

A Geometrical Approach to Estimate the Coverage Measure of the Area Explored by a Robot.

Maria Luiza Costa Vianna

2023

Contents

1	Introduction	5
1.1	Planet Ocean	7
1.2	Robotics Terminology	13
1.3	Thesis Context	14
1.3.1	Outlines and Contributions	16
1.4	Outline	17
I	Theoretical Background	18
2	Sonars	20
2.1	Introduction	21
2.2	Side-Scan Sonar	23
2.2.1	Waterfall and Mosaic Images	26
2.2.2	Features Detection in Sonar Images	28
2.2.3	Hypotheses Adopted on the Side-Scan Sonar Model	33
2.3	Coverage Measure to Aid in the Post-Treatment Process	36
2.4	Conclusion	38
3	Interval Analysis	39
3.1	Introduction	40
3.2	Set Theory	41
3.2.1	Operations on Sets	41
3.2.2	Set Image	43
3.3	Interval Analysis	43
3.3.1	Intervals	43
3.3.2	Boxes or Interval Vectors	46
3.3.3	Inclusion Functions	47
3.4	Set-Valued Constraint Satisfaction Problems (SVCSPs)	49
3.4.1	Definition of a SVCSP	50
3.4.2	Contractors	50
3.4.3	Separators	53
3.4.4	Image Contractor	56
3.5	Subpaving	61
3.5.1	Set Inversion Via Interval Analysis (SIVIA) Algorithm .	62
3.5.2	Paver	64
3.6	Dedicated Sets	69

3.6.1	Interval of Sets or Thick Sets	69
3.6.2	Interval of Functions or Tubes	70
3.7	Detection of Self-Intersections on Tubes	73
3.8	Conclusion	77
4	Topology	79
4.1	Common Euclidean Subspaces	81
4.2	Topological Spaces	82
4.2.1	Common Examples of Topologies	84
4.2.2	Representation of Arbitrary Sets on Topological Spaces .	84
4.2.3	Convergence	86
4.2.4	Continuity	87
4.2.5	Basis for a Topology	88
4.3	Interesting Topological Spaces	89
4.3.1	Hausdorff Spaces	89
4.3.2	Manifolds	91
4.3.3	Quotient Spaces	94
4.4	Connectedness and Compactness	96
4.4.1	Connected Space	96
4.4.2	Compact Space	97
4.4.3	Compactification	98
4.5	The Fundamental Group	99
4.5.1	Groups and Isomorphism	99
4.5.2	Homotopy	100
4.5.3	Homotopy Equivalence	101
4.5.4	Construction of the Fundamental Group	102
4.6	CW Complexes	104
4.7	Homology	105
4.7.1	Homology Group	106
4.8	Topological Degree	108
4.8.1	Winding Number	109
4.9	Conclusion	110
II	Contributions	112
5	Coverage Measure and Explored Area	114
5.1	One-Dimensional Sensors	115
5.1.1	Problem Statement	115
5.1.2	Coverage Measure and the Topological Degree	121
5.1.3	Dealing with Uncertainties	129

5.2	Two-Dimensional Sensors	132
5.2.1	Problem Statement	132
5.2.2	Coverage Measure and Topological Degree	143
5.2.3	Sweeping backwards and Uncertainties	146
5.3	Conclusion	147
6	Area Characterization	148
6.1	Problem Definition	149
6.2	Creating the Mission Contour	151
6.3	Computing the Extended Winding Number	157
6.3.1	Method 1: Turning Around	160
6.3.2	Method 2: Alexander Theorem	169
6.3.3	Method 3: Jordan Curves Sum	189
6.4	Waterfall Characterization	193
6.5	Conclusion	199
7	Experiments	200
7.1	Daurade	201
7.2	SpeBoat	206
7.3	Conclusion	220
8	Conclusions and Prospects	221
8.1	Summary	222
8.2	Prospects	222

Chapter 1

Introduction

"Given what you know about the world, where should you move to gain as much new information as possible?" - Brian Yamauchi [1]

Contents

1.1	Planet Ocean	7
1.2	Robotics Terminology	13
1.3	Thesis Context	14
1.3.1	Outlines and Contributions	16
1.4	Outline	17

A mobile robot is a specialized type of robot that, unlike stationary robotic systems that remain fixed in one location, has the capability to change its position and navigate within its environment [2]. This characteristic allows this type of robots to mimic human capabilities, and they are increasingly being used to carry out dangerous tasks that otherwise would put human lives at risk, such as bomb disposal [3], firefighting [4], and Search and Rescue (SAR) missions [5]. Their use in these situations can considerably reduce the risk to human responders while providing more detailed and accurate information about the situation. Additionally, these robotic platforms can be equipped with specialized tools that enable them to perform a wide range of tasks that would be difficult or impossible for humans to do. For example, thermal cameras for fault detection on transmission lines [6], and soft grippers for a non-destructive sampling of reefs in deep waters [7].

In the context of these operations, mobile robots often need to perform an area-covering mission. During these missions, the robot must completely cover, with its sensors or tools, an area of interest. On a domestic scale, a simple example of area-covering mission can be found in autonomous vacuum cleaning robots. These robots autonomously navigate indoor environments, covering the entire area to remove dust and debris.

These missions are also employed for thoroughly searching or monitoring a designated part of the environment, allowing a complete understanding of the situation or identification of potential threats or opportunities. For example, we have Mine CounterMeasure (MCM) missions [8] and SAR operations. For the latter, robots are deployed to explore areas stricken by disasters, such as earthquake-affected areas or wildfires. Their mission is to cover a whole area of interest, searching for survivors, assessing the extent of damage, and delivering life-saving supplies.

Environmental monitoring is another significant application of area-covering missions. Robots are deployed in natural ecosystems, including forests and oceans, to collect data on environmental conditions, wildlife populations, and pollution levels. This information is essential for conservation efforts and scientific research. In [9], for example, an Autonomous Underwater Vehicle (AUV) was deployed for collecting observations over regular time intervals within an area of interest. The objective was to examine the impacts of human interventions and climate change on benthic organisms, which inhabit the seafloor.

In this thesis, our main motivation is area-covering missions for ocean exploration, even though the research conducted can be applied to different scenarios. Throughout the remainder of this chapter, we discuss our main motivations to make this environment our primary focus and the importance of

developing methods for efficient marine exploration. We then present the specific challenges encountered in this domain, providing an insight into how this thesis finds application in this scenario. Finally, we summarize the principal contributions of this study, and we present an overview of this document's roadmap.

1.1 Planet Ocean

"How inappropriate to call this planet Earth when it is quite clearly Ocean." (- Arthur C. Clarke)

Oceans cover around 71% of the Earth's surface and hold about 96.5% of all Earth's water [10]. Life on Earth is thought to have originated in the ocean over three billion years ago, and the ocean continues to support life on earth. Phytoplankton, for example, are responsible for producing a significant portion of the world's oxygen, around 50% according to scientists [11].

Furthermore, oceans are an essential source of sustenance for millions of people globally, playing a central role in maintaining food security.

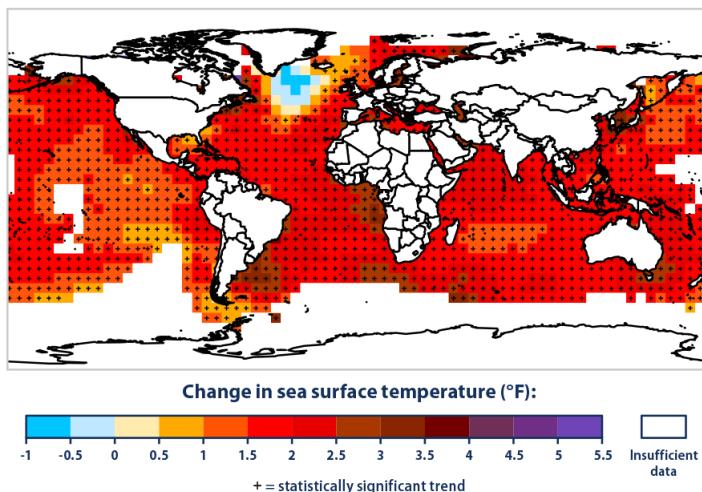


Figure 1.1: Image credits: [12]. Change in sea surface temperature between 1901 and 2020.

However, the stability of this security is increasingly threatened by the rising temperatures of our oceans, as depicted in Figure 1.1. The consequences of ocean warming are profound, causing disruptions within the marine food chain and elevating the risk of fishery collapses [13]. Such an occurrence would have

severe humanitarian consequences for our planet. These potential crises encompass not only food scarcity in coastal regions but also political instabilities and conflicts arising from competition for food.

To address these challenges and mitigate the impacts of ocean warming, gathering data and monitoring are crucial. Understanding how ecosystems respond to these changes is essential for recognizing and planning future responses. And while data collection alone does not resolve the issue of ocean warming, it allows the development of solutions to reduce the impacts locally. For example, through the implantation of artificial reefs for improving coral recovery [14].

Underwater exploration also holds great significance in the military sector. According to the US Navy [15], Autonomous Underwater Vehicles (AUVs) play a vital role in various operations, including Intelligence, Surveillance, and Reconnaissance (ISR) missions, Mine Countermeasure (MCM) missions, Anti Submarine Warfare (ASW), among others.

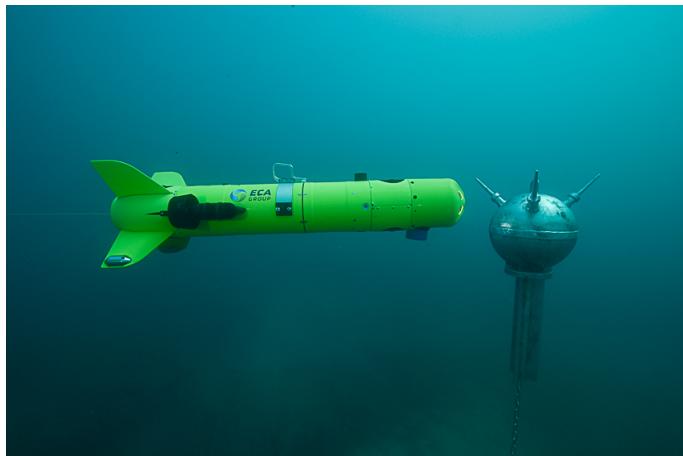


Figure 1.2: Image credits: ECA Group. AUV inspecting underwater mine.

In Figure 1.2, we observe an AUV from ECA Group engaged in an MCM mission, where it is employed on the recognition of an underwater mine. Replacing human effective in MCM tasks is essential for reducing risks in these operations. In addition, mines often contain explosives and hazardous materials. When they deteriorate, they can release toxic substances into the marine environment [16]. Therefore, MCM missions are essential, not only to minimize possible human loss and to reduce humanitarian hazards, but also for minimizing environmental impacts.

Despite the vital importance of underwater exploration, we still know relatively little about the undersea domain. More precisely, according to the

American National Oceanic and Atmospheric Administration (NOAA), more than 80% of the oceans remain "unexplored" [17].

In fact, satellites allowed us to map the whole ocean floor with a small resolution, enough to identify bumps and dips on the seafloor. A more precise map was made by sonar systems embedded aboard ships. However, since these systems can only measure to a certain distance range below the ship in which they are mounted, less than 20% of the oceans were mapped under this configuration, justifying the amount of seabed considered to be unexplored by researchers.

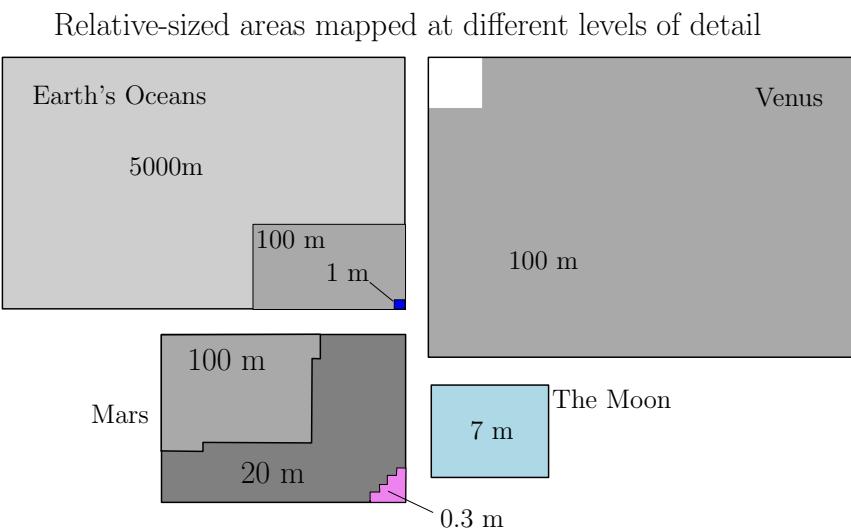


Figure 1.3: This comparative figure demonstrates how much of each one of these regions have been explored and with each precision. For example, 95% of the surface of Venus has been explored with a precision of up to 100m, the other 5% remains unexplored. And the whole surface of the moon has been explored with a precision of up to 7m. Credits to Jeffrey Marlow.

For this reason, it is common to hear that we know less about our oceans than about the surface of other planets. Figure 1.3 provides a comparison between the precision of the available mapping of earth's oceans and the mapping of Venus, Mars and the moon. We can see on the image that the entirety of the earth's oceans has been mapped with up to 5km of precision, this corresponds to satellite mapping. A smaller part, has a precision of up to 100m, corresponding to areas mapped by sonar systems embedded aboard surface vehicles. This precision is still not enough for allowing an identification of small size features on the ocean floor such as deep sea vents, underwater mines and wreckage of missing ships and air-crafts. For that, we need to get sonar systems closer to the seabed, for example, using submersibles. However, at

present, very little of the ocean floor has been mapped on these conditions, with a precision of up to 1 meter.

From June 2014 to June 2017, a survey of 120000 km^2 , approximately 0.03% of the ocean's surface, was performed in the search of the Malaysian Airlines Flight 370, which disappeared on March 2014 [18]. Although some first indices of debris were supposedly found in 2023, the remains of the wrecked aircraft have still not been found.

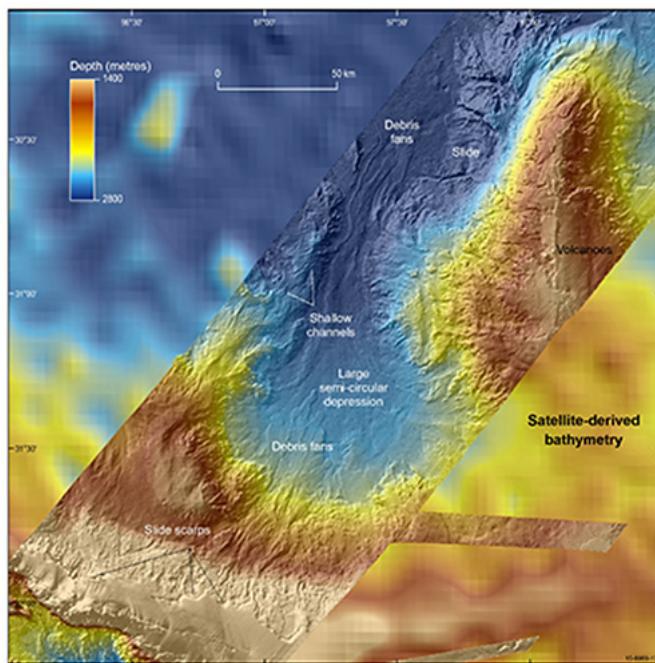


Figure 1.4: Image credits: [19]. Difference in resolution between multi-beam sonar and satellite-based bathymetry.

Seabed mapping for the Indian Ocean on this area before this search had an average resolution, per pixel, of more than 5km^2 . The new map has a resolution smaller than 0.01 km^2 [19]. In Figure 1.4 we can see the difference between the sonar bathymetry, acquired during the search, and the previous satellite-derived bathymetry, illustrating how submersible robotics systems improve the mapping quality.

The mission for the search of this aircraft was classified as the most expensive search in aviation history, with a huge deployment of maritime means. Its unfortunate lack of success demonstrates the complexity of exploring the underwater environment, that presents many challenges. Some of these challenges are now discussed.

Challenges

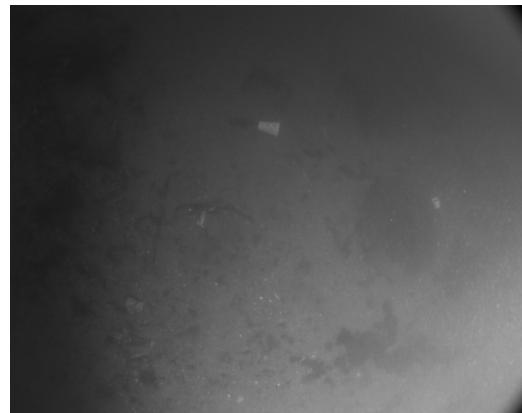
High pressure, corrosive salinity, that can also affect buoyancy, and extreme temperature variations in the water column. These are only some of the many challenges encountered in the underwater environment. Adding to this list of challenges is the limited propagation of electromagnetic waves in water. Unlike in the air, where electromagnetic waves can travel long distances with minimal attenuation, water significantly attenuates these waves. As a consequence, GPS signals are practically unusable underwater, making precise positioning, navigation, and mapping of underwater environments complex tasks. As a result, underwater vehicles often need to rely on alternative navigation methods like inertial navigation systems [20] and acoustic navigation [21]. While these alternatives have shown promising results [22], they often require complex technologies, that have to be robust and precise enough for dealing with the challenging underwater conditions.

Communication underwater is also a significant challenge resultant to the lack of electromagnetic wave propagation. Consequently, data collected by underwater vehicles typically must be stored onboard until the vehicle resurfaces because real-time transmission is often unfeasible. All of these constraints result in systems that have to carry onboard precise sensors, expensive navigation systems, and storage solutions.

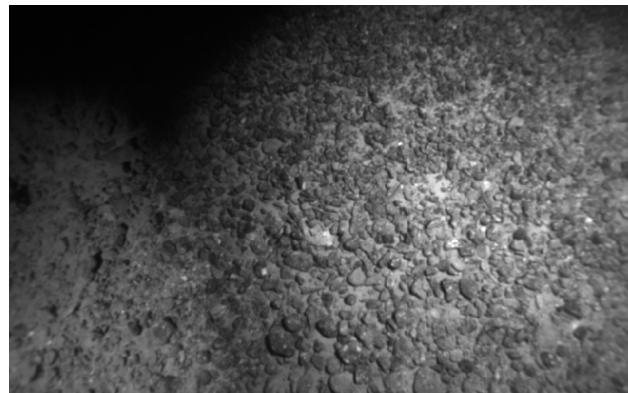
Moreover, observing the underwater environment presents its own set of difficulties. Inexpensive sensors like cameras provide high-dimensional measurements with rich information [24]. Furthermore, cameras are a natural choice for replicating human perception, and are probably the best sensor to replace the human eyes in autonomous systems. However, due to the poor visibility and the homogeneity of features in the underwater environment, extract meaningful information from underwater cameras is very challenging, as exemplified in Figure 1.5. Acoustic imagery, acquired through sonar systems, is often preferred over cameras in this context, but sonar data is less intuitive and often requires specialized analysis. Chapter 2 is dedicated to presenting sonars and its own challenges.

The consequence of all of this is a ratio cost/exploration for underwater robots that is usually too high and simply not accessible. For covering very small areas, the cost can be very elevated. And we have to add to this the cost of the deployment of the robots, that usually requires a team of engineers offshore and big ships to carry all the necessary material, see Figure 1.6.

Reducing the cost of exploration can be achieved by enhancing the robustness of embedded navigation and observation algorithms in submersibles, as well as by improving the autonomy of these robots to adapt to new scenarios. A robot with the ability to adapt and reconfigure its mission without



(a)



(b)

Figure 1.5: Image Credits: Dataset Aqualoc [23] (a): Water turbidity; (b): Lack of diversity in textures.



Figure 1.6: Image credits: DGA/GESMA. Deployment of the AUV Daurade in the roadstead of Brest from a ship.

the necessity of resurfacing and waiting for human intervention will conserve its battery power and discretion. As we increasingly rely on algorithmic solutions, however, the need for guarantees becomes imperative. It is essential to be sure about how the system will perform prior to deployment. In other words, once a mission has been defined, it is crucial to thoroughly verify potential outcomes while accounting for all associated uncertainties. This approach minimizes risks, enhances the predictability and success of underwater missions and reduce costs by avoiding non-conclusive missions and possible lost of material.

The context of this thesis is stated within this framework, aiming to create guaranteed methods for verifying area-covering missions. The primary emphasis continues to be on the underwater environment, yet the theories and algorithms developed during this work can be extended to various other scenarios. Before presenting the thesis context we present some important terms in robotics that will be used through this document.

1.2 Robotics Terminology

Making a system autonomous means to make it capable of performing its tasks without human intervention. To achieve this level of autonomy, we must furnish the machine with sufficient means to perceive its surrounding environment, mirroring the natural way in which humans interact with the world. With this purpose, in robotics, we use sensors, that work as a replacement of

our senses and natural perception. We can classify sensors according to the type of data they gather:

- Proprioceptive Sensors: These sensors acquire data about the robot itself. These sensors provide information that help the robot understand its own position, orientation, and movement. They are essential for navigation and control. Some common examples of proprioceptive sensors are: gyroscopes, accelerometers and encoders.
- Exteroceptive Sensors: These sensors are used to observe the external environment, and are also called observation or exploration sensors. They allow the robot to perceive and interact with the world. Cameras, lidars and sonars are examples of exteroceptive sensors.

Exteroceptive sensors can be categorized based on the dimensionality of the area they cover or interact with. One-dimensional sensors typically measure a property or parameter along a single linear axis. They are usually called line-sweep or sweep sensors because they must sweep the environment in order to create a two-dimensional representation of its surroundings. Side-scan sonars and small-aperture radars are examples of one-dimensional sensors. Cameras, on the other hand, are an example of two-dimensional sensors. Depth cameras, like kinetic, are three-dimensional because they capture depth information in addition to 2D color imagery.

Using this terminology, we present this thesis context next.

1.3 Thesis Context

Determining the area explored by a mobile robot during an area-covering mission is important to establish if the mission is successful. It is also essential for validating path-planning algorithms that will lead to complete coverage of an area of interest or complete avoidance of an area of risk. Overall, determining the explored area is essential for ensuring efficient and safe operations, planning future actions, and gaining valuable insights from the acquired data.

The most common approach for determining the explored area is through grid based methods. In this approach, the area to be explored is divided into a grid, or cells, of the same size and shape, whose union approximates the target region. It is assumed that the robot explores the whole cell once it enters the cell, and then the status for this particular cell is updated. The idea was first introduced by [25], and it has been largely applied in the planning of coverage paths for mobile robots [26] [27] [28]. This approach presents some limitations, mostly related to the fact that the space is represented by a fixed structure

of cells with the same size. If the grid cells are too small, it can result in extremely large data structures and increased computational complexity. On the other hand, if the cells are too large, fine details of coverage may be lost. Furthermore, storing and updating grids in real-time can be a challenge for resource-constrained robots.

In [29] and [30], an alternative method for estimating the explored area by a robot is introduced, using interval analysis. This approach does not present the problems mentioned above for the grid-based methods. Instead of relying on fixed grid cells of uniform size, this method produces an adaptive subpaving to represent the explored area, capable of representing different shapes with more precision. Moreover, this approach determines the explored area from very basic information regarding the robot's state and the range of visibility of its sensors. It eliminates the need for storing and updating predefined regions, significantly reducing memory consumption. Furthermore, it allows taking into consideration the crucial issue of uncertainty associated to a mobile robot's trajectory, proposing a guaranteed estimation of the explored area. This approach, however, does not perform well when we have to determine the area explored by one-dimensional exploration sensors. And as it is discussed in Chapter 2, the representation of the area explored by sonars is often one-dimensional.

The approach that we propose in this work presents the advantages of the interval method proposed in the literature, and it is capable of efficiently estimating the area explored by one-dimensional sensors. In addition, it is also able to determine the coverage measure of points in the environment. The coverage measure represents how many times a point was covered by the robot's sensors or tools, in other words, how many times it was explored.

Counting the number of times an area was explored is of interest for different reasons, for example, when assessing revisiting missions. In these missions the robot is required to come back to a previous point, therefore to revisit it, to improve the quality of information collected around this point through redundancy. Indeed, studies have shown that target classification improves dramatically when a multi-view approach is adopted. Usually, single-view approaches do not provide enough information to make a confident identification with, for example, Synthetic Aperture Sonars (SAS) [31] [32], Synthetic Aperture Radars [33]. A multi-view method is also essential when recognizing or reconstructing 3-dimensional objects from 2-dimensional data such as camera images [34]. In these examples, counting how many times a point or an area, as a set of points, has already been explored will be essential to determine the mission completeness. On the contrary, if the robot is not supposed to cover areas previously visited, the coverage measure will be useful for planning

optimal paths, reducing unnecessary effort.

Another reason for working with multi-view in mine recognition underwater is that strong water currents can directly affect the quality of the image acquired by sonars. For example, navigating perpendicularly to the direction of the current reduces image quality [35], and important features might not be detected. Under those circumstances, when assessing the completion of an MCM mission, we need to determine if the whole area of interest was explored and also the number of times that areas with a possible target in it have been scanned by the robot. In Chapter 2 we discuss more about how the coverage measure can be useful in improving the detection of features underwater.

Determining the coverage measure is an act of counting. Topological concepts have already been explored for counting [36] and for addressing coverage problems in robotics contexts, e.g. [37], [38]. In this work, we establish a relation between the topological degree and the coverage measure of points explored by the robot. And in this framework, we developed algorithms for characterizing the explored area through the computation of the topological degree.

Different works [39],[40] proposed algorithms for general topological degree computation. However, methods available in the literature will compute the topological degree of a cycle with respect to a single point, needing to be applied to each point individually for a full characterization of the environment. In this context, we present a set-membership approach that efficiently determines the topological degree for a whole area of interest. The resulting algorithm and all the concepts defined in this work are applied to determine the area explored by a real autonomous underwater vehicle and by an Unmanned Surface Vehicle (USV).

1.3.1 Outlines and Contributions

To summarize, we identify four points of contribution in this thesis:

- Introduction of the notion of coverage measure and uncertain coverage measure, for when the measurements made by the robot are noisy. In addition, we establish, and prove, a relation between the coverage measure and the concept of the topological degree;
- An algorithm that integrates the topological structure of the explored environment for characterizing the two-dimensional plane with respect to the coverage measure of its points and for determining the area explored by a mobile robot;

- An algorithm for efficiently computing the winding number of a cycle with respect to all points in \mathbb{R}^2 ;
- Experiments in the real world illustrating applications of the algorithms developed.

1.4 Outline

This thesis is organized into two main parts. The first part serves as the theoretical foundation and comprises three chapters. Chapter 2 present sonars and some sonar terminology that will be used in this document. Furthermore, it introduces some challenges inherent in sonar imagery and discuss how the theory developed in this thesis can be helpful for enhancing target detection in acoustic images. The subsequent two chapters within this first part provide the essential mathematical tools required to comprehend the thesis. Chapter 3 addresses interval analysis, and Chapter 4 topology.

The second part of the thesis is dedicated to presenting our contributions. This section encompasses three chapters. Chapter 5 focuses on formalizing concepts related to the explored area and the coverage measure. It also introduces the primary contribution of this thesis that is to establish a relation between topological tools, particularly the topological degree, and the exploration problem within a two-dimensional space. Chapter 6 presents algorithms for characterizing areas based on the coverage measure of its points and algorithms for computing the topological degree through interval analysis. Chapter 7 delves into real-world experiments that offer practical illustrations of the theory developed within the thesis.

Finally, the final chapter, Chapter 8, summarizes our findings, and discusses future research directions.

Part I

Theoretical Background

Chapter 2

Sonars

Let there be sound

Contents

2.1	Introduction	21
2.2	Side-Scan Sonar	23
2.2.1	Waterfall and Mosaic Images	26
2.2.2	Features Detection in Sonar Images	28
2.2.3	Hypotheses Adopted on the Side-Scan Sonar Model	33
2.3	Coverage Measure to Aid in the Post-Treatment Process	36
2.4	Conclusion	38

2.1 Introduction

Sonar, short for Sound Navigation and Ranging, is a vital technology employed in underwater exploration and research that works as a non-invasive mean of investigation. Water is denser and more uniform compared to air, allowing sound waves to propagate further and with less loss of energy, overall enabling sonar systems to achieve greater ranges and resolution underwater [41]. In addition, the higher speed of sound in this environment allows for shorter acoustic pulses and quicker return times, resulting in better temporal resolution and more accurate measurements. This rapid data acquisition capability is invaluable for mapping large areas [42], identifying potential hazards [43], and conducting extensive surveys in general.

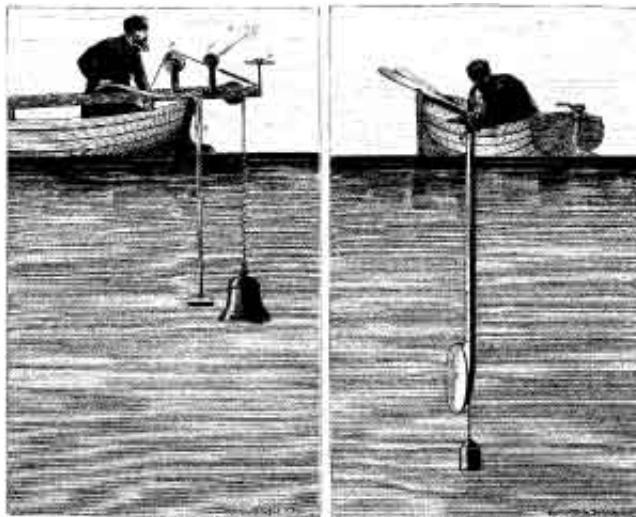


Figure 2.1: Image extracted from [44]. Colladon and Sturm’s experiment for measuring the speed of sound in 1826.

The concept of using sound waves for underwater detection can be traced back to the early 19th century. In 1826, a physicist named Jean-Daniel Colladon and the mathematician Charles-François Sturm conducted experiments to determine the speed of sound in water by using an underwater bell and measuring the time it took for the sound to reach a listening tube, Figure 2.1. This laid the foundation for the fundamental principles of underwater acoustics [44]. During World War I, the need for detecting submerged submarines and underwater obstacles became critical and technology advanced rapidly. By

World War II, sonar systems were already highly developed engineering systems and played a crucial role in detecting and tracking underwater threats, significantly influencing the outcome of naval battles [45].

Two distinct approaches to underwater acoustic sensing have been developed: passive and active sonar.

Passive sonar systems rely on listening to ambient sounds or acoustic signals generated by underwater sources. It acts as an acoustic "microphone," picking up and analyzing the existing acoustic signals in the environment. This approach offers several advantages, such as the ability to detect and track marine organisms, for example acoustics serves as the primary mode of communication for dolphins and whales, or vessels and ships without revealing the presence of the monitoring system. However, passive sonars often face challenges such as background noise, signal interpretation, and the need for sophisticated algorithms to differentiate between various acoustic sources.

In contrast, active sonar operates by emitting sound waves into the water and analyzing the returning echoes, actively probing the environment to detect and locate objects or features. It offers several benefits, including the ability to determine the underwater topography and even the size or shape of submerged objects. The main disadvantage is that it is possible to detect its presence.

In this work we focus on active sonar systems, for which there are several types that are commonly used depending on the application. Some examples are [46]:

1. Single-Beam Echo Sounder: Commonly used for measuring water depth locally and creating basic bathymetric profiles.
2. Doppler Velocity Logger (DVL): Specialized sonar system that measures the velocity of a moving platform, such as a submarine or underwater vehicle, relative to the seafloor or water column. Usually used for navigation, underwater vehicle positioning;
3. Forward-Looking Sonar (FLS): Provides imaging of the underwater environment in front of a vessel or underwater vehicle. They are utilized for navigation, obstacle avoidance, and underwater surveillance;
4. Side-Scan Sonar: Extensively used for seafloor mapping, underwater imaging, and target detection. It provides detailed imagery of the seafloor and submerged objects;
5. Multi-Beam Sonar: Frequently employed for bathymetric mapping and seafloor profiling. They offer high-resolution data and accurate depth measurements, making them essential tools for hydrographic surveys, coastal engineering, and offshore exploration.

For the exploratory context of this thesis, side-scan sonars and multi-beam sonars are the most appropriate tools. The choice between these two sensors depends on the specific objectives of the project. Multi-beam sonars were designed for ocean survey and have a narrow coverage area in shallow waters, therefore, they are better suited for deep water environments. Side-scan sonar is efficient at covering large areas quickly in shallow and intermediate water depths, where it can provide a detailed seafloor image in a shorter time interval. In addition, side-scan sonar excels at detecting and identifying individual objects or features on the seafloor. In this case, if the covering mission aims at target detection, side-scan sonar is a better option. Multi-beam sonar, on the other hand, are preferred for detailed bathymetric mapping and for obtaining accurate depth information, not offering the same level of target discrimination.

Considering the above, throughout this work, side-scan sonars will be considered as the primary research focus. Therefore, this chapter aims to provide an overview of sonar imagery principles, with a particular focus on the geometry acquisition aspect of this type of sonars. We also present an overview of the challenges faced during detection of targets on acoustic images and how this work can be useful in this context.

2.2 Side-Scan Sonar

The architecture of active sonar systems comprises three distinct modules: the emission module, hydrophone array and reception module. The emission module, also known as the transmitter, is responsible for generating the acoustic signals that are emitted into the water. It typically utilizes a transducer, which converts electrical energy into sound waves, controlling the frequency, duration, and pattern of the emitted signals and also allowing for customization based on the specific application requirements. The hydrophone array, or antenna, serves as the receiver of the acoustic signals reflected or scattered by underwater objects. It consists of multiple individual hydrophones strategically positioned to capture the incoming signals from different directions. These hydrophones convert the acoustic energy into electrical signals. Finally, the reception module, also known as the receiver or signal conditioning unit, processes the electrical signals received from the hydrophone array.

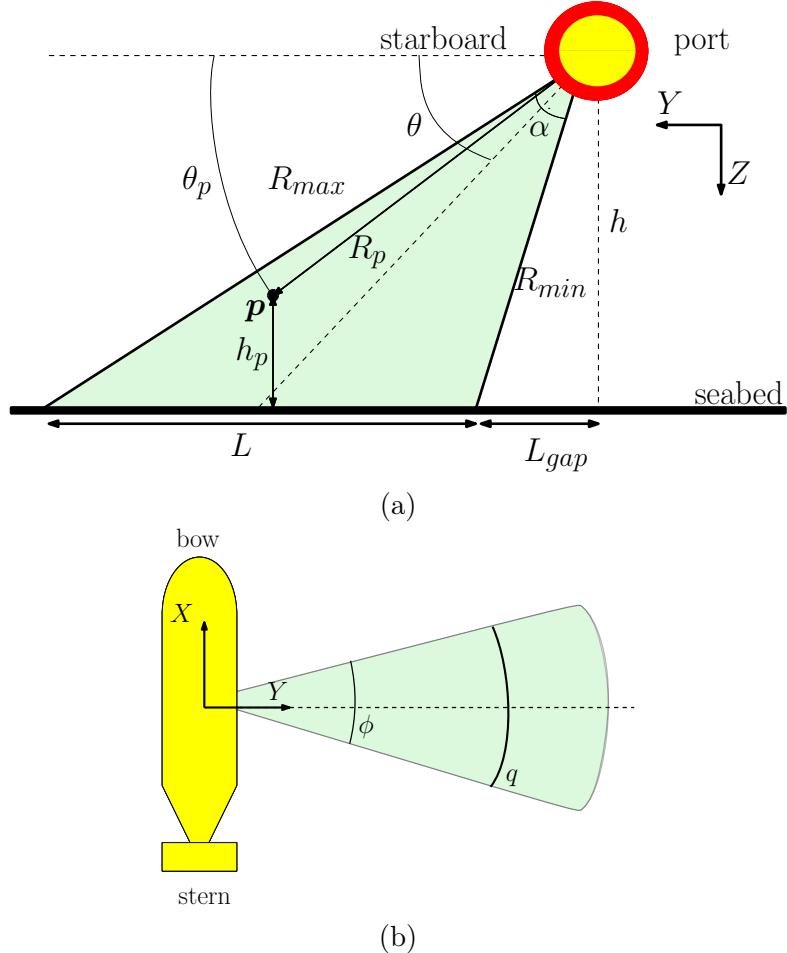


Figure 2.2: The Figure presents an AUV at an altitude h with an embedded starboard side-scan sonar. The area represented in green corresponds to the area ensonified by the sonar. Figure (a) offers a front-view representation (across-track direction) and Figure (b) a top view (along-track direction).

In Figure 2.2 a characterization of a starboard side-scan sonar is presented as proposed in [47]. The sensor is mounted on an AUV with a fixed angle θ between the axis of the sound propagation and a horizontal reference axis. The sonar's openings in the YZ and XY planes are, respectively, represented by α and ϕ . These two angles depend directly on the speed of sound on these two different planes. In practice, vertical openings α assume large values while horizontal openings ϕ are small and usually irrelevant, as it is explained later in this chapter on Section 2.2.3.

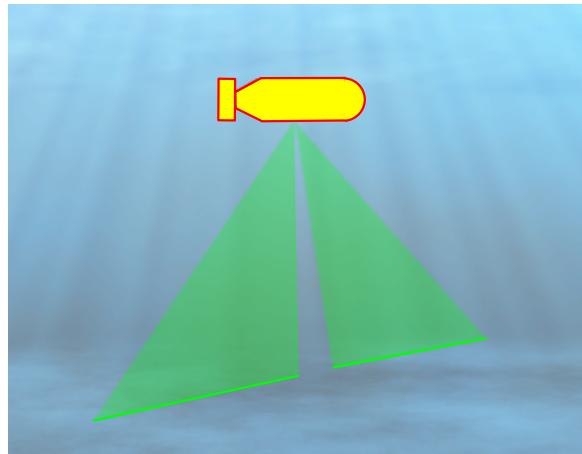


Figure 2.3: AUV navigating with two side-scan sonars symmetrically placed on port and starboard. A visibility gap under the robot, as illustrated, is common in these systems.

As illustrated in 2.2a, in side-scan sonars systems, the emission module emits a series of narrow acoustic beams in a slant direction towards the seabed and perpendicular to the direction of motion of its carrier. Each emitted beam captures information along a specific track or line and together, the emitted beams can be conceptualized as a line sweeping across the environment due to the narrow horizontal aperture ϕ . Typically, the system consists of a pair of antennas positioned on port and starboard of the towing vehicle, Figure 2.3. As the sonar system moves inside the environment, at every ping emission, a narrow stripe of the seabed is ensonified, resulting in a sequence of lines that, if concatenated, effectively cover the entire survey area.

Frequently, data received at the starboard and the port antenna are treated simultaneously, and data acquired from side-scan sonars are typically represented as a 2D image, named waterfall, capturing both port and starboard views.

2.2.1 Waterfall and Mosaic Images

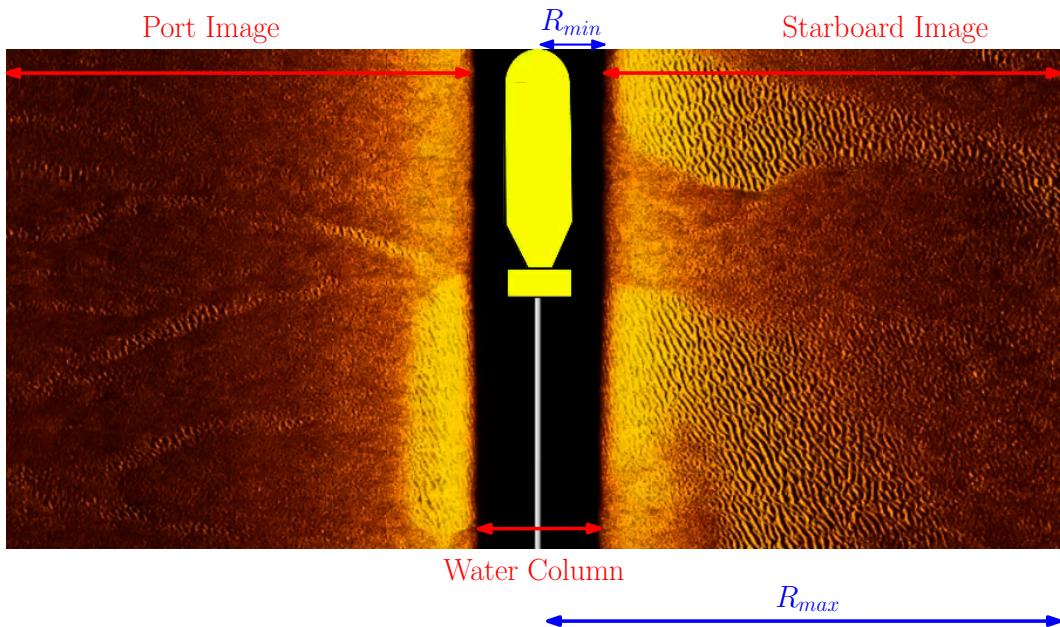


Figure 2.4: Waterfall image created by a two side-scan sonar system mounted on AUV with the configuration presented on Figure 2.3. The black band in the middle represents the visibility gap, the region under the robot from where we have transmission loss. R_{min} and R_{max} correspond, respectively, to the minimal and maximal slant range in the robot's visible area as represented in Figure 2.2a.

Each row in the waterfall image corresponds to a ping emission and essentially, with each new ping emission, a new row is added to the image. Therefore, as new echoes returned from the seafloor and objects arrive, the current image shifts downward, creating space for the new data on the top. For this aspect they are called waterfall images, Figure 2.4.

The vertical axis of the image represents the time at which the sonar beam or ping was emitted, in this case, each row in the image represents a specific ping emission. The horizontal axis represents the slant range of the sonar pulse, that represents the actual distance traveled by the acoustic signal during the time-of-flight of the sonar pulse in the across-track direction before being reflected. In other words, the slant range is half the distance that the pulse traveled during the time that the signal took from being emitted, backscattered and received again by the sonar's antenna. It is perpendicular to the direction of motion or the sonar platform's trajectory. In this context, each column in

the image represents the received signal at a specific time interval, providing information about the spatial distribution of the sonar data across the swath or coverage area. The different color intensities represent the amplitude of the backscattered signal, creating an understanding about the type of surface that reflected it (sand, gravel, mud, etc.) or even, in the case of low amplitude, if an area was not ensonified at all, indicating the presence of an obstacle obstructing the passage of the sound waves.

The waterfall image represents the explored area in the sonar's relative coordinates. Georeferencing the sonar image is called mosaicking, the focus is on associating the relative positions within the sonar images to their respective longitude and latitude coordinates. This process allows for the integration of the sonar data with other geospatial information, enabling accurate positioning and spatial referencing of the features represented in the waterfall images.

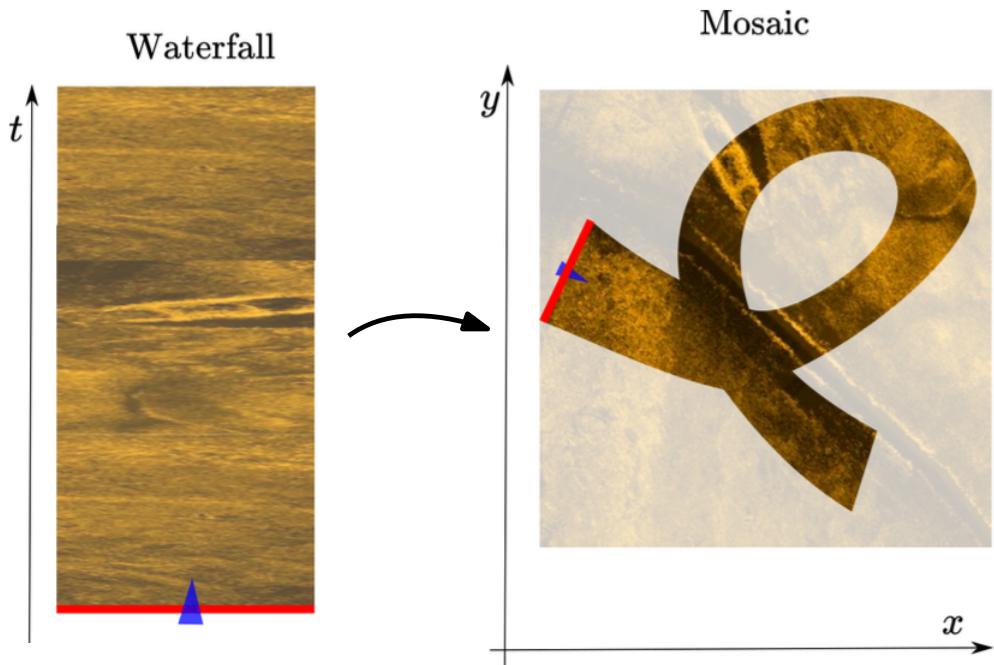


Figure 2.5: Example of mosaicking of a Waterfall image into Cartesian coordinates. For illustration purposes the visibility gap, or water column, is not represented.

Mosaicking can also be a transformation of the waterfall relative coordinates into any absolute coordinate system such as Cartesian, Figure 2.5.

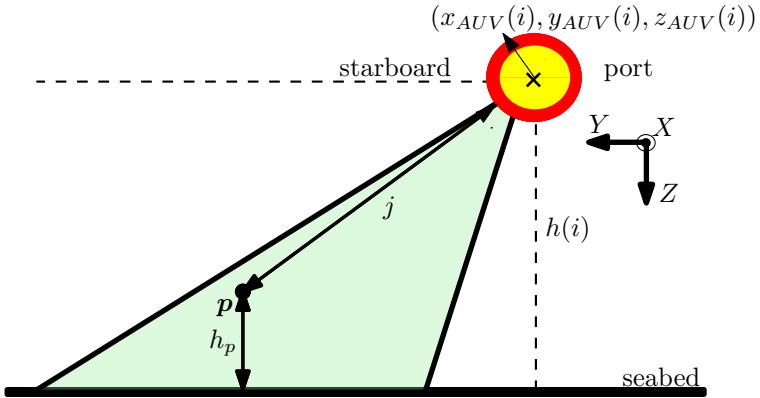


Figure 2.6: The i index represents the time in which the ping was emitted and j is the distance traveled by a sound sample to be backscattered by a point p across-track. The vector $(x_{AUV}(i), y_{AUV}(i), z_{AUV}(i))$ represents the AUV's position in a global reference frame.

The transformation in this example can be computed using a rotation matrix R_m associated to the robot's orientation angles (raw, pitch and yaw) and its position $(x_{AUV}, y_{AUV}, z_{AUV})$:

$$\begin{pmatrix} x_{ij} \\ y_{ij} \\ z_{ij} \end{pmatrix} = R_m(i) \begin{pmatrix} 0 \\ \pm\sqrt{j^2 - (h(i) - h_p)^2} \\ h(i) - h_p \end{pmatrix} + \begin{pmatrix} x_{AUV}(i) \\ y_{AUV}(i) \\ z_{AUV}(i) \end{pmatrix}$$

where the i index represents the specific ping emitted (vertical axis on the waterfall image), while the j index represents the individual samples across the track (horizontal axis on the waterfall image), Figure 2.6. The robot's altitude h can be estimated by a Doppler Velocity Logger (DVL) or by a single-beam echo sounder. The height of the feature responsible for reflecting the signal at range j is represented by h_p , we discuss more about estimating this parameter later in the chapter. To indicate which sensor is being considered, the sign \pm is used, where $+$ indicates the starboard side and $-$ denotes the port side. Georeferenced sonar images are called mosaic images.

Now, we describe the process of feature detection on the waterfall image.

2.2.2 Features Detection in Sonar Images

Processing waterfall images and detecting features within them is out of the scope of this work. However, a general understanding about this process is important for recognizing its complexity and challenges.

A sonar system detects underwater features through remote sensing. This technique is applied when the feature cannot be directly inspected, and the

information about it is to be gained secondarily, without direct physical contact.

The receiver of a sonar system encounters various noises that need to be addressed before performing feature detection on the acoustic image [48]. One common type of noise is ambient noise, which includes background sounds generated by natural sources such as waves, marine life, and geological activities. Another source of noise is reverberation, which occurs when sonar signals bounce off objects and surfaces in the underwater environment, creating unwanted echoes. These echoes from the seafloor can overlap and mask the weaker signals from underwater targets, reducing detection range and signal clarity. Additionally, electronic noise from the sonar equipment itself, such as thermal noise and electrical interference, can further degrade the received signals. To address these challenges, the receiver employs various signal processing techniques to suppress noise and enhance the desired signals.

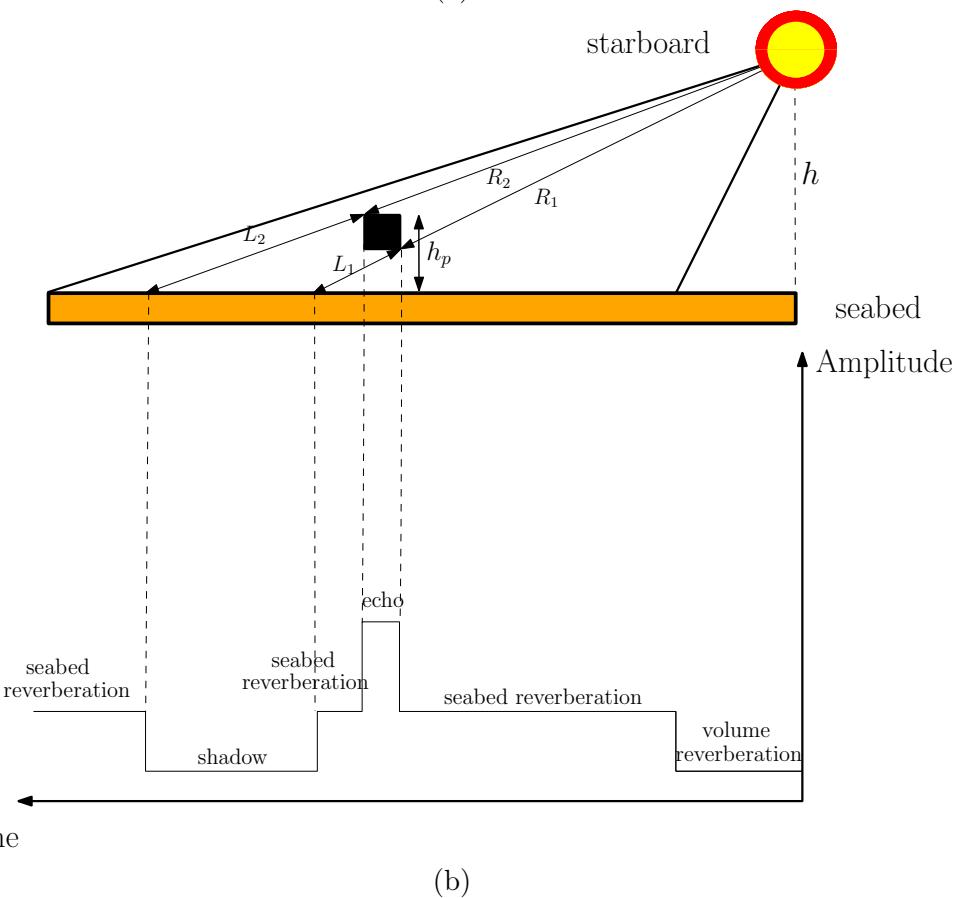
The quality and detail of the generated imagery will depend on the capacity of the receiver to treat these noises but also on two parameters that are intrinsic to the sonar: the azimuth and range resolutions. The first refers to the ability of a side-scan sonar to distinguish between objects or features that are located close to each other along the horizontal plane (plane XY in Figure 2.2b). The azimuth resolution depends directly on the distance to the sonar. This means that the closer that two objects are to the sonar, the smaller the distance between them will have to be, so they can indeed be classified as two different individuals. Generally, higher-frequency signals and larger hydrophone arrays result in improved azimuth resolution [49].

Range resolution, on the other hand, corresponds to the minimal distance along the propagation direction (plane YZ in Figure 2.2a) between two objects to see them separately. A higher range resolution allows for the detection of smaller objects or features at different depths, and it is primarily determined by the bandwidth of the emitted acoustic signals [49].

These resolutions are important parameters because they determine a threshold for the minimal size an underwater feature must have for being reliably detected. We also have to add to that the acoustic properties of the feature, such as its composition and reflectivity. Features that have strong acoustic reflectivity are more likely to be detected compared to features with weak reflection capability. But in general, if a feature is smaller than the azimuth or the range resolution of the sonar system, it may not be adequately resolved and could appear as a blurred or indistinguishable part of the surrounding background. Otherwise, an acoustic shadow will indicate the presence of an object on the seafloor.



(a)



(b)

Figure 2.7: A shadow on the waterfall image represents an obstacle (a black box on the scheme) on the sonar's visibility range. (a) waterfall image, extracted from [50], acquired by a side-scan sonar where a fish is detected; (b) example of amplitude signal received by the sonar's reception module over time for a waterfall image like the one in (a).

While the direct echo in sonar imagery provides valuable information, shad-

ows offer additional details. For example, the distortion or elongation of the shadow can reveal object's dimensions and geometry that may not be apparent from the direct echo. This is well illustrated on Figure 2.7a, where the fish's shadow on the waterfall has more details than its own echo.

In Figure 2.7b one may see an example of what the received signal for Figure 2.7a can look like. Different amplitudes of the signal are represented [51]:

- Low amplitude volume reverberation that refers to the scattering of sound waves by particles in the water column or seabed. Volume reverberation can be a beneficial or challenging aspect in sonar imagery depending on the application. On the positive side, it can provide valuable information about the environment, such as the density and distribution of particles in the water column. This information can be used for tasks like water column characterization, biomass estimation, or monitoring sediment transport [52]. The latter being important, for example, for understanding the effects of tsunamis on the underwater environment [53]. On the negative side, it can introduce background noise and reduce the signal-to-noise ratio, making it more difficult to detect and distinguish individual targets or objects of interest;
- Mean amplitude due to seabed reverberation that refers to the scattering and reflection of sound waves by the irregularities and composition of the seabed. Different bottom types, such as sand, mud, gravel, or rocks, have distinct scattering properties, resulting in variations in the reverberation signal. It can also contribute to reduce the signal-to-noise ratio;
- High amplitude revealing an echo after interacting with an object in the environment;
- Low amplitude due to the shadow area of the object, it corresponds to a part of the space that is not explored by the sonar.

An approximation of the object's height h_p and vertical size can be inferred from L_1 , L_2 , R_1 , R_2 . These values can all be approximated from direct measurements on the waterfall image. For example, the width of the shadow on the waterfall, in theory, is equal to $L_2 - L_1$. Of course, in practice, it may not provide highly precise results due to several factors and limitations. For example, waterfall images have a pixel resolution and that might not provide the level of detail necessary to precisely distinguish and measure the shadow's boundaries. Environmental conditions, including water turbidity, sediment,

and suspension can also affect the shape of a shadow as beam spreading, signal attenuation, and noise. For all these reasons, estimating object's height only based on sonar data is a challenging task that has been the subject of different works [54].

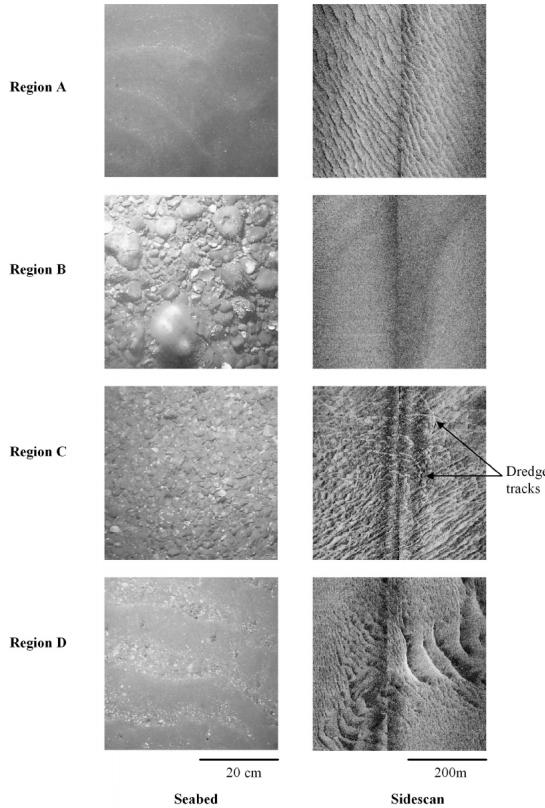


Figure 2.8: Image extracted from [55]. Examples of regions swept by a side-scan sonar (left column) and the resultant acoustic image for each of them on the right column. The left column corresponds to images acquired by a camera in an underwater video.

The analysis of this backscattered signal indicates the presence of an object, a fish to be precise, but many shaped or shapeless features can also be seen by the sonar. In Figure 2.8, extracted from [55], different textures on acoustic images (right column) are presented in function of the seabed composition. Textures can be characterized by varying intensities and repetitive patterns. These variations are directly related to the physical characteristics of the seabed, which can differ based on meteorological conditions, seasons, and geographical locations. Region B and C represent gravel areas. Even though

particular textures appear on the correspondent sonar images, revealing a certain type of seabed, it is difficult to give a definition. Region B in particular presents a very noisy sonar image, possibly due to different size and irregularly shaped gravels that cause scattering and reflection of sound waves in various directions. In regions A and D sandy areas are more easily observed, region D being slightly gravelly.

2.2.3 Hypotheses Adopted on the Side-Scan Sonar Model

We discuss here the hypotheses adopted on the side-scan sonar model and the assumptions made regarding its visible area, which serve to simplify the mathematical model used throughout the rest of this work. We aim to strike a balance between model simplicity and practical applicability.

First, we are going to assume that the speed of propagation of the sound in the water is constant during a mission, not taking into account possible changes due to water salinity, temperature, etc. As a consequence, we have that angles α and ϕ , illustrated in Figure 2.2, are also constant. We also make assumptions about the visibility gap between starboard and port sonars, the local bathymetry of the environment and about the horizontal expansion of the sound. These hypotheses are discussed in the remaining of this section.

No Visibility Gap

We assume that the visibility gap, represented by L_{gap} in the model presented on Figure 2.2a, is zero. We also consider that there is no overlap between the visible areas of the starboard and port sonars. In this case, both visible areas can be represented as a single connected set in the environment.

In practice, a configuration like this is hardly feasible, it would require a careful positioning and alignment of the sonar systems and appropriate sonar beam patterns and coverage angles. Even if this could be managed, this ideal configuration would change for different altitude values of the system carrier. However, it is important to notice that this assumption does not impose a limitation on the method proposed in this work, it only simplifies its mathematical presentation. Our method will still be able to take into account overlaps and gaps on the ensonified area.

The Flat Sea-Floor Assumption

The position of a point $p \in \mathbb{R}^3$ that is visible by the sonar in the environment is usually represented by polar coordinates (R_p, θ_p) , as presented in [47], along with its height h_p , Figure 2.2a, where R_p corresponds to the slant range of

the point, or its distance to the sonar, and the angle θ_p might be obtained as follows:

$$\theta_p = \arcsin\left(\frac{h - h_p}{R_p}\right)$$

As discussed earlier in this chapter, estimating the object's height h_p solely from sonar data might be a hard fallible task. But if h_p is unknown then θ_p can not be computed. In this context, the only knowledge about an ensonified point p will be that it is responsible for an echo at R_p but, it might be anywhere inside the angular interval $[\theta - \frac{\alpha}{2}, \theta + \frac{\alpha}{2}]$. A common approach, considering the above, is to assume that the floor is flat [56], [57], where it is assumed that all detectable features within the ensonified region have $h_p = 0$.

In addition, as exemplified in Figure 2.7, an object or any underwater feature with a certain height conceals a part of the seabed from the sonar. Therefore, the projection of the sonar's visible area on the sea-floor can no longer be represented by a fully connected region if the flat-floor assumption is not assumed. As a matter of fact, we would have to represent the ensonified region as a set of non-intersecting regions and without bathymetric data about the environment this would be impossible before treating the waterfall image.

For presentation purposes, we assume that this condition is met at least locally. As it can be expected, the flat sea-floor assumption leads to errors when estimating the position of features in the environment and, as a consequence, in the process of building a map from sonar data. There are many works in the literature dedicated to mitigate the effects of this assumption [58],[47]. Nevertheless, the flat sea-floor is still a reasonable approach in many contexts specially if we consider that the sensor's altitude h will be usually significantly bigger than most features height, making the effects of the assumption neglectable.

We insist on the fact that if one wants to incorporate external bathymetry to account for areas that may be hidden due to a non-flat sea-floor, our method can still be effectively applied by treating each one of the unconnected sets that form the visible area individually.

No Horizontal Opening - The Zero Azimuth Assumption

The angle ϕ considered in this model, Figure 2.2, represents the sensor's opening and the beam spreading effects on the horizontal plane. This angle is usually called the sonar's azimuth angle in the literature. This opening creates uncertainty on the positioning of a point in the sonar's range of visibility, not being possible to exactly determine its positioning on the XY plane. For example, consider a feature in the environment that can be represented by $(R_p, \theta_p, 0)$, assuming the flat sea-floor condition. On the XY plane this point

can be found anywhere within an arc q , see Figure 2.2b, and we can not be more precise than that.

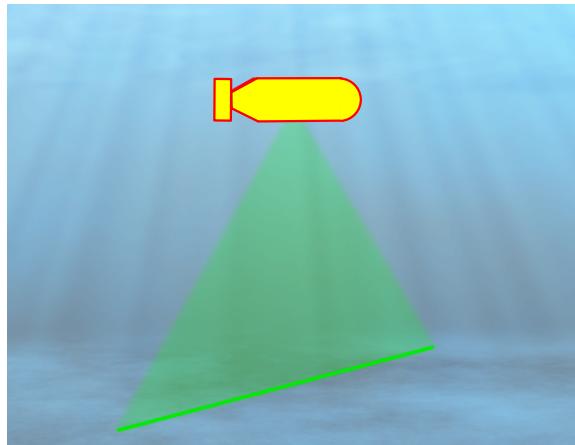


Figure 2.9: The robot's visible area considering the conditions established for the modeling of the two side-scan sonars system used in this thesis.

The angle ϕ is usually very small and often considered to be zero, resulting in no horizontal opening. In a first moment, we make this assumption for simplifying the presentation of the model. However, later in this work, we extend our methodology to accommodate more general scenarios.

Here, we consider the sonar beam to be perfectly perpendicular to the motion direction, resulting in a narrow and focused beam. In practice this assumption is reasonable when dealing with Synthetic-aperture sonars (SAS), for example. These sensors use a post-processing technique named beamforming [59] that introduces constructive interference on signals from hydrophones with particular angles and destructive interference on others, improving along-track resolution [60].

By taking $\phi = 0$, and considering all the other hypotheses established in this chapter, we will be dealing with a visible area for a starboard/port side-scan sonar system that can be represented as illustrated in Figure 2.9.

This condition is necessary for line sweep exploration. If the zero azimuth condition is not met, the visible area becomes two-dimensional, posing a different challenge. Later in this manuscript, we present how our method can be properly extended to address this context.

2.3 Coverage Measure to Aid in the Post-Treatment Process

Detection and classification of features underwater have numerous applications across different domains. It is essential for marine exploration and scientific research, for example, by mapping the seafloor, identifying geological formations, locating underwater ecosystems, and studying marine organisms and their habitats. It also enables the discovery and preservation of ancient shipwrecks, submerged cities, historical artifacts, providing valuable insights into maritime history and cultural preservation. Some other applications might present a critical character, for example, search and rescue operations, mine-countermeasure missions, and detecting and classifying other vessels and underwater vehicles in the environment. Accurate classification is essential when distinguishing between friendly and hostile targets in order to support effective decision-making and tactical responses. However, detection and classification are two different tasks that present individually many challenges due to several factors inherent to the undersea environment and the nature of sonar technology itself.

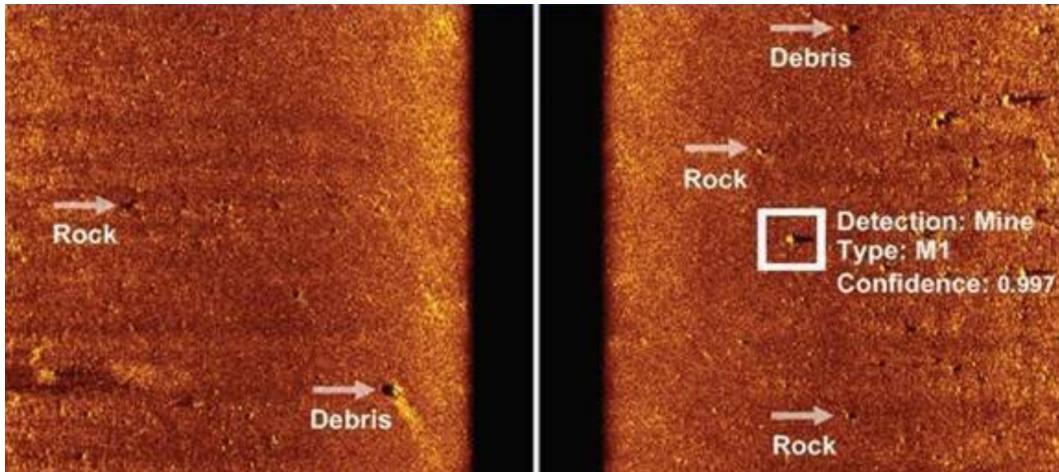


Figure 2.10: Waterfall image with some debris and an underwater mine detected. Image extracted from [61].

One of the primary reasons for the difficulty in classifying objects in sonar images is the limited information available in the data. As presented previously, waterfall images typically provide grayscale or color intensity representations of the received signal strength, which may not contain sufficient discriminative features to distinguish between different object classes. Additionally, the acoustic properties of underwater objects can vary significantly.

2.3. COVERAGE MEASURE TO AID IN THE POST-TREATMENT PROCESS37

Different objects may have similar acoustic signatures or exhibit variations in their response due to factors such as shape, orientation, material composition, and environmental conditions. These variations can lead to ambiguities and overlapping characteristics, making it difficult to establish clear boundaries between different classes. For example, in Figure 2.10 we present a waterfall image with some debris and an underwater mine detected. From this image, one can understand the difficulty in classifying a potential hazard as so, since it resembles a lot to an ordinary feature. A correct classification in this scenario, however, is essential for protecting human lives.

The study of methods for automatic target detection and classification in sonar images have, of course, been considered by many works, applying classical signal processing [62] and machine learning [63]. Nonetheless, underwater environments are highly dynamic and diverse, with a wide range of underwater objects, such as rocks, wrecks, marine organisms, and geological formations. Objects may exhibit variations in size, shape, and appearance, further adding to the complexity of classification. All of that contributes to the scarcity of labeled training data for specific underwater objects or classes [64], [65]. In addition, the creation of a proper dataset is also affected by reverberation, and interference in sonar signals since these unwanted signals can distort the acoustic signatures of objects. In view of the above, poor robustness and a low identification rate are common outputs of these researches [66].

