



Department of Computer Architecture and Technology

MSc Thesis

# TOWARDS COVERAGE PATH PLANNING FOR AUTONOMOUS UNDERWATER VEHICLES

Thesis presented by  
**Enric Galceran,**

to obtain the degree of:  
**Màster en Informàtica Industrial,  
Automàtica, Control i Sistemes.**

Supervisor:  
Dr. Marc Carreras

Girona, July 2011



# Acknowledgements

I would like to express my sincere gratitude to the persons who, with their keen support, contributed to making this thesis possible.

First and foremost, I would like to thank my supervisor, Dr. Marc Carreras, who is always up for sharing with me a lot of his expertise and research insight.

It is difficult to overstate how pleased I was to be the team leader this year at the SAUC-E competition, such a needless task when the team leads on its own navigating at full sail with the kindness, enthusiasm and passion of Carlos, Simone, Chee Sing, Arnau, Pere, Carles, Tali, Narcís and David. Mere words run short to express what a good time was that. Likewise, I am fortunate to work hand by hand with the rest of my lab colleagues Andrés, Emili, Aggelos, Lluís and Pere.

I also wish and need to thank Dr. Vladimir Djapic for welcoming me at NATO Undersea Research Center (NURC), Dr. Warren Fox, Dr. David Williams and Dr. Alberto Álvarez for sharing with me their expertise and Alberto, Stefano, Ruben, Francesca and all the people I shared time with during my stay at NURC who with their support and the gift of their company made my days there more enjoyable and worth living.

A special mention is for my friends Dula and Nikola from the Laboratory for Underwater Systems and Technology (LABUST) at the University of Zagreb, who I first met at NURC. With their kind and simple gestures, they showed me an endless pile of new things both in the fields of research and engineering and on the most important one after all: life.

Thanks a lot to Miquel, for sharing with me his mastery of computer aided design.

I would like also to thank my friends for always being there, joining me in the discovery of what is life about and how to make the best of it.

Lastly, and most importantly, I wish to thank my parents, Mari and Jordi. They have always supported and encouraged me to do my best in all matters of life. To them I dedicate this thesis.

Thank You, Hvala, Grazie, Gràcies.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Overview and motivation of the thesis . . . . .	1
1.2	Research framework . . . . .	1
1.3	Goal of the thesis . . . . .	2
1.3.1	Objectives . . . . .	2
1.4	Planning of the thesis . . . . .	3
1.5	Outline of the thesis . . . . .	3
<b>2</b>	<b>Problem definition</b>	<b>5</b>
2.1	Problem context . . . . .	5
2.2	Brief introduction to path planning concepts . . . . .	6
2.2.1	Configuration space . . . . .	6
2.2.2	Free space . . . . .	8
2.3	The coverage path planning problem . . . . .	8
2.4	Problem statement . . . . .	8
<b>3</b>	<b>State of the art</b>	<b>9</b>
3.1	Heuristic and randomized approaches . . . . .	10
3.2	Cellular decompositions . . . . .	11
3.2.1	Aproximate cellular decomposition . . . . .	12
3.2.2	Semiapproximate cellular decomposition . . . . .	13

---

3.2.3	Exact cellular decomposition . . . . .	13
3.3	Other approaches . . . . .	23
3.3.1	Artificial potential fields . . . . .	23
3.3.2	Template-based models . . . . .	24
3.3.3	Neural networks and fuzzy logic . . . . .	24
3.3.4	Miscellaneous approaches . . . . .	25
3.4	Multi-robot coverage . . . . .	27
3.5	Coverage path planning for underwater environments . . . . .	28
3.5.1	Coverage path planning in current fields . . . . .	30
3.6	Discussion . . . . .	30
<b>4</b>	<b>Underwater obstacle detection</b>	<b>37</b>
4.1	Acoustic sensing review . . . . .	38
4.1.1	Single-beam sounders . . . . .	38
4.1.2	Mechanically scanned profiler . . . . .	39
4.1.3	Multibeam sounders . . . . .	40
4.1.4	Forward-looking sonars . . . . .	40
4.2	Sonar-based obstacle detection . . . . .	42
4.3	Discussion . . . . .	43
<b>5</b>	<b>Experimental work</b>	<b>45</b>
5.1	Experimentation with the Formigues islands dataset . . . . .	45
5.1.1	Goal of the experiment . . . . .	45
5.1.2	The Formigues islands dataset . . . . .	46
5.1.3	Experimental setup . . . . .	47
5.1.4	Results . . . . .	47
5.2	Experimentation at the SAUC-E competition . . . . .	48

5.2.1	Goal of the experiment . . . . .	48
5.2.2	The SAUC-E competition . . . . .	49
5.2.3	The Sparus AUV . . . . .	50
5.2.4	Experimental setup . . . . .	51
5.2.5	Results . . . . .	52
<b>6</b>	<b>Conclusion</b>	<b>55</b>
6.1	Contributions . . . . .	55
6.2	Future work . . . . .	56
6.2.1	PhD thesis planning . . . . .	56
	<b>Bibliography</b>	<b>58</b>



# List of Figures

1.1	The Gantt diagram of the MSc thesis planning.	3
2.1	Phase I (left) and phase II (right) of the TRIDENT project.	5
2.2	Example of 3-dimensional workspace	7
2.3	Configuration space of a point-sized robot.	7
2.4	Configuration space for a rectangular translating robot.	8
3.1	Sample randomized coverage path.	11
3.2	Classes of cellular decomposition	12
3.3	The path a robot $R$ follows in a non-simply connected environment when applying the algorithm proposed in [Hert et al., 1996].	14
3.4	Sample polygonal configuration space.	15
3.5	Trapezoidal decomposition and associated adjacency graph for the configuration space in figure 3.4.	15
3.6	With fewer cells shorter paths are obtained.	16
3.7	A boustrophedon path is composed of back-and-forth motions.	16
3.8	Trapezoidal decomposition (left) compared to the boustrophedon decomposition (right).	17
3.9	Critical points are points where the connectivity of the slice changes. An slice containing a critical point is termed a critical slice.	17
3.10	Boustrophedon construction process by critical point detection with slice sweeping.	18
3.11	Boustrophedon decomposition of a space and its associated adjacency graph.	18
3.12	Connectivity of $\mathcal{Q}_{\text{free}\lambda}$ on an obstacle intersection.	19

---

3.13	Connectivity of $\mathcal{Q}_{\text{free}\lambda}$ after $\mathcal{Q}_\lambda$ departs an obstacle. . . . .	19
3.14	Morse-based boustrophedon decomposition of a nonpolygonal environment. . . . .	20
3.15	Morse decompositions of a nonpolygonal configuration space obtained using different Morse functions. . . . .	21
3.16	Combination of Morse decomposition and Generalized Voronoi Diagram (GVD) for extended range sensor coverage. . . . .	22
3.17	Illustration of the artificial island technique in three-dimensional space. . . . .	29
4.1	Obstacle perception with a single-beam sounder. . . . .	39
4.2	Obstacle perception with a multibeam sounder. . . . .	40
4.3	Obstacle perception with a multibeam FLS. . . . .	41
4.4	Several sonar images captured with a BlueView P900 FLS at NURC. . . . .	42
5.1	The Formigues islands. . . . .	46
5.2	Bathymetric data recorded in the Formigues islands dataset. . . . .	46
5.3	2D workspace corresponding to the planar section at 7.5 meters depth of the Formigues islands bathymetry. . . . .	47
5.4	Boustrophedon decomposition of the Formigues islands workspace. . . . .	48
5.5	Adjacency graph associated to the boustrophedon decomposition of the Formigues islands workspace. . . . .	48
5.6	The SAUC-E competition arena. . . . .	49
5.7	Underwater pipeline to be inspected in SAUC-E. . . . .	50
5.8	The Sparus AUV. . . . .	51
5.9	Workspace for the SAUC-E pipeline search task plotted in Google Earth. . . . .	51
5.10	Waypoints conforming the generated boustrophedon path for the SAUC-E pipeline search task. . . . .	52
5.11	Generated boustrophedon path for the SAUC-E pipeline search task plotted over the SAUC-E competition arena on Google Earth. . . . .	53

# List of Tables

3.1	Summary of the analyzed cellular decomposition coverage path planning methods.	33
3.2	Summary of the analyzed coverage path planning methods which don't use cellular decomposition.	34
3.3	Summary of the analyzed multi-robot coverage path planning methods.	35
3.4	Summary of the analyzed coverage path planning methods for underwater environments.	36



# Acronyms

<b>APF</b> Artificial Potential Fields .....	23
<b>ASV</b> Autonomous Surface Vehicle .....	2
<b>AUV</b> Autonomous Underwater Vehicle .....	1
<b>CIRS</b> Centre d'Investigació en Robòtica Submarina .....	46
<b>DOF</b> Degree of Freedom .....	51
<b>DVL</b> Doppler Velocity Log .....	51
<b>FLS</b> Forward-Looking Sonar	
<b>GPS</b> Global Positioning System	
<b>GVD</b> Generalized Voronoi Diagram .....	x
<b>I-AUV</b> Intervention Autonomous Underwater Vehicle .....	2
<b>LADAR</b> Laser Detection and Range .....	37
<b>MCM</b> Mine CounterMeasures .....	5
<b>MRU</b> Motion Reference Unit .....	51

---

<b>NURC</b> NATO Undersea Research Center .....	iii
<b>RRT</b> Rapidly-exploring Random Trees .....	26
<b>SAS</b> Synthetic Aperture Sonar .....	29
<b>SAUC-E</b> Student Autonomous Underwater Challenge - Europe .....	49
<b>VICOROB</b> Computer Vision and Robotics Group .....	2
<b>USBL</b> Ultra-Short Base Line .....	6

# Chapter 1

## Introduction

### 1.1 Overview and motivation of the thesis

In the past few decades, the number of Autonomous Underwater Vehicle (AUV)s has increased remarkably. This is largely because of their excellent ability to explore and assess the underwater environment. And as the technology development progresses at a steady pace, the cost for AUVs are decreasing and becoming available to an equally increasing number of people. Whether it is mapping of the ocean floor for oil installations, assessing a naval mine threat or collection of oceanographic data, these vehicles provide users a great resource for better understanding of the ocean in general.

A typical AUV mission consists in surveying an area of interest. Today, the most common way of planning such missions is to use previous knowledge (for instance, a bathymetry\* map) of the area to be explored and program a path consisting on a sequence of waypoints that the vehicle must follow. This approach has two important drawbacks. First, it requires a considerable amount of human intervention for programming the path. Second, some uncharted, unexpected obstacles may come across the vehicle trajectory arising a collision threat. The cost of losing a vehicle due to a collision is unjustifiable both in terms of expenses and replacement time. Therefore, a more automated, more flexible way of specifying those missions is desired.

### 1.2 Research framework

This thesis is located in the research framework of the European research project TRIDENT –Marine Robots and Dexterous Manipulation for Enabling Autonomous Underwater Multipurpose Intervention Missions. Several European universities, institutions and companies work together in this project: Universitat Jaume I de Castellón (Spain), Universitat de Girona

---

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

(Spain), Universitat de les Illes Balears (Spain), Università di Bologna (Italy), Università di Genova (Italy), Instituto Superior Técnico (Portugal), Heriot Watt University (United Kingdom) and Graal Tech (Italy). Universitat de Girona participates in the project through the Computer Vision and Robotics Group (VICOROB).

This project proposes a new methodology to provide multipurpose dexterous manipulation capabilities for intervention operations in unknown, unstructured underwater environments. In the TRIDENT project, a multipurpose generic intervention is composed of two phases. The first phase consists in performing a survey. An Intervention Autonomous Underwater Vehicle (I-AUV) is launched from an Autonomous Surface Vehicle (ASV) towards an area to be surveyed. Then, both vehicles start a coordinated survey path to explore the area. After the survey, an acoustic/optical map of the surveyed area is obtained. Using this map, the end-user selects an object of interest in the map. In the second phase, the intervention phase, the ASV/AUV team navigates towards the selected target position. Once there, the vehicle team perform a search looking for the object of interest. Then, once the object is found, some intervention task (grasping, hooking, etc.) is carried out with target object.

This thesis focus on the first phase of the project, where suitable survey paths need to be planned and executed by the ASV/AUV team in real time as well as on the target object search step of the second phase.

## 1.3 Goal of the thesis

After introducing the research framework, the goal of this thesis is stated. The goal can be summarized as follows:

*To investigate automatic methods capable of determining a path that explores an underwater area by completely covering it while avoiding any obstacles that may arise.*

### 1.3.1 Objectives

The general goal of this thesis can be subdivided into the following, more specific, objectives:

- To exhaustively review the state of the art of coverage survey methods.
- To study methods and available technologies suitable for underwater obstacle detection aiming to use them in an automated survey task.
- To implement and apply coverage path planning methods to real-world underwater datasets in order to extract results as a first step for evaluating their suitability for the application addressed in this thesis.

## 1.4 Planning of the thesis

The followed planning to develop this thesis is shown in the Gantt diagram of figure 1.1. The three outlined objectives of this thesis are addressed by the following planning stages:

- **State of the art** In this stage the state of the art on coverage survey planning is exhaustively reviewed.
- **Study of underwater obstacle detection** In this stage methods and available technologies suitable for underwater obstacle detection aiming to use them in an automated survey task are studied.
- **Experimental work** Here relevant coverage path planning methods are applied to real-world underwater datasets in order to extract results as a first step to test their suitability for the application addressed in this thesis.
- **Research stay at NURC** As part of the experimental work of this thesis, one-month research stay at NATO Undersea Research Center (NURC) is performed in order to collect and analyze acoustic perception datasets using state-of-the art sonar technology.
- **Participation in the SAUC-E competition** SAUC-E is European student competition of AUVs held annually in La Spezia, Italy. The author participates in the competition as part of the University of Girona team, and the event is took also as an opportunity to perform experimental work related to this MSc thesis.
- **Documentation** In the final stage, the present document is elaborated.

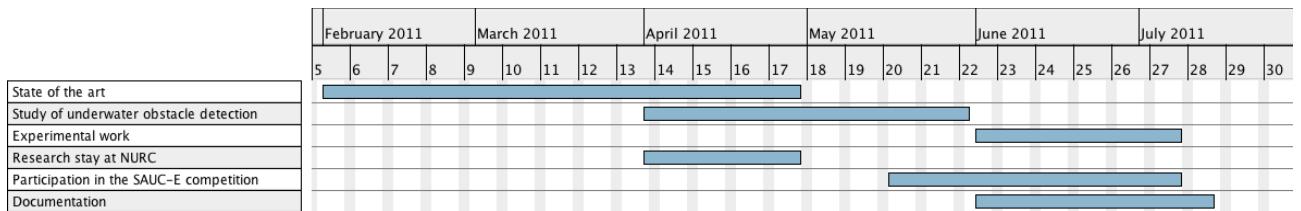


Figure 1.1: The Gantt diagram of the MSc thesis planning.

## 1.5 Outline of the thesis

The outline of this thesis is next presented by giving a brief description of the forthcoming chapters.

- **Chapter 2 Problem definition.** This goes deeper in the problem addressed in this thesis. A more specific description of the survey task aimed to be solved as part of the TRIDENT

project is given. Then, some basic path planning concepts are described in order to introduce the coverage path planning problem, a well-known research problem that is strongly related with the survey task we address in this thesis.

- **Chapter 3** *State of the art.* In this chapter the coverage path planning studies found in the literature are surveyed. First, we discuss general path planning approaches, and then we address the proposals presented in the specific context of underwater environments.
- **Chapter 4** *Underwater obstacle detection.* This addresses the problem of obstacle detection in underwater environments as a key factor to deploy coverage path planning methods in underwater environments. Different methods and available technologies suitable for underwater obstacle detection are discussed. The bulk of this work has been carried out at NURC, where several state-of-the-art sonar datasets were collected and analyzed.
- **Chapter 5** *Preliminary results.* This presents several datasets acquired during the development of this thesis in real underwater environments. Then, relevant coverage path planning methods are applied on the datasets in order to extract results as a preliminary step towards assessing their suitability for the application addressed in this thesis.
- **Chapter 6** *Conclusion.* This concludes the thesis by summarizing the work and pointing out the contributions made in this research project. It comments further work that may succeed this thesis as well.

# Chapter 2

## Problem definition

### 2.1 Problem context

Today, many robotic applications involve surveying an area in a manner such that the vehicle passes over all points in that area. Just to name a household example, think in the Roomba® cleaning robots from iRobot Corporation recently put on the market and similar products from other companies as well. But the list goes beyond: swimming pool cleaning robots, painter robots, demining robots, land mine detectors, lawn mowers, automated harvesters, window cleaners, and so on. In the case of underwater robotics, many missions also involve covering an area: image mosaicking, bathymetry mapping and Mine CounterMeasures (MCM) operations just to name a few.

As explained in section 1.2, this thesis is located in the research framework of the TRIDENT research project. This project proposes a multipurpose generic intervention mission in unknown underwater environments composed of two phases. A diagram describing the project phases and their specific steps is shown in figure 2.1. Next we overview the project steps:

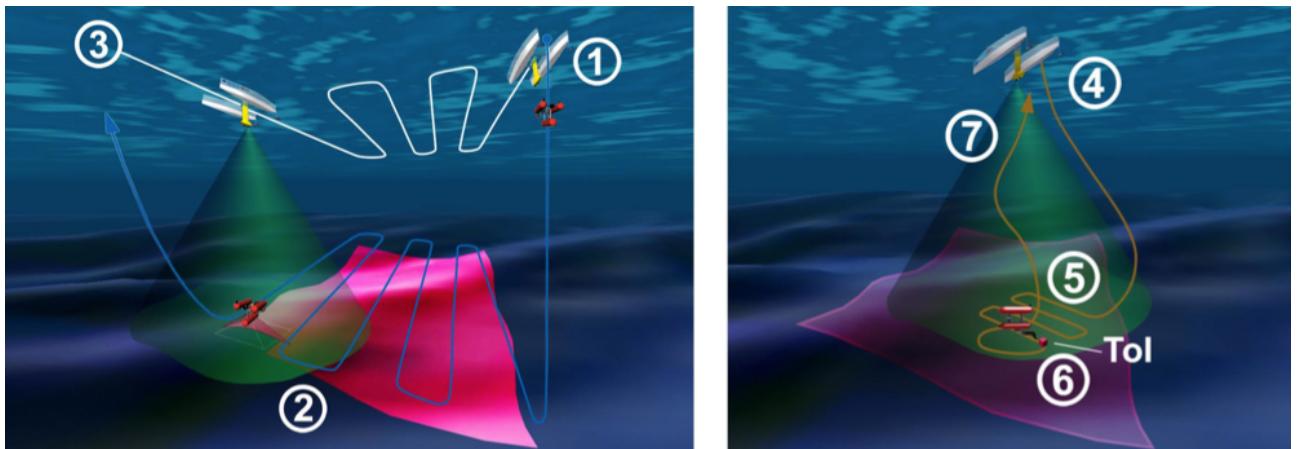


Figure 2.1: Phase I (left) and phase II (right) of the TRIDENT project.

