely coloured target (Figure 2) Although these targets may appear large relative to the actual size of the Minnow robot, they are very light and are designed to be easily detected using standard color-blob detection algorithms (e.g., [6]). One concern with the use of such coloured targets to aid in the localization of the minnows is that the centroid of the coloured blob does not necessarily correspond to the centre of the robot (its origin) and that the centroid of the blob will not lie on the water's surface, thus violating the planar target assumption that underlies the use of a Homography to map image coordinates to positions around Eddy. Figure 2 illustrates this potential problem for a Minnow operating on the surface of the water.

In order to address this concern controlled tests were made of reported robot positions for various Minnow robot positions and orientations within 5m of Eddy (see Figure 3). A Minnow robot was placed at different displacements from Eddy in four different orientations; facing Eddy, facing away from Eddy, sideways onto Eddy and at an oblique angle so that two faces of the target were visible. For each point the individual errors as well as the maximum error in x and y positions are plotted. As can be seen maximum errors were relatively small, and certainly less than one meter. This is to be contrasted with the 0.5m-2.0m error associated with the DGPS signal available on Eddy. Figure 4 shows raw position

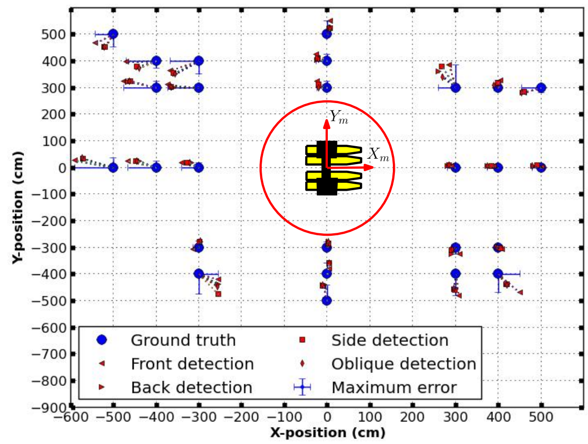


Fig. 3. Maximum Minnow localization error as a function of position and orientation relative to Eddy.

tracking results for a number of linear motion tracks of the Minnow robot. Ground truth motion is shown as dotted lines. These raw measurements can be integrated with the on-board compass via a Kalman filter to estimate individual Minnow pose as a function of time.

IV. Communication infrastructure

Each Minnow robot is equipped with a tilt-compensated compass that provides heading information. Providing each robot with a complete instantaneous pose estimate involves integrating pose data available on the Eddy robot (Eddy robot pose, and positional offsets to each of the Minnow robots with orientation information available on each of the Minnow robots). Critical to this process is the development of an appropriate communications infrastructure operating between Eddy and the Minnows.

ROS [5] has become the *de facto* standard middleware for experimental robot development in the terrestrial domain. One notable issue with ROS is its performance when the communication infrastructure between ROS nodes becomes unreliable. In a single master roscore implementation, operation of the robot becomes problematic under such conditions as robust and timely communication between nodes within the ROS environment is a critical underlying assumption of the ROS architecture. This assumption can lead to catastrophic failure of the entire fleet. Communications failures are not easily detected within ROS, and last sent commands (e.g., boat velocity and steering angles) may remain in effect with catastrophic results..

A straightforward solution to this problem is to deploy multiple master roscores within the fleet, with one roscore per vessel. Each robot is then independent and communication failure within a single hardwired robot is unlikely. Communications between the various roscores can be accomplished in many ways. The fleet described here utilizes a "foreign relay" model within which the various roscore's relay specific

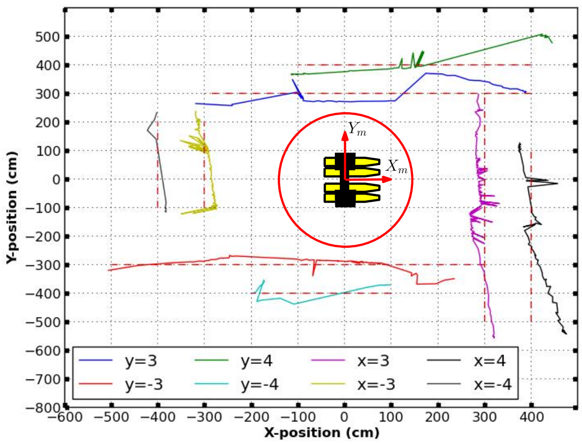


Fig. 4. Minnow tracking for various linear motion tracks. Ground truth lines are also shown as dotted lines.