Due to all the above, when target classification presents a vital importance we still have human classifiers looking through hours of waterfall images. Trained experts can indeed identify and classify targets [67], but this approach is time-consuming, resource-intensive, and expensive. In addition, as described in [68], it can be very monotonous, in the end, also leading to a frequent misclassification of targets. A prior characterization of waterfall images according to the coverage measure of its points would allow the user to know beforehand how many times each point should appear on the sequence, facilitating its analysis and matching between different points of view of the same feature. Overall, the coverage measure of points on the waterfall allows for a more informed interpretation, enhancing the decision-making process and reducing the likelihood of misinterpretation or false conclusions.

The coverage measure of sonar data can also be useful information for assessing the creation of a dataset for machine learning algorithms. By incorporating information about the amount of times a certain region has been explored to create a dataset, models can have a better judgement about their own performance. Areas that have been extensively explored will have models that can learn better to distinguish characteristics and patterns of features in those areas. In general, analyzing the distribution and density of explo-

ration efforts across different areas can provide insights into the likelihood of encountering certain features. Understanding where previous detections and classifications have occurred helps in establishing baseline expectations and identifying regions with higher or lower probabilities of specific features. This knowledge might help to set appropriate detection thresholds and adapt classification algorithms accordingly.

2.4 Conclusion

Sonars have revolutionized underwater exploration, marine research, and various applications across different domains. In this chapter an overview about sonar technology and more particular side-scan sonars was presented. We have explored the principles and techniques involved in side-scan sonar imaging, including the acquisition of waterfall images and the subsequent mosaicking process.

Hypotheses assumed about the sonar's mathematical model were also presented. Our method can still, however, be applied to more general scenarios, since the hypotheses presented only aim at simplifying its representation for presentation purposes.

Finally, we established a relation between the work presented on this thesis and the challenges encountered during the classification of underwater targets from sonar images.

Chapter 3

Interval Analysis

"Uncertainty is the only certainty there is, and knowing how to live with insecurity is the only security." - John Allen Paulos

Contents

3.1	Introduction	40
3.2	Set Theory	41
3.2.1	Operations on Sets	41
3.2.2	Set Image	43
3.3	Interval Analysis	43
3.3.1	Intervals	43
3.3.2	Boxes or Interval Vectors	46
3.3.3	Inclusion Functions	47
3.4	Set-Valued Constraint Satisfaction Problems (SVCSPs)	49
3.4.1	Definition of a SVCSP	50
3.4.2	Contractors	50
3.4.3	Separators	53
3.4.4	Image Contractor	56
3.5	Subpaving	61
3.5.1	Set Inversion Via Interval Analysis (SIVIA) Algorithm . .	62
3.5.2	Paver	64
3.6	Dedicated Sets	69
3.6.1	Interval of Sets or Thick Sets	69
3.6.2	Interval of Functions or Tubes	70
3.7	Detection of Self-Intersections on Tubes	73
3.8	Conclusion	77

3.1 Introduction

There are numerous information and events in the world that we might desire to measure or monitor, however, the world is noisy and unpredictable. When we use a scale, for example, or any other sensor for the matter, we know that it is not capable of giving us a precise value, and we also know that if we weight the same object twice, it will probably give us different readings. Nevertheless, we continue to use the scale because we assume that the unknown actual solution is probably not that far from the noisy reading, we accept a certain margin of incertitude. In brief, uncertainty is an inherent feature of the world we live in, and we are used to dealing with the uncertain by wrapping it under a set of possibilities where the real answer is known to lie within.

Indeed, enclosing the unknown into a guaranteed solution set is the motivation behind a huge body of work. This is particularly important in applications where errors and uncertainties in the input data can propagate through the computation, leading to significant deviations in the output.

In the fields of robotics and control theory, probabilistic approaches are popular for assessing the best solution to a problem considering some knowledge about its uncertainties. These approaches, however, imply a probability distribution of the problem that is usually unverifiable. Our choice in this work is to address uncertainty through set-membership approaches. In this case, we will be dealing with a set of possible solutions to a problem, in contrast to traditional methods that rely on single-point values.

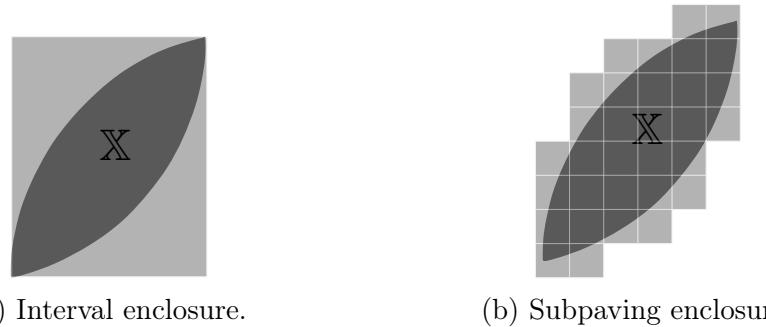


Figure 3.1: The set \mathbb{X} is represented by outer approximations.

A set might assume any shape and even be disconnected, in this case, in order to be represented and manipulated by a computer, sets have to be represented by outer approximations. With this purpose, different techniques can be found in the literature, such as zonotopes or polyhedral enclosures [69, 70], ellipsoids [71], intervals or subpavings [72]. An example of an outer approximation obtained using intervals and subpaving is illustrated in Figure 3.1. These

two methods have been proven to be robust to non-linear constraints [73], an important attribute considering the characteristics of problems in mobile robotics. Moreover, interval analysis provides a rigorous structure for bounding and quantifying uncertainties, allowing guaranteed and robust computations. In this context, we adopt the interval approach.

Interval computation is a particular case of set computation. Therefore, we first recall some fundamental concepts from set theory that lay the foundations for interval analysis. Readers who are comfortable with this content can feel free to go directly to Section 3.3 from where we present the interval analysis tools used in the following chapters.

3.2 Set Theory

A set is a collection of distinct objects, which can be anything from numbers and symbols to words, colors, or even other sets. The objects in a set are called its elements or members, and there is no repetition of elements within a set.

Let \mathbb{A} be a set. Set theory is based on a binary relation that says that any element a either belongs to \mathbb{A} , $a \in \mathbb{A}$, or not, $a \notin \mathbb{A}$. There are two important sets that must be considered: the empty set \emptyset , that contains no element and the universe Ω , that contains all elements. The complement of \mathbb{A} is denoted $\overline{\mathbb{A}}$ and by definition:

$$\overline{\mathbb{A}} = \{a \in \Omega | a \notin \mathbb{A}\}$$

3.2.1 Operations on Sets

Let \mathbb{A} and \mathbb{B} be sets, the following basic operations can be applied:

Intersection $\mathbb{A} \cap \mathbb{B} \triangleq \{a | a \in \mathbb{A} \wedge a \in \mathbb{B}\}.$

Union $\mathbb{A} \cup \mathbb{B} \triangleq \{a | a \in \mathbb{A} \vee a \in \mathbb{B}\}.$

Difference $\mathbb{A} \setminus \mathbb{B} \triangleq \{a | a \in \mathbb{A} \wedge a \notin \mathbb{B}\}.$

Inclusion $\mathbb{A} \subset \mathbb{B} \Leftrightarrow \forall a \in \mathbb{A}, a \in \mathbb{B}.$

Equality $\mathbb{A} = \mathbb{B} \Leftrightarrow \mathbb{A} \subset \mathbb{B} \text{ and } \mathbb{B} \subset \mathbb{A}.$

Cartesian Product $\mathbb{A} \times \mathbb{B} \triangleq \{(a, b) | a \in \mathbb{A} \wedge b \in \mathbb{B}\}$

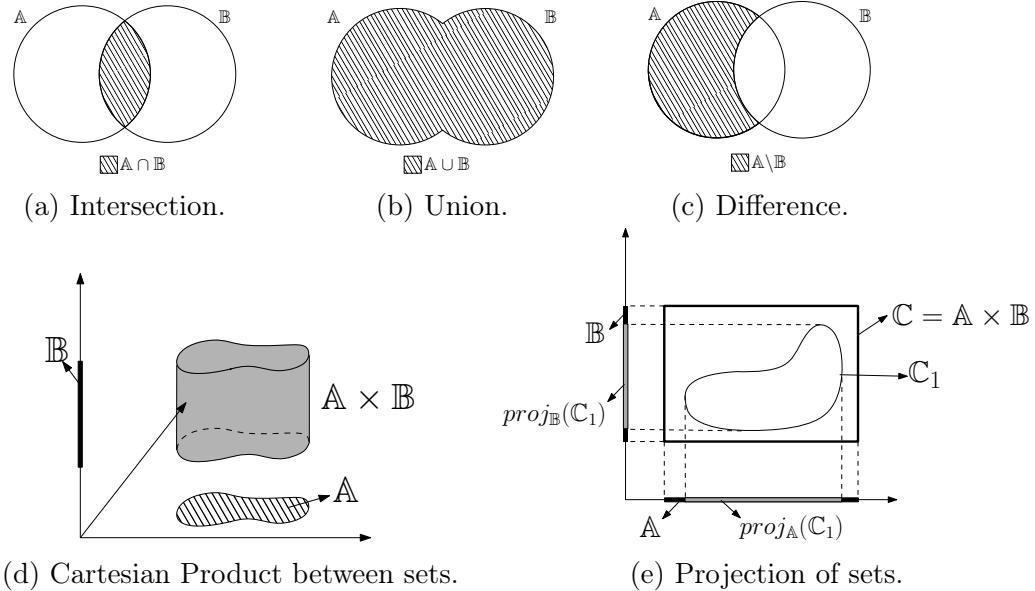


Figure 3.2: Basic set operations.

Projection Let $C = \mathbb{A} \times \mathbb{B}$ and C_1 be a subset of C . Two projections can be defined:

The projection of C_1 onto \mathbb{A} (with respect to \mathbb{B}):

$$\text{proj}_{\mathbb{A}}(C_1) \triangleq \{a \in \mathbb{A} \mid \exists b \in \mathbb{B} \text{ such that } (a, b) \in C_1\}$$

and the projection of C_1 onto \mathbb{B} (with respect to \mathbb{A}):

$$\text{proj}_{\mathbb{B}}(C_1) \triangleq \{b \in \mathbb{B} \mid \exists a \in \mathbb{A} \text{ such that } (a, b) \in C_1\}$$

We illustrate some of these operations in Figure 3.2. Some useful examples are:

- $\mathbb{A} \cup \emptyset = \mathbb{A}$,
- $\mathbb{A} \cap \emptyset = \emptyset$,
- $\mathbb{A} \cup \Omega = \Omega$,
- $\mathbb{A} \cap \Omega = \mathbb{A}$.

3.2.2 Set Image

We can extend operations on numbers (or vectors) to operations on sets. For example, consider the sets \mathbb{A} and \mathbb{B} and a function $\mathbf{f} : \mathbb{A} \subset \mathbb{R}^n \rightarrow \mathbb{B} \subset \mathbb{R}^m$. If $\mathbb{A}_1 \subset \mathbb{A}$,

$$\mathbf{f}(\mathbb{A}_1) \triangleq \{\mathbf{f}(\mathbf{a}) \mid \mathbf{a} \in \mathbb{A}_1\} \quad (3.1)$$

And if $\mathbb{B}_1 \subset \mathbb{B}$,

$$\mathbf{f}^{-1}(\mathbb{B}_1) \triangleq \{\mathbf{a} \in \mathbb{A} \mid \mathbf{f}(\mathbf{a}) \in \mathbb{B}_1\}$$

Interval analysis has been proved to be effective when dealing with the classical Set Inversion Problem [72], even if \mathbf{f} is non-linear. The method is presented in Section 3.5.1.

The simple concepts from set theory presented in this section provide a foundation for interval analysis. Going further, interval analysis introduces tools that make it possible to compute and represent sets that are defined by constraints. These tools are now presented in the following sections.

3.3 Interval Analysis

The main idea behind interval analysis is to represent real numbers as intervals, defined by their lower and upper bounds, rather than single point values. Over the years, interval analysis has been refined and expanded, finding applications in various fields such as optimization [74], numerical methods [75], and robotics [76].

In this section, we present the definitions and concepts that constitute the foundations of interval analysis as stated in [77].

3.3.1 Intervals

We denote $[x]$ an interval, and it represents a closed and connected subset of \mathbb{R} . It is defined by a lower bound x^- and an upper bound x^+ . The interval encompasses all real numbers between x^- and x^+ , including x^- and x^+ themselves such that

$$[x] = [x^-, x^+] = \{x \in \mathbb{R} \mid x^- \leq x \leq x^+\}$$

The width $w([x])$ of this interval is equal to $x^+ - x^-$. If $w([x]) = 0$, it means that $x^- = x^+$ and that it contains only one element, in this case, being called a degenerate interval or a singleton.

Basic Operations Between Intervals

Basic mathematical operations, such as addition, subtraction, multiplication, and division, can be defined between intervals. Let \diamond be an operator representing one of the elementary operations mentioned above, $\diamond \in \{+, -, ., /\}$ and $[x]$ and $[y]$ are intervals of \mathbb{R} . We have that

$$[x] \diamond [y] = [\{x \diamond y \mid x \in [x] \text{ and } y \in [y]\}] \quad (3.2)$$

where the brackets $[.]$ around the set denote its convex hull. It represents the smallest interval containing all feasible values for the operation $x \diamond y$, for all $x \in [x]$ and $y \in [y]$ [78].

The addition of two intervals, $x = [x^-, x^+]$ and $y = [y^-, y^+]$ is given by $[x^- + y^-, x^+ + y^+]$. This means that the lower bound of the resulting interval is obtained by adding the lower bounds of the input intervals, and the upper bound is obtained by adding the upper bounds.

Example:

$$\begin{aligned} [-1, 5] + [2, 3] &= [1, 8] \\ [-1, 5] - [2, 3] &= [-1, 5] + [-3, -2] = [-4, 3] \end{aligned}$$

Multiplication and division of intervals are more complex due to the potential for overlap and division by zero. The resulting interval is determined using all possible combinations of values resulting from the multiplication or division of the input intervals. The convex hull of these combinations represents the resulting interval. For example, the multiplication of two intervals, $x = [x^-, x^+]$ and $y = [y^-, y^+]$, gives the resulting interval $[\min(x^- y^-, x^- y^+, x^+ y^-, x^+ y^+), \max(x^- y^-, x^- y^+, x^+ y^-, x^+ y^+)]$. Similarly, the division of these two intervals, where the divisor interval does not include zero, results in $[\min(x^- / y^-, x^- / y^+, x^+ / y^-, x^+ / y^+), \max(x^- / y^-, x^- / y^+, x^+ / y^-, x^+ / y^+)]$. If $[y]$, the divisor, includes zero, then we have:

$$1/[y] = \begin{cases} \emptyset & \text{If } [y] = [0, 0], \\ [1/y^+, +\infty] & \text{If } y^- = 0 \text{ and } y^+ > 0, \\]-\infty, 1/y^-] & \text{If } y^- < 0 \text{ and } y^+ = 0, \\]-\infty, +\infty[& \text{If } y^- < 0 \text{ and } y^+ > 0 \end{cases}$$

Example:

$$\begin{aligned} [-2, 5].[2, 4] &= [-8, 20] \\ [-3, 6]/[2, 3] &= [-1.5, 3] \end{aligned}$$

General Operations on Intervals

Intervals are sets of \mathbb{R} , therefore, as it was done for sets, general operations f on numbers can also be extended to deal with intervals on the input. One should note, however, that not all operations on intervals will yield an interval on the output. This will be particularly the case if f is discontinuous. For example, let us take $f : \mathbb{R} \rightarrow \mathbb{R}$ such that

$$f(x) = \begin{cases} 1 & \text{if } x \leq 1, \\ 2 & \text{otherwise} \end{cases}$$

By definition, as stated on Equation 3.1, if f is applied to interval $[-1, 1]$, the output set $f([-1, 1]) = \{1, 2\}$ is not an interval. In order to assure an interval on the output, we will be dealing with the convex hull $[f]$ of f , and for the previous example we obtain $[f]([-1, 1]) = [1, 2]$, actually enclosing all the range of possible values. The convex hull of a function over a set is an inclusion function. We discuss more about it later in the general context of operations on interval vectors.

A general rule for evaluating a function f over an interval is to evaluate directly on its bounds if f is monotonic. Otherwise, a more detailed evaluation should be done [79].

Monotonic Example: A monotonic example is $f(x) = \log(x)$, here $[f]([x])$ can be directly evaluated : $[f]([x]) = [\log(x^-), \log(x^+)]$.

Non-Monotonic Example: Now let us take $f(x) = x^2$. If $[f]([x])$ is directly evaluated from the bounds of an input interval $[x] = [-2, 2]$, we obtain an output interval equal to the singleton $[4, 4]$. This result is different from the real solution that is $f([x]) = [0, 4]$. In order to obtain $[f]([x]) = [0, 4]$ we must consider the following rule:

$$[x^-, x^+]^2 = \begin{cases} [0, \max(x^{-2}, x^{+2})] & \text{If } x^- \leq 0 \leq x^+, \\ [\min(x^{-2}, x^{+2}), \max(x^{-2}, x^{+2})] & \text{otherwise} \end{cases}$$

Set Operations on Intervals

All set operations can also be naturally applied to intervals. For example, the intersection of two intervals is also an interval, representing the common region shared by both intervals :

$$[x] \cap [y] = \{a \in \mathbb{R} | a \in [x] \text{ and } a \in [y]\}$$

On the other hand, the union of two intervals may not always be an interval. It combines all values present in either interval. However, there are cases where the resulting set is not a connected range of values and hence does not satisfy the criteria to be considered an interval :

$$[x] \cup [y] = \{a \in \mathbb{R} | a \in [x] \text{ or } a \in [y]\}$$

In order to assure that the union of two intervals will also be an interval we can consider an interval union that is, by definition, the convex hull of $[x] \cup [y]$. The interval union is denoted by \sqcup such that

$$[x] \sqcup [y] \triangleq [[x] \cup [y]]$$

Now we generalize the definitions stated in this section to other dimensions.

3.3.2 Boxes or Interval Vectors

An interval vector $[\mathbf{x}]$ of \mathbb{R}^n is a box, and it is, by definition, a Cartesian product between n intervals. It can be seen as a collection of intervals, one for each dimension, that specify the ranges along each axis. We denote by \mathbb{IR}^n the set of all the boxes of \mathbb{R}^n .

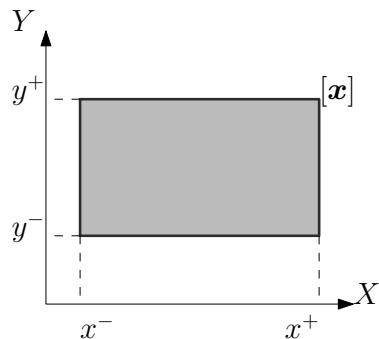


Figure 3.3: Interval box $[\mathbf{x}] = [x^-, x^+] \times [y^-, y^+]$. $[\mathbf{x}] \in \mathbb{IR}^2$.

Each interval within the box represents the uncertainty or variability along a specific dimension. For example, in two dimensions, a box can be represented as $[x^-, x^+] \times [y^-, y^+]$, where $[x^-, x^+]$ represents the interval along the x-axis and $[y^-, y^+]$ represents the interval along the y-axis. The box encompasses all points within the range of the intervals, effectively defining a rectangular region in the xy-plane, Figure 3.3.

Operations on boxes in interval analysis involve applying the respective operations on the intervals within each dimension. For instance, addition

of two boxes is performed by adding the corresponding intervals along each dimension.

Classically, the width of a box $[\mathbf{x}] \in \mathbb{IR}^n$ is defined as the maximum absolute difference between the upper and lower bounds of each component interval:

$$w([\mathbf{x}]) = \max_{1 \leq i \leq n} (w([x_i]))$$

We say that a box $[\mathbf{x}] \in \mathbb{IR}^n$ is empty if exists one of its components that is empty:

$$[\mathbf{x}] = \emptyset \text{ if } \exists i \in \{1, \dots, n\} \text{ such that } [x_i] = \emptyset$$

3.3.3 Inclusion Functions

Inclusion functions plays a vital role in interval analysis by providing means to propagate uncertainties throughout calculations involving intervals. It serves the purpose of accurately computing a box that encompasses the direct image of a given input box by a function \mathbf{f} .

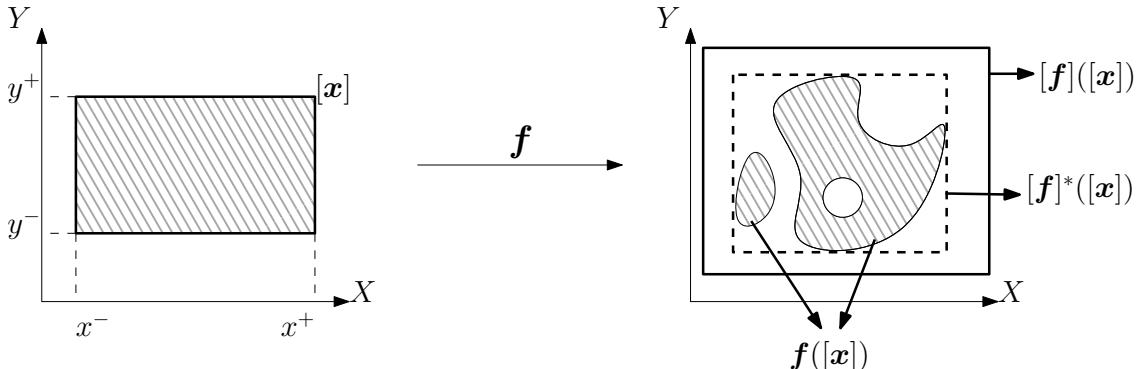


Figure 3.4: The image of a two-dimensional box by a function $\mathbf{f} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ can assume any shape. An inclusion function $[\mathbf{f}] : \mathbb{IR}^2 \rightarrow \mathbb{IR}^2$ computes a box guaranteed to contain all the possible solutions. The minimal inclusion function is represented by $[\mathbf{f}]^*$.

We consider a function $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, that can assume a non-linear character. An inclusion function of \mathbf{f} is $[\mathbf{f}] : \mathbb{IR}^n \rightarrow \mathbb{IR}^m$, such that

$$\forall [\mathbf{x}] \in \mathbb{IR}^n, \mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}])$$

The image $\mathbf{f}([\mathbf{x}])$ can be non-convex (it contains at least one pair of points for which the line segment connecting them lies partially or entirely outside the set) or disconnected (it can be split into two or more disjoint subsets that have no common points or overlap), assuming any shape. The inclusion function

$[f]$ of f computes a box $[f]([x])$ guaranteed to contain $f([x])$. This notion is illustrated in Figure 3.4 for a two-dimensional example. We say that the inclusion function is minimal if for any input box $[x]$, $[f]([x])$ is the smallest box that contains $f([x])$, and it is denoted $[f]^*([x])$.

An inclusion function $[f]$ for f is thin if produces intervals that collapse to a single value when the input intervals are reduced to points. Therefore, if $[f]$ is thin the following relation stands:

$$[x] = \mathbf{x} \in \mathbb{R}^n \Rightarrow [f]([x]) = f(\mathbf{x}) \in \mathbb{R}^n$$

We say that $[f]$ is convergent if it consistently provides narrower enclosures for the range of possible values as the size, or precision, of the input intervals is refined or increased. In other words, as the input intervals become more precise or granular, the inclusion function approaches the exact or optimal enclosure of the function's range. Therefore, if $[f]$ is convergent it is also necessarily thin. A mathematical definition for convergence of inclusion functions is as follows: for any sequence of boxes $[x](k)$ we have

$$\lim_{k \rightarrow \infty} w([x](k)) = 0 \Rightarrow \lim_{k \rightarrow \infty} w([f]([x](k))) = 0$$

This property ensures that the computed intervals progressively improve in accuracy and approach the true range of values of the underlying function, and it is essential for proving the convergence of interval solvers such as the ones presented in Section 3.5.

Pessimism

There are many reasons that contribute for having inclusion functions that are not minimal, that lead to wider intervals than strictly necessary, reducing precision. Nonlinearity, for example, is one of the main reasons to pessimism when evaluating a function f through interval arithmetic. In addition, interval computations assume that the inputs are independent when in reality, there can be functional dependencies among the variables. When overlooked, these functional dependencies may lead to wider enclosures and increased pessimism. This will usually be the case when the same variable appears several times on a function, for example, if we take $f(x) = \cos(x).\sin(x)$. Now, assuming an input interval $[x] = [0, \frac{\pi}{2}]$, we obtain

$$\begin{aligned} [f]([x]) &= \cos([x]).\sin([x]) \\ &= \cos([0, \frac{\pi}{2}]).\sin([0, \frac{\pi}{2}]) \\ &= [0, 1].[0, 1] \\ &= [0, 1] \end{aligned}$$

Interval arithmetic does not take into consideration that $\cos(x) = 1$ can not happen at the same moment as $\sin(x) = 1$, which would make it impossible to have 1 in the solution set, that in reality is equal to $[0, 0.5]$.

When intervals are used as inputs to subsequent operations, the inherent overestimation from previous computations can amplify and accumulate. This phenomenon is referred to as wrapping effect.

Later in this chapter, in Section 3.5, we present a technique called sub-paving that addresses the issue of pessimism by partitioning the input interval into smaller sub-intervals, enabling finer analysis and reducing overestimation.

3.4 Set-Valued Constraint Satisfaction Problems (SVC-SPs)

Constraint Satisfaction Problems (CSPs) are concerned with determining values for a set of variables, subject to specific constraints that define the relationships and limitations between them. Formally, a CSP consists of three components: variables, domains, and constraints.

Variables represent the unknowns or quantities that need to be assigned values to satisfy the problem. Each variable has a domain, which is a set of possible values it can take. Constraints define the conditions or relationships that must be satisfied among the variables, also imposing restrictions on the allowed combinations of values. The goal in a CSP is to find an assignment of values to the variables such that all constraints are satisfied. This assignment is often referred to as a solution or a feasible solution.

Many problems in computer science can be modeled as a constraint satisfaction problem. For example, the automatic detection of the type of an expression in a formal language [80], map coloring in graph theory [81], logic puzzles [82], etc.

Solving CSPs typically involves employing algorithms and techniques that explore the solution space and incrementally refine assignments to satisfy the constraints. Backtracking algorithms, constraint propagation, and search heuristics are common approaches adopted [83]. In the case where the variables of the CSP are represented as intervals of \mathbb{R}^n , feasible solutions can be narrowed using interval analysis [84]. In this case, problems are named Set-Valued Constraint Satisfaction Problems (SVCSPs), and the constraints can be expressed as interval relations, such as equality, inequality, containment, etc.

Many problems in robotics can be tackled as a SVCSP, such as determining the area explored by a robot [29] and range-only localization [85], for example.

Within the interval analysis framework we find two operators that can be

employed to refine interval enclosures and propagate constraints accurately when solving SVCSPs, they are named contractors and separators. These operators play a crucial role aiding in the removal of infeasible regions that do not satisfy given constraints, and they will be employed in different contexts in the subsequent Chapters of this thesis. Therefore, these operators are presented in this section.

3.4.1 Definition of a SVCSP

We represent a SVCSP by \mathcal{H} . It is composed by n_x variables $x_i \in \mathbb{R}$, each of them belonging to a corresponding domain $[x_i] \in \mathbb{IR}$, $i \in \{1, \dots, n_x\}$. The vector of variables is defined as

$$\mathbf{x} = (x_1, \dots, x_{n_x})^T$$

and the initial knowledge about its value is the interval vector

$$[\mathbf{x}] = [x_1] \times \dots \times [x_{n_x}]$$

These variables are linked by n_f relations or constraints that can be represented by a function $\mathbf{f} : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_f}$. The SVCSP problem can be formulated as:

$$\mathcal{H} : (\mathbf{f}(\mathbf{x}) \leq \mathbf{0}, \mathbf{x} \in [\mathbf{x}])$$

and $\mathbb{S}_{\mathcal{H}}$ represents its solutions set

$$\mathbb{S}_{\mathcal{H}} = \{\mathbf{x} \in [\mathbf{x}] \mid \mathbf{f}(\mathbf{x}) \leq \mathbf{0}\}$$

Contracting \mathcal{H} means to replace the current domain $[\mathbf{x}]$ by a smaller domain $[\mathbf{x}']$ such that $\mathbb{S}_{\mathcal{H}} \subset [\mathbf{x}'] \subset [\mathbf{x}]$, meaning that a valid contraction does not discard any possible solution while removing pessimism. An optimal contraction of \mathcal{H} will replace $[\mathbf{x}]$ by the smallest interval vector that contains $\mathbb{S}_{\mathcal{H}}$.

3.4.2 Contractors

Let us consider a constraint $f(\mathbf{x}) = 0$ with solution set \mathbb{S} , we say that \mathcal{C} is a contractor consistent with this solution set. It is an operator that when applied to $[\mathbf{x}] \in \mathbb{IR}^{n_x}$ returns another interval vector $\mathcal{C}([\mathbf{x}]) \in \mathbb{IR}^{n_x}$ such that $\mathcal{C}([\mathbf{x}]) \subseteq [\mathbf{x}]$. In this operation, only vectors $\mathbf{x} \in [\mathbf{x}]$ that do not respect constraint $f(\mathbf{x}) = 0$ are removed from $[\mathbf{x}]$. Therefore, we can say that $\mathcal{C}([\mathbf{x}]) \cap \mathbb{S} = [\mathbf{x}] \cap \mathbb{S}$. The definition bellow was first presented by [86] and adapted by [87]:

Definition 1 (Contractor). A contractor is a mapping $\mathcal{C} : \mathbb{IR}^{n_x} \rightarrow \mathbb{IR}^{n_x}$ such that:

$$\forall [\mathbf{x}] \in \mathbb{IR}^{n_x}, \mathcal{C}([\mathbf{x}]) \subseteq [\mathbf{x}] \text{ (Contractility)}$$

$$\text{If } f(\mathbf{x}) \leq 0 \text{ and } \mathbf{x} \in [\mathbf{x}] \Rightarrow \mathbf{x} \in \mathcal{C}([\mathbf{x}]) \text{ (Consistency)}$$

$$[\mathbf{x}] \subset [\mathbf{y}] \Rightarrow \mathcal{C}([\mathbf{x}]) \subset \mathcal{C}([\mathbf{y}]) \text{ (Monotonicity)}$$

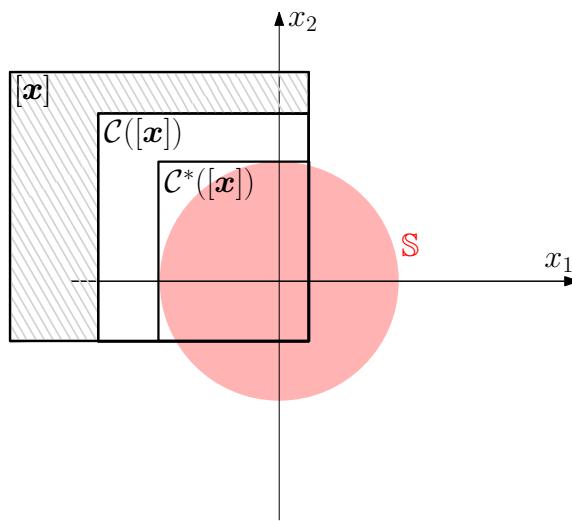


Figure 3.5: Contractor \mathcal{C} consistent with solution set \mathbb{S} . The dashed area is removed during contraction of $[\mathbf{x}]$ without removing any single point that belongs to \mathbb{S} . The minimal contractor is denoted \mathcal{C}^* .

The consistency property establishes that a solution consistent with f can not be removed during contraction by \mathcal{C} , the contractor associated to the constraint. For example, take the following constraint:

$$f(x_1, x_2) = x_1^2 + x_2^2 - 4 = 0 \quad (3.3)$$

In Figure 3.5 we illustrate in red the set of points that respect this constraint, and we name this set \mathbb{S} . The contractor \mathcal{C} is consistent with \mathbb{S} , and it will not remove a single point of \mathbb{S} from a box on which it is applied.

We say that a contractor is minimal if for all $[\mathbf{x}] \in \mathbb{IR}^{n_x}$, we have $\mathcal{C}([\mathbf{x}]) = [[\mathbf{x}] \cap \mathbb{S}]$. This means that the minimal contractor returns the convex hull, or the smallest box, that can be obtained by contracting $[\mathbf{x}]$ without removing a single point of \mathbb{S} . Where \mathbb{S} is the set of solutions that respect the constraint associated to \mathcal{C} . In Figure 3.5 we represent the minimal contractor associated to f in Equation 3.3 by \mathcal{C}^* .

Developing algorithms for contractors have been the subject of different works [88],[86], [89]. The major principle, in practice, is to evaluate contractors using interval arithmetic, as demonstrated below for the contractors of some simple constraints.

Example 1:

$$\boldsymbol{x} = (x_1, x_2, x_3)^T$$

$$x_1 - x_2 - x_3 = 0$$

$$\mathcal{C} : \begin{pmatrix} [x_1] \\ [x_2] \\ [x_3] \end{pmatrix} = \begin{pmatrix} [x_1] \cap ([x_2] + [x_3]) \\ [x_2] \cap ([x_1] - [x_3]) \\ [x_3] \cap ([x_1] - [x_2]) \end{pmatrix}$$

If the initial knowledge about the variables is, for example, the interval vector $\boldsymbol{x} = [0, 3] \times [-2, 2] \times [4, 5]$, the contraction will give $\mathcal{C}(\boldsymbol{x}) = [2, 3] \times [-2, -1] \times [4, 5]$.

Example 2:

$$\boldsymbol{x} = (x_1, x_2)^T$$

$$\exp(x_2) - x_1 = 0$$

$$\mathcal{C} : \begin{pmatrix} [x_1] \\ [x_2] \end{pmatrix} = \begin{pmatrix} [x_1] \cap \exp([x_2]) \\ [x_2] \cap \log([x_1]) \end{pmatrix}$$

In [86], the concept of contractor programming was introduced with the objective of extending standard operations on sets to contractors. With this, a basic formalism was defined where basic contractors can be combined in order to solve complex constraints. For example, we have :

Intersection $(\mathcal{C}_1 \cap \mathcal{C}_2)([\boldsymbol{x}]) \triangleq \mathcal{C}_1([\boldsymbol{x}]) \cap \mathcal{C}_2([\boldsymbol{x}])$

Union $(\mathcal{C}_1 \cup \mathcal{C}_2)([\boldsymbol{x}]) \triangleq \mathcal{C}_1([\boldsymbol{x}]) \sqcup \mathcal{C}_2([\boldsymbol{x}])$

Composition $(\mathcal{C}_1 \circ \mathcal{C}_2)([\mathbf{x}]) \triangleq \mathcal{C}_1(\mathcal{C}_2([\mathbf{x}]))$

The negation $\neg\mathcal{C}$ of a contractor \mathcal{C} can be defined as follows:

$$\neg\mathcal{C}([\mathbf{x}]) = \{\mathbf{x} \in [\mathbf{x}] \mid \mathbf{x} \notin \mathcal{C}([\mathbf{x}])\}$$

As illustrated on Figure 3.5 this set is not necessarily a box and to deal with this problem, separators were introduced.

3.4.3 Separators

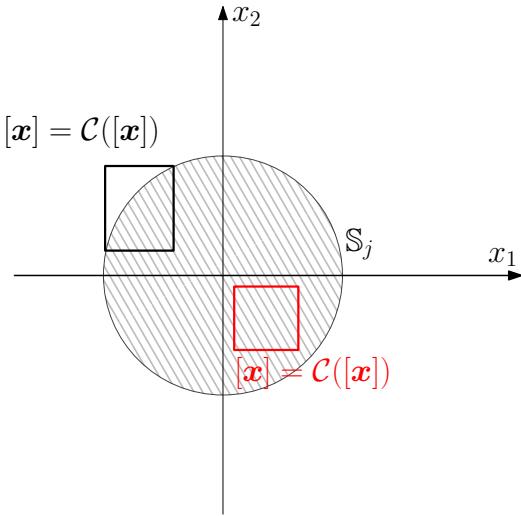


Figure 3.6: The dashed area represents the solutions set \mathbb{S} for a constraint f in a SVCSP with $[\mathbf{x}] \in \mathbb{IR}^2$. The contractor \mathcal{C} associated to this constraint is not able to contract any of the boxes $[\mathbf{x}]$ represented in black and in red without removing a possible solution to the problem. Therefore, the non contractility of a box is not able to prove that a box is completely inside the solution set or only on its border.

Imagine that we have a box $[\mathbf{x}] \in \mathbb{IR}^{n_x}$ and we want to know if all $\mathbf{x} \in [\mathbf{x}]$ are a possible solution for a constraint f . In other words, if $\forall \mathbf{x} \in [\mathbf{x}], \mathbf{x} \in \mathbb{S}$, where \mathbb{S} represents the solution set for constraint f , that we assume to be of non-empty volume. We know that if this is true, we have that $\mathcal{C}([\mathbf{x}]) = [\mathbf{x}]$, however, the non contractility of $[\mathbf{x}]$ is not enough for proving that all the points inside it are also a solution to the problem. As presented in Figure 3.6, a contractor does not provide enough information to determine if a non-contractile box is fully inside the solution set or only on its boundary. In

this case, it can only provide an outer approximation of the set of possible solutions.

In this context, the concept of a separator is introduced in interval analysis to characterize both inner and outer approximations of a solution set, providing a more complete representation. For that, the complementary contractor $\bar{\mathcal{C}}$, associated to the solution set $\bar{\mathbb{S}}$ will also be employed. Following the opposite definition of contractors, the complementary contractor guarantees that every non-solution to the original constraint is contained in the box resultant of the contraction. A separator \mathbb{S} associated to f is composed by \mathcal{C} and $\bar{\mathcal{C}}$ that can also be respectively named \mathcal{S}^{out} , because it contracts the space out of the solution set, and \mathcal{S}^{in} because it contracts the interior of the solution set.

We present below its formal definition as stated in [90]:

Definition 2 (Separator). *A separator associated to a constraint f with solution set \mathbb{S} is an operator that can be seen as a pair of contractors $\{\mathcal{S}^{in}, \mathcal{S}^{out}\}$, such that:*

$$\begin{aligned}\mathcal{S} : \mathbb{IR}^{n_x} &\rightarrow \mathbb{IR}^{n_x} \times \mathbb{IR}^{n_x} \\ [\mathbf{x}] &\mapsto ([\mathbf{x}_{in}] \times [\mathbf{x}_{out}])\end{aligned}$$

and it respects the following properties:

- (i) $[\mathbf{x}_{in}] = \mathcal{S}^{in}([\mathbf{x}]),$
- (ii) $[\mathbf{x}_{out}] = \mathcal{S}^{out}([\mathbf{x}]),$
- (iii) $[\mathbf{x}] = [\mathbf{x}_{in}] \cup [\mathbf{x}_{out}],$
- (iv) $[\mathbf{x}_{out}] \cap \mathbb{S} = [\mathbf{x}] \cap \mathbb{S},$
- (v) $[\mathbf{x}_{in}] \cap \bar{\mathbb{S}} = [\mathbf{x}] \cap \bar{\mathbb{S}}.$

Furthermore, we have the following relations:

$$[\mathbf{x}] \setminus [\mathbf{x}_{in}] \subset \mathbb{S}$$

$$([\mathbf{x}] \setminus [\mathbf{x}_{out}]) \cap \mathbb{S} = \emptyset$$

The notions established on this definition are illustrated in Figure 3.7.

If we consider the problem initially stated at the beginning of this section of determining if all vectors on a box can be a solution to a constraint, we can now determine this by analysis the output boxes of a separator. As illustrated in Figure 3.8, when the separator is applied to a box that is fully inside the

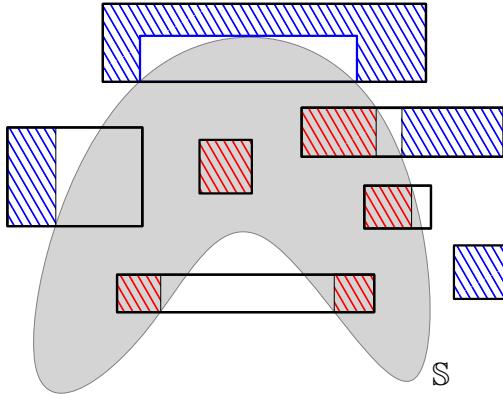


Figure 3.7: Examples of applications of separators considering a solution set \mathbb{S} represented in gray. Dashed areas in red are removed by the inner contractor \mathcal{S}^{in} and dashed blue areas are removed by the outer contractor \mathcal{S}^{out} .

solution, we obtain $[\mathbf{x}_{out}] = [\mathbf{x}]$ and $[\mathbf{x}_{in}] = \emptyset$. If, on the contrary, $[\mathbf{x}_{in}] \cap [\mathbf{x}_{out}] \neq \emptyset$, it means that this box intersects the boundary $\partial\mathbb{S}$ of the solution set. All this information will be important when constructing a paver, as it will be explained in Section 3.5.2.

The same operations applied on contractors can be extended to separators [86]. Let $\mathcal{S} = \{\mathcal{S}^{in}, \mathcal{S}^{out}\}$ be a separator, its complement can be defined as

$$\overline{\mathcal{S}} = \{\mathcal{S}^{out}, \mathcal{S}^{in}\}$$

and now if we take two separators, $\mathcal{S}_1 = \{\mathcal{S}_1^{in}, \mathcal{S}_1^{out}\}$ and $\mathcal{S}_2 = \{\mathcal{S}_2^{in}, \mathcal{S}_2^{out}\}$, the following set operations can be defined for them:

$$\text{Intersection } \mathcal{S}_1 \cap \mathcal{S}_2 \triangleq \{\mathcal{S}_1^{in} \cup \mathcal{S}_2^{in}, \mathcal{S}_1^{out} \cap \mathcal{S}_2^{out}\}$$

$$\text{Union } \mathcal{S}_1 \cup \mathcal{S}_2 \triangleq \{\mathcal{S}_1^{in} \cap \mathcal{S}_2^{in}, \mathcal{S}_1^{out} \cup \mathcal{S}_2^{out}\}$$

$$\text{Difference } \mathcal{S}_1 \setminus \mathcal{S}_2 \triangleq \mathcal{S}_1 \cap \overline{\mathcal{S}_2}$$

The application of the projection operator needs to be discussed more in details.

Projection of Separators

The definition of the projection of sets was presented in Section 3.2.1, now we extend it for separators.

Given sets $\mathbb{X} \subset \mathbb{R}^n$, $\mathbb{Y} \subset \mathbb{R}^q$ and $\mathbb{Z} = \mathbb{X} \times \mathbb{Y}$, let us consider a separator \mathcal{S} consistent with a set $\mathbb{Z}_1 \subset \mathbb{Z}$. The projection of \mathcal{S} along \mathbb{X} will be consistent with the set $\text{proj}_{\mathbb{X}}(\mathbb{Z}_1)$ if the following definition is considered:

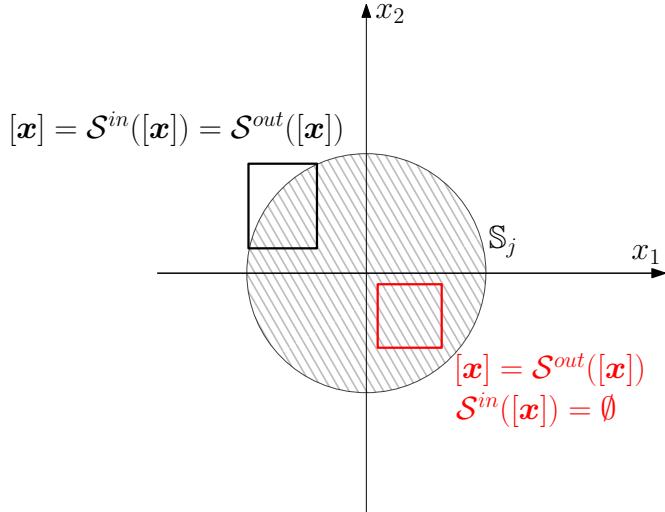


Figure 3.8: The solution to the problem stated at the beginning of this section can be found by analysis of the boxes returned by the separator associated to the solution set \mathbb{S} . For the box in black we have that $[\mathbf{x}_{in}] = [\mathbf{x}_{out}]$ because it is impossible to contract $[\mathbf{x}]$ without removing a solution to both contractors associated \mathcal{S} .

Definition 3 (Projection of Separators). We consider the separator $\mathcal{S}([\mathbf{x}], [\mathbf{y}]) = \{\mathcal{S}^{in}([\mathbf{x}], [\mathbf{y}]), \mathcal{S}^{out}([\mathbf{x}], [\mathbf{y}])\}$ consistent with \mathbb{Z}_1 . The projection of \mathcal{S} along \mathbb{X} with respect to $[\mathbf{y}]$ can be defined as

$$\text{proj}_{\mathbf{x}}(\mathcal{S})([\mathbf{x}]) = \left\{ \bigcap_{\mathbf{y} \in [\mathbf{y}]} \partial_{\mathbf{x}} \mathcal{S}^{in}([\mathbf{x}], \mathbf{y}), \bigcup_{\mathbf{y} \in [\mathbf{y}]} \partial_{\mathbf{x}} \mathcal{S}^{out}([\mathbf{x}], \mathbf{y}) \right\}$$

where $\partial_{\mathbf{x}} \mathcal{S}^{in}([\mathbf{x}], \mathbf{y}) = \{\mathbf{x} \in [\mathbf{x}] \mid (\mathbf{x}, \mathbf{y}) \in \overline{\mathbb{Z}_1}\}$ and $\partial_{\mathbf{x}} \mathcal{S}^{out}([\mathbf{x}], \mathbf{y}) = \{\mathbf{x} \in [\mathbf{x}] \mid (\mathbf{x}, \mathbf{y}) \in \mathbb{Z}_1\}$.

The proof for this definition can be found in [91]. And for more details about separators and its algebra, the reader can refer to [90].

Separators are employed by pavers in order to characterize sets by an inner and outer approximation as it will be presented in Section 3.5.

3.4.4 Image Contractor

As previously presented in this chapter, SVCSPs are systems doted of a set of constraints that are usually expressed by equations. Then, from these equations, one can construct, using separators, an outer and an inner approximation for the solution set of the concerned system. However, quite often, in practice, this order is not respected, and we might have a solution set for a problem but

not the equations for the constraints that defined it. Without these equations, one can not create contractors and separators that will allow for a characterization of the solution set through a classical approach using set operators. Finding the constraint equations for an existent solution set might be an easy task for sets such as \mathbb{S} illustrated in Figure 3.8. However, in most cases, this will be cumbersome [92], as it would be for the set \mathbb{S} presented in Figure 3.7.

In this scenario, in [92] and [93], the image contractor was introduced with the objective of constructing contractors and separators for sets without having to explicitly define equations for the constraints that created it.

Let us consider a set $\mathbb{S} \in \mathbb{IR}^2$, this set can be represented through an occupancy grid that in turn can be represented by a binary image $\mathcal{B}_{\mathbb{S}}$. In this representation, each pixel of the image has a value of 1 if it intersects \mathbb{S} , and 0 otherwise. The binary image $\mathcal{B}_{\mathbb{S}}$ is defined by a set of parameters: its size $p_x \times p_y$, pixel resolution $\epsilon_y \in \mathbb{R}^+$ and $\epsilon_x \in \mathbb{R}^+$ respectively on the vertical and horizontal axes, and a point on the plane (x_0, x_1) that serves as the initial point for the grid representation.

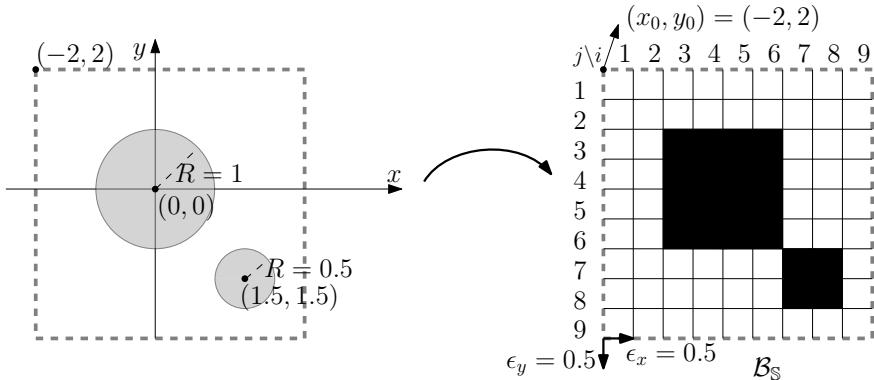


Figure 3.9: On the right we have an example of a binary image $\mathcal{B}_{\mathbb{S}}$ representing the set illustrated in gray on the left composed of two disconnected subsets that are disks centered in $(0, 0)$ and $(1.5, 1.5)$. The binary image is of size $p_x \times p_y = 9$ pixels $\times 9$ pixels, pixel resolutions $\epsilon_x = 0.5$ and $\epsilon_y = 0.5$ and initial position $(x_0, x_1) = (-2, 2)$. Pixels represented in black on the binary image are pixels whose value is equal to one.

For $i \in \{1, \dots, p_x\}$ and $j \in \{1, \dots, p_y\}$, we have that $\mathcal{B}_{\mathbb{S}}(i, j) \in \{0, 1\}$. The pixel value will be defined according to the distribution of set \mathbb{S} on the grid space. The area in the plane represented by a pixel (i, j) is an interval vector $P_{ij} \in \mathbb{IR}^2$ such that $P_{ij} = [x_0 + i\epsilon_x, x_0 + (i+1)\epsilon_x] \times [y_0 - (j+1)\epsilon_y, y_0 - j\epsilon_y]$

and we have

$$\mathcal{B}_{\mathbb{S}}(i, j) = \begin{cases} 1 & \text{if } P_{ij} \cap \mathbb{S} \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \quad (3.4)$$

There are different numerical methods for determining if a box intersects the set \mathbb{S} depending on the information that is available about this set. In this work, we will have a trajectory, or a set of trajectories, representing the boundary of sets. The method for determining grid occupancy from this information is presented later in this document. In Figure 3.9, we have an example of a binary image for a simple disconnected set, the precision of the occupancy grid will depend on pixel resolution.

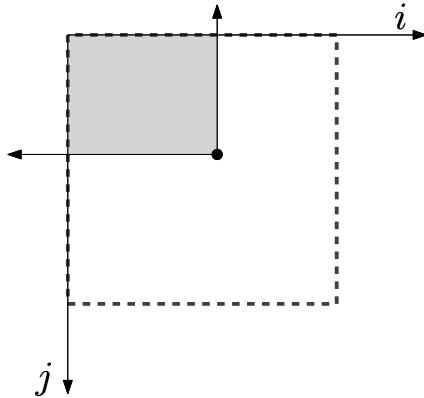


Figure 3.10: On an integral image, the value of a pixel corresponds to the cumulative sum of pixels, on the original image, on the rectangular area above and on the left of this pixel.

The central concept underlying the image contractor is the integral image. An integral image is a technique extensively used in computer vision [94]. It is a form of an image where each pixel value represents the cumulative sum of the pixel values in the original image, up to that position, in both horizontal and vertical directions, as demonstrated in Figure 3.10. The integral image associated to a set \mathbb{S} , that we name $\mathcal{I}_{\mathbb{S}}$, is constructed by iteratively summing the pixel values in the respective binary image $\mathcal{B}_{\mathbb{S}}$. The resulting integral image allows for quick retrieval of pixel sums in any rectangular region, facilitating efficient computation within the image contractor. It also provides a fast and efficient way to compute various properties and statistics of an image, such as means and variances, within a region.

For $i \in \{1, \dots, p_x\}$ and $j \in \{1, \dots, p_y\}$, the integral image can be defined

$j \setminus i$	1	2	3	4	5	6	7	8	9
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	0	0	1	2	3	4	4	4	4
4	0	0	2	4	6	8	8	8	8
5	0	0	3	6	9	12	12	12	12
6	0	0	4	8	12	16	16	16	16
7	0	0	4	8	12	16	17	18	18
8	0	0	4	8	12	16	18	20	20
9	0	0	4	8	12	16	18	20	20

 $\mathcal{I}_{\mathbb{S}}$

Figure 3.11: Integral image associated to the binary image of the example presented in Figure 3.9.

as:

$$\mathcal{I}_{\mathbb{S}}(i, j) = \sum_{\substack{i' \leq i \\ j' \leq j}} \mathcal{B}_{\mathbb{S}}(i', j')$$

In Figure 3.11 we have the resultant integral image $\mathcal{I}_{\mathbb{S}}(i, j)$ for the binary image $\mathcal{B}_{\mathbb{S}}(i, j)$ illustrated in Figure 3.9.

The integral image can be computed once from the binary image and then, from this new representation, the number of 1-valued pixels contained in the original image within a rectangular area of any size can be obtained from only four operations. One does not have to parse the whole rectangular area for computing this value that is calculated in constant time with $O(1)$ complexity.

Let $\phi_{\mathbb{S}} : \mathbb{IN} \rightarrow \mathbb{N}_0$ be a function that returns the sum of pixels in a box $[\mathbf{n}] = [n_1^-, n_1^+] \times [n_2^-, n_2^+]$ on the original image. This function evaluates the value of pixels on the integral image on the coordinates of the corners of this box:

$$\phi_{\mathbb{S}}([\mathbf{n}]) = \mathcal{I}_{\mathbb{S}}(n_1^+, n_2^+) - \mathcal{I}_{\mathbb{S}}(n_1^+, n_2^-) - \mathcal{I}_{\mathbb{S}}(n_1^-, n_2^+) + \mathcal{I}_{\mathbb{S}}(n_1^-, n_2^-)$$

A visual explanation of this function is given on Figure 3.12.

Function $\phi_{\mathbb{S}}$ is used by the image contractor. Consider a box $[\mathbf{y}] = [y_0] \times [y_1] \in \mathbb{IR}^2$ that we want to contract with respect to a solution set \mathbb{S} . The following steps will take place

- (i) Create the binary image $\mathcal{B}_{\mathbb{S}}$ and its correspondent integral image $\mathcal{I}_{\mathbb{S}}$ for set \mathbb{S} ;
- (ii) Map box $[\mathbf{y}] \in \mathbb{IR}^2$ to its pixel representative $[\mathbf{n}] = n_x \times n_y$ with

$$n_x = [\text{floor}((y_0^- - x_0)/\epsilon_x), \text{ceil}((y_0^+ - x_0)/\epsilon_x)]$$

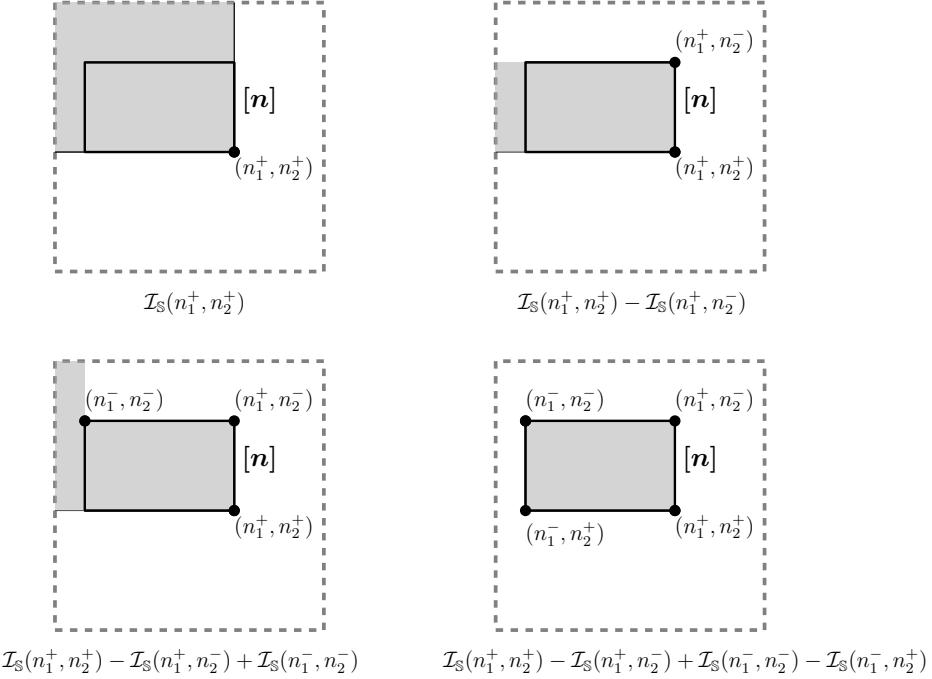


Figure 3.12: The four images show the steps for computing the number of non-zero pixels within box $[x]$ on the original image. The gray rectangles correspond to the area on the binary image comprised by the respective sum of pixels of the integral image.

and

$$n_y = [\text{floor}((y_1^- - x_1)/\epsilon_y), \text{ceil}((y_1^+ - x_1)/\epsilon_y)]$$

where *floor* and *ceil* are functions that round a given number to the nearest integer in different directions. The former downward and the latter upward;

- (iii) Contract $[n]$ in order to find the smallest box $[m] \subset [n]$ such that $\phi_S([n]) = \phi_S([m])$. This is done by the image contractor that we denote $\mathcal{C}_{\mathcal{I}}$ and it satisfies:

$$\mathcal{C}_{\mathcal{I}}([n]) = [m] = [m_0^-, m_0^+] \times [m_1^-, m_1^+]$$

where

$$m_0^- = \max(\text{for } x \in [n_0] \text{ such that } \phi_S([n_0^-, x] \times [n_1]) = 0) \text{ or } n_0^-$$

$$m_0^+ = \min(\text{for } x \in [n_0] \text{ such that } \phi_S([x, n_0^+] \times [n_1]) = 0) \text{ or } n_0^+$$

$$m_1^- = \max(\text{for } x \in [n_1] \text{ such that } \phi_S([n_0] \times [n_1^-, x]) = 0) \text{ or } n_1^-$$

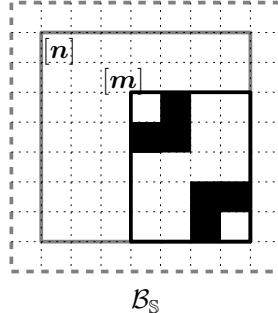


Figure 3.13: Example of contraction of box $[n]$ into $[m]$ without removing any 1-valued pixel from the resultant box.

$$m_1^+ = \min(\text{for } x \in [n_1] \text{ such that } \phi_{\mathcal{S}}([n_0] \times [x, n_1^+]) = 0) \text{ or } n_1^+$$

An example of contraction is illustrated in Figure 3.13.

- (iv) Map $[m]$ from image coordinates to workspace coordinates again to obtain $[y']$, that is the contraction of the original box $[y]$ with respect to set \mathbb{S} .

$$[y'] = [m_0^- \epsilon_x + x_0, m_0^+ \epsilon_x + x_0] \times [m_1^- \epsilon_y + x_1, m_1^+ \epsilon_y + x_1]$$

The image contractor can be generalized to other dimensions [95] [96] but in this work we only use the 2-dimensional case presented in this section.

An image separator for a set \mathbb{S} is constructed from an image contractor associated to set \mathbb{S} and from the complementary image contractor associated to set $\bar{\mathbb{S}}$.

3.5 Subpaving

In Section 3.3.3, we presented the concept of pessimism in representing solution sets as intervals, along with the consequential wrapping effect that arises when this pessimism accumulates through successive operations on the intervals. To mitigate this pessimism, a potential solution involves subdividing the space into smaller interval vectors and classifying them based on whether they intersect with specific portions of the set under-representation or not. By employing this approach, we can effectively reduce the impact of pessimism and refine the accuracy of interval-based representations. This new wrapper is called subpaving and it will be an import asset when solving SVCSPs and to the set-inversion problem, that is also presented in this section.

Let us consider a set $\mathbb{X} \subset \mathbb{R}^n$ that can assume any form and even be disconnected and we want to wrap this set under a subpaving representation.

The process starts with an initial interval vector, that can also represent a union of disconnected interval vectors, that encompass the entire space of interest $[x_0] \in \mathbb{IR}^n$. This initial estimation is then further divided into smaller subintervals using bisection.

Through bisection and analysis of the subintervals of the initial estimation $[x_0]$ for set \mathbb{X} , the subpaving \mathbb{K} of $[x_0]$ is created as a union of non overlapping boxes [97], and as presented in [87] we have

$$[x_0] = \bigcup_{[\mathbf{b}] \in \mathbb{K}} [\mathbf{b}]$$

During the division process of $[x_0]$ into sub interval vectors, each box $[\mathbf{b}] \in \mathbb{K}$ is examined to determine its properties and characteristics. This examination involves evaluating the set membership or intersection with the target set \mathbb{X} being represented. Based on this evaluation, the subintervals are classified into different categories, such as those completely contained within the target set, those completely outside the set, and those that intersect with the set boundary. In this context, we can define an inner-approximation \mathbb{K}^- of \mathbb{X} as a union of boxes of \mathbb{K} that are subsets of \mathbb{X} :

$$\mathbb{K}^- = \bigcup_{\substack{[\mathbf{b}] \in \mathbb{K}, \\ [\mathbf{b}] \subset \mathbb{X}}} [\mathbf{b}]$$

An outer-approximation \mathbb{K}^+ can also be defined such that:

$$\mathbb{K}^+ = \bigcup_{\substack{[\mathbf{b}] \in \mathbb{K}, \\ [\mathbf{b}] \cap \mathbb{X} \neq \emptyset}} [\mathbf{b}]$$

From these definitions we can conclude that $\mathbb{R}^n \setminus \mathbb{K}^+$ is a set of points that do not belong to the target set and that the following relation will be respected:

$$\mathbb{K}^- \subset \mathbb{X} \subset \mathbb{K}^+$$

In Figure 3.14 the concepts of subpaving and of inner and outer approximations are illustrated.

3.5.1 Set Inversion Via Interval Analysis (SIVIA) Algorithm

The set inversion problem involves determining the pre-image of a given set under a given function. In other words, given a function that maps elements from one set to another, the goal is to identify the set of all input values that map to a specific target set in the range. Let us consider a function

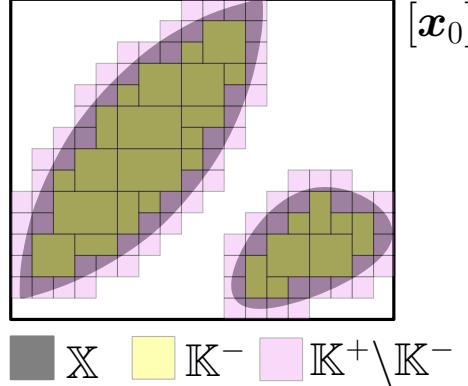


Figure 3.14: Inner and Outer approximation of set \mathbb{X} through subpaving.

$\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, we want to estimate the set $\mathbb{X} \subset \mathbb{R}^n$ from the knowledge of its image by \mathbf{f} that we denote $\mathbb{Y} \subset \mathbb{R}^m$. This is a set inversion problem that can be formally stated as follows:

$$\mathbb{X} = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{f}(\mathbf{x}) \in \mathbb{Y}\} = \mathbf{f}^{-1}(\mathbb{Y})$$

In [97] a solution to this problem using interval analysis was proposed through the Set Inversion Via Interval Analysis (SIVIA) algorithm that we present in this section. The algorithm assumes an inclusion function $[\mathbf{f}] : \mathbb{IR}^n \rightarrow \mathbb{IR}^m$ and it approximates a set $\mathbb{X} \subset \mathbb{R}^n$ from any $\mathbb{Y} \subset \mathbb{R}^m$ using a subpaving technique.

The SIVIA algorithm, that is presented in Algorithm 1, starts with an inclusion function $[\mathbf{f}]$, an initial enclosure for the solution set \mathbb{X} represented by $[\mathbf{x}_0]$, the image set \mathbb{Y} such that $\mathbf{f}(\mathbb{X}) = \mathbb{Y}$ and a precision parameter ϵ that determines when bisection should cease. The outputs of the algorithm are an outer approximation set \mathbb{X}^+ and an inner approximation set \mathbb{X}^- such that the relation $\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+$ is valid.

The algorithm examines the relation between the image of an interval vector $[\mathbf{x}]$ under the action of $[\mathbf{f}]$ and the target set \mathbb{Y} . From this analysis four distinct cases may arise:

1. $[\mathbf{x}]$ is out of the solution set $\rightarrow [\mathbf{x}]$ is discarded:

$$[\mathbf{f}](\mathbf{x}) \cap \mathbb{Y} = \emptyset$$

2. All vector $\mathbf{x} \in [\mathbf{x}]$ is on the solution set $\mathbb{X} \rightarrow [\mathbf{x}]$ is stored in both \mathbb{X}^+ and \mathbb{X}^- :

$$[\mathbf{f}](\mathbf{x}) \subset \mathbb{Y}$$

3. Undetermined case, there might be a vector $\mathbf{x} \in [\mathbf{x}]$ on the solution set \mathbb{X} but $[\mathbf{x}] \not\subset \mathbb{X}$ and $w([\mathbf{x}]) < \epsilon \rightarrow [\mathbf{x}]$ is stored in \mathbb{X}^+ :

$$[\mathbf{f}]([\mathbf{x}]) \not\subset \mathbb{Y}$$

$$[\mathbf{f}]([\mathbf{x}]) \cap \mathbb{Y} \neq \emptyset$$

4. Undetermined case, there might be a vector $\mathbf{x} \in [\mathbf{x}]$ on the solution set \mathbb{X} but $[\mathbf{x}] \not\subset \mathbb{X}$ and $w([\mathbf{x}]) \geq \epsilon \rightarrow [\mathbf{x}]$ is bisected and new tests are applied on each of the resulting interval vectors :

$$[\mathbf{f}]([\mathbf{x}]) \not\subset \mathbb{Y}$$

$$[\mathbf{f}]([\mathbf{x}]) \cap \mathbb{Y} \neq \emptyset$$

3.5.2 Paver

In Section 3.4 we presented the concept of constraint systems where a solution set to the problem is defined by a set of constraints. These solution sets can be approximated by a subpaving wrapper constructed by a paver. A paver uses separators to classify if an interval vector is part of the outer or inner approximation of a solution set or not a part of the solution at all.

Algorithm 2 was proposed by [91] and it is quite similar to Algorithm 1, with some differences that are now described. Algorithm 2 assumes on the input a separator \mathcal{S} associated to a solution set \mathbb{X} , that we want to characterize by a subpaving wrapper, an initial interval vector enclosure $[\mathbf{x}_0]$ and a precision parameter ϵ that determines when bisection should cease. The outputs of the algorithm are an outer approximation set \mathbb{X}^+ and an inner approximation set \mathbb{X}^- such that the relation $\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+$ is valid.