- **Phase I (Survey):** The ASV is launched to carry the I-AUV towards the area to be surveyed. Then, the I-AUV is deployed (1) and both vehicles start a coordinated survey path (2) to explore the area. The ASV/AUV team gathers navigation data for georeferencing the measurements (seabed images and multibeam bathymetry profiles). Finally, the I-AUV surfaces (3) and contacts the end user, to whom an acoustic/optical map of the surveyed area is presented. Using this map, the end user selects a target object (an object of interest) as well as a suitable intervention task (grasping, hooking, etc.).
- **Phase II (Intervention):** After selecting the target, the ASV/AUV team navigates towards the target position. Then, the ASV performs dynamic position (4) while keeping the I-AUV inside the Ultra-Short Base Line (USBL) cone of coverage. Then, the I-AUV performs a search (5) looking for the target of interest. When the object appears in the robot field of view, it is identified and the I-AUV switches to free floating mode using its robotic arm as well as the dexterous hand to do the smart manipulation (6). Finally (7), the I-AUV docks to the ASV before recovery.

In particular, this thesis focus on the survey phase, where an underwater area needs to be explored and mapped, as well as on the target object search on the intervention phase. Therefore, we aim to survey an area with an AUV in order to record data from it and, later on, in an offline fashion, detect some features of interest on it. After that, a search all across the area needs to be performed. To accomplish these tasks, we need the vehicle to cover all the points in that area. Here, two important considerations need to be taken into account. First, the robot must plan the path to follow automatically, without human intervention explicitly providing a list of waypoints to go through. This way the amount of human aid is reduced and so are the mission expenses. Second, the robot must be able to avoid any obstacles that may come across its path. As a secondary consideration, we need to take into account that some applications like image mosaicking may require the robot to go through the same point more than once or to draw a specific pattern in order to do feature matching.

## 2.2 Brief introduction to path planning concepts

Prior to go into a more detailed description of the problem we address in this thesis, we give a brief introduction to some basic path planning concepts. This will ease the understanding of this document from now on, as we also introduce the terminology that will be used hereafter. Thus, next the concepts of *workspace* and *configuration space* are discussed.

### 2.2.1 Configuration space

A basic path planning problem is to produce a continuous path that connects a start configuration  $S$  and a goal configuration  $G$ , while avoiding collision with known obstacles. The robot

and obstacle geometry is described in a 2D or 3D *workspace*, while the motion is represented as a path in (possibly higher-dimensional) *configuration space*.

A configuration describes the pose of the robot, and the configuration space  $C$  is the set of all possible configurations. For example:

- If the robot is a single point (zero-sized) translating in a 2-dimensional plane (the workspace),  $C$  is a plane, and a configuration can be represented using two parameters  $(x, y)$ .
- If the robot is a 2D shape that can translate and rotate, the workspace is still 2-dimensional. However,  $C$  is the special Euclidean group  $SE(2) = R^2 \times SO(2)$  (where  $SO(2)$  is the special orthogonal group of 2D rotations), and a configuration can be represented using 3 parameters  $(x, y, \theta)$ .
- If the robot is solid 3D shape that can translate and rotate, the workspace is 3-dimensional, but  $C$  is the special Euclidean group  $SE(3) = R^3 \times SO(3)$ , and a configuration requires 6 parameters:  $(x, y, z)$  for translation, and Euler angles  $(\alpha, \beta, \gamma)$ .
- If the robot is a fixed-base manipulator with  $N$  revolute joints (and no closed-loops),  $C$  is  $N$ -dimensional.

See figures 2.2, 2.3 and 2.4 for an illustration.

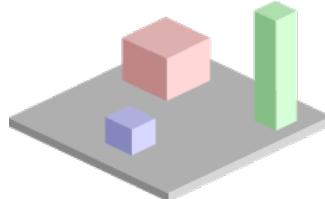


Figure 2.2: Example of 3-dimensional workspace.

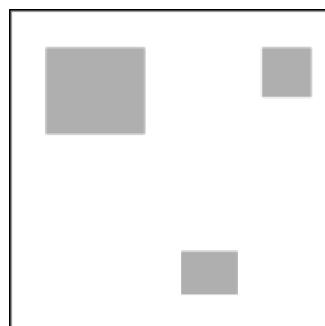


Figure 2.3: Configuration space of a point-sized robot. White =  $C_{free}$ , gray =  $C_{obs}$ .

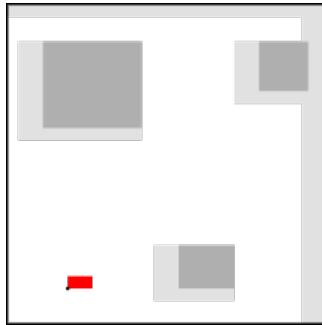


Figure 2.4: Configuration space for a rectangular translating robot (pictured red). White =  $C_{free}$ , gray =  $C_{obs}$ , where dark gray = the objects, light gray = configurations where the robot would touch an object or leave the workspace.

### 2.2.2 Free space

The set of configurations that avoids collision with obstacles is called the free space  $C_{free}$ . The complement of  $C_{free}$  in  $C$  is called the obstacle or forbidden region  $C_{obs}$ . Often, it is prohibitively difficult to explicitly compute the shape of  $C_{free}$ . However, testing whether a given configuration is in  $C_{free}$  is efficient. First, forward kinematics determine the position of the robot's geometry, and collision detection tests if the robot's geometry collides with the environment's geometry.

## 2.3 The coverage path planning problem

The task of determining a path that passes an effector (e.g., a robot, a detector, etc.) over all points in a free space is called the coverage path planning problem [Choset et al., 2005]. This allows for an exhaustive exploration of an area of interest without specifying a concrete waypoint trajectory while in turn avoiding any obstacles that may damage the effector. As the reader will notice, this problem is closely related to the survey task we address in this thesis.

## 2.4 Problem statement

Aiming to autonomously accomplish the task introduced in section 2.1, in this thesis we address *the coverage path planning problem in the particular case of underwater robotic operations*. This, in turn, arises the problem of detecting obstacles that may arise along the path and that must be avoided. Therefore, in chapter 3 we exhaustively review the studies on coverage path planning found in the literature. Then, in chapter 4, we analyze methods and available technologies suitable for underwater obstacle detection aiming to use them in an automated survey task.

# Chapter 3

## State of the art

In this chapter, we survey the coverage path planning studies found in the literature. In our survey, we first focus on approaches to coverage path planning as a general problem, that is, without paying special attention to any specific subfield of Robotics (sections 3.1-3.4). Then, we focus on coverage path planning approaches that take into account the particularities of underwater environments (section 3.5). Last, we discuss the surveyed methods (section 3.6).

Coverage path planning (also called region filling or area covering, in 2-D environments) is needed in several robotic applications, such as vacuum cleaning robots [Yasutomi et al., 1988], painter robots [Atkar et al., 2001], autonomous underwater covering vehicles [Hert et al., 1996], demining robots [Gage, 1994], land mine detectors [Najjaran and Kircanski, 2000], lawn mowers [Cao et al., 1988], agricultural crop harvesting equipment [Ollis and Stentz, 1996], automated harvesters [Ollis and Stentz, 1997], and window cleaners [Farsi et al., 1994]. The research interest in mobile robotics (indoors and outdoors) has clearly motivated the research of coverage path planning. [Cao et al., 1988] define the criteria for the region filling operation (for a mobile robot) as follows:

1. Robot must move through an entire area (cover the whole area).
2. Robot must fill the region without overlapping path.
3. Continuous and sequential operation without any repetition of paths is required.
4. Robot must avoid all obstacles.
5. Simple motion trajectories (e.g., straight lines or circles) should be used (for simplicity in control).
6. An “optimal” path is desired under available conditions.

As [Cao et al., 1988] note, “It is not always possible to satisfy all these criteria for complex environments. Sometimes a priority consideration is required.”

The coverage path planning problem is related to the covering salesman problem, a variant of the traveling salesman problem where, instead of visiting each city, an agent must visit a neighborhood of each city. This minimizes the length of travel for the agent [Arkin and Hassin, 1994; Choset, 2001]. As is well known, the traveling salesman problem is NP-hard. Thus, the computational time to solve the problem increases drastically when the dimension of the problem increases. Actually, [Arkin et al., 1993] have proved with the “lawnmower problem” that the problem of coverage path planning is NP-hard.

Coverage algorithms can be classified as *heuristic* or *complete* depending on whether or not they provably guarantee complete coverage of the free space. At the same time, they can be classified as *offline* or *online*. Offline algorithms rely only on the stationary information, and the environment is assumed to be known. Usually online algorithms are needed if some kind of adaptivity to the environment is required. Online algorithms usually utilize real-time sensor measurements. Thus, these algorithms can also be called *sensor-based coverage algorithms*.

Many studies on coverage path planning can be found in the literature. Also, some authors have surveyed those studies in the past. [Choset, 2001] conjectured in his survey that most complete (that is, not heuristic) coverage algorithms used an exact cellular decomposition, either explicitly or implicitly, to achieve coverage. Thus, he organized the coverage algorithms into four categories: heuristic and approximate, partial-approximate and exact cellular decompositions.

One decade later, after our review, we share this conjecture. Nonetheless, several methods found in the literature don't use a cellular decomposition at all. Therefore, we also provide a category for such methods. Thus, we classify the analyzed methods in the following categories: heuristic and randomized approaches (section 3.1), cellular decompositions (3.2), other approaches (3.3, which includes as subcategories template-based models, potential fields, neural networks and fuzzy logic and miscellaneous approaches) and multi-robot coverage approaches (3.4). Finally, and taking into account the specific requirements of the problem addressed in this thesis, we focus in coverage methods for underwater environments (3.5).

It is worth noticing that, to the best of the author's knowledge, no updated surveys on coverage path planning exist in the literature. Therefore, the review presented in this chapter contributes significantly to this issue.

### 3.1 Heuristic and randomized approaches

One approach to solve the problem is to randomize. This is an approach that some floor-cleaning robots rely on. If you sweep the floor randomly for long enough, it should become cleaned. Figure 3.1 shows a sample random coverage path.

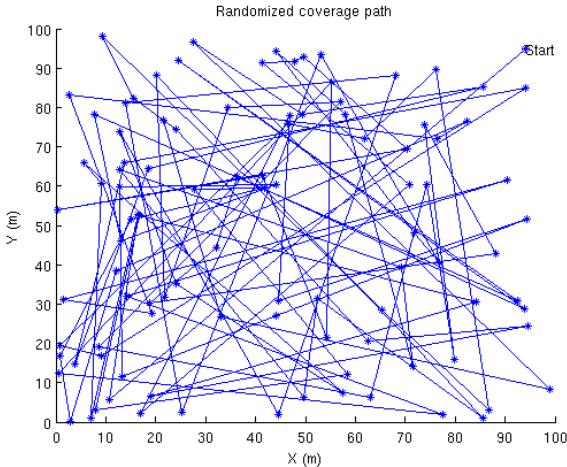


Figure 3.1: Sample randomized coverage path.

There are advantages to this approach (see [Choset, 2001], and references therein), the main one being that no sensors are needed for localization. The only sensor required is the one to detect the hit to boundaries. On the other hand, no expensive computer running complex algorithms is necessary onboard. [Choset, 2001] concludes that if a robot with a random algorithm can be constructed at 1/5 of the price of one with localization and advanced path planning, it is cost efficient. However, for covering vast areas, and especially for underwater or aerial robotics operations, it is difficult to think that a randomized “algorithm” could be usable, as the cost of operating the vehicle (energy and time) would be unaffordable, even greater than the cost of construction itself.

## 3.2 Cellular decompositions

Many algorithms either implicitly or explicitly use cellular decomposition of the free space to achieve coverage.

In cellular decomposition, the free space is broken into simple regions that can be covered using simple motions (like straight lines or circles), which should guarantee the coverage. The shared boundaries of cells often have a physical meaning such as a change in the closest obstacle or a change in line of sight to surrounding obstacles. Two cells are *adjacent* if they share a common boundary. An *adjacency graph*, as its name suggests, represents the adjacency relationships of the cells, where a node corresponds to a cell and an edge connects nodes of adjacent cells.

The cellular decomposition algorithms can be classified into three classes: approximate, semi-approximate, and exact [Choset, 2001; Latombe, 1991]. See figure 3.1.

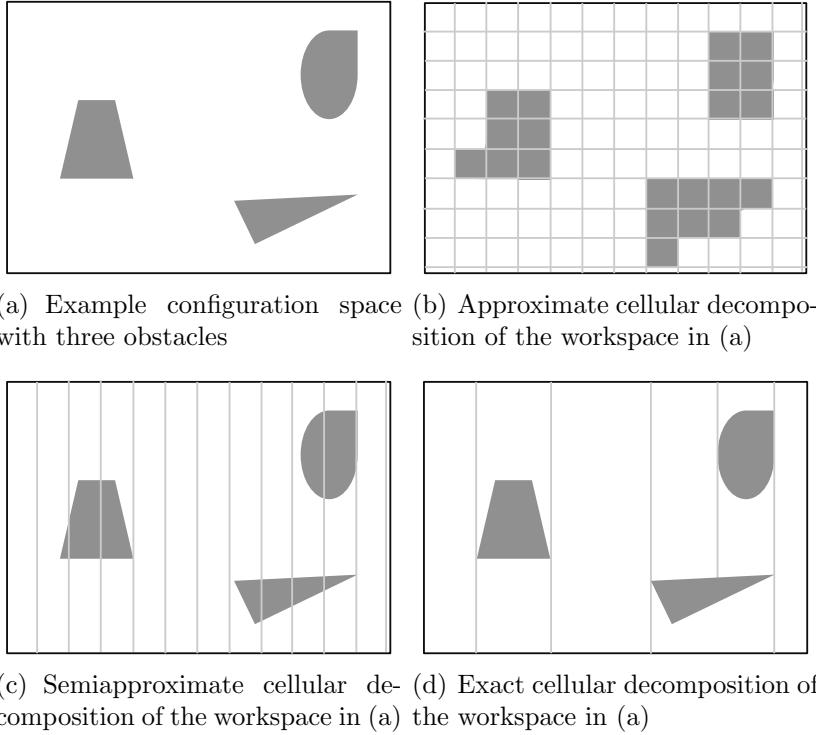


Figure 3.2: Classes of cellular decomposition on the configuration space shown in (a): approximate (b), semiapproximate (c) and exact (d).

### 3.2.1 Aproximate cellular decomposition

In approximate cellular decomposition, the region is approximated with a fine-grid based representation of the free space that covers the region, and the algorithm is applied to the grid. Here, the cells are all of the same size and shape so that the union of the cells only approximates the target region [Spires and Goldsmith, 1998].

[Moravec and Elfes, 1985] first proposed an approximate cellular decomposition model, where the workspace is decomposed into cells with the same size and shape.

[Zelinsky et al., 1993] developed another approximate cellular decomposition approach using a grid based complete coverage model. A distance transform algorithm is used to assign a specific number to each grid element. The complete coverage is then achieved by a gradient descent rule. The main problem in that algorithm is that it does not take into account kinematic constraints.

Also as an approximate cellular decomposition approach, [Gabriely and Rimon, 2002] propose the the Spiral-STC algorithm, which consists in subdividing the workspace into discrete cells and following a spanning tree of a graph induced by the cells. The robot is able to cover every point precisely once, and travel an optimal path in a grid-like representation in the workspace. A similar approach is the BSA [Gonzalez et al., 2005]. As a remarkable advantage, they propose

an extension to cover not only unoccupied cells, but also the partially occupied ones. This extension can be applied to most approximate cell decomposition algorithms.

[Choi et al., 2009] propose an online complete coverage path planning solution for mobile robots in which simple spiral paths are traced based on active wall-finding using the history of sensor readings and then linked using the inverse distance transform.

### 3.2.2 Semiapproximate cellular decomposition

The semiapproximate cellular decomposition relies on a partial discretization of the space where the cells are fixed in width but their tops and bottoms can have any shape. [Hert et al., 1996] present a coverage algorithm which applies this kind of decomposition. Their solution applies to a 3D projectively planar environment by extending a 2D terrain-covering algorithm. Their 2D terrain-covering algorithm applies to both simply and nonsimply connected environments. A robot following this algorithm may start at an arbitrary point in the environment and will zigzag along parallel straight lines (*grid lines*) to cover the given area. Portions of the area that either would not be covered or would be covered twice using the zigzag procedure are detected by the robot and covered using the same procedure; that is, the procedure is applied recursively. These smaller areas, called *inlets*, are covered as soon as they are detected and inlets within inlets are treated in the same way. Hence, the inlets are covered in a depth-first order. By requiring the robot to remember the points at which it enters and exits every inlet it covers (which define the inlet *doorways*), the algorithm assures that each inlet is covered only once.