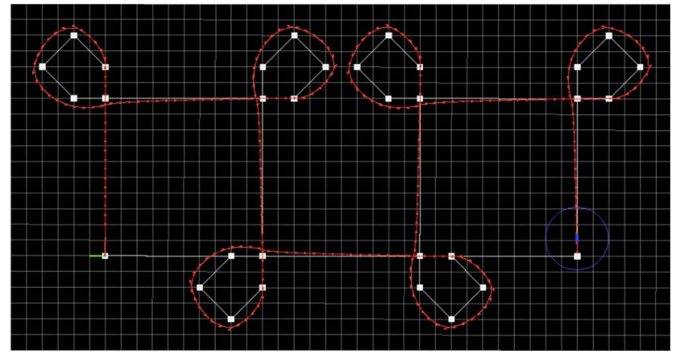


Fig. 5. Motion of a Minnow robot given a sequence of waypoints. A Minnow robot cannot decouples changes of orientation with changes in position, and so looping maneuvers are required to make the 90 degree turns requires by the BCD algorithm. The sequence of red arrows show the position and orientation of the Minnow robot as it moves through the waypoints (in white). In terms of the BCD algorithm the Minnow robot executes the necessary straight-line portions of the required path.

messages from foreign roscore's and inject these messages into the local roscore network when they are received. Although this addresses the requirement of liveliness within the software infrastructure of the fleet, it does not allow elements of the fleet to detect communication failure or to monitor the ongoing communication state of the fleet. In the context of maintaining pose estimates throughout the fleet, it is critical for Eddy to know when orientation estimates from the Minnow robots have become stale due to communication failure, and similarly the Minnow robots need to be aware when their position estimates are stale as well. (Detection of communication failure and re-acquisition is critical for other aspects too, of course.)

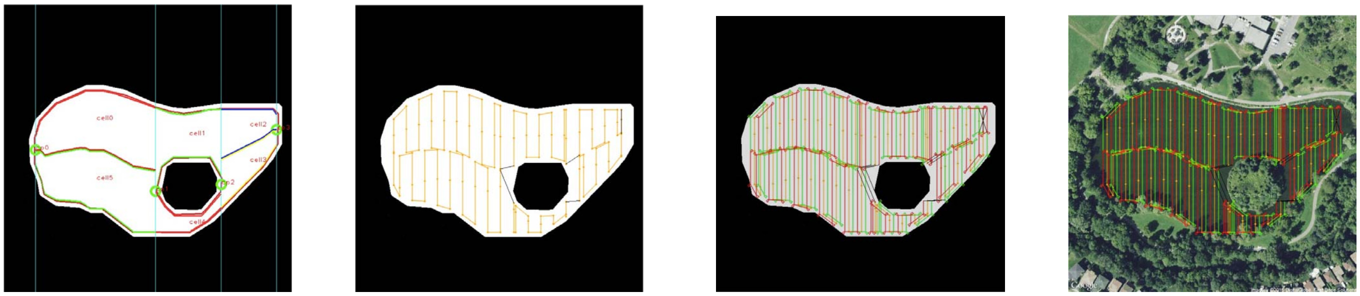
Given the hierarchical structure of the heterogeneous fleet, a simple heartbeat strategy is used between Eddy and each of the Minnow robots. This heartbeat strategy message includes current pose estimates of all of the Minnow robots and Eddy, as well as the current communication status of all of the robots in the fleet. The heartbeat message originates from each Minnow robot and is is echo'd from Eddy with updated position information, closing the pose estimation process.