The algorithm analyzes if an interval vector represents a part of the space inside or outside the solution set \mathbb{X} associated with \mathcal{S} . For that, it starts by calling separator \mathcal{S} to contract $[\mathbf{x}]$ into $[\mathbf{x}_{in}]$ and $[\mathbf{x}_{out}]$. As presented in Section 3.4.3, $[\mathbf{x}] \setminus [\mathbf{x}_{in}]$ is by definition a subset of \mathbb{X} . Therefore, $[\mathbf{x}] \setminus [\mathbf{x}_{in}]$ can be directly added to \mathbb{X}^- and \mathbb{X}^+ . Then it analyzes the interval vector resultant of the intersection $\partial\mathcal{S}([\mathbf{x}]) = [\mathbf{x}_{in}] \cap [\mathbf{x}_{out}]$. If $w(\partial\mathcal{S}([\mathbf{x}])) < \epsilon$ this box is stored only in \mathbb{X}^+ and it will not be analyzed anymore. Otherwise, it is bisected and the separator is applied again on each one of the resultant boxes.

Example: Two Attached Goats

We present a classical problem [98] that can be solved using pavers. The problem considers two goats, each one attached to a fixed peg on a grassy

Algorithm 1 Set Inversion Via Interval Analysis (SIVIA) (in: $[f]$, $[x_0]$, \mathbb{Y}, ϵ , out: $\mathbb{X}^-, \mathbb{X}^+$)

```

1: Initialize:
     $\mathbb{X}^-, \mathbb{X}^+ = \emptyset$ 
     $\mathbb{Q} = \{[x_0]\}$ 
2: if  $\mathbb{Q} \neq \emptyset$  then take a t-box  $[x] \in \mathbb{Q}$  and remove it from  $\mathbb{Q}$  else return
   endif
3: if  $[f]([x]) \cap \mathbb{Y} = \emptyset$  then
   ▷  $[x]$  has no solutions
4: elseif  $[f]([x]) \subset \mathbb{Y}$  then
5:      $\mathbb{X}^+ \leftarrow \mathbb{X}^+ \cup [x]$                                 ▷ outer approximation set
6:      $\mathbb{X}^- \leftarrow \mathbb{X}^- \cup [x]$                                 ▷ inner approximation set
7: elseif  $w([x]) < \epsilon$  then
8:      $\mathbb{X}^+ \leftarrow \mathbb{X}^+ \cup [x]$                                 ▷ outer approximation set
9: else
10:    bisect  $[x]$  into  $[x]^{(1)}$  and  $[x]^{(2)}$ 
11:     $\mathbb{Q} \leftarrow \mathbb{Q} \cup \{[x]^{(1)}, [x]^{(2)}\}$ 
endif
12: Return to 2.

```

area, with ropes of known lengths L_1 and L_2 . The goats can move freely within the radius of their respective ropes, but their movements are limited by the length of it. We are going to tackle this problem from a 2-dimensional perspective and assume that one of the pegs is at a position $\mathbf{p}_1 \in \mathbb{R}^2$ known to be encompassed by the interval vector $[\mathbf{p}_1] \in \mathbb{IR}^2$ and the other at a position $\mathbf{p}_2 \in \mathbb{R}^2$ guaranteed to lie within the interval vector $[\mathbf{p}_2] \in \mathbb{IR}^2$. Assuming $L_1 = L_2 = 10$, $[\mathbf{p}_1] = [0, 1] \times [2, 10]$ and $[\mathbf{p}_2] = [10, 16] \times [0, 1]$, using the library Codac [99] for Python 3.10 we solve the following problems:

Calculate the grazing area for each individual goat on the lawn: This problem can be represented by a SVCSP with constraint $f : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$. Assuming that $\mathbf{p} = (p_x, p_y)$ is the position on the plane on which the goat in

Algorithm 2 Paver (in: \mathcal{S} , $[\mathbf{x}_0], \epsilon$, out: $\mathbb{X}^-, \mathbb{X}^+$)

```

1: Initialize:
     $\mathbb{X}^-, \mathbb{X}^+ = \emptyset$ 
     $\mathbb{Q} = \{[\mathbf{x}_0]\}$ 
2: if  $\mathbb{Q} \neq \emptyset$  then take a t-box  $[\mathbf{x}] \in \mathbb{Q}$  and remove it from  $\mathbb{Q}$  else return
endif

3:  $[\mathbf{x}_{in}], [\mathbf{x}_{out}] = \mathcal{S}([\mathbf{x}])$ 

4:  $\mathbb{X}^+ \leftarrow \mathbb{X}^+ \cup [\mathbf{x}] \setminus [\mathbf{x}_{in}]$   $\triangleright$  outer approximation set
5:  $\mathbb{X}^- \leftarrow \mathbb{X}^- \cup [\mathbf{x}] \setminus [\mathbf{x}_{in}]$   $\triangleright$  inner approximation set

6:  $\partial\mathcal{S}([\mathbf{x}]) = [\mathbf{x}_{in}] \cap [\mathbf{x}_{out}]$ 
7:
8: if  $w(\partial\mathcal{S}([\mathbf{x}])) < \epsilon$  then
9:      $\mathbb{X}^+ \leftarrow \mathbb{X}^+ \cup \partial\mathcal{S}([\mathbf{x}])$   $\triangleright$  outer approximation set

10: else
11:     bisect  $\partial\mathcal{S}([\mathbf{x}])$  into  $[\mathbf{x}]^{(1)}$  and  $[\mathbf{x}]^{(2)}$ 
12:      $\mathbb{Q} \leftarrow \mathbb{Q} \cup \{[\mathbf{x}]^{(1)}, [\mathbf{x}]^{(2)}\}$ 

endif

13: Return to 2.


---



```

consideration is attached to and let L be the size of the rope, this constraint can be formalized as:

$$f(\mathbf{x}, \mathbf{p}) = (x_1 - p_x)^2 + (x_2 - p_y)^2 - L^2$$

For a given peg position \mathbf{p} , we want to determine the set \mathbb{S} of points such that

$$\mathbb{S} = \{\mathbf{x} \in \mathbb{R}^2 \mid f(\mathbf{x}, \mathbf{p}) \leq 0\} \quad (3.5)$$

For this purpose a separator $\mathcal{S}([\mathbf{x}], [\mathbf{p}]) = \{\mathcal{S}^{in}([\mathbf{x}], [\mathbf{p}]), \mathcal{S}^{out}([\mathbf{x}], [\mathbf{p}])\}$ associated to \mathbb{S} can be created. Here we have $[\mathbf{x}] \in \mathbb{IR}^2$ and $[\mathbf{p}] \in \mathbb{IR}^2$, respectively representing the workspace for points on the lawn and possible positions of the peg.

The projection of \mathcal{S} along \mathbf{x} with respect to the peg position of each goat will give us the separator necessary to solve the problem. For example, for

goat one we have $\text{proj}_{\mathbf{x}}(\mathcal{S})([\mathbf{x}])$ with respect to $[\mathbf{p}] = [\mathbf{p}_1]$, that for simplicity we name it \mathcal{S}_1 . This separator is associated to the solution set:

$$\mathbb{S}_1 = \{\mathbf{x} \in \mathbb{R}^2 \mid \exists \mathbf{p}_1 \in [\mathbf{p}_1], f(\mathbf{x}, \mathbf{p}_1) \leq 0\}$$

Analogously, we have for goat two the separator $\text{proj}_{\mathbf{x}}(\mathcal{S})([\mathbf{x}])$ with respect to $[\mathbf{p}] = [\mathbf{p}_2]$, that we name \mathcal{S}_2 associated to the solution set:

$$\mathbb{S}_2 = \{\mathbf{x} \in \mathbb{R}^2 \mid \exists \mathbf{p}_2 \in [\mathbf{p}_2], f(\mathbf{x}, \mathbf{p}_2) \leq 0\}$$

The result to the problem can be obtained by a paver applying each of these separators as illustrated in Figures 3.15a and 3.15b.

Compute the combined grazing area accessible to at least one of the goats on the lawn: We are now interested in the set of points of the plane that can possibly be explored either by goat one or by goat two. The solution set \mathbb{S}_2 for this problem can be formally expressed as:

$$\begin{aligned}\mathbb{S}_{1|2} &= \{\mathbf{x} \in \mathbb{R}^2 \mid \exists \mathbf{p}_1 \in [\mathbf{p}_1], f(\mathbf{x}, \mathbf{p}_1) \leq 0 \text{ or } \exists \mathbf{p}_2 \in [\mathbf{p}_2], f(\mathbf{x}, \mathbf{p}_2) \leq 0\} \\ &= \mathbb{S}_1 \cup \mathbb{S}_2\end{aligned}$$

and it can be characterized by a separator $\mathcal{S}_{1|2}$ resultant of the union of separators \mathcal{S}_1 and \mathcal{S}_2 .

$$\mathcal{S}_{1|2} = \mathcal{S}_1 \cup \mathcal{S}_2$$

The subpaving obtained by a paver applying this separator is illustrated in Figure 3.15c.

Compute the regions on the lawn where only one of the goats can potentially graze: For this problem, the solution set can be formalized as:

$$\mathbb{S}_3 = (\mathbb{S}_1 \cup \mathbb{S}_2) \setminus (\mathbb{S}_1 \cap \mathbb{S}_2)$$

We are going to name the separator associated to this set \mathcal{S}_3 . Considering all the concepts introduced in Section 3.4.3 and using De Morgan rules, it can be concluded that:

$$\begin{aligned}\mathcal{S}_3 &= (\mathcal{S}_1 \cup \mathcal{S}_2) \cap (\overline{\mathcal{S}_1 \cap \mathcal{S}_2}) \\ &= (\mathcal{S}_1 \cup \mathcal{S}_2) \cap (\overline{\mathcal{S}_1} \cup \overline{\mathcal{S}_2}) \\ &= (\mathcal{S}_1 \cap \overline{\mathcal{S}_1}) \cup (\mathcal{S}_1 \cap \overline{\mathcal{S}_2}) \cup (\mathcal{S}_2 \cap \overline{\mathcal{S}_1}) \cup (\mathcal{S}_2 \cap \overline{\mathcal{S}_2}) \\ &= (\mathcal{S}_1 \cap \overline{\mathcal{S}_2}) \cup (\mathcal{S}_2 \cap \overline{\mathcal{S}_1})\end{aligned}$$

The resulting subpaving associated to \mathcal{S}_3 is illustrated in Figure 3.15d.

Find the overlapping region on the lawn where both goats can potentially graze: Now we consider the following solution set :

$$\begin{aligned}\mathbb{S}_4 &= \{\mathbf{x} \in \mathbb{R}^2 \mid \exists \mathbf{p}_1 \in [\mathbf{p}_1], f(\mathbf{x}, \mathbf{p}_1) \leq 0 \text{ and } \exists \mathbf{p}_2 \in [\mathbf{p}_2], f(\mathbf{x}, \mathbf{p}_2) \leq 0\} \\ &= \mathbb{S}_1 \cap \mathbb{S}_2\end{aligned}$$

We are going to name the separator associated to this set \mathcal{S}_4 such that:

$$\mathcal{S}_4 = \mathcal{S}_1 \cap \mathcal{S}_2$$

The resulting subpaving associated to \mathcal{S}_4 is illustrated in Figure 3.15e.

Compute the grazing area on the lawn guaranteed that can be accessed by at least one of the goats: For this problem we will associate a separator \mathcal{S}_5 to the solution set \mathbb{S}_5 defined as follows:

$$\mathbb{S}_5 = \{\mathbf{x} \in \mathbb{R}^2 \mid \forall \mathbf{p}_1 \in [\mathbf{p}_1], f(\mathbf{x}, \mathbf{p}_1) \leq 0 \text{ or } \forall \mathbf{p}_2 \in [\mathbf{p}_2], f(\mathbf{x}, \mathbf{p}_2) \leq 0\}$$

For that we are going to consider the complement of separator \mathcal{S} consistent with set \mathbb{S} in Equation 3.5. Its complement $\overline{\mathcal{S}}$ will be consistent with the following set:

$$\overline{\mathbb{S}} = \{\mathbf{x} \in \mathbb{R}^2 \mid f(\mathbf{x}, \mathbf{p}) > 0\}$$

Then, projecting $\overline{\mathcal{S}}$ along \mathbf{x} with respect to the peg position of each goat we obtain $\mathcal{S}_{\bar{1}} = \text{proj}_{\mathbf{x}}(\overline{\mathcal{S}})$ consistent with:

$$\mathbb{S}_{\bar{1}} = \{\mathbf{x} \in \mathbb{R}^2 \mid \exists \mathbf{p}_1 \in [\mathbf{p}_1], f(\mathbf{x}, \mathbf{p}_1) > 0\}$$

and $\mathcal{S}_{\bar{2}} = \text{proj}_{\mathbf{x}}(\overline{\mathcal{S}})$ consistent with:

$$\mathbb{S}_{\bar{2}} = \{\mathbf{x} \in \mathbb{R}^2 \mid \exists \mathbf{p}_2 \in [\mathbf{p}_2], f(\mathbf{x}, \mathbf{p}_2) > 0\}$$

And since \mathbb{S}_5 can be rewritten as

$$\begin{aligned}\mathbb{S}_5 &= \overline{\mathbb{S}_{\bar{1}}} \cup \overline{\mathbb{S}_{\bar{2}}} \\ &= \{\mathbf{x} \in \mathbb{R}^2 \mid \forall \mathbf{p}_1 \in [\mathbf{p}_1], f(\mathbf{x}, \mathbf{p}_1) \leq 0\} \cup \{\mathbf{x} \in \mathbb{R}^2 \mid \forall \mathbf{p}_2 \in [\mathbf{p}_2], f(\mathbf{x}, \mathbf{p}_2) \leq 0\}\end{aligned}$$

we obtain the following separator for the problem:

$$\mathcal{S}_5 = \overline{\mathcal{S}_{\bar{1}}} \cup \overline{\mathcal{S}_{\bar{2}}}$$

The result of an analysis of this separator by a paver is illustrated in Figure 3.15f.

3.6 Dedicated Sets

We can construct an interval of any mathematical object as long as the set of domain in which this interval is defined forms a complete lattice [100]. In mathematics, a lattice is a partially ordered set in which every pair of elements has both a greatest lower bound (infimum) and a least upper bound (supremum) and it is a structure where elements can be compared and combined in terms of meet \wedge and join \vee operations.

Formally, a lattice is defined as (ε, \leq) , where ε is a set and \leq is a partial order relation on ε . For any two elements x and y in ε , the infimum (or meet) of x and y , denoted as $x \wedge y$, is the greatest element in ε that is less than or equal to both x and y . Similarly, the supremum (or join) of x and y , denoted as $x \vee y$, is the smallest element in ε that is greater than or equal to both x and y .

Considering the definition above, in this section we present two special type of intervals, of elements whose domains can be represented as complete lattices, that are used in the next Chapters of this work.

3.6.1 Interval of Sets or Thick Sets

The concept of interval of sets or thick sets was introduced by [101]. This is an important type of interval that will be used in this work in order to characterize uncertain sets as it will be presented later in this document.

When considering sets in Euclidean spaces they can be endowed with a lattice structure that arises from the natural ordering induced by the inclusion relation \subset . Therefore, we can define an interval of elements of the lattice $(\mathcal{P}(\mathbb{R}^n), \subset)$ that is named thick set. The following definition is adapted from [102], to where the readers may refer to for further information on the subject.

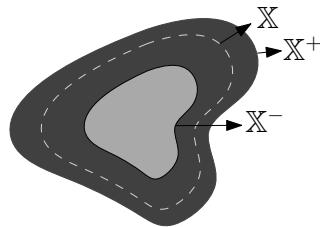


Figure 3.16: Representation of thick sets.

Definition 4 (Thick Set). *We denote $\llbracket \mathbb{X} \rrbracket \in \mathbb{I}\mathcal{P}(\mathbb{R}^n)$ a thick set of \mathbb{R}^n if there are two subsets of \mathbb{R}^n called the lower bound \mathbb{X}^- and the upper bound \mathbb{X}^+ such*

that

$$\llbracket \mathbb{X} \rrbracket = [\mathbb{X}^-, \mathbb{X}^+]$$

$$= \{\mathbb{X} \in \mathcal{P}(\mathbb{R}^n) \mid \mathbb{X}^- \subseteq \mathbb{X} \subseteq \mathbb{X}^+\}$$

A thickset partitions the environment into three zones, the clear zone \mathbb{X}^- , the penumbra $\mathbb{X}^+ \setminus \mathbb{X}^-$ (both illustrated in Figure 3.16) and the dark zone $\mathbb{R}^n \setminus \mathbb{X}^+$.

If for the thick set $\mathbb{X} = [\mathbb{X}^-, \mathbb{X}^+]$, we have $\mathbb{X}^- = \mathbb{X}^+$ then \mathbb{X} is said to be thin. It corresponds to a singleton on $\mathcal{P}(\mathbb{R}^n)$ or a classical subset of \mathbb{R}^n .

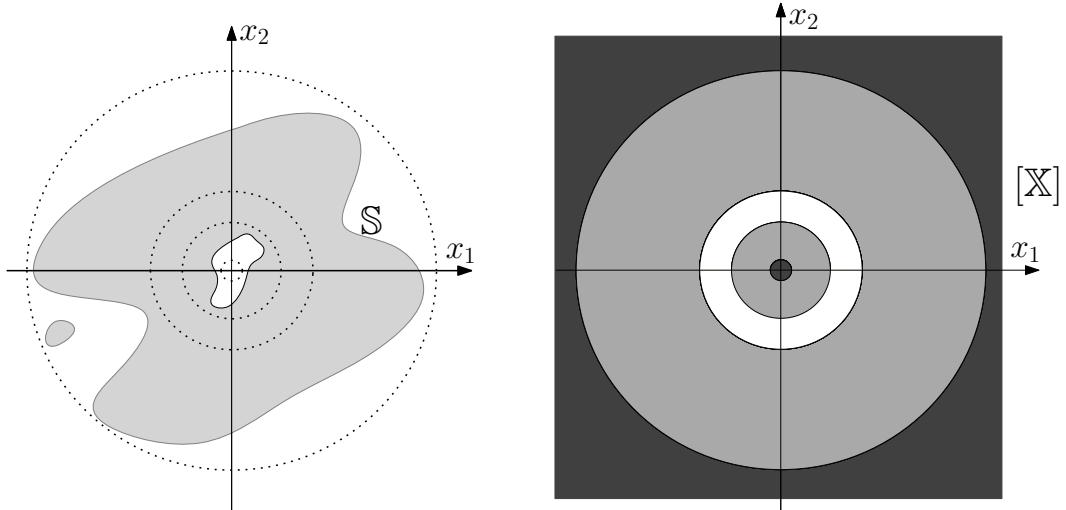


Figure 3.17: On the left we have a disconnected thin set S represented in gray. On the right we have a thick set represented by disks on the planes. The clear zone is illustrated in white, the penumbra in gray and the dark zone in black. We have $S \in [\mathbb{X}]$.

In Figure 3.17 we demonstrate how a thick set can be used to encompass a set and how information about the distribution of this set within the three different zones (clear zone, penumbra and dark zone) can be interpreted. If $S \in [\mathbb{X}]$, it can be concluded that $\mathbb{X}^- \subseteq S$ and therefore that all the points in \mathbb{X}^- are a solution to S . We can also conclude that $\mathbb{X}^+ \cap S \neq \emptyset$ and that a point that belongs to the penumbra $\mathbb{X}^+ \setminus \mathbb{X}^-$ might either be a solution to S or not. Finally, all the points in the dark zone $\mathbb{R}^n \setminus \mathbb{X}^+$ are guaranteed to not be a solution to S .

3.6.2 Interval of Functions or Tubes

In mobile robotics, the inherent uncertainties associated with a robot's pose often require the consideration of a set of possible trajectories to represent

the robot's dynamics during a mission, acknowledging that the true trajectory lies within this set of possibilities. Rather than relying on a single trajectory, by considering a range of potential trajectories, mobile robots can account for variations in sensor measurements, environmental conditions, and other sources of uncertainty, enhancing their ability to navigate and interact effectively in complex and dynamic environments. The robot's trajectory during a mission is a function of time and in this section we formalize the notion of set of functions.

A set of functions is represented by means of tubes. The concept was introduced by [103], and we adopt the definition formally presented in [104] and [105], that allows the use of interval arithmetic within this framework.

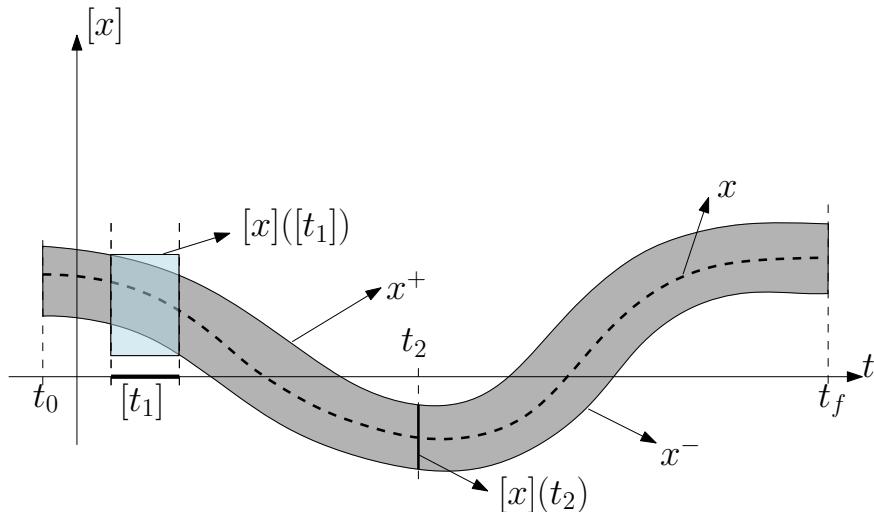


Figure 3.18: One-dimensional tube $[x]$ enveloping function $x : \mathbb{R} \rightarrow \mathbb{R}$.

Let \mathcal{F}^n be the set of all functions from \mathbb{R} to \mathbb{R}^n , \mathcal{F}^n is a complete lattice with partial order $\mathbf{x} \leq \mathbf{y}$ for $\mathbf{x} \in \mathcal{F}^n$ and $\mathbf{y} \in \mathcal{F}^n$, such that $\forall t \in \mathbb{R}, \mathbf{x}(t) \leq \mathbf{y}(t)$ [105]. In this context, we can define an interval of functions $[\mathbf{x}] = [\mathbf{x}^-, \mathbf{x}^+]$ of \mathcal{F}^n such that $\forall t \in [t_0, t_f] \subset \mathbb{R}, \mathbf{x}^-(t) \leq \mathbf{x}^+(t)$. A function $\mathbf{x} \in \mathcal{F}^n$ belongs to a tube $[\mathbf{x}] \in \mathbb{I}\mathcal{F}^n$ if $\forall t \in [t_0, t_f], \mathbf{x}(t) \in [\mathbf{x}](t)$.

The interval evaluation of a tube $[\mathbf{x}]$ over an interval $[t] \in \mathbb{I}\mathbb{R}$ was given in [105] as follows:

$$[\mathbf{x}](t) = [\{\mathbf{x}(t) \mid \mathbf{x} \in [\mathbf{x}], t \in [t]\}]$$

that is the smallest box enclosing all the possible solutions within the domain $[t]$ and considering all the functions enveloped by $[\mathbf{x}]$. In Figure 3.18 we illustrate a one-dimensional tube $[\mathbf{x}] \in \mathbb{I}\mathcal{F}$ and its evaluation over an interval $[t_1]$ and over a single point value t_2 . We know from the illustrated tube $[\mathbf{x}]$, that a function

$x \in [x]$ has at least two different values in $t \in [t_0, t_f]$ such that $[x](t) = 0$. If, additionally, we have another tube $[\dot{x}]$ representing its first derivative, we could also conclude that x has exactly two roots.

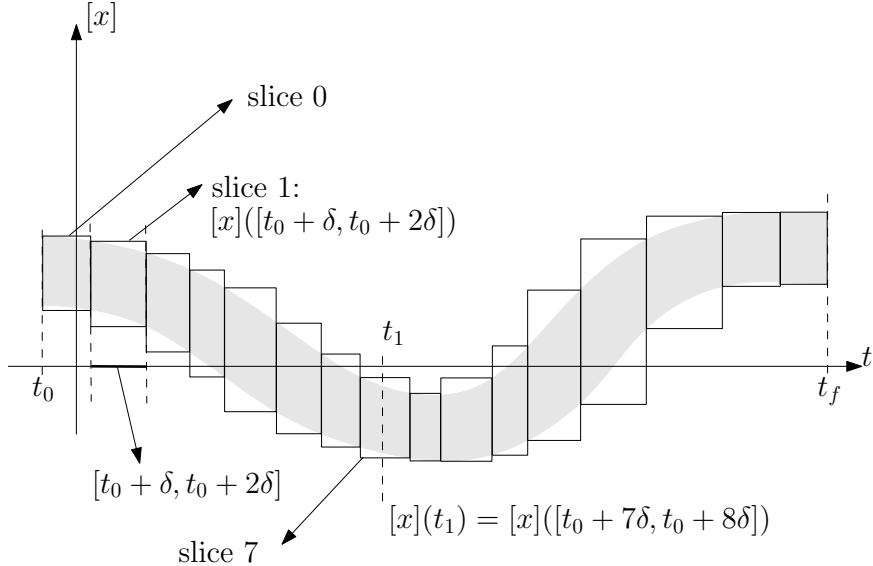


Figure 3.19: Numerically a tube is represented as a sequence of boxes that are called slices of the tube.

Numerically, a tube can be represented as a sequence of boxes that correspond to temporal slices. For that a sampling time $\delta > 0$ is associated to the tube representation and the tube is constant for all $t \in [t_0 + k\delta, t_0 + (k+1)\delta]$, with $k \in \mathbb{Z}$. Meaning that an inspection of the value of $[x]$ at any $t \in [t_0 + k\delta, t_0 + (k+1)\delta]$ will be equal to $[x]([t_0 + k\delta, t_0 + (k+1)\delta])$. The box $[t_0 + k\delta, t_0 + (k+1)\delta] \times [x]([t_0 + k\delta, t_0 + (k+1)\delta])$ is called the k th slice of the tube. In Figure 3.19 the discretization into slices of a one-dimensional tube is illustrated. This implementation results in a piece-wise constant enclosure for the tube, enabling straightforward execution of standard interval operations.

The arithmetic between intervals is naturally extended to tubes. For example, consider $[\mathbf{x}] \in \mathbb{IF}^n$ and $[\mathbf{y}] \in \mathbb{IF}^n$. For an operator $\diamond \in \{+, -, ., /\}$, we will have that

$$[\mathbf{x}] \diamond [\mathbf{y}] = [\{\mathbf{x} \diamond \mathbf{y} \in \mathbb{R}^n \mid \mathbf{x} \in [\mathbf{x}] \text{ and } \mathbf{y} \in [\mathbf{y}]\}]$$

This equation is an extension of Equation 3.2, and it is the smallest tube containing all feasible values for the operation considering the possible combinations between functions enveloped by $[\mathbf{x}]$ and $[\mathbf{y}]$.

Furthermore, if we have a function $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, this function can be applied to a tube $[\mathbf{x}] \in \mathbb{I}\mathcal{F}^n$ and the resultant inclusion function $[\mathbf{f}]$ will be defined as

$$[\mathbf{f}]([\mathbf{x}]) = \{\{\mathbf{f}(\mathbf{x}) \mid \mathbf{x} \in [\mathbf{x}]\}\}$$

In the next section we will be interested on the integral of a tube. For an interval $[t_1, t_2] \subset [t_0, t_f]$, it is the smallest box containing all feasible integrals:

$$\int_{t_1}^{t_2} [\mathbf{x}](\tau) d\tau = \left\{ \int_{t_1}^{t_2} \mathbf{x}(\tau) d\tau \mid \mathbf{x} \in [\mathbf{x}] \right\}$$

and from the monotonicity of the integral function:

$$\int_{t_1}^{t_2} [\mathbf{x}](\tau) d\tau = \left[\int_{t_1}^{t_2} \mathbf{x}^-(\tau) d\tau, \int_{t_1}^{t_2} \mathbf{x}^+(\tau) d\tau \right]$$

We will also be interested in evaluating the integral of tubes over interval domains $[t_1], [t_2] \in \mathbb{IR}$. It was proved in Section 3.3 of [106] that this integral can be computed by:

$$\int_{t_1}^{t_2} [\mathbf{x}](\tau) d\tau = \left[lb(\mathbf{y}^-[t_2]) - \mathbf{y}^-[t_1]), ub(\mathbf{y}^+[t_2]) - \mathbf{y}^+[t_1]) \right]$$

where $[\mathbf{y}](t) = [\mathbf{y}^-, \mathbf{y}^+] = \int_{t_0}^t [\mathbf{x}](\tau) d\tau$ is the interval primitive of $[\mathbf{x}]$ and operators lb and ub return the lower bound and the upper bound of an interval, respectively.

3.7 Detection of Self-Intersections on Tubes

Detecting self-intersections on a tube will play an important role in this work. The problem was the subject of [106] and in this section we present the formalism, concepts and the method introduced by them in order to deal with this problem.

Let \mathbf{x} be a continuous function from \mathbb{R} to \mathbb{R}^n . A self-intersection of \mathbf{x} is determined by two parameters $t_1, t_2 \in \mathbb{R}$, $t_1 \neq t_2$, and it is a point $\mathbf{p} \in \mathbb{R}^n$ such that $\mathbf{p} = \mathbf{x}(t_1) = \mathbf{x}(t_2)$. We are interested in identifying self-intersections along sets of continuous functions in the plane.

Let $\mathbf{x} : [t_0, t_f] \rightarrow \mathbb{R}^2$ be a continuous function, for some interval $[t_0, t_f] \subset \mathbb{R}$. In [106] a pair defining a self-intersection is called a t-pair. We are interested in computing the following set, of all t-pairs that characterize self-intersections:

$$\mathbb{T}^* = \{(t_1, t_2) \in [t_0, t_f]^2 \mid \int_{t_1}^{t_2} \dot{\mathbf{x}}(\tau) d\tau = 0, t_1 < t_2\} \quad (3.6)$$

Where $\dot{\mathbf{x}}$ represents the first derivative of \mathbf{x} . With the intention of simplifying the notation, from now on, in this section, we will be using $\mathbf{v} = \dot{\mathbf{x}}$.

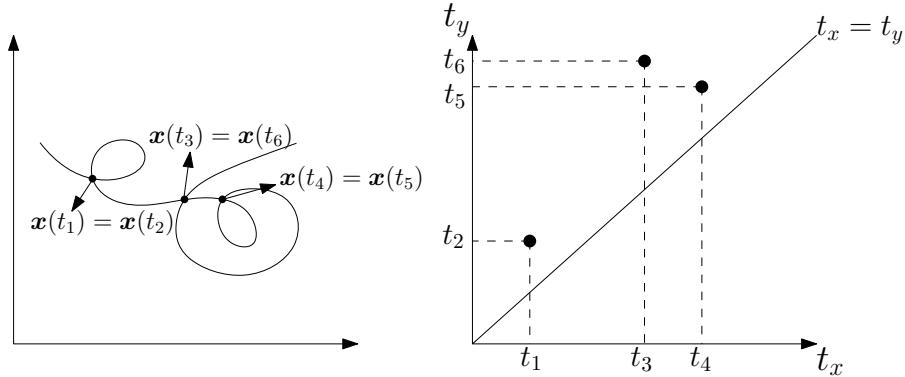


Figure 3.20: Right: curve with three self-intersections (t_1, t_2) , (t_4, t_5) and (t_3, t_6) . Note that $[t_4, t_5] \subset [t_3, t_6]$, therefore $\mathbf{x}([t_3, t_6])$ is not a simple cycle; Left: t-plane, with the three self-intersections identified.

A t-pair can be graphically represented in a t-plane. In Figure 3.20 an example of t-plane is given considering the set $\mathbb{T}^* = \{(t_1, t_2), (t_3, t_6), (t_4, t_5)\}$ for a function \mathbf{x} that is also illustrated.

In practice, we will be dealing with an unknown function \mathbf{v}^* guaranteed to be represented inside a tube $[\mathbf{v}]$. As a consequence, we will not be able to estimate the set \mathbb{T}^* directly as defined in Equation (3.6). Instead, we estimate the set \mathbb{T} that contains all the possible self-intersections from all the feasible curves that we obtain by integrating $[\mathbf{v}]$:

$$\mathbb{T} = \{(t_1, t_2) \in [t_0, t_f]^2 \mid \exists \mathbf{v} \in [\mathbf{v}], \int_{t_1}^{t_2} \mathbf{v}(\tau) d\tau = 0, t_1 < t_2\} \quad (3.7)$$

Since \mathbf{v}^* is guaranteed to lie within $[\mathbf{v}]$, we also have that $\mathbb{T}^* \subseteq \mathbb{T}$. Finally, the problem of detecting self-intersections within a tube amounts to computing the set \mathbb{T} and that is a typical set-inversion problem. Algorithm 3 is a self-membership algorithm to identify self-intersections along tubes. It is a SIVIA algorithm, as presented in Section 3.5.1, that applies some tests to determine if a t-box $[\mathbf{t}] \in \mathbb{IR}^2$, that is a set of t-pairs, belongs or not to the solution set \mathbb{T} . These tests, that are now presented in this document, were proposed by [106] to where we also refer the reader for the proofs of what follows.

Testing if $[\mathbf{t}]$ does not contain any possible solution: If $[\mathbf{t}]$ is not a solution, it means that it does not exist $\mathbf{t} \in [\mathbf{t}]$ such that $\mathbf{t} \in \mathbb{T}$. Let $\mathbf{t} = [t_1] \times [t_2]$ be the t-box to be evaluated, we are only interested in t-pairs (t_1, t_2) for which

$t_1 < t_2$. Therefore, if $[t_1] - [t_2] \subset \mathbb{R}^+$, the box $[\mathbf{t}]$ does not contain any solution to the problem and it can be discarded. Additionally, we can say that,

$$\mathbf{0} \notin \int_{[t_1]}^{[t_2]} [\mathbf{v}](\tau) d\tau \Rightarrow [\mathbf{t}] \cap \mathbb{T} = \emptyset$$

It implies that if there is no $\mathbf{v} \in [\mathbf{v}]$ or $\mathbf{t} \in [\mathbf{t}]$ that leads to $\int_{t_1}^{t_2} \mathbf{v}(\tau) d\tau = \mathbf{0}$, the t-box $[\mathbf{t}]$ can also be discarded. These conditions are enough for rejecting a t-box with no solution within as long as $[t_1] \cap [t_2] = \emptyset$, on the contrary, another test must be applied.

If $[t_1] \cap [t_2] \neq \emptyset$, $[\mathbf{t}]$ will pass the last two tests even if it does not contain any possible solution. This happens because in this context $\exists t \in [t_1]$ and $\exists t \in [t_2]$ such that $\int_t^t [\mathbf{v}](\tau) d\tau = \mathbf{0}$ and naturally $[t_1] - [t_2] \not\subset \mathbb{R}^+$. In this case, the injective test is introduced, it consists of proving that functions in $[\mathbf{x}]$, for which we are computing the self-intersections, are injective in $[t_1^-, t_2^+]$. This can be verified by the following test:

$$\mathbf{0} \notin [\mathbf{v}]([t_1^-, t_2^+]) \Rightarrow [\mathbf{t}] \cap \mathbb{T} = \emptyset$$

Finally, these three tests have been proven to be robust for eliminating t-boxes without discarding any possible solution to the problem.

Testing if $[\mathbf{t}]$ is a subset of the solution set: We can also apply some tests in order to verify if a t-box $[\mathbf{t}]$ is a subset of the solution set. First, we know that this will only be true if the condition $[t_1] - [t_2] \subset \mathbb{R}^-$ is met. Next, we need to determine if $\forall \mathbf{t} = (t_1, t_2) \in [\mathbf{t}], \exists \mathbf{v} \in [\mathbf{v}]$ such that $\int_{t_1}^{t_2} \mathbf{v}(\tau) d\tau = \mathbf{0}$. This condition can be restated using the intermediate value theorem:

$$\int_{[t_1]}^{[t_2]} \mathbf{v}^-(\tau) d\tau \leq \mathbf{0} \leq \int_{[t_1]}^{[t_2]} \mathbf{v}^+(\tau) d\tau \Rightarrow [\mathbf{t}] \subset \mathbb{T}$$

where

$$\int_{[t_1]}^{[t_2]} \mathbf{v}^-(\tau) d\tau = \left\{ \int_{t_1}^{t_2} \mathbf{v}^-(\tau) d\tau \mid t_1 \in [t_1] \text{ and } t_2 \in [t_2] \right\}$$

and if $\mathbb{A} \subset \mathbb{R}^2$, the comparison $\mathbb{A} \leq \mathbf{0}$ means that $\forall \mathbf{a} = (a_1, a_2) \in \mathbb{A}, a_1 \leq 0$ and $a_2 \leq 0$.

Finally, if after the tests stated above, it is concluded that the t-box is not a subset of \mathbb{T} but it can not be discarded either, the box is bisected if it has a minimum width or it is added to the outer approximation set. The complete procedure is presented in Algorithm 3.

Algorithm 3 Detecting self-intersections on tubes (in: $[\mathbf{v}]$, $[t_0, t_f]$, ϵ , out: $\mathbb{T}^-, \mathbb{T}^+$)

```

1: Initialize:
     $\mathbb{Q} = \{[t_0, t_f]^2\}$ 

2: if  $\mathbb{Q} \neq \emptyset$  then take a t-box  $[\mathbf{t}] \in \mathbb{Q}$  and remove it from  $\mathbb{Q}$  else return
endif

3: if  $[t_1] - [t_2] \subset \mathbb{R}^+$  or  $\mathbf{0} \notin \int_{[t_1]}^{[t_2]} [\mathbf{v}](\tau) d\tau$  or  $\mathbf{0} \notin [\mathbf{v}](t_1^- - t_2^+)$  then
     $\triangleright [\mathbf{t}]$  has no solutions

4: elseif  $[t_1] - [t_2] \subset \mathbb{R}^-$  and  $\int_{[t_1]}^{[t_2]} \mathbf{v}^-(\tau) d\tau \leq \mathbf{0} \leq \int_{[t_1]}^{[t_2]} \mathbf{v}^+(\tau) d\tau$  then
    5:    $\mathbb{T}^+ \leftarrow \mathbb{T}^+ \cup [\mathbf{t}]$                                  $\triangleright$  outer approximation set
    6:    $\mathbb{T}^- \leftarrow \mathbb{T}^- \cup [\mathbf{t}]$                                  $\triangleright$  inner approximation set

7: elseif  $w([\mathbf{t}]) < \epsilon$  then
8:    $\mathbb{T}^+ \leftarrow \mathbb{T}^+ \cup [\mathbf{t}]$                                  $\triangleright$  outer approximation set

9: else
10:   bisect  $[\mathbf{t}]$  into  $[\mathbf{t}]^{(1)}$  and  $[\mathbf{t}]^{(2)}$ 
11:    $\mathbb{Q} \leftarrow \mathbb{Q} \cup \{[\mathbf{t}]^{(1)}, [\mathbf{t}]^{(2)}\}$ 

endif

12: Return to 2.

```

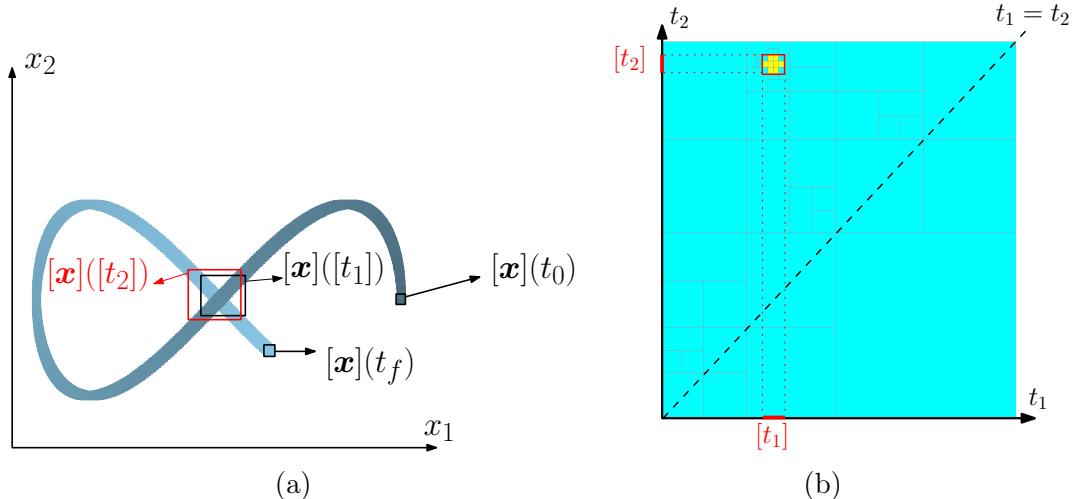


Figure 3.21: Images created using the library Codac [99]. (a): A tube $[\mathbf{x}] \in \mathbb{I}\mathcal{F}^2$ represented in blue with one self-intersection. (b) : Paving of the t-plane to find a self-intersection on $[\mathbf{x}]$. The box $[t_1] \times [t_2]$ is the smallest box containing the outer-approximation, represented by the union of the yellow boxes, of t-pairs representing a possible self-intersection.

In Figure 3.21 we have an example of the application of the algorithm just presented to a tube where one self-intersection can be detected. Each self-intersection is represented by a connected set and if many self-intersections are detected the set \mathbb{T} will be disconnected.

3.8 Conclusion

In conclusion, this chapter presented interval analysis, a powerful mathematical tool for handling uncertainty and approximating solutions to complex problems. The theory and concepts covered here lay the groundwork for the forthcoming chapters of this document. After recalling concepts from set theory, we explored the fundamental concepts of interval arithmetic, interval enclosures, and the representation of functions using intervals. The use of contractors and separators have been discussed as essential operators for efficiently propagating constraints and refining approximations. Moreover, we have explored the challenges and benefits of subpaving, which serves as a valuable technique for reducing pessimism and enhancing accuracy in interval-based representations. Additionally, we examined the set inversion problem via interval analysis and we introduced pavers, presenting algorithms that will be used in different contexts throughout this document. Finally, two important dedicated sets, that are going to be employed to characterize uncertainty in this work, were presented.

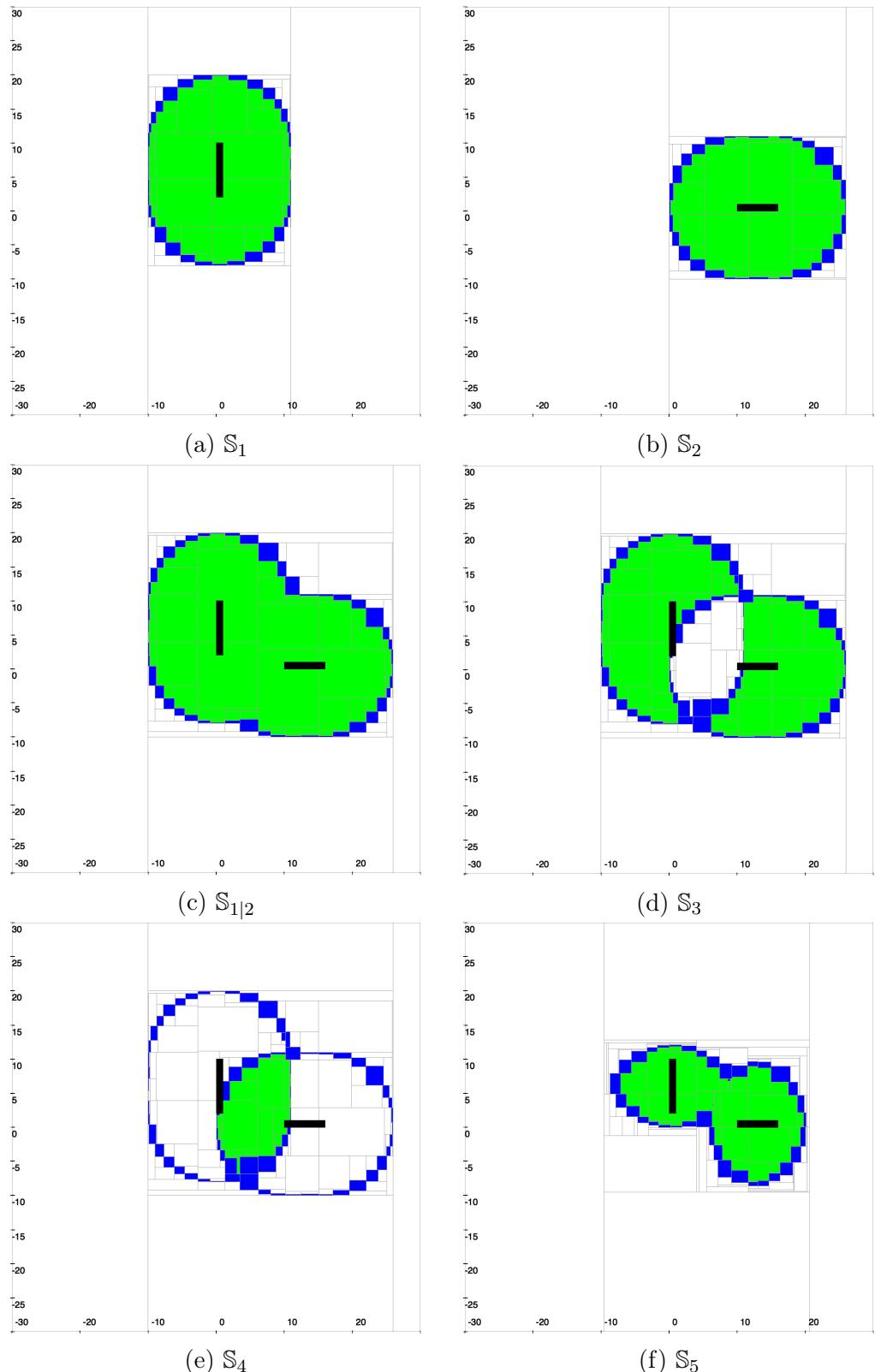


Figure 3.15: Green boxes represent inner approximations of the solution set and blue boxes outer approximations. Boxes in white do not contain points that might be a solution to the problem and black boxes represent $[p_1]$ and $[p_2]$.

Chapter 4

Topology

“Mathematicians are indifferent to the replacement of objects by others as long the relations don’t change. Matter is not important, only form interests them.”-

Henri Poincaré

Contents

4.1	Common Euclidean Subspaces	81
4.2	Topological Spaces	82
4.2.1	Common Examples of Topologies	84
4.2.2	Representation of Arbitrary Sets on Topological Spaces .	84
4.2.3	Convergence	86
4.2.4	Continuity	87
4.2.5	Basis for a Topology	88
4.3	Interesting Topological Spaces	89
4.3.1	Hausdorff Spaces	89
4.3.2	Manifolds	91
4.3.3	Quotient Spaces	94
4.4	Connectedness and Compactness	96
4.4.1	Connected Space	96
4.4.2	Compact Space	97
4.4.3	Compactification	98
4.5	The Fundamental Group	99
4.5.1	Groups and Isomorphism	99
4.5.2	Homotopy	100
4.5.3	Homotopy Equivalence	101
4.5.4	Construction of the Fundamental Group	102
4.6	CW Complexes	104

4.7	Homology	105
4.7.1	Homology Group	106
4.8	Topological Degree	108
4.8.1	Winding Number	109
4.9	Conclusion	110

Topology emerged as a branch of mathematics born out of the necessity to categorize and classify geometric objects. Briefly stated, topology deals with properties and characteristics of sets that remain unchanged under continuous deformations, such as stretching, bending, twisting, and shrinking, but without tearing or gluing.

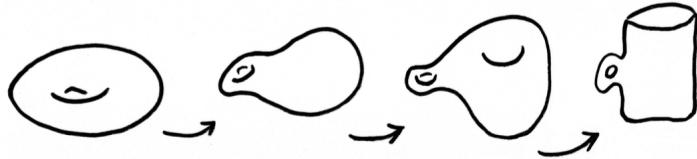


Figure 4.1: A topological transformation of a doughnut into a coffee cup. Image extracted from [107].

In topology, what matters is the underlying structure and the relation of closeness between points in a space, rather than their specific geometric appearance. This concept often leads to intriguing and sometimes counterintuitive results. For example, there is a famous mathematical joke that says that a topologist is someone who cannot distinguish between a doughnut and a coffee mug. Although they may look different as physical objects, from a topological perspective, they are "the same". By the same we mean that one object can be transformed into the other in a continuous and reversible manner without tearing or gluing. Figure 4.1 illustrates this transformation.

In this Chapter, we will introduce fundamental concepts from topology that will serve as the building blocks of the following sections of this document providing a solid mathematical framework. Furthermore, these topological tools will prove invaluable for the mathematical formalization presented ahead in this work, and they also play a central role in the development of a solution for the problem at hand.

The content presented here has only an introductory character and it is mostly based on the following text books [108], [109], [110], [111], [112].

4.1 Common Euclidean Subspaces

Before presenting the concept of topological spaces we find essential to first define some common Euclidean spaces that often serve as foundational examples:

Unit Interval $I \subset \mathbb{R}$ is the unit interval.

$$I = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$$

Unit Ball $B^n \subseteq \mathbb{R}^n$ is the unit ball in \mathbb{R}^n .

$$B^n = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\| < 1\}$$

Unit n-sphere $S^n \subseteq \mathbb{R}^{n+1}$ is the unit n-sphere.

$$S^n = \{\mathbf{x} \in \mathbb{R}^{n+1} \mid \|\mathbf{x}\| = 1\}$$

In \mathbb{R}^2 , S^1 is also named the unit circle.

Unit Disk $D^n \subseteq \mathbb{R}^{n+1}$ is the unit disk, it will also be referred to as a closed ball.

$$D^n = B^{n+1} \cup S^n$$

4.2 Topological Spaces

In this work, our focus remains exclusively on Euclidean spaces. However, within this chapter, we introduce concepts from topology at a more abstract level, considering broader spaces referred to as topological spaces. The intention is to provide a deeper understanding of topological tools and to illustrate their extensive applicability by introducing them within a more general context.

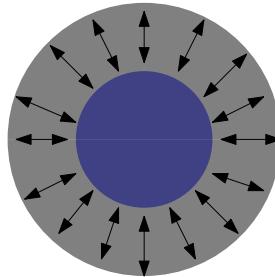


Figure 4.2: Any two opens balls in \mathbb{R}^n can be continuously deformed onto the other by radial retraction. This transformation does not preserve the distance between points in the space.

On the introduction of this Chapter we mentioned that two different objects that can be transformed onto another through a continuous deformation

are considered equivalent in topology, the coffee mug and the doughnut example were used for illustrating this idea. Another example, that is simpler, is the transformation between two balls of different radii, Figure 4.2. In this transformation, it is obvious that the continuous deformation between these two metric spaces, in any direction, will not preserve the distance between points [109]. Therefore, in order to establish equivalence between spaces from a topological point of view, the concept of distance is irrelevant, and it does not have to be well-defined.

In this context, while metric spaces rely on the notion of distance, a topological space seeks to remove this specific metric structure and focus solely on the closeness and continuity between points. Hence, the main motivation behind the definition of a topological space is to provide a generalization of the concept of a metric space, advancing in abstraction and enabling an exploration of more general and flexible mathematical structures.

In a metric space, distances between points are explicitly defined, and open sets are constructed based on these distances. In order to define a topology on a set we designate certain subsets of that space to be open. This collection of subsets need to satisfy three axioms that are presented on the following definition, adapted from [113]:

Definition 5 (Topological Space). *A topological space is a pair (X, T) consisting of a set X and a topology T on X . The topology T is a collection of subsets of X , called open sets, that satisfy the following properties:*

- (i) *X and \emptyset are elements of T , or we can say that X and \emptyset are open;*
- (ii) *The intersection of a finite collection of open sets is open.*
If $U_1, \dots, U_n \in T$ then their intersection $U_1 \cap \dots \cap U_n$ is in T ;
- (iii) *The union of any arbitrary collection of open sets is open.*
If C is any finite or infinite collection of elements in T , then the union of its elements $\bigcup_{c \in C} c$ is in T .

A subset $S \subseteq X$ is said to be closed in (X, T) if its complement $X \setminus S$ is an open set, in other words, if $X \setminus S \in T$. According to this definition a set can be both open and closed, for example X and \emptyset . The three axioms in Definition 5 can be rewritten in terms of closed sets:

- (i) X and \emptyset are closed;
- (ii) Finite unions of closed sets are closed;
- (iii) Arbitrary intersections of closed sets are closed.

4.2.1 Common Examples of Topologies

Some common examples of topologies are:

Discrete Topology: The discrete topology on X is the power set of X , $T = \mathcal{P}(X)$. All possible subsets of X are open sets and it is the finest topology.

Trivial Topology: The trivial topology is the coarsest topology on X , it has the minimum number of subsets of X in order to respect the three axioms established by definition. In this case, $T = \{X, \emptyset\}$.

Example: Let us consider set $X = \{1, 2, 3\}$. Its discrete topology is $T = \{\{1, 2, 3\}, \emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$, its trivial topology is $T = \{\{1, 2, 3\}, \emptyset\}$ and another example of possible topology would be $T = \{\{1, 2, 3\}, \emptyset, \{1\}, \{1, 2\}\}$. It can be easily verified that these three propositions of collections respect the three axioms presented in Definition 5.

The subsequent definitions, taken from [112], present two topologies that hold significant relevance and are frequently encountered in various mathematical contexts.

Definition 6 (Quotient Topology). *Let X be a topological space, Y a set and the map $q : X \rightarrow Y$ is a surjective function.*

The quotient topology on Y is defined by setting $U \subseteq Y$ to be open if and only if $q^{-1}(U)$ is open in X .

We discuss more about spaces with a quotient topology in Section 4.3.3.

Metric spaces are topological spaces with the metric topology.

Definition 7 (Metric Topology). *Let (M, d) be any metric space equipped with distance function d . For the metric topology, T will be a collection of open sets in the usual metric space sense. Therefore, we have $O \in T$ if for every $\mathbf{x} \in O$ there is a real positive value ϵ such that $B_\epsilon(\mathbf{x}) \subseteq O$. Where, $B_\epsilon(\mathbf{x})$ is an open ball of radius ϵ centered in \mathbf{x} or $B_\epsilon(\mathbf{x}) = \{\mathbf{y} \in M | d(\mathbf{x}, \mathbf{y}) < \epsilon\}$.*

4.2.2 Representation of Arbitrary Sets on Topological Spaces

In a topological space we have a topology that is a collection of open sets of the space. Closed sets are the complement of these open sets. If we have some arbitrary subset of the original space, not necessarily closed or open, we might be interested in representing it by an open or closed set of the topological space. This can be obtained with the closure and the interior of the arbitrary subset.

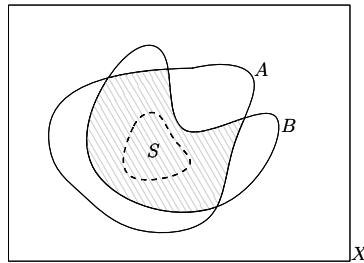


Figure 4.3: Let (X, T) be a topological space where $\{X, \emptyset, A, B, A \cup B, A \cap B\}$ are closed sets. The closure of S is the result of the intersection of all the closed sets of (X, T) that contain S and it is represented by the striped area.

The closure is the smallest over approximation of a set by closed sets. Equivalently, we have the following definition:

Definition 8 (Closure). *Let (X, T) be a topological space and S is a subset of X , $S \subseteq X$. The closure of S in X is the set*

$$\bar{S} := \bigcap \{A \subseteq X \mid S \subseteq A \text{ and } X \setminus A \in T\}$$

The closure of the arbitrary subset S is the intersection of all the closed sets A that contain S . This notion is illustrated in Figure 4.3.

Since the intersection of an arbitrary collection of closed sets is also a closed set, the closure of a subset S is the smallest closed set of the topological space that contain S .

The interior is an inner approximation of a set by open sets.

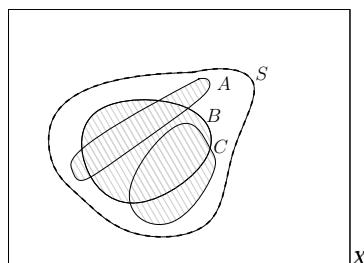


Figure 4.4: Let (X, T) be a topological space where A , B and C are open sets. The interior of S is the result of the union of all the open sets of (X, T) that are subsets of S . In this example $\overset{\circ}{S} = A \cup B \cup C$ and is represented by the striped area.

Definition 9 (Interior). *Let (X, T) be a topological space and S is a subset of X , $S \subseteq X$. The interior of S in X is the set*

$$\mathring{S} := \bigcup\{A \subseteq X \mid A \subseteq S \text{ and } A \in T\}$$

The interior of the arbitrary subset S is the union of all the open sets A that are a subset of S . This notion is illustrated in Figure 4.4.

Since the union of an arbitrary collection of open sets is open, the interior of a set S is the biggest set that is open and a subset of S .

The definition of interior can be used to define the neighborhood of a point.

Definition 10 (Neighborhood). *If (X, T) is a topological space and $\mathbf{p} \in X$, then a neighborhood of \mathbf{p} is a subset $S \subseteq X$ such that there exists an open set $O \in T$ containing \mathbf{p} , where O is entirely contained within S .*

In other words, $S \subseteq X$ is a neighborhood of $\mathbf{p} \in X$ if $\mathbf{p} \in \mathring{S}$.

The exterior and the boundary of an arbitrary subset can also be defined.

Definition 11 (Exterior). *Let (X, T) be a topological space and S is a subset of X , $S \subseteq X$. The exterior of S in X can be defined as the complement of its closure:*

$$\text{Ext } S := X \setminus \overline{S}$$

The exterior of a set is an open set.

Definition 12 (Boundary). *Let (X, T) be a topological space and S is a subset of X , $S \subseteq X$. The boundary of S in X can be defined as:*

$$\partial S := X \setminus (\mathring{S} \cup \text{Ext } S)$$

4.2.3 Convergence

On a metric space convergence refers to the behavior of a sequence of elements in that space as the sequence gets longer. Let $\{\mathbf{x}_n\}$ be a sequence of elements in the metric space (M, d) . The sequence is said to converge to a point $\mathbf{x} \in M$ if, for any positive real number $\epsilon > 0$, there exists a positive integer N such that for all $n \geq N$, the distance between \mathbf{x}_n and \mathbf{x} , given by $d(\mathbf{x}_n, \mathbf{x})$, is less than ϵ . In other words, as the sequence gets longer, the points of the sequence become close to \mathbf{x} .

In topological spaces, the notion of "close to" is generalized using neighborhoods. And convergence is defined through the two definitions below, respectively from [113] and [109].

Definition 13 (Limit Point or Accumulation Point). *Let (X, T) be a topological space and S is a subset of X , $S \subseteq X$. A point \mathbf{q} is a limit point of S if every neighborhood of \mathbf{q} contains a point of S other than \mathbf{q} itself.*

All points of X are either limit points or isolated.

Definition 14 (Convergent Sequence). *Let (X, T) be a topological space and $\{\mathbf{x}_n\}$ is a sequence of points in X . For $\mathbf{x} \in X$, we say that the sequence converges to the limit \mathbf{x} , if for every neighborhood S of \mathbf{x} there exists a positive integer N such that $\mathbf{x}_i \in S$ for all $n > N$.*

4.2.4 Continuity

Now we define the central concept in topology: continuous maps between topological spaces.

The open set criterion for continuity shows that continuous functions between metric spaces can be detected knowing only the open sets [109]. For example, let \mathbf{f} be a function between two Euclidean spaces, say $\mathbf{f} : \mathbb{E}^m \rightarrow \mathbb{E}^n$. One can say that \mathbf{f} is continuous if given any $\mathbf{x} \in \mathbb{E}^m$ and any open set O that contains $\mathbf{f}(\mathbf{x})$ in \mathbb{E}^n , then $\mathbf{f}^{-1}(O)$ is open.

In topological spaces continuous functions can still be defined without explicitly relying on distances. Continuous functions in topology preserve the topological structure of spaces, ensuring that points that are close together in the domain are mapped to points that are close together in the codomain.

Let (X, T_x) and (Y, T_y) be topological spaces, from now we omit their topology from the notation, that should be understood from the context. Therefore, we will just simply say that X and Y are topological spaces.

A map $\mathbf{f} : X \rightarrow Y$ is continuous if for every open set $O \subset Y$, $\mathbf{f}^{-1}(O)$ is open in X . The following lemmas were extracted from [109] and define some properties of continuous maps.

Lemma 4.2.1. *Let X, Y and Z be topological spaces.*

- (i) *Any constant map $\mathbf{f} : X \rightarrow Y$ is continuous.*
- (ii) *The identity map $Id : X \rightarrow X$ is continuous.*
- (iii) *If $\mathbf{f} : X \rightarrow Y$ is continuous and $S \subseteq X$ is open then $\mathbf{f}|_S$ is also continuous.*
- (iv) *If $\mathbf{f} : X \rightarrow Y$ and $\mathbf{g} : Y \rightarrow Z$ are continuous, then $\mathbf{g} \circ \mathbf{f} : X \rightarrow Z$ is also continuous.*

Lemma 4.2.2 (Local Criterion for Continuity). *A function $\mathbf{f} : X \rightarrow Y$ is continuous \iff each point of X has a neighborhood on which \mathbf{f} is continuous.*

Definitions previously presented on this Chapter are enough for proving all the properties defined on Lemmas 4.2.1 and 4.2.2.

Now that the concept of a continuous function on topological spaces is defined we can define homeomorphism.

Definition 15 (Homeomorphism). *A homeomorphism between topological spaces X and Y is a continuous function $f : X \rightarrow Y$ which has a continuous inverse $f^{-1} : Y \rightarrow X$.*

If such a function exists, we can say that X and Y are homeomorphic and we write $X \simeq Y$.

In topology being homeomorphic is an equivalence relation. Therefore, any space is homeomorphic to itself by the identity function and it is a transitive property, meaning that if $X \simeq Y$ and $Y \simeq Z$ then $X \simeq Z$ as well.

A property of a topological space that is invariant under homeomorphisms is called a topological property or topological invariant.

Let (X, T_X) and (Y, T_Y) be topological spaces and let $f : X \rightarrow Y$ be bijective. Then f is a homeomorphism if and only if for every $O \in T_X \iff f(O) \in T_Y$

In this chapter we already mentioned two examples of homeomorphisms illustrated in Figures 4.1 and 4.2. Other examples are:

1. Any two closed balls in \mathbb{R}^n are homeomorphic,
2. An open ball in \mathbb{R}^n is homeomorphic to \mathbb{R}^n ,
3. The cube $C = \{(x, y, z) \mid \max(x, y, z) = 1\}$ is homeomorphic to the 2-sphere S^2 .
4. Any *Jordan curve* is homeomorphic to S^1 .

A *Jordan curve* is a non-self-intersecting, continuous, and closed curve in the plane. It divides the plane into two distinct regions, an inside region bounded by the curve and an outside region that extends to infinity.

4.2.5 Basis for a Topology

Let X be a topological space. A collection $\mathcal{B} \in \mathcal{P}(X)$ is called a basis for X if:

- (i) Every $B \in \mathcal{B}$ is open;
- (ii) Every open subset of X is the union of elements in \mathcal{B} .

Since X is open we can conclude that the basis covers X with its elements.

For example, considering the definition of open sets on the topology induced by the metric on a metric space, presented in Definition 7, open balls form a basis for the topology on metric topologies. For a set endowed with the discrete topology, the collection of all singletons will be a basis.

The notion of basis can be used to bound the size of a topology on some space, and the size of a topology is related to the notion of countability. In this thesis we are going to be interested on topological spaces that are second countable because they are important for the definition of manifolds, discussed in Section 4.3.2. By definition, from [114], we have :

Definition 16 (Second Countable). *A topological space is called second countable if at least one of the possible basis for its topology is countable.*

Every Euclidean Space \mathbb{R}^n is second countable.

4.3 Interesting Topological Spaces

In this section we define some important types of topological spaces that will be mentioned in this document.

4.3.1 Hausdorff Spaces

One of the key questions in topology is how points in a space can be separated and in this context the separation axioms are a set of axioms that determine the level of "closeness" and "separation" between points in a topological space. We mentioned that the notion of distance is not necessarily defined for topological spaces, therefore we are going to generalize this notion using open sets. The separation axioms specify conditions under which distinct points in a space can be separated by open sets and help classify different types of topological spaces based on their separation properties. The two most commonly encountered separation axioms are the T0 separation axiom and the T1 separation axiom.

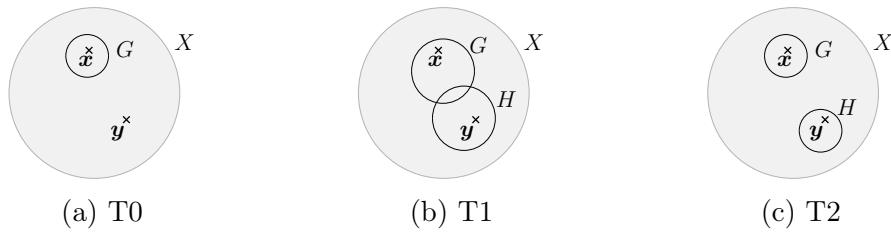


Figure 4.5: Example of separation axioms for a topological space X .

T0 A topological space X satisfies the T0 separation axiom if for any pair of distinct points \mathbf{x} and \mathbf{y} in X , there exists an open set $G \subseteq X$ containing one of the points but not the other. In other words, distinct points can be distinguished by at least one open set. Formally, for every $\mathbf{x} \neq \mathbf{y}$ in X , there exists an open set G such that either $\mathbf{x} \in G$ and $\mathbf{y} \notin G$ or $\mathbf{y} \in G$ and $\mathbf{x} \notin G$. Let $X = \{1, 2, 3\}$ be a topological space with topology $T = \{X, \emptyset, \{1\}, \{2\}, \{1, 2\}\}$ then X is T0 because:

1. for 1 and 2: there exists $\{1\}$ such that $1 \in \{1\}$ and $2 \notin \{1\}$,
2. for 1 and 3: there exists $\{1\}$ such that $1 \in \{1\}$ and $3 \notin \{1\}$,
3. for 2 and 3: there exists $\{2\}$ such that $2 \in \{2\}$ and $3 \notin \{2\}$.

The main idea is illustrated in Figure 4.5a.

T1 A topological space X satisfies the T1 separation axiom if for any pair of distinct points \mathbf{x} and \mathbf{y} in X , there exists open sets containing each point separately. In other words, every point in the space has its own neighborhood. Formally, for every $\mathbf{x} \neq \mathbf{y}$ in X , there exists open sets G and H such that $\mathbf{x} \in G$, $\mathbf{y} \notin G$, $\mathbf{y} \in H$, and $\mathbf{x} \notin H$. The main idea is illustrated in Figure 4.5b. A T1 space is always T0, but the converse is not necessarily true.