When entering or exiting a certain type of inlet, the robot may cover the same area more than once, or miss some area at the inlet. Those inlets are called *diversion inlets*, and special procedures are necessary for efficiently covering them. The robot enters a diversion inlet by moving along its boundary. After covering a given diversion inlet, the robot exits it by resuming its path as if the diversion inlet did not exist. When the area to be covered is not simply connected and contains islands as well as inlets, the same basic procedures are used, but with minor modifications to ensure that the area surrounding every island is covered. The robot is able to convert the part of the area around each island that would normally not be covered into an artificial inlet by remembering certain points along its path. Artificial inlets are covered in the same way that real diversion inlets are. See figure 3.2.

### 3.2.3 Exact cellular decomposition

The exact cellular decomposition divides the target environment into a set of non-intersecting cells of arbitrary shape, whose union fills the target environment, and then, the robot searches the connectivity graph that represents the adjacency relation among cells. Each cell is then typically covered using simple motions, like back-and-forth motions.

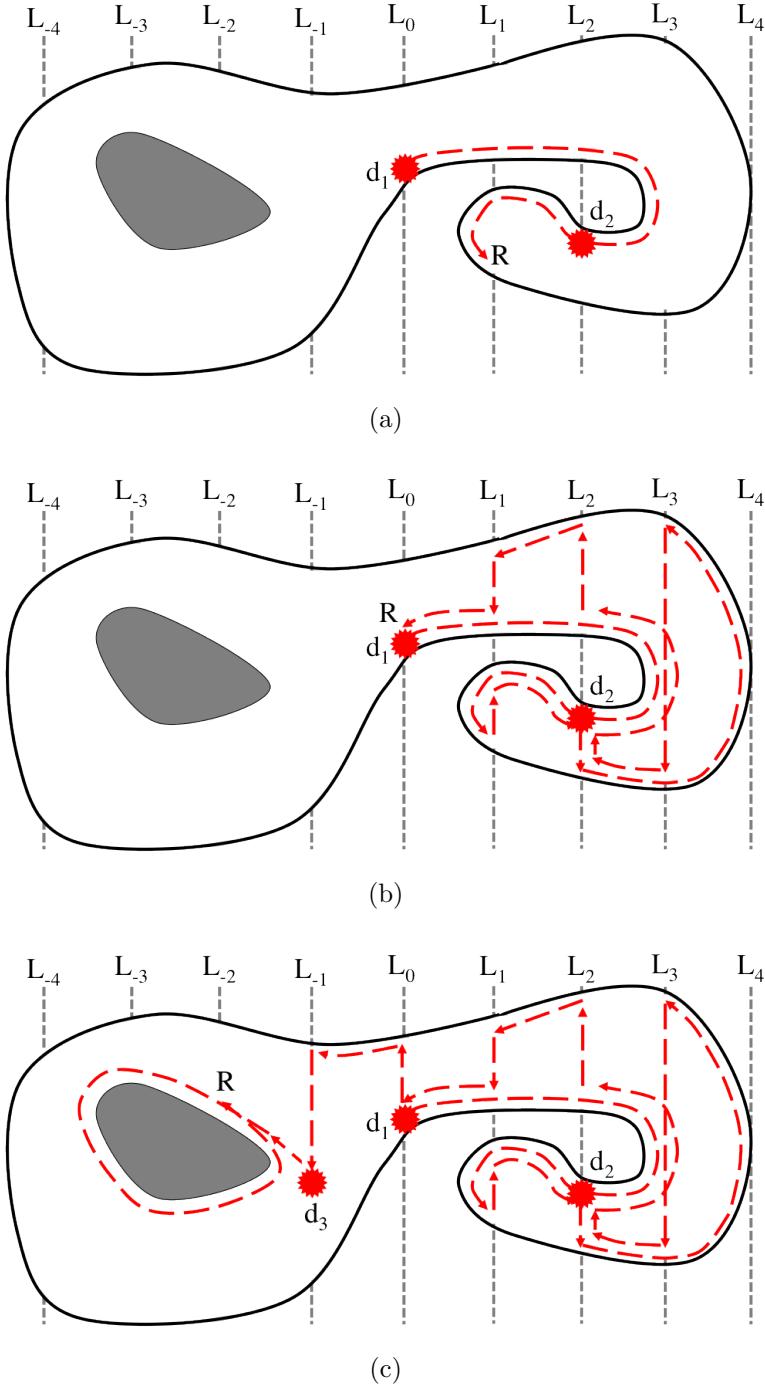


Figure 3.3: The path a robot  $R$  follows in a non-simply connected environment when applying the algorithm proposed in [Hert et al., 1996]. First (a), the robot detects an inlet at  $d_1$  and starts to cover it following a depth-first order. A second inlet is detected at  $d_2$ , and the robot starts covering it likewise. The robot continues to cover the rest of the inlets until it goes back to  $d_1$  (b). Here, the robot continues to cover the main region until it detects an inlet at  $d_3$ . This inlet corresponds to an island, and hence the robot continues to circumnavigate it completely (c). Then, the robot will eventually pass through  $d_3$  again and there it will resume the covering of the main area.

## Trapezoidal decomposition

One popular exact cellular decomposition technique, which can yield a complete coverage path solution, is the trapezoidal decomposition [Latombe, 1991] in which the robot's free space is decomposed into trapezoidal cells. Since each cell is a trapezoid, coverage in each cell can easily be achieved with simple back-and-forth motions. However, this technique requires the obstacles to be polygonal, and can only be applied offline with a priori knowledge of the environment.

Figure 3.4 shows a sample polygonal workspace. The trapezoidal decomposition and the associated adjacency graph for this workspace is shown in figure 3.5. Recall that two cells are adjacent if they share a common boundary, i.e., a common vertical extension. Then, the adjacency graph can be searched to determine the coverage path. To achieve complete coverage, an exhaustive walk on the graph (a Hamiltonian path), must be determined, i.e., a path that visits each vertex in the graph exactly once. However, it is worth noticing that the result of the graph search is just a sequence of nodes, not a sequence of points embedded in the free space to go through. Therefore, an explicit path for each one of the cells must be derived.

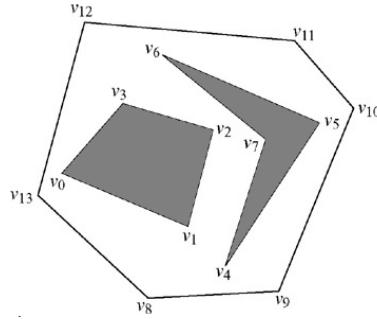


Figure 3.4: Sample polygonal configuration space.

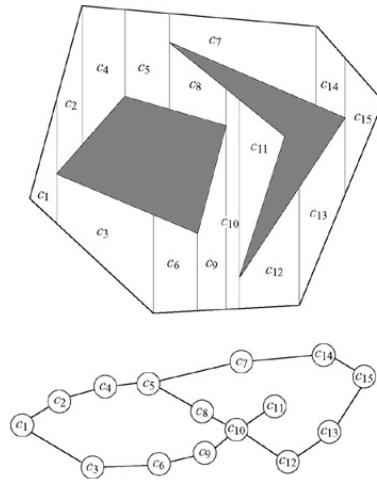


Figure 3.5: Trapezoidal decomposition and associated adjacency graph for the configuration space in figure 3.4.

## Boustrophedon decomposition

A shortcoming of the trapezoidal decomposition is that many small cells are formed that can seemingly be aggregated with neighboring cells. Observe that cells in the trapezoidal decomposition can be “clumped” together to form shorter and more efficient coverage paths. For example, in the left side of figure 3.6 the robot needs no make one additional lengthwise motion to achieve complete coverage.

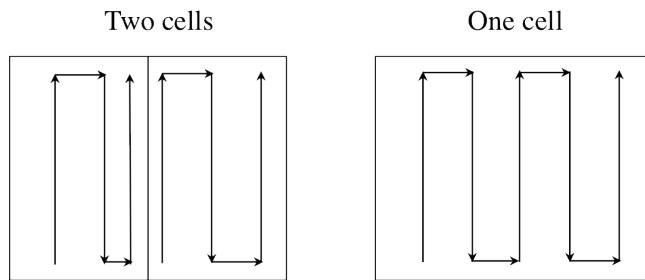


Figure 3.6: With fewer cells shorter paths are obtained.

To address this issue, [Choset and Pignon, 1997] introduced the boustrophedon —the word coming from Greek, “the way an ox drags a plough” (see figure 3.7)— cell decomposition approach, where back-and-forth motions are used. The boustrophedon decomposition is formed by considering the vertices at which a vertical line can be extended both up and down in the free space (figure 3.8). Such vertices are called *critical points*. A slice containing a critical point is termed a *critical slice*. See figure 3.9.

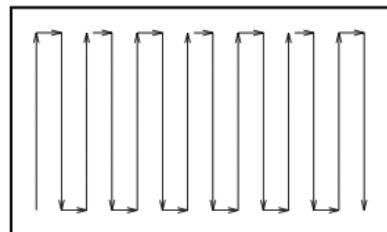


Figure 3.7: A boustrophedon path is composed of back-and-forth motions.

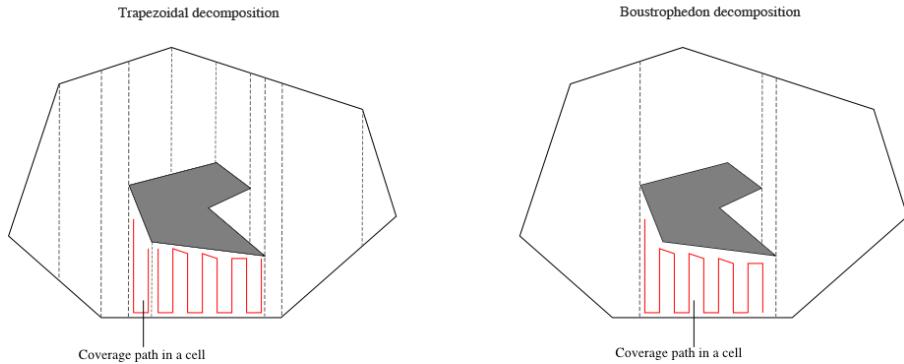


Figure 3.8: Trapezoidal decomposition (left) and boustrophedon decomposition (right) for the same space. Each cell, in both decompositions, can be covered with simple back and forth motions. However, note that the coverage path in the boustrophedon decomposition is a little bit shorter. The region below the polygon obstacle requires an extra pass because the planner has to “start over” each time the robot enters a new cell. Since there are fewer cells under the polygonal obstacle in the boustrophedon decomposition, the coverage path is shorter.

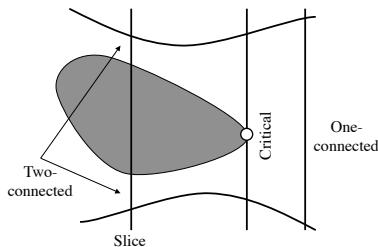


Figure 3.9: Critical points are points where the connectivity of the slice changes. An slice containing a critical point is termed a critical slice.

To generalize beyond polygons, they sweep a vertical slice through the configuration space and look for connectivity changes in this slice. They use a change in connectivity of the slice to indicate the boundary location of the cells: when connectivity increases, new cells are spawned (figure 3.10(a)); when connectivity decreases, multiple cells are completed and single cell is created (or cells disappear) (figure 3.10(b)).

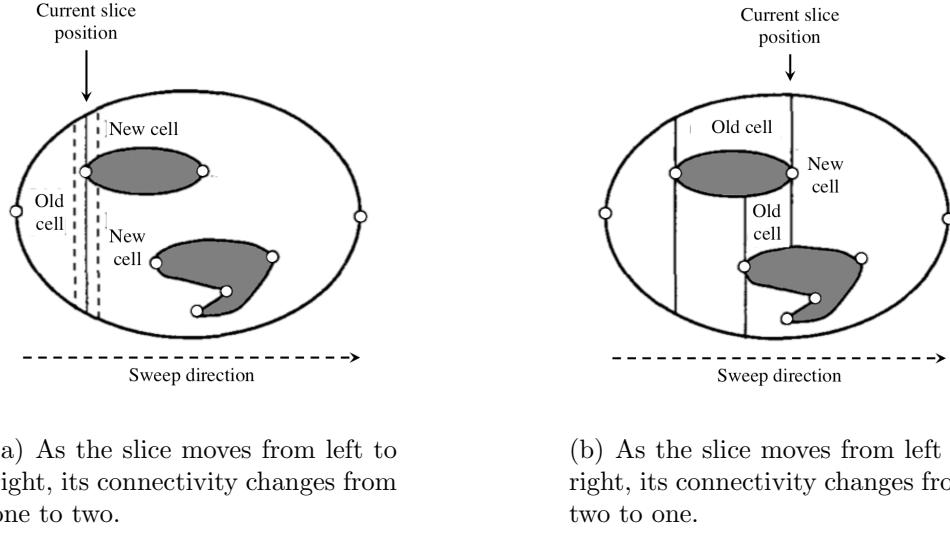


Figure 3.10: Boustrophedon construction process by critical point detection with slice sweeping.

Once the cell decomposition is determined, the planner achieves coverage in two steps. First, it determines an exhaustive walk through the adjacency graph (figure 3.11). The walk can be computed by means of a depth-first algorithm. Then, the planner determines the explicit robot motions within each cell. These paths consists in a repeated sequence of back-and-forth motions.

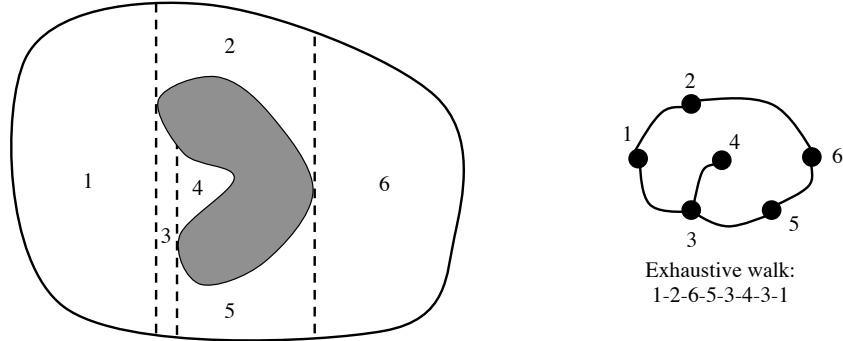


Figure 3.11: Boustrophedon decomposition of a space and its associated adjacency graph. Nodes represent cells and edges indicate a relationship of adjacency between cells. An exhaustive walk through the graph is calculated.

Though this method does not require the obstacles to be polygonal, it can only be applied with prior knowledge of the environment, i.e., it is an offline method.

### Morse cellular decompositions

[Choset et al., 2000] generalized the boustrophedon decomposition by proposing a novel bous-

trophedon cellular decomposition approach where the decomposition is based on critical points of Morse functions [Milnor, 1963]. In fact, they show that the boustrophedon decomposition is a particular case of a Morse decomposition.

Morse functions are those whose critical points are nondegenerate\*. In this method, Morse functions are used as slices that are swept through the space. Formally, “a *slice* is a codimension one manifold denoted by  $\mathcal{Q}_\lambda$ . The slices are parametrized by  $\lambda$  (varying  $\lambda$  sweeps a slice through the space). The portion of the slice in the free configuration space,  $\mathcal{Q}_{\text{free}}$ , is denoted by  $\mathcal{Q}_{\text{free}\lambda}$ , i.e.,  $\mathcal{Q}_{\text{free}\lambda} = \mathcal{Q}_\lambda \cap \mathcal{Q}_{\text{free}}$ ” [Choset et al., 2005]. Connectivity changes in  $\mathcal{Q}_{\text{free}\lambda}$  are used to define cells in the cell decomposition.

A slice can be defined in terms of the preimage of a real-valued Morse function  $h: \mathcal{Q} \rightarrow \mathbb{R}$ . In the particular case of the boustrophedon decomposition, which happens in the plane,  $h(x, y) = x$  and the slice  $\mathcal{Q}_\lambda = h^{-1}(\lambda)$  corresponds to a vertical slice. Increasing the value of  $\lambda$  sweeps the slice from left to right through the configuration space. As the slice sweeps the space it intersects (or stops intersecting) obstacles, which sever it into smaller pieces as the slice first encounters an obstacle (figure 3.12). Also, immediately after the slice departs an obstacle, smaller pieces are merged into larger pieces (figure 3.13). The points where connectivity changes occur are the critical points (notice those are the same critical points introduced in the description of the “non-Morse” boustrophedon decomposition).

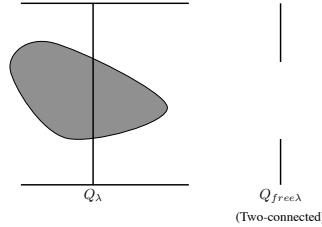


Figure 3.12: Connectivity of  $\mathcal{Q}_{\text{free}\lambda}$  on an obstacle intersection. Here, an obstacle intersects  $\mathcal{Q}_\lambda$  and  $\mathcal{Q}_{\text{free}\lambda}$  is hence two-connected.

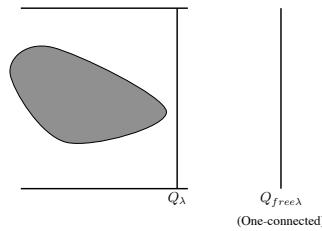


Figure 3.13: Connectivity of  $\mathcal{Q}_{\text{free}\lambda}$  after  $\mathcal{Q}_\lambda$  departs an obstacle. Here,  $\mathcal{Q}_{\text{free}\lambda}$  becomes one-connected.

Figure 3.14 shows a Morse-based boustrophedon decomposition of a space where the dashed

---

\*Recall that in the case of a function of a single real variable,  $f(x)$ , a critical point is a value  $x_0$  in the domain of  $f$  where either the function is not differentiable or its derivative is 0,  $f'(x_0) = 0$ . Also in the case of a single-variable real function, a critical point is called nondegenerate if its second derivative is nonzero.

lines represent the slice intervals lying in the free space. Each of these slice intervals has at least one critical point on an obstacle boundary.