V. Sensor coverage

The problem of coverage path planning of a known environment is strongly related to the well known art gallery problem [7]. In two dimensions, the basic version of the art gallery problem is defined as follows. We are given a known simple polygon P, and we wish to know if a set of points $S \subset P$, has the property that for every point $p \in P$, there exists some $q \in S$ such that the line segment between p and q does not leave the polygon [7]. In essence, the set of points S 'guards' all of P. This decision problem is known to be NP-hard [7]. In the computational geometry version of the problem the layout of the art gallery is represented by a simple polygon and each guard is represented by a point in the polygon. Here the problem is to find the minimum number of guards needed to station in a polygonal gallery, and hence each point in the gallery is visible to at least one guard. The Art Gallery problem does not require the determination of the path that the guard must follow nor does it consider a limited sensor range for the guard's sensor. In the autonomous systems literature the problem of robotic coverage and exploration has developed quickly, from early heuristic-based single robot techniques (e.g., [8], [9]) to multiple heterogeneous robots working in a coordinated fashion (e.g., [10], [11]). The problem has been extensively investigated in both the single-robot domain [12], [13] as well as for multi-robot systems [14]–[17].

As coverage of a given trapezoid is relatively straight-forward using a sequence of back and forth motions, one promising approach to sensor coverage is to decompose the environment into a collection of trapezoids and then to process each polygon in sequence. One of the simplest *exact cellular decomposition* techniques is trapezoidal decomposition [18]. In this approach a vertical line, named a *slice*, is passed left to right through a known environment which is occupied with polygonal obstacles. When a slice intersects a vertex (this is described as an *event*) *cells* are constructed via a sequence of open and close operations. There are three types of events: IN, OUT, and MIDDLE. At an IN event the current cell is closed and two new cells are opened. An OUT event is the reverse: two cells are closed, and a new one is opened. At a MIDDLE event, the current cell is closed, and a new one is constructed. The result of these events is a free space that is broken down into trapezoidal cells. Since each cell is a trapezoid, simple back-and-forth motions can be used to cover it. A depth-first-like graph search algorithm is used to obtain a path list that represents an exhaustive walk through the adjacency graph such that the path visits each node at least once.

Unfortunately the trapezoidal approach can become very inefficient in terms of the number of back-and-forth motions required. In order to address this problem, the now classical



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

Fig. 6. Motion coverage of “Loafer’s Lake” in Brampton, Ontario, Canada incorporating vessel non-holonomic constraints. The boundary is dilated ensuring that the robots can move outside of the coverage area. (a) The BCD algorithm is used to decompose the space into cells. (b) General fleet motion through the cells is planned as is motion between them. (c) Individual fleet element motions are planned based on this. (d) The final motion plan is overlayed on Google Earth imagery of the pond surface.

Boustrophedon[1] Cellular Decomposition (BCD) algorithm was introduced by Choset in [19], [20]. This decomposition is an enhancement of the trapezoidal decomposition that merges all cells between IN and OUT events into one cell to reduce excessive lengthwise motions. The BCD algorithm relies on changes in connectivity of a slice to determine the existence of an event instead of exploiting the structure of polygons to determine IN, OUT, and MIDDLE events.

The BCD algorithm results in a collection of regions that can be easily swept via motions of the robot using back-and-forth motion. All that remains is to develop a strategy for moving the robot from one cell to the other. A Reeb graph [21] represents the topology of the cellular decomposition where the nodes of the Reeb graph represent the critical points and the edges represent the edges connecting the neighbouring critical points, i.e., correspond to cells [22].