To illustrate the previous definitions we consider set $X = \{1, 2, 3\}$ with topology

$$T = \{\{1\}, \{1, 2\}, \{1, 2, 3\}, \emptyset\}$$

This space presents two characteristics that usually make the definition of topological spaces counterintuitive if we are used to working with the topology on Euclidean spaces. They are:

- (i) Each individual element of X is not necessarily closed. For example, $\{1\}$ is not closed because its complement $\{2, 3\}$ is not open. Individual points are always closed in Euclidean spaces.
- (ii) Let us consider a constant sequence set in X , for example, $(2, 2, 2, \dots)$. By intuition the limit of this sequence should be 2 but according to Definition 14, this sequence will have two different limits in X : 2 and 3. Therefore, the limit of the sequence is not unique. Point 3 is considered a limit of the sequence because there is only one neighborhood around 3, containing the whole space, that is so big that it does not allow to distinguish the behavior of the sequence. This ambiguity could be solved for sequence $(2, 2, 2, \dots)$ if we set $\{1, 3\}$ to also be open, for example.

The topological space considered satisfies the T0 separation axiom but not T1. However, a space satisfying the T1 separation axiom is not guaranteed to avoid these two undesired characteristics either. For example, if \mathbf{x} and \mathbf{y} are distinct points in a T1 space, there are open neighborhoods around \mathbf{x} , N_x and around \mathbf{y} , N_y such that $\mathbf{x} \notin N_y$ and $\mathbf{y} \notin N_x$. It is possible, however, that $N_x \cap N_y \neq \emptyset$ for all N_x, N_y . Thus, a sequence $\{\mathbf{x}_n\}$ can converge to both \mathbf{x} and \mathbf{y} if there exists two positive integers I_x and I_y such that for $n > I_x$, $\mathbf{x}_n \in N_x$ and for $n > I_y$, $\mathbf{x}_n \in N_y$, then for $n > \max(I_x, I_y)$, \mathbf{x}_n is in both N_x and N_y , which is allowed in T1 spaces.

To solve this a new separation axiom named T2, or the Hausdorff separation axiom, was defined.

T2 A topological space satisfies the T2 separation axiom, commonly known as a Hausdorff space, if for any pair of distinct points \mathbf{x} and \mathbf{y} in the X , there exist disjoint open sets containing each point separately. In other words, every pair of distinct points can be separated by disjoint open sets. Formally, for every $\mathbf{x} \neq \mathbf{y}$ in X in the space, there exist open sets G and H such that $\mathbf{x} \in G$, $\mathbf{y} \notin G$, $\mathbf{y} \in H$, $\mathbf{x} \notin H$, and $G \cap H = \emptyset$. Metric spaces and discrete topological spaces are examples of Hausdorff spaces.

In a Hausdorff space, the limit of a convergent sequence is unique. This property is particularly useful in analysis and other areas where convergence is important. The uniqueness of limits simplifies proofs and makes it easier to reason about the behavior of sequences in these spaces.

The concept of a Hausdorff space introduced here is next employed to define an important kind of topological space which are manifolds. Later in this document, manifolds will be a condition to define the area established by the range of visibility of the sensors employed by the robo during exploration.

4.3.2 Manifolds

Manifolds are topological spaces that offer a rich and versatile mathematical framework that arises naturally in different contexts. One of the most compelling aspects of manifolds is their local Euclidean structure. Locally, they resemble Euclidean spaces, making them amenable to analysis and providing an intuitive geometric understanding. This local linearity allows for the use of calculus and differential equations on manifolds, facilitating the study of smooth functions and geometric properties. Despite their local simplicity, manifolds can possess global topological and geometric characteristics that can be highly nontrivial. Notions and definitions presented in this Section are adapted from [111].

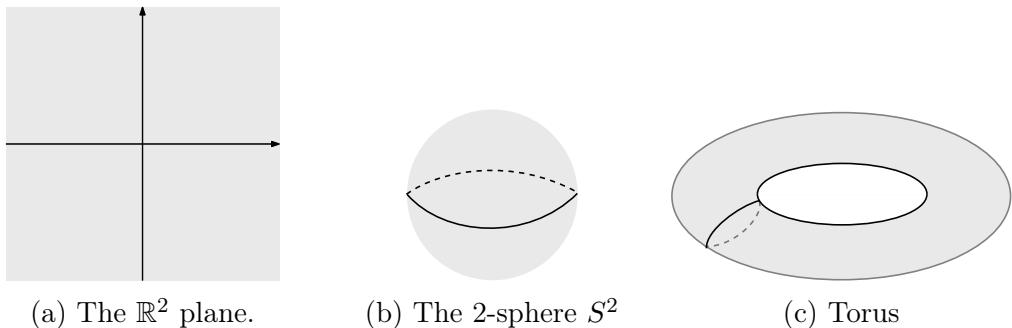


Figure 4.6: Examples of 2-Dimensional Manifolds.

A space M is locally Euclidean of dimension n if every point $x \in M$ has a neighborhood homeomorphic to an open ball $B^n \subseteq \mathbb{R}^n$. Considering this, a formal definition of a manifold can be presented:

Definition 17 (Topological Manifold). *An n -dimensional topological manifold is a second countable Hausdorff space that is locally Euclidean of dimension n .*

Figure 4.6 illustrates some examples of manifolds.

Manifolds with Boundary

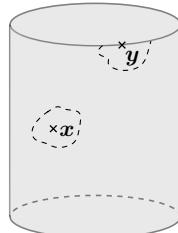


Figure 4.7: The cylinder is a 2-dimensional manifold with boundary. The neighborhood of point x is homeomorphic to B^2 but the neighborhood of y is not.

Let us consider a cylinder as presented in Figure 4.7. Some points on the surface of this cylinder will have a neighborhood that is homeomorphic to an open disk in \mathbb{R}^2 . However, some other points, such as y in Figure 4.7, will not have a neighborhood homeomorphic to an open disk on the plane and they are called boundary points. Therefore, the cylinder, because of its boundaries, can not be considered a manifold by the classical definition presented in this Section.

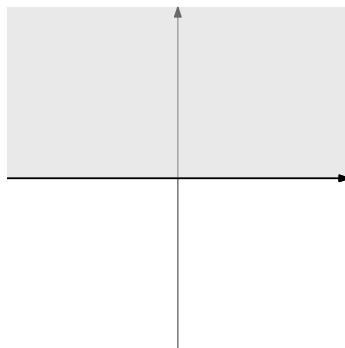


Figure 4.8: Upper Half-Space of \mathbb{R}^2 represented by the grey area.

In order to formalize these objects that are "almost" a manifold but with boundaries we need to formally define the notion of upper half-space that is illustrated in Figure 4.8. The upper half-space $\mathbb{H}^n \subseteq \mathbb{R}^n$ is

$$\mathbb{H}^n := \{(x_1, \dots, x_n) \in \mathbb{R}^n \mid x_n \geq 0\}$$

Then, we have the following definition:

Definition 18 (Topological Manifold with Boundary). *An n-dimensional manifold with boundary is a second countable Hausdorff space in which every point has a neighborhood homeomorphic to an open subset of \mathbb{R}^n or \mathbb{H}^n .*

Points of an n-dimensional manifold with boundary homeomorphic to \mathbb{R}^n are named interior points and the others are named boundary points. No point of a manifold with boundary is both an interior point and a boundary point.

If we consider a cube in \mathbb{R}^3 or a rectangle in \mathbb{R}^2 we can notice that these topological spaces do not fit into the aforementioned classifications due to the presence of *corners*. A definition for manifolds with corners can be found in the existing literature, but exploring this topic is beyond the scope of our current work. Interested readers can refer to [115] and [116] for further information.

Differentiable Manifold

Differentiable manifolds are manifolds on which the concept of smoothness can be defined. This concept allows us to extend calculus techniques to more complex spaces.

A simple example of a differentiable manifold is a sphere. For instance, take a small region of S^2 . Locally, this region looks like a flat two-dimensional plane where smooth functions can be defined, and the usual rules of calculus

apply. Different regions on the sphere might have different local coordinates and the way calculus works within might change. But as long as we take small enough patches of the sphere and can smoothly transition between them, we have a differentiable manifold.

Diffeomorphism

We can now introduce a relation between differentiable manifolds:

Definition 19 (Diffeomorphism). *A diffeomorphism between two differentiable manifolds of same dimension, take M and N , for example, is a bijective map*

$$\mathbf{f} : M \rightarrow N$$

that is itself differentiable and has a differentiable inverse $\mathbf{f}^{-1} : N \rightarrow M$.

If such a function exists, we can say that M and N are diffeomorphic.

Note that the unit cube is homeomorphic to the 2-sphere, but this transformation is not diffeomorphic. One can be continuously deformed into the other but not smoothly, since the sphere is a 2-manifold and the cube a 2-manifold with boundary.

4.3.3 Quotient Spaces

Quotient spaces serve as a mean to simplify complex topological spaces by identifying points that are equivalent under an equivalence relation. Consider the following definition, adapted from [112], for an equivalence relation on a set:

Definition 20 (Equivalence Relation). *Let X be a set. An equivalence relation on X is a binary relation \sim satisfying these three properties:*

- (i) $\mathbf{x} \sim \mathbf{x}$ for all $\mathbf{x} \in X$ (reflexivity);
- (ii) $\mathbf{x} \sim \mathbf{y}$ then $\mathbf{y} \sim \mathbf{x}$ (symmetry);
- (iii) $\mathbf{x} \sim \mathbf{y}$ and $\mathbf{y} \sim \mathbf{z}$ then $\mathbf{x} \sim \mathbf{z}$ (transitivity).

If X is a set with an equivalence relation \sim , then we can define equivalence classes for elements in X . Let $\mathbf{x} \in X$, its equivalence class is the subset $E_x \subseteq X$ consisting of all elements that are in relation with \mathbf{x} , such that

$$E_x = \{\mathbf{y} \in X \mid \mathbf{x} \sim \mathbf{y}\}$$

The following two properties are respected:

- (i) If $\mathbf{x} \sim \mathbf{y}$ then $E_x = E_y$;
- (ii) If $\mathbf{x} \not\sim \mathbf{y}$ then $E_x \cap E_y = \emptyset$;

We denote $X \setminus \sim$ the set of equivalence classes of X . It is a decomposition of the original set into a collection of disjoint subsets whose union is the original set. Function $\pi : X \rightarrow X \setminus \sim$ is called the quotient map, and it maps each element of X to its equivalence class.

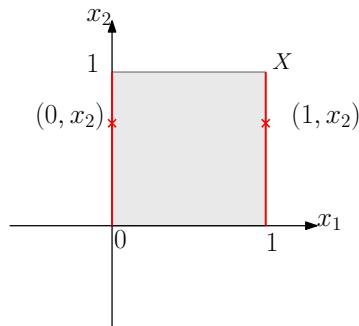


Figure 4.9: X is a set of the plane with an equivalence relation defined between its two vertical edges such that $(0, x_2) \sim (1, x_2)$ for $x_2 \in [0, 1]$.

For example, let us consider the set $X = \{(x_1, x_2) \in \mathbb{R}^2 \mid 0 \leq x_1 \leq 1 \text{ and } 0 \leq x_2 \leq 1\}$, that corresponds to a square in the plane. We can establish an equivalence relation on X between the two vertical edges, represented in red on Figure 4.9, such that, $(0, x_2) \sim (1, x_2)$ for $x_2 \in [0, 1]$. We can take, for example, the equivalence class for $(0, x_2)$ and it will be equal to $E = \{(0, x_2), (1, x_2)\}$. Point $(1, x_2)$ will have the same equivalence class. For all the other points of X for which $x_1 \in]0, 1[$, points that are not on the vertical axes, their equivalence class will consist of only the point itself.

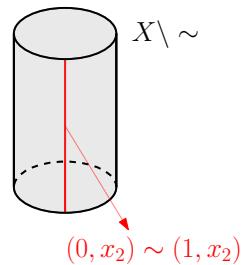


Figure 4.10: $X \setminus \sim$.

The set $X \setminus \sim$ can then be obtained by "gluing" together points that are equivalent on the original space. Therefore, for the precedent example, we obtain a cylinder by "gluing" the vertical edges of the square.

Now we can define quotient spaces using the definition presented in [112].

Definition 21 (Quotient Spaces). *Let X be a topological space and let \sim be an equivalence relation on X . The set $X \setminus \sim$ with a quotient topology determined by the quotient map function π is called the quotient space of X with respect to the relation \sim .*

The quotient topology was defined in Definition 6.

4.4 Connectedness and Compactness

Here we discuss the notion of connectedness and compactness of a topological space.

4.4.1 Connected Space

A topological space X is disconnected if and only if it can be expressed as the union of two disjoint non-empty open subsets [114]. A space that is not disconnected is said to be connected. We can also say that a space is connected if there are no two nonempty open sets whose union is the entire space, and whose intersection is empty. Intuitively, a connected space is one that is "all in one piece," and there is no way to split it into two separate, non-overlapping parts.

An example of a connected space is the interval $[0, 1]$ in the standard Euclidean topology. An example of disconnected space is the real line \mathbb{R} without the origin 0 that can be divided into two non overlapping open sets $[-\infty, 0[$ and $]0, \infty]$.

X is connected if and only if X itself and \emptyset are the only subsets of X that are simultaneously open and closed. Otherwise, we have a non-empty subset $O \subset X$ that is open but its complement $X \setminus O$ is also open. The union of O with its complement is a union of two disjoint open subsets that result in X , therefore, by definition, X is disconnected.

Any space homeomorphic to a connected space is also connected, therefore connectedness is a topological property.

Path-Connectedness

A stronger property than connectedness is path-connectedness. It describes the existence of continuous paths between any two points in the space. A path-connected space is one in which, for every pair of points, there is a continuous function, called a path, that maps the unit interval onto the space and joins the two points.

Formally, a path from \mathbf{x} to \mathbf{y} on a topological space X is a continuous map $\mathbf{f} : [0, 1] \rightarrow X$ with $\mathbf{f}(0) = \mathbf{x}$ and $\mathbf{f}(1) = \mathbf{y}$.

A topological space X is path-connected if and only if for all pair of points \mathbf{x} and \mathbf{y} in X , there is a path in X from \mathbf{x} to \mathbf{y} . A path-connected space is always connected, but a space can be connected without being path connected. The topologist's sinecurve is one of the most common examples of this [114], and it can be defined by the following equation:

$$f(x) = \begin{cases} 0 & \text{if } x = 0 \\ \sin(\frac{1}{x}) & \text{if } x \in (0, 1] \end{cases}$$

4.4.2 Compact Space

Compactness is a concept in topology that captures the notion of finiteness in topological spaces. A topological space X is considered compact if it has the property that every open cover of X contains a finite subcover of X . Breaking down this last statement we have that an open cover of a space X is a collection C of open subsets of X whose union is X . A subcover of C on the other hand is a subcollection of elements of C that still covers X .

For example, every finite topological space is compact because finite spaces have a finite number of open elements. Every topological space with trivial topology will also be compact.

Let us consider set $X = \{1, 2, 3\}$ with topology

$$T = \{X, \emptyset, \{1\}, \{3\}, \{1, 2\}, \{1, 3\}\}$$

An open cover of X can be the collection

$$C = \{\{1, 2\}, \{1, 3\}, \{1\}, \{3\}\}$$

and it has different possible subcovers that remove redundant subsets from the original open cover. For example, we have $S_C = \{\{1, 2\}, \{1\}, \{1, 3\}\}$, but we can also have $S_C = \{\{1, 2\}, \{1, 3\}\}$. The latter is the smallest subcover contained in this open cover.

An interesting property of compact spaces is that compact subsets of a Hausdorff space are bounded. Proof of this proposition can be found in [109]. As a consequence, any subset of \mathbb{R}^2 homeomorphic to D^2 , for example, is compact.

Locally Compact

A locally compact space is a topological space such that for every point in the space, there exists a compact subspace containing an open neighborhood of that point. In other words, each point in a locally compact space has a compact neighborhood.

Formally, a space X is locally compact at $\mathbf{x} \in X$ if there is some compact subspace C of X that contains a neighborhood of \mathbf{x} . If X is locally compact at each of its points, X is locally compact [112].

4.4.3 Compactification

Compactification is a technique that allows us to extend a given topological space to a compact space by adding boundary points to it. The process of compactification is particularly useful when we want to study the original space in a more complete and structured manner, as it will be the case later in this work. We only discuss the compactification of Hausdorff spaces.

Formally, let X be a topological space. A compactification of X is a pair (Y, \mathbf{f}) , where Y is a compact Hausdorff space and $\mathbf{f} : X \rightarrow Y$ is a continuous map. In addition, \mathbf{f} is an embedding, meaning that it is a homeomorphism between X and its image $\mathbf{f}(X)$ in Y .

As it was mentioned, the compactification process typically involves adding new points to the original space X to create the compact space Y . These new points, often referred to as boundary points or points at infinity, capture the behavior of sequences or directions in X that go off to infinity. The resulting compact space Y may have a different topological structure from X , but it retains essential topological properties.

For example, $[0, 1]$ is a compactification of $[0, 1[$. Here, the compactified space was obtained by adding only one point to the original space. This process can be generalized to arbitrary locally compact Hausdorff spaces with the one-point Compactification [114].

Alexandroff One-Point Compactification

Definition 22 (One-Point Compactification). *Let X be a locally compact, non-compact Hausdorff space and \mathbf{p} a point not in X . We define a new space $X^* = X \cup \{\mathbf{p}\}$. We assume that the neighborhoods of \mathbf{p} are sets of the form $\{\mathbf{p}\} \cup (X \setminus L)$, where L are compact sets of X . Neighborhoods of X are unchanged in X^* . We call X^* the one-point compactification, or equivalently Alexandroff compactification, of X and we have the following properties:*

- (i) X^* is compact;

(ii) X^* is Hausdorff.

The proofs for the properties presented on this definition were discussed by [114] and are out of the scope of this document.

To summarize, we can say that if X^* is a compact Hausdorff space and X is a subspace of X^* whose closure equals X^* , then X^* is said to be a compactification of X . If in addition $X^*\setminus X$ equals a single point, then X^* is called the one-point compactification of X [112].

In order to make the plane \mathbb{R}^2 compact one can add one point at infinity, resulting in the S^2 sphere under stereographic projection. Similarly, S^1 is a compactification of R .

4.5 The Fundamental Group

Until this moment, in this Chapter, we discussed concepts from general topology. In this section we start discussing concepts from algebraic topology. Algebraic topology introduces algebraic techniques to study topological spaces. It focuses on their invariants that are preserved under continuous deformations and transformations.

The fundamental group is a concept from algebraic topology. This is a key concept that will be used in forthcoming chapters to define a solution to the exploration problem considered in this work. At its core, the fundamental group measures how loops in a space can be continuously deformed without tearing or breaking. *Loops* are closed paths that start and end at the same point in the space.

Definition 23 (Loop). *Let $\gamma : [0, 1] \rightarrow X$ be a path on a topological space X . We say that γ is a loop if and only if $\gamma(0) = \gamma(1)$.*

We say that a loop in a topological space X is *based at* point $\mathbf{x} \in X$ if $\gamma(0) = \gamma(1) = \mathbf{x}$. In this case, \mathbf{x} is a *basepoint*.

In this Section we present how the fundamental group classifies these loops. But first, we remember the definition of groups and define the notion of isomorphism on groups.

4.5.1 Groups and Isomorphism

A group is a set G together with an operation \cdot that satisfies the following properties [117]:

- (i) Closure: For all elements $\mathbf{a}, \mathbf{b} \in G$, the result of the operation $\mathbf{a} \cdot \mathbf{b}$ is also an element of G ;

- (ii) Associativity: For all elements \mathbf{a} , \mathbf{b} and $\mathbf{c} \in G$, $(\mathbf{a} \cdot \mathbf{b}) \cdot \mathbf{c} = \mathbf{a} \cdot (\mathbf{b} \cdot \mathbf{c})$;
- (iii) Identity Element: There exists an element $\mathbf{i} \in G$, called the identity element, such that for any element $\mathbf{a} \in G$, $\mathbf{a} \cdot \mathbf{i} = \mathbf{i} \cdot \mathbf{a} = \mathbf{a}$;
- (iv) Inverse Element: For every element $\mathbf{a} \in G$, there exists an element $\mathbf{b} \in G$, called the inverse of \mathbf{a} , such that $\mathbf{a} \cdot \mathbf{b} = \mathbf{b} \cdot \mathbf{a} = \mathbf{i}$, where \mathbf{i} is the identity element;

If, in addition to the properties listed above, a group also satisfies the following property:

- Commutativity: For all elements \mathbf{a} , $\mathbf{b} \in G$, $\mathbf{a} \cdot \mathbf{b} = \mathbf{b} \cdot \mathbf{a}$.

we say that the group is abelian.

Abelian groups are among the simplest algebraic structures, consisting of a set and a single binary operation. Their simplicity and versatility give them a wide range of applications. The commutativity property, which states that the order of elements does not matter under the binary operation, is a distinguishing feature of abelian groups, simplifying calculations and allowing for easier manipulation of group elements. A simple example of abelian group is the group $(\mathbb{Z}, +)$ of integers under addition.

Isomorphism

Isomorphism is a broad and complex topic, but for our current purpose, we provide a simple definition that conveys the essential information needed to comprehend the subsequent discussions.

Let (G, O_G) and (H, O_H) be two groups. A function $f : G \rightarrow H$ is an isomorphism if it is bijective and if it preserves the group structure. That is, the operation O_G in the group G corresponds to the operation O_H in the group H .

If an isomorphism exists between two groups G and H , we say that G and H are isomorphic, denoted as $G \simeq H$. Intuitively, isomorphic groups have the same group structure, meaning they are essentially the same group with different labels on the elements.

The key concept for understanding the fundamental group lies in the definition of *homotopy* presented on next section.

4.5.2 Homotopy

Let us consider the definition below, given in [113]:

Definition 24 (Homotopy). Consider two topological spaces X and Y and let $\mathbf{f}, \mathbf{g} : X \rightarrow Y$ be continuous functions. Then \mathbf{f} is homotopic to \mathbf{g} if there exists a continuous function $F : X \times I \rightarrow Y$ such that $F(\mathbf{x}, 0) = \mathbf{f}(\mathbf{x})$ and $F(\mathbf{x}, 1) = \mathbf{g}(\mathbf{x})$ for all points $\mathbf{x} \in X$.

F is called a homotopy between \mathbf{f} and \mathbf{g} and we can say $\mathbf{f} \simeq \mathbf{g}$. Being homotopic is an equivalence relation on the set of continuous functions from X to Y .

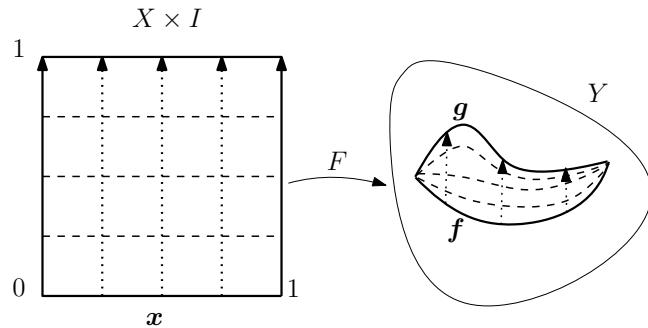


Figure 4.11: Example of homotopy between two paths \mathbf{f} and \mathbf{g} .

As illustrated in Figure 4.11, the second parameter of function F is usually referred to as a "slider control" that allows us to smoothly transition from \mathbf{f} to \mathbf{g} as the slider moves from 0 to 1, and vice versa.

4.5.3 Homotopy Equivalence

While homotopy allows us to smoothly deform maps within a space, homotopy equivalence establishes a deeper connection between spaces themselves. Briefly, a homotopy equivalence between two spaces implies the existence of continuous transformations that seamlessly connect one space to the other while preserving essential topological properties.

From [109]:

Definition 25 (Homotopy Equivalence). Let $\mathbf{f} : X \rightarrow Y$ be a continuous map between two topological spaces X and Y and $\mathbf{g} : Y \rightarrow X$ a continuous map from Y to X . If $\mathbf{g} \circ \mathbf{f}$ is homotopic to the identity map id_x and $\mathbf{f} \circ \mathbf{g}$ homotopic to the identity map id_y then we say that X is homotopy equivalent to Y , or X has the same homotopy type as Y .

Instinctively, homotopy equivalence refers to the ability to continuously deform two topological spaces into each other through bending, shrinking, and expanding operations. For example, the ball B^2 is homotopy equivalent to a

point $\mathbf{p} \in \mathbb{R}^2$. Let \mathbf{p} be a point in B^2 , for any $\alpha \in B^2$ we have $\mathbf{f}(\alpha) = \mathbf{p}$, then clearly $(\mathbf{f} \circ \mathbf{g})(\mathbf{p}) = \mathbf{p}$ and $(\mathbf{f} \circ \mathbf{g}) \simeq I_{\mathbf{p}}$. Then we have $(\mathbf{g} \circ \mathbf{f})(\alpha) = \mathbf{p}$ and by radial retraction $(\mathbf{g} \circ \mathbf{f})$ is homotopic to I_{B^2} as expected. Two distinct points are homotopy equivalent, then, this reasoning can be extended to any point \mathbf{p} and to any open set in the plane.

Spaces that can be continuously collapsed to a single point, and therefore homotopy equivalent to a single point, are called *contractible*.

Difference with Homeomorphism

Homeomorphism establishes a stronger equivalence, preserving the entire topological structure, including open sets, compactness, and connectedness. Homotopy equivalence allows for more flexible transformations, while homeomorphism implies a one-to-one correspondence between spaces with identical topological properties.

For example, B^2 is homotopy equivalent to a single point by a continuous map that "collapses" the entire space to a single point along radial lines. However, B^2 is not homeomorphic to a single point, for that there should exist a bijective continuous map and its inverse between the two spaces.

4.5.4 Construction of the Fundamental Group

Let X be a topological space and consider the set of all loops in X based at a basepoint $\mathbf{p} \in X$. The relation of homotopy is an equivalence relation on this set and based on this notion the fundamental group can be defined.

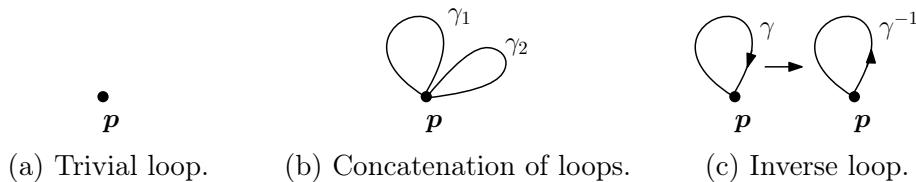


Figure 4.12: Concatenation of loops as the group operation.

Definition 26 (Fundamental group). *The fundamental group of a topological space X with respect to a basepoint \mathbf{p} , denoted, $\pi_1(X, \mathbf{p})$, is a set of equivalence classes of loops based at \mathbf{p} , where the equivalence relation is given by homotopy. The operation of the group is the concatenation of loops. The identity element of the group is the trivial loop that stays at the basepoint, and we define the inverse of a loop as the loop traced in the reverse direction. These concepts are illustrated in Figure 4.12.*

Since any loop based at p must lie entirely inside the path component of X which contains p , we restrict ourselves to path-connected spaces. With this restriction, the fundamental group is independent of p , and we can use notation $\pi_1(X)$ to refer to the fundamental group of a path-connected space X [113].

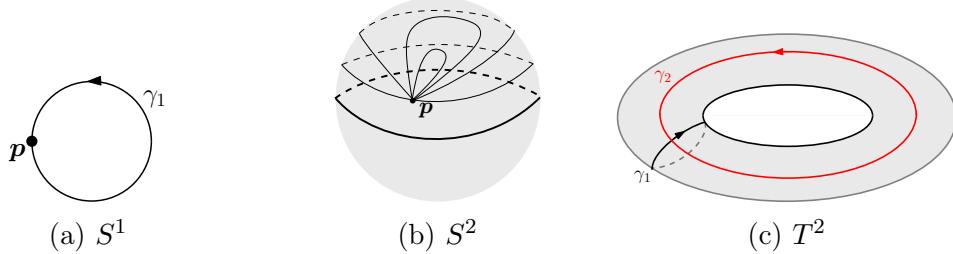


Figure 4.13: Examples of fundamental groups.

To give a few examples, let us consider S^1 , Figure 4.13a. There are two non homotopic loops that can be easily identified, they are the trivial loop, that can be any point in S^1 , and γ_1 that goes around all the points in S^1 . We can then consider the concatenation of γ_1 with itself and obtain $\gamma_1^2, \gamma_1^3, \gamma_1^4, \dots$ that are non-homotopic to γ_1 or to each other. The inverse of the concatenated cycles $\gamma_1^{-1}, \gamma_1^{-2}, \gamma_1^{-3}, \gamma_1^{-4}, \dots$ are also non-homotopic to each other and neither to the precedent loops. Thus, one can conclude that for each integer $n \in \mathbb{Z}$ we have a different homotopy class on the 1-sphere. Therefore, we define the fundamental group $\pi_1(S^1) = \{\gamma_1^n | n \in \mathbb{Z}\}$ where each homotopy class consists of all loops that wind around the 1-sphere a given number of times, this number can be negative or positive considering the direction of winding. Finally, we can say that $\pi_1(S^1)$ is isomorphic to the group $(\mathbb{Z}, +)$, the additive group of integers.

For S^2 , we have that all loops are homotopic to the trivial loop since they can all be continuously deformed to a single point, as illustrated in Figure 4.13b. In this case $\pi_1(S^2)$ is often denoted by 0, we have $\pi_1(S^2) = 0$. Generalizing, for $n \in \mathbb{N}$, $\pi_1(S^n) = \mathbb{Z}$ if $n = 1$ and $\pi_1(S^n) = 0$ if $n > 1$.

The last example is the torus, that we name T^2 . The fundamental group $\pi_1(T^2)$ is the product of the fundamental group of the circle with itself: $\pi_1(T^2) = \pi_1(S^1) \times \pi_1(S^1) \simeq \mathbb{Z} \times \mathbb{Z}$. This is represented in Figure 4.13c.

Considering again the fundamental group on the 1-sphere and let γ be a loop on this space. This loop is a continuous map from the 1-sphere to itself, and we can assign a number $\deg(\gamma) \in \mathbb{Z}$ to γ that measure the number of times it winds positively around S^1 . By convention, we assume the counterclockwise direction to be positive. This number determines to which equivalence class in

$\pi_1(S^1)$ γ belongs to. Section 4.8 formalizes this notion that we just introduced, $\deg(\gamma)$ is named the degree of function γ , and this concept plays a major role in this work.

4.6 CW Complexes

Combinatorial models for constructing topological spaces are methods that use combinatorial structures to represent and study topological properties of spaces. These models serve as a tool for understanding the topology of spaces in a more concrete and discrete manner, making complex topological problems more amenable to analysis.

Some common combinatorial models used for constructing topological spaces are graphs. Graphs are simple combinatorial structures consisting of vertices and edges. They can be used to represent and study topological spaces with a 1-dimensional structure, such as curves and networks.

In this Section we present a combinatorial structure, that is useful in the context of this work. They are called CW complexes.

A CW complex is composed of cells of varying dimensions, where each cell is essentially a Euclidean space of a certain dimension.

There are two types of cells in a CW complex, open and closed cells. An open cell of dimension n is a subset of Euclidean space \mathbb{R}^n that is homeomorphic to an n -dimensional open ball, and it is called open n -cell. A closed cell of dimension n is the closure of an open cell of dimension n , which means it includes all the points on the boundary of the open cell. It is homeomorphic to a closed n -dimensional ball, and it is called closed n -cell.

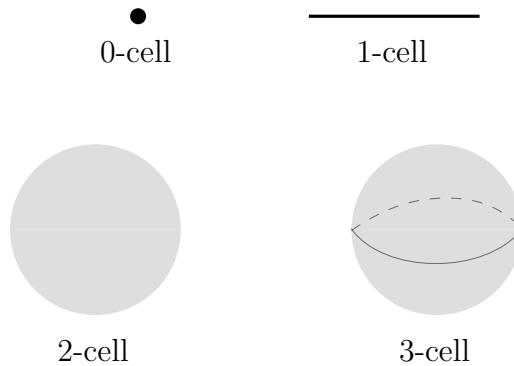


Figure 4.14: Building blocks of CW complexes.

In Figure 4.14 we illustrate some building blocks of CW complexes.

The CW complex is built by iteratively attaching cells of various dimensions to a growing skeleton. The n -skeleton of a CW complex consists of cells of

dimension at most n . The 0-skeleton consists of 0-cells, which are just the vertices of the complex. To form the 1-skeleton, 0-cells and 1-cells are connected. Then, k -cells are attached to the $k - 1$ -skeleton along their boundaries.

The two following rules must be respected when attaching cells together to create a complex:

1. Cells are attached along their boundaries in a specific way. For example, a lower-dimensional cell can be attached to the boundary of a higher-dimensional cell.
2. The attachment process should be continuous, meaning that as cells are attached, their boundaries are smoothly connected to the boundaries of other cells.

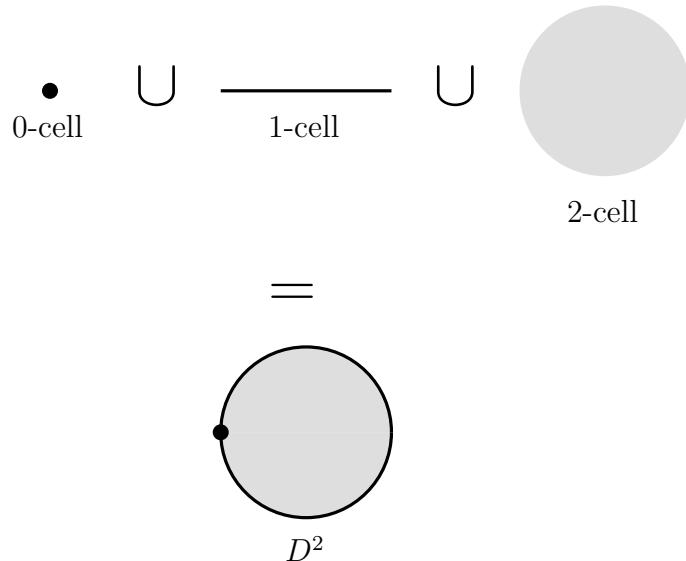


Figure 4.15: CW complex covering D^2 .

In Figure 4.15 an example of CW complex covering D^2 is presented.

4.7 Homology

We present in this Section another group that can be assigned to a topological space and that is topologically invariant, the homology group. Homology groups capture the idea of cycles and boundaries in a topological space.

4.7.1 Homology Group

Homotopy groups are challenging to obtain and analyze, especially for higher dimensions. In this context, the homology group is often referred to as a simplification of the homotopy group because it captures essential topological information about a space while being computationally more accessible and easier to work with than the homotopy group.

As a consequence of Hurewicz theorem, we can say that the first homology group for a path-connected space is created through an abelianization of the fundamental group, where computability problems are easier to be approached.

In a nutshell, the n -th homology group of a space measures the $n + 1$ -dimensional holes within that space or, equivalently, of holes with n -dimensional boundary. A precise definition includes concepts beyond the scope of this document. Instead, we provide an understanding of homology groups through illustrative examples, which will bring enough comprehension for the context in which they are considered in this work.

For instance, if 0 represents the trivial group, we have the following:

Example 1: (Spheres) The homology group of spheres are

$$H_k(S^n) = \begin{cases} \mathbb{Z} & k = 0, n \\ 0 & \text{otherwise} \end{cases}$$

Take S^1 , for example. The 1-sphere is single-connected, and, as a consequence, $H_0(S^1) = \mathbb{Z}$. This happens because 0-dimensional-boundary holes can be detected by analyzing the connectivity of the space and describe the path-connected components of the topological space [118]. In addition, S^1 has one two-dimensional hole with a 1-dimensional boundary, then $H_1(S^1) = \mathbb{Z}$.

The two-dimensional sphere S^2 is single-connected, hence we have $H_0(S^2) = \mathbb{Z}$ representing its path-connectedness. This space also has one 3-dimensional hole with a 2-dimensional boundary, therefore, we have $H_2(S^2) = \mathbb{Z}$. The extension to higher dimensions follows the same logical framework.

Example 2: (Disk) For a disk D^n we have only one path-connected component but no holes, therefore

$$H_k(D^n) = \begin{cases} \mathbb{Z} & k = 0 \\ 0 & \text{otherwise} \end{cases}$$

Example 3: (Torus) The torus $T^2 = S^1 \times S^1$ has a single-path connected component, two 1-dimensional-boundary holes and one 2-dimensional-boundary hole (its interior). In this case,

$$H_k(T^2) = \begin{cases} \mathbb{Z} & k = 0, 2 \\ \mathbb{Z} \times \mathbb{Z} & k = 1 \\ 0 & \text{otherwise} \end{cases}$$

The three examples above have in common the fact that their homology group H_1 is equal to their respective fundamental group π_1 . This happens because for these examples the fundamental group is already abelian. Now, let us consider an example where this will not be the case:

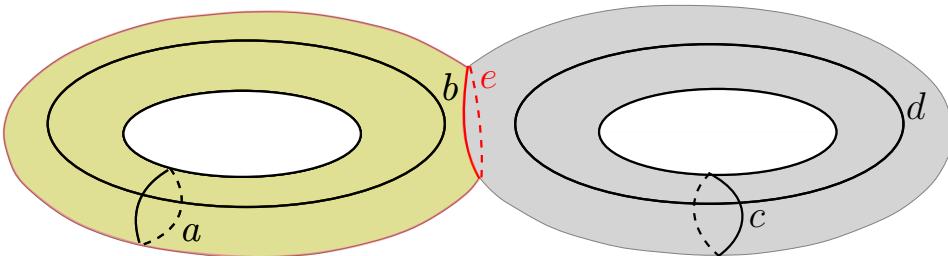


Figure 4.16: There are four independent loops on the two-holed torus: a, b, c and d . The loop e is the boundary of the area represented in yellow, and therefore it is considered to be trivial in Homology.

Example 4: (2-holed Torus) The 2-holed torus denoted $2T^2$, Figure 4.16, is the connected sum of two copies of T^2 . Its fundamental group π_1 and first homology groups H_1 are composed of combinations of loops a, b, c and d , represented on the Figure. Therefore, in practice, we have

$$\pi_1(2T^2) = \mathbb{Z} \times \mathbb{Z} \times \mathbb{Z} \times \mathbb{Z}$$

and

$$H_1(2T^2) = \mathbb{Z} \times \mathbb{Z} \times \mathbb{Z} \times \mathbb{Z}$$

However, intrinsically, group H_1 differs of π_1 .

This happens because loop e , represented in red in Figure 4.16, is not trivial in π_1 , because this cycle can not be deformed into the trivial loop. As a consequence some loops in the space will not commute, the order in which we combine a, b, c and d matters, and the group is non-abelian. On the other hand, loop e is considered to be trivial in H_1 because it is the boundary of the surface represented in yellow on the figure (remember that in homology groups we count boundaries). By considering cycle e as trivial, the non-commutative problem no longer exists in H_1 , and it is an abelian group.

Having established the concepts of fundamental group and homology group, we are now primed to define two pivotal notions: the topological degree and the winding number. These fundamental concepts will play a central role in this thesis.

4.8 Topological Degree

The topological degree quantifies the winding behavior of continuous maps from one compact oriented manifold to another of same dimension. Simply stated, it is an integer that represents how many times the original space wraps around a point of the target manifold under the mapping.

In this work, we concentrate on the definition of the topological degree of mappings between Euclidean spaces. Here we give a general axiomatic definition and recap the main properties that we use, as stated in [119],

Definition 27 (Topological degree). *Let D be an open subset of \mathbb{R}^n and \mathbf{f} a continuous function from its closure \overline{D} to \mathbb{R}^n . A degree of \mathbf{f} is a family of functions $\deg : (\mathbf{f}, D, \mathbf{p}) \rightarrow \mathbb{Z}$ for all D open subsets of \mathbb{R}^n , \mathbf{f} continuous and $\mathbf{p} \in \mathbb{R}^n \setminus \mathbf{f}(\partial D)$ such that:*

- (identity) $\deg(Id_D, D, \mathbf{p}) = 1$ if $\mathbf{p} \in D$
- (excision) $\deg(\mathbf{f}, D, \mathbf{p}) = \deg(\mathbf{f}, D_1, \mathbf{p}) + \deg(\mathbf{f}, D_2, \mathbf{p})$ where D_1, D_2 are opens in D with $\mathbf{p} \notin \mathbf{f}(\overline{D} \setminus (D_1 \cup D_2))$
- (homotopy invariance) $\deg(\mathbf{h}(\alpha, .), D, \mathbf{p}(\alpha))$ is independent of α for any homotopy $\mathbf{h} : [0, 1] \times \overline{D} \rightarrow \mathbb{R}^n$, and $\mathbf{p}(\alpha) \notin \mathbf{h}(\alpha, \partial D)$ for all $\alpha \in [0, 1]$.

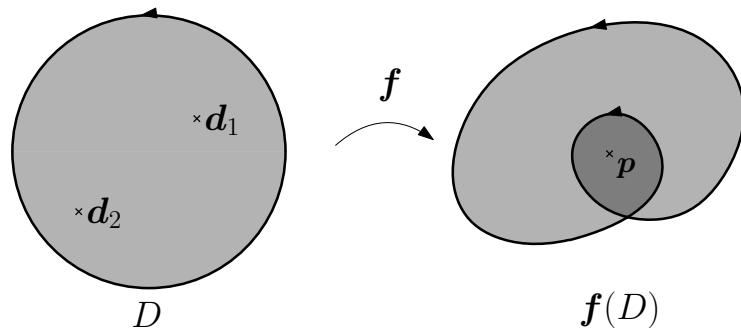


Figure 4.17: Example of mapping defined on \mathbb{R}^2 , defined by a function $\mathbf{f} : D \rightarrow \mathbb{R}^2$ that wraps the source domain D in a way that points \mathbf{d}_1 and \mathbf{d}_2 coincide in the target domain. Equivalently we can say $\mathbf{f}(\mathbf{d}_1) = \mathbf{f}(\mathbf{d}_2) = \mathbf{p}$. It can be concluded in this case that $\deg(\mathbf{f}, D, \mathbf{p}) = 2$.

When such a family of function exists, it is known to be unique and the notion presented on this definition is illustrated in Figure 4.17.

In particular, when \mathbf{f} is at least continuously differentiable, and \mathbf{p} is a regular value of \mathbf{f} (i.e. the determinant of the Jacobian of \mathbf{f} , $\det(J_{\mathbf{f}})$ is non-zero on each \mathbf{d} with $\mathbf{f}(\mathbf{d}) = \mathbf{p}$):

$$\deg(\mathbf{f}, D, \mathbf{p}) = \sum_{\mathbf{d} \in \mathbf{f}^{-1}(\mathbf{p})} \text{sign}(\det(J_{\mathbf{f}}(\mathbf{d}))) \quad (4.1)$$

4.8.1 Winding Number

The topological degree of differentiable functions from D^2 to \mathbb{R}^2 is linked to the winding number of $\mathbf{f}(\partial D^2)$.

Let \mathbf{p} be a point in the interior of the image by \mathbf{f} of D^2 , in other words, $\mathbf{p} \in \mathbf{f}(B^2)$. Function \mathbf{f} maps S^1 , on a cycle in \mathbb{R}^2 , and the winding number is the number of times this cycle turns around \mathbf{p} :

Definition 28 (Winding number). *Let $\mathbf{f} : D^2 \rightarrow \mathbb{R}^2$ be a continuous function and $\mathbf{p} \in \mathbf{f}(D^2) \setminus \mathbf{f}(S^1)$. Consider its restriction $\mathbf{f}|_{S^1} : S^1 \rightarrow \mathbb{R}^2 \setminus \{\mathbf{p}\}$. It induces a linear map in homology:*

$$\tilde{\mathbf{f}} : H_1(S^1) \rightarrow H_1(\mathbb{R}^2 \setminus \{\mathbf{p}\})$$

i.e. from \mathbb{Z} to \mathbb{Z} , i.e. is of the form $\tilde{\mathbf{f}}(C) = \eta C$, where C represents an equivalence class in $H_1(S^1)$. This η is called the winding number of $\gamma = \mathbf{f}(S^1)$ around point $\mathbf{p} \in \mathbf{f}(D^2) \setminus \mathbf{f}(S^1)$. For all other points in $\mathbb{R}^2 \setminus \mathbf{f}(S^1)$ the winding number is set to zero.

We can now state the relation between the topological degree and the winding number:

Lemma 4.8.1. *Let \mathbf{f} be a continuously differentiable map from D^2 to \mathbb{R}^2 and let $\mathbf{y} \in \mathbb{R}^2 \setminus \mathbf{f}(\partial D^2)$ such that $\mathbf{f}^{-1}(\mathbf{y})$ is finite and \mathbf{y} is a regular point for \mathbf{f} . Then $\deg(\mathbf{f}, D^2, \mathbf{y})$ is equal to the winding number $\eta(\mathbf{f}(\partial D^2), \mathbf{y})$ of $\mathbf{f}(\partial D^2)$ at \mathbf{y} .*

Proof. For all $\mathbf{y} \in \mathbb{R}^2 \setminus \mathbf{f}(\partial D^2)$, either there exists no \mathbf{d} such that $\mathbf{y} = \mathbf{f}(\mathbf{d})$, or there exists a finite, non-zero number of \mathbf{d} , $\mathbf{d}_1, \dots, \mathbf{d}_m$ in D^2 , such that $\mathbf{f}(\mathbf{d}_i) = \mathbf{y}$.

In the first case, this means that both, $\deg(\mathbf{f}, D^2, \mathbf{y})$ is zero and \mathbf{y} is in the complement of $\mathbf{f}(D^2)$ and the winding number $\eta(\mathbf{f}(\partial D^2), \mathbf{y})$ is also zero.

In the second case, \mathbf{y} being regular for \mathbf{f} , we have $\deg(\mathbf{f}, D, \mathbf{y}) = \sum_{i=1}^m \text{sign}(\det(J_{\mathbf{f}}(\mathbf{d}_i)))$. Take small enough open neighborhoods U_i of \mathbf{d}_i in D

such that the sign of $\det(J_{\mathbf{f}}(\mathbf{d}))$ is the same as the sign of $\det(J_{\mathbf{f}}(\mathbf{d}_i))$ for all $\mathbf{d} \in U_i$. This is always possible since $J_{\mathbf{f}}$ is continuous. Note that this implies that \mathbf{f} restricted to U_i induces an homeomorphism onto its image. Also we can always choose the U_i to have empty pairwise intersections and to have \mathbf{f} being an homeomorphism from $\overline{U_i}$ onto its image, by taking them small enough (the \mathbf{d}_i are isolated points within D).

Now, the map $\tilde{\mathbf{f}}$ is the same as the map induced in homology $\tilde{\mathbf{f}}$ by $\mathbf{f} : D^2 \setminus \bigcup_{i=1}^m U_i \rightarrow \mathbb{R}^2 \setminus \{\mathbf{y}\}$. We note also that within $D^2 \setminus \bigcup_{i=1}^m U_i$, the cycle ∂D^2 is homologous to the sum of the $\partial(U_i)$, for $i = 1, \dots, m$. Hence $\tilde{\mathbf{f}}(\partial D^2) = \sum_{i=1}^m \tilde{\mathbf{f}}(\partial(U_i))$.

But $\mathbf{f}(\partial(U_i))$ is a Jordan curve homeomorphic (by \mathbf{f}) to $\partial(U_i)$, since we chose U_i such that \mathbf{f} restricted to $\overline{U_i}$ onto its image is a homeomorphism. Hence $\tilde{\mathbf{f}}(\partial U_i)$ is either plus or minus identity, according to the orientation of $\tilde{\mathbf{f}}(\partial U_i)$, i.e. $\tilde{\mathbf{f}}(\partial U_i) = \text{sign}(\det(J_{\mathbf{f}}(\mathbf{d})))$ for any $\mathbf{d} \in U_i$, which we know is equal to $\text{sign}(\det(J_{\mathbf{f}}(\mathbf{d}_i)))$. Hence

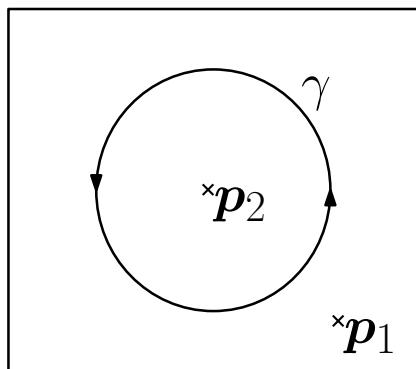
$$\eta(\mathbf{f}(\partial D^2), \mathbf{y}) = \sum_{i=1}^m \text{sign}(\det(J_{\mathbf{f}}(\mathbf{d}_i))) = \deg(\mathbf{f}, D^2, \mathbf{y})$$

□

In Figure 4.18 we present some examples of winding numbers for different mapping functions.

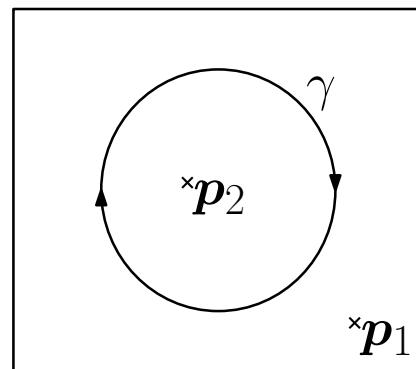
4.9 Conclusion

On this Chapter we presented how topology remove limitations imposed by metrics on spaces by focusing on the properties that remain invariant under continuous deformations, an approach that allows the study of features of spaces that might otherwise be obscured by rigid geometric notions. We also presented topological tools used to analyze and classify spaces. In the upcoming chapters, these tools will be applied, along with the notions of interval analysis presented on the previous Chapter, to formalize and to establish a solution to the problem envisioned in this work.



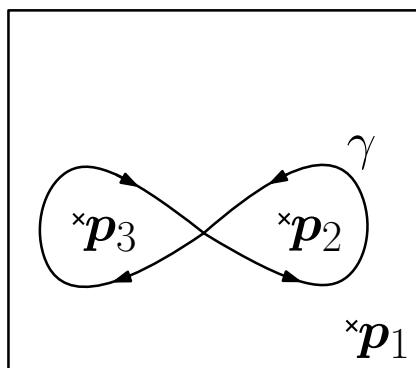
$$\eta(\gamma, \mathbf{p}_1) = 0$$

$$\eta(\gamma, \mathbf{p}_2) = 1$$



$$\eta(\gamma, \mathbf{p}_1) = 0$$

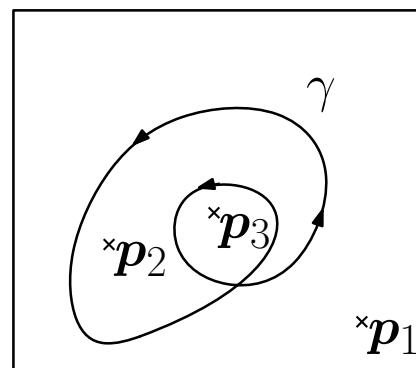
$$\eta(\gamma, \mathbf{p}_2) = -1$$



$$\eta(\gamma, \mathbf{p}_1) = 0$$

$$\eta(\gamma, \mathbf{p}_2) = 1$$

$$\eta(\gamma, \mathbf{p}_3) = -1$$



$$\eta(\gamma, \mathbf{p}_1) = 0$$

$$\eta(\gamma, \mathbf{p}_2) = 1$$

$$\eta(\gamma, \mathbf{p}_3) = 2$$

Figure 4.18: In these examples we have $\gamma = \mathbf{f}(\partial D^2)$ where function \mathbf{f} represents a continuous map $\mathbf{f} : D^2 \rightarrow \mathbb{R}^2$.

Part II

Contributions

Chapter 5

Coverage Measure and Explored Area

Contents

5.1	One-Dimensional Sensors	115
5.1.1	Problem Statement	115
5.1.2	Coverage Measure and the Topological Degree	121
5.1.3	Dealing with Uncertainties	129
5.2	Two-Dimensional Sensors	132
5.2.1	Problem Statement	132
5.2.2	Coverage Measure and Topological Degree	143
5.2.3	Sweeping backwards and Uncertainties	146
5.3	Conclusion	147

The primary objective of this thesis is to propose an algorithm that provides a reliable estimation of the area explored by a mobile robot in a two-dimensional environment and the coverage measure of the points within. The preceding sections have laid out the necessary theoretical background for addressing this particular challenge. In this Chapter, we provide a mathematical representation of the problem at hand, formally defining the notions of coverage measure and explored area.

In this work, we consider two different scenarios: the first scenario assumes the robot exploring the environment using a one-dimensional sensor, while the second scenario generalizes the method for two-dimensional sensors. The fundamental contribution of this work is then presented: A relation between topological tools, more specifically the topological degree, and the coverage measure is established. Finally, we extend the problem's definition to encompass the aspect of uncertainty. The theory developed here will later be used for developing the ultimate application of this thesis in Chapter 6.

5.1 One-Dimensional Sensors

5.1.1 Problem Statement

Consider a mobile robot exploring an unknown planar environment. At first, we assume that the robot's pose can be fully described by $(\mathbf{x} \ \psi)^T \in \mathbb{R}^3$ where $\mathbf{x} : \mathbb{R} \rightarrow \mathbb{R}^2$ and $\psi : \mathbb{R} \rightarrow \mathbb{R}$ are functions of time. Function \mathbf{x} represents the robot's trajectory on the plane, and we assume to be at least C^2 . Function ψ represents the robot's orientation on the plane, and it is at least C^1 . Given the context, the assumptions regarding the continuity of these functions derivatives are reasonable. This is because acceleration and angular speed measurements can be easily acquired using basic proprioceptive sensors, like inertial measurement units, often integrated on the robot. The robot's trajectory satisfies the following differential equation

$$\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}, \psi) \quad (5.1)$$

In the scenario considered here, the robot navigates within a two-dimensional environment, using its exploration sensors to sense its surroundings throughout the mission. We denote $[0, T]$ as the mission's time interval where T is a real positive value.

Visible Area

We define $\mathbb{V} : [0, T] \rightarrow \mathcal{P}(\mathbb{R}^2)$ a set-valued function, $\mathbb{V}(t)$ represents the robot's visible area at instant $t \in [0, T]$. This set encompasses all the points on the

plane inside the robot's range of visibility for a given instant within exploration time.

On this first part we assume that the robot's visible area is one-dimensional. In Chapter 2, we presented Side-Scan Sonars and emphasized their significance in underwater exploration missions. With the hypotheses introduced at the end of the chapter, the visible area of a side-scan sonar can be conceptualized as one-dimensional. Therefore, in this section, we consider a visible area inspired by the configuration of a robot with one side-scan sonar. We assume that the sensor scans the environment on the robot's left side while it moves around the plane.

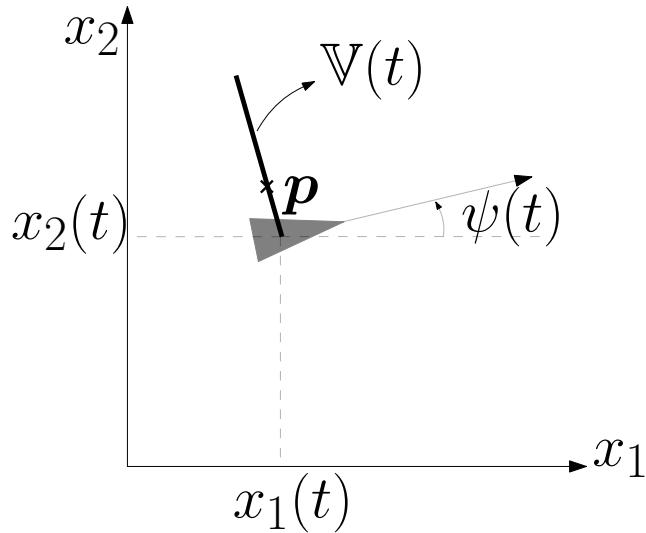


Figure 5.1: Mobile robot with a one-dimensional exploration sensor on the plane. At instant t the point \mathbf{p} is sensed by the robot, therefore $\mathbf{p} \in \mathbb{V}(t)$.

This configuration is presented in Figure 5.1 where the robot is illustrated at some instant $t \in [0, T]$ with its position defined by the pair $\mathbf{x}(t) = (x_1(t), x_2(t))$.

Let L be a positive real number and the sensor's visible range, the visible area in the example can then be defined as

$$\mathbb{V}(t) = \bigcup_{l \in [0, L]} (x_1(t) - l \sin(\psi(t)), x_2(t) + l \cos(\psi(t))) \quad (5.2)$$

One can notice that the visible area depends on the robot's pose and its path and movements dictate which parts of the environment fall within the sensor's observation. The union of the sensor's visible area throughout the mission defines the extent of the environment that is observed by the robot's exploration sensors.

Explored Area

The set $\mathbb{A}_{\mathbb{E}}$ corresponds to the area explored by the robot during the mission. This subset of \mathbb{R}^2 contains all the points sensed by the robot's embedded sensors as it navigates within the environment, and we have:

$$\mathbb{A}_{\mathbb{E}} = \bigcup_{t \in [0, T]} \mathbb{V}(t) \quad (5.3)$$

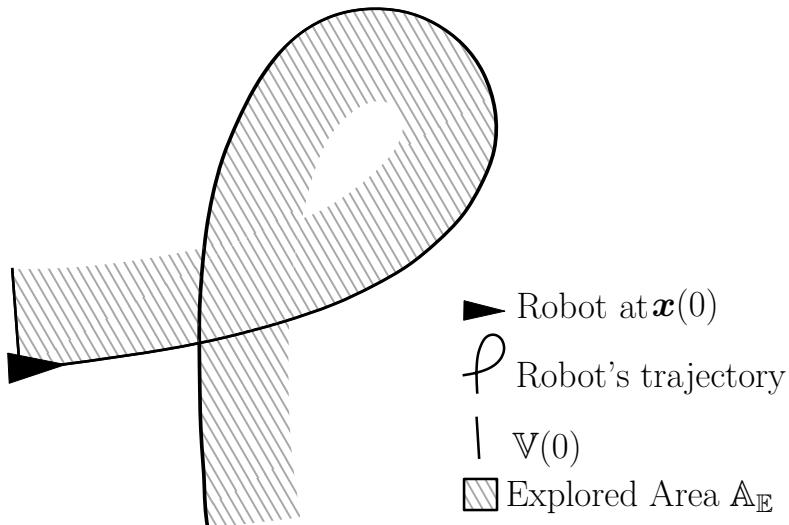


Figure 5.2: Area explored by a one-dimensional sensor on the robot's left side along its trajectory.

Figure 5.2 exemplifies the resultant $\mathbb{A}_{\mathbb{E}}$ if we consider the illustrated robot's trajectory and the visible area function described by Equation (5.2).

Waterfall and Mosaic Spaces

One-dimensional visible areas can be parameterized by a single parameter, this means that each point within the visible area can be uniquely identified or represented using one single variable. This parameter corresponds to the position along the linear axis of the visible area. For example, in Equation (5.2), this parameter is given by $l \in [0, L]$, and it determines the lateral distance of an explored point to the robot. As the robot sweeps the environment, the union of its one-dimensional visible areas over time creates a two-dimensional space that can be described as $[0, L] \times [0, T]$ for the entire mission duration. A resemblance can be observed between this space and the space used to portray the data forming the sonar's waterfall image, as explained in Section 2.2.1. In

the case of the waterfall image representation, the abscissa corresponds to the radial distance of a reflective point in the environment from the robot, while the ordinates-axis represents the exploration time. In this context, we name this space $W = [0, L] \times [0, T]$ and refer to it as the mission's waterfall space.

In sonar literature we usually represent data from the left-side sonar on the waterfall image with negative abscissa values. Therefore, in order to respect convention, for the example considered here, we assume $W = [-L, 0] \times [0, T]$ to be the mission's waterfall space.

All points $(u, t) \in W$ are points that were in the robot's visible area at least once and therefore, points that were explored during the mission, but represented in a relative coordinate frame. The robot's trajectory, its visible area and the explored area are all defined inside an absolute coordinate system. Still drawing a parallel with sonar terminology, we name the mission's absolute coordinate frame the Mosaic Space $M \subseteq \mathbb{R}^2$. In robotics, this space is usually called the world frame. In practice, the mosaic space represents, in the context of area covering missions, a parcel of the complete environment that we might want to explore.

We define the sweep function $\mathbf{f} : W \rightarrow M$ as a continuously differentiable function whose image over $[-L, 0] \times t$, with $t \in [0, T]$, represents the visible area $\mathbb{V}(t)$,

$$\mathbb{V}(t) = \mathbf{f}([-L, 0], t)$$

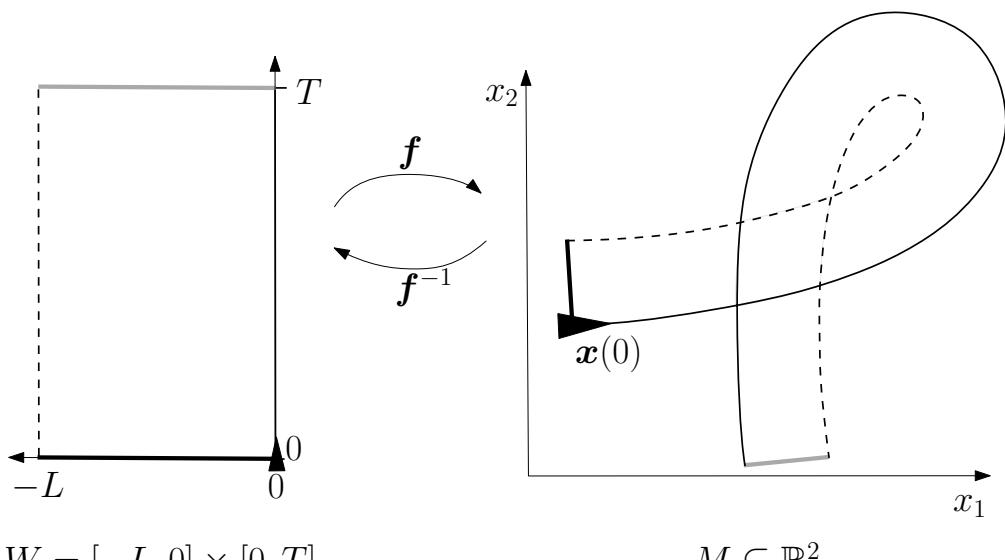
Moreover, we have,

$$\mathbb{A}_{\mathbb{E}} = \mathbf{f}(W)$$

Figure 5.3 illustrates the sweep function mapping.

Coverage Measure

When the robot encounters a point for the first time, it becomes part of the explored area. However, if the robot subsequently leaves the vicinity of that point and later returns to find the same point within its visible area again, that point is considered to be revisited.



$$W = [-L, 0] \times [0, T]$$

$$M \subseteq \mathbb{R}^2$$

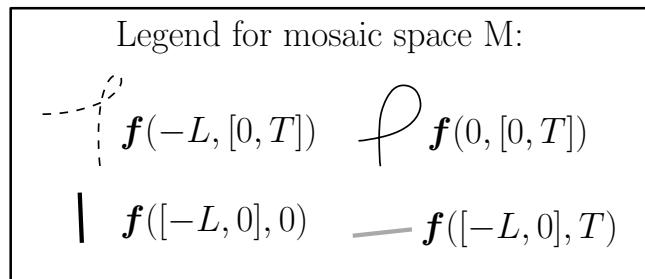


Figure 5.3: Waterfall and Mosaic Spaces related by the sweep function f . The image by f of $0 \times [0, T]$ represents the robot's trajectory on the mosaic space and $f([-L, 0], t)$ represents the visible area for some $t \in [0, T]$.

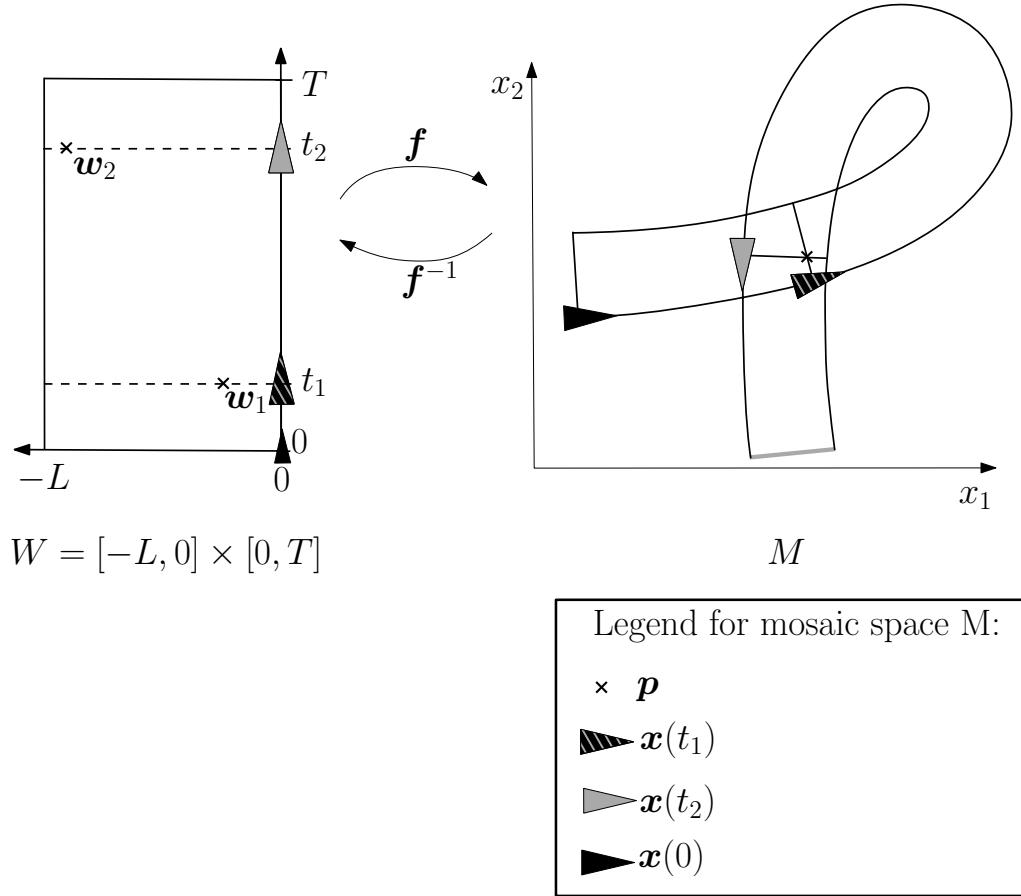


Figure 5.4: Point p is explored at instants t_1 and t_2 . Therefore, we have a point $w_1 \in W$ with ordinate value equal to t_1 and a point $w_2 \in W$ with ordinate value equal to t_2 . Both these points are mapped by f onto p .

The coverage measure, or how many times a point in the environment was explored by the robot during a mission, is given by the function $c_m : M \rightarrow \mathbb{N}$. In Figure 5.4, for example, point p is sensed for the first time at instant t_1 , and there is a point $w_1 = f(l, t_1)$ with $l \in [-L, 0]$ in the waterfall space. Then, the point is revisited at instant t_2 and therefore, it exists $w_2 = f(l, t_2)$ with $l \in [-L, 0]$ in the waterfall space.