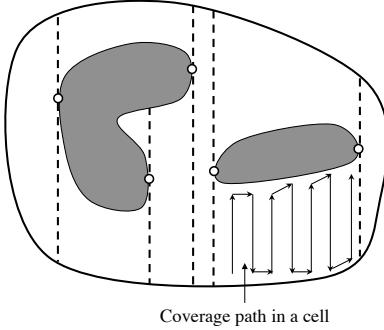


Figure 3.14: Morse-based boustrophedon decomposition of a nonpolygonal environment. As a vertical slice is swept from left to right, its connectivity in the free space changes. At the points where these connectivity changes occur (the critical points, shown here as black-bordered white circles), the cell boundaries in the free space are located.

“One can see that within a cell, the slice interval remains connected and only extends or contracts. Morse theory assures us that between critical slices, ‘merging’ and ‘severing’ of slices do not occur, i.e., the topology of the slice remains constant.” [Choset et al., 2005] This allows a robot to guarantee complete coverage of a cell by just performing simple motions between critical points (3.14).

As with the other previously introduced exact cell decompositions, using Morse decompositions a planner also achieves coverage in two parts. First, it determines an exhaustive walk through the adjacency graph. Then, it plans the explicit coverage in each uncovered cell. Here, the coverage pattern within each cell has three parts: motion along a slice, motion orthogonal to the slice, and motion along the cell boundary. First, the robot performs a swath along the current slice,  $Q_\lambda$ . Then the robot steps outward of the slice by going orthogonally to it by an interlap distance, typically by a distance of one robot size;  $\lambda$  is also increased by this distance to form a new slice. If the robot encounters an obstacle (i.e., the cell boundary) while moving along the slice, the planner directs the robot to follow the obstacle boundary until it has moved an interlap distance and then a new swath along a slice is started. The process repeats until the cell is completely covered. Therefore, the shape of the slice also determines the explicit coverage patterns within cells apart from the decomposition itself.

A key point of Morse decompositions is that by choosing different Morse functions to define the slice that is swept through the space produces different decomposition patterns, like the spiral, spike and squarel patterns [Acar et al., 2002]. Figure 3.15 shows different Morse decompositions obtained by using different Morse functions.

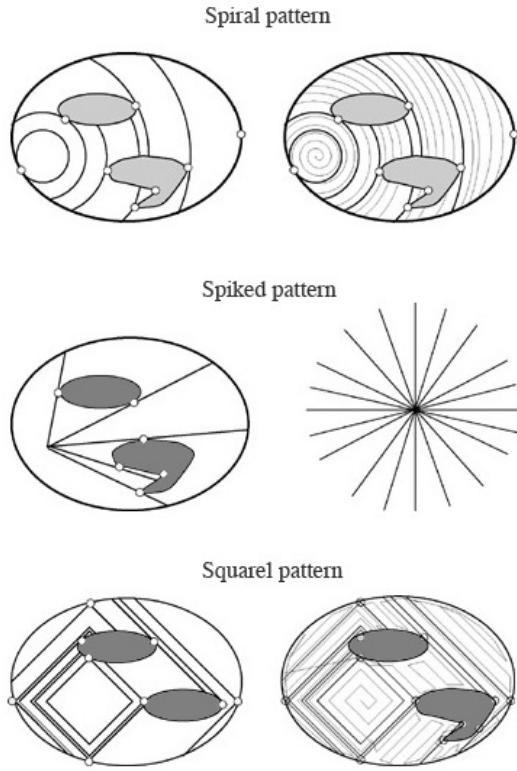


Figure 3.15: Morse decompositions of a nonpolygonal configuration space obtained using different Morse functions. From top to bottom:  $h(x, y) = \sqrt{x^2 + y^2}$  produces the spiral pattern;  $h(x, y) = \tan \frac{x}{y}$  produces the spiked pattern;  $h(x, y) = |x| + |y|$  produces the squarel pattern.

In contrast to the “classic” boustrophedon decomposition presented in [Choset and Pignon, 1997], Morse decompositions can be applied to nonpolygonal spaces and to high-dimensional spaces as well. However, the method proposed here requires the prior knowledge of the obstacle locations and the critical points, i.e., it is an offline method.

### Sensor-based coverage using Morse decompositions

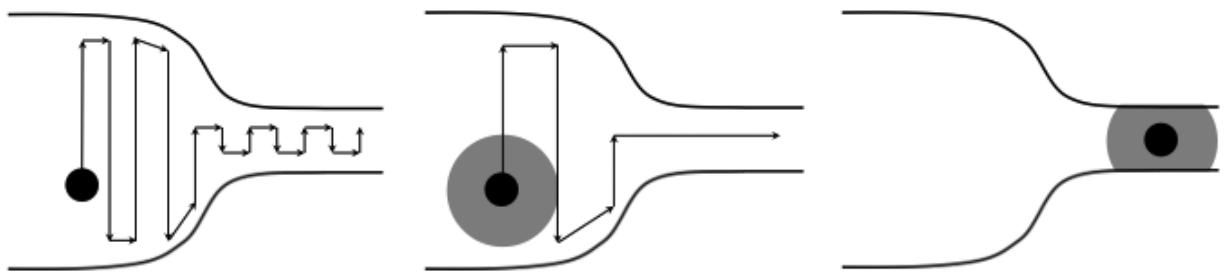
[Acar and Choset, 2000, 2002] later proposed a method for online critical point detection to ensure the covering task using Morse decomposition does not require any prior knowledge. Here, they incrementally construct the decomposition using range sensor information. Their critical point detection method is based on the gradient of the slice and the surface normals. [Atkar et al., 2001] applied the Morse decomposition to complete coverage of a closed orientable surface in three dimensions for a robot painting task. Here, they also propose a method for critical point detection. This critical point detection method is based on connectivity changes in the free space. For a more detailed description of these critical point detection methods, see [Acar and Choset, 2000], [Acar and Choset, 2002] and [Atkar et al., 2001].

As [Choset, 2000] advances, boustrophedon decomposition method may fail to detect some crit-

ical points in nonconvex obstacles. Later, [Garcia and de Santos, 2004] proposed an improvement to that algorithm that allows also for online coverage of unknown spaces with nonconvex obstacles.

### Sensor-based coverage with extended range detectors combining Morse cellular decomposition and GVD

[Acar et al., 2006] presented a sensor-based coverage with extended range sensors approach. As they point out, “prior work in coverage tends to fall into one of two extremes: coverage with an effector the same size of the robot, and coverage with an effector that has infinite range.” In this work, they consider coverage in the middle of this spectrum: coverage with a detector range that goes beyond the robot, and yet is still finite in range. They term these sensors *extended range sensors*. They achieve coverage in two steps: the first step considers vast, open spaces, where the robot can use the full range of its detector; the robot covers these spaces as if it were as big as its detector range. Here previous work in using Morse cell decompositions [Acar et al., 2002] is employed to cover unknown spaces. As explained above, cell in this decomposition can be covered via simple back-and-forth motions, and coverage of the vast space is then reduced to ensuring that the robot visits each cell in the vast space. The second step considers the narrow or cluttered spaces where obstacles lie within detector range, and thus the detector “fills” the surrounding area. In this case, the robot can cover the cluttered space by simply following the Generalized Voronoi Diagram (GVD) of that space. To sum up, hierarchical decomposition that combines the Morse decompositions and the GVDs is introduced to ensure that the robot indeed visits all vast, open, as well as narrow, cluttered, spaces. See figure 3.15. It is shown how to construct this decomposition online with sensor data that is accumulated while the robot enters the environment for the first time.



(a) Coverage path followed by a robot with a robot-sized detector. (b) Coverage path followed by a robot with an extended range detector. (c) Sensor coverage in a cluttered space.

Figure 3.16: Combination of Morse decomposition and GVD for extended range sensor coverage. A robot with a robot-sized detector 3.16(a) must perform a multiple swath pattern (i.e. perform back-and-forth motions) both in a vast area (left) and in a cluttered area (right) to achieve complete sensor coverage. However, a robot with an extended range detector 3.16(b) has to perform multiple swaths only in the vast area (left), as in the cluttered area (right) its sensor range is infinite in practice, as shown in 3.16(c).

### Other exact cell decomposition approaches

[Cao et al., 1988] proposed an approach to fill the whole region with obstacle avoidance for a lawn mower robot using a raster scanning strategy in which they implicitly used a Morse decomposition. This region-filling algorithm requires the boundary of obstacles and walls in advance, and they assume all obstacles are convex.

The Morse decompositions, described above, assume that the critical points are not degenerate, but [Butler et al., 1999] developed an algorithm termed  $CC_{RM}$  that uses decompositions of rectilinear spaces where all critical points are degenerate. The algorithm incrementally constructs a cellular decomposition of the environment ( $C$ ). After each straight-line trajectory is executed by the robot,  $CC_{RM}$  chooses a new trajectory based only on  $C$  and its current position. The trajectory is determined by a list of rules that are designed to continue coverage in all possible cases. Completeness of  $CC_{RM}$  is shown by creating a finite state machine (FSM) that describes all ways in which  $C$  can evolve under  $CC_{RM}$ , and demonstrating that the FSM has no infinite loops and terminates only when coverage is complete. The decomposition of  $CC_{RM}$  can be viewed as the boustrophedon decomposition that has only degenerate critical points. In other words, a slice is passed through the robot's free space. When the slice changes connectivity or its length, a new cell is formed.

[Huang, 2001] proposed an offline line-sweep cellular decomposition of the space which allows for assigning a different line-sweep direction to each cell in order to optimize some cost such as time.

[Acar et al., 2003] discuss coverage path planning in relation to a demining application. In that article, the vehicle is considered omnidirectional, and two algorithms are discussed: exact cellular decomposition with back-and-forth motion and a probabilistic method with online sensor data. In this approach is shown that online cellular decomposition in terms of critical points of Morse functions [Choset et al., 2000] outperform random coverage, which used to be considered state of the art for mine clearance.

Also in the context of exact cellular decompositions, [Wong and MacDonald, 2003] presented a topological coverage algorithm for mobile robots based on natural landmarks detection. In contrast to [Acar and Choset, 2002], here corners are used as landmarks and retracing is not required, so shorter coverage paths are generated. Nonetheless, complete coverage is not guaranteed.

## 3.3 Other approaches

### 3.3.1 Artificial potential fields

Some approaches to coverage path planning use Artificial Potential Fields (APF). [Pirzadeh and Snyder, 1990] proposed an indirect control strategy to deal with the coverage and search

problem, where the complete coverage is accomplished using an APF. The underlying strategy of the algorithm is to discretize the workspace and the robot motion. The robot movement is designated with four orthogonal directions—up, down, left, and right—without considering any diagonal neighbors.

Most APF-based approaches suffer from local minima problems, and hence are not complete. The wave-front planner discussed in [Barraquand et al., 1992] addresses this problem, but requires resources exponential in the dimension of the space. Another solution to the local minima problem are a special kind of potential functions called *navigation functions* [Koditschek and Rimon, 1990; Rimon and Koditschek, 1992]. However, none of those proposals address the specific problem of coverage path planning.

### 3.3.2 Template-based models

Another common method for coverage path planning is template-based approaches. Using a template based method, the complete trajectory is planned as a sequence of pre-defined trajectories or templates.

[Hofner and Schmidt, 1995] proposed a template based approach using motion templates and motion mosaic. This model requires a predefined map and memorizes the templates. Therefore, it is difficult to handle environmental changes. In addition, it has difficulty identifying what area is uncovered. Thus, there exist uncovered areas near obstacles even after more templates such as side-shift (“SS”) template are supplemented.

[De Carvalho et al., 1997] proposed another template based coverage path planning model based on the prior knowledge of the 2-D map of a cleaning environment. An important advantage of this approach is that it is able to deal with newly appearing obstacles.

### 3.3.3 Neural networks and fuzzy logic

There are some neural networks and fuzzy logic based approaches to coverage path planning. [Yasutomi et al., 1988] presented a learning based coverage path planning approach, which can effectively avoid obstacles and walls in an unknown environment. Due to the computational complexity for the learning, this model has difficulty dealing with unstructured environments. Thus it is suitable for well-structured indoor environments.

[Tse et al., 1998] developed a backpropagation neural network based model, which can generate a robot path through self-learning. During the cleaning process, the robot should record the previously generated path, the number of grid points traveled and the number of turns. If the robot encounters a new environment, the memory map has to be updated.

[Lang and Chee, 1998] proposed a behavior model based on fuzzy logic, which has the ability to guide a robot to clean an unstructured room environment from any starting location. Due

to the difficulty in defining suitable fuzzy rules, the generated paths are not smooth enough at turning and traversing.

[Fu and Lang, 1999] proposed a fuzzy logic based method for coverage path planning, which is able to effectively correct and reduce the robot motion direction errors. When the robot sees obstacles with irregular shape, however, it ignores some regions in the vicinity of obstacles as it can follow straight lines only to skirt the obstacles.

[Luo et al., 2002] propose a solution based on a biologically inspired neural network capable of planning a real-time path to reasonably cover every area in the vicinity of obstacles. The robot path is autonomously generated through the dynamic neural activity landscape of the neural network and the previous robot location. The solution is proposed in the context of cleaning mobile robots. In a similar approach, [Yang and Luo, 2004] present a neural network approach to solve the coverage path planning problem. Their application area is cleaning robots in nonstationary environments.

Another approach intended for mobile robot coverage path planning is presented in [Qiu et al., 2006], where a biologically inspired neural network is used to model the environment and calculate the environment information, while a rolling path planning technique and an heuristic searching algorithm are utilized for the path planning. This approach handles sudden removal of obstacles and moving obstacles as well.

### 3.3.4 Miscellaneous approaches

Various other approaches have also proposed for coverage path planning. [Lumelsky et al., 1990] proposed a two algorithms for sensor-based coverage path planning capable of dealing with obstacles of any shape. Those algorithms are conceptually similar to the Bug 1 and Bug 2 algorithms for “start-to-goal” path planning presented in [Latombe, 1991].

The first algorithm, called the sightseer algorithm, applies the following strategy. While standing at the start position, the robot scans for and registers all the visible obstacle boundaries. If nothing can be seen, the task is accomplished. Otherwise, the robot navigates toward the nearest obstacle and then circumnavigates it completely while updating the map. Then, the robot marks the obstacle as visited, finds from the map the nearest unvisited obstacle, and navigates to it. The procedure repeats until no unvisited obstacles remain. If a new obstacle is encountered along the way toward the next targeted obstacle, the robot first circumnavigates the newly found obstacle and then chooses a new unvisited obstacle, which becomes the new target. On the other hand, if the robot encounters a visited obstacle on its way, it simply passes it around using a local direction that leads to a shorter path and resumes its linear-navigation motion; since the geometry of the obstacle is already known, choosing the shorter path presents no difficulties. Important constraints of this algorithm rely on obstacle visibility, as it requires that at least one obstacle is visible from the robot starting position, and all the obstacles are mutually visible from each other, that is, for any pair of obstacles  $X$  and  $Y$ , there is a sequence of obstacles visible from one another that leads from  $X$  to  $Y$ .

The other algorithm, called the seed spreader algorithm, does not impose those constraints. Its strategy is as follows. Assume that the terrain is rectangular, with dimensions  $A$  by  $B$ . If the terrain has many obstacles in it, then moving from obstacle to obstacle and circumnavigating every obstacle, as in the Sightseer algorithm, may produce very long paths. On the other hand, if the obstacles are “nicely distributed” and are of “nice geometry,” then encircling a group of obstacles –for example, by a rectangular path– may make them completely known without actually visiting and circumnavigating each of them. Thus, the terrain is divided into a number of strips of equal width by a set of lines parallel, say, to the side  $A$  of the terrain. It is hoped that most of the obstacles in a strip will be acquired without actually visiting them in the course of the movement along the strip boundaries. If, however, it becomes apparent that an obstacle cannot be acquired from the strip boundaries, the obstacle is explicitly visited via a divergence route. For a more detailed description of the algorithm, see [Lumelsky et al., 1990].

Both the sightseer algorithm and the seed spreader algorithm can be applied online. However, the sightseer algorithm imposes severe constraints on the obstacles in the terrain to be covered, and the seed spreader algorithm assumes that the terrain to be covered is known beforehand.

[Russell, 1997] proposed a coverage path planning approach by marking the robot path with a heat trail; then, the robot can avoid repeating previously marked areas. However, this approach requires the robot to effectively mark the path with a heater effector, and to measure the terrain temperature in order to determine the marked areas.

[Park and Lee, 1997] proposed a coverage path planning model for cleaning robots in unknown environments that consists of three components: a sweeping algorithm, a point-to-point moving algorithm and a corner work algorithm. However, the algorithm may overlap some areas and miss some corner areas.

[Jimenez et al., 2007] propose a genetic algorithm to achieve optimal coverage. In this proposal, workspace and obstacles are assumed to be polygonal and known beforehand. Then, the free space is divided in subregions using the trapezoidal cellular decomposition method [Choset et al., 2005]. From the decomposition, a global undirected graph representing the connections between subregions is obtained. Finally, the genetic algorithm is used to plan an optimal path that covers all the subregions. This proposal is tested and verified in simulation.

[Bosse et al., 2007] present two complementary coverage algorithms for an under-actuated car-like vehicle taking into account its kinematic constraints. In this approach, these two algorithms act as global path planners that generate paths without accounting for any obstacles. Then, those paths are delegated to an underlying Rapidly-exploring Random Trees (RRT)-based [LaValle and Kuffner, 1999] local path planning and Virtual Force Field [Borenstein and Koren, 1989] obstacle avoidance. The local path planner uses the waypoints passed by the global path planner to generate a more feasible and precise path.