The cellular decomposition and the corresponding Reeb graph may result in inefficiencies in terms of the direction of coverage, how to move through the Reeb graph, and also in terms of where the start and end points are in the individual cells of the decomposition. Mannadiar and Rekleitis [23] presented an enhanced algorithm based on BCD for the complete coverage of a known environment with obstacles. The algorithm leads a mobile robot through a sequence of areas to be passed without wasting energy or time. The optimal solution to the Chinese Postman Problem [24] from graph theory was adapted for the calculation of cell ordering. By splitting selected cells into two components, the single cell coverage used in the BCD algorithm is modified in order to eliminate repeated coverage of a given area. In [25], the optimal coverage algorithm presented in [23] was extended for the general class of non-holonomic robots in the aerial robotics domain using a set of generated waypoints outlining the desired coverage path, which is then given as input to a robot motion controller. The authors also investigated the quality of the coverage path by changing the direction of coverage (sweep direction) before running the BCD algorithm, since fewer numbers of turns and longer straight path are more desirable for non-holonomic vehicles. They proposed three different strategies for sweep direction: setting it orthogonal to the dominant edge orientation for obstacle boundaries, aligning it directly with the distribution of the free space, and aligning it to be perpendicular to the dominant wind heading.

A critical assumption underlying these algorithms is that the vehicle is holonomic (a holonomic vehicle can turn in place and move independently in all directions). When actually applying one of these algorithms using real robots it may be necessary to have the robot undergo additional motions in order to actually make the instantaneous motions assumed by the (theoretical) algorithm.

Given the requirement that the fleet remain in close proximity for communication and localization purposes, the fleet is structured to move in a synchronized *line abreast* formation such that the sensor footprints of the various robots overlap. Such a formation maximizes the coverage perpendicular to the direction of motion during the back and forth sweeps that take place in the BCD algorithm. In this line abreast formation Eddy is positioned in the centre with the Minnow robots arrayed to the left and right.

The output of the BCD algorithm is a sequence of waypoints for the fleet. Unfortunately, it is not a sequence of waypoints that can actually be accomplished by vessels such as the Minnow robots that utilize a rudder and propeller for motion. These devices cannot change heading independently of changing position, and thus the output of the BCD algorithm must be adapted to the realities of the underlying hardware platform. Beyond this, there is the requirement that the fleet geometry remain constant during the back and forth motions within each cell. In order to address these concerns, each of the Minnow robots executes a complex maneuver as it approaches the 90 degree turn required by the BCD algorithm. (Note that a similar maneuver is not required for Eddy, as its differential drive arrangement of propellers allows it to decouple changes in position with changes in orientation.)

Figure 5 illustrates the motion behaviour executed by a Minnow robot when following the grid-like motions required by the BCD algorithm. When the robot approaches a 90 degree turn, it passes through the point and follows a curved

[1] From the Greek for ox-turning.

trajectory in the opposite direction of the intended new heading so that it can re-acquire the required BCD path shown in white in the Figure. In order for the robot to be able to follow this trajectory there must exist clear space outside of the free space covered by the BCD algorithm and the boundary obstacles are dilated by the necessary turning radius of the Minnow robot to permit this.

VI. Experimental validation

Figure 6 shows the decomposed cells and Boustrophedon motions in each cell for a sample real world environment. The environment, its Reeb graph and planned back-and-forth motion in each cell is shown in Figure 6. An Eulerian circuit is determined in order to identify the order in which the cells to be passed by the robot. The output from this process is a tour path connecting all of the environment cells is also shown.

VII. Summary and future work

Sensor coverage with fleets of robots is a complex task requiring solutions to localization, communication, navigation and basic sensor coverage. Given a heterogeneous collection of robots that must remain in close proximity a BCD-based approach will obtain a basic plan for the motion of the fleet. This requires a solution to global localization, a problem that can be solved in a cooperative fashion between the Eddy robot and the Minnows. Cooperative localizaiton requires an effective communication strategy among the fleet members. This strategy must deal with the reality of intermittent communication failure among the fleet members, and the potential for catastrophic failure of one or more of the fleet elements. Existing algorithms for sensor coverage – such as the BCD-based approach used here – make strong assumptions about individual vehicle locomotion capabilities. Many aquatic robots rely on a propeller/rudder strategy. This imposes kinematic/dynamic constraints on the robot which precludes the sharp turns that are assumed by the BCD algorithm. A turning maneuver is used to allow the robots to meet the requirements of the BCD algorithm.

In this paper we have demonstrated many of the underlying components required for sensor-based aquatic coverage using real robots operating outdoors, however the full algorithm has only been validated in simulation. Full field tests of the fleet performing sensor coverage is planned for later this year.