Let \det be the determinant function and let J_f represent the Jacobian matrix of the sweep function. In a first moment we assume the following:

$$\forall \mathbf{w} \in W, \det(J_f(\mathbf{w})) > 0 \quad (5.4)$$

The determinant of the Jacobian matrix of a mapping function between two spaces represents the amount by which the mapping stretches or compresses the

source space as they are transformed into the target space. If the determinant of the Jacobian matrix is positive, it indicates that the mapping preserves orientation and, on the contrary, if the determinant is negative, it signifies that the mapping reverses orientation. In our context, a positive determinant of J_f throughout the time indicates that each new set of sensor readings, represented by a horizontal line in W , expands the size of the resultant explored area. Furthermore, it determines that $\mathbb{A}_{\mathbb{E}}$ can not be reduced or be constant during time, it is always expanding. Summarizing, this condition implies that the robot is constantly moving and that the sensor sweeps the environment on the same direction of the robot's advancement movement.

This hypothesis establishes, among others, that the number of times that a point appears in the waterfall space corresponds to the number of times that this point was explored during a mission. This is the case because, considering the one-dimensional character of the sensor, a point can not be inside the visible area for a continuous interval of time, only punctually. Therefore, for a point $\mathbf{p} \in M$, we have that $f^{-1}(\mathbf{p})$ is either an empty set or a finite set of individually unconnected points in W .

Thus, if $\text{Ker } f$ is the kernel of function f and for $\mathbf{p} \in M$, it can be concluded that

$$c_m(\mathbf{p}) = \#\text{Ker } (f - \mathbf{p}) \quad (5.5)$$

Finally, for the example presented in Figure 5.4, considering hypothesis in Equation (5.4), we have $f^{-1}(\mathbf{p}) = \{\mathbf{w}_1, \mathbf{w}_2\}$ and $c_m(\mathbf{p}) = 2$.

Coverage Measure and the Explored Area

The explored area $\mathbb{A}_{\mathbb{E}}$ can be defined as the set of points that were sensed by the robot at least once and therefore in terms of the coverage measure of its points:

$$\mathbb{A}_{\mathbb{E}} = \{\mathbf{p} \in M | c_m(\mathbf{p}) \geq 1\}$$

Describing the mosaic space using the coverage measure of its points is the method adopted in this work for defining the explored area. The following section establishes a connection between the topological degree and the coverage measure. This constitutes the primary contribution of this work, and it enables the development of an algorithm, presented in Chapter ??, for efficiently determining the coverage measure across an infinite number of points.

5.1.2 Coverage Measure and the Topological Degree

"The mathematical facts worthy of being studied are those which, by their analogy with other facts, are capable of leading us to the knowledge

of a physical law. They reveal the kinship between other facts, long known, but wrongly believed to be strangers to one another."

(- Henri Poincaré)

We recall from Section 4.8 that the topological degree is a function $\deg : (\mathbf{f}, S, \mathbf{p}) \rightarrow \mathbb{Z}$ where S is an open subset of \mathbb{R}^n , $\mathbf{f} : \bar{S} \rightarrow \mathbb{R}^n$ is continuous, and $\mathbf{p} \in \mathbb{R}^n \setminus \mathbf{f}(\partial S)$. This function respects the hypothesis established on Definition 27. Furthermore, if \mathbf{f} is continuously differentiable and $\det(J_{\mathbf{f}}(\mathbf{d}))$ is non-zero for each \mathbf{d} such that $\mathbf{d} \in \mathbf{f}^{-1}(\mathbf{p})$, Equation (4.1) can be used to compute the output value of this function. Then, let \mathbf{f} be a map from the two-dimensional disk to the plane, such that $\mathbf{f} : D^2 \rightarrow \mathbb{R}^2$, and $\mathbf{p} \in \mathbb{R}^2 \setminus \mathbf{f}(\partial D^2)$. Then, in Section 4.8.1 we prove that the winding number $\eta(\mathbf{f}(\partial D^2), \mathbf{p})$ is equal to $\deg(\mathbf{f}, D^2, \mathbf{p})$.

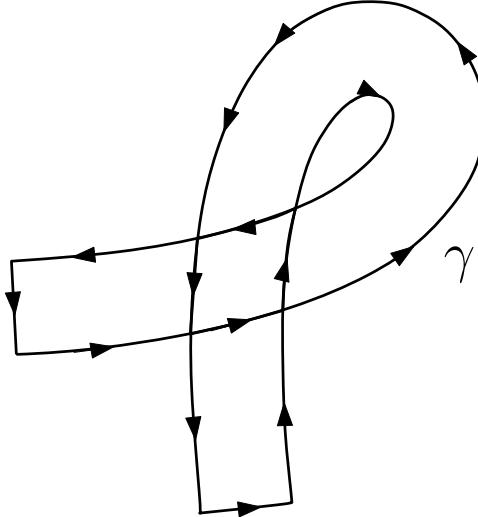


Figure 5.5: The sensor's contour γ is a counterclockwise oriented continuous closed curve on the plane.

Now let \mathbf{f} represent the sweep function, mapping from W , which is homeomorphic to D^2 , to the mosaic space M . We make the following proposition:

Proposition 1. *For any $\mathbf{p} \in M \setminus \mathbf{f}(\partial W)$:*

$$\eta(\gamma, \mathbf{p}) = c_m(\mathbf{p}), \quad (5.6)$$

where $\gamma = \mathbf{f}(\partial W)$ represents the mission contour. We choose γ to be counterclockwise oriented, and it is a closed curve that surrounds all the points that have been explored. In Figure 5.5 we illustrate the mission contour for the example mission illustrated on Figure 5.2.

Proof. Under hypothesis (5.4), we know that J_f is positive for all points in the Waterfall Space. Therefore, condition for using Equation (4.1) is met for any $\mathbf{p} \in M \setminus f(\partial W)$. Then we obtain

$$\deg(f, W, \mathbf{p}) = \sum_{w \in f^{-1}(\mathbf{p})} +1 = \#\text{Ker } (f - \mathbf{p})$$

Finally, from Equation (5.5), it can be concluded that $\deg(f, W, \mathbf{p}) = c_m(\mathbf{p})$. Since $\deg(f, W, \mathbf{p}) = \eta(f(\partial W), \mathbf{p})$, the proposition is proved. \square

On next sections we extend the relation between the coverage measure and the topological degree, so it comprehends more general scenarios.

Coverage Measure of Points with Undefined Winding Numbers

We assume, for the moment, that the robot's pose and its visible area are well-defined along the mission. Meaning that there is no uncertainty associated to them. In this case, it is expected that the coverage measure of all the points in the environment can be uniquely determined. However, if we use the winding number for estimating the coverage measure, as proposed in (5.6), the coverage measure of a point $\mathbf{p} \in \gamma$ will be undefined.

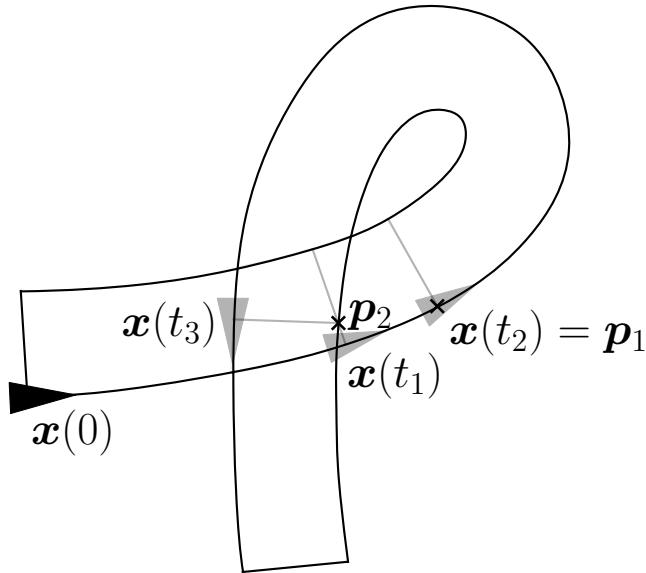


Figure 5.6: The coverage measure of point p_1 is equal to 1 and of point p_2 is equal to 2, but the winding number of γ with respect to these points is undefined.

For example, in Figure 5.6, point $\mathbf{p}_1 \in \gamma$ is the image by the sweep function \mathbf{f} of a point $(0, t_2) \in W$. This point is inside the robot's visible area $\mathbb{V}(t_1)$ and according to the definition of the coverage measure on Equation (5.5), $c_m(\mathbf{p}_1) = 1$, but $\eta(\gamma, \mathbf{p}_1)$ is undefined. We have a similar scenario for point $\mathbf{p}_2 \in \gamma$ that is the image by \mathbf{f} of a point $(-L, t_3) \in W$ but also of another point $(l, t_1) \in W$ with $l \in (-L, 0)$. Thus, we have $c_m(\mathbf{p}_2) = 2$ although $\eta(\gamma, \mathbf{p}_2)$ is undefined.

In this context, to extend the validity of (5.6), we define a bounded function $\bar{\eta}$ as the extension of the winding number function to the full domain $\mathbf{f}(W)$. For that, we consider the following definition, adapted from [120]:

Definition 29 (Upper Limit). *Let M be a metric space and g a function from M to \mathbb{R} . For any limit point (Definition 13) $\mathbf{y} \in M$ the upper limit, when it exists, is defined as:*

$$\limsup_{\mathbf{p} \rightarrow \mathbf{y}} g(\mathbf{p}) = \lim_{\epsilon \rightarrow 0} (\sup\{g(\mathbf{p}) \mid \mathbf{p} \in B(\mathbf{y}, \epsilon) \setminus \{\mathbf{y}\}\})$$

where $B(\mathbf{y}, \epsilon)$ denotes the ball within M , centered at \mathbf{y} , of radius ϵ .

The sweep function \mathbf{f} is a continuous map from a compact subset W to a subset of \mathbb{R}^2 , therefore $\mathbf{f}(W) \setminus \mathbf{f}(\partial W)$ is composed of a disjoint union of opens V_i , $i \in I$, for some index set I . All points of $\mathbf{f}(\partial W)$ are limits of some sequence of points $\mathbf{f}(\mathbf{y})$, with $\mathbf{y} \in \mathring{W}$. We can now state:

Theorem 5.1.1. *Consider a function $w : \bigcup_{i \in I} V_i \rightarrow \mathbb{Z}$. Suppose that w is bounded on $\bigcup_{i \in I} V_i$ then there is an upper semi-continuous extension of w , $\bar{w} : \mathbf{f}(W) \rightarrow \mathbb{Z}$ defined as:*

$$\bar{w}(\mathbf{p}) = \begin{cases} w(\mathbf{p}) & \text{if } \mathbf{p} \in \bigcup_{i \in I} V_i \\ \limsup_{\substack{\mathbf{p}' \in \bigcup_{i \in I} V_i \rightarrow \mathbf{p}}} w(\mathbf{p}') & \text{otherwise} \end{cases}$$

Proof. This is immediate: the upper limit exists since w is bounded on $\bigcup_{i \in I} V_i$, and the definition of \bar{w} precisely imposes that \bar{w} is upper semi-continuous. \square

Supposing that the number of connected components of $\mathbf{f}(W) \setminus \mathbf{f}(\partial W)$ is finite, as the winding number is constant on each component, this defines a bounded function η that we can extend to the full domain $\mathbf{f}(W)$ by Theorem 5.1.1 to obtain $\bar{\eta}$. Finally, if the condition expressed in Equation (5.4) is satisfied, we can say that for any $\mathbf{p} \in M$,

$$\bar{\eta}(\gamma, \mathbf{p}) = c_m(\mathbf{p}) \quad (5.7)$$

Considering Definition 29, if $\mathbf{p} \in \gamma$, its coverage measure will be equal to the coverage measure of points on the open V_i with the biggest winding number value for which \mathbf{p} is a limit. For example, point \mathbf{p}_2 in Figure 5.6 is the limit for two opens. One of the opens has a winding number value of +1, and the other a winding number value of +2. Therefore, according to (5.7), we obtain $c_m(\mathbf{p}_2) = 2$, as expected by the original definition on Equation (5.5).

This new definition extends the applicability of the method but condition (5.4) is still necessary for Equation (5.7) to be true. Next section introduces new concepts to remove this constraint.

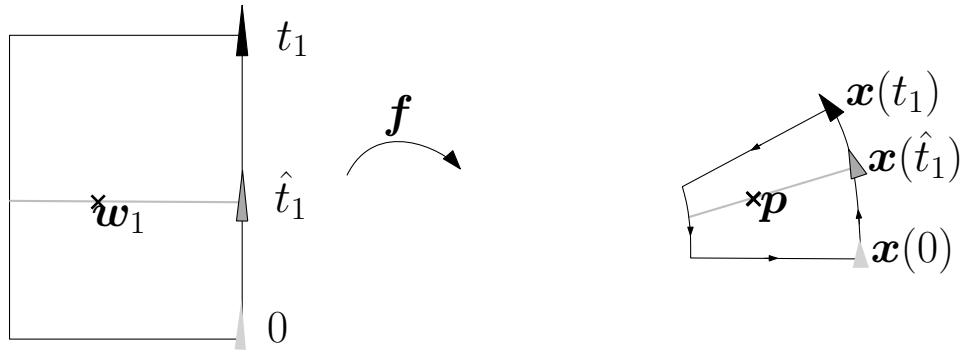
Coverage Measure for Points Swept Backwards

If the condition established on Equation (5.4) is not satisfied there are two scenarios to be considered:

1. $\exists(l, t) \in W$ such that $J_f(l, t) < 0$: This translates by a point $\mathbf{p} = f(l, t)$ in M that is explored by the robot at t but swept on the opposite direction with respect to the robot's advancement movement, defined by vector $(\dot{x}_1(t) \ \dot{x}_2(t))^T$.
2. $\exists(l, t) \in W$ such that $J_f(l, t) = 0$: Point $\mathbf{p} = f(l, t)$ in the mosaic space is called a non-regular value of f . Point (l, t) corresponds to a critical point in the waterfall space where the mapping loses its local injectivity. In practice, it implies that there might be a continuous path within W that is mapped by f onto \mathbf{p} and therefore Equation (5.5) is no longer valid for determining the coverage measure of this point. This problem arises for example if for some $t \in [0, T]$ we have $(\dot{x}_1(t) \ \dot{x}_2(t))^T = \mathbf{0}$.

In this Section, we extend the method in order to deal with a negative determinant of the Jacobian of the sweep function (scenario 1 presented on the list above). As a consequence, by continuity, there will also be points in W where this determinant is zero. However, we assume that these points are singularities that arise from a change in the sweeping direction. A cease on the robot's movement that would be represented by $(\dot{x}_1(t) \ \dot{x}_2(t))^T = \mathbf{0}$, for some $t \in [0, T]$, still will not be comprised by the resultant method.

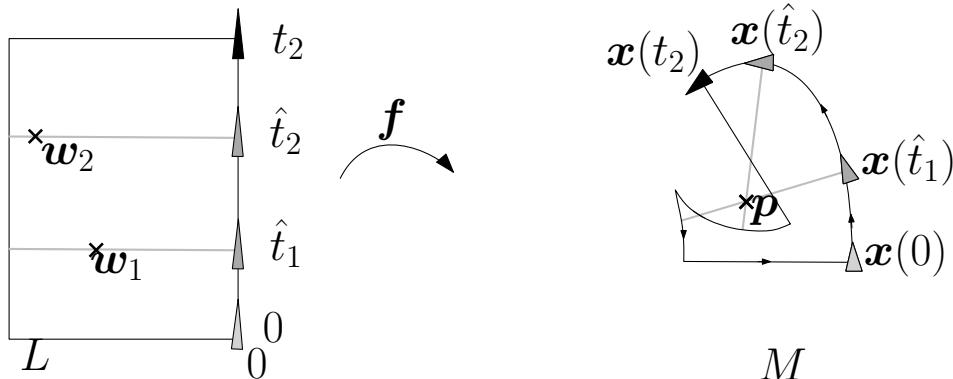
The example illustrated in Figure 5.7 demonstrates the inconsistency that arises in the equality presented in Equation (5.7) on a mission that does not respect condition (5.4).



$$W_1 = [-L, 0] \times [0, t_1]$$

$$\gamma_1 = \mathbf{f}(\partial W_1)$$

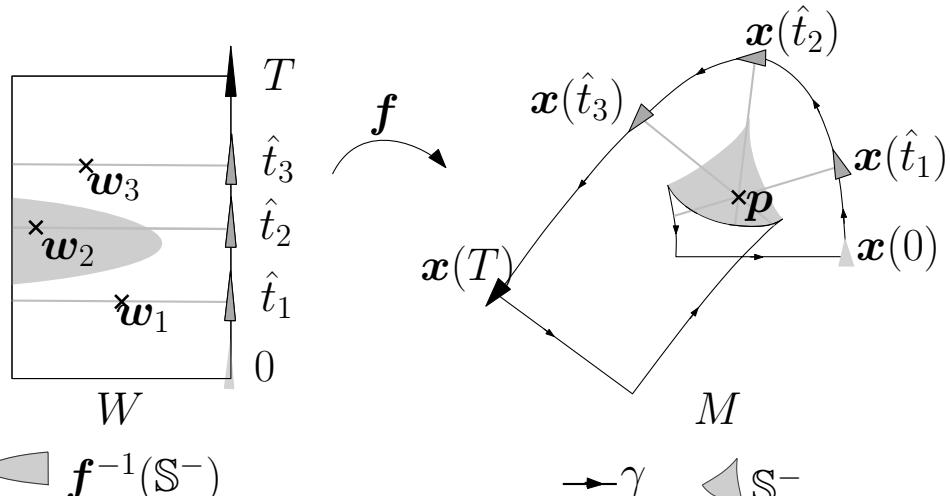
(a) Mission during time interval $[0, t_1]$, point p is sensed for the first time at \hat{t}_1 and $c_m(p) = 1$.



$$W_2 = [-L, 0] \times [0, t_2]$$

$$\gamma_2 = \mathbf{f}(\partial W_2)$$

(b) Point p is swept on the opposite direction of the robot's movement at \hat{t}_2 . At t_2 , $c_m(p) = 2$.



(c) The mission ends at T and the point p is sensed for the last time at \hat{t}_3 , the final coverage measure of this point is 3 although $\eta(\gamma, p) = 1$.

Figure 5.7: Example of mission that does not satisfy condition (5.4).

At the beginning of the mission, in Figure 5.7a, the robot moves from its initial position $\mathbf{x}(0)$ to position $\mathbf{x}(t_1)$, $t_1 > 0$. During the interval $[0, t_1]$, condition (5.4) is satisfied. Point $\mathbf{p} \in M$ is sensed for the first time at instant $\hat{t}_1 \in [0, t_1]$ and this occurrence is represented in the mission's Waterfall Space W by point \mathbf{w}_1 . The sensor's contour associated with this first part of the mission is the closed curve $\gamma_1 = \mathbf{f}(\partial([-L, 0] \times [0, t_1]))$ and we have:

$$\eta(\gamma_1, \mathbf{p}) = \text{sign}(\det(J_{\mathbf{f}}(\mathbf{w}_1))) = 1$$

that is indeed equal to the coverage measure of \mathbf{p} at t_1 .

The mission continues as the robot advances to position $\mathbf{x}(t_2)$, $t_2 > t_1$ and point \mathbf{p} is revisited at \hat{t}_2 . For the time interval $[0, t_2]$, we have $\mathbf{f}^{-1}(\mathbf{p}) = \{\mathbf{w}_1, \mathbf{w}_2\}$ and $\gamma_2 = \mathbf{f}(\partial([-L, 0] \times [0, t_2]))$ represents the sensor's contour. As illustrated in Figure 5.7b, at \hat{t}_2 , point \mathbf{p} is swept with the opposite direction with respect to the robot's advancement movement. In this context, the Jacobian of function \mathbf{f} at \mathbf{w}_2 is negative and

$$\eta(\gamma_2, \mathbf{p}) = \sum_{i=1}^2 \text{sign}(\det(J_{\mathbf{f}}(\mathbf{w}_i))) = 1 - 1 = 0$$

although, according to (5.5), $c_m(\mathbf{p}) = 2$ at t_2 .

Exploration ends at position $\mathbf{x}(T)$, $T > t_2$ and the complete mission is represented in Figure 5.7c. Point \mathbf{p} is sensed for the third and last time at \hat{t}_3 and at the end of the mission $\mathbf{f}^{-1}(\mathbf{p}) = \{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\}$. At \hat{t}_3 , point \mathbf{p} is sensed by a forward movement of the sensor on the plane, therefore we have

$$\eta(\gamma, \mathbf{p}) = \sum_{i=1}^3 \text{sign}(\det(J_{\mathbf{f}}(\mathbf{w}_i))) = 1 - 1 + 1 = 1$$

but $c_m(\mathbf{p}) = 3$ is expected.

To address this problem, we can divide the Waterfall Space W into two sets, \mathbb{S}^+ and \mathbb{S}^- ,

$$\mathbb{S}^+ = \{\mathbf{w} \in W \mid \det(J_{\mathbf{f}}(\mathbf{w})) > 0\} \quad (5.8)$$

$$\mathbb{S}^- = \{\mathbf{w} \in W \mid \det(J_{\mathbf{f}}(\mathbf{w})) < 0\} \quad (5.9)$$

We define two new positively oriented contours, γ^+ and γ^- as the image by \mathbf{f} of the boundaries of these sets:

$$\gamma^+ = \mathbf{f}(\partial\mathbb{S}^+) \quad (5.10)$$

$$\gamma^- = \mathbf{f}(\partial\mathbb{S}^-) \quad (5.11)$$

Figure 5.8 illustrates this division.

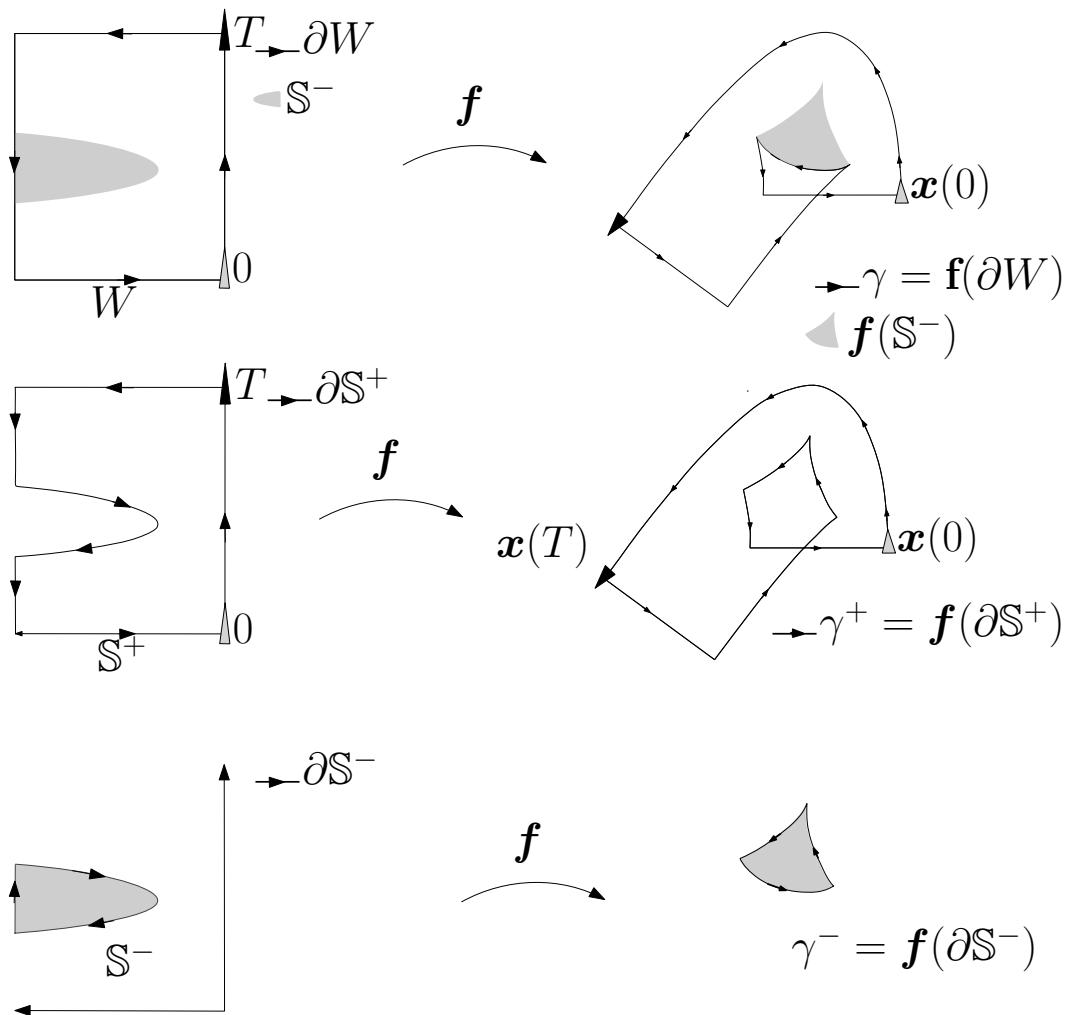


Figure 5.8: Decomposition of the Waterfall Space and γ according to the sweeping direction.

For a regular value $\mathbf{p} \in M$ we will have $\text{Ker } (\mathbf{f} - \mathbf{p}) \subset \mathbb{S}^+ \cup \mathbb{S}^-$ and

$$\text{Ker } (\mathbf{f} - \mathbf{p}) = \text{Ker}(\mathbf{f}_{|\mathbb{S}^+} - \mathbf{p}) \cup \text{Ker}(\mathbf{f}_{|\mathbb{S}^-} - \mathbf{p})$$

Equation (5.5) can also be rewritten:

$$c_m(\mathbf{p}) = \#\text{Ker } (\mathbf{f}_{|\mathbb{S}^+} - \mathbf{p}) + \#\text{Ker } (\mathbf{f}_{|\mathbb{S}^-} - \mathbf{p})$$

The cardinality of a set can be obtained by a sum of +1 through its elements:

$$c_m(\mathbf{p}) = \sum_{\mathbf{w} \in \mathbf{f}_{|\mathbb{S}^+}^{-1}(\mathbf{p})} +1 + \sum_{\mathbf{w} \in \mathbf{f}_{|\mathbb{S}^-}^{-1}(\mathbf{p})} +1$$

and considering the definitions of sets \mathbb{S}^+ and \mathbb{S}^- , respectively on (5.8) and (5.9), we obtain

$$c_m(\mathbf{p}) = \sum_{\mathbf{w} \in \mathbf{f}_{|\mathbb{S}^+}^{-1}(\mathbf{p})} \text{sign}(\det(J_{\mathbf{f}})(\mathbf{w})) - \sum_{\mathbf{w} \in \mathbf{f}_{|\mathbb{S}^-}^{-1}(\mathbf{p})} \text{sign}(\det(J_{\mathbf{f}})(\mathbf{w}))$$

Finally, from (5.10), (5.11), (4.1) and Definition 28, we have

$$c_m(\mathbf{p}) = \bar{\eta}(\gamma^+, \mathbf{p}) + \bar{\eta}(\gamma^-, \mathbf{p}) \quad (5.12)$$

for any regular point $\mathbf{p} \in M$.

Note that all non-regular values $\mathbf{p} \in M$, representing a singularity to the method, lay on the intersection of curves γ^+ and γ^- . In this context, their coverage measure is naturally defined considering the upper semi-continuous extension of the winding number $\bar{\eta}$.

5.1.3 Dealing with Uncertainties

It is well known that estimating the position of robotic systems involves dealing with uncertainty due to various factors that can affect the accuracy and reliability of the final estimation. In order to provide a guaranteed characterization of the environment in terms of the coverage measure of its points, uncertainty needs to be taken into account on the formalization of the notion of coverage measure.

In this context, we now consider that the robot's pose can be uncertain, but we keep the assumption that the sensor's model is exact. Nonetheless, since the visible area depends on the robot's pose, uncertainty is naturally propagated to the coverage measure.

Let \mathbf{x}^* be the function representing the robot's trajectory on the plane during a mission. From now on, we assume that \mathbf{x}^* is unknown and that instead,

\mathbf{x}^* belongs to a set $\mathbb{S}_x \in \mathcal{P}(\mathbb{R} \rightarrow \mathbb{R}^2)$ of all the possible functions describing the robot's behavior. Modeling the behavior of a mobile robot by a set of possible solutions containing the ground truth is a common approach since they are usually non-holonomic systems. These are systems whose behavior can be modeled by differential equations and physical constraints, implying that if bounded uncertainties are introduced, they create a bounded disturbance around the real solution.

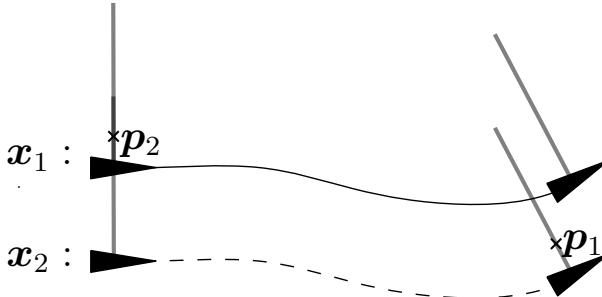


Figure 5.9: Point \mathbf{p}_1 is explored only if $\mathbf{x}^* = \mathbf{x}_2$ and point \mathbf{p}_2 is explored either if $\mathbf{x}^* = \mathbf{x}_1$ or $\mathbf{x}^* = \mathbf{x}_2$.

The coverage measure $c_m(\mathbf{p})$ for a point $\mathbf{p} \in M$ can take different values for distinct functions $\mathbf{x} \in \mathbb{S}_x$. In this work, we propose a solution for computing the uncertain coverage measure based on interval analysis. For example, let us consider a set $\mathbb{S}_x = \{\mathbf{x}_1, \mathbf{x}_2\}$ with two possible solutions as illustrated in Figure 5.9. The coverage measure of \mathbf{p}_1 can either be 0 or 1. In this case, we want its coverage measure to be represented by an interval $[0, 1]$ containing all the possible solutions. For point \mathbf{p}_2 , its coverage measure is always equal to 1. Therefore, we represent its coverage measure by the singleton $[1, 1]$.

We adopt the notation $[c_m]_{|\mathbf{x}}(\mathbf{p})$ for representing the coverage measure of a point $\mathbf{p} \in M$ for a given \mathbf{x} . We are interested in estimating $[c_m](\mathbf{p}) \in \mathbb{IZ}$, the smallest interval of relative integers such that

$$[c_m](\mathbf{p}) = \bigsqcup_{\mathbf{x} \in \mathbb{S}_x} [c_m]_{|\mathbf{x}}(\mathbf{p}) \quad (5.13)$$

As it will be discussed on the next chapter, in practice, we have to deal with the propagation of pessimism. In this case, $[c_m]$ is ideally the smallest interval to represent the set of solutions, but this will not always be the case.

From each $\mathbf{x} \in \mathbb{S}_x$, we can generate a different γ , a possible sensor's contour for the mission. Then, we define $\mathbb{S}_\gamma \in \mathcal{P}(S^1 \rightarrow \mathbb{R}^2)$ as the set of all possible γ . Furthermore, the sweep function \mathbf{f} will also depend on \mathbf{x} , and we denote \mathbf{f}_x the sweep function associated to some function $\mathbf{x} \in \mathbb{S}_x$. Then, we have the set

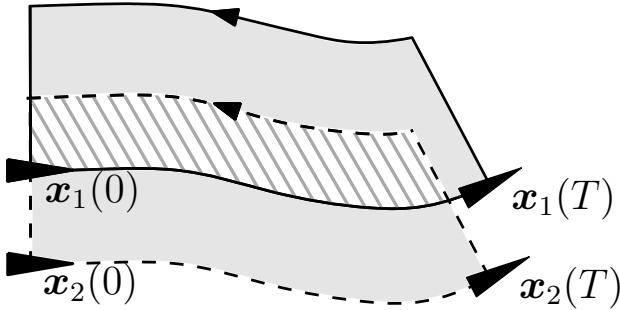


Figure 5.10: The solid line represents the sensor's contour γ associated to the robot's trajectory represented by function \mathbf{x}_1 . The dashed line represents γ for the robot's trajectory represented by function \mathbf{x}_2 . The hatched area contains points surrounded by both contours and therefore points that have certainly been explored. Gray areas represent points that could have been explored depending on which of the trajectories is the ground truth.

$\mathbb{S}_f \in \mathcal{P}(W \rightarrow \mathbb{R}^2)$ such that

$$\mathbb{S}_f = \bigcup_{\mathbf{x} \in \mathbb{S}_x} f_{\mathbf{x}} \quad (5.14)$$

To simplify the definitions in this section, first we consider a point $\mathbf{p} \in M$ such that for any $\mathbf{x} \in \mathbb{S}_x$, we have $\det(J_{f_x}(\mathbf{w})) > 0$ for all $\mathbf{w} \in f_x^{-1}(\mathbf{p})$. According to (5.7), we can obtain the coverage measure through the computation of the winding number of the sensor's contour. Therefore, we want to determine $[\bar{\eta}](\mathbb{S}_{\gamma}, \cdot) \in \mathbb{IZ}$, that is ideally the smallest interval such that

$$\forall \gamma \in \mathbb{S}_{\gamma}, \bar{\eta}(\gamma, \cdot) \in [\bar{\eta}](\mathbb{S}_{\gamma}, \cdot) \quad (5.15)$$

and we can define the uncertain coverage measure of \mathbf{p} as

$$[c_m](\mathbf{p}) = [\bar{\eta}](\mathbb{S}_{\gamma}, \mathbf{p}). \quad (5.16)$$

Figure 5.10 presents set \mathbb{S}_{γ} considering the example presented on the beginning of this section, we have $\mathbb{S}_{\gamma} = \{\gamma_1, \gamma_2\}$ where γ_1 is the sensor's contour associated to trajectory \mathbf{x}_1 and γ_2 to trajectory \mathbf{x}_2 from Figure 5.9.

A generalization for points swept backwards can be directly obtained considering a decomposition of each of the cycles $\gamma \in \mathbb{S}_{\gamma}$ in γ^+ and γ^- . This decomposition results in sets $[\gamma^+] \in \mathcal{P}(S^1 \rightarrow \mathbb{R}^2)$ and $[\gamma^-] \in \mathcal{P}(S^1 \rightarrow \mathbb{R}^2)$. Equation (5.12) can then be adapted to the uncertain coverage measure, we obtain:

$$[c_m](\mathbf{p}) = [\bar{\eta}](\gamma^+, \mathbf{p}) + [\bar{\eta}](\gamma^-, \mathbf{p}) \quad (5.17)$$

In the initial section of this chapter, we proposed and proved Proposition 1, that uses the concept of winding number to compute the coverage measure of points in the environment. We demonstrated that our approach has the capability to account for points that are swept in a direction opposite to the robot's forward movement, as well as to accommodate uncertainties in the robot's pose throughout the mission. The remaining of this chapter expands upon the formalization introduced in this initial part to address sensors with two-dimensional visible areas.

5.2 Two-Dimensional Sensors

5.2.1 Problem Statement

We return to the assumption that the robot's pose is known through the mission and that it can be represented by $(\mathbf{x} \ \psi)^T \in \mathbb{R}^3$, with $\mathbf{x} : [0, T] \rightarrow \mathbb{R}^2$ at least C^2 and $\psi : [0, T] \rightarrow \mathbb{R}$ at least C^1 . The robot employs integrated two-dimensional exteroceptive sensors, like cameras for example, that allow it to explore a portion of its surroundings defined by the sensor's visibility range. Unlike one-dimensional sensors, that provide data along a single axis, a local two-dimensional representation of the observed environment is created around the robot at each instant.

The primary challenge encountered when attempting to generalize the notions presented on the first part of this chapter to this new configuration, lies in formally defining the coverage measure while employing two-dimensional sensors. Given the two-dimensional nature of the observable area, a point within the environment that has been explored only once during a mission could potentially remain within the robot's visible area for a continuous time interval. As a result, the coverage measure definition provided in (5.5) is no longer valid because there is possibly an infinite number of instants for which this point would be inside the visible area. Furthermore, Equation (5.5) is necessary for establishing a relation between the exploration problem and the notion of topological degree, as presented in (5.6) and on both of its extensions in (5.7) and (5.12). In this context, a more general definition of the coverage measure is proposed to deal with two-dimensional sensors and its relation to the topological degree is proved again for this new scenario.

Visible Area

We define the robot's visible area as a set-valued function $\mathbb{V} : [0, T] \rightarrow \mathcal{P}(\mathbb{R}^2)$ that respects the following assumption:

Assumption 1 We suppose that $\mathbb{V}(t)$, for $t \in [0, T]$, is a differentiable manifold with boundary, diffeomorphic to the disk D^2 with differential structure inherited from \mathbb{R}^2 by the obvious embedding $D^2 \subseteq \mathbb{R}^2$.

As a consequence, the boundary $\partial\mathbb{V}(t)$ of $\mathbb{V}(t)$ is diffeomorphic, as a manifold without boundary, to S^1 , the unit circle of \mathbb{R}^2 .

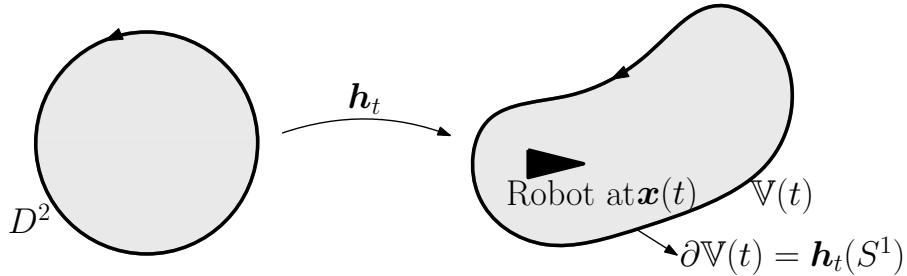


Figure 5.11: The robot's visible area is diffeomorphic to D^2 . We denote h_t the diffeomorphism between these two spaces for an instant $t \in [0, T]$.

We note the orientation preserving diffeomorphism

$$h_t : D^2 \rightarrow \mathbb{V}(t) \quad (5.18)$$

We suppose it varies continuously in time t and that D^2 is counterclockwise oriented. Figure 5.11 illustrates an example of visible area diffeomorphic to D^2 .

Explored Area

The set $\mathbb{A}_{\mathbb{E}}$ is the union of the robot's visible area through the mission, as it was defined for the one-dimensional context (5.3).

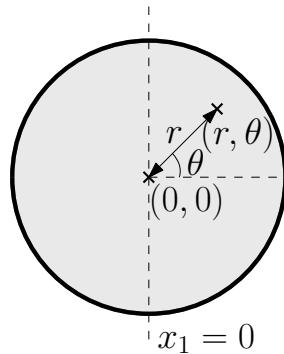


Figure 5.12: Polar representation of points on disk D^2 .

We use a simple example for illustrating the resultant explored area in the context of two-dimensional visible areas. For that, we parameterize the disk D^2 with polar coordinates of the form $\{(r, \theta) \mid r \in [0, 1], \theta \in [0, 2\pi)\}$. This parameterization is illustrated on Figure 5.12.

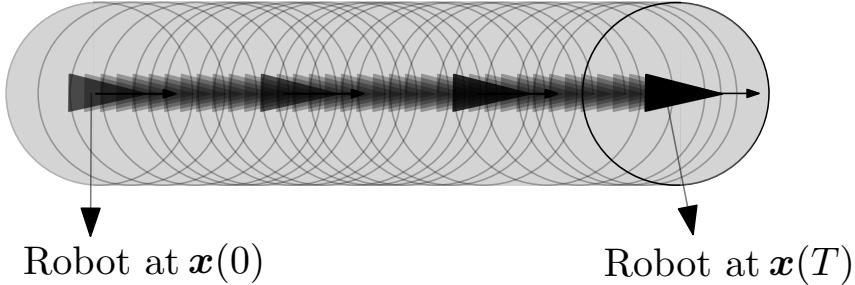


Figure 5.13: In gray, the area explored by a mobile robot moving straight forward during time interval $[0, T]$ with a circular visible area.

Then, we assume a scenario where the robot performs a forward movement in the environment with a speed vector

$$\mathbf{g}(\mathbf{x}, \psi) = \begin{pmatrix} \dot{x}_1 \\ 0 \end{pmatrix} \quad (5.19)$$

where $\dot{x}_1 > 0$ is constant. The visible area $\mathbb{V}(t)$ for $t \in [0, T]$ is a simple translation of D^2 to be centered at $(x_1(t), x_2(t))$, and we have

$$\mathbf{h}_t(r, \theta) = \begin{pmatrix} x_1(t) + r \cos(\theta) \\ x_2(t) + r \sin(\theta) \end{pmatrix} \quad (5.20)$$

Figure 5.13 illustrates the area explored by the robot on this context.

Waterfall and Mosaic Spaces

The mosaic space $M \subset \mathbb{R}^2$ in this new context is the same as the one we had on the one-dimensional case. The equivalent of the waterfall space is constructed as follows.

First, consider $I = D^2 \times [0, T]$, the origin domain that is mapped by a differentiable function $\mathbf{f} : D^2 \times [0, T] \rightarrow M$, with M representing the global reference frame for the mission, or simply the mosaic space. We note

$$\mathbf{f}(\mathbf{d}, t) = \mathbf{h}_t(\mathbf{d}) \quad (5.21)$$

with $\mathbf{d} = [0, 1] \times [0, 2\pi)$. This mapping is illustrated on Figure 5.14.

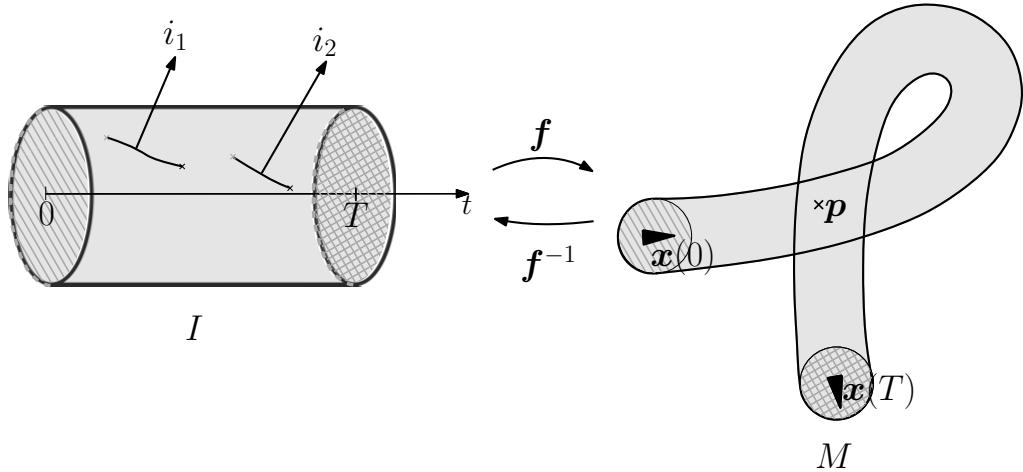


Figure 5.14: Mapping from a local coordinate frame I to the world frame, that is represented by the mosaic space M . We assume a circular visible area centered on the robot's position and rotated according to the robot's orientation on the plane. Point $\mathbf{p} \in M$ is explored two times during the mission. In this case, we have $f^{-1}(\mathbf{p})$ that is a non-connected set of points made of the union of subsets i_1 and i_2 .

Space I represents points that were explored during the mission in a relative coordinate frame, and, in a certain manner, the waterfall space notion that we previously had. For the one-dimensional context, however, we naturally had a local injectivity of function \mathbf{f} . This was a result of not having connected paths within the waterfall space that could be mapped by \mathbf{f} to the same point on the mosaic space. In order to construct a waterfall space with this property on the two-dimensional scenario we have to quotient I by the following equivalence relation \sim , to obtain $W = I / \sim$:

$$(r, \theta, t) \sim (r', \theta', t') \quad (5.22)$$

if a continuous path from (r, θ, t) to (r', θ', t') within I is a subset of $f^{-1}(f(r, \theta, t))$.

At first, this quotient space seems similar to the one that we had in the one-dimensional scenario. However, in general, it can be challenging to characterize and to define the properties of a quotient space. Therefore, we prefer to adopt a different approach, when defining the notion of coverage measure, independent of the notion of the waterfall space. This new approach, that is now presented, simplify our future proofs when reestablishing a relation between the coverage measure and the topological degree.

Coverage Measure

We recall that a point is only considered to be revisited if once in the robot's visibility range, it goes out of reach and then is sensed again later in time.

We can then define the following set, for all points \mathbf{p} on the set $\mathbb{A}_{\mathbb{E}}$ associated to the mission:

$$T_{\mathbf{p}} = \{t \in [0, T] \mid \exists (r, \theta) \in [0, 1] \times [0, 2\pi), \mathbf{h}_t(r, \theta) = \mathbf{p}\}$$

Still identifying points in D^2 by their polar coordinates, this set contains all the instants when point $\mathbf{p} \in \mathbb{A}_{\mathbb{E}}$ is inside the robot's visible area.

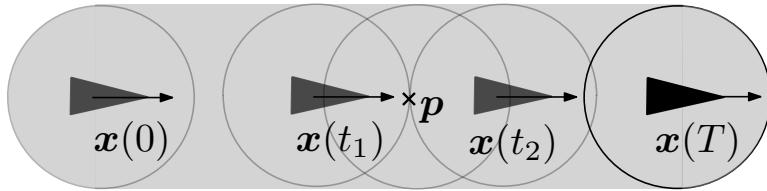


Figure 5.15: In this example, the illustrated point \mathbf{p} enters the robot's visible area at instant t_1 and leaves it at instant t_2 . During time interval $[t_1, t_2]$ the point stays on the robot's visibility range.

We consider again the context defined for the mission presented on Figure 5.13, and we take point \mathbf{p} as illustrated on Figure 5.15. This point is on the robot's visible area $\mathbb{V}(t)$ for all $t \in [t_1, t_2]$. Therefore, we have

$$T_{\mathbf{p}} = \{t \in [0, T] \mid t_1 \leq t \leq t_2\} = [t_1, t_2]$$

and $T_{\mathbf{p}}$ is a closed subset of $[0, T]$.

In more complex examples, $T_{\mathbf{p}}$ will be a union of connected components, that are closed subsets of $[0, T]$. The set $T_{\mathbf{p}}$ will be composed of finitely many disjoint intervals of the form $[a_i, b_i]$ with $a_i, b_i \in [0, T]$ and i indexed by positive integers.

Figure 5.16 presents another example of mission, now assuming a diffeomorphism defined by

$$\mathbf{h}_t(r, \theta) = \begin{pmatrix} x_1(t) + r \cos(\theta + \psi(t)) \\ x_2(t) + r \sin(\theta + \psi(t)) \end{pmatrix} \quad (5.23)$$

for $r \in [0, 1]$ and $\theta \in [0, 2\pi)$. In this new mission, we have associated to the illustrated point \mathbf{p} the set

$$T_{\mathbf{p}} = \{t \in [0, T] \mid t_1 \leq t \leq t_2 \text{ or } t_3 \leq t \leq t_4\} = [t_1, t_2] \cup [t_3, t_4]$$

Moreover, we present the following definition:

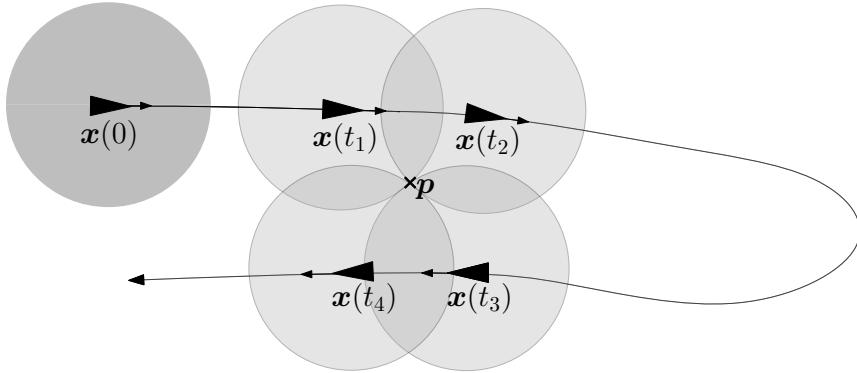


Figure 5.16: Point p is explored twice by the robot during the illustrated mission. It enters the visible area at instant t_1 and t_3 , and it leaves it at t_2 and t_4 , staying on the visible area for two continuous time intervals $[t_1, t_2]$ and $[t_3, t_4]$.

Definition 30 (Coverage Measure). *The number of connected components of T_p defines what we call “the number of times our robot sees p ”, or its coverage measure $c_m(p)$.*

For point p on Figure 5.15, the set T_p associated to this point has only one connected component, therefore $c_m(p) = 1$. And for point p on Figure 5.16 we have $c_m(p) = 2$.

The topological approach established for one-dimensional sensors needs to be proved valid for the case of the two-dimensional visible area, considering this new, more general, definition of the coverage measure. For that we define three sets that will be important in this context: the entrance, exit and contour sets.

Entrance, Exit and Contour Sets

Before defining these sets we define $\mathbf{g}_t(p) \in \mathbb{R}^2$ as the opposite vector flow for a point $p \in M$ at instant t . It represents the point's relative speed in the robot's perspective. Let \mathbf{p}_t^r be the representation of a point $p = (p_1, p_2)$ in the robot's coordinate frame at some instant t , then we have

$$\mathbf{p}_t^r = \begin{pmatrix} \cos(\psi(t))(p_1 - x_1(t)) + \sin(\psi(t))(p_2 - x_2(t)) \\ -\sin(\psi(t))(p_1 - x_1(t)) + \cos(\psi(t))(p_2 - x_2(t)) \end{pmatrix} \quad (5.24)$$

The opposite vector flow of this point is

$$\mathbf{g}_t(p) = \frac{\partial \mathbf{p}_t^r}{\partial t} \quad (5.25)$$

The entrance set at time t is denoted $\mathbb{E}(t)$, and contains all points entering the visible area at instant t . It can be defined as a subset of $\partial\mathbb{V}(t)$ such that the opposite vector flow $\mathbf{g}_t(\mathbf{p})$ is pointing towards the interior of $\mathbb{V}(t)$ for all $\mathbf{p} \in \mathbb{E}(t)$.

The boundary $\partial\mathbb{V}(t)$ is defined by points with polar coordinates of the form $(1, \theta)$, with $\theta \in [0, 2\pi)$ through the orientation preserving diffeomorphism \mathbf{h}_t . In this case, entrance sets can be deduced from the computation of the determinant of the Jacobian of $(t, \theta) \rightarrow \mathbf{h}_t(1, \theta)$:

$$\det J_{\mathbf{h}} = \begin{vmatrix} \frac{\partial h_t^x}{\partial t}(1, \theta) & \frac{\partial h_t^x}{\partial \theta}(1, \theta) \\ \frac{\partial h_t^y}{\partial t}(1, \theta) & \frac{\partial h_t^y}{\partial \theta}(1, \theta) \end{vmatrix} \quad (5.26)$$

writing $\mathbf{h}_t = (h_t^x, h_t^y)$. Then we have:

$$\mathbb{E}(t) = \{\mathbf{h}_t(1, \theta) \mid \det J_{\mathbf{h}}(1, \theta) > 0\}$$

Similarly, we define $\mathbb{X}(t)$, the exit set at time t :

$$\mathbb{X}(t) = \{\mathbf{h}_t(1, \theta) \mid \det J_{\mathbf{h}}(1, \theta) < 0\}$$

and $\mathbb{C}(t)$ the contour set is:

$$\mathbb{C}(t) = \{\mathbf{h}_t(1, \theta) \mid \det J_{\mathbf{h}}(1, \theta) = 0\}$$

Figure 5.17 illustrates the vector field \mathbf{g}_{t_1} considering the context presented for the mission illustrated on Figure 5.15 at instant t_1 . For this example we have

$$\mathbf{g}_{t_1}(\mathbf{p}) = \begin{pmatrix} -\dot{x}_1(t_1) \\ 0 \end{pmatrix} \quad (5.27)$$

for any point $\mathbf{p} \in M$. We can see the illustrated point \mathbf{p} , that represents the same point in Figures 5.17 and 5.15, entering the robot's visible area, as expected on the example at instant t_1 . All the other points on $\partial\mathbb{V}(t_1)$ with vectors pointing towards the interior of the visible area are also entering the visible area at this instant.

Now, consider the following example:

Example 1. We assume a Dubins vehicle with unit linear velocity and constant angular velocity $\dot{\psi} = a \in \mathbb{R}$. We have

$$\mathbf{g}(\mathbf{x}(t), \psi(t)) = \begin{pmatrix} \cos(\psi(t)) \\ \sin(\psi(t)) \end{pmatrix}$$

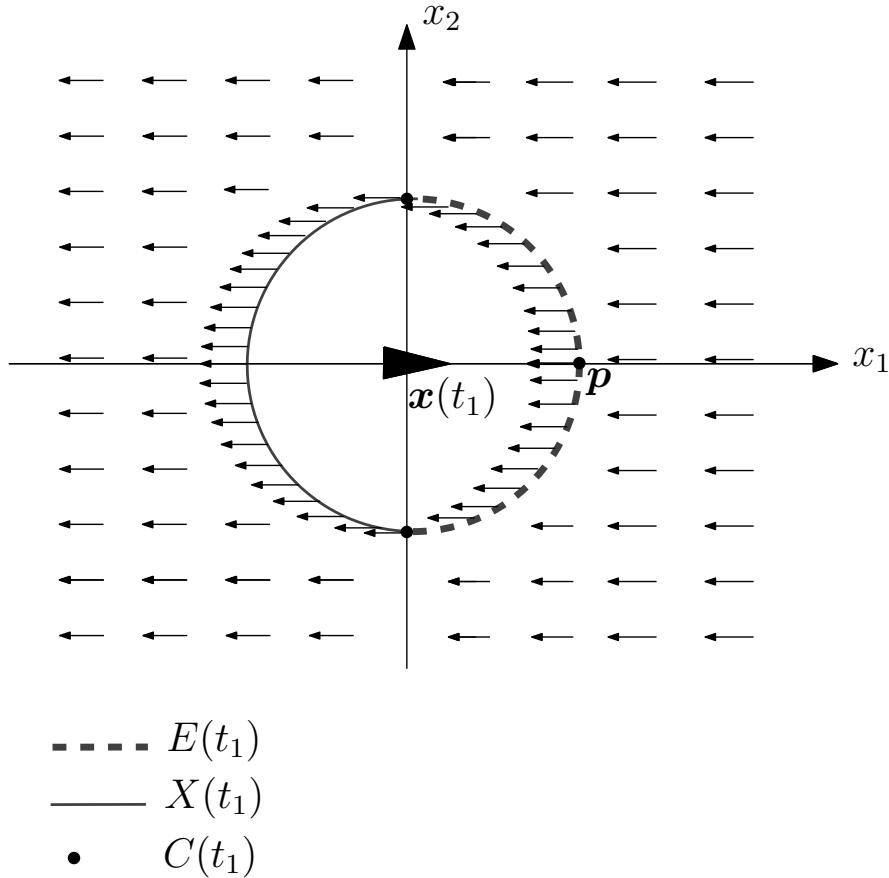


Figure 5.17: Robot represented at instant t_1 considering the context of the mission illustrated on Figure 5.15. The vector field \mathbf{g}_{t_1} is also represented. The dashed curved is composed of points on the boundary of the visible area $\mathbb{V}(t_1)$ entering the robot's visible range. The solid line contains points on the boundary leaving the visible area at instant t_1 .

The visibility region is a disk of radius 1, parameterized as follows:

$$\begin{pmatrix} v_x(r, \theta) \\ v_y(r, \theta) \end{pmatrix} = \begin{pmatrix} r \cos(\theta) \\ r \sin(\theta) \end{pmatrix}$$

The visibility region is, at all times t , a translation of the disk of radius 1, oriented at angle $\psi(t)$ and centered at $(x_1(t), x_2(t))$. Overall, this gives a parameterization represented by the diffeomorphism in (5.23).

We first compute the contour set $\mathbb{C}(t)$. For that, in order to compute (5.26), we define the following derivatives of $\mathbf{h}_t(1, \theta)$ with respect to θ and t :

$$\frac{\partial \mathbf{h}_t}{\partial \theta}(1, \theta) = \begin{pmatrix} -\sin(\theta + \psi(t)) \\ \cos(\theta + \psi(t)) \end{pmatrix}$$

$$\frac{\partial \mathbf{h}_t}{\partial t}(1, \theta) = \begin{pmatrix} \cos(\psi) - \sin(\theta + \psi(t))a \\ \sin(\psi) + \cos(\theta + \psi(t))a \end{pmatrix}$$

The determinant of these two vectors is $\det J_h(1, \theta)$:

$$\begin{aligned} & \left| \begin{array}{cc} \cos(\psi(t)) - \sin(\theta + \psi(t))a & -\sin(\theta + \psi(t)) \\ \sin(\psi(t)) + \cos(\theta + \psi(t))a & \cos(\theta + \psi(t)) \end{array} \right| \\ &= \cos(\psi(t)) \cos(\theta + \psi(t)) + \sin(\psi(t)) \sin(\theta + \psi(t)) \\ &= \cos((\theta + \psi(t)) - \psi(t)) \\ &= \cos(\theta) \end{aligned}$$

The contour set is given by $\det J_h(1, \theta) = 0$, for $\theta \in [0, 2\pi]$, hence the contour set is given by two points: $\mathbf{c}_1(t) = \mathbf{h}_t(1, \pi/2)$ and $\mathbf{c}_2(t) = \mathbf{h}_t(1, 3\pi/2)$.

We now make two assumptions that will be necessary to formalize a relation between the topological degree and the coverage measure of two-dimensional sensors. In what follows, we will write $S^+ \subseteq S^1$ (resp. $S^- \subseteq S^1$) for the strict eastern (resp. western) hemisphere, i.e. points (x_1, x_2) of S^1 such that $x_1 > 0$ (resp. $x_1 < 0$).

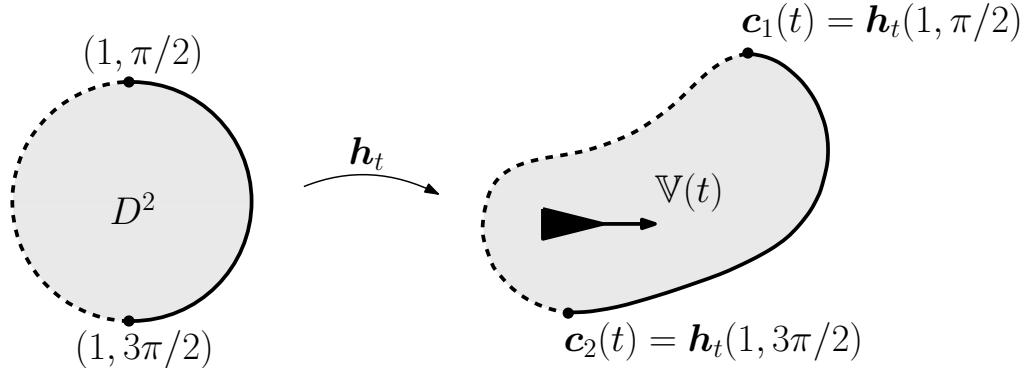


Figure 5.18: The robot is illustrated at some position $\mathbf{x}(t)$ with dynamic vector $\mathbf{g}(\mathbf{x}(t), \psi(t))$ for some $t \in [0, T]$. In this example, the contour set $\mathbb{C}(t)$ is composed of two points $\mathbf{c}_1(t)$ and $\mathbf{c}_2(t)$, these are the only two points where vector \mathbf{g}_t is tangent to the boundary $\partial \mathbb{V}(t)$. All other points on the boundary are either entering or leaving the visible area. We choose \mathbf{h}_t as illustrated so that points on the contour set are represented in $(1, \pi/2)$ and $(1, 3\pi/2)$ on D^2 . The dashed (resp. bold) lines represent S^- (resp. S^+) on the disk and $\mathbb{X}(t)$ (resp. $\mathbb{E}(t)$) on the visible area $\mathbb{V}(t)$.

Assumption 2 We suppose that $\mathbb{C}(t)$ is composed of exactly two distinct points $\mathbf{c}_1(t)$ and $\mathbf{c}_2(t)$ for all $t \in [0, T]$, as it is the case in the example before.

This implies that $\mathbb{E}(t)$ and $\mathbb{X}(t)$ are open intervals of S^1 , for all $t \in [0, T]$. In addition, we suppose that the diffeomorphism $\mathbf{h}_t : D^2 \rightarrow \mathbb{V}(t)$, varying continuously in t , is chosen such that, once again as in the example before:

- $\mathbf{h}_t(1, \pi/2) = \mathbf{c}_1(t)$,
- $\mathbf{h}_t(1, 3\pi/2) = \mathbf{c}_2(t)$,
- $\mathbf{h}_t(S^+) = \mathbb{E}(t)$,
- $\mathbf{h}_t(S^-) = \mathbb{X}(t)$

This is always possible up to reparameterization.

Figure 5.18 illustrates a possible mapping \mathbf{h}_t .

Assumption 3 We assume we can find a curve α on D^2 from $(1, 3\pi/2)$ to $(1, \pi/2)$, such that $\alpha : [-1, 1] \rightarrow D^2$ and

- $\alpha(-1) = (1, 3\pi/2)$,
- $\alpha(1) = (1, \pi/2)$.

And we suppose that we have two suitable families of curves from S^- to α :

$$pre : [-1, 0] \times [-1, 1] \rightarrow D^2$$

and from α to S^+ :

$$post : [T, T + 1] \times [-1, 1] \rightarrow D^2$$

such that:

- $u \rightarrow pre(-1, u)$ sweeps S^- , and $u \rightarrow pre(0, u)$ is equal to $u \rightarrow \alpha(u)$;
- $pre(s, 1) = (1, \pi/2)$, $pre(s, -1) = (1, 3\pi/2)$ for any $s \in [-1, 0]$;
- $u \rightarrow post(T + 1, u)$ sweeps S^+ , and $u \rightarrow post(T, u)$ is equal to $u \rightarrow \alpha(u)$;
- $post(s, 1) = (1, \pi/2)$, $post(s, -1) = (1, 3\pi/2)$ for any $s \in [-1, 0]$.

We also note

- $\mathbf{b}_{pre}(s, u) = \mathbf{h}_0(pre(s, u))$;
- $\mathbf{b}_{post}(s, u) = \mathbf{h}_T(post(s, u))$;

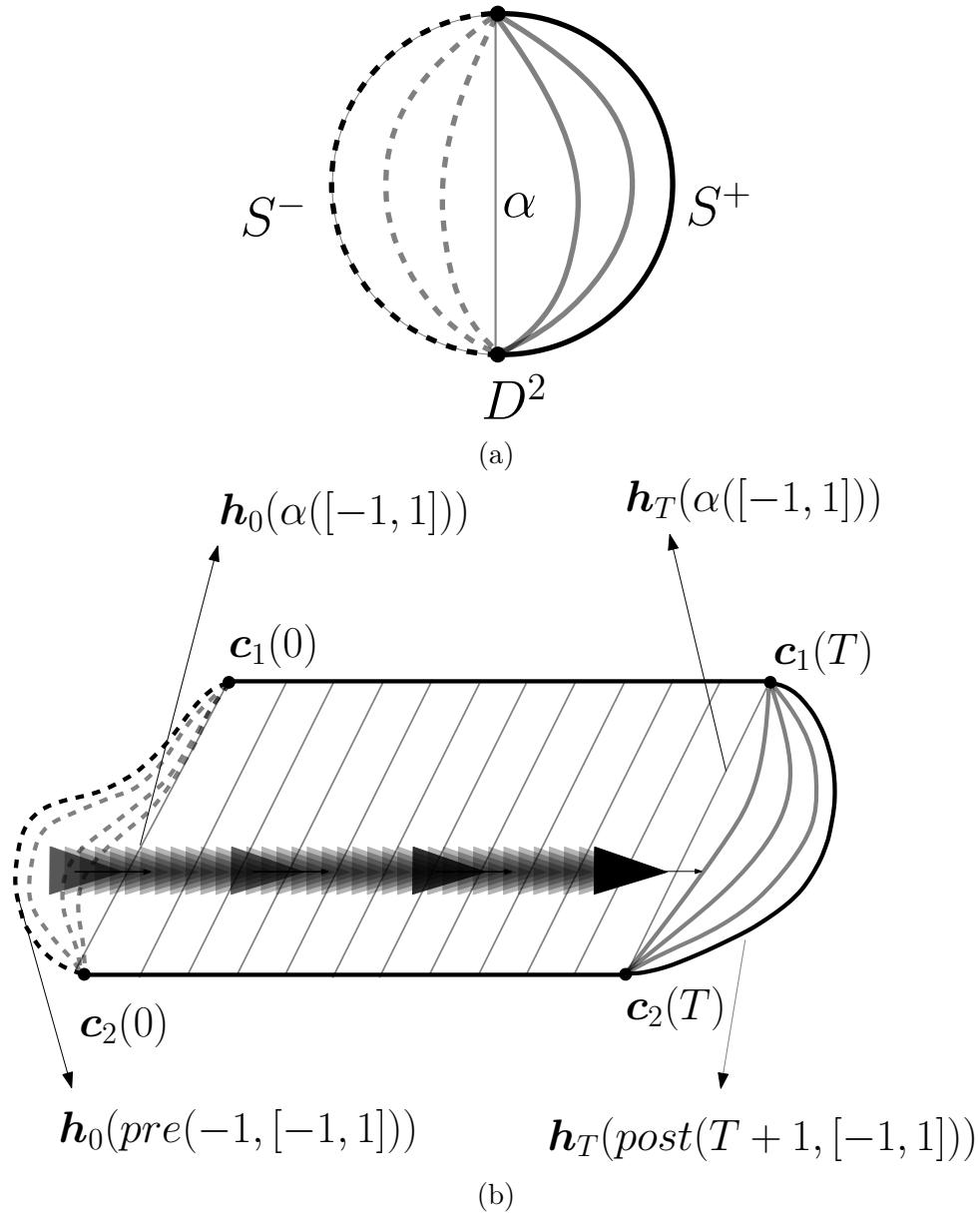


Figure 5.19: In (b) we illustrate a mission where the robot moves straight forward in the environment with a visible area \mathbb{V} as the one illustrated in Figure 5.11. Dashed lines represent in (a) curves of family *pre* and in (b) their image by function \mathbf{h}_0 . Bold solid lines represent in (a) curves of family *post* and in (b) their image by function \mathbf{h}_T . The fine solid line in (a) represents curve α and in (b) fine solid lines represent the image of α by \mathbf{h}_t with $t \in [0, T]$.

- $\mathbf{b}(s, u) = \mathbf{h}_s(\alpha(u))$.

We suppose that determinants $\det(J_{\mathbf{b}_{pre}})$, $\det(J_{\mathbf{b}_{post}})$ and $\det(J_{\mathbf{b}})$ are strictly positive.