[Oksanen and Visala, 2009] presented two novel algorithms for the coverage path planning problem in the case of agricultural fields and agricultural machines. Both of them are offline algorithms and assume previous knowledge of the terrain. In the first algorithm (called the split-and-merge approach in the article) the view is from on top of the field, and the goal is to split a single field plot into subfields that are simple to drive or operate. This algorithm utilizes

trapezoidal decomposition [Choset et al., 2005], and a search is developed of the best driving direction and selection of subfields. The second algorithm (called the predictive recursive online approach in the article) is also an incremental algorithm, but the path is planned on the basis of the machine’s current state and the search is on the next swath instead of the next subfield. This article also presents other practical aspects that are taken into account, such as underdrainage and laying headlands. There are advantages and disadvantages with both algorithms, neither of them solving the problem of coverage path planning problem optimally. Nevertheless, the developed algorithms are remarkable steps toward finding a way to solve the coverage path planning problem with nonomnidirectional vehicles and taking into consideration agricultural aspects.

### 3.4 Multi-robot coverage

There are several studies on coverage path planning of multirobot systems. [Kurabayashi, Ota, Arai and Yoshida, 1996] proposed an offline path planning algorithm for multiple cleaning robots, which is suitable for covering unstructured environments. The cleaning performance depends on the step length of the robot movement and the shape of the cleaned area. Thus, the robot is unable to plan in real time. They also proposed a floor cleaning path planning algorithm for cooperative sweeping with movable obstacles [Kurabayashi, Ota, Arai, Ichikawa, Koga, Asama and Endo, 1996], which is able to calculate an appropriate path distribution so that it can generate a complete coverage path. A key point of these approaches is that robots are made to move obstacles, not just avoid them, as we humans do with furniture and other objects when we clean a room. However, both approaches require the number, size and location of all the movable obstacles [Kurabayashi, Ota, Arai and Yoshida, 1996], [Kurabayashi, Ota, Arai, Ichikawa, Koga, Asama and Endo, 1996].

[Wagner et al., 1999] presented an approximate cellular decomposition approach for multirobot sweeping, each cell corresponding to a tile on the room’s floor. Each robot communicates with the others by leaving chemical odor traces that evaporate with time. Robots are assumed to be able to evaluate the strength of smell at every point they reach. Complete coverage is not guaranteed in this approach.

[Butler et al., 2000] proposed the  $DC_R$  (distributed coverage of rectilinear environments) algorithm.  $DC_R$  operates independently on each robot in a team. It applies to rectangular robots that use only contact sensing to detect obstacles and operate in a shared, connected rectilinear environment. The basic concept of  $DC_R$  is that cooperation and coverage are algorithmically decoupled. This means that a coverage algorithm for a single robot can be used in a cooperative setting, and the proof of completeness is much easier to obtain.  $DC_R$  is based on the complete single-robot coverage algorithm  $CC_{RM}$  [Butler et al., 1999] which incrementally constructs a cellular decomposition of the environment ( $C$ ). To produce cooperative coverage,  $CC_{RM}$  is enhanced with two additional components. Of these, the *overseer* is the more important. Its job is to take incoming data from other robots and integrate it into  $C$ , which it must do in such a manner that  $C$  remains admissible to  $CC_{RM}$ . The overseer adds a completed cell to  $C$

as follows. The cell is first shrunk to avoid overlap with existing cells, then added to  $C$ . Incomplete cells in  $C$  are reduced to avoid overlap with the new cell, and all connections between cells (which are necessary for path planning, among other things) are updated to reflect this addition. It can be shown that the overseer of  $DC_R$  indeed performs this operation in such a way that coverage can continue under the direction of  $CC_{RM}$  without  $CC_{RM}$  even knowing that cooperation occurred.

[Rekleitis et al., 2000] employed a trapezoidal decomposition [Choset et al., 2005] of the free space to deal with the cooperative sweeping problem. The robots together explore the environmental information and carry out the cooperative task, where each robot is regarded as a beacon for the others.

[Rekleitis et al., 2008] presented algorithmic solutions for the complete coverage path planning problem using a team of mobile robots. The algorithms use the same planar cell-based decomposition as the online Boustrophedon [Choset et al., 2000] single robot coverage algorithm, but provide extensions to handle how robots cover a single cell, and how robots are allocated among cells.

[Wang and Syrmos, 2009] describe a coverage-based path planning algorithm for multiple robotic agents with the application on the automated inspection of an unknown 2D environment. The proposed path planning algorithm determines a motion path that a robotic agent will follow to sweep and survey all areas of the unknown environment, which is enclosed by the known boundary. The 2D unknown environment is decomposed into a union of simplices using the principle of Delaunay triangulation. A hierarchical mission planner is designed to allocate mission tasks among multiple agents.

Recently, [Xu and Stentz, 2011] addressed the problem of environmental coverage with incomplete prior map information using  $k$  robots. A prior map is assumed to be available, but it may be inaccurate due to factors such as occlusion, age, dynamic objects, and resolution limitations. To utilize related algorithms in graph theory, the environment is represented as a graph and the coverage problem is modeled as a  $k$ -Rural Postman Problem. Using this representation, a graph coverage approach is presented for plan generation that can handle graph changes online. This approach proposes two improvements to an existing heuristic algorithm for the coverage problem. The proposed improvements seek to equalize the length of the  $k$  paths by minimizing the length of the maximum tour. The approach is evaluated on a set of comparison tests in simulation.

### 3.5 Coverage path planning for underwater environments

Coverage path planning is a key factor in many underwater applications, such as mine countermeasures, seabed mapping or underwater image mosaicking. As described in section 3.2.1, [Hert et al., 1996] proposed an online terrain-covering algorithm that uses a semiapproximate cellular decomposition. Though it can be applied to other environments, this algorithm is intended for performing underwater image mosaicking. Recently, [Lee et al., 2009] extended the proposal of

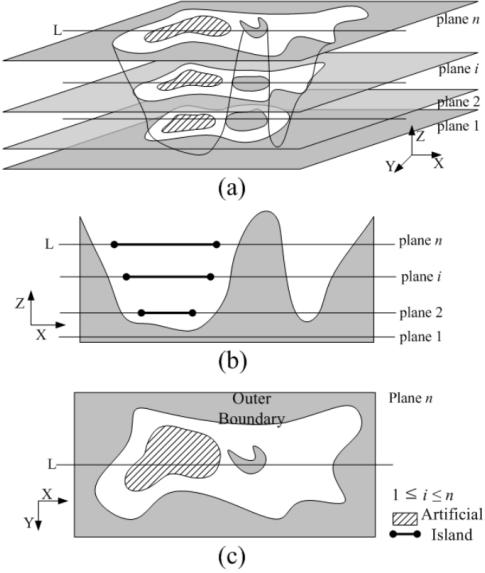


Figure 3.17: Illustration of the artificial island technique in three-dimensional space: (a) three dimension view of the entire terrain, (b) a vertical sectional view of (a), (c) a cross sectional view of (a).

[Hert et al., 1996]. They consider that, “since most of the resources in the deep sea or mines are located near the surface boundary it is sufficient to sense whether the boundary is changing or not and conduct missions near it”. Hereby, they propose the *artificial island technique*, which consists in covering only the areas near the terrain and obstacle boundaries, and letting the further away areas uncovered. They mark further away areas as *artificial islands*, as shown in figure 3.17, and then apply a planar algorithm at different depths taking those artificial islands into account to cover the three-dimensional space.

[Stack and Smith, 2003] present a method for underwater mine detection. In this class of missions, complete coverage may not be always feasible in terms of time and energy costs. Therefore, they investigate a path planning scheme for incomplete coverage. This method uses information about the mine laying patterns and plans a partial coverage resulting in a probability of missing a mine that is less than the percent of unsearched area.

[Fang and Anstee, 2010] consider offline algorithms for pre-mission path planning to achieve coverage of the seabed of complex but well-characterised planar areas using an AUV fitted with side-looking sonar.

[Williams, 2010] addresses the task of designing the optimal survey route that an AUV should take in MCM operations. It is assumed that the AUV is equipped with a side-looking sonar that is capable of generating high-resolution imagery of the underwater environment. The objective of the path-planning task is framed in terms of maximizing the success of detecting underwater mines in such imagery. In this approach, an offline algorithm is proposed which works in conjunction with Synthetic Aperture Sonar (SAS) data to predict detection performance and

efficiently design AUV routes that outperform standard ladder surveys.

### 3.5.1 Coverage path planning in current fields

In underwater environments, presence of currents is a factor that can't be neglected. As [Soulignac, 2011] note, "in such situations, existing approaches are subject to incorrectness and incompleteness issues. That is, they may return physically infeasible paths or no path at all, even if a feasible path exists."

Some studies in motion planning for underwater environments faced the presence of currents in the *start-goal* problem. [Alvarez et al., 2004] proposed a genetic algorithm for path planning of an autonomous underwater vehicle in an ocean environment characterized by strong currents. They claimed the algorithm "suitable for situations in which the vehicle has to operate energy-exhaustive missions". [Kruger et al., 2007] addressed the problems of automatically planning Autonomous Underwater Vehicle (AUV) paths which best exploit complex current data, from computational estuarine model forecasts, while also avoiding obstacles. In particular they examined the possibilities for a novel type of AUV mission deployment in fast flowing tidal river regions which experience bi-directional current flow. In these environments, by choosing an appropriate path in space and time, an AUV may both bypass adverse currents which are too fast to be overcome by the vehicle's motors and also exploit favorable currents to achieve greater speeds than the motors could otherwise provide, while substantially saving energy. These paths take maximum advantage of the river currents in order to minimize energy expenditure, journey time and other cost parameters.

Recently, [Soulignac, 2011] proposed a new approach to path planning in strong current fields called the *sliding wavefront expansion*. This algorithm, which combine an appropriate cost function and continuous optimization techniques, guarantees the existence of a path with an arbitrary precision. The validity and the global optimality of the path are theoretically proven.

However, few or no attention has been given to coverage path planning in current fields.

## 3.6 Discussion

In this survey, we have seen that the coverage path planning problem has been addressed using many different approaches. One can solve the problem by randomizing. Despite the fact that a random search does not guarantee complete coverage, there are advantages to this approach. Robots executing random searches may not require costly localization sensors, nor do they consume valuable computational resources for calculating their position. Though [Choset, 2001] concludes that if a robot with a random algorithm can be constructed at 1/5 of the price of one with localization and advanced path planning, it is cost efficient, it is difficult to think that a randomized "algorithm" could be feasible for covering vast areas.

Many studies use the approximate cellular decomposition approach to achieve complete coverage, though most of these methods are resolution complete, that is, their ability to achieve completeness rely on the grid's grain size. An exception to this is the approach proposed in [Gonzalez et al., 2005], which proposes an extension to cover also partially occupied cells.

[Hert et al., 1996] proposes a semiapproximate cellular decomposition algorithm for terrain covering. The algorithm is proven to be correct and its complexity is measured. The length of the robot's path in a planar environment is in the worst case linear in the lengths of the outer boundary and the island boundaries, and in the length of the grid line segments in the interior of the area to be covered. Another advantage of this approach is that it could provide complete coverage without assuming any prior information about the robot's free space, i.e., the algorithm can be applied on-line. However, the algorithm is not complete.

Exact cellular decomposition approaches appear to be the most reliable. Morse decompositions can be used to perform complete coverage path planning online [Choset et al., 2000]. However, critical points detection for cell subdivision is a non-trivial and complex task [Acar and Choset, 2000, 2002; Garcia and de Santos, 2004]. [Acar et al., 2006] extended the sensor-based Morse decomposition by combining it with the GVD for covering cluttered spaces where the range of an extended range sensor is infinite in practice, hence obtaining more efficient paths.

Other approaches which don't use cell decompositions have been discussed. One of them is using APF, but the potential function needs to be computed offline. Moreover, using APF raises the problem of falling into local minima, where the vehicle would get blocked forever without accomplishing the coverage task. Some template-based models have been proposed where planning is performed as sequences of pre-computed trajectories. However, those methods cannot be applied online. Several neural networks and fuzzy logic approaches have been also proposed, but all of them rise the handicap of tuning parameters and rules to their appropriate values. While some of those methods achieve complete coverage online, they strongly depend on the parametrization of the neural network and fuzzy logic models. It is also worth mentioning the hierarchical 2-level coverage path planning proposed in [Bosse et al., 2007], which accounts for kinematic constraints, and the algorithms described in [Oksanen and Visala, 2009] which also take into account nonholonomic constraints and other specific requirements of agricultural fields.

Also, we have discussed several approaches to coverage path planning utilizing multiple robots. Among them, the most remarkable are the  $DC_R$  algorithm presented in [Butler et al., 2000], the extension of Choset's Morse decomposition to multiple robots presented in [Rekleitis et al., 2008] and the approach proposed in [Xu and Stentz, 2011], as all of them are complete online approaches. It is worth noticing the ability of the proposal in [Xu and Stentz, 2011] to handle environmental changes in the environment.

In the particular context of underwater environments, we have discussed the terrain covering algorithm presented in [Hert et al., 1996], which is intended for underwater environments, and how [Lee et al., 2009] extend the former method by applying the artificial island technique. In turn, several methods address the design of optimal survey routes for specific applications, like MCM. Finally, we saw that, though several studies account for current fields in start-to-

goal path planning in underwater environments, no studies were found addressing this issue in coverage path planning.

Tables 3.1, 3.2 and 3.3 show a summary of the reviewed general coverage path planning methods. Table 3.4 summarizes the reviewed coverage path planning methods for underwater environments. In these tables, every method is classified as whether or not it achieves complete coverage and as either online or offline; also, we classify every method as whether or not it needs prior knowledge of the environment, and as whether or not it handles non-polygonal environments; we also give some remarks on some of the methods and its intended application.

Several methods discussed above guarantee complete coverage online, that is, they can be used to cover all points on the free space of unknown environments. Furthermore, some of those methods account for kinematic constraints on the vehicle and also efforts have gone in optimization of the coverage path. However, a universal algorithm that guarantees an optimal path has not yet been developed.

In the case of underwater environments, solutions for surveying an underwater area have been presented and some methods addressing specific applications like MCM have been developed. Nonetheless, no provably complete, optimal algorithm addressing the specific requirements of underwater environments has been presented. Moreover, no coverage path planning studies addressing the important problem of current fields in underwater environments have been found.

As considered in the previous chapter (section 2.1), some applications like image mosaicking might impose certain constraints on the coverage path, such that pass through certain points more than once to perform feature matching. It is worth noticing that, based on our review, it can be said that no studies treated this issue; hence, a door opens for further research.

In conclusion, it can be said that coverage path planning is still being researched and an optimal and universal algorithm has not been developed yet, so there remains a need for further research.

Table 3.1: Summary of the analyzed cellular decomposition coverage path planning methods. Methods are grouped by their decomposition approach, i.e., approximate, semiapproximate, and exact cellular. “n/a” stands for “not available.” SCD stands for “semiapproximate cellular decomposition.”

	<b>Article</b>	<b>Algorithm</b>	<b>Completeness</b>	<b>On/Off line</b>	<b>Prior knowledge required</b>	<b>Handles non-polygonal obstacles</b>	<b>Remarks</b>	<b>Intended application</b>
Approximate cell decomp.	[Zelinsky et al., 1993]	n/a	Resolution-complete	Online	Yes	Yes	Does not account for kinematic constraints	Mobile robotics
	[Gabriely and Rimon, 2002]	n/a	Resolution-complete	Online	No	Yes		Mobile robotics
	[Gonzalez et al., 2005]	n/a	Complete	Online	No	Yes	Fills also partially occupied cells.	Mobile robotics
	[Choi et al., 2009]	n/a	Complete	Online	No	Yes		Mobile robotics
SCD	[Hert et al., 1996]	n/a	Not complete	Online	No	Yes		Underwater environments, underwater mosaicking
Exact cell decomposition	[Latombe, 1991]	Trapezoidal decomposition	Complete	Offline	Yes	No	Probably the most popular exact cell decomposition.	2D spaces
	[Choset and Pignon, 1997]	Boustrophedon decomposition	Complete	Offline	Yes	No	Relatively easy to implement	2D spaces
	[Choset et al., 2000]	Morse decomposition	Complete	Offline	Yes	Yes		Generic
	[Acar and Choset, 2000, 2002]	Morse decomposition	Complete	Online	No	Yes	First online (sensor-based) exact cell decomposition proposed.	Generic
	[Atkar et al., 2001]	n/a	Complete	Online	No	Yes	Alternative critical point detection method to [Acar and Choset, 2000]	Generic
	[Garcia and de Santos, 2004]	n/a	Complete	Online	No	Yes	Improves sensor-based Morse decomposition by detecting critical points also on nonconvex obstacles	Mobile robotics
	[Acar et al., 2006]	Morse cell decompositions + GVD	Complete	Online	No	Yes	Pioneer approach to consider limited range sensors, that is, longer range than the robot size but not infinite. Interesting 2-technique combination	Generic
	[Cao et al., 1988]	n/a	Complete	Online	Yes	Yes	Requires the boundary of obstacles and walls in advance	Lawn mower robots
	[Butler et al., 1999]	$CC_{RM}$	Complete	Online	No	Yes		Mobile robotics
	[Huang, 2001]	n/a	Not complete	Offline	No	Yes		Mobile robotics
	[Wong and MacDonald, 2003]	n/a	Not complete	Online	No	Yes		Mobile robotics

Table 3.2: Summary of the analyzed coverage path planning methods which don't use cellular decomposition. Methods are grouped by their approach, i.e., APF, template-based models, neural networks and fuzzy logic and other miscellaneous approaches. “n/a” stands for “not available.” TBM stands for “template-based models.”