References

[1] http://en.wikipedia.org/wiki/Deepwater_Horizon_oil_spill, 2013.

[2] P. M. Forooshani, "Coordinated sensor-based area coverage and cooperative localization of a heterogeneous fleet of autonomous surface vessels (asvs)," Master's thesis, York University, 2015, in progress.

[3] A. Calce, P. M. Forooshani, A. Speers, K. Watters, T. Young, and M. Jenkin, "Autonomous aquatic agents," in *Proceedings of International Conference on Agents and Artificial Intelligence, ICAART 2013*, 2013.

[4] Clearpath Robotics, *Kingfisher M100 Overview*, http://www.clearpathrobotics.com/kingfisher, 2011.

[5] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *Open-Source Software workshop at the International Conference on Robotics and Automation (ICRA)*, 2009.

[6] "Opencv," opencv.org.

[7] J. O'Rourke, *Art Gallery Theorems and Algorithms*. Oxford University Press New York, 1987.

[8] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE Journal of of Robotics and Automation*, vol. 2, no. 1, pp. 14 – 23, 1986.

[9] E. Gat and G. Dorais, "Robot navigation by conditional sequencing," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 1994*, San Diego, CA, 1994.

[10] A. Howard, L. E. Parker, and G. S. Sukhatme, "Experiments with a large heterogeneous mobile robot team: Exploration, mapping, deployment and detection," *International Journal of Robotics Research*, vol. 25, pp. 431– 447, 2006.

[11] L. E. Parker, "Alliance: An architecture for fault tolerant multi-robot cooperation," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 2, pp. 220 – 240, 1998.

[12] J. Colegrave and A. Branch, "A case study of autonomous household vacuum cleaner," in *AIAA/NASA Conference on Intelligent Robots for Factory Field, Service, and Space (CIRFFSS)*, 1994.

[13] Y. Gabriely and E. Rimon, "Spanning-tree based coverage of continuous areas by a mobile robot," *Annals of Mathematics and Artificial Intelligence*, vol. 31, pp. 77 – 98, 2001.

[14] N. Hazon and G. A. Kaminka, "Redundancy, efficiency and robustness in multi-robot coverage," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 2005*, April 2005, pp. 735 – 741.

[15] I. Rekleitis and G. Dudek, "Multi-robot collaboration for robust exploration," *Annals of Mathematics and Artificial Intelligence*, vol. 31, pp. 7 – 40, 2001.

[16] I. Rekleitis, V. Lee-Shue, A. P. New, and H. Choset, "Limited communication, multi-robot team based coverage," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 2004*, vol. 4, 2004, pp. 3462 – 3468.

[17] X. Zheng, S. Jain, S. Koenig, and D. Kempe, "Multi-robot forest coverage," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2005*, 2005, pp. 3852–3857.

[18] J. C. Latombe, *Robot Motion Planning: Edition en anglais*. Springer, 1991.

[19] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," in *International Conference on Field and Service Robotics*, 1997.

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

[21] A. Fomenko and T. L. Kunii, *Topological modelling for visualization.* Tokyo: Springer-Verlag, 1997.

[22] E. U. Acar and H. Choset, "Sensor-based coverage of unknown environments: Incremental construction of morse decompositions," *The International Journal of Robotics Research*, vol. 21, no. 4, pp. 345 – 366, 2002.

[23] R. Mannadiar and I. Rekleitis, "Optimal coverage of a known arbitrary environment," in *IEEE International Conference on robotics and Automation, ICRA 2010*, Anchorage, AK, May 2010, pp. 5525 – 5530.

[24] J. Edmonds and E. L. Johnson, "Matching euler tours and the chinese postman problem," *Mathematic Programming*, pp. 88–124, 1973.

[25] A. Xu, C. Viriyasuthee, and I. Rekleitis, "Optimal complete terrain coverage using an unmanned aerial vehicle," in *IEEE International Conference on Robotics and Automation, ICRA 2011*, Shanghai, China, 2011, pp. 2513 – 2519.