Figure 5.19 represents these families of curve considering the simple example of a robot moving straight forward during a mission. One can notice that curves from *pre*, along with curves from *post* and α cover space D^2 and functions \mathbf{b}_{pre} , \mathbf{b}_{post} and \mathbf{b} do the same for the explored area of the mission.

We can then define $B : [-1, T + 1] \times [-1, 1] \rightarrow \mathbb{R}^2$ by:

$$B(s, u) = \begin{cases} \mathbf{b}_{pre}(s, u) & \text{if } s \in [-1, 0) \\ \mathbf{b}(s, u) & \text{if } s \in [0, T] \\ \mathbf{b}_{post}(s, u) & \text{if } s \in (T, T + 1] \end{cases}$$

Assumptions 1,2 and 3 allow us to demonstrate on next section that the seemingly more complex problem of determining the coverage measure for two-dimensional visible areas can be reduced to the one-dimensional case.

5.2.2 Coverage Measure and Topological Degree

Lemma 5.2.1. *Under the assumptions 1, 2 and 3, we have*

$$B([-1, T + 1], [-1, 1]) = \mathbb{A}_{\mathbb{E}} \quad (5.28)$$

More precisely, writing for $\mathbf{p} \in \mathbb{R}^2$,

$$N(\mathbf{p}) = \# \{ \tau \in [-1, T + 1] \mid \exists u \in [-1, 1], B(\tau, u) = \mathbf{p} \} \quad (5.29)$$

and

$$N(\mathbf{p}) = c_m(\mathbf{p}) \quad (5.30)$$

if $\det J_B(\tau, u) > 0$ for all $\tau \in [-1, T + 1]$ and $u \in [-1, 1]$.

Proof. We recall that, by Definition 30, we have $N = c_m(\mathbf{p})$ closed intervals $[a_i, b_i]$, such that $T_{\mathbf{p}} = \bigcup_{i=1}^N [a_i, b_i]$. We have three cases for each interval $[a_i, b_i]$:

- $[a_i, b_i]$ is a proper sub-interval of $[0, T]$: time a_i is the moment when point \mathbf{p} enters the visibility region of the robot, necessarily at a point in $\mathbb{E}(a_i) \cup \mathbb{C}(a_i)$, for the i th time, and time b_i is the moment when point \mathbf{p} leaves the visibility region, necessarily at a point in $\mathbb{X}(b_i) \cup \mathbb{C}(b_i)$. Since $\det(J_{\mathbf{p}})$ is strictly positive, the image of curve α , by \mathbf{h}_t , with $t \in [a_i, b_i]$, is sweeping the set of points on the plane in the direction of the differential flow, and will only intersect point \mathbf{p} once during this time interval. As $a_i > 0$ and $b_i < T$, this corresponds to a unique (s, u) within $[a_i, b_i] \times [-1, 1]$, such that $B(s, u) = \mathbf{p}$.

- Time a_i is equal to 0 and $b_i < T$: this means that \mathbf{p} was already in the visibility region of the robot at the beginning of the mission, and leaves it at some point of $\mathbb{X}(b_i) \cup \mathbb{C}(b_i)$ at time b_i . As $\text{pre}(-1, [-1, 1])$ is the curve S^- and $\mathbf{h}_t(S^-) = \mathbb{X}(t)$, curves from family pre sweep the area between $\mathbb{X}(0)$ to curve $\mathbf{b}(0, [-1, 1])$. Furthermore, since $\det(J_{\mathbf{b}_{\text{pre}}})$ is always strictly positive, this sweep is made in the direction of the flow of the underlying dynamical system. In this case, it can be concluded that there is a unique (s, u) within $[-1, b_i] \times [-1, 1]$, such that $B(s, u) = \mathbf{p}$.
- Time b_i is equal to T and $a_i > 0$: this means that \mathbf{p} enters the visibility region of the robot, necessarily at a point in $\mathbb{E}(a_i) \cup \mathbb{C}(a_i)$, for the i th time, and stays within the visibility region up to the end of the mission at instant T . This is similar to the case above and we have $\text{post}(T+1, [-1, 1])$ that is the curve S^+ and $\mathbf{h}_t(S^+) = \mathbb{E}(t)$. Curves from family post sweep the area from curve $\mathbf{b}(T, [-1, 1])$ to curve $\mathbb{E}(T)$. Since $\det(J_{\mathbf{b}_{\text{post}}})$ is always strictly positive, this sweep is made in the direction of the flow of the underlying dynamical system. Therefore, there is a unique (s, u) within $[a_i, T+1] \times [-1, 1]$, such that $B(s, u) = \mathbf{p}$.

This shows that there is a bijection between the set of connected components of $T_{\mathbf{p}}$ and the number of solutions (s, u) to equation $B(s, u) = \mathbf{p}$. \square

Lemma 5.2.1 makes it possible to reduce the computation of the number of times a robot with a two-dimensional visibility region sees a point $\mathbf{p} \in \mathbb{R}^2$ to the same computation considering a one-dimensional visibility region (given by the curve $u \rightarrow B(\tau, u)$ for a parametrization $\tau \in [-1, T+1]$). This allows us to use the topological degree of map B , to compute $c_m(\mathbf{p})$ using the winding number of some associated cycle:

Proposition 2. *Let γ be the curve which is the concatenation of $u \rightarrow B(-1, -u)$ (seen as a path with u evolving from -1 to 1) with $t \rightarrow B(t, -1)$ (a path with t evolving from $t = 0$ to $t = T$), with $u \rightarrow B(T+1, u)$ (with u varying from -1 to 1) and with $t \rightarrow B(T-t, 1)$ (with t varying from 0 to T). Then γ is the sensor's contour associated to the mission and we have*

$$c_m(\mathbf{p}) = \bar{\eta}(\gamma, \mathbf{p})$$

Proof. We define space $W = [-1, T+1] \times [-1, 1]$ as a space whose points have all been explored, represented in a relative coordinate system. Space W is equivalent to the waterfall space that we had for the one-dimensional scenario and function B is equivalent to the sweep function, that maps points from the waterfall space to a global coordinate frame. Furthermore, cycle γ , as defined in the proposition, is the image of the boundary of space W by B , as expected. \square

Example 2. We assume again the context presented on Example 1, where we have

$$\mathbf{g}(\mathbf{x}(t), \psi(t)) = \begin{pmatrix} \cos(\psi(t)) \\ \sin(\psi(t)) \end{pmatrix}$$

and $\dot{\psi} = a \in \mathbb{R}$ is constant. The visibility region is a disk of radius 1 respecting the diffeomorphism in (5.23).

Consider now the curve:

$$\alpha(u) = \begin{pmatrix} 0 \\ u \end{pmatrix}$$

in cartesian coordinates with $u \in [-1, 1]$. Using an orientation preserving transform of α by \mathbf{h}_t , with $t \in [0, T]$, we obtain a set of curves on the global coordinate frame. These curves do not span the entire explored area, still, we can use them to characterize the coverage measure as shown above.

First, we prove that for such a choice, the induced \mathbf{b} function has a Jacobian with positive determinant under some mild conditions. We compute:

$$\mathbf{b}(t, u) = \begin{pmatrix} -u \sin(\psi) + x_1(t) \\ u \cos(\psi) + x_2(t) \end{pmatrix}$$

And:

$$\det J_b = \begin{vmatrix} (1 - au) \cos(\psi) & -\sin(\psi) \\ (1 - au) \sin(\psi) & \cos(\psi) \end{vmatrix}$$

Therefore $\det(J_b) = (1 - au)$ which is strictly positive for all $u \in [-1, 1]$ if and only if $-1 < a < 1$. Assuming an angular velocity small enough to respect this condition is a reasonable hypothesis.

Now we define $\text{pre} : [-1, 0] \times [-1, 1] \rightarrow D^2$ by:

$$\text{pre}(r, u) = \beta(1/r, u/|\arcsin(1/r)|)$$

where

$$\beta(s, \theta) = \begin{pmatrix} r \cos(\theta) + \sqrt{s^2 - 1} \\ r \sin(\theta) \end{pmatrix}$$

for $r \in (-\infty, -1]$, and $\theta \in [-\arcsin(\frac{1}{s}), \arcsin(\frac{1}{s})]$. We extend it at 0 by continuity.

Checking $\det(J_{\mathbf{b}_{\text{pre}}}) > 0$ is equivalent to checking $\det(J_c)$, where

$$\mathbf{c}(\theta) = \mathbf{h}_0(\beta(r, \theta))$$

up to the change of coordinate we made (which is orientation preserving). We compute:

$$\begin{aligned}\mathbf{b}_{pre}(s, \theta) &= \left(\begin{array}{c} s \cos(\theta + \psi(0)) + \sqrt{s^2 - 1} \cos(\psi(0)) \\ s \sin(\theta + \psi(0)) + \sqrt{s^2 - 1} \sin(\psi(0)) \end{array} \right) \\ &\quad + \left(\begin{array}{c} x_1(0) \\ x_2(0) \end{array} \right)\end{aligned}$$

We have now:

$$\frac{\partial \mathbf{b}}{\partial t} = \left(\begin{array}{c} s' \cos(\theta + \psi) + \frac{ss'}{\sqrt{s^2 - 1}} \cos(\psi) \\ s' \sin(\theta + \psi) + \frac{ss'}{\sqrt{s^2 - 1}} \sin(\psi) \end{array} \right)$$

and,

$$\frac{\partial \mathbf{b}}{\partial \theta} = \left(\begin{array}{c} -s \sin(\theta + \psi) \\ s \cos(\theta + \psi) \end{array} \right)$$

Finally:

$$\det(J_{\mathbf{b}_{pre}}) = ss' \left(1 + \frac{s}{\sqrt{s^2 - 1}} \cos(\theta) \right)$$

But:

$$-\frac{\sqrt{s^2 - 1}}{s} \leq \cos(\theta) \leq 1$$

and $s < 0$, hence:

$$1 + \frac{s}{\sqrt{s^2 - 1}} \leq 1 + \frac{s}{\sqrt{s^2 - 1}} \cos(\theta) \leq 0$$

Supposing $s' > 0$ we get $\det(J_{\mathbf{b}_{pre}}) \geq 0$ (and only 0 for $\theta = \pi/2$ and $\theta = 3\pi/2$).

Defining post similarly from curve:

$$\beta'(s, \theta) = \left(\begin{array}{c} s \cos(\theta) - \sqrt{s^2 - 1} \\ s \sin(\theta) \end{array} \right)$$

for $s \in [1, \infty)$ depending on $t \in [0, T]$ and $\theta \in [-\arcsin(\frac{1}{s}), \arcsin(\frac{1}{s})]$ enjoys the properties we need.

5.2.3 Sweeping backwards and Uncertainties

By taking the sensor's contour γ as defined in Proposition 2, we can immediately apply relations established in extensions (5.12) and (5.16) to the two-dimensional scenario.

5.3 Conclusion

In conclusion, this chapter formally defined the key concepts of coverage measure and explored area within two distinct scenarios, one involving a one-dimensional sensor and the other extending the method to encompass two-dimensional sensors. The chapter presents the main contribution of this work that establishes a relation between the topological degree and the coverage measure.

Furthermore, we expanded our problem definition to include the crucial aspect of uncertainty, acknowledging the real-world challenges that robots face during exploration tasks. This theoretical discussion has provided a foundation for addressing the primary objective of this thesis that is tackled on the next chapter.

Chapter 6

Area Characterization

Contents

6.1	Problem Definition	149
6.2	Creating the Mission Contour	151
6.3	Computing the Extended Winding Number	157
6.3.1	Method 1: Turning Around	160
6.3.2	Method 2: Alexander Theorem	169
6.3.3	Method 3: Jordan Curves Sum	189
6.4	Waterfall Characterization	193
6.5	Conclusion	199

The previous chapter introduced a method to estimate the coverage measure of a point within the environment through the computation of certain winding numbers. It becomes evident that if we can define the coverage measure for all the points within an area of interest, we can determine which part of this area has been explored during a mission. Therefore, in this chapter, we address the challenge of developing an algorithm to estimate the coverage measure for all points within a subset of the plane. This is an area characterization problem, which is centered around defining the environment within a particular context, and the context considered here revolves around classifying points based on their coverage measure. To achieve this objective, we use the concepts introduced in Chapter 3 and approach the problem from the perspective of interval analysis.

6.1 Problem Definition

We are going to assume that the actual trajectory \mathbf{x}^* of the robot during the mission is known to be inside a set of possible trajectories $\mathbb{S}_{\mathbf{x}}$, as presented on last chapter in Section 5.1.3.

In practice, this collection of possible solutions is represented by a tube $[\mathbf{x}] \in \mathbb{I}\mathcal{F}^2$ that encompasses all possible elements of $\mathbb{S}_{\mathbf{x}}$ in such a way that any $\mathbf{x} \in \mathbb{S}_{\mathbf{x}}$ also belongs to $[\mathbf{x}]$, see Figure 6.1. In this context, the orientation of the robot and derivatives are also represented by tubes. Employing tubes to represent uncertainty in the dynamics of a mobile robot is a customary approach in robotics [121] [122] [123], offering a guaranteed framework for addressing the problem.

The interval $[0, T] \in \mathbb{IR}$ represents the time interval during which the mission takes place, and $S \in \mathbb{IR}^2$ is a box representing a subset of the environment to be characterized, or the area of interest of the mission. We assume that the area explored during the mission is completely inside S , i.e. $\mathbb{A}_{\mathbb{E}} \subset S$. Then, we assume that the following data concerning the dynamics of the robot is available:

- (i) $[\mathbf{x}] \in \mathbb{I}\mathcal{F}^2$ represents all possible trajectories of the robot for the mission. This tube is obtained during the process of localization, not within the focus of this study, that typically involves the integration of proprioceptive information. This integration introduces uncertainty into the estimation. To mitigate this uncertainty, data from exteroceptive sensors are often fused with proprioceptive data [124] [125]. The tube $[\mathbf{x}]$

Images in this chapter use the AUV icon  for illustrative purposes. However, the algorithms presented here can be applied to different types of mobile robots.

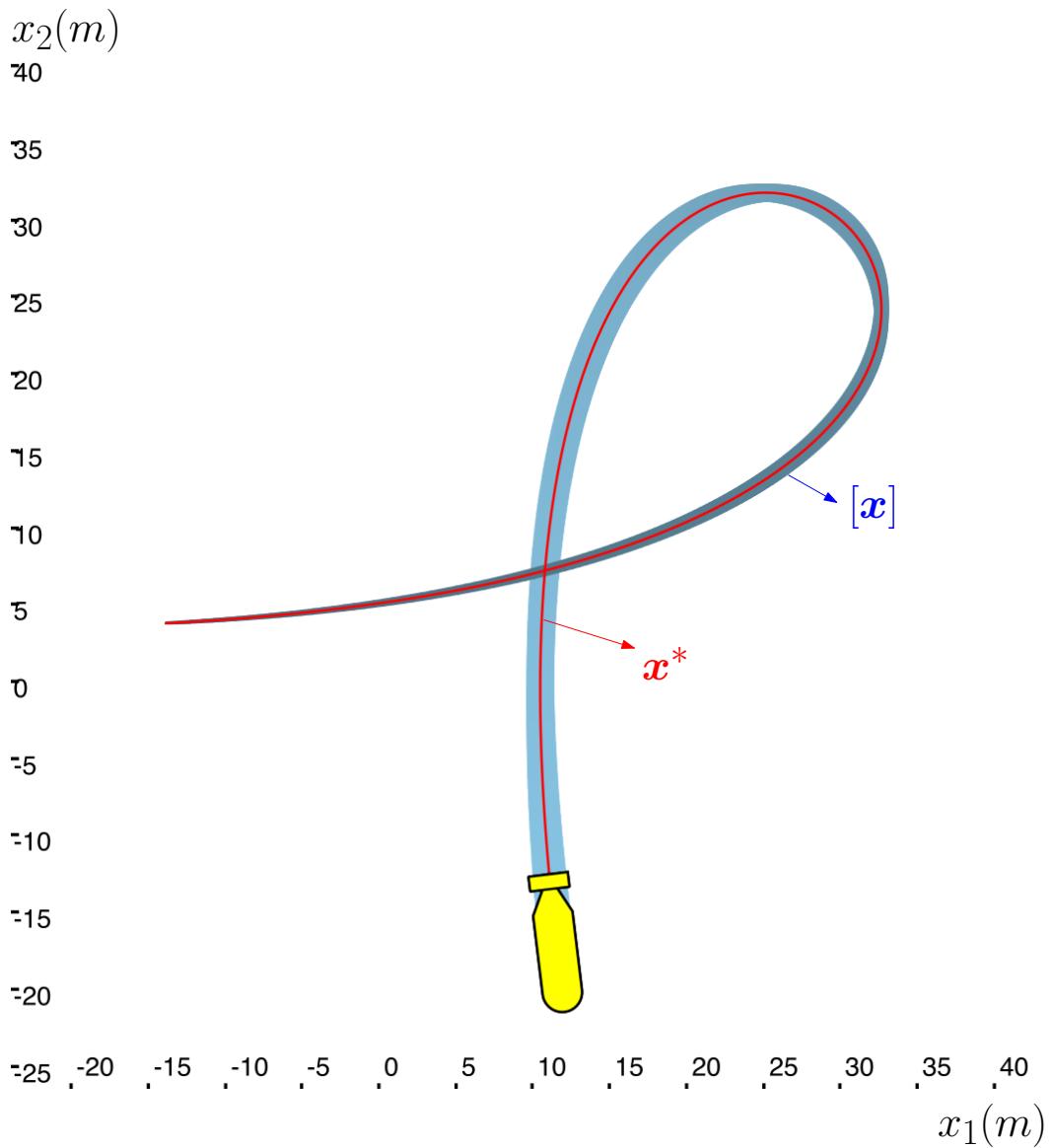


Figure 6.1: The robot is represented at the end of the mission at instant T . Function \mathbf{x}^* , that represents the ground-truth trajectory of the robot during the mission, is unknown. Instead, we will be dealing with a tube $[\mathbf{x}]$ known to enclose \mathbf{x}^* . The tube $[\mathbf{x}]$ starts out narrow at the beginning of the mission, and uncertainty increases over time. This is a typical feature of tubes enveloping the trajectory of a mobile robot. The growth in uncertainty is a consequence of the propagation of uncertainty through the data integration process as time progresses. This Figure was generated using the Codac library [99].

- encapsulates the final estimation and envelops the resulting uncertainty derived from the localization process;
- (ii) $[\dot{\mathbf{x}}] \in \mathbb{I}\mathcal{F}^2$ is a tube representing the derivative of $[\mathbf{x}]$, it can be obtained through integration of data or through direct sensor measurements;
 - (iii) $[\ddot{\mathbf{x}}] \in \mathbb{I}\mathcal{F}^2$ is a tube representing the second derivative of $[\mathbf{x}]$. It is usually obtained by direct sensor measurements, such as accelerometers, and incorporates the inherent uncertainties associated with these sensors;
 - (iv) $[\psi] \in \mathbb{I}\mathcal{F}$ envelopes all functions possibly representing the orientation of the robot during time. This tube is also obtained as a result of the localization process through integration of data from embedded sensors;
 - (v) $[\dot{\psi}] \in \mathbb{I}\mathcal{F}$ represents the first derivative of $[\psi]$. The tube $[\dot{\psi}]$ is typically obtained through direct sensor measurements, from gyroscopes, for example. These measurements inherently include associated sensor uncertainties that are enveloped by tube $[\dot{\psi}]$;

From this information, and considering the characteristics of the visible area of the sensor employed for exploration, a tube $[\gamma]$ representing all possible mission contours can be computed, along with $[\gamma^+]$ and $[\gamma^-]$. Tube $[\gamma^+]$ represents the set of potential mission contours surrounding points swept by the robot in the same direction as its advancement movement. Tube $[\gamma^-]$ denotes the set of possible mission contours surrounding those swept in the opposite direction. Notice we employ the notation established in Section 5.1.2. In Figure 6.2 we illustrate the mission contour for the mission presented in Figure 6.1. In Section 6.2, we discuss in details the process of creating these tubes.

The final output of the area characterization algorithm is a subpaving of box S according to the coverage measure of its points. With this purpose, we compute the extended winding number of tubes $[\gamma^+]$ and $[\gamma^-]$, and then the coverage measure can be obtained as proposed on Equation (5.17). Therefore, winding number computation is the principal building block of the area characterization algorithm and Section 6.3 elaborates on the development of methods to fulfill this purpose.

6.2 Creating the Mission Contour

In this section, we portray the process of generating tubes $[\gamma^+]$ and $[\gamma^-]$, along with their corresponding derivatives $[\dot{\gamma}^+]$ and $[\dot{\gamma}^-]$, assuming an exploration sensor with one-dimensional visible area. We consider the configuration described in Section 2.2.3 where two colinear side-scan sonars are deployed, each

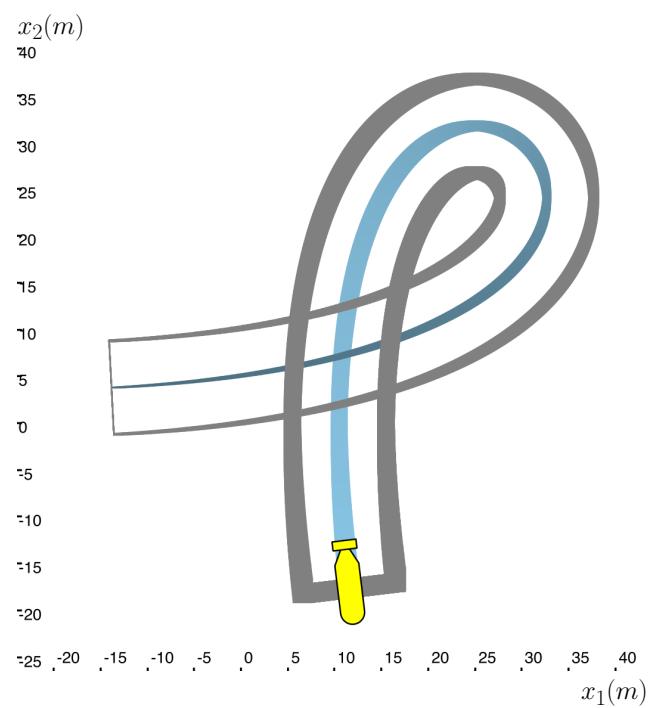


Figure 6.2: In this example, where the trajectory of the robot is represented by the blue tube, the mission contour $[\gamma]$ is represented by the gray tube. All the points are swept forward during this mission, therefore $[\gamma] = [\gamma^+]$ and there is no tube $[\gamma^-]$ associated to the mission.

scanning one side of the robot. For the sake of simplicity in presentation, we assume that the maximal visibility range is represented by a fixed constant value $L \in \mathbb{R}$. However, in practical applications, this constant can be substituted with a time-dependent function if the visibility range changes over time. Additionally, it can even be replaced with a tube representation if not only does the range vary over time but is also subject to uncertainty.

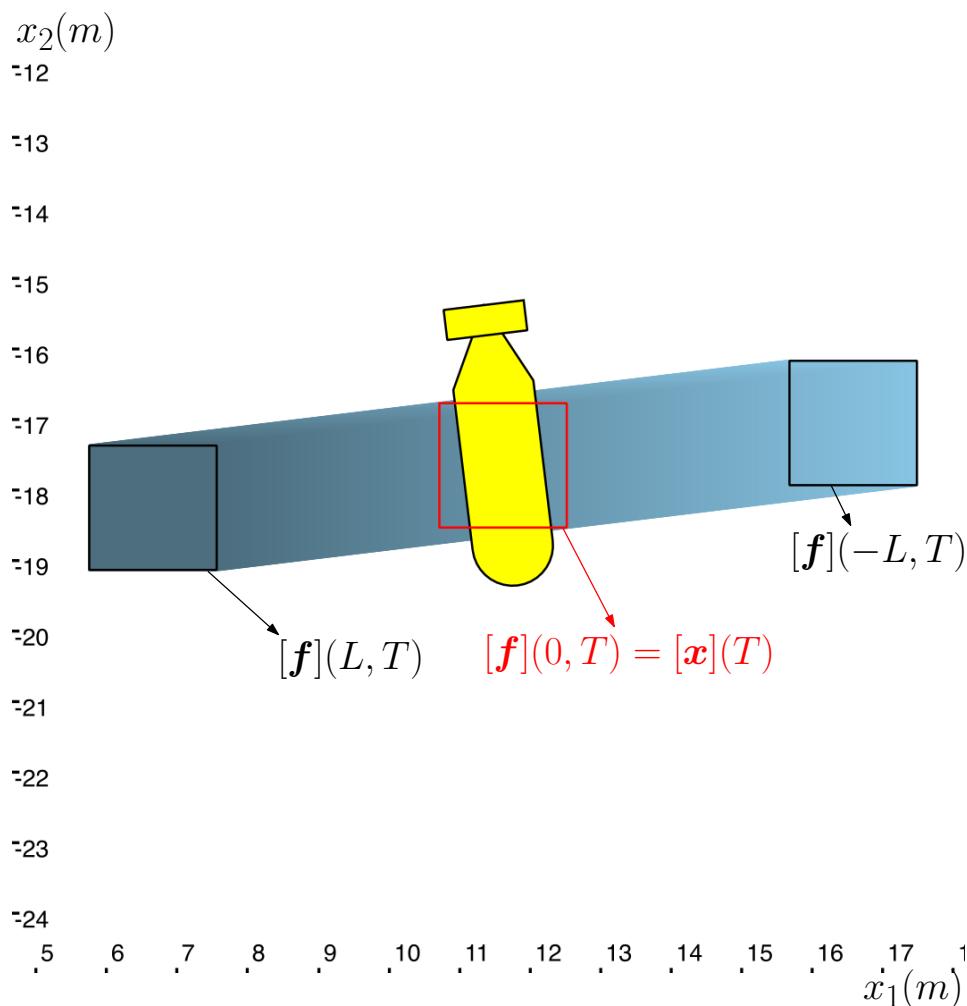


Figure 6.3: Tube $[f]([-L, L], T)$ illustrating the visible area of the robot at instant T assuming the mission illustrated on Figure 6.1. The red box represents $[x](T)$ within the tube and black boxes represent $[f](-L, T)$ and $[f](L, T)$. This Figure was generated using the Codac library [99].

In this context, let us assume the following sweep function:

$$\mathbf{f}(l, t) = \mathbf{x} + l \begin{pmatrix} \sin(\psi(t)) \\ -\cos(\psi(t)) \end{pmatrix} \quad (6.1)$$

Where $l \in [-L, L]$. Figure 6.3 illustrates the resultant visible area at final instant T if this sweep function is applied on the context of the mission presented on Figure 6.1.

We can compute the Jacobian matrix of this sweep function to obtain

$$J_{\mathbf{f}}(l, t) = \begin{bmatrix} \frac{\partial f_1}{\partial l}(l, t) & \frac{\partial f_1}{\partial t}(l, t) \\ \frac{\partial f_2}{\partial l}(l, t) & \frac{\partial f_2}{\partial t}(l, t) \end{bmatrix} = \begin{bmatrix} \sin(\psi(t)) & \dot{x}_1(t) + l \cos(\psi(t)) \dot{\psi}(t) \\ -\cos(\psi(t)) & \dot{x}_2(t) + l \sin(\psi(t)) \dot{\psi}(t) \end{bmatrix} \quad (6.2)$$

The cycle γ^+ , on the literature developed on this document, surrounds points $(l, t) \in W$ where the determinant of the Jacobian matrix $J_{\mathbf{f}}(l, t)$ is greater than zero. As discussed on last chapter, the mapping defined by \mathbf{f} maintains orientation for these specific points. This signifies that the order of points in the input space remains unchanged in the target space, without any flipping or reversing. Hence, we refer to these points as being swept in the direction of the advancement of the robot.

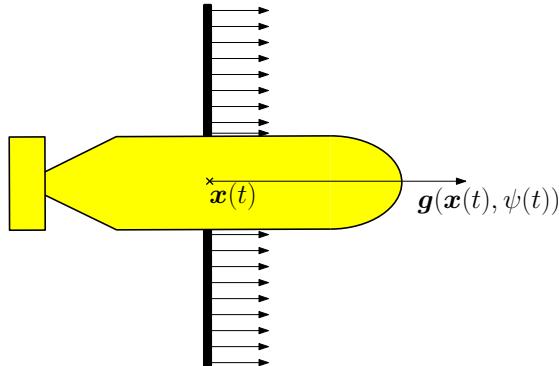


Figure 6.4: Example of robot positioned at $\mathbf{x}(t)$ and with derivatives represented by vector $\mathbf{g}(\mathbf{x}(t), \psi(t))$ for some instant t during the mission. At this moment $\dot{\psi}(t) = 0$ and the speed vector of any point within the visible area (represented by the bold black line) is parallel to the one of the robot.

To illustrate this concept, we consider a scenario in which a robot moves forward with a constant angular speed $\dot{\psi} = 0$ throughout its mission, as depicted in Figure 6.4. At any given moment, the velocity vector at each point within the visible area aligns with the velocity vector of the robot, resulting in all points being swept in the same direction as the advancement movement of the robot.

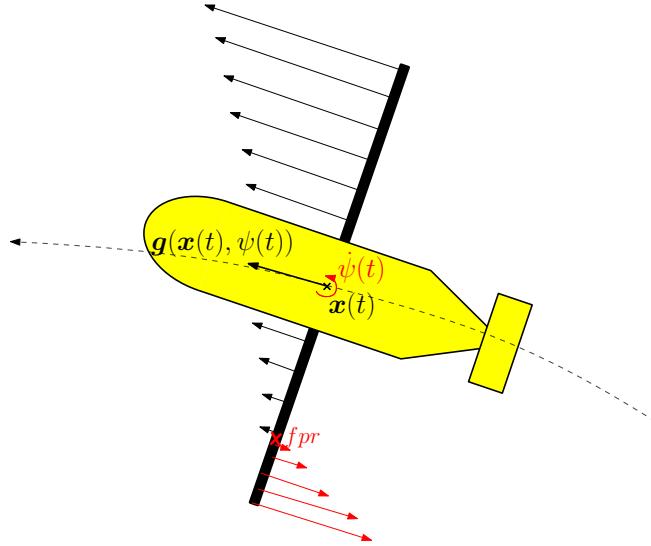


Figure 6.5: Robot with a non-null angular velocity. The fixed point of rotation fpr represented by the red cross divides the visible area between points that turn on the same direction of the robot and points that turn on the opposite direction. This is illustrated by the speed vectors of points on the visible area.

However, when the robot has a non-zero angular speed, the fixed point of rotation becomes crucial in determining which points on the line perpendicular to the speed vector of the robot turn counterclockwise and which ones turn clockwise, as illustrated in Figure 6.5. If the robot itself is turning counterclockwise, points that also turn counterclockwise exhibit a positive determinant of the Jacobian on their representatives in the original domain. Same thing for points turning clockwise if the robot is also turning clockwise. Conversely, points turning on the opposite direction of the robot exhibit a negative determinant of the Jacobian on their representatives in the original domain.

If the lateral distance of the robot to the fixed point of rotation falls within the range of its visible area, namely $[-L, L]$, this implies that there are visible points being swept in a backward direction. Otherwise, all points within the visible area are swept forward, even when the robot has some angular speed.

The lateral distance l_{fpr} of the fixed point of rotation to the robot at instant $t \in [0, T]$ can be obtained by analyzing $\det(J_f(l_{fpr}(t), t)) = 0$. We have

$$l_{fpr}(t) = -\frac{(\dot{x}_1(t) \cos(\psi(t)) + \dot{x}_2(t) \sin(\psi(t)))}{\dot{\psi}(t)} \quad (6.3)$$

It's worth observing that parameter l_{fpr} exhibits an inverse relationship with the angular speed rate. When $\dot{\psi}(t)$ approaches zero, the fixed point of rotation

effectively becomes located at infinity and lies beyond the bounds of the visible area. Consequently, all the points falling within a lateral distance range of $[-L, L]$ are swept in the same direction as the motion of the robot. Generally, as the rate of rotation increases, the number of points swept in the backward direction increases. As a consequence, the chances of having the fixed point of rotation creating a division of rotation direction within the visible area also increases.

For this reason, in order to define $[\gamma^+]$, the first step is to compute a tube $[l_{fpr}]$ with the lateral distance of the fixed point of rotation to the robot during the mission:

$$[l_{fpr}](t) = -\frac{([\dot{x}_1](t) \cos([\psi](t)) + [\dot{x}_2](t) \sin([\psi](t)))}{[\dot{\psi}](t)} \quad (6.4)$$

This allows to define tubes $[L_r]$ and $[L_l]$:

$$[L_r](t) = \begin{cases} [l_{fpr}](t) \cap [0, L] & \text{if } [l_{fpr}](t) \cap [0, L] \neq \emptyset \\ [L, L] & \text{otherwise} \end{cases} \quad (6.5)$$

$$[L_l](t) = \begin{cases} [l_{fpr}](t) \cap [-L, 0] & \text{if } [l_{fpr}](t) \cap [-L, 0] \neq \emptyset \\ [-L, -L] & \text{otherwise} \end{cases} \quad (6.6)$$

These two tubes represent the maximal range of visibility, respectively, on the right and left side of the robot, if we take into consideration only points that are swept forward. Finally, the set of cycles $[\gamma^+]$ can be defined as a concatenation of tubes

$$[\gamma^+] = [\mathbf{f}]([L_r](t), t) * \underset{t=0 \rightarrow T}{\underset{l=[L_r](T) \rightarrow [L_l](T)}} [\mathbf{f}](l, T) * [\mathbf{f}]([L_l](t), t) * \underset{t=T \rightarrow 0}{\underset{l=[L_l](0) \rightarrow [L_r](0)}} [\mathbf{f}](l, 0) \quad (6.7)$$

as its derivative

$$[\dot{\gamma}^+] = [\dot{\mathbf{f}}]([L_r](t), t) * \underset{t=0 \rightarrow T}{\underset{l=[L_r](T) \rightarrow [L_l](T)}} [\dot{\mathbf{f}}](l, T) * [\dot{\mathbf{f}}]([L_l](t), t) * \underset{t=T \rightarrow 0}{\underset{l=[L_l](0) \rightarrow [L_r](0)}} [\dot{\mathbf{f}}](l, 0) \quad (6.8)$$

where the symbol $*$ indicates concatenation.

The set \mathbb{S}^- of points $(l, t) \in W$ for which $J_f(l, t) < 0$ can be an unconnected set in W , and in this case, $\gamma^- = \mathbf{f}(\partial\mathbb{S}^-)$ can be a set of disconnected cycles in the plane. For this reason, we define $\{[\gamma^-]\}$, that is a set of tubes, each one enveloping a connected component of $\mathbf{f}(\partial\mathbb{S}^-)$. Furthermore, each one of these cycles can be associated to a connected time interval $[t] = [t^-, t^+] \subset [0, T]$ for which $[l_{fpr}](t) \subset [-L, L]$ such that for any $[\gamma^-] \in \{[\gamma^-]\}$ we have

$$[\gamma^-] = [\mathbf{f}](-L, t) * [\mathbf{f}]([L_l](t), t) \quad (6.9)$$

and

$$[\dot{\gamma}^-] = [\dot{f}]_{t=t^+ \rightarrow t^-}(-L, t) * [\dot{f}]_{t=t^- \rightarrow t^+}([L_l](t), t) \quad (6.10)$$

The process of creating $[\gamma^+]$ and $\{[\gamma^-]\}$ is illustrated in Figures 6.6 and 6.7 for two different missions. Once these cycles are defined along with their derivatives, their winding numbers can be computed.

6.3 Computing the Extended Winding Number

In the introductory chapter of this document, we discussed how existing methods in the literature for topological degree computation in the plane compute the winding number of a cycle with respect to a single point. These methods require individual application to each point inside an area of interest for an entire characterization. In addition, they were designed primarily for calculating the winding number of a single cycle, and to obtain all the conceivable winding number values, a computation would need to be done to every potential cycle. This approach would not be feasible in our context given the infinite number of possible cycles we need to consider. In this scenario, here, we introduce three novel set-membership algorithms developed during this thesis, for efficiently determining the winding number of a tube with respect to all the points within an area of interest.

The first approach is based on the line-crossing principle used in point-in-polygon algorithms [126]. This method is the most suitable, among the three methods discussed here, when we want to estimate the extended winding number of a tube with respect to a single box in the environment. Nevertheless, it does come with the highest computational complexity when the goal is to calculate the extended winding number for multiple boxes, particularly within the context of area characterization.

The second method presented is based on combinatorial rules proposed by [127]. It stands out as the most computationally efficient method for area characterization. However, it does rely on certain strong assumptions regarding self-intersections within the tube for which the winding number is being calculated.

The final approach we propose involves partitioning the cycle into Jordan curves, making the winding number computation on each segment straightforward. It serves as a viable alternative when the cycle fails to respect the conditions necessary for the application of the second method.

These three approaches are now presented.

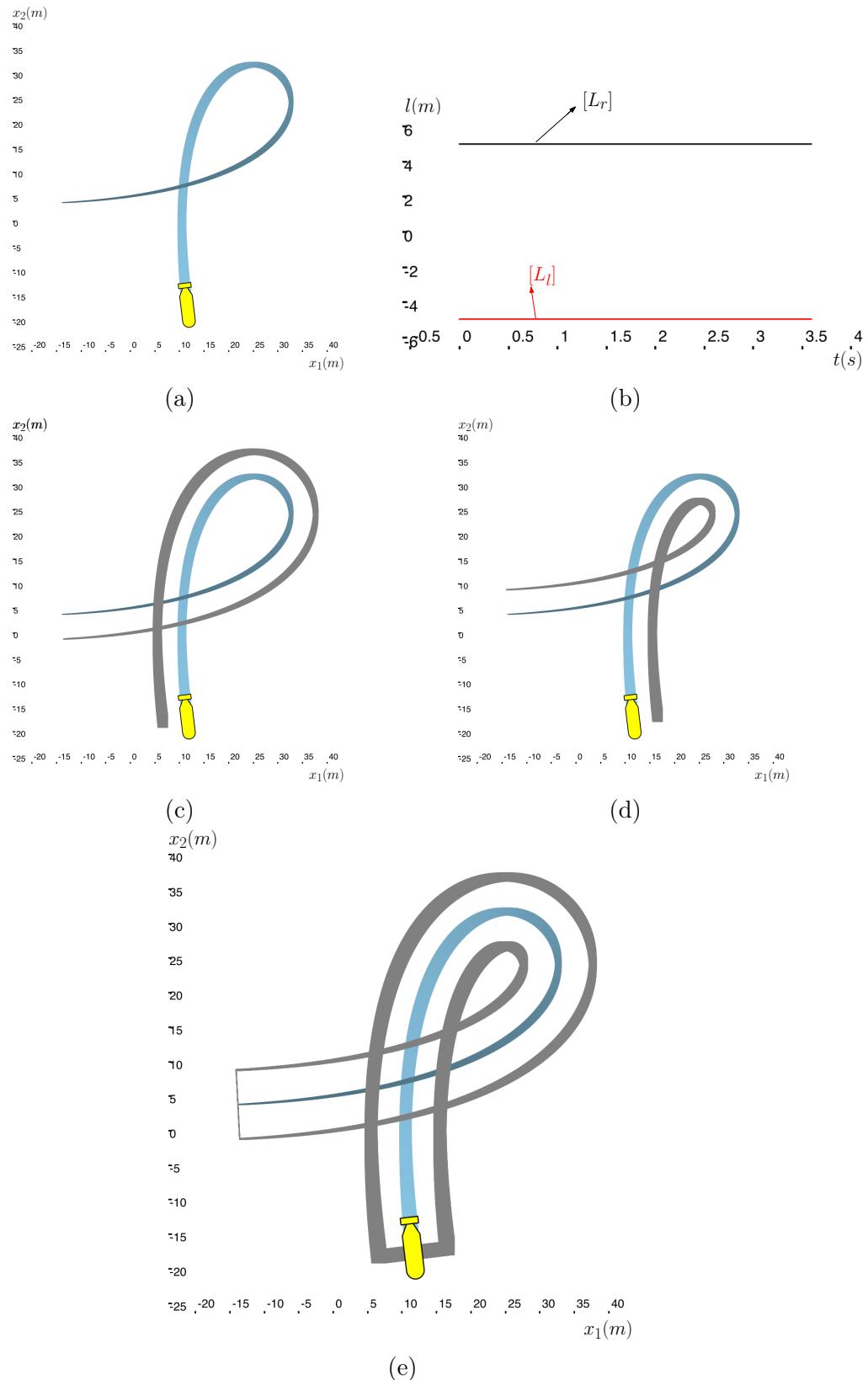


Figure 6.6: (a): $[\mathbf{x}]$ with robot illustrated at instant T ; (b): Maximal range of visibility, considering only points that are swept forward, on the right side $[L_r]$ and on the left side $[L_l]$. For this mission we have $L = 5m$, and we can notice that there are no points swept backwards; (c): In gray: $[\mathbf{f}]([L_r], [0, T])$; (d): In gray: $[\mathbf{f}]([L_l], [0, T])$; (e): In gray: $[\gamma^+]$.

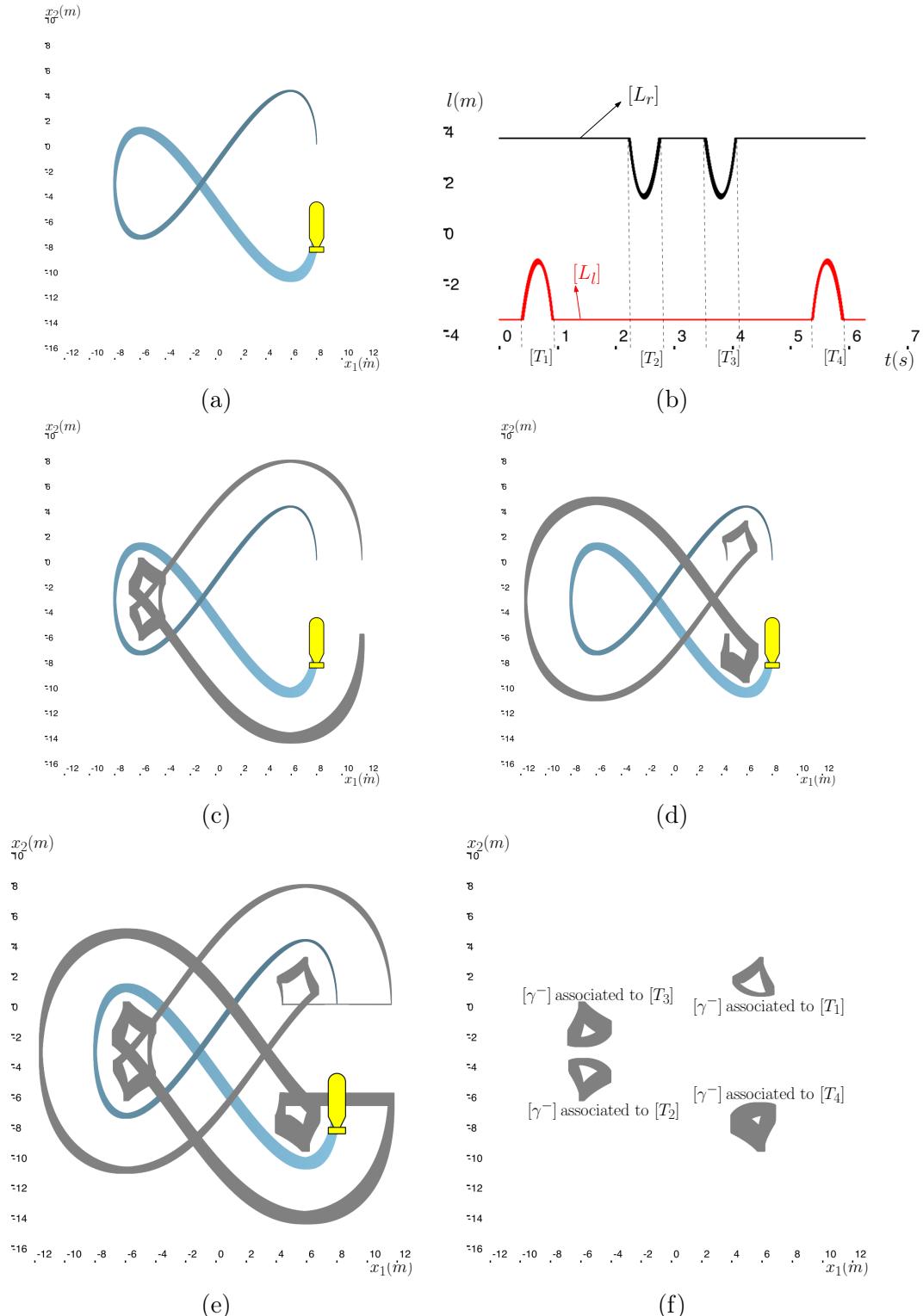


Figure 6.7: (a): $[\mathbf{x}]$ with robot illustrated at instant T ; (b): Maximal range of visibility, considering only points that are swept forward, on the right side $[L_r]$ and on the left side $[L_l]$. For this mission we have $L = 3.6m$. There are four intervals of time $[T_1]$, $[T_2]$, $[T_3]$ and $[T_4]$ during which points are swept backwards during the mission; (c): In gray: $[\mathbf{f}]([L_r], [0, T])$; (d): In gray: $[\mathbf{f}]([L_l], [0, T])$; (e): In gray: $[\gamma^+]$; (e): In gray: $\{[\gamma^-]\}$ that is a list of four tubes representing the illustrated cycles.

6.3.1 Method 1: Turning Around

This first approach is inspired by the classical problem in computational geometry, which involves determining whether a point is inside a polygon. The utilization of axis-crossing methods has long been recognized as a viable solution to this problem [128], [129], and [130] introduced the concept of winding numbers using this methodology.

In more recent times, at the beginning of the 21st century, [131] presented a new algorithm for determining if a point resides inside a polygon. To achieve this objective, the algorithm calculates the winding number of a polygon with respect to the point, employing the axis-crossing method. Indeed, a non-zero winding number suffices to establish whether the polygon contains the point.

Now, we introduce the axis-crossing technique for computing winding numbers, assuming a continuous closed curve in the plane, rather than a polygon.

Let us assume a path $\gamma : [0, 1] \rightarrow \mathbb{R}^2$, with $\gamma(0) = \gamma(1)$, and $\mathbf{p} \in \mathbb{R}^2 \setminus \gamma$. Sunday's algorithm [131] for computing the winding number of γ with respect to \mathbf{p} consists of choosing an infinite ray cast from \mathbf{p} in any direction, that we name \mathcal{D} , and we initialize the result variable with 0. Then, we identify intersections of \mathcal{D} with γ . For each intersection we analyze if γ is crossing \mathcal{D} from right to left or from left to right. Based on the direction, the result variable is updated. Intersections crossing \mathcal{D} from right to left add +1 to the result and intersections crossing from left to right add -1. After summing all intersections according to the direction of crossing, the resultant value correspond to the winding number of γ with respect to \mathbf{p} . This process is exemplified on Figure 6.8.

In Figure 6.9, we consider a more complex example of cycle γ . Here, for illustration purposes, we choose a random cycle, not necessarily representing a mission contour. One can see that visually determining how many times γ winds around point zero is easy with the axis-crossing method, independently of the complexity of the considered cycle. This is one of the advantages of the method.

Now, we formalize this approach for computing the winding number of loops with respect to point $\mathbf{0} \in \mathbb{R}^2$. We set $\mathbf{p} = \mathbf{0}$ and $\mathcal{D} = \{\mathbf{0}\} \times [0, \infty]$. We assume that γ intersects \mathcal{D} on a finite number of instants and in this context, the two following set-valued functions can be defined:

$$\mathbb{T}^-(t) = \{\tau \in [0, t] \mid \gamma(\tau) \in \mathcal{D} \text{ and } \dot{\gamma}_1(\tau) > 0\} \quad (6.11)$$

$$\mathbb{T}^+(t) = \{\tau \in [0, t] \mid \gamma(\tau) \in \mathcal{D} \text{ and } \dot{\gamma}_1(\tau) < 0\} \quad (6.12)$$

where $t \in [0, 1]$. We also define the notion of cross number $w_\gamma : [0, 1] \rightarrow \mathbb{Z}$ that determines how many times loop γ crosses \mathcal{D} , considering the direction

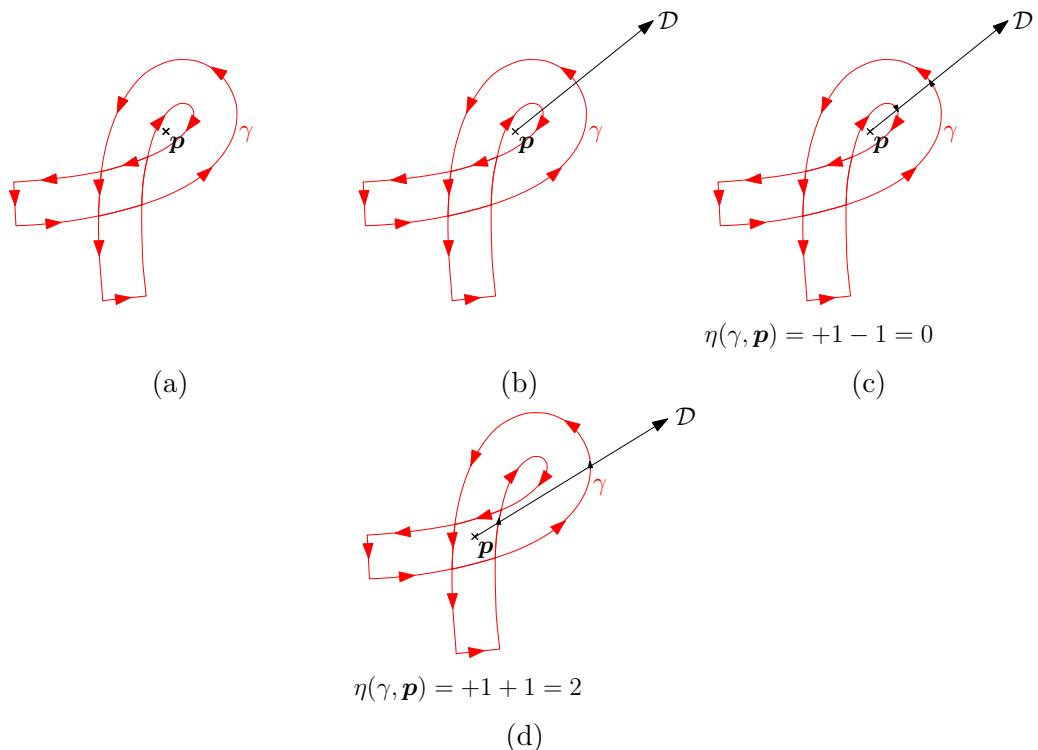


Figure 6.8: In [131] an algorithm was presented for computing the winding number of a loop with respect to a point in the plane. In this figure we demonstrate this algorithm for cycle γ and point p as illustrated in Figure (a); (b): We start by defining an infinite ray \mathcal{D} from p ; (c): We identify the intersections of γ with \mathcal{D} . We sum $+1$ for the ones crossing \mathcal{D} from right to left and -1 for the ones crossing it from left to right. For this example the result is $\eta(\gamma, p) = +1 - 1 = 0$. (d): We apply the method to another point p such that γ intersects the new \mathcal{D} twice, crossing it from right to left in both cases. Therefore, for this example, we have $\eta(\gamma, p) = +1 + 1 = 2$.

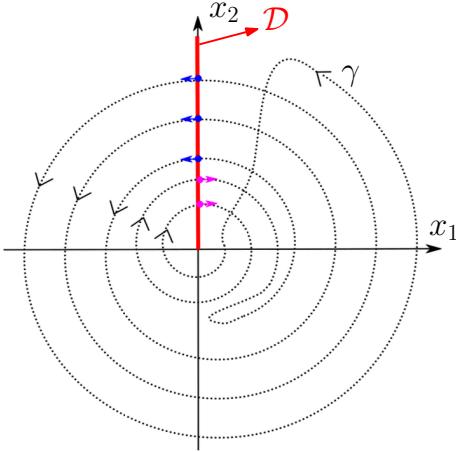


Figure 6.9: We consider a more complex cycle and we choose $\mathcal{D} = \{0\} \times [0, 1]$. The objective is to demonstrate that the axis-crossing method facilitates the visual calculation of winding numbers. Here, for example, we can easily determine that intersections of γ with \mathcal{D} that are represented in blue sum +1 and intersections represented in pink sum -1. And finally we can conclude that $\eta(\gamma, \mathbf{0}) = +1$.

of each crossing, from its start at 0 till some instant $t \in [0, 1]$. By definition

$$w_\gamma(t) = \#\mathbb{T}^+(t) - \#\mathbb{T}^-(t) \quad (6.13)$$

and we have

$$\eta(\gamma, \mathbf{0}) = w_\gamma(1) \quad (6.14)$$

This equation can be easily generalized to $\mathbf{p} \in \mathbb{R}^2 \setminus \gamma$ considering that

$$\eta(\gamma, \mathbf{p}) = \eta(\gamma - \mathbf{p}, \mathbf{0}) \quad (6.15)$$

Where $\gamma - \mathbf{p}$ is a displacement of the original cycle γ where point \mathbf{p} is mapped to $\mathbf{0}$. In this context, (6.14) can be used in order to compute the winding number of this new cycle. A generalization for computing the winding number of γ with respect to a point $\mathbf{p} \in \gamma$ is straightforward considering the extended winding number definition presented in Section 5.1.2.

Implementation

In practice, we want to apply this method for a tube $[\gamma]$, and in the context of the area characterization problem, we want to compute the extended winding number of $[\gamma]$ with respect to boxes. With this purpose, we developed an algorithm for computing the extended winding number of a tube $[\gamma]$ with

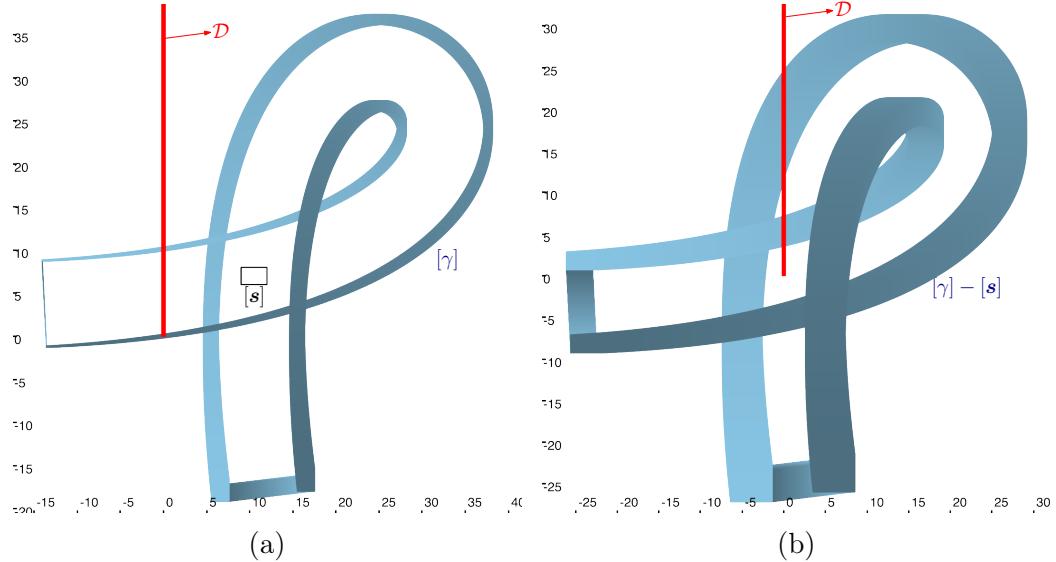


Figure 6.10: (a): The tube represents the uncertain mission contour $[\gamma]$ and we want to determine its winding number with respect to box $[s]$; (b): Instead of computing $[\bar{\eta}]([\gamma], [s])$, we compute $[\bar{\eta}]([\gamma] - [s], \mathbf{0})$.

respect to a box $[s]$. The algorithm analyzes how $[\gamma] - [s]$ intersects the infinite ray $\mathcal{D} = \{0\} \times [0, \infty]$ and computes $[\bar{\eta}]([\gamma] - [s], \mathbf{0})$, that is equal to $[\bar{\eta}]([\gamma], [s])$, see Figure 6.10.

We represent the cross number of $[\gamma]$ by the set-valued function $[w] : [0, 1] \rightarrow \mathbb{IZ}$ that we define as follows:

$$[w](t) = \bigsqcup_{\gamma \in [\gamma]} w_\gamma(t) \quad (6.16)$$

and $[\bar{\eta}]([\gamma], [s]) = [w](1)$.

The algorithm starts by setting $[\gamma] = [\gamma] - [s]$ and this new tube is divided into slices. Denote $[\gamma](\tau) \in \mathbb{IR}^2$ a slice of the tube. It is the smallest box such that

$$[\gamma](\tau) = \{\mathbf{p} \in \mathbb{R}^2 \mid \exists \tau \in [\tau] \text{ and } \gamma \in [\gamma] \text{ s.t. } \gamma(\tau) = \mathbf{p}\} \quad (6.17)$$

We start by analyzing slice $[\gamma](0, 1)$, and we verify if it intersects \mathcal{D} . If this is not the case, then $[\bar{\eta}]([\gamma], [s]) = [0, 0]$. Otherwise, we divide $[\gamma]$ into a finite number $k \in \mathbb{N}$ of consecutive slices $[\gamma]_1, [\gamma]_2, \dots, [\gamma]_k$. The division is made such that there is no slice $[\gamma]_i$, with $i \in \{1, 2, \dots, k\}$, for which $\mathbf{0} \in [\gamma]_i$, $\mathbf{0} \in [\dot{\gamma}_1]_i$ and $\mathbf{0} \in [\dot{\gamma}_2]_i$. If such a division is impossible, the algorithm returns $[\bar{\eta}]([\gamma], [s]) = [-\infty, \infty]$, because it can not determine the number of times a curve in $[\gamma]$ might wind around zero.

Let $[\tau_i]$ be a subset of $[0, 1]$ such that $[\gamma]_i = [\gamma]([\tau_i])$, we know that

$$\bigcup_{i=1}^k [\tau_i] = [0, 1] \quad (6.18)$$

and $[\tau_i] - [\tau_j] \subset \mathbb{R}^- \cup \{0\}$ if $j > i$ and $i, j \in \{1, 2, \dots, k\}$. We use notation $[w]_i$ for representing $[w](\tau_i^+)$.

Assuming that $[\gamma]_1 \cap \mathcal{D} = \emptyset$, we can initialize $[w]_1 = [0, 0]$. Then, we analyze the next slices in order, and we update the uncertain cross number value on each slice according to the following rules:

1. Rule 1:

$$\begin{cases} [\gamma]_i \cap \mathcal{D} = \emptyset \\ |i - j| = 1 \\ [\gamma]_j \cap \mathcal{D} = \emptyset \end{cases} \Rightarrow [w]_i = [w]_j$$

2. Rule 2:

$$\begin{cases} [\gamma]_i \cap \mathcal{D} \neq \emptyset \\ \mathbf{0} \notin [\gamma]_i \\ i - j = 1 \end{cases} \Rightarrow [w]_i = [w]_j + [\varepsilon]$$

with

$$[\varepsilon] = \begin{cases} [0, 1] & \text{if } [\gamma_1]_j \subset \mathbb{R}^+ \\ [-1, 0] & \text{if } [\gamma_1]_j \subset \mathbb{R}^- \\ [0, 0] & \text{otherwise} \end{cases}$$

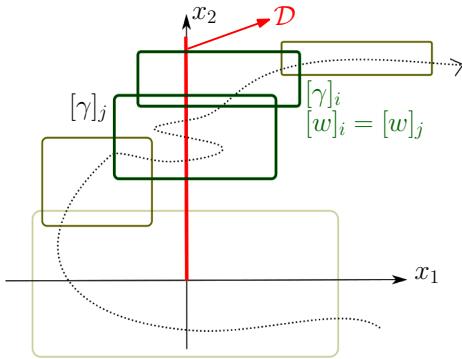


Figure 6.11: Example of application of Rule 2.

Figure 6.11 provides an example of application of rule 2.

3. Rule 3:

$$\begin{cases} \mathbf{0} \in [\gamma]_i \\ 0 \notin [\dot{\gamma}_1]_i \text{ or } 0 \notin [\dot{\gamma}_2]_i \\ i - j = 1 \\ \mathbf{0} \notin [\gamma]_j \end{cases} \Rightarrow [w]_i = [w]_l + [-1, 1]$$

where $l = \max(1, \dots, j)$ s.t $[\gamma]_l \cap \mathcal{D} = \emptyset$.

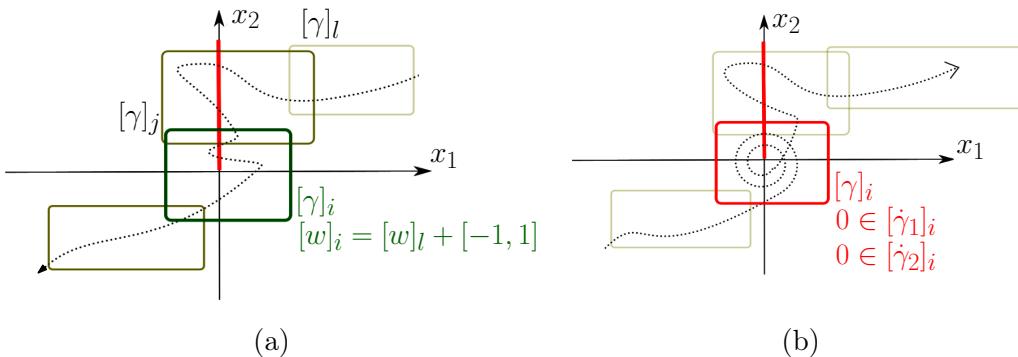


Figure 6.12: (a): Example of application of rule 3; (b) : Rule 3 can not be applied, and the algorithm returns $[-\infty, \infty]$.

The second condition, concerning the derivatives of $[\gamma]_i$, determines that there is no $\gamma \in [\gamma]$ that winds around zero inside this slice. This is illustrated in Figure 6.12.

4. Rule 4:

$$\begin{cases} [\gamma]_i \cap \mathcal{D} = \emptyset \\ i - j = 1 \\ [\gamma]_j \cap \mathcal{D} \neq \emptyset \end{cases} \Rightarrow \begin{cases} [w]_i = [[w]_j^-, [w]_j^+ - 1] \text{ if } [\gamma_1]_i \subset \mathbb{R}^+ \text{ and } [\gamma_1]_l \subset \mathbb{R}^- \\ [w]_i = [[w]_j^- + 1, [w]_j^+] \text{ if } [\gamma_1]_i \subset \mathbb{R}^- \text{ and } [\gamma_1]_l \subset \mathbb{R}^+ \end{cases}$$

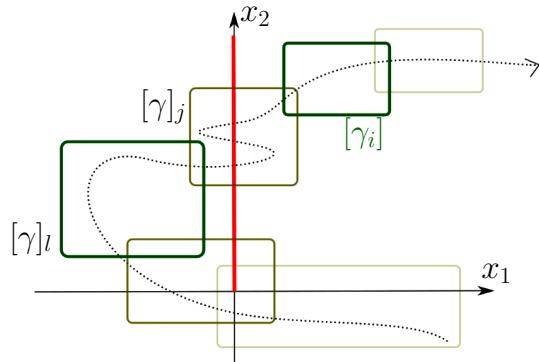
where $l = \max(1, \dots, j - 1)$ s.t $[\gamma]_l \cap \mathcal{D} = \emptyset$.

Figure 6.13 provides an example of application of this rule.

5. Rule 5:

$$\begin{cases} \mathbf{0} \in [\gamma]_i \\ 0 \notin [\dot{\gamma}_1]_i \text{ or } 0 \notin [\dot{\gamma}_2]_i \\ i - j = 1 \\ \mathbf{0} \in [\gamma]_j \end{cases} \Rightarrow \begin{cases} [w]_i = [w]_j, \\ \text{if } 0 \notin [\dot{\gamma}_1](\tau_j^-, \tau_i^+) \text{ or } 0 \notin [\dot{\gamma}_2](\tau_j^-, \tau_i^+) \\ [w]_i = [w]_j + [-1, 1], \\ \text{otherwise} \end{cases}$$

These rules can be easily proven considering that γ is continuous.



$$[w]_j = [w]_l + [-1, 0]$$

$$[w]_i = [[w_j]^-, [w_j]^+ - 1] = [w]_l + [-1, -1]$$

Figure 6.13: Example of application of rule 4.

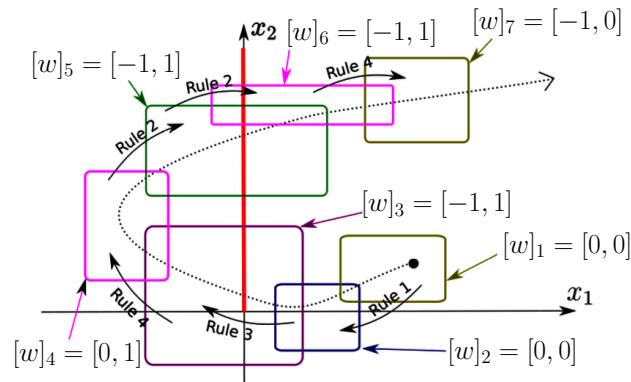


Figure 6.14: Example of application of rules for the update of the initial cross number $[w]_1$.