	Article	Algorithm/ method	Completeness	On/Off line	Prior knowl- edge re- quired	Handles non- polygonal obstacles	Remarks	Intended application
APF	[Pirzadeh and Snyder, 1990]	n/a	Not complete	Offline	Yes	Yes	Applies to a general class of configuration spaces, i.e., high-dimensional spaces. Suffers from local minima problems.	n/a
TBM	[Hofner and Schmidt, 1995]	n/a	Not complete	Offline	Yes	Yes	Not complete. Unable to handle environmental changes.	Generic
	[De Carvalho et al., 1997]	n/a	Complete	Offline	Yes	Yes		Generic
Neural networks & fuzzy logic	[Yasutomi et al., 1988]	Neural network	Not complete	Online	No	Yes		Mobile robotics
	[Tse et al., 1998]	Neural network	Not complete	Online	No	Yes		Mobile robotics
	[Lang and Chee, 1998]	Fuzzy logic	Not complete	Online	No	Yes		Mobile robotics
	[Fu and Lang, 1999]	Fuzzy logic	Not complete	Online	No	Yes		Mobile robotics
	[Luo et al., 2002]	Neural network	-	-	-	-		
	[Yang and Luo, 2004]	Neural network	Complete	Online	No	Yes	Model strongly depends on parameter settings	Mobile robotics
	[Qiu et al., 2006]	Neural network	Complete	Online	No	Yes	Model strongly depends on parameter settings	Mobile robotics
Miscellaneous approaches	[Lumelsky et al., 1990]	Sightseer algo- rithm	Complete	Offline	No	Yes	Easy to implement. Severe constraints on obstacle visibility.	Generic
	[Lumelsky et al., 1990]	Seed spreader algorithm	Complete	Online	Yes	Yes	Prior knowledge of the environment required.	Generic
	[Russell, 1997]	Heat trail	Complete	Online	No	Yes	Requires heater effectors to mark the path and a temperature sensor to determine previously marked areas.	Mobile robotics
	[Park and Lee, 1997]	3-component model	Not complete	Online	No	Yes		Cleaning robots
	[Jimenez et al., 2007]	Genetic algo- rithm	Complete	Offline	Yes	No		Generic
	[Bosse et al., 2007]	2-level path planning	Not complete	Online	No	Yes	Interesting 2-level approach	Lawn mowing
	[Oksanen and Visala, 2009]	Split-and- merge algo- rithm	Complete	Offline	Yes	Yes	Take into account particularities of agricultural fields	Agricultural fields
	[Oksanen and Visala, 2009]	Predictive recursive algo- rithm	Complete	Online	Yes	Yes	"	"

Table 3.3: Summary of the analyzed multi-robot coverage path planning methods. “n/a” stands for “not available.”

Article	Algorithm	Completeness	On/Off line	Prior knowledge required	Handles non-polygonal obstacles	Remarks	Intended application
Multi-robot approaches							
[Kurabayashi, Ota, Arai and Yoshida, 1996]	n/a	Complete	Offline	Yes	Yes	Robots move obstacles to accomplish the cleaning tasks. Strong assumptions about obstacles’ shape are made.	Cleaning robots
[Wagner et al., 1999]	n/a	Not complete	Online	No	n/a	Added difficulty of using chemical odor traces	Mobile robotics
[Butler et al., 2000]	DCR	Complete	Online	No	Yes	Cooperation and coverage are decoupled: each individual robot does not even need to know about cooperation.	Generic
[Rekleitis et al., 2000]	n/a	Complete	Offline	Yes	No		Mobile robotics
[Rekleitis et al., 2008]	n/a	Complete	Online	No	Yes	Extension to multiple robots of Choset’s Morse decomposition	Generic
[Wang and Syrtimos, 2009]	n/a	Complete	Online	Yes	Yes		Generic
[Xu and Stentz, 2011]	n/a	Complete	Online	Yes	Yes	Handles environmental changes. Suited for network environments such as roads and intersections	Generic

Table 3.4: Summary of the analyzed coverage path planning methods for underwater environments. “n/a” stands for “not available.”

Article	Algorithm	Completeness	On/Off line	Prior knowledge required	Handles non-polygonal obstacles	Remarks	Intended application
Underwater approaches							
[Hert et al., 1996]	n/a	Not complete	Online	No	Yes	Applies to 3D underwater environments	Underwater mosaicking
[Lee et al., 2009]	n/a	Not complete	Online	No	Yes	Avoids covering areas of potentially few interest which are far away from the terrain boundaries	Underwater mosaicking
[Stack and Smith, 2003]	n/a	Not complete	Online	Yes	Yes	Intended for partially covering vast areas using information about the mine laying patterns to reduce the probability of missing a mine while avoiding the expenses of completely covering the area.	MCM
[Fang and Anslee, 2010]	n/a	Not complete	Offline	Yes	No		Underwater robotics
[Williams, 2010]	n/a	Not complete	Offline	Yes	Yes		MCM

# Chapter 4

## Underwater obstacle detection

In this chapter, we discuss methods found in the literature and available technologies for performing obstacle detection in underwater environments.

Most online coverage path planning methods described in the previous chapter rely on range detection sensors for perceiving the environment information and direct the robot accordingly. Hence, to apply a sensor-based coverage path planning method, we must first be able to accurately detect the boundaries of the terrain and the obstacles present in the workspace. Usually, these sensors are supposed to be perfect in theory, that is, without any noise and with arbitrary precision. While in a context like mobile robotic applications range sensors such as those based on Laser Detection and Range (LADAR) technology can perceive the range information of the robot environment very precisely, such technology is not applicable in an underwater context, where the lack of visibility and the particularities of the environment restrict the accuracy of the sensorial information. In fact, the lack of visibility due to attenuation of the light in underwater environments makes acoustic perception the most reliable and widely used technology for sensing underwater environments. Acoustic sensors can operate in larger visibility ranges and provide 3D information even under water turbidity conditions though at expense of a coarse resolution and a more difficult to interpret data.

Thus, we next review the available acoustic sensor technology for detecting obstacles in underwater environments, and then we discuss several underwater obstacle detection studies found in the literature.

The bulk of the work presented in this chapter has been carried out at NATO Undersea Research Center (NURC), where acoustic real-world datasets from state-of-the-art sonars could be collected and analyzed.

## 4.1 Acoustic sensing review

In underwater environments, acoustic sensors have been proved useful to a wide range of applications including localization, obstacle avoidance and seabed mapping. For our purposes a brief review on acoustic sensing, specially laying emphasis on forward-looking sonars, will be useful before examining proposed underwater obstacle detection approaches.

“Active sonars transmit an acoustic signal and objects within the path of the acoustic beam reflect some of the energy back towards the sonar (backscattering). The excellent propagation of sound in water makes possible for an acoustic wave to travel hundreds of meters without the signal losing significant energy, allowing to measure at long ranges even in turbid water conditions.” [Hurtós, 2009]

Concerning the backscattered signals two different kinds of information can be obtained. On one hand, two-way travel time of the reflected acoustic pulse can be recorded and converted to depth assuming a particular sound velocity in water thus obtaining bathymetric information. On the other hand, some sonars can also measure the intensity of the returning echo which can be arranged in a shape of “acoustic image” describing the composition of the scanned terrain.

It is worth to underline that despite the increase in visibility range with respect to vision sensors, sound transmission underwater is also attenuated, specially when using high frequencies. Therefore, a tradeoff exists between a longer range provided by a low frequency sonar and the higher resolution data produced by a high frequency sonar.

In the following subsections the principal types of sonar sensors for obstacle detection are briefly described with the aim of introducing its main characteristics, configuration, data structure and applications. For a more technical and extensive review on acoustic sensing the reader can refer to [Lurton, 2002].

### 4.1.1 Single-beam sounders

Single-beam sounders, used since the 1920’s, are the most basic and common underwater acoustic systems. They are usually mounted on a vessel’s hull facing downwards in order to measure the water depth at the vertical of the vessel. However, they can be used to detect approaching obstacles from the front by mounting it in an horizontal plane parallel to the vessel’s motion. For instance, an approaching wall is an obstacle feasible to be detected by a single-beam sounder.

Its functioning is based on a single transducer that emits a short sound pulse (around  $10^{-4}$  –  $10^{-3}$  s) vertically below the ship in a beam of a narrow aperture (typically  $5^\circ$  –  $15^\circ$ ). This transducer receives the echo signal and a corresponding range is computed (see figure 4.1). Although the bathymetric function is the most frequent one for single-beams sounders they can also be used to detect targets in the water column below the ship (and therefore be useful for fishing). The operating frequencies of single-beam sounders depend on the working ranges.

Usually it varies from 12kHz for deep waters (up to 11000 meters) to 400KHz or even more for shallower waters (from 1 to 50 meters).

There are also dual-frequency transducers available in the market. Regarding the resolution, single-beam sounders have a vertical resolution that depends on the emitted pulse duration but typically is between 0.075 and 0.75 meters. Single beam sounders are not a precise obstacle detection instrument as only with only one beam they can easily miss obstacles approaching from nonorthogonal directions to the beam direction. However, its easiness to be transported and deployed in small vehicles, together with its low-cost acquisition and processing has made single-beam sounders popular in a great number of applications .

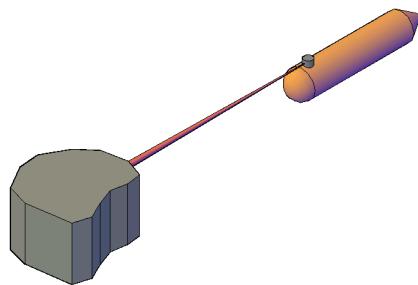


Figure 4.1: Obstacle perception with a single-beam sounder.

Notice that, with single-beam sounders, only range information from a front-approaching obstacle can be obtained, i.e., the distance to the obstacle.

#### 4.1.2 Mechanically scanned profiler

This sensor is composed of a mechanically actuated transducer which can be sequentially oriented at different angles and produces a series of range measurements. Usually, the size of the scan sector can be selected from a few degrees to a complete  $360^\circ$  scan around the transducer, which is particularly interesting for obstacle detection tasks. When mounted in a down-looking position, they can also be employed to collect bathymetric data. An important issue that has to be taken into account when operating with mechanically scanning transducers is that it needs some time to perform the rotation over a sector. Hence if the vehicle is moving at high speeds the acoustic returns can get distorted as a consequence of the vehicle's motion.

With a mechanically scanned profiler a point cloud corresponding to the incidence of the beams along the profile of an obstacle can be obtained. Nonetheless, due to aforementioned distortion issue, this point cloud might not be very accurate.

### 4.1.3 Multibeam sounders

Motivated by the limitations of single beam sounders, multibeam sounders appeared in the 1970's and since then they have greatly evolved and are nowadays a widespread system for seafloor mapping tasks. However, its use in obstacle detection tasks is not so popular.

As mechanically scanned profilers, multibeam sounders produce a series of range measurements along a scan sector, but rather than using a single rotating beam, they provide multiple readings at the same point in time. That is, multibeam sounders transmit a fan of individual acoustic beams (usually 100-200 beams, around  $1^\circ - 3^\circ$  each), obtaining an array of range information around the transducer at a time (see figure 4.2). The angular aperture of the beam's fan is typically from  $90^\circ$  to  $180^\circ$ . Its operation is usually based in two transducer arrays, one for transmission and one for reception that are fixed in the vessel in a particular configuration. Each time-angle couple is used to obtain one range measurement. In this way, the job of a single beam sounder is performed at multiple locations at once.

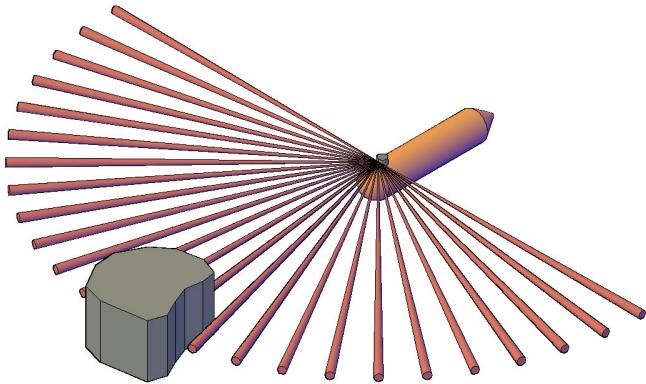


Figure 4.2: Obstacle perception with a multibeam sounder.

With obstacle detection in mind, a multibeam sounder provides a point cloud corresponding to the incidence of each beam along the profile of the obstacle. As an advantage over mechanically scanned profilers, multibeam sounders are much less affected by distortion due to motion.

### 4.1.4 Forward-looking sonars

Forward looking sonars (FLS) are similar to single beam and multibeam technologies but they usually provide imagery data rather than range information (see figure 4.3). Forward-looking sonars have been used for many years for remotely controlled navigation, obstacle avoidance, midwater mine detection and localization around known structures. The major advantage of this type of sonar is its capability of detecting objects or seabed features, such as rocks proud of the ocean floor, at large distances so they can be observed in subsequent scans and tracked. Unfortunately, an important limitation to FLS range is depth; due to surface and bottom echo

interference, FLS can generally see ahead only about six times the depth of the water column. Two main types of forward-looking sonars are available: **mechanically scanned sonars** and **forward-looking multibeam sonars**.

Mechanically scanned sonars consist of a single transducer which is mechanically scanned along the horizontal axis, sweeping a so-called sector. The returns are then used to create an image. Most systems provide the user with the option of choosing the size of the sector to scan and with some degree of control on the resolution. However, as happens with mechanically scanned profilers, mechanically scanned sonars suffer from distortion due to motion.

Forward-looking multibeam sonars use a fixed array of transducers, scanned electronically, which allows much faster updates of sectors (e.g., the Seabat 6012 can update a sector up to 30 times a second). These sonars are more expensive than mechanical systems; nevertheless their popularity in the underwater community has been growing. Automatic methods for obstacle avoidance, motion estimation, and image recognition using forward-looking multibeam sonar images have already appeared. Fast update rates make distortion due to motion practically negligible in Forward-looking multibeam sonars.

Some high-frequency multibeam forward-looking sonars have recently emerged under the name of “acoustic cameras” regarding their capabilities to offer real-time acoustic video imagery (high refresh rates and high resolution at short ranges). One of the most popular commercial solutions are BlueViews’s wide range of products. Other popular commercial solutions are DIDSON (Dual-frequency Identification SONar) which using an acoustic lenses system offers high resolution (under centimeter level) 2D images. The cost of these kind of sensors is still very expensive (almost three times the cost of a multibeam forward-looking sonar), though decreasing, but its size and capabilities have evolved a lot recently and thus they are becoming an interesting opportunity for several applications, especially those involving AUVs.

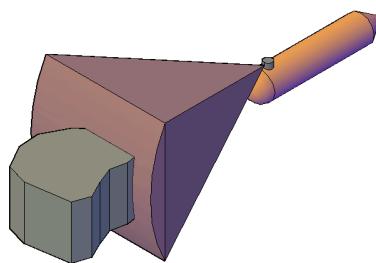


Figure 4.3: Obstacle perception with a multibeam FLS.

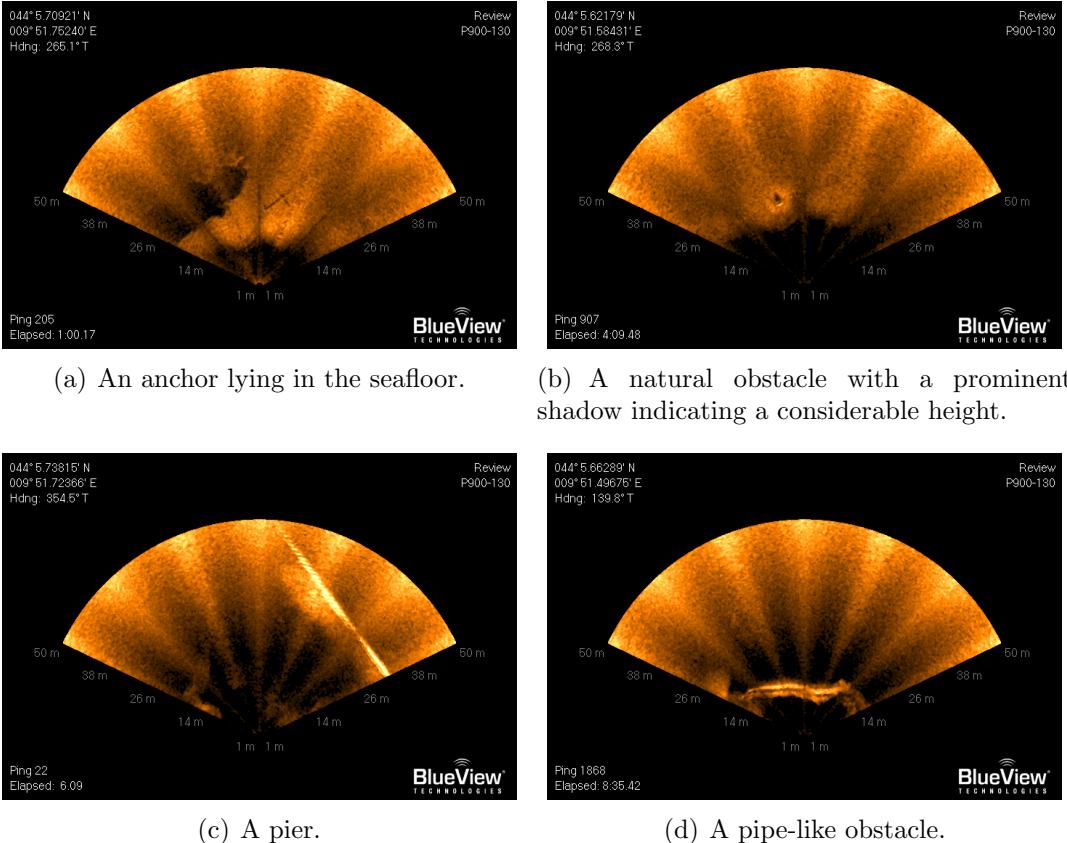


Figure 4.4: Several sonar images captured with a BlueView P900 FLS at NURC.

## 4.2 Sonar-based obstacle detection

Underwater target recognition techniques are very important for an AUV to complete its missions such as relocation of lost targets, removal of toxic wastes and mine counter measures and for obstacle detection tasks.

In the underwater target classification domain, [Lu and Sang, 1998] address the problem of how to extract the target's size and shape features from a sequence of 2D acoustic images acquired with a SeaBat 6012 forward looking sonar. They utilize image processing techniques and the near field acoustic scattering principles of underwater target to estimate the two-dimensional position and size/shape of a nearby target. Some experimental and analytical results about these features are provided to demonstrate the feasibility of these methods.

[Guo et al., 1998] use the continuous image sequences generated by an electronic scanning forward-looking sonar to achieve the aim of obstacle avoidance and visual navigation for an AUV. As they notice, “using sonar systems for sensing of unknown underwater environments is the best selection in practice.” However, the critical demand for real-time signal processing and the uncertainties of AUV’s dynamics make online detection of obstacles a challenging

task. Here, they use a track-before-detect strategy to extract information contained in image sequences to estimate the dynamics of the AUV, then they apply a dynamic programming algorithm to solve the problem of detection. This method aims to reduce the computational cost to meet the real-time demand on obstacle avoidance and navigation of an AUV system.

[Petillot et al., 2001] describe a framework for segmentation of sonar images, tracking of underwater objects and motion estimation. This framework is applied to the design of an obstacle avoidance and path planning system for underwater vehicles based on a multi-beam forward looking sonar sensor. The real-time data flow (acoustic images) at the input of the system is first segmented and relevant features are extracted. They also use of the real-time data stream to track the obstacles in following frames to obtain their dynamic characteristics. This allows for optimizing the preprocessing phases in segmenting only the relevant part of the images. Once the static (size and shape) as well as dynamic characteristics (velocity, acceleration, ...) of the obstacles have been computed, a representation of the vehicle's workspace is created based on these features. This representation uses constructive solid geometry (CSG) to create a convex set of obstacles defining the workspace. The tracking takes also into account obstacles which are no longer in the field of view of the sonar. A well-proven nonlinear search (sequential quadratic programming) is then employed, where obstacles are expressed as constraints in a search space. This approach is less affected by local minima than classical methods using potential fields. The proposed system is not only capable of obstacle avoidance but also of path planning in complex environments which include fast moving obstacles. Results obtained on real sonar data are shown and discussed. Nonetheless, the obstacle feature extraction from sonar data is based on image processing techniques.

[Trucco and Curletto, 2003] proposed a method for estimation of 3D information out of a sequence of 2D forward-looking sonar images. However, their approach requires the selection of a previous feature of interest which is tracked all along the sequence. Hence, it's not a feasible approach for obstacle detection. However, it indeed can be used for characterization of known obstacles.

## 4.3 Discussion

In this chapter we first reviewed available acoustic sensor technology solutions for underwater obstacle detection. Among them, forward-looking sonar is the most widespread technology for this task, as it potentially offers more information for characterization of obstacles than single-beam and multibeam sounders and mechanically scanned profiler sonar approaches. Inside this category, multibeam forward looking sonar solutions offer higher resolutions and higher refresh rates than mechanically scanned sonar solutions, though at a most expensive cost and at a more limited range.

Then, we discussed some relevant sonar-based obstacle detection approaches in underwater environments. From them, we can conclude that the state of the art for obstacle detection on acoustic images is to apply segmentation methods and then extract features out of the obtained blobs by applying computer vision image processing techniques.



# Chapter 5

## Experimental work

After defining the problem, exhaustively reviewing the coverage path planning methods proposed in the literature and studying obstacle detection methods for underwater environments, we proceeded to perform experimentation in order to extract results which contribute to evaluate the suitability of the reviewed methods for underwater environments.

As a first step towards the aforementioned evaluation, we performed two experiments in which we implemented the boustrophedon decomposition based on critical points of Morse functions [Choset et al., 2000].

In the first experiment (section 5.1), the boustrophedon decomposition is applied in simulation to a real-world bathymetric dataset recorded in the Formigues islands. The goal of the experiment is to achieve complete coverage of the whole region.

In the second experiment (section 5.2), we apply the boustrophedon decomposition in a real mission with an AUV as part of a task performed in the SAUC-E student underwater robotics competition.

### 5.1 Experimentation with the Formigues islands dataset

#### 5.1.1 Goal of the experiment

The goal of this experiment is to apply the boustrophedon decompositon method in order to achieve complete coverage of the workspace determined by the bathymetric data recorded in the Formigues islands dataset at a constant depth.

### 5.1.2 The Formigues islands dataset

The Formigues islands dataset used in this experiment was recorded by members of Centre d'Investigació en Robòtica Submarina (CIRS) from the University of Girona in July 2009 near the Formigues islands. The Formigues islands are a group of sixteen little islands located 1300 meters away from Canet's cape in the Brave Coast, at  $41^{\circ} 52' 35.3''$  N,  $3^{\circ} 11' 58.12''$  E. Figure 5.1 shows a picture of the area.



Figure 5.1: The Formigues islands.

The dataset consist in a bathymetry of an approximately 60-by-100 meters area. The bathymetric data was obtained by means of a down-looking multibeam sonar fixed to a boat's hull. Figure 5.2 shows the recorded bathymetric data. Depth ranges from around 5.5 meters to 12 meters.

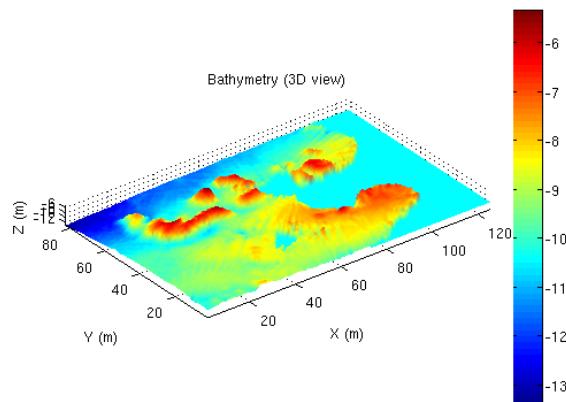


Figure 5.2: Bathymetric data recorded in the Formigues islands dataset.

### 5.1.3 Experimental setup

In order to apply the boustrophedon decomposition, which is intended for planar spaces, what we did in this experiment was to obtain an horizontal, planar section out of the bathymetry in the dataset. That is, we took all points laying at 7.5 meters depth in the bathymetry to obtain the target workspace. We chose this depth because it produces a challenging (in terms of number and shape of obstacles) workspace to apply the boustrophedon decomposition on. Figure 5.3 shows the obtained 2D workspace.



Figure 5.3: 2D workspace corresponding to the planar section at 7.5 meters depth of the Formigues islands bathymetry.

For simplicity, we assume a point-sized robot in this experiment, and thus the workspace and the configuration space are the same. The interlap distance between the boustrophedon swaths is a parameter that can be set in our implementation.

### 5.1.4 Results

Figure 5.4 shows the obtained boustrophedon decomposition of the Formigues islands workspace. The corresponding adjacency graph is shown in figure 5.5.

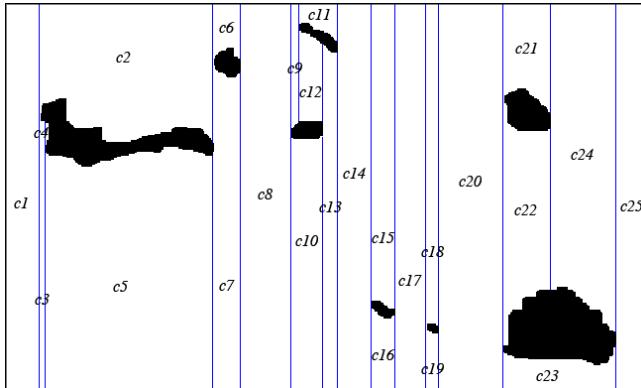


Figure 5.4: Boustrophedon decomposition of the Formigues islands workspace.

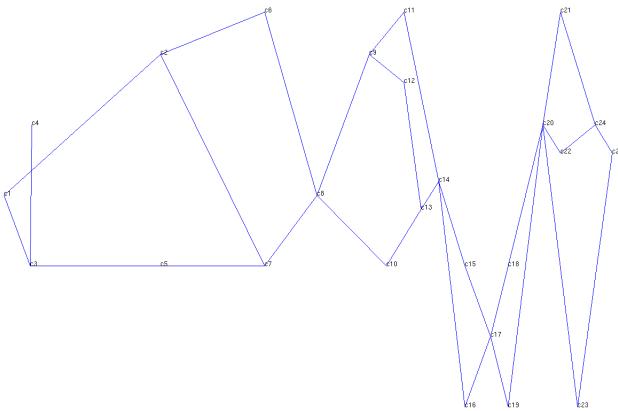


Figure 5.5: Adjacency graph associated to the boustrophedon decomposition of the Formigues islands workspace.

Complete coverage of the target workspace is guaranteed by determining an exhaustive walk through the graph. Once the walk is obtained, an explicit path inside each cell should be generated. However, these steps have not been applied to the present dataset at the time of writing this document.

## 5.2 Experimentation at the SAUC-E competition

### 5.2.1 Goal of the experiment

The goal of this experiment is to apply the boustrophedon decomposition coverage path planning method to a target region of the SAUC-E competition arena in order to look for an underwater target (an underwater pipeline structure). The coverage task must be finished either when the target is detected by the AUV or when the area is completely covered.

### 5.2.2 The SAUC-E competition

Held since 2006, the Student Autonomous Underwater Challenge - Europe (SAUC-E) competition challenges the next generation of engineers to design and build an AUV capable of performing realistic missions. The event is designed to encourage students to think about underwater technology and related applications while fostering innovation and technology. From 2010 SAUC-E has been held in La Spezia (Italy) at NATO Undersea Research Center (NURC). Since then, competitors have to deal with real life conditions (i.e. limited visibility and salty water) since it is being held outside at the NURC waterfront, which is a sheltered harbor. Our team from the University of Girona has participated in SAUC-E in the 2006 and 2010 editions, where the team achieved to be the champion, and also in the 2011 edition, where the team achieved the second position and also the Innovation Award.

The competition consist of a sequence of tasks that a robot from every team must perform in a completely autonomous manner. The competition arena in NURC's sheltered harbor is approximately 50-by-80 meters and its depth ranges from 4 to 5 meters. Figure 5.6 shows a diagram of the SAUC-E competition arena. Therein, the sequence of tasks to be performed was the following in the 2011 edition:

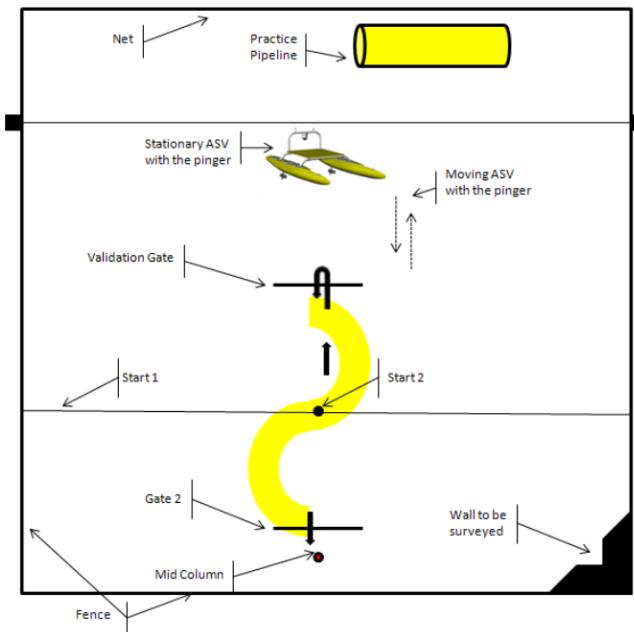


Figure 5.6: The SAUC-E competition arena.

- Move from a launch/release point, submerge, and pass through the validation gate shown in figure 5.6.
- Inspect an irregular underwater pipeline, shown in yellow in the center of figure 5.6. The pipe will be laying on the seafloor. This is the task in what the present experiment is focused.

- Look for a mid-water target (shown in red in the bottom-center of figure 5.6) and free the rope that tethers it to the seafloor.
- Perform a wall inspection. The wall is located in the bottom-right corner of figure 5.6.
- Track the NURC's ASV, which moves slowly in the competition arena. An acoustic pinger is attached to the ASV, which emits a signal at a specified frequency and rate. The ASV might be tracked either by finding the pinger location or by looking up with an onboard sensor.
- Emerge in a position bounded by the NURC's ASV, which will move fast to the surfacing zone. Again, the emerging position can be reached by finding the pinger or by looking up with a sensor. However the ASV might move too fast to be tracked by looking up, so this task is intentionally designed to be performed by means of pinger detection.

### The pipe inspection task

This experiment focus on the pipe inspection task of the competition, in where two main issues must be considered. First, the pipeline structure can be moved or changed in shape from trial to trial. Second, the attenuation of light and the turbidity of the water make the pipeline visible by the cameras\* of the robot only from a close distance (< 1.5 m). Therefore, a search must be performed by the AUV in order to find the pipe's location and start the inspection procedure.



Figure 5.7: Underwater pipeline to be inspected in SAUC-E. Visibility conditions only allow the pipe to be detected

#### 5.2.3 The Sparus AUV

The Sparus AUV (figure 5.8) is the vehicle used by our team in the SAUC-E competition.

It was built from scratch by our 2010 SAUC-E team, and was designed with the main goal of having a small and simple torpedo-shaped vehicle with hovering capabilities. Though it was intended to face the competition, Sparus is intensively used as a research platform at CIRS.

---

\*Though the pipeline could be potentially detected with acoustic perception, the sonar mounted on our robot doesn't allow enough resolution for such a task. Thus, we relied on computer vision for the pipe detection.

The vehicle is equipped with a sensor suite composed by three color video cameras (down-looking, forward-looking and up-looking), an MTi Motion Reference Unit (MRU) from XSens Technologies, a Micron mechanically scanned imaging sonar from Tritech, a single-beam echosounder, a pressure sensor and a Doppler Velocity Log (DVL) from LinkQuest which also includes a compass/tilt sensor. Sparus operates three thrusters: one vertical for the heave Degree of Freedom (DOF) and two horizontal for the surge and yaw DOFs. Also, two servo-actuated fins allow the robot to control the pitch DOF.

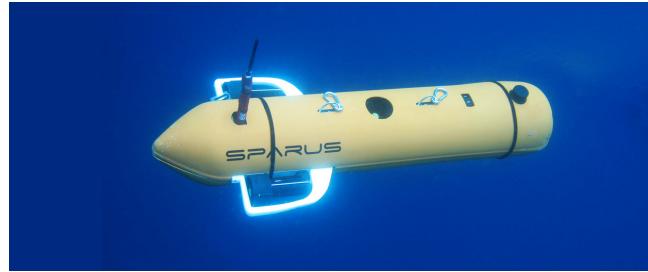


Figure 5.8: The Sparus AUV.

#### 5.2.4 Experimental setup

We use previous knowledge of the competition arena to set up the experimental procedure. First, we determine a rectangular target region where we conjecture the pipeline lies in. This will be the workspace where the boustrophedon method will be applied. In this experiment, the workspace will be a 20-by-20 squared region placed in the middle of the competition arena. The path will be submerged at a constant depth of 3 meters, as the pipeline is expected to be laying at around 4 meters depth. As explained in the previous experiment, the interlap distance between the boustrophedon swaths is a parameter that can be set. Here, we set an interlap distance of 2 meters, accounting for the camera field of view. Given that the robot knows its initial position (before submerging) by means of a GPS fix, and that the environment is known in advance, we can georeference the workspace. Figure 5.9 shows the georeferenced workspace in the competition arena plotted in Google Earth.



Figure 5.9: Workspace for the SAUC-E pipeline search task plotted in Google Earth (in white in the center of the image).

As no potential obstacles lie in the target region, the cell decomposition here is trivial: one cell corresponding to the whole configuration space. Nonetheless, an explicit boustrophedon path (recall that a boustrophedon path mimics a typical farming style path) must be generated inside that cell.

Regarding the configuration space, we can also assume a point-sized robot in this experiment, as no hard boundaries nor obstacles jeopardize our vehicle inside and in the immediate vicinity of the target region. Thus, the workspace and the configuration space are the same in this experiment as well.

To start the pipeline search task during the mission, the robot will navigate to a starting point in the target region, corresponding to its bottom-left corner. Thereafter, the robot will be directed to follow the path planned using the boustrophedon method. If the pipeline is detected while following the path, the search procedure will be stopped as it finished successfully. Otherwise, the robot will continue to completely cover the target region.

### 5.2.5 Results

Figure 5.10 shows the waypoints conforming the explicit path generated for the one-cell decomposition of the workspace. These waypoints are georeferenced in UTM coordinates. Figure 5.11 shows the generated path plotted in Google Earth.

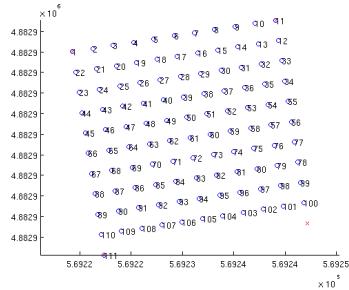


Figure 5.10: Waypoints conforming the generated boustrophedon path for the SAUC-E pipeline search task. Points are georeferenced using UTM coordinates.



Figure 5.11: Generated boustrophedon path for the SAUC-E pipeline search task plotted over the SAUC-E competition arena on Google Earth (in white in the center of the image).

Sparus AUV was directed through the generated path during the SAUC-E competition in several trials for the pipeline search task. The path was successfully followed by the robot until a positive detection was assessed by the pipeline detection system. However, due to the tough visibility conditions, the detection assessment turned out to be a false positive in all attempts. In any case, the planned search path was successfully executed and the target area was indeed covered.



# Chapter 6

## Conclusion

This chapter concludes the work presented throughout this document by first pointing out the contributions made in this research work. Finally, some interesting future research issues are commented.

### 6.1 Contributions

First, we exhaustively reviewed the state of the art on coverage path planning. We classified the methods in heuristic approaches, cellular decomposition approaches and other approaches including APF, template-based models, neural networks and fuzzy logic and other miscellaneous approaches. We also reviewed several multi-robot coverage path planning methods and, finally, we accounted for methods specifically focused on underwater environments. We pointed out the strengths and weaknesses of the reviewed methods and concluded that an optimal and universal coverage path planning algorithm has not been yet developed, so there remains a need for further research. In the particular case of underwater environments, no studies on coverage path planning addressing the important issue of current fields in underwater environments were found. On the other hand, no studies considering the constraints that some applications might impose on the coverage path, such as passing through certain points more than once in order to perform feature matching, were found. Part of this work will be submitted to an international journal.