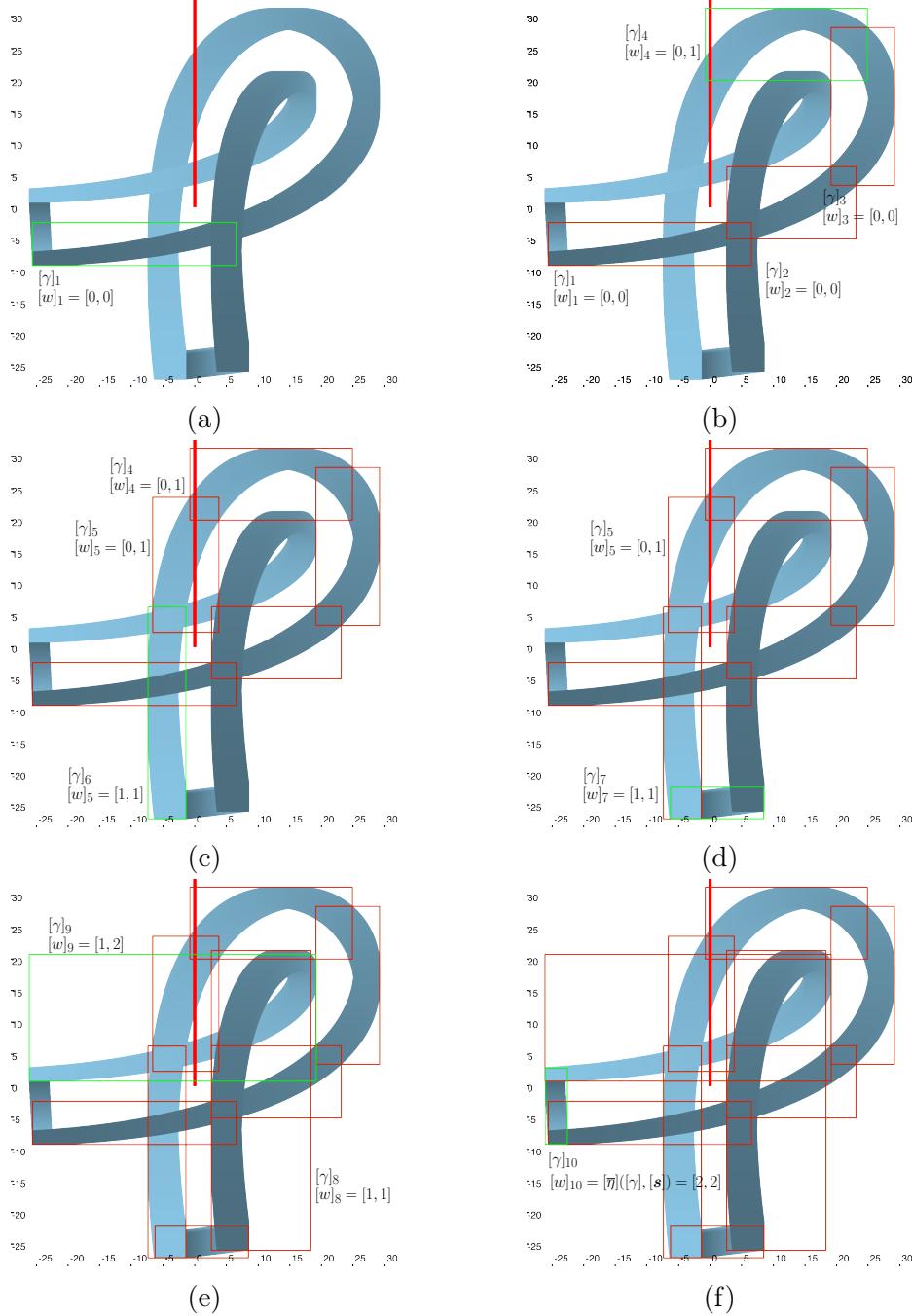


Figure 6.15: These figures demonstrate the steps of the turning around algorithm for computing the winding number of tube $[\gamma]$ with respect to box $[\mathbf{s}]$, as illustrated in Figure 6.10. (a): The first slice of the new tube does not intersect \mathcal{D} , and we can initialize the output with $[w]_1 = [0, 0]$; (b): We apply rule 1 consecutively to obtain $[w]_2 = [0, 0]$ and $[w]_3 = [0, 0]$ and rule 2 to obtain $[w]_4 = [0, 1]$; (c): We apply rule 2 again for $[\gamma]_5$ and rule 4 on $[\gamma]_6$ to obtain $[w]_6 = [1, 1]$; (d): We apply rule 1 at $[\gamma]_7$ and at $[\gamma]_8$; (e): Rule 2 is applied on $[\gamma]_9$ and we have $[w]_9 = [1, 2]$; (f): Finally, we apply rule 4 to obtain $[w]_{10} = [2, 2]$ and $[w]_{10} = [\bar{\eta}]([\gamma], [\mathbf{s}]) = [\bar{\eta}]([\gamma - [\mathbf{s}], \mathbf{0}])$.

Figure 6.14 provides a simple example of the application of these rules and Figure 6.15 an overview of the steps of the algorithm for winding number computation.

Algorithm 4 *Area Characterization (Axis-Crossing Method)*

in: $S, \{[\gamma]\}, \{[\dot{\gamma}]\}, \epsilon$

- 1: $\mathbb{Q} \leftarrow S$
 - 2: **if** $\mathbb{Q} \neq \emptyset$ **then** take a box $[s] \in \mathbb{Q}$ and remove it from \mathbb{Q} **else return**
endif
 - 3: $[c_m] = [0, 0]$ ▷ Initialize the coverage measure of the box with $[0, 0]$
 - 4: **for** $([\gamma], [\dot{\gamma}])$ in $(\{[\gamma]\}, \{[\dot{\gamma}]\})$ **do**
 ▷ Compute the extended winding number of $([\gamma], [\dot{\gamma}])$ w.r.t. $[s]$ using the axis-crossing method and sum the result to $[c_m]$.
 - 5: **end for**
 - 6: **if** $[c_m]$ is a singleton **then**
 ▷ The coverage measure of points in $[s]$ can be uniquely determined.
 - 7: **elseif** $w([s]) < \epsilon$ or $([s] \in [\gamma], \text{ for any } [\gamma] \in \{[\gamma]\}, \text{ and } c_m^+ - c_m^- = 1)$ **then**
 ▷ The box is already smaller than the required precision ϵ or its coverage measure value can not be refined through bisection.
 - 8: **else**
 11: bisect $[s]$ into $[s]^{(1)}$ and $[s]^{(2)}$
 12: $\mathbb{Q} \leftarrow \mathbb{Q} \cup \{[s]^{(1)}, [s]^{(2)}\}$
endif
 - 13: Return to line 2.
-

In order to do an area characterization of box S in terms of the coverage measure of its points, this method can be applied. Algorithm 4 was developed with this purpose, where set $\{[\gamma]\}$ represents the union of tube $[\gamma^+]$ with all the tubes from set $\{[\gamma^-]\}$. These tubes are previously computed considering the context of the mission.

Utilizing this method for area characterization is not optimized due to the need to apply the axis-crossing method to compute the winding number for every box in the paving. Depending on the characteristics and uncertainty associated with $[\gamma]$, parsing slices could become more time-consuming, potentially leading to overly pessimistic results. Furthermore, this approach does

not leverage the property of winding numbers to be constant within connected regions in $\mathbb{R}^2 \setminus [\gamma]$. The next approach, that we present in the following section, Section 6.3.2, uses this property and significantly reduces the computational complexity of the area characterization computation.

6.3.2 Method 2: Alexander Theorem

Let $\gamma : [0, 1] \rightarrow \mathbb{R}^2$ be a closed path such that $\gamma(0) = \gamma(1)$. A self-intersection, or vertex, of loop γ is determined by two parameters $\tau_0, \tau_1 \in (0, 1)$, with $\tau_0 \neq \tau_1$, and it is a point \mathbf{p} such that $\mathbf{p} = \gamma(\tau_0) = \gamma(\tau_1)$. The multiplicity of such self-intersection is the number, finite or infinite, of distinct $\tau \in [0, 1]$ such that $\mathbf{p} = \gamma(\tau)$, minus one. Then, we make the following assumptions, similar to those of [127]:

- γ has a finite number of self-intersections, each one of them with multiplicity one;
- At each self-intersection, the two tangent vectors to γ are linearly independent.

Such a cycle divides $\mathbb{R}^2 \setminus \gamma$ into a finite number of connected open regions, one of which is not compact. Each one of these regions can be seen as a *2-cell* of the CW complex $C(\gamma)$, constructed from the cycle γ . To be fully formal, we would need to use the fact that γ determines a cell decomposition of the one-point compactification of the plane, homeomorphic to the 2-sphere S_2 , Figure 6.16. The 0-cells of $C(\gamma)$ are self-intersections of γ , and the 1-cells are connected components of γ minus its self-intersections, it can also be seen as parts of the curve separating the 2-cells.

Since all open 2-cells are homotopy equivalent to a point within that cell and considering the degree axioms presented in Definition 27, we can conclude that all the points within the same open 2-cell of $C(\gamma)$ have the same winding number value. In this context, a correct and coherent numbering of the 2-cells is enough for determining the winding number value of all the points in the plane.

With this purpose, we can use a combinatorial rule proposed by Möbius in 1865 [132]. The rule says that two contiguous 2-cells that are separated by a 1-cell are numbered with a value that must differ by exactly 1. The winding number of the region on the left, considering the orientation of the curve, is greater. This method leads to a unique numbering of the space considering that the winding number in the non-compact region, that we name A_0 , is known and equal to 0 for all of its points. This is true because, if we assume that γ is the image of the boundary of the waterfall space ∂W by the sweep

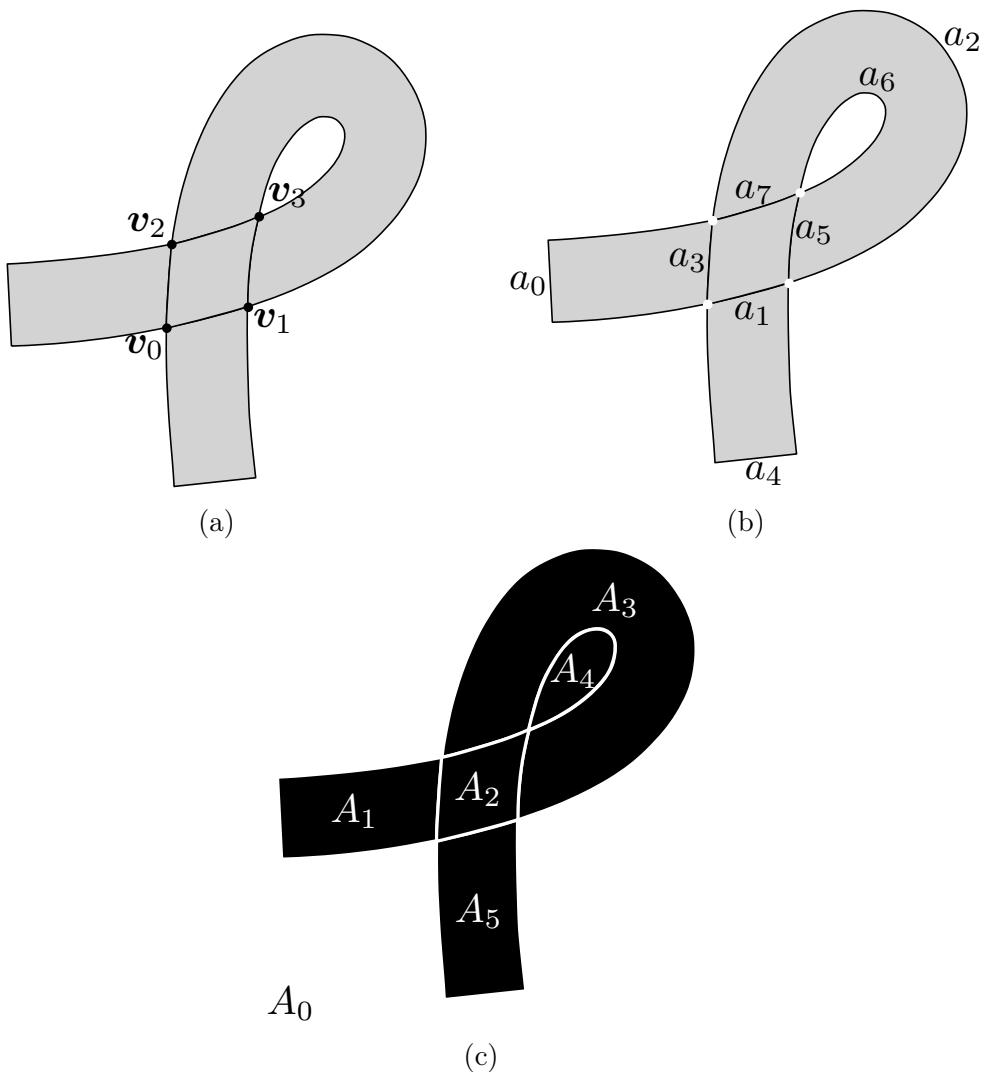


Figure 6.16: (a): $CW(\gamma)$ is the CW complex created from cycle γ . In this example, it has four 0-cells $\{v_0, v_1, v_2, v_3\}$; (b): eight 1-cells, connected components of $\gamma \setminus \{v_0, v_1, v_2, v_3\}$; (c): The plane is divided into six 2-cells, five compacts (from A_1 to A_5) and one extending to the infinity (A_0).

function \mathbf{f} and that A_0 is not bounded by γ , differently from the other 2-cells of $C(\gamma)$, then we know that $A_0 \subseteq \mathbb{R}^2 \setminus \mathbf{f}(W)$. This implies, from Definition 28, that for any $\mathbf{p} \in A_0$, $\eta(\gamma, \mathbf{p}) = 0$.

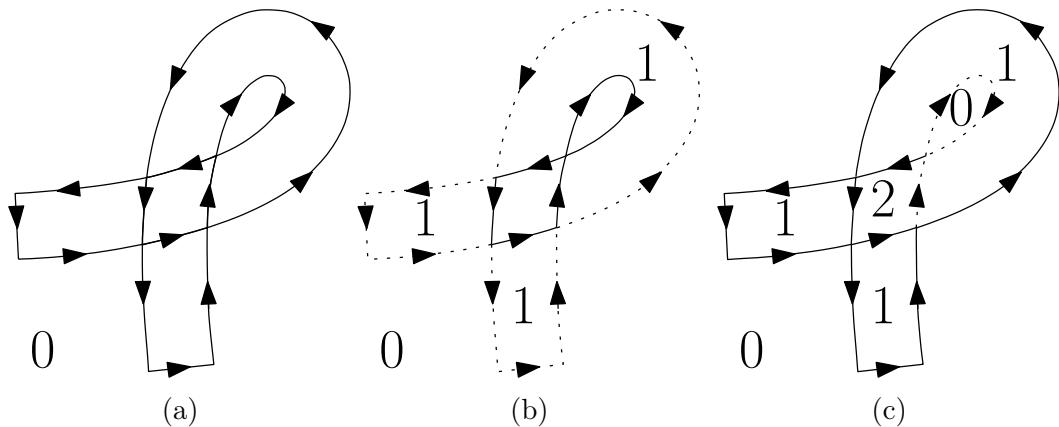


Figure 6.17: Propagation of the winding number value through regions that share a boundary according to Möbius.

In Figure 6.17 we apply this rule. We start with the numbering of A_0 with 0 in Figure 6.17a. Then, all its adjacent 2-cells can be numbered according to Möbius. Using cell names proposed in Figure 6.16, we have cells A_1 , A_3 , and A_5 numbered with +1. This happens because they are on the left side of the 1-cells, represented by a dashed line in Figure 6.17b, separating them from the non-compact region. Then, in Figure 6.17c, cells A_2 and A_4 can be respectively numbered with 2 and 0. The first is because it is on the left side of the boundary a_5 , shared with A_3 , and the second because it is on the right side of the boundary a_6 , also shared with A_3 .

As a direct application of Möbius rules, a method proposed by Alexander [127] allows a coherent numbering of the regions only through an analysis of the tangent vectors to the curve on its self-intersections.

Let $\mathbf{v} \in \mathbb{R}^2$ be a self-intersection of γ represented by the pair (τ_0, τ_1) with $\tau_1 > \tau_0$. Considering the assumptions adopted for γ , a self-intersection \mathbf{v} will divide the plane into four regions. There are only two rules for numbering these four regions, according to whether $\dot{\gamma}(\tau_1)$ crosses $\dot{\gamma}(\tau_0)$ from the right to the left or from the left to the right, as illustrated in Figure 6.18.

In Figure 6.19 we consecutively apply the Alexander numbering rules to the example considered previously. We start by numbering regions around \mathbf{v}_2 , Figure 6.19a. For that, we assume that A_0 has a winding number value of 0 and that the later self-intersection, represented by the dashed line, crosses the previous one from left to the right. The same is done around vertices

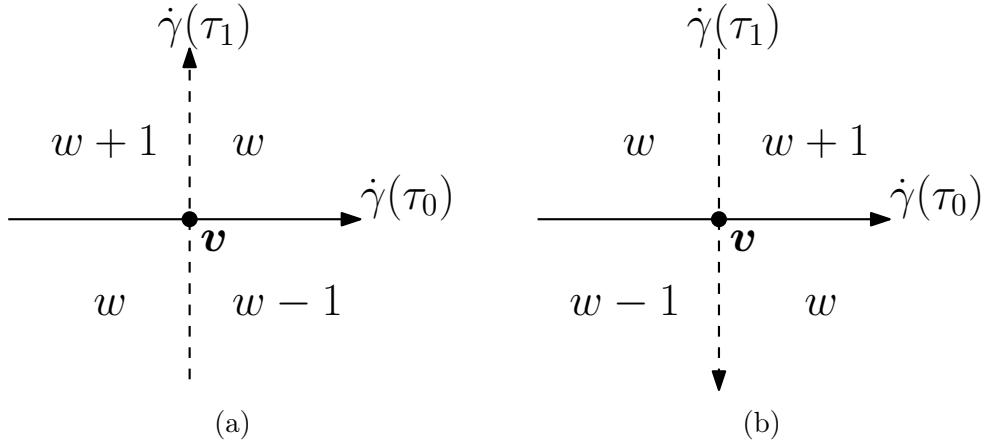


Figure 6.18: Alexander numbering of 2-cells surrounding a self-intersection represented by point $\mathbf{v} = \gamma(\tau_0) = \gamma(\tau_1)$ with $w \in \mathbb{Z}$ if : (a): $\dot{\gamma}(\tau_1)$ crosses $\dot{\gamma}(\tau_0)$ from right to left; (b): $\dot{\gamma}(\tau_1)$ crosses $\dot{\gamma}(\tau_0)$ from left to right.

\mathbf{v}_3 and \mathbf{v}_1 on Figures 6.19b and 6.19c, respectively, resulting in a complete characterization of 2-cells in terms of winding number values.

To fully characterize the plane, it is essential to number points on the 1-cells and the 0-cells based on their extended winding numbers. Within this framework, we extend Alexander rules so points within a 1-cell of γ have assigned to them the same winding number value as the 2-cell located on their left side. As for the 0-cells of γ , they are numbered based on the 2-cell with the highest winding number value among the four that surround each individual 0-cell. The resultant complete numbering has edges a_1, a_3, a_5 and a_7 , as well as vertices $\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2$ and \mathbf{v}_3 numbered with +2, and edges a_0, a_2, a_4 and a_6 numbered with +1.

Once a complete numbering of the plane is done we can define the winding sets of γ . We denote \mathbb{W}_γ^i a winding set associated with γ and with a natural number i , by definition

$$\mathbb{W}_\gamma^i := \{\mathbf{p} \in \mathbb{R}^2 \mid \bar{\eta}(\gamma, \mathbf{p}) \geq i\} \quad (6.19)$$

There are, for example, two non-empty winding sets associated with the curve γ considered on this section, as illustrated on Figure 6.20. They are \mathbb{W}_γ^1 and \mathbb{W}_γ^2 .

Let $\chi_{\mathbb{W}_\gamma^i} : \mathbb{R}^2 \rightarrow \{0, 1\}$ be the characteristic function for a winding set \mathbb{W}_γ^i , we have

$$\chi_{\mathbb{W}_\gamma^i}(\mathbf{p}) = \begin{cases} 1, & \text{if } \mathbf{p} \in \mathbb{W}_\gamma^i, \\ 0 & \text{otherwise} \end{cases} \quad (6.20)$$

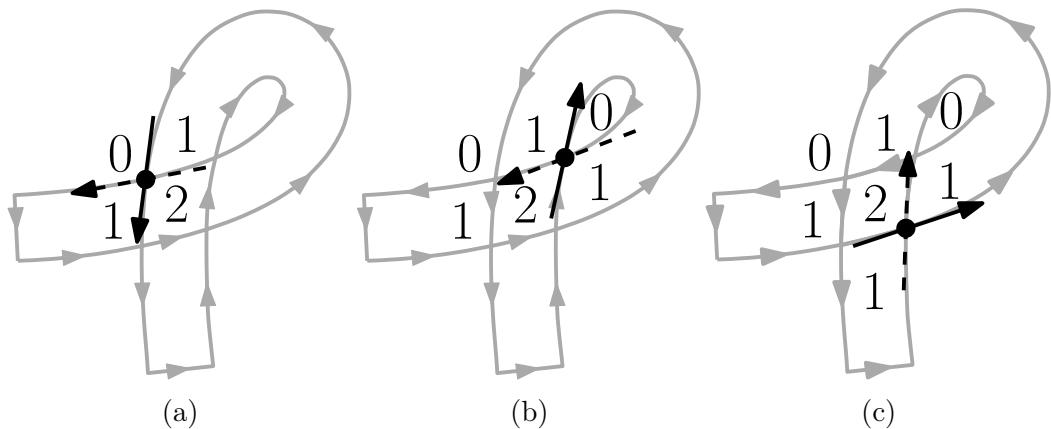


Figure 6.19: Numbering of regions according to Alexander around : (a): \mathbf{v}_2 ; (b): \mathbf{v}_3 and (c): \mathbf{v}_1 . The dashed line represents the later tangent vector to γ on the self-intersection, and we analyze if it crosses the solid vector from right to left or left to right in order to decide which Alexander rule to apply on the numbering of the concerned 2-cells. Notice that the numbering of all the 2-cells can be done by analyzing different 0-cells than the ones chosen, or in a different order, but the result will be the same.

Then, as proved in [133], the winding number $\eta(\gamma, \mathbf{p})$ of a point $\mathbf{p} \in \mathbb{R}^2 \setminus \gamma$, and therefore its extension $\bar{\eta}(\gamma, \mathbf{p})$ for any $\mathbf{p} \in \mathbb{R}^2$, can be calculated using the winding sets of γ ,

$$\bar{\eta}(\gamma, \mathbf{p}) = \sum_{i>0} \chi_{W_i}(\mathbf{p}) \quad (6.21)$$

Uncertain Alexander Numbering

In practice, cycle γ is represented by a tube $[\gamma] : [0, 1] \rightarrow \mathbb{IF}^2$, and now we adapt the original Alexander theorem presented in this section to deal with this context. We will be using the tube $[\gamma]$ represented in Figure 6.21 to illustrate the ideas that are developed next.

The notion of self-intersections on tubes, and how to detect them, was extensively discussed in Section 3.7. We assume that all self-intersections of $[\gamma]$ are represented by two parameters $[\tau_1] \in \mathbb{IR}$ and $[\tau_2] \in \mathbb{IR}$ such that $[\tau_1] - [\tau_2] \subset \mathbb{R}^-$ and we make the following assumptions:

- $\forall \gamma \in [\gamma], \exists! (\tau_1, \tau_2) \in [\tau_1] \times [\tau_2] \mid \gamma(\tau_1) = \gamma(\tau_2)$;
- At each self-intersection, the two tangent vectors to $[\gamma]$, $[\dot{\gamma}]([\tau_1])$ and $[\dot{\gamma}]([\tau_2])$, are linearly independent. In other words

$$0 \notin ([\dot{\gamma}]([\tau_1]) \times [\dot{\gamma}]([\tau_2])) \quad (6.22)$$

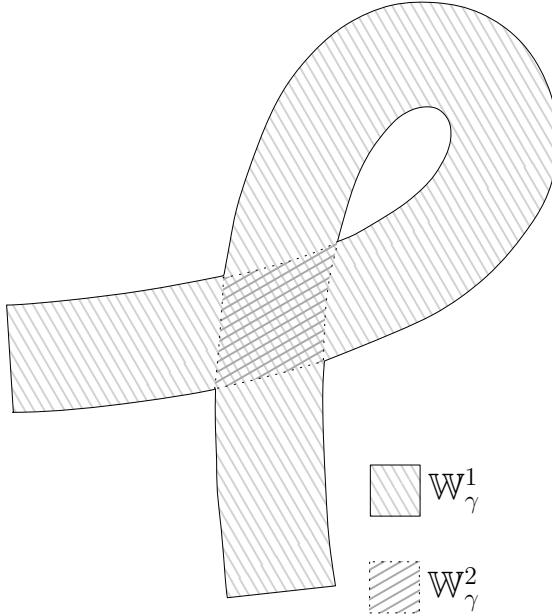


Figure 6.20: Winding sets \mathbb{W}_γ^1 and \mathbb{W}_γ^2 associated with curve γ from the illustrating example used on this section.

- $[\gamma]$ has a finite number of self-intersections, each one of them with multiplicity one.

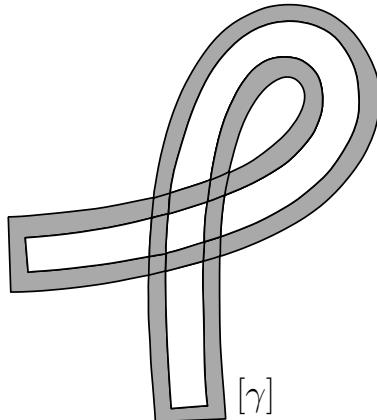
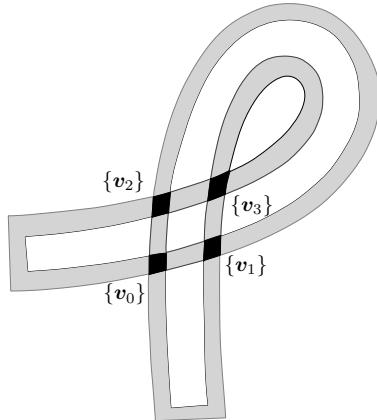
We can then define $C([\gamma])$, that is an equivalent of the CW-complex for a set of cycles $[\gamma]$. In this scenario, a 0-cell is no longer a point in \mathbb{R}^2 but a set of points of possible self-intersections of $[\gamma]$ within a period of time. Or, it can also be seen as the union of 0-cells of each $\gamma \in [\gamma]$. For example, each 0-cell is represented by $\{\mathbf{v}\} \in \mathcal{P}(\mathbb{R}^2)$ and it is associated with a self-intersection of the tube that is defined by parameters $[\tau_1]$ and $[\tau_2]$, and that respects the conditions listed above. Then, we have

$$\{\mathbf{v}\} = \{\mathbf{v} \in \mathbb{R}^2 \mid \exists \gamma \in [\gamma], \exists (\tau_1, \tau_2) \in [\tau_1] \times [\tau_2] \text{ s.t. } \gamma(\tau_1) = \gamma(\tau_2) = \mathbf{v}\} \quad (6.23)$$

These sets are illustrated in 6.22 for the example considered here. A 1-cell is a connected component of $[\gamma]$ after removing its 0-cells, and 2-cells of $C([\gamma])$ are connected components of $\mathbb{R}^2 \setminus [\gamma]$.

For any $\mathbf{p} \in \mathbb{R}^2$, we have $[\bar{\eta}]([\gamma], \mathbf{p})$ that is the interval union of the extended winding number value of all $\gamma \in [\gamma]$ with respect to \mathbf{p} . Formally $[\bar{\eta}]([\gamma], \mathbf{p}) \in \mathbb{IR}$ and

$$[\bar{\eta}]([\gamma], \mathbf{p}) = \bigsqcup_{\gamma \in [\gamma]} \bar{\eta}(\gamma, \mathbf{p}) \quad (6.24)$$

Figure 6.21: Example of tube $[\gamma]$.Figure 6.22: 0-cells of $C([\gamma])$ representing all the possible self-intersections in $[\gamma]$.

Notice that 2-cells are still homotopy equivalent to a point within the cell, and therefore all the points inside the same 2-cell have the same winding number value. Furthermore, this value is independent of the cycle $\gamma \in [\gamma]$ and $[\bar{\eta}]([\gamma], \mathbf{p})$ is a singleton for any \mathbf{p} in a 2-cell of $C([\gamma])$.

If $([\dot{\gamma}]([\tau_1]) \times [\dot{\gamma}]([\tau_2])) \subset \mathbb{R}^-$, we can conclude that $[\dot{\gamma}]([\tau_2])$ crosses $[\dot{\gamma}]([\tau_1])$ from the left to right, otherwise it crosses from the right to the left. Using this information, the fours 2-cells surrounding the self-intersection represented by pair $([\tau_1], [\tau_2])$ can be numbered using the same Alexander rules presented previously.

The extended winding number value for points on 1-cells and 0-cells, on the other hand, will differ depending on cycle $\gamma \in [\gamma]$. Therefore, these cells are numbered with their uncertain extended winding number values. This uncer-

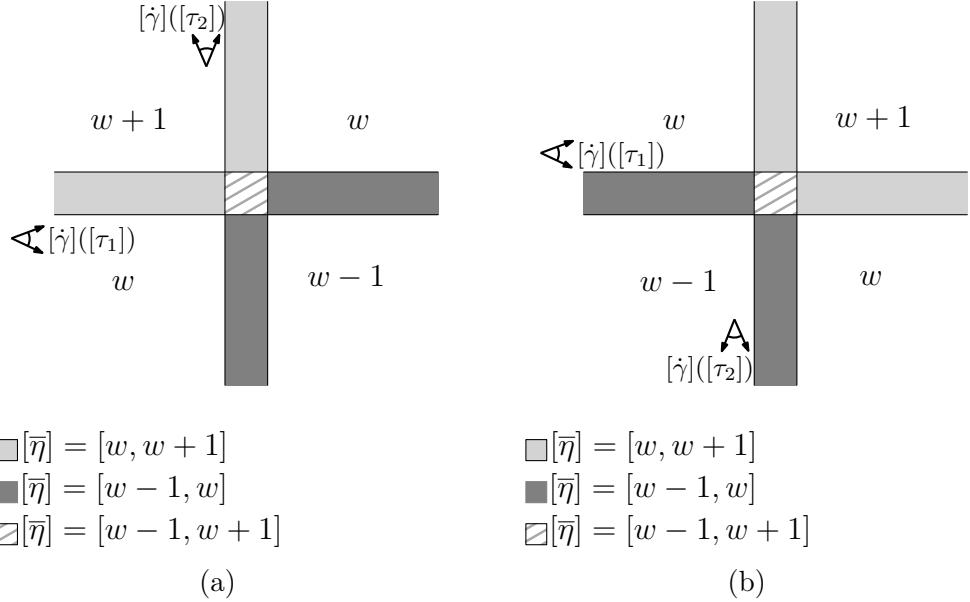


Figure 6.23: Uncertain Alexander numbering with $w \in \mathbb{Z}$ if: (a): $[\dot{\gamma}](\tau_2)$ crosses $[\dot{\gamma}](\tau_1)$ from the right to the left. Equivalently we have $([\dot{\gamma}](\tau_1)) \times ([\dot{\gamma}](\tau_2)) \subset \mathbb{R}^+$; (b): $[\dot{\gamma}](\tau_2)$ crosses $[\dot{\gamma}](\tau_1)$ from the left to the right. Equivalently we have $([\dot{\gamma}](\tau_1)) \times ([\dot{\gamma}](\tau_2)) \subset \mathbb{R}^-$. The 2-cells are numbered as they are numbered on the original rules, the 1-cells and 0-cells are numbered with their uncertain extended winding number value $[\bar{\eta}]$.

tain numbering can be obtained from the 2-cells for which 1-cells and 0-cells are a boundary, and considering the conditions established for $[\gamma]$, there are only two rules for numbering all the cells of $C([\gamma])$. These rules are illustrated on Figure 6.23 and constitute an extension of the original Alexander rules to deal with uncertainty.

To the tube $[\gamma]$ we associate winding sets represented by thick sets, that were introduced in Chapter 3, Section 3.6.1. We denote $[\mathbb{W}_{[\gamma]}^i] = [\mathbb{W}_i^-, \mathbb{W}_i^+]$, for $i \in \mathbb{N}$, a winding set associated to tube $[\gamma]$, such that

$$\mathbb{W}_i^- = \bigcap_{\gamma \in [\gamma]} \mathbb{W}_\gamma^i \quad (6.25)$$

$$\mathbb{W}_i^+ = \bigcup_{\gamma \in [\gamma]} \mathbb{W}_\gamma^i \quad (6.26)$$

A set-membership analysis to winding sets can be applied to determine the winding number of points, and therefore, considering the exploration context, to determine the coverage measure of these points. In this scenario, the

clear zone of a winding set $\llbracket \mathbb{W}_{[\gamma]}^i \rrbracket$, represented by \mathbb{W}_i^- , translates as a set of points that were certainly explored at least i times. Analogously, the dark zone $\mathbb{R}^2 \setminus \mathbb{W}_i^+$ is a set of points that have a coverage measure smaller than i , independently of which of the cycles in $[\gamma]$ is the ground truth for the mission contour. The penumbra $\mathbb{W}_i^+ \setminus \mathbb{W}_i^-$ is a set of points whose coverage measure might be equal or greater than i , depending on each of the cycles $\gamma \in [\gamma]$ is the ground truth for the exploration mission.

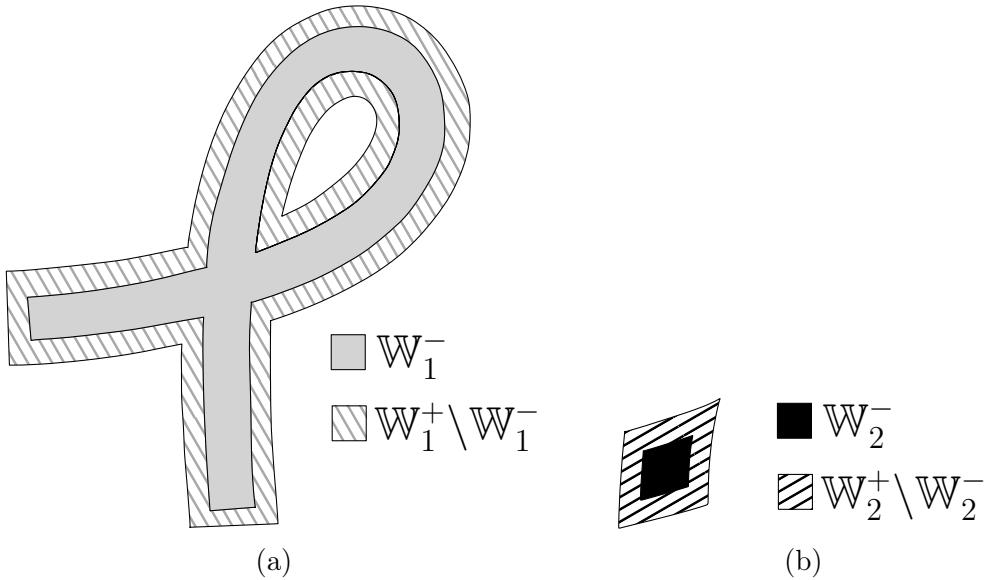


Figure 6.24: Uncertain winding sets associated with $[\gamma]$. On Figure (a) we have $\llbracket \mathbb{W}_{[\gamma]}^1 \rrbracket$ and on Figure (b) we have $\llbracket \mathbb{W}_{[\gamma]}^2 \rrbracket$.

We redefine the characteristic function to deal with thick sets on the plane, we have $[\chi] : \mathbb{R}^2 \rightarrow \mathbb{N}$ and

$$[\chi]_{\llbracket \mathbb{W}_{[\gamma]}^i \rrbracket}(\mathbf{p}) = \begin{cases} [1, 1], & \text{if } \mathbf{p} \in \mathbb{W}_i^-, \\ [0, 1], & \text{if } \mathbf{p} \in \mathbb{W}_i^+ \setminus \mathbb{W}_i^-, \\ [0, 0], & \text{otherwise} \end{cases} \quad (6.27)$$

Then, adapting (6.21) we obtain

$$[\bar{\eta}]([\gamma], \mathbf{p}) = \sum_{i>0} \chi_{\llbracket \mathbb{W}_{[\gamma]}^i \rrbracket}(\mathbf{p}) \quad (6.28)$$

In Figure 6.24 we have an illustration of thick sets $\llbracket \mathbb{W}_{[\gamma]}^1 \rrbracket$ and $\llbracket \mathbb{W}_{[\gamma]}^2 \rrbracket$ for the example considered through this section and in Figure 6.25 the resultant uncertain extended winding number values considering these sets.

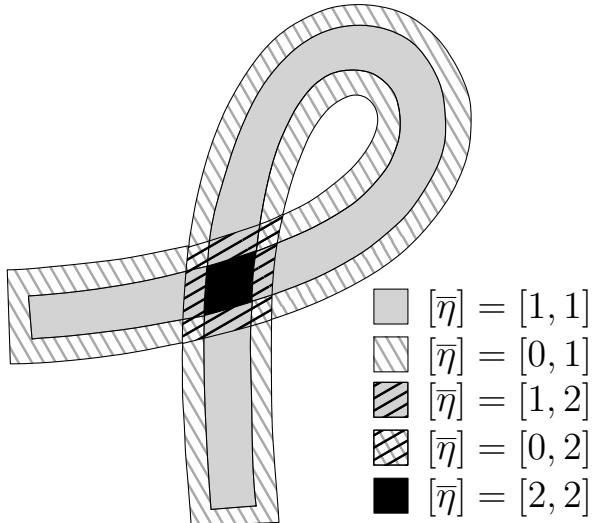


Figure 6.25: Graphical representation of the uncertain extended winding number value of $[\gamma]$ with respect to points in the plane.

Implementation

The first step of the implementation of this approach is to identify the self-intersections on tube $[\gamma^+]$ and on each of the tubes in $\{[\gamma^-]\}$. With this purpose, Algorithm 3 can be applied. Then, in order to determine if the tubes in question respect the conditions established by the method, we need to analyze if each of its self-intersections can be proved [134]. By proved, we mean if for each self-intersection $([\tau_1], [\tau_2])$ there is indeed $(\tau_1, \tau_2) \in [\tau_1] \times [\tau_2]$ such that $\gamma(\tau_1) = \gamma(\tau_2)$, for all $\gamma \in [\gamma]$. In addition, we verify the uniqueness of a solution within this self-intersection for each possible cycle in $[\gamma]$ with, for example, a Newton's approach [135], [106] or a topological approach [134]. Last verification consists of making sure that the derivatives of $[\gamma]$ on each self-intersection are linearly independent by analyzing its cross-product. From now, we explain the implementation considering an individual tube $[\gamma]$, knowing that the method is applied to each $[\gamma]$ in $[\gamma^+] \cup \{[\gamma^-]\}$.

The combination of 0-cells and 1-cells within $[\gamma]$ can be numerically represented as a graph, where 0-cells are the vertices and edges are the union of 1-cells with the vertices they are connecting. These elements have a particular structure carrying intrinsic information about $[\gamma]$, and they are related to each other in a well-defined structure. For example, vertices are the self-intersections that we computed on the precedent step. An edge is a subset of $[\gamma]$ defined within two consecutive intervals of time that can be deduced from its self-intersections.

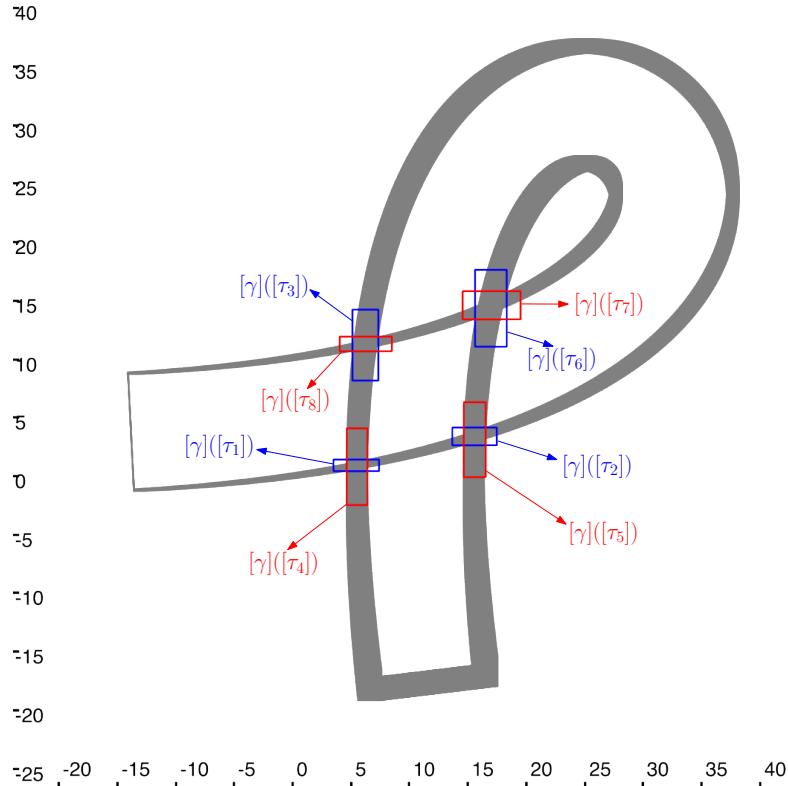


Figure 6.26: Self-intersections of tube $[\gamma]$ computed using Algorithm 3.

We are going to illustrate this using the tube constructed on the example presented in Figure 6.6. Let us consider Figure 6.26, we have four proved 0-cells in this example. These self-intersections are represented by pairs of time intervals $([\tau_1], [\tau_4]), ([\tau_2], [\tau_5]), ([\tau_3], [\tau_8]),$ and $([\tau_6], [\tau_7]),$ with $[\tau_i] - [\tau_j] \subset \mathbb{R}^-,$ for $i, j \in \{1, 2, 3, \dots, 8\}$ and $i < j.$ The precision of the estimation of time intervals representing a self-intersection will depend on the precision set by parameter ϵ on Algorithm 3. In practice, this estimation is guaranteed but usually has some pessimism. In this case, as it can be seen in Figure 6.26, time intervals $[\tau_i]$, for $i \in \{1, 2, 3, \dots, 8\}$ are not optimal but guaranteed to contain all possible self intersections of the tube.

Then, we have eight edges that are tubes $[\gamma](\tau_i^-, \tau_j^+)$, for $i \in \{1, 2, \dots, 7\}$ and $j = i + 1,$ plus the result of the concatenation $[\gamma](t) * [\gamma](t).$ These edges

$$t=\tau_8^- \rightarrow 1 \quad t=0 \rightarrow \tau_1^+$$

are represented in Figure 6.27.

Edges carry information about the initial τ_i^- and ending time τ_j^+ of the curve within $[\gamma]$ they represent, as well as the winding number value of points on their right side and left side. Additionally, each edge has an update pa-

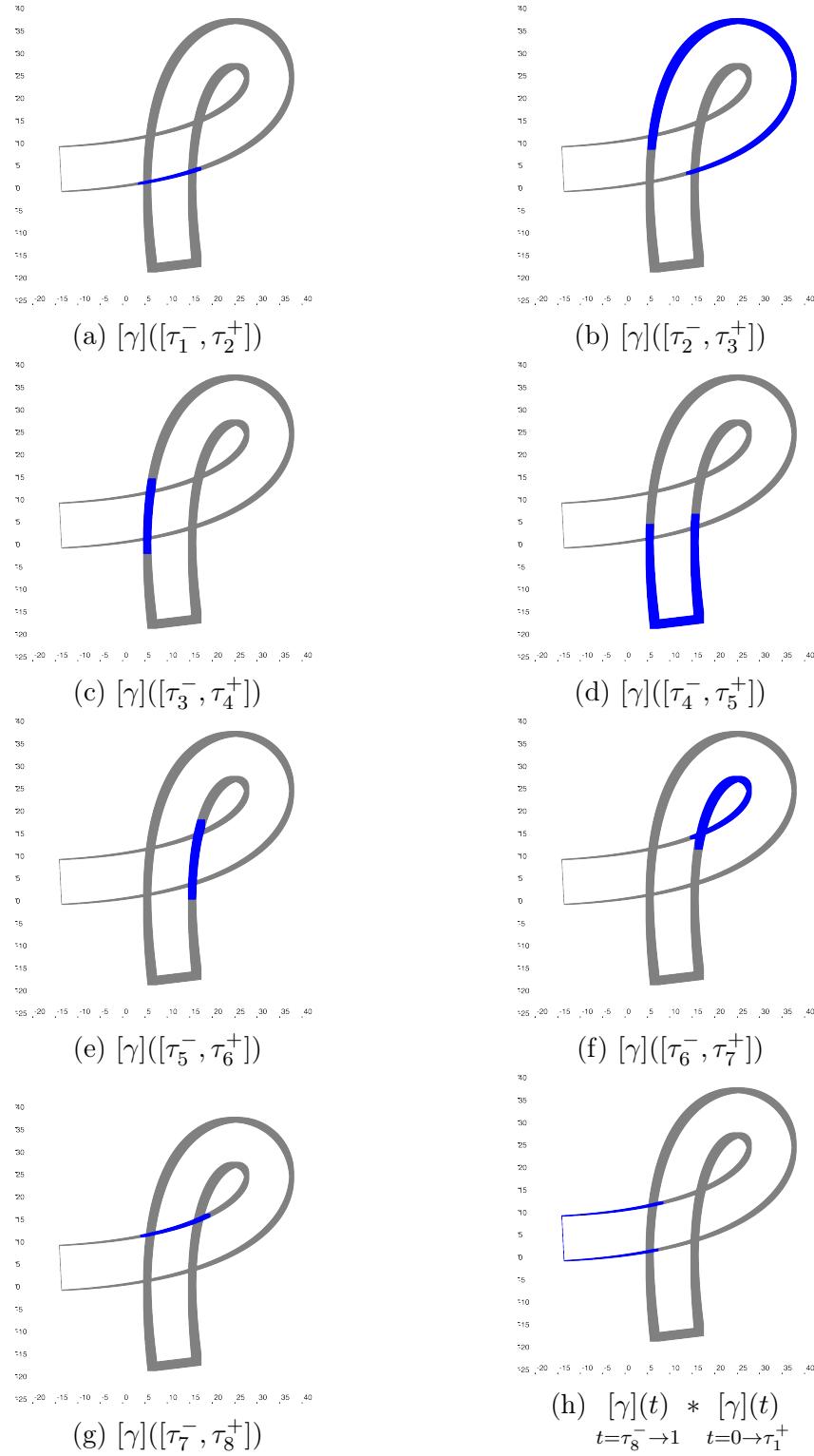


Figure 6.27: Edges of $[\gamma]$, each one representing the union of a 1-cell with the vertices they connect.

parameter $u \in \{-1, 1\}$ that determines how speed vectors on $[\gamma]$ at the initial vertex of the edge cross each other. For example, an update parameter $u = -1$ means that, at the initial vertex of this edge, we have the other speed vector that defines this self-intersection crossing $[\dot{\gamma}]([\tau_i])$ from the right to the left. We will have $u = +1$ if the crossing happens from the left to the right. This information is essential for propagating the winding number values that surround this edge through the other edges of the graph according to Alexander theorem.

We number the values surrounding the 1-cells using the two Alexander rules presented in Figure 6.23 and an arbitrary winding number value represented by variable $w \in \mathbb{Z}$. This numbering process is presented in Figure 6.28. We know that the smallest winding number value in the plane, in the exploration context considered here, is 0. In this case, we can replace variable w by real values, see Figure 6.28e.

Once the edges are numbered according to the winding number value on both of its sides, we can construct the winding sets associated to $[\gamma]$. To each non-empty winding set $\llbracket \mathbb{W}_{[\gamma]}^i \rrbracket$ we construct a separator \mathcal{S}^i . For that, we build two image contractors, as presented in Section 3.4.4, one representing the set of points that belong to \mathbb{W}_i^+ , and the other, for those inside $S \setminus \mathbb{W}_i^-$, where S is the subset of the space to be characterized.

To illustrate this step of the algorithm we present here how separators \mathcal{S}^1 and \mathcal{S}^2 for winding sets $\llbracket \mathbb{W}_{[\gamma]}^1 \rrbracket$ and $\llbracket \mathbb{W}_{[\gamma]}^2 \rrbracket$, associated to the mission exemplified in Figure 6.6, are created.

For $i \in \{1, 2\}$, we create a binary image $\mathcal{B}_{\mathbb{S}^i}^{in}$ where pixels representing a subset of the space with at least one point inside \mathbb{W}_i^+ are set to 1 and all the others are set to 0, as stated on (3.4) assuming $\mathbb{S} = \mathbb{W}_i^+$. For that, we start with $\mathcal{B}_{\mathbb{S}^i}^{in}$ filled by zeros. Then, we draw on the image the edges numbered with i on their left side, as demonstrated on Figures 6.29 and 6.30. By drawing we mean that we set to 1 pixels that represent these edges on the image. Then, using a graphical library, such as OpenCV in Python or C++ [136], we can fill the pixels inside the area surrounded by the closed cycle, resultant of the union of the edges drawn on the image, with 1, see Figures 6.29e and 6.30e. From this resulting binary image we create an image contractor \mathcal{S}_i^{out} for contracting points outside \mathbb{W}_i^+ .

Then, in order to construct separator \mathcal{S}^i , we also need a contractor \mathcal{S}_i^{in} for contracting points inside \mathbb{W}_i^- . We construct another binary image $\mathcal{B}_{\mathbb{S}^i}^{out}$ whose pixels representing a subset of the space with at least one point inside the solutions set $\mathbb{S} = S \setminus \mathbb{W}_i^-$ are set to 1 and all the others are set to 0. We can obtain $\mathcal{B}_{\mathbb{S}^i}^{out}$ by defining the value on each of its pixels as the result of the binary sum between their equivalent in $\mathcal{B}_{\mathbb{S}^i}^{in}$ and 1, in addition we redraw the

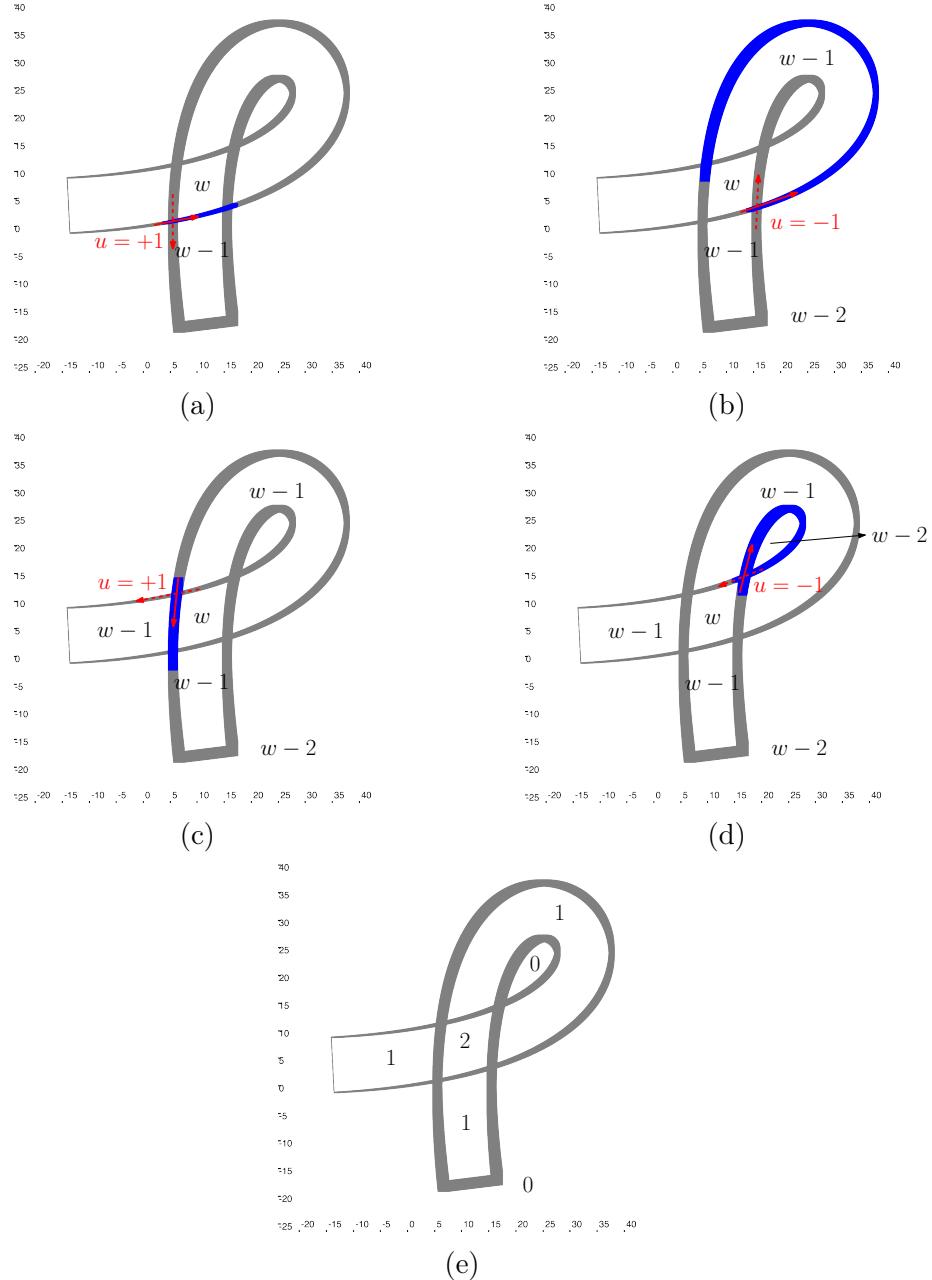


Figure 6.28: Numbering of the 2-cells surrounding each edge according to Alexander rules. We start with the numbering of the first edge represented in Figure (a) and then we propagate this value to all the consecutive edges considering their associated update value u . Figures from (a) to (d) illustrate this process on some edges, enough to have a complete numbering of all the 2-cells. Then, we know that the smallest value must be 0 and in this case, $w-2 = 0$. In Figure (e) we replace w by 2 to obtain the final numbering of 2-cells.

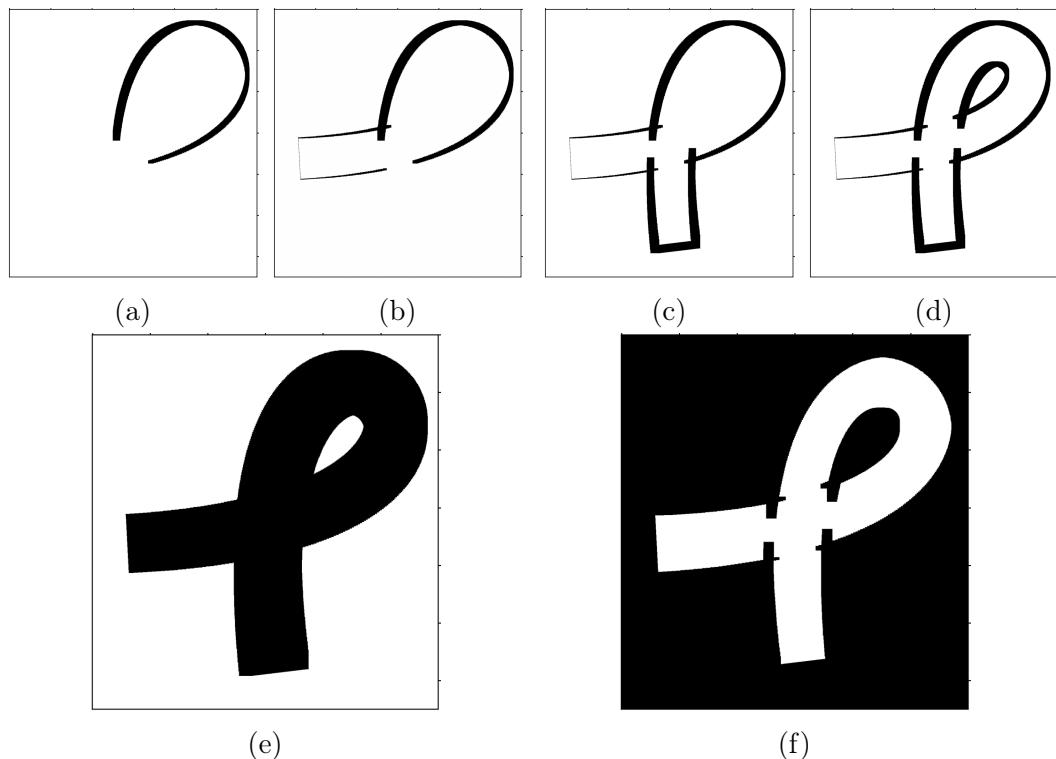


Figure 6.29: Process for creating images $\mathcal{B}_{\mathbb{S}^1}^{in}$ and $\mathcal{B}_{\mathbb{S}^1}^{out}$. In these figures, black pixels correspond to pixels with value +1 and white pixels are pixels with value set to 0. Figures from (a) to (d) illustrate the construction of $\mathcal{B}_{\mathbb{S}^1}^{in}$ through the drawing of edges that have the region on its left side numbered with +1. After filling the area surrounded by these edges we obtain in Figure (e) the final $\mathcal{B}_{\mathbb{S}^1}^{in}$. In Figure (f) we have $\mathcal{B}_{\mathbb{S}^1}^{out}$.

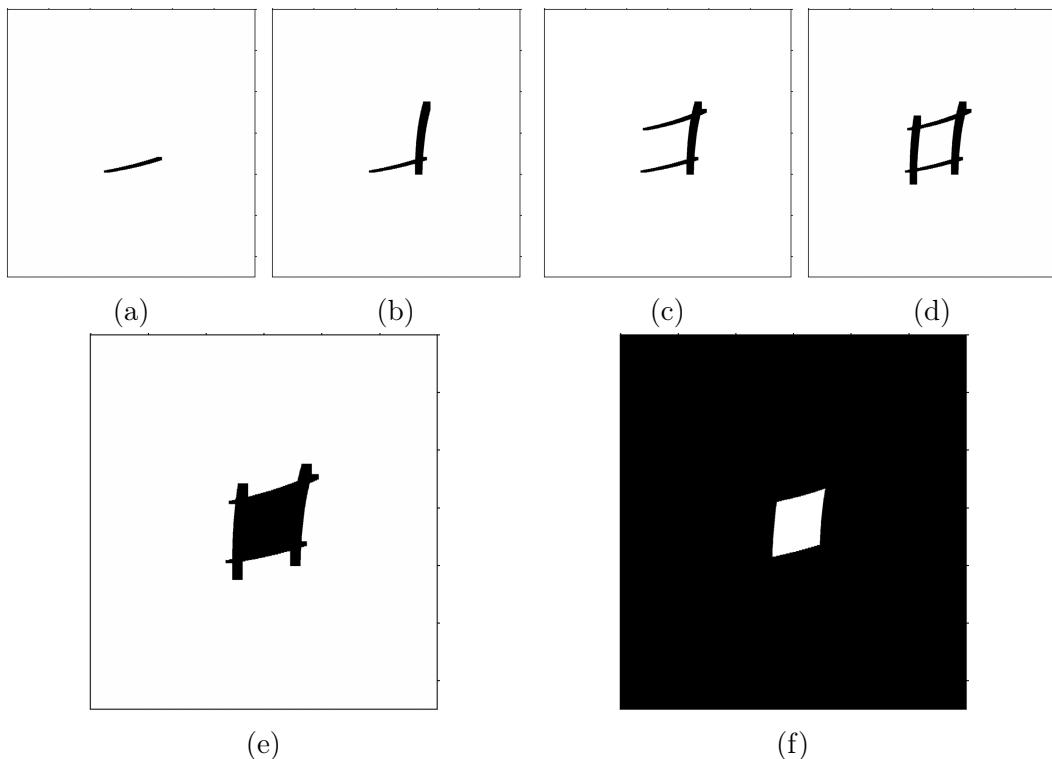


Figure 6.30: Process for creating images $\mathcal{B}_{S^2}^{in}$ and $\mathcal{B}_{S^2}^{out}$. In these figures, black pixels correspond to pixels with value +1 and white pixels are pixels with value set to 0. Figures from (a) to (d) illustrate the construction of $\mathcal{B}_{S^2}^{in}$ through the drawing of edges that have the region on its left side numbered with +2. After filling the area surrounded by these edges we obtain in Figure (e) the final $\mathcal{B}_{S^2}^{in}$. In Figure (f) we have $\mathcal{B}_{S^2}^{out}$.

edges numbered with i on their left side on the image. The result is illustrated in Figures 6.29f and 6.30f.

Each of these separators can then be used to create a paving of the environment using Algorithm 2, and the result for the considered example is presented on Figure 6.31.

Algorithm 5 *Area Characterization* (Alexander Method)

in: $S, \{\mathcal{S}\}, \epsilon$

```

1:  $\mathbb{Q} \leftarrow S$ 
2: if  $\mathbb{Q} \neq \emptyset$  then take a box  $[s] \in \mathbb{Q}$  and remove it from  $\mathbb{Q}$  else return
   endif

   ▷ Compute the uncertain coverage measure of box  $[s]$  using separators.

3:  $[c_m] \leftarrow [0, 0]$ 
4: for  $\mathcal{S}$  in  $\{\mathcal{S}\}$  do
5:    $[s_{in}], [s_{out}] \leftarrow \mathcal{S}([s])$ 
6:   if  $[s_{in}]$  is empty then
7:      $[c_m] \leftarrow [c_m] + [1, 1]$ 
8:   elseif  $[s_{out}]$  is not empty then
9:      $[c_m] \leftarrow [c_m] + [0, 1]$ 
10:  end for

11: if  $[c_m]$  is a singleton then
12:   ▷ The coverage measure of points in  $[s]$  can be uniquely determined.
13:   ▷ Draw box with its correspondent color.
14: elseif  $w([s]) < \epsilon$  or ( $[s] \in [\gamma]$ , for any  $[\gamma] \in \{[\gamma]\}$ , and  $c_m^+ - c_m^- = 1$ ) then
15:   ▷ The box is already smaller than the required precision  $\epsilon$  or its coverage
      measure value can not be refined through bisection.
16:   ▷ Draw box with its correspondent color.
17: else
18:   bisect  $[s]$  into  $[s]^{(1)}$  and  $[s]^{(2)}$ 
19:    $\mathbb{Q} \leftarrow \mathbb{Q} \cup \{[s]^{(1)}, [s]^{(2)}\}$ 
   endif

20: Return to line 2.

```

Using the approach just described, we create separators for the non-empty winding sets of $[\gamma^+]$ and for each non-empty winding set for all $[\gamma^-]$ in $\{[\gamma^-]\}$. At the end we have a list of separators $\{\mathcal{S}\}$ that can be used to do a charac-

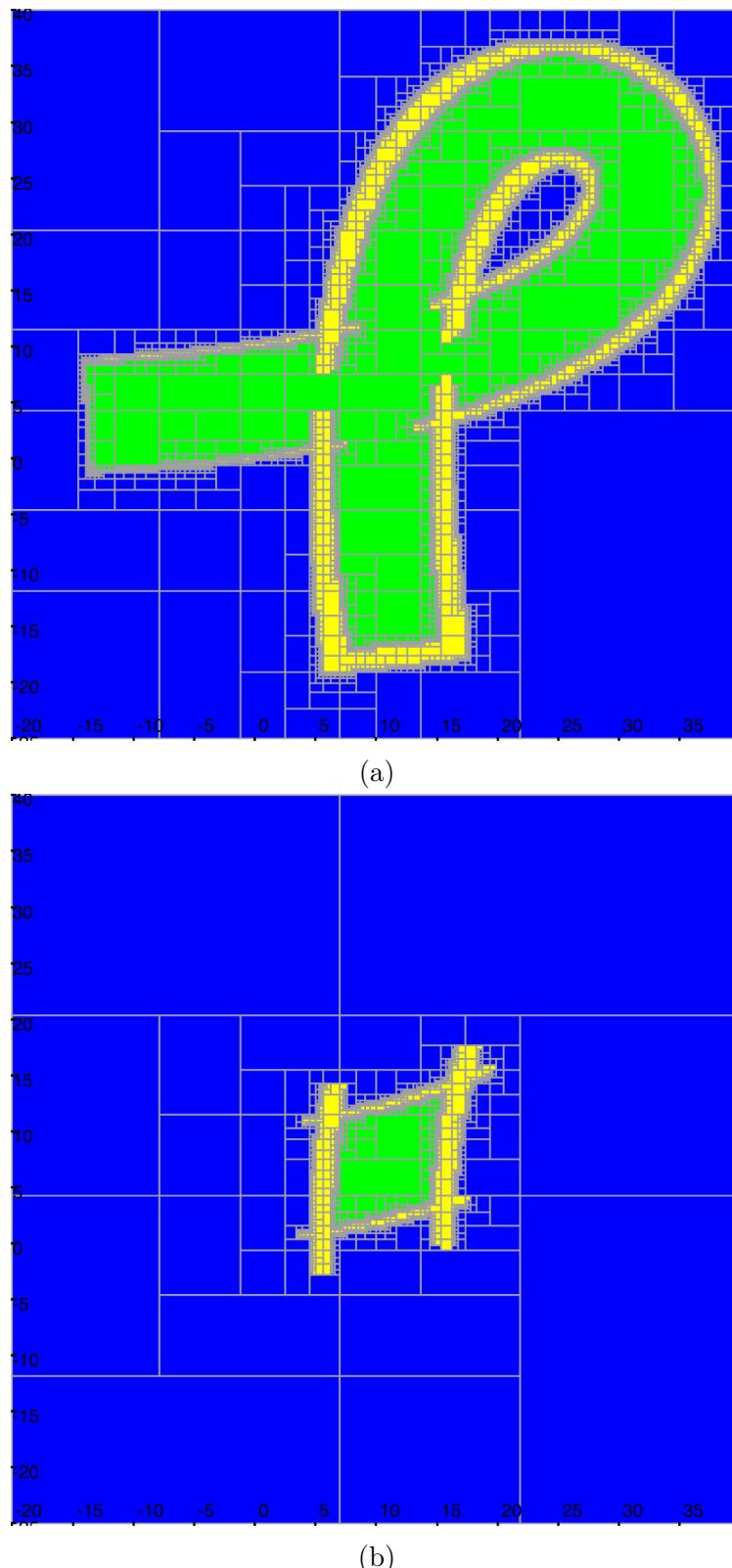


Figure 6.31: Paving of the environment using separator S^1 on Figure (a) and S^2 on Figure (b). Green boxes correspond to the inner approximation to the solution set. The union of green boxes with yellow boxes offer an outer-approximation to the set of points with winding number equal or greater than 1 in (a) and equal or greater than 2 in (b). Blue boxes correspond to points that are not in the solution set.