Second, we presented a study of methods and available technologies for underwater obstacle detection. Based on our acoustic sensing review, we concluded that multibeam forward-looking sonars offer the best performance for underwater obstacle detection and characterization, though at more elevated cost than other solutions. On the other hand, we pointed out that the state of the art for automated obstacle detection and characterisation on sonar imagery consists in applying computer vision image processing techniques.

Third, we implemented and applied coverage path planning methods to real-world datasets and extracted results. One one hand, we performed offline coverage path planning

with the Morse-based boustrophedon decomposition on a real-world bathymetric dataset. On the other hand, we applied the aforementioned method on a real underwater search mission with the Sparus AUV as part of the SAUC-E competition.

Finally, it is worth mentioning that this thesis opens a new door in the Computer Vision and Robotics Group (VICOROB) from the University of Girona, where no previous work on coverage path planning had been carried out. This also contributes to the workload of the TRIDENT project introduced in the first and second chapters of this thesis.

## 6.2 Future work

We performed an in-depth analysis of the state of the art of coverage path planning and, as we mentioned already, no updated surveys in this research topic were found. Thus, right after submitting the present MSc thesis, we will focus on writing a survey on coverage path planning which will be submitted to an international journal.

A need for further experimental work on the application of coverage path planning methods in underwater environments remains. Addressing this issue, a second research stay at NURC is planned where coverage path planning methods will be applied in underwater environments taking advantage of NURC's facilities.

The application of coverage path planning methods is intrinsically related to the study of underwater acoustic obstacle detection. The state-of-the-art sonar technology available at NURC will contribute to a deeper treat of this issue during the planned stay at the research center.

Also, another research stay aiming to gather deeper expertise in coverage path planning methods is planned mid-term.

As previously stated, this thesis is located in the research framework of the TRIDENT European research project. The progress on the development of the present research work aims to contribute achieve the project objectives.

A PhD thesis will follow this MsC thesis. Naturally, the present research work will contribute to this PhD thesis. Next, the PhD thesis planning is discussed.

### 6.2.1 PhD thesis planning

Following this MsC thesis, we will go further investigating coverage path planning methods for AUVs as the research line of a PhD thesis under a FI grant awarded by the Generalitat de Catalunya. Our main goal is the proposal of a coverage path planning method for underwater environments.

As stated above, just after the submisison of this document we will focus our efforts on submitting a survey on coverage path planning to an international journal.

Also at immediate term, a second research stay at NURC is planned. The stay will focus on the application of coverage path planning methods to real underwater environments and on performing a deeper study of acoustic sensing techniques for underwater obstacle detection.

At mid term, another research stay is planned aiming to gain expertise in coverage path planning.

Submission for publication of the research work performed during the research stays is planned.

At long term, we plan to state our proposal of a coverage path planning method for underwater environments and submit it to international conferences and international journals.



# Bibliography

- Acar, E. and Choset, H.: 2002, Sensor-based coverage of unknown environments: Incremental construction of morse decompositions, *International Journal of Robotics Research* **21**(4), 345–366.
- Acar, E. U. and Choset, H.: 2000, Critical point sensing in unknown environments, *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*.
- Acar, E. U., Choset, H. and Lee, J. Y.: 2006, Sensor-based coverage with extended range detectors, **22**(1), 189–198.
- Acar, E. U., Choset, H., Rizzi, A. A., Atkar, P. N. and Hull, D.: 2002, Morse decompositions for coverage tasks, *International Journal of Robotics Research* **21**(4), 331–344.
- Acar, E. U., Choset, H., Zhang, Y. and Schervish, M.: 2003, Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods, *International Journal of Robotics Research* **22**(7-8), 441–466.
- Alvarez, A., Caiti, A. and Onken, R.: 2004, Evolutionary path planning for autonomous underwater vehicles in a variable ocean, **29**(2), 418–429.
- Arkin, E. M., Fekete, S. and Mitchell, J.: 1993, The lawnmower problem, *Proceedings of 5th Canadian Conference on Computational Geometry*, pp. 461–466.
- Arkin, E. M. and Hassin, R.: 1994, Approximation algorithms for the geometric covering salesman problem, *Discrete Applied Mathematics* **55**(3), 197–218.
- Atkar, P. N., Choset, H., Rizzi, A. A. and Acar, E. U.: 2001, Exact cellular decomposition of closed orientable surfaces embedded in 3, *Proc. ICRA Robotics and Automation IEEE Int. Conf*, Vol. 1, pp. 699–704.
- Barraquand, J., Langlois, B. and Latombe, J. C.: 1992, Numerical potential field techniques for robot path planning, *IEEE Transactions on Man and Cybernetics* **22**(2), 224–241.
- Borenstein, J. and Koren, Y.: 1989, Real-time obstacle avoidance for fast mobile robots, **19**(5), 1179–1187.
- Bosse, M., Nourani-Vatani, N. and Roberts, J.: 2007, Coverage algorithms for an underactuated car-like vehicle in an uncertain environment, *Proc. IEEE Int Robotics and Automation Conf*, pp. 698–703.

- Butler, Z. J., Rizzi, A. A. and Hollis, R. L.: 1999, Contact sensor-based coverage of rectilinear environments, *Proc. IEEE Int Intelligent Control/Intelligent Systems and Semiotics Symposium*, pp. 266–271.
- Butler, Z. J., Rizzi, A. A. and Hollis, R. L.: 2000, Complete distributed coverage of rectilinear environments, *Proc. of the Workshop on the Algorithmic Foundations of Robotics*.
- Cao, Z. L., Huang, Y. and Hall, E. L.: 1988, Region filling operations with random obstacle avoidance for mobile robotics, *Journal of Robotic Systems* **5**(2), 87–102.
- Choi, Y.-H., Lee, T.-K., Baek, S.-H. and Oh, S.-Y.: 2009, Online complete coverage path planning for mobile robots based on linked spiral paths using constrained inverse distance transform, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems IROS 2009*, pp. 5788–5793.
- Choset, H.: 2000, Coverage of known spaces: the boustrophedon cellular decomposition, *Autonomous Robots* **9**, 247–253.
- Choset, H.: 2001, Coverage for robotics—a survey of recent results, *Annals of Mathematics and Artificial Intelligence* **31**, 113–126.
- Choset, H., Acar, E., Rizzi, A. A. and Luntz, J.: 2000, Exact cellular decompositions in terms of critical points of morse functions, *Proc. IEEE Int. Conf. Robotics and Automation ICRA '00*, Vol. 3, pp. 2270–2277.
- Choset, H., Lynch, K., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L. and Thrun, S.: 2005, *Principles of Robot Motion: Theory, Algorithms, and Implementation*, The MIT Press.
- Choset, H. and Pignon, P.: 1997, Coverage path planning: The boustrophedon cellular decomposition, *Proceedings of International Conference on Field and Service Robotics*.
- De Carvalho, R. N., Vidal, H. A., Vieira, P. and Ribeiro, M. I.: 1997, Complete coverage path planning and guidance for cleaning robots, *Proc. IEEE Int Industrial Electronics ISIE '97. Symp.*, Vol. 2, pp. 677–682.
- Fang, C. and Anstee, S.: 2010, Coverage path planning for harbour seabed surveys using an autonomous underwater vehicle, *Proc. OCEANS 2010 IEEE - Sydney*, pp. 1–8.
- Farsi, M., Ratcliff, K., Johnson, J. P., Allen, C. R., Karam, K. Z. and Pawson, R.: 1994, Robot control system for window cleaning, *Proc. American Control Conf*, Vol. 1, pp. 994–995.
- Fu, Y. and Lang, S. Y. T.: 1999, Fuzzy logic based mobile robot area filling with vision system for indoor environments, *Proc. IEEE Int. Symp. Computational Intelligence in Robotics and Automation CIRA '99*, pp. 326–331.
- Gabriely, Y. and Rimon, E.: 2002, Spiral-stc: an on-line coverage algorithm of grid environments by a mobile robot, *Proc. IEEE Int. Conf. Robotics and Automation ICRA '02*, Vol. 1, pp. 954–960.

- Gage, D. W.: 1994, Randomized search strategies with imperfect sensors, *Proc. SPIE, Mobile Robots VIII–Int. Soc. Optical Engineering*, Boston, MA, pp. 270–279.
- Garcia, E. and de Santos, P. G.: 2004, Mobile-robot navigation with complete coverage of unstructured environments, *Robotics and Autonomous Systems* **46**, 195–204.
- Gonzalez, E., Alvarez, O., Diaz, Y., Parra, C. and Bustacara, C.: 2005, Bsa: A complete coverage algorithm, *Proc. IEEE Int. Conf. Robotics and Automation ICRA 2005*, pp. 2040–2044.
- Guo, J., Cheng, S.-W. and Liu, T.-C.: 1998, Auv obstacle avoidance and navigation using image sequences of a sector-scanning sonar, *Proceedings of the 1998 International Symposium on Underwater Technology*.
- Hert, S., Tiwari, S. and Lumelsky, V.: 1996, A terrain-covering algorithm for an auv, *Autonomous Robots* **3**, 91–119.
- Hofner, C. and Schmidt, G.: 1995, Path planning and guidance techniques for an autonomous mobile cleaning robot, *Robotics and Autonomous Systems* **14**(2-3), 199 – 212. Research on Autonomous Mobile Robots.
- URL:** <http://www.sciencedirect.com/science/article/B6V16-3Y5FDNC-G/2/dd9431bc9cf11cae175326df6c4c4f5c>
- Huang, W. H.: 2001, Optimal line-sweep-based decompositions for coverage algorithms, *Proc. ICRA Robotics and Automation IEEE Int. Conf*, Vol. 1, pp. 27–32.
- Hurtós, N.: 2009, *Integration of optical and acoustic sensor data for 3d underwater scene reconstruction*, Master's thesis, University of Girona.
- Jimenez, P. A., Shirinzadeh, B., Nicholson, A. and Alici, G.: 2007, Optimal area covering using genetic algorithms, *Proc. IEEE/ASME international conference Advanced intelligent mechatronics*, pp. 1–5.
- Koditschek, D. E. and Rimon, E.: 1990, Robot navigation functions on manifolds with boundary, *Advances in Applied Mathematics* **11**, 412–442.
- Kruger, D., Stolkin, R., Blum, A. and Briganti, J.: 2007, Optimal auv path planning for extended missions in complex, fast-flowing estuarine environments, *Proc. IEEE Int Robotics and Automation Conf*, pp. 4265–4270.
- Kurabayashi, D., Ota, J., Arai, T., Ichikawa, S., Koga, S., Asama, H. and Endo, I.: 1996, Cooperative sweeping by multiple mobile robots with relocating portable obstacles, *Proc. IEEE/RSJ Int Intelligent Robots and Systems '96, IROS 96 Conf*, Vol. 3, pp. 1472–1477.
- Kurabayashi, D., Ota, J., Arai, T. and Yoshida, E.: 1996, Cooperative sweeping by multiple mobile robots, *Proc. Conf. IEEE Int Robotics and Automation*, Vol. 2, pp. 1744–1749.
- Lang, S. Y. T. and Chee, B.-Y.: 1998, Coordination of behaviours for mobile robot floor cleaning, *Proc. Conf. IEEE/RSJ Int Intelligent Robots and Systems*, Vol. 2, pp. 1236–1241.

- Latombe, J. C.: 1991, *Robot motion planning*, Kluwer Academic Publishers.
- LaValle, S. M. and Kuffner, J. J., J.: 1999, Randomized kinodynamic planning, *Proc. IEEE Int Robotics and Automation Conf*, Vol. 1, pp. 473–479.
- Lee, T.-S., Choi, J.-S., Lee, J.-H. and Lee, B.-H.: 2009, 3-d terrain covering and map building algorithm for an auv, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems IROS 2009*, pp. 4420–4425.
- Lu, Y. and Sang, E.: 1998, Underwater target's size/shape dynamic analysis for fast target recognition using sonar images, *Proceedings of the 1998 International Symposium on Underwater Technology*.
- Lumelsky, V. J., Mukhopadhyay, S. and Sun, K.: 1990, Dynamic path planning in sensor-based terrain acquisition, **6**(4), 462–472.
- Luo, C., Yang, S. X., Stacey, D. A. and Jofriet, J. C.: 2002, A solution to vicinity problem of obstacles in complete coverage path planning, *Proc. IEEE Int. Conf. Robotics and Automation ICRA '02*, Vol. 1, pp. 612–617.
- Lurton, X.: 2002, *An introduction to underwater acoustics: principles and applications*, Springer.
- Milnor, J.: 1963, *Morse Theory*, Princeton University Press.
- Moravec, H. and Elfes, A.: 1985, High resolution maps from wide angle sonar, *Proc. IEEE Int. Conf. Robotics and Automation*, Vol. 2, pp. 116–121.
- Najjaran, H. and Kircanski, N.: 2000, Path planning for a terrain scanner robot, *Proc. 31st Int. Symp. Robotics*, Montreal, QC, Canada, pp. 132–137.
- Oksanen, T. and Visala, A.: 2009, Coverage path planning algorithms for agricultural field machines, *Journal of Field Robotics* **26**, 651–558.
- Ollis, M. and Stentz, A.: 1996, First results in vision-based crop line tracking, *Proc. Conf. IEEE Int Robotics and Automation*, Vol. 1, pp. 951–956.
- Ollis, M. and Stentz, A.: 1997, Vision-based perception for an automated harvester, *Proc. IEEE/RSJ Int Intelligent Robots and Systems IROS '97. Conf*, Vol. 3, pp. 1838–1844.
- Park, J. Y. and Lee, K. D.: 1997, A study on the cleaning algorithm for autonomous mobile robot under the unknown environment, *Proc. Workshop th IEEE Int Robot and Human Communication RO-MAN '97*, pp. 70–75.
- Petillot, Y., Ruiz, I. T. and Lane, D. M.: 2001, Underwater vehicle obstacle avoidance and path planning using a muli-beam forward looking sonar, *IEEE Journal of Oceanic Engineering* **26**.
- Pirzadeh, A. and Snyder, W.: 1990, A unified solution to coverage and search in explored and unexplored terrains using indirect control, *Proc. Conf. IEEE Int Robotics and Automation*, pp. 2113–2119.

- Qiu, X., Song, J., Zhang, X. and Liu, S.: 2006, A complete coverage path planning method for mobile robot in uncertain environments, *Proc. Sixth World Congress Intelligent Control and Automation WCICA 2006*, Vol. 2, pp. 8892–8896.
- Rekleitis, I. M., Dudek, G. and Milios, E. E.: 2000, Multi-robot collaboration for robust exploration, *Proc. IEEE Int. Conf. Robotics and Automation ICRA '00*, Vol. 4, pp. 3164–3169.
- Rekleitis, I., New, A., Rankin, E. and Choset, H.: 2008, Efficient boustrophedon multi-robot coverage: an algorithmic approach, *Annals of Mathematics and Artificial Intelligence* **52**, 109–142. 10.1007/s10472-009-9120-2.  
**URL:** <http://dx.doi.org/10.1007/s10472-009-9120-2>
- Rimon, E. and Koditschek, D. E.: 1992, Exact robot navigation using artificial potential functions, *IEEE Transactions on Robotics and Automation* **8**(15), 501–518.
- Russell, R. A.: 1997, Heat trails as short-lived navigational markers for mobile robots, *Proc. Conf. IEEE Int Robotics and Automation*, Vol. 4, pp. 3534–3539.
- Souignac, M.: 2011, Feasible and optimal path planning in strong current fields, **27**(1), 89–98.
- Spires, S. V. and Goldsmith, S. Y.: 1998, Exhaustive geographic search with mobile robots along space-filling curves, *Proceedings of the 1st International Workshop in Collective Robotics*.
- Stack, J. R. and Smith, C. M.: 2003, Combining random and data-driven coverage planning for underwater mine detection, *Proc. OCEANS 2003*, Vol. 5, pp. 2463–2468.
- Trucco, A. and Curletto, S.: 2003, Extraction of 3-d information from sonar image sequences, *IEEE Transactions on Systems, Man, and Cybernetics* **33**, 687–699.
- Tse, P. W., Lang, S., Leung, K. C. and Sze, H. C.: 1998, Design of a navigation system for a household mobile robot using neural networks, *Proc. IEEE Int. Joint Conf. IEEE World Congress Computational Intelligence Neural Networks*, Vol. 3, pp. 2151–2156.
- Wagner, I. A., Lindenbaum, M. and Bruckstein, A. M.: 1999, Distributed covering by ant-robots using evaporating traces, **15**(5), 918–933.
- Wang, X. and Syrmos, V. L.: 2009, Coverage path planning for multiple robotic agent-based inspection of an unknown 2d environment, *Proc. 17th Mediterranean Conf. Control and Automation MED '09*, pp. 1295–1300.
- Williams, D. P.: 2010, On optimal auv track-spacing for underwater mine detection, ICRA.
- Wong, S. C. and MacDonald, B. A.: 2003, A topological coverage algorithm for mobile robots, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS 2003)*, Vol. 2, pp. 1685–1690.
- Xu, L. and Stentz, A.: 2011, An efficient algorithm for environmental coverage with multiple robots, *Proceedings of the 2011 IEEE International Conference on Robotics & Automation*.

- Yang, S. X. and Luo, C.: 2004, A neural network approach to complete coverage path planning, **34**(1), 718–724.
- Yasutomi, F., Yamada, M. and Tsukamoto, K.: 1988, Cleaning robot control, *Proc. Conf. IEEE Int Robotics and Automation*, pp. 1839–1841.
- Zelinsky, A., Jarvis, R. A., Byrne, J. C. and Yuta, S.: 1993, Planning paths of complete coverage of an unstructured environment by a mobile robot, *Proceedings of International Conference on Advanced Robotics*, pp. 533–538.