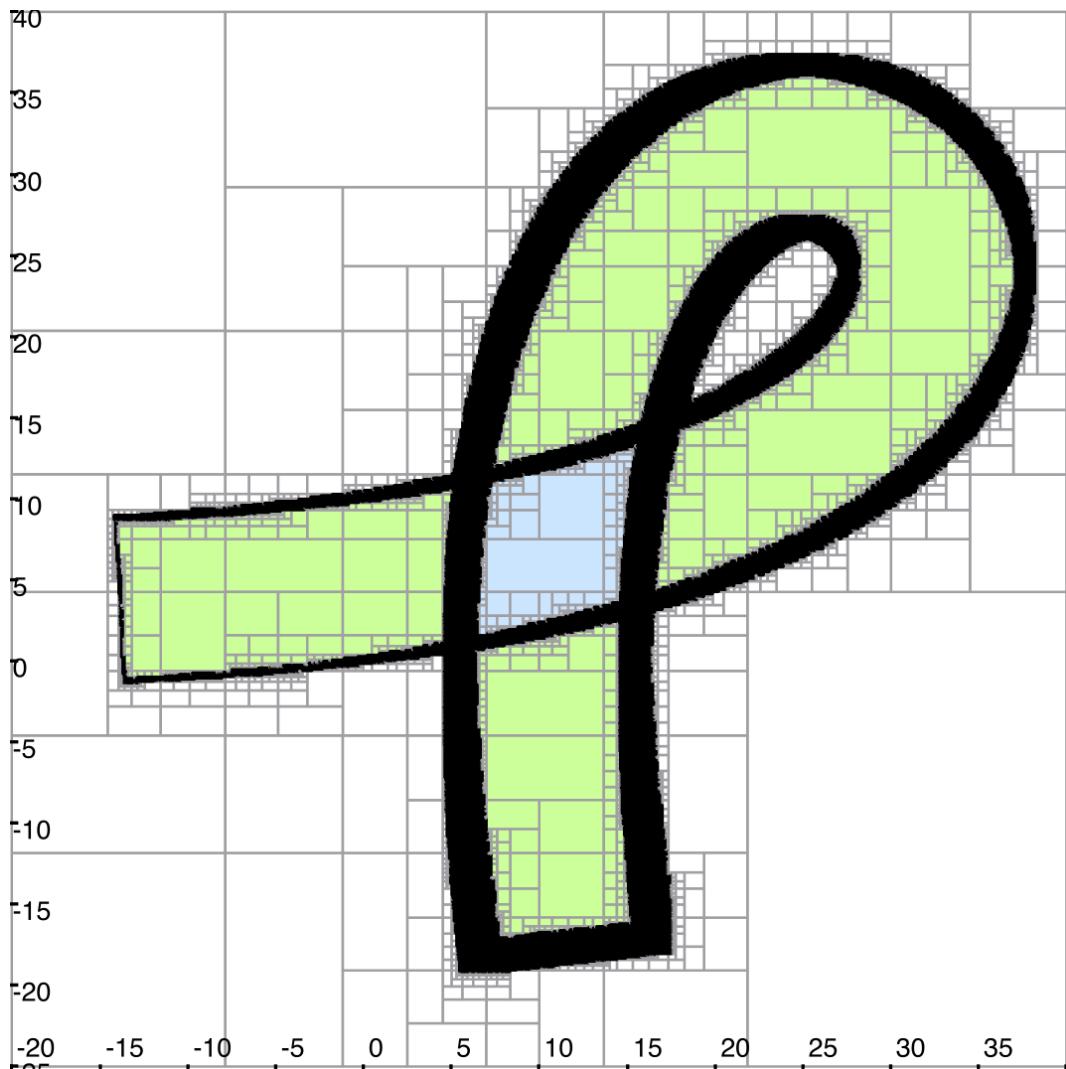


Figure 6.32: Resultant paving from Algorithm 5 using $\{\mathcal{S}\} = \{\mathcal{S}^1, \mathcal{S}^2\}$. Points inside blue boxes have $[c_m] = [2, 2]$, in green boxes $[c_m] = [1, 1]$, in white boxes $[c_m] = [0, 0]$, and in black boxes $\bar{\eta}^- < \bar{\eta}^+$.

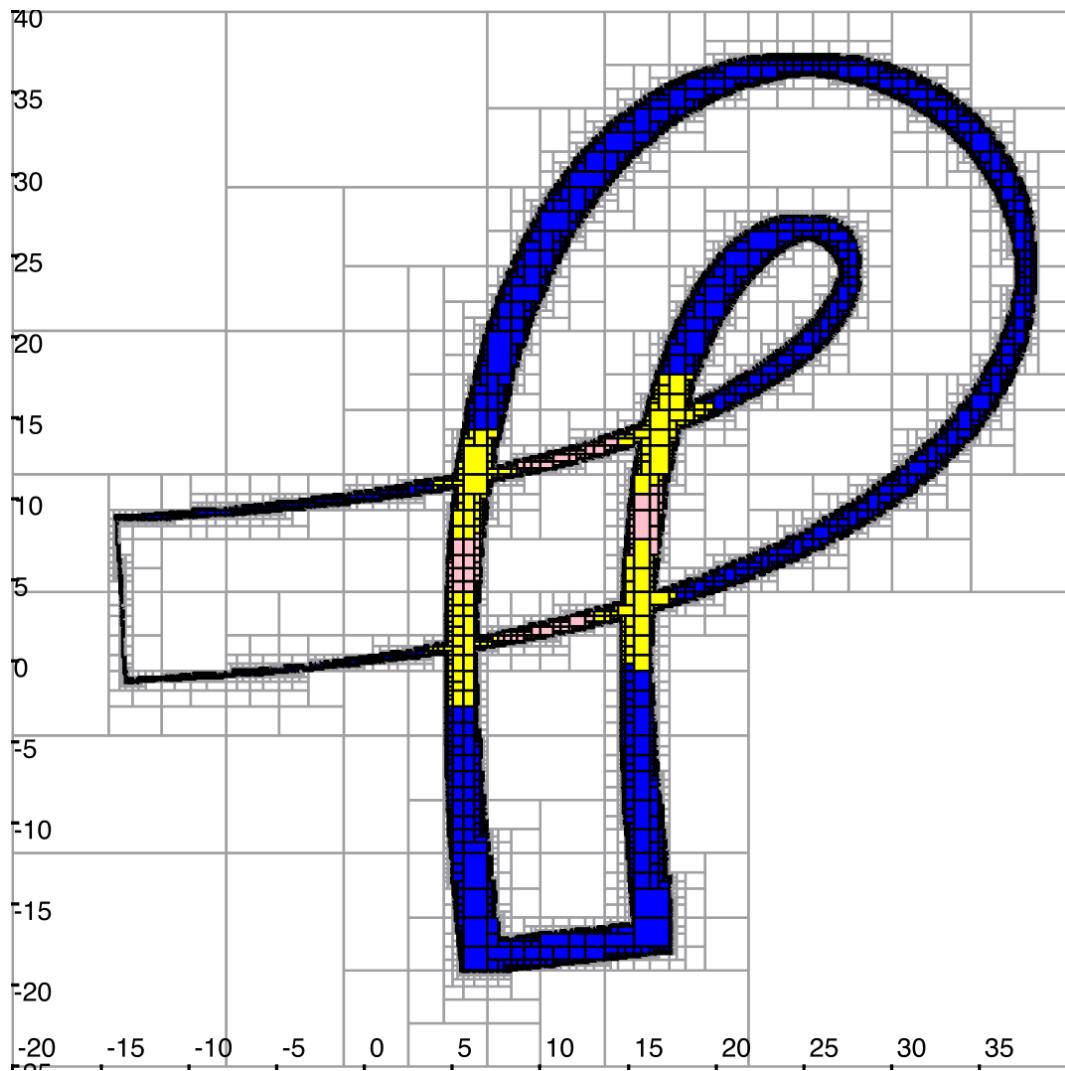


Figure 6.33: Resultant paving from Algorithm 5 using $\{\mathcal{S}\} = \{\mathcal{S}^1, \mathcal{S}^2\}$. Points inside blue boxes have $[c_m] = [0, 1]$, in yellow boxes $[c_m] = [0, 2]$, and in pink boxes $[c_m] = [1, 2]$.

terization of the environment in terms of the coverage measure of its points as presented in Algorithm 5. The result for the example considered here is illustrated in Figures 6.32 and 6.33. In practice, the pessimism from last steps is propagated, and the resultant characterization is guaranteed but not optimal.

The result can be refined by creating binary images with more pixels that can provide a more accurate representation of the environment. Additionally, decreasing the ϵ value in Algorithms 5 and 3 can contribute to this refinement. However, it is important to note that augmenting precision also increases computational complexity, so these adjustments should be tailored to suit the specific requirements of the application.

6.3.3 Method 3: Jordan Curves Sum

In the context of this thesis, the last approach that we introduce for characterizing areas based on the coverage measure dispenses the need for imposing robust assumptions concerning the self-intersections of the mission contour γ . Nevertheless, like the previous approach, it continues to utilize the consistency of winding numbers within connected 2-cells of the CW complex $C(\gamma)$.

In order to apply this approach we are going to assume that the robot's trajectory \boldsymbol{x} is not uncertain and that all the points are swept forward during the mission.

The approach is based on the idea of dividing the mission contour γ into tubes representing Jordan Curves. A Jordan curve is a continuous, closed curve in the plane that does not cross itself except at its endpoints. One of the important properties of a Jordan curve is that it divides the plane into two connected regions: an interior region and an exterior region. The interior region of a Jordan curve consists of all the points in the plane that are enclosed by the curve. The exterior region, on the other hand, consists of all the points in the plane that are outside the area enclosed by the curve. The exterior region can also be seen as the unbounded connected region in $\mathbb{R}^2 \setminus \gamma$, if γ is a Jordan curve. This property is proved by the Jordan Curve Theorem, which is a fundamental result in topology [137].

Moreover, another property that is important for our context is that the winding number of a Jordan curve around the plane can only be $+1$, 0 , or -1 . The interior region is numbered with $+1$ or -1 depending on the orientation of the curve, and the exterior is always numbered with 0 . See Figure 6.34.

Considering our exploration context, a Jordan curve γ , representing the mission contour, can be obtained when the robot moves with small variations in its orientation throughout the mission. Figure 6.35 provides an example. In this illustration, all points inside the interior region \mathbb{I} have a winding number, and a coverage measure, of $+1$, and all points in \mathbb{E} were not explored, having

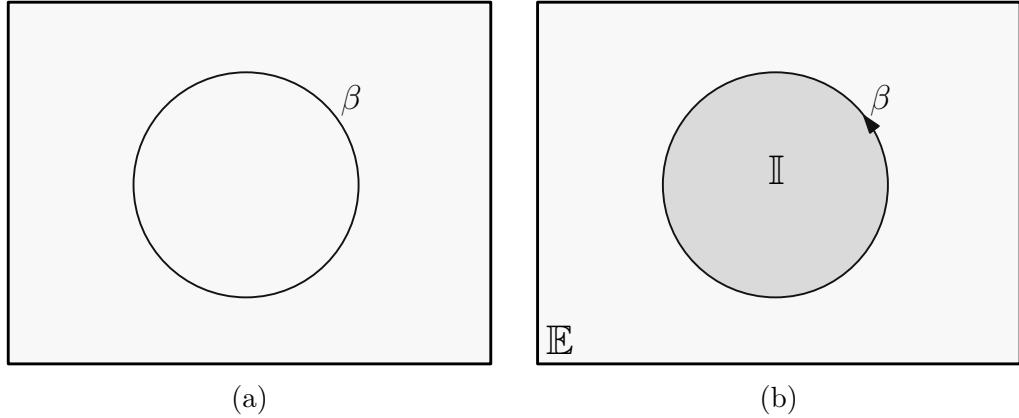


Figure 6.34: In these figures $\beta : S^1 \rightarrow \mathbb{R}^2$ is a Jordan Curve that divides the plane into an interior \mathbb{I} and an exterior \mathbb{E} . If the curve is counterclockwise oriented, as illustrated in Figure (b), the winding number of points in the interior region is +1.

a coverage measure of 0.

Hence, this approach consists of breaking down the exploration mission into submissions, ensuring that each submission has an associated contour that is a Jordan Curve. For that, if $\mathbf{x} : [0, T] \rightarrow \mathbb{R}^2$ is the original trajectory of the robot, we divide it into $k \in \mathbb{N}$ sub trajectories $\mathbf{x}_1, \dots, \mathbf{x}_k$ that are a restriction of the original time domain $[0, T]$ to a subset $[t_i^-, t_i^+] \subseteq [0, T]$ such that $\mathbf{x}_i = \mathbf{x}|_{[t_i^-, t_i^+]}$, for $i \in \{1, \dots, k\}$.

We start with $t_1^- = 0$, for the first trajectory, and the criterion for division is $\dot{\mathbf{x}}(t_i^-) \cdot \dot{\mathbf{x}}(t_i^+) = 0$. For example, take Figure 6.36. The original cycle γ associated to \mathbf{x} is not a Jordan curve. Then, the figure illustrates a possible division of the trajectory to create $\gamma_1, \gamma_2, \gamma_3$ and γ_4 , that are Jordan curves. This division is not optimal, we could have the original γ divided into two Jordan curves instead of four, for example. We chose, however, the presented division criterion, so we will not have to verify if the sub mission contours γ_i , with $i = 1, \dots, n$, self-intersects, reducing computational complexity.

We also define the juncture sets J_i , with $i \in \{1, \dots, k-1\}$:

$$J_i = \mathbb{V}([t_i^-, t_i^+] \cap [t_{i+1}^-, t_{i+1}^+]) \quad (6.29)$$

Where \mathbb{V} is the visible area set function. The juncture sets associated to the previous example are illustrated in Figure 6.37.

From the set of simple cycles $\{\gamma_i\}$, $i \in \{1, \dots, k\}$ and juncture sets $\{J_i\}$, $i \in \{1, \dots, k-1\}$, we can compute the extended winding number of the original

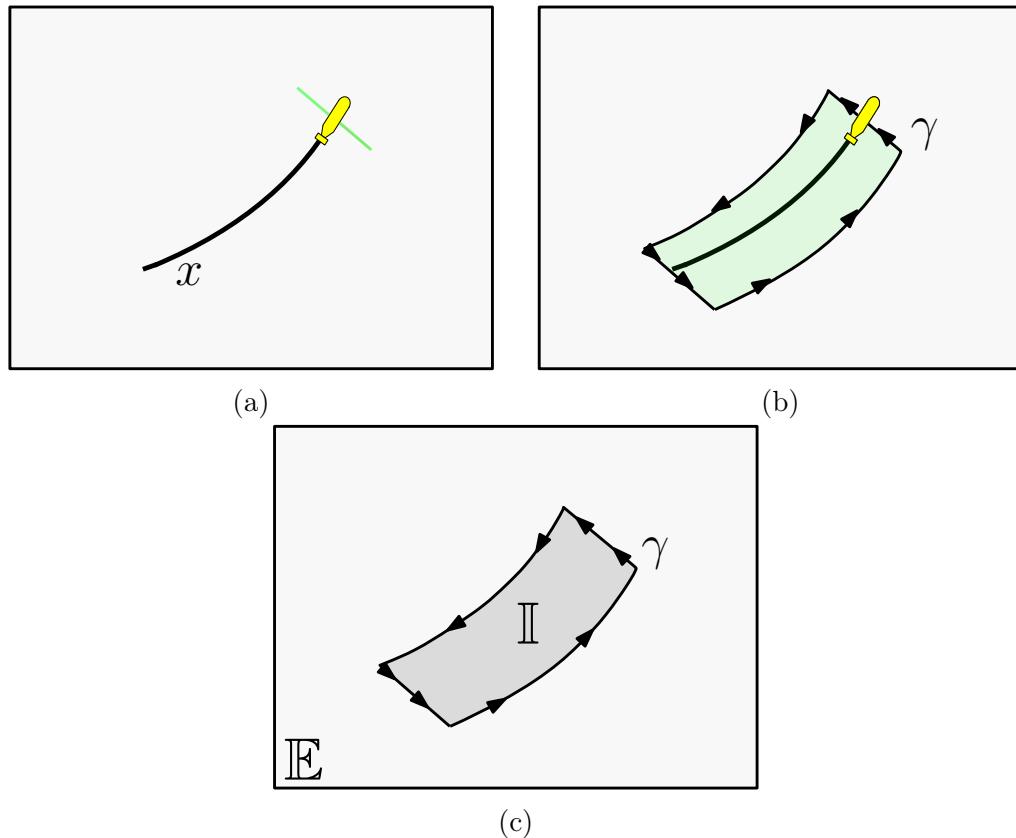


Figure 6.35: We assume a robot exploring with two one-dimensional sensors, one on each of its sides. If during a mission, the trajectory of the robot is represented by x , as illustrated in (a), we obtain the mission contour γ in Figure (b), that is a Jordan curve. This curve divides the plane into an interior and an exterior region, Figure (c).

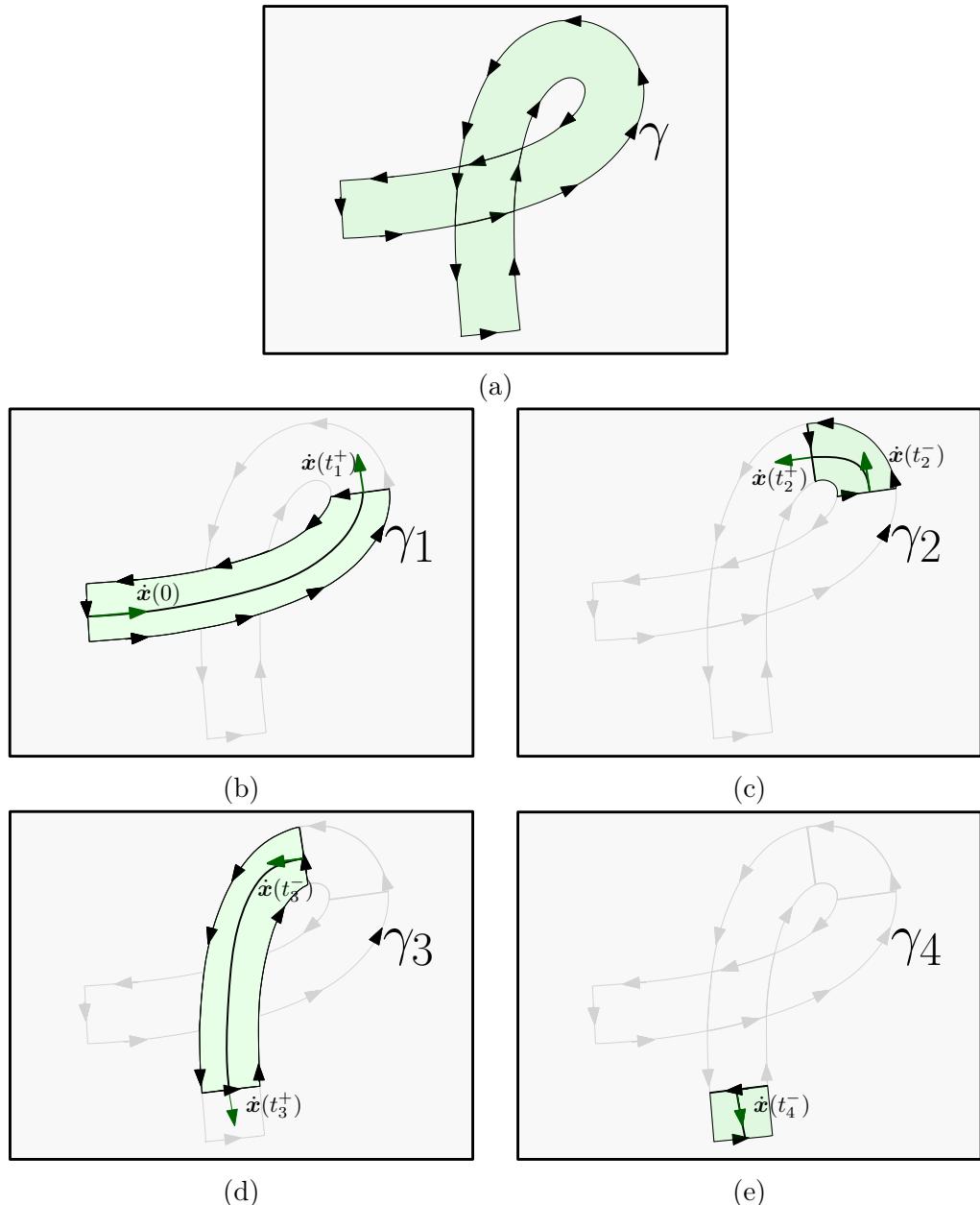


Figure 6.36: Division of cycle γ illustrated in (a) into Jordan cycles; (b): γ_1 is the mission contour associated to the trajectory $\dot{x}_{[0,t_1^+]}$; (c): γ_2 is the mission contour associated to the trajectory $\dot{x}_{[t_2^-,t_2^+]}$; (d): γ_3 is the mission contour associated to the trajectory $\dot{x}_{[t_3^-,t_3^+]}$ and (d): γ_4 is the mission contour associated to the trajectory $\dot{x}_{[t_4^-,T]}$.

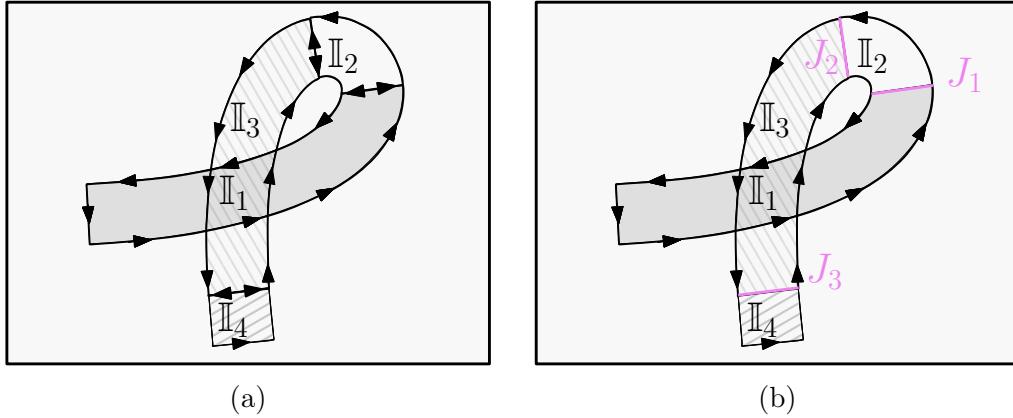


Figure 6.37: (a): The original cycle is divided into five Jordan curves, they are: $\gamma_1, \gamma_2, \gamma_3$ and γ_4 ; (b): For this division we have junction set $J_1 = \mathbb{V}(t_1^+) = \mathbb{V}(t_2^-)$, $J_2 = \mathbb{V}(t_2^+) = \mathbb{V}(t_3^-)$ and $J_3 = \mathbb{V}(t_3^+) = \mathbb{V}(t_4^-)$.

loop γ with respect to any point $\mathbf{p} \in \mathbb{R}^2$:

$$\bar{\eta}(\gamma, \mathbf{b}) = \sum_{i=1}^k \chi_{\bar{I}_i} - \sum_{i=1}^{k-1} \chi_{\bar{J}_i} \quad (6.30)$$

where \bar{I}_i is the closure of the interior of the Jordan curve γ_i . Figure 6.38 provides some examples of calculation for some points.

In practice, we can construct a separator for the closure of the interior of each Jordan curve and a separator for each juncture set, using the technique presented last section with image contractors. And then, a characterization of the environment can be obtained with an algorithm similar to Algorithm 5.

6.4 Waterfall Characterization

Once the area of interest S is fully characterized in terms of the coverage measure of its points, we can use a set inversion algorithm, as presented in Algorithm 1, to characterize the waterfall space associated to the mission.

For illustrating this idea we are going to assume that the trajectory of the robot is known, without uncertainty. In this case, the set $\mathbb{S}_{\mathbf{x}}$ of possible trajectories is a singleton. We illustrate the robot's trajectory and the mission contour γ that we consider in this Section in Figure 6.39.

Figure 6.40 presents the waterfall characterization as a result of Algorithm 6, that is an adaptation of Algorithm 1.

One can see that there are many uncertain boxes in the waterfall display, mostly around its borders. This uncertainty is a result of the propagation of

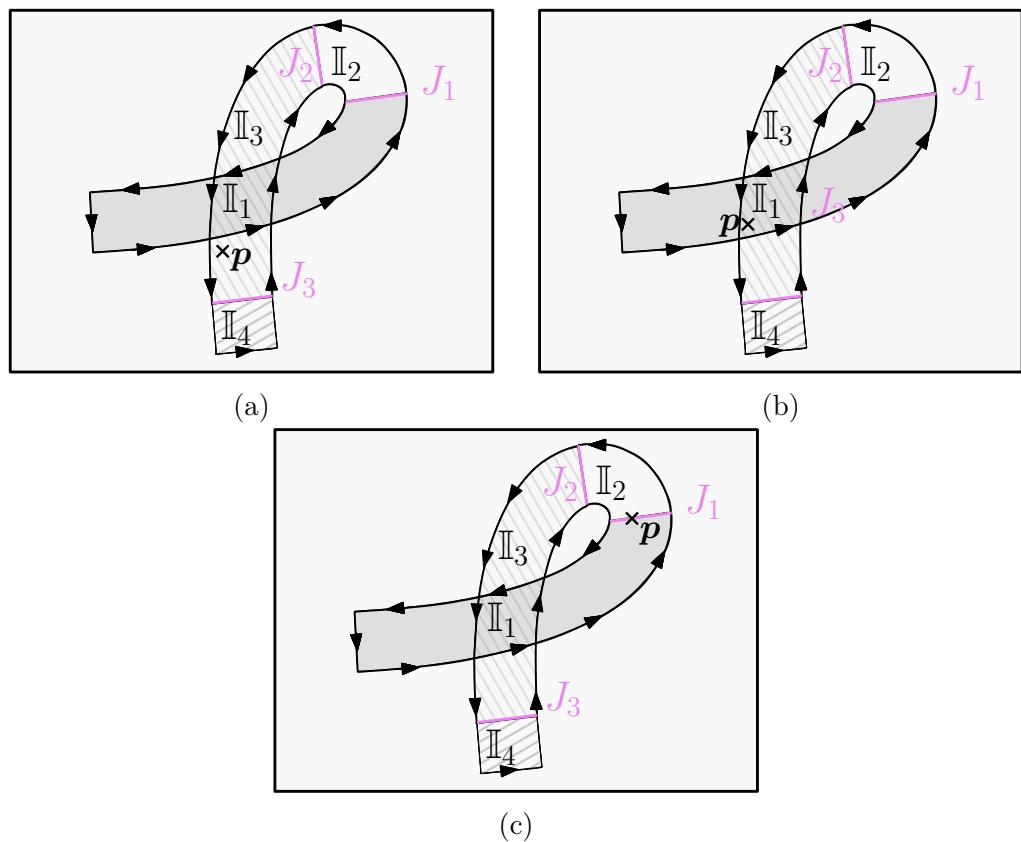


Figure 6.38: (a): Point p , as illustrated, belongs to the interior of curve γ_3 and $\bar{\eta}(\gamma, p) = 1$; (b): Point p belongs to the interior of curves γ_1 and γ_3 , therefore $\bar{\eta}(\gamma, p) = 2$; (c): Point p belongs to the closure of the interior of curves γ_1 and γ_2 but also to J_1 , then $\bar{\eta}(\gamma, p) = 1 + 1 - 1 = 1$.

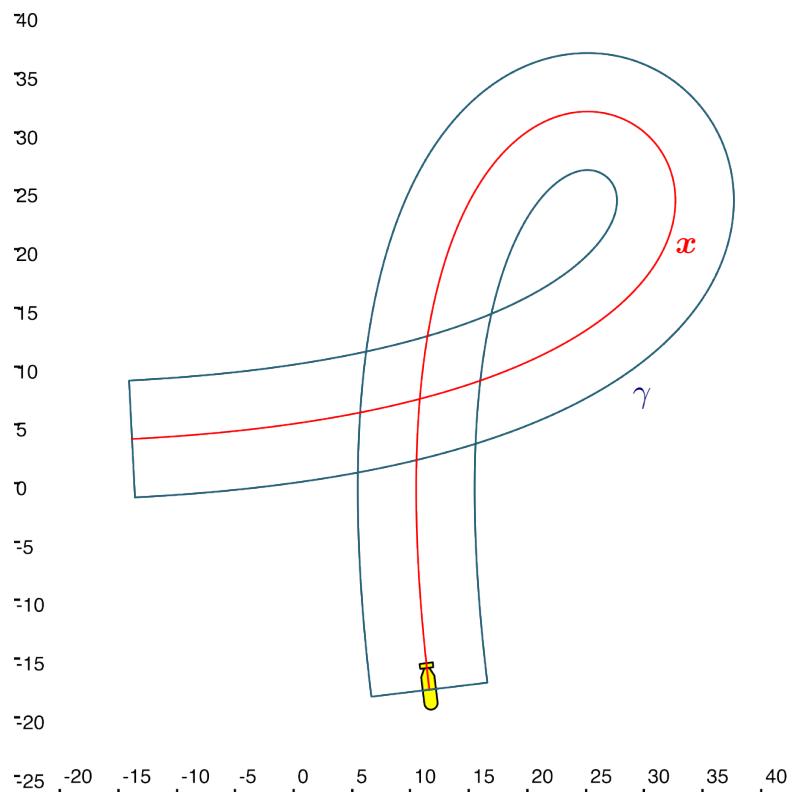


Figure 6.39: The robot is illustrated at final instant T , the robot's trajectory x is represented in red and in blue we have the associated mission contour γ .

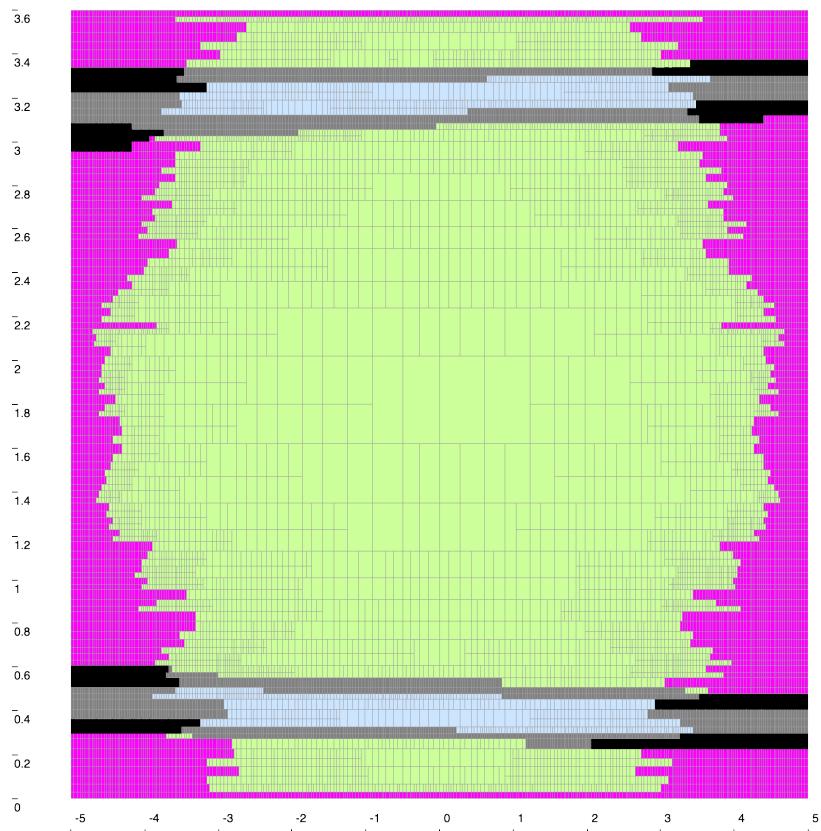


Figure 6.40: Characterization of the waterfall space $W = [-5, 5] \times [0, 1.8]$. Green boxes contain points with coverage measure equal to 1, blue boxes with coverage measure equal to 2 and the others can not be uniquely determined. Pink boxes contain points with a coverage measure that can be represented by $[0, 1]$, gray boxes $[1, 2]$ and black boxes $[0, 2]$.

Algorithm 6 *Waterfall Characterization*in: $W = [-L, L] \times [0, T], f, \epsilon$

- 1: $\mathbb{Q} \leftarrow W$
- 2: **if** $\mathbb{Q} \neq \emptyset$ **then** take a box $[w] \in \mathbb{Q}$ and remove it from \mathbb{Q} **else return**
endif
- 3: $[m] \leftarrow [f](w)$
- 4: \triangleright Compute $[c_m]$, the uncertain coverage measure of box $[m]$ using one of
the three methods proposed in this chapter.
- 5: **if** $[c_m]$ is a singleton **then**
- 6: \triangleright The coverage measure of points in $[w]$ can be uniquely determined.
- 7: \triangleright Draw box $[w]$ with its correspondent color on the waterfall.
- 8: **elseif** $w([w]) < \epsilon$ or $([m] \in [\gamma], \text{ for any } [\gamma] \in \{[\gamma]\}, \text{ and } c_m^+ - c_m^- = 1)$
then
- 9: \triangleright The box is already smaller than the required precision ϵ or its coverage
measure value can not be refined through bisection.
- 10: \triangleright Draw box $[w]$ with its correspondent color on the waterfall.
- 11: **else**
- 12: bisect $[w]$ into $[w]^{(1)}$ and $[w]^{(2)}$
- 13: $\mathbb{Q} \leftarrow \mathbb{Q} \cup \{[w]^{(1)}, [w]^{(2)}\}$
- 14: **endif**
- 15: Return to line 2.

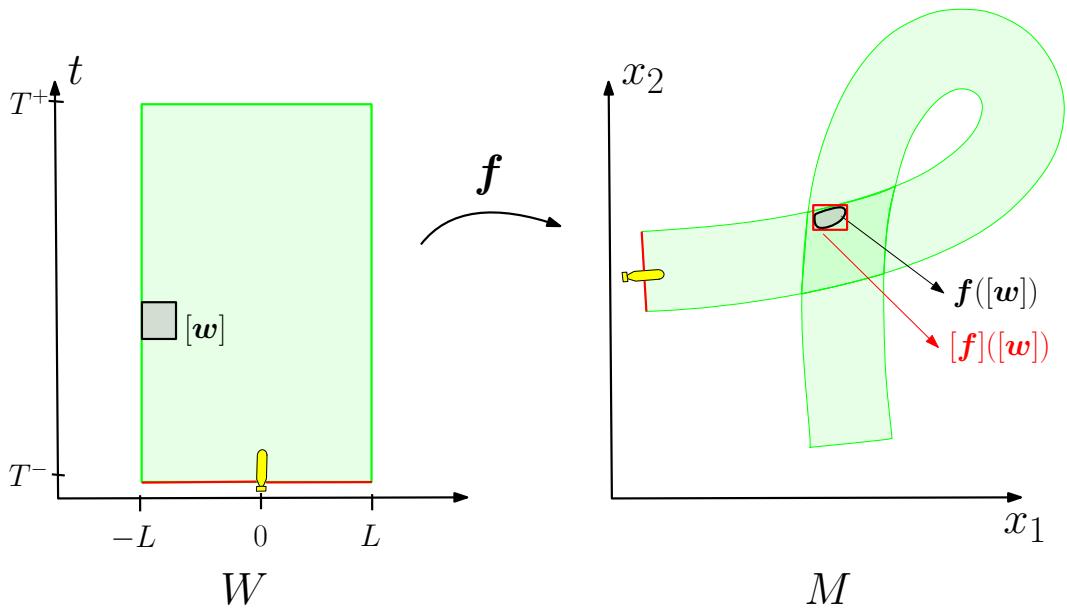


Figure 6.41: Take a box $[\mathbf{w}] \in W$, the image of this box on the mosaic space by the sweep function \mathbf{f} is a set that can assume any form. Numerically, we work with an interval representation of this set, that is $[\mathbf{f}]([\mathbf{w}])$, that adds some pessimism in order to obtain a guaranteed approximation. Therefore, when $[\mathbf{w}]$ is close to a border of W , there is a high possibility that its interval image $[\mathbf{f}]([\mathbf{w}])$ will intersect $\mathbf{f}(\partial W)$. As a consequence this adds uncertainty to the coverage measure estimation. Notably, for the example on the Figure, we have $[c_m](\mathbf{f}([\mathbf{w}])) = [2, 2]$ and $[c_m]([\mathbf{f}]([\mathbf{w}])) = [1, 2]$.

pessimism by the sweep function $[f]$, as illustrated in Figure 6.41. Some solutions could be tailored for reducing this uncertainty on borders. For example, we know that all points in the Waterfall space were explored by the robot at least once, therefore, pink boxes in Figure 6.40, representing a coverage measure of $[0, 1]$, should be replaced by green boxes with coverage measure of $[1, 1]$. Other solutions should be explored on future work.

6.5 Conclusion

In conclusion, this chapter is focused on the challenge of estimating the coverage measure of points within a subset of the plane. Our aim was to develop an algorithm that could efficiently compute the coverage measure for various points, using the principles of interval analysis discussed in Chapter 3.

We began by outlining the process for numerically generating tubes $[\gamma^+]$ and $[\gamma^-]$, along with their corresponding derivatives $[\dot{\gamma}^+]$ and $[\dot{\gamma}^-]$. Then, we introduced three novel set-membership algorithms for determining the winding number of a tube concerning all the points within an area of interest.

Finally, we demonstrated that a characterization of the waterfall space in terms of the coverage measure of its points is also a possibility.

In the following chapter, the algorithms introduced in this chapter will be applied to address real-world data within the framework of two distinct robotics experiments.

Chapter 7

Experiments

Contents

7.1	Daurade	201
7.2	SpeBoat	206
7.3	Conclusion	220

Throughout this thesis, we conducted experiments to illustrate the methods studied during our research. In this chapter, we present real-world applications of the algorithms introduced in the previous chapter. For this purpose, we use datasets acquired by two distinct platforms: the autonomous underwater vehicle, Daurade, and the unmanned surface vehicle, Speboat. The details and results of these experiments are presented in the following sections.

7.1 Daurade



Figure 7.1: The AUV Daurade in the roadstead of Brest.

The autonomous underwater vehicle, known as Daurade and depicted in Figure 7.1, was designed and constructed by ECA Robotics. It has been employed in missions by the Direction Générale de l’Armement - Techniques Navales (DGA - TN) and the Service Hydrographique et Océanographique de la Marine (SHOM). This robot was developed with the objective of performing Rapid Environment Assessment (REA) missions, whose purpose is to conduct environmental surveys of specific locations to detect potential threats. More precisely, Daurade was conceptualized in the context of counter-mine warfare, and it maps the seafloor, using acoustic sensors, for the detection of potential underwater mines.

In November 2015, Daurade was deployed in the roadstead of Brest, located in Brittany, France, where it executed a 45-minute survey path as part of an REA mission. The data acquired during this mission constitutes a dataset of proprioceptive measurements, that was first used in [29]. The exteroceptive dataset, with data acquired by the sonars, is not publicly available.

Daurade employs two side-scan sonars (Klein 5500), with one scanning its right side and the other scanning its left side. The observable area of

each sensor can be individually represented as a line-sweep sensor in a two-dimensional plane. Assuming a configuration where there is neither a visibility gap nor an overlap between the ranges of visibility of the two sensors, as proposed in Section 2.2.3 of Chapter 2, the combination of both sensors can be conceptualized as a single line-sweep sensor.

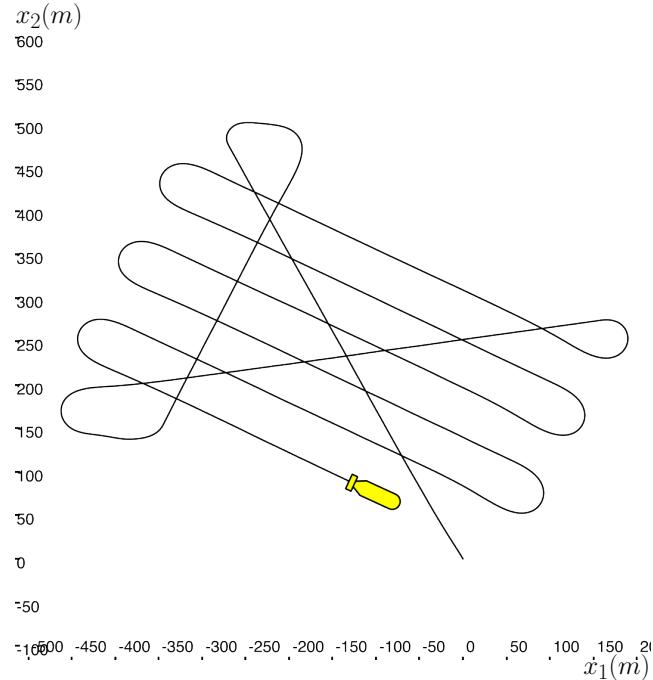


Figure 7.2: Trajectory $\tilde{\mathbf{x}}$ of the robot in the plane, without uncertainty, estimated through integration of proprioceptive data acquired during the mission. Daurade is represented by the AUV icon at its final pose.

The position and orientation of the robot underwater are determined through the integration of data collected by two key components: an Inertial Measurement Unit (IMU), which measures angular speed and linear accelerations, and a Doppler Velocity Logger (DVL), responsible for measuring the speed of the robot with respect to the seabed. Additionally, a pressure sensor can be incorporated to estimate depth, enabling a complete three-dimensional localization. Initially, we assume that the result of this estimation, that we name $\tilde{\mathbf{x}}$, is exact, as illustrated in Figure 7.2.

As illustrated in Figure 2.2, it is evident that the altitude h of the robot has a direct impact on the range of visibility, represented as L , of the side-scan sonar. Given the availability of bathymetry data for the roadstead of Brest and the inclusion of a depth sensor within Daurade, it was possible to estimate its altitude during the mission. Using the intrinsic information of the employed

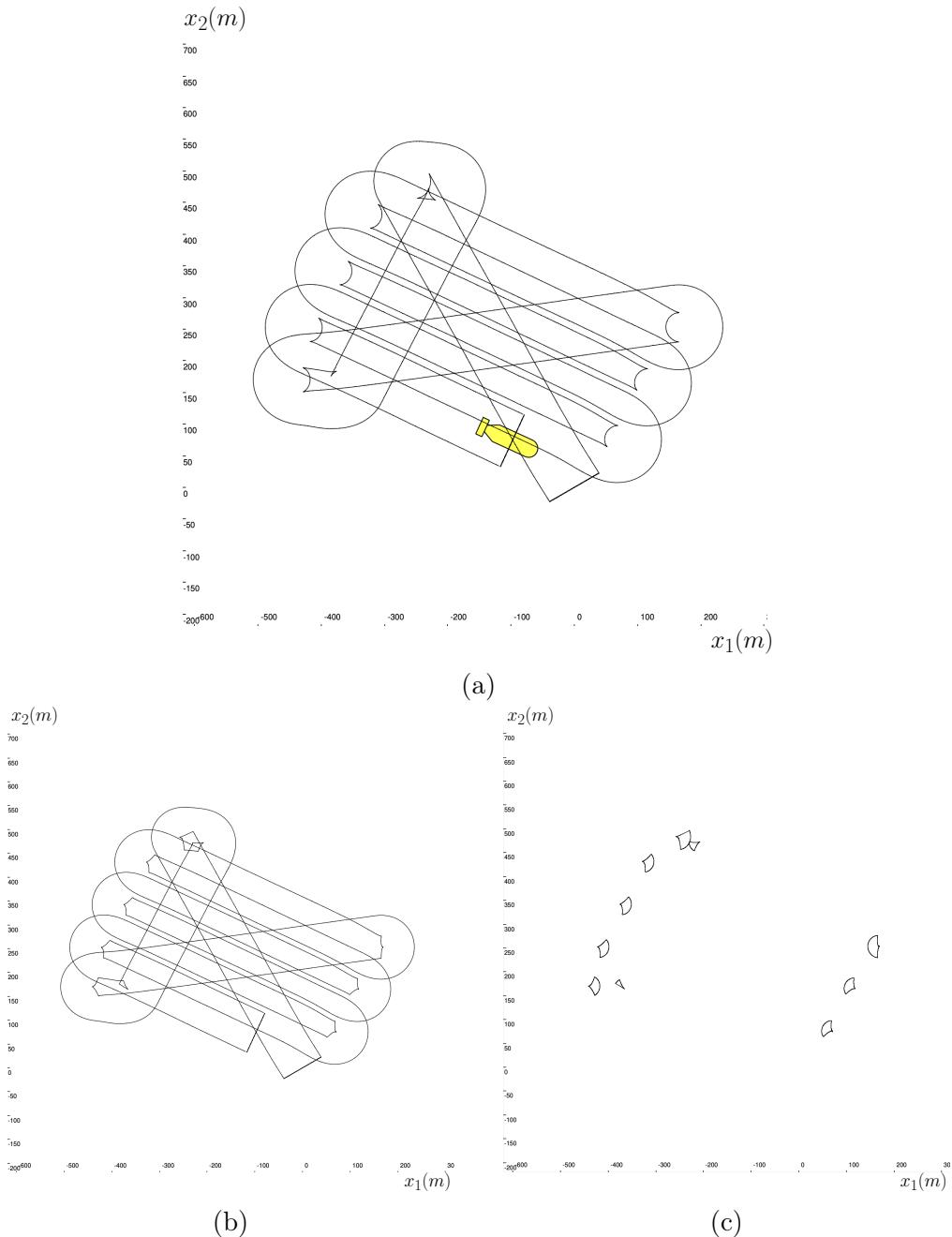


Figure 7.3: (a): The mission contour $\tilde{\gamma}$ associated to trajectory \tilde{x} . (b): $\tilde{\gamma}^+$ surrounding points swept forward; (b): For this mission we have 10 contours $\tilde{\gamma}^-$ surrounding points swept backwards. All the contours are counter-clockwise oriented.

sensors, we could conclude that L varied from 41 to 50 meters. In this initial analysis where we are not considering uncertainties, we make the assumption that L remains constant, with a fixed value of 45 meters on each side, resulting in the mission contour $\tilde{\gamma}$ illustrated in Figure 7.3. The division of the mission contour into $\tilde{\gamma}^+$ and $\tilde{\gamma}^-$ is also illustrated.

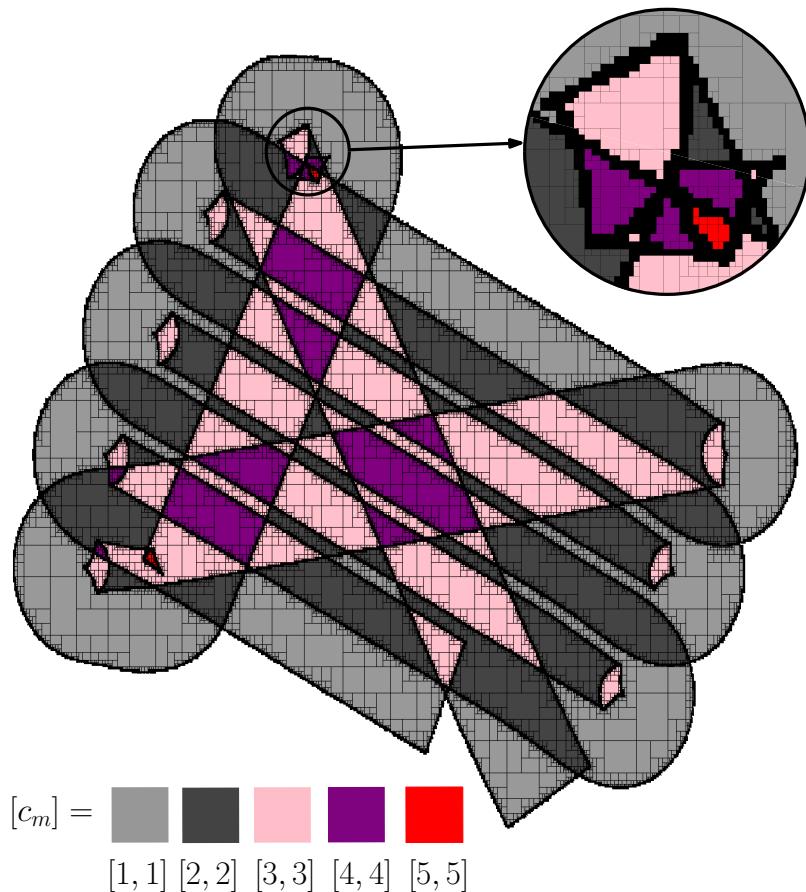


Figure 7.4: Result of the area characterization algorithm for the classification of the explored area. Boxes in black have an uncertain coverage measure value.

In the given scenario, the first method for area characterization, presented in Section 6.3.1 in the previous chapter, would be impractical due to the extensive area that needs to be characterized. Given that the area in question is large, we opted for the second method outlined in Section 6.3.2. Notably, this choice was driven by the fact that all self-intersections of $\tilde{\gamma}$ meet the conditions required by the method. After applying the area characterization algorithm based on the Alexander method, we obtain the paving illustrated in Figure 7.4. Uncertain boxes, surrounding contour $\tilde{\gamma}^+$ and all the contours $\tilde{\gamma}^-$ are

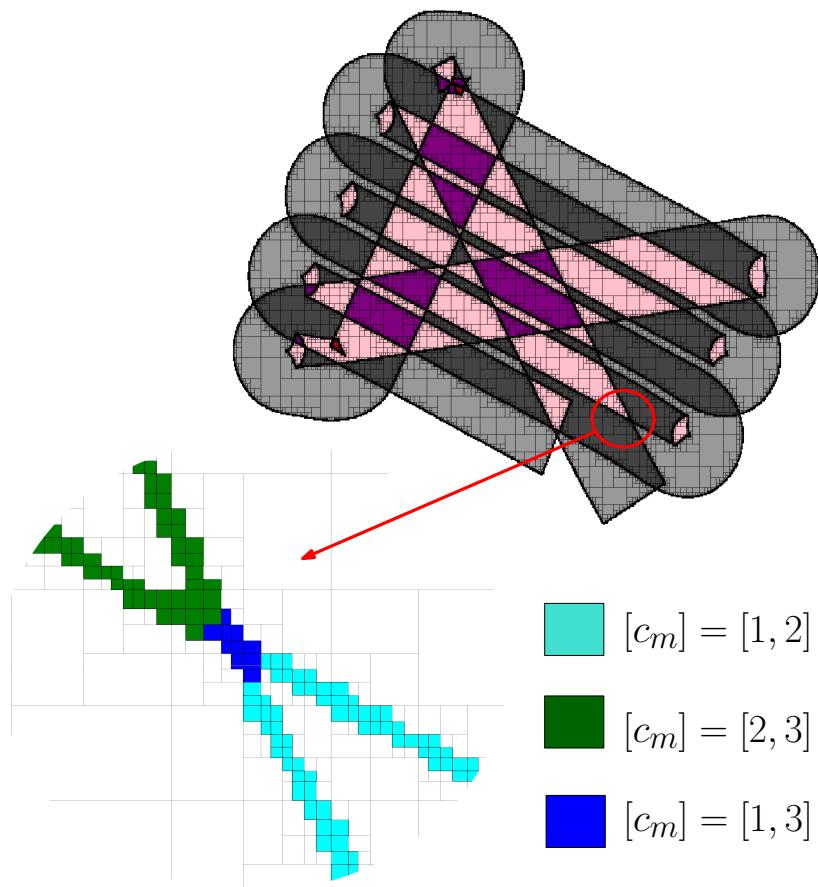


Figure 7.5: Example of classification of boxes with an uncertain coverage measure value.

represented in black. The uncertain winding number value for each of these black boxes can also be defined. In Figure 7.5, we give an overview of the classification of these boxes for a part of the mission.

Now, when we account for the uncertainties related to sensor measurements that are propagated during pose estimation, we obtain a tube denoted as $[x]$. This tube is a guaranteed representation of all the possible trajectories of Daurade during the mission, and as a consequence, the mission contour naturally becomes uncertain. If in addition we set the range L as an interval $L = [41, 50]$, we obtain the uncertain mission contour $[\gamma]$. These tubes are represented in Figure 7.6.

One can notice that certain self-intersections within $[\gamma]$ fail to meet the conditions stipulated by the Alexander method, particularly the non-collinearity condition, which is essential for dividing the environment into four distinct regions around the self-intersection. This division is necessary for the application of Alexander's rules for numbering. Consequently, the issue at hand cannot be directly resolved using the Alexander method as it was done for the case without uncertainty.

However, we have applied the Alexander method around a specific uncertain self-intersection within $[\gamma]$, which adheres to the non-collinearity assumption. This application serves to illustrate the extension of the Alexander algorithm to uncertain curves. The outcome of this application is depicted in Figure 7.7.

An extension of the Jordan curves method, outlined in Section 6.3.3 in the previous chapter, to deal with uncertainties and points swept backwards would be the most appropriate method for characterizing the environment explored by Daurade while accounting for uncertainties. This extension is a guide for future work.

Next section presents the robot SpeBoat, that was also used for exploration missions, but this time using a 2-dimensional sensor for covering the environment.

7.2 SpeBoat

The Speboat is an USV that was designed and built at ENSTA Bretagne as part of the research conducted by [138], and it was kindly loaned to us for conducting tests during the course of this doctoral thesis. This USV was tailored for operations in shallow waters, and it was primarily designed for seafloor mapping and surveillance missions. It offers the flexibility of being teleoperated over considerable distances or to operate autonomously.

To estimate its position, it relies on a satellite positioning system (GPS),

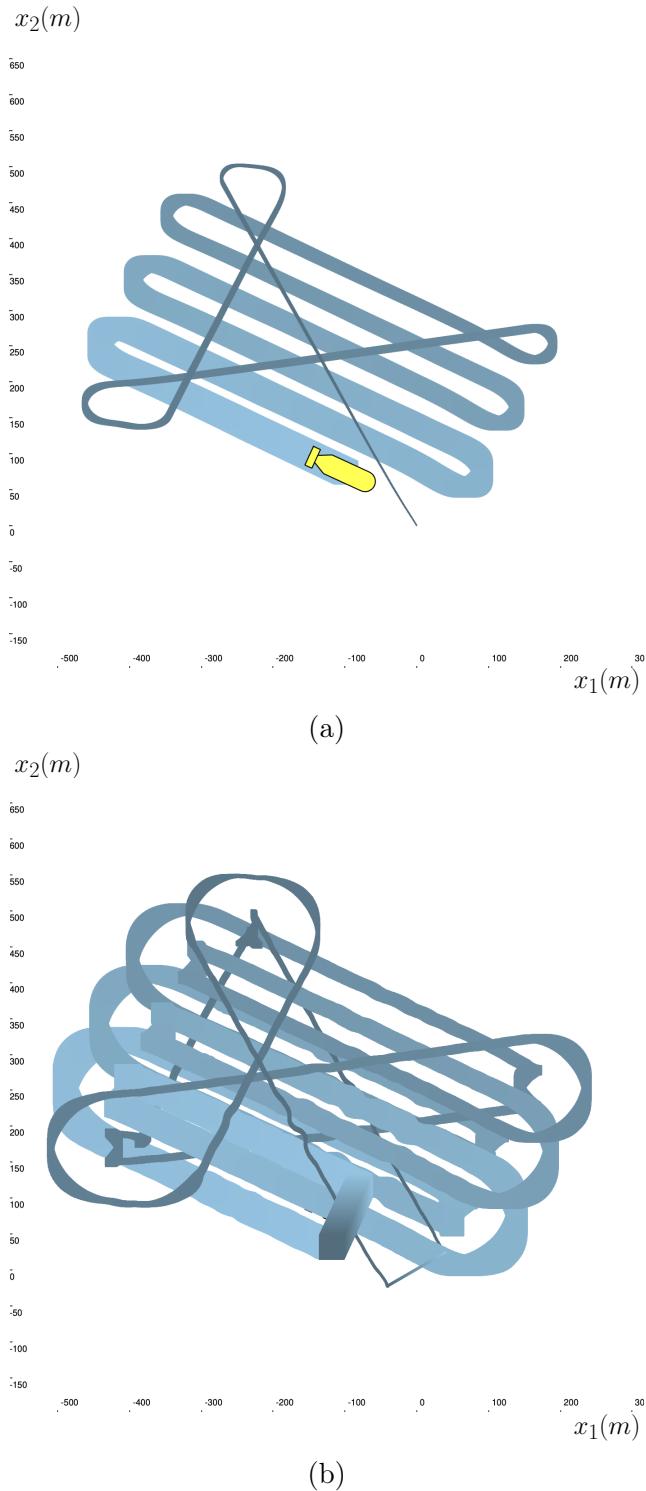


Figure 7.6: (a): The tube $[\mathbf{x}]$ envelopes all the possible trajectories of Daurade in the plane during the mission; (b): and tube $[\gamma]$ all the possible mission contours.

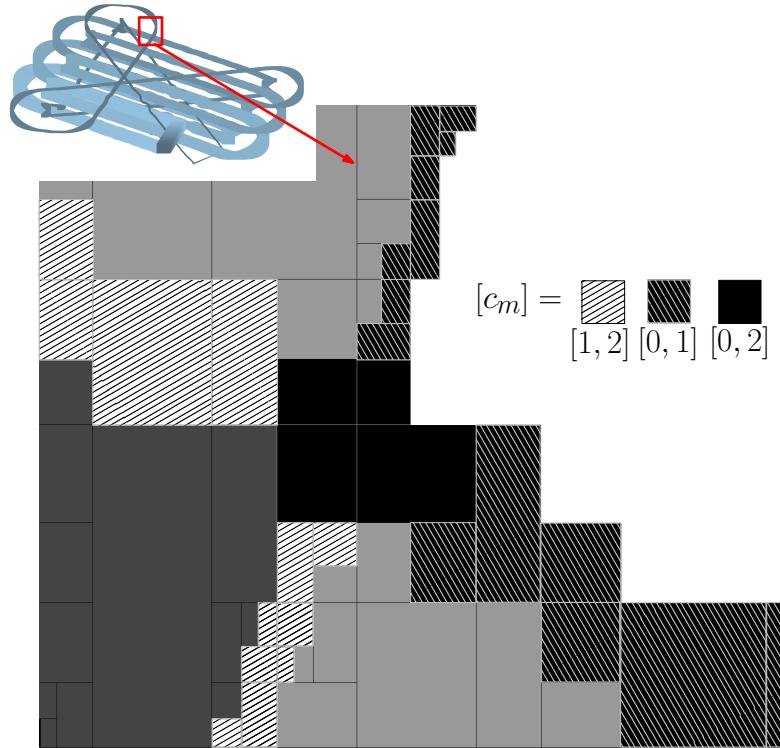


Figure 7.7: Application of the uncertain Alexander Rule for one self-intersection that respects the conditions established by the method. Boxes in light gray are guaranteed to have been explored once, and in dark gray twice. Other boxes have an uncertain coverage measure with values guaranteed to be within the intervals presented on the legend.



Figure 7.8: USV SpeBoat.

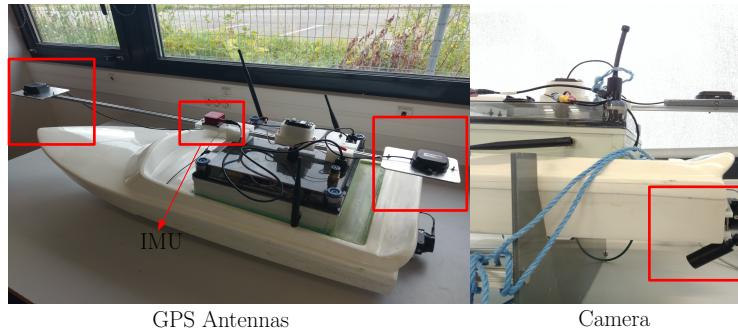


Figure 7.9: Speboat Sensors.

specifically the u-blox ZED-F9P chip paired with a dual-frequency antenna. The two antennas allow an estimation, in addition of the position, of the orientation of the drone. This data is further processed via a Kalman filter that incorporates gyroscope data from an embedded IMU. Environment exploration is done by an underwater camera looking downward placed on the bottom of its hull, as illustrates Figure 7.9.

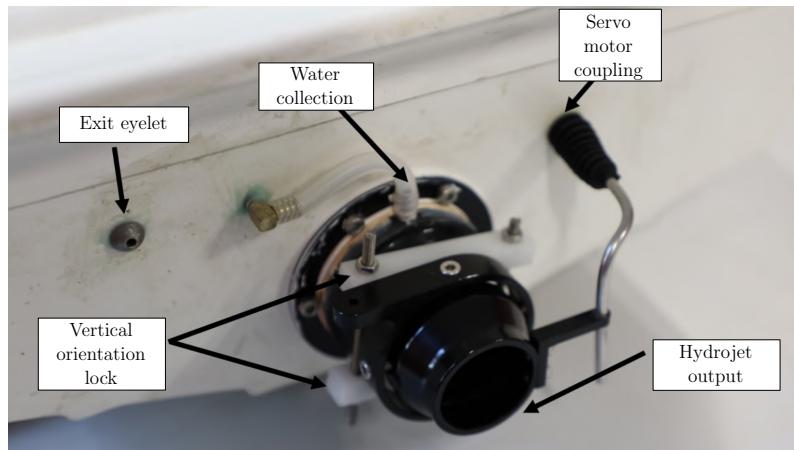


Figure 7.10: Hydrojet output on the rear of the robot. The tube for water collection retrieves water that circulates through a flexible watertight tube surrounding the internal electronics, serving the purpose of cooling the system. The water from the cooling system is ultimately expelled through the exit eyelet.

The vehicle is propelled by a hydrojet, that draws water from an open cavity under the hull and expels it on the rear of the robot, propelling it forward. The output of the hydrojet is visible in Figure 7.10, and it can be oriented to the right or to the left using a servo motor that is attached to it

from the interior of the robot.

In terms of its electronic architecture, the motor is controlled by a water-cooled Electronic Speed Control (ESC), and the onboard computer is an Intel NUC, used for autonomous computation of control commands. For example, the robot can autonomously follow lines and track waypoints, and the system has the Robot Operating System (ROS) as its middleware. An Arduino board handles the reception of either autonomous commands from the NUC or remote control signals, transmitting Pulse-Width Modulation (PWM) commands to the actuators.

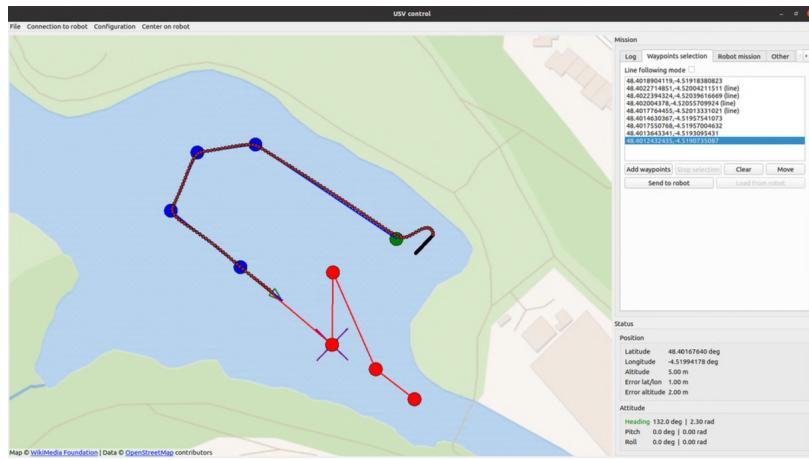


Figure 7.11: Human-Machine Interface of Speboat. The green point denotes the starting point. Visited locations are marked with blue points, while the next chosen waypoints for the mission are indicated by red points. The dotted black trajectory represent GPS measurements of the robot’s real position, with the current robot location indicated by the robot icon.

The trajectory of the robot can be chosen and displayed in real-time through the Human-Machine Interface (HMI) developed by [138], see Figure 7.11. This HMI was created using QT Creator and during this thesis it has been customized to accommodate additional functionalities, such as visualizing trajectories from rosbags and the explored area.

During this thesis, we did some exploration missions using this robot for exploring the Guerlédan lake, in Brittany, France. They are now presented.

Exploring the Guerlédan Lake

The Guerlédan lake is an artificial lake in center Brittany in France with a hydroelectric dam on one of its ends. It is responsible for producing every day the equivalent of the daily electricity consumption of 15000 people. Often,

mobile robots are required to navigate the lake [139] with the objective of identifying potential hazards underwater, such as loose tree trunks, that could affect the good functioning of the hydroelectric station.



Figure 7.12: Research team for the AUV Riptide at the Guerlédan lake. The SpeBoat was employed on its research by the group composed by three master students from ENSTA Bretagne. From left to right, Yohann Gourret, Katell Lagattu and Bernardo Hummes Flores.

In this context, I supervised a group of three master students in a project where we were inspired by a problem that frequently occurs during robotics trials, that is the lost of material. Indeed, during one of the tests that took place in the lake, the AUV Riptide, in orange in Figure 7.12, was lost underwater, and the SpeBoat was used as part of the research team.

Subsequently, after finding the lost Riptide, we reenacted the situation by replacing the AUV with a buoy of the same color, positioned approximately one meter below the surface, Figure 7.13. The objective was then to search for the buoy using algorithms for estimating the area already explored. This was crucial to identify areas that are not relevant for the search because they have already been analyzed.

The missions of exploration were autonomous and planned on the HMI taking into consideration an estimation of the area already explored, in order to avoid redundancy. The search would cease when the orange buoy was detected by the underwater camera, as presented in Figure 7.14.

In a similar context, Speboat was subsequently employed in an exploration mission aimed at locating two submerged storage LEGO bricks. These storage bricks, which represent potential underwater hazards for identification, are



(a)



(b)

Figure 7.13: (a): Positioning of orange buoys for reenacting the situation of a lost AUV by boat; (b): And Katell positioning them by swimming.



Figure 7.14: (a): Display of the area already explored on the HMI that would guide the planning of new missions when searching for the submerged buoy; (b): Detection of the orange buoy by the underwater camera.

distinguishable by their colors, one being red and the other yellow, and each measures 18x50x24 cm. The primary goal was to showcase the capability of mobile robots in conducting underwater target research. The algorithms for area characterization presented in this study were utilized to ascertain which parts of the environment had already been explored by the robot and how many times, thereby enhancing the capacity for well-informed decision-making.

Each LEGO brick was connected to a rigid bar that was submerged at a depth of one meter below the surface. This depth was achieved through a balance between anchors on the bottom and buoys on the surface. The scheme for the positioning of the bricks is represented on Figure 7.15, and the buoys stay over the surface.

The robot navigated autonomously following points previously selected on the available HMI. In Figure 7.16 we can see the trajectory of the robot x and the mission contour γ associated to the final mission. In this example, we ignore the uncertainty associated to the trajectory. For defining the mission contour we consider that the visible area at one meter depth can be represented by a closed disc of 2 meters radius centered on the position of the robot and oriented according to its heading.

Figure 7.17 illustrates the resultant characterization of the environment using the Alexander method.

On the video recorded by the camera during the mission, one of the bricks, the red brick, appears twice and the other, the yellow brick, appears four times. The number of times that each of the bricks appears on the video corresponds to the number of times the brick entered on the robot's visibility area and therefore, the coverage measure of its position on the plane. This information can be coupled with the knowledge of the coverage measure of the points of

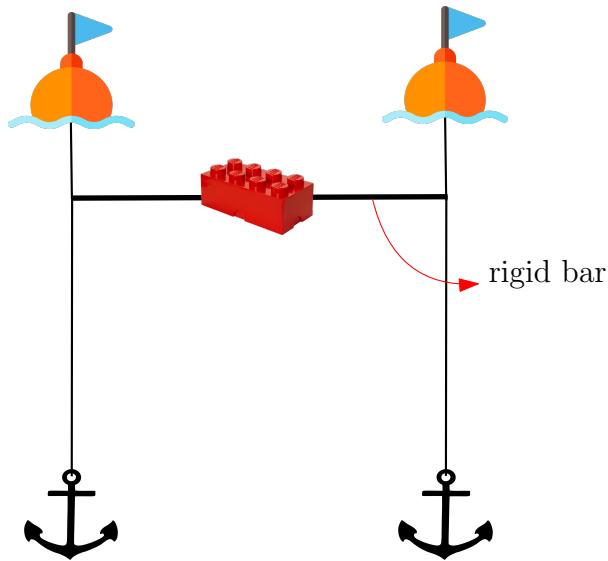


Figure 7.15: Representation of the structure adopted for the placement of the LEGO storage brick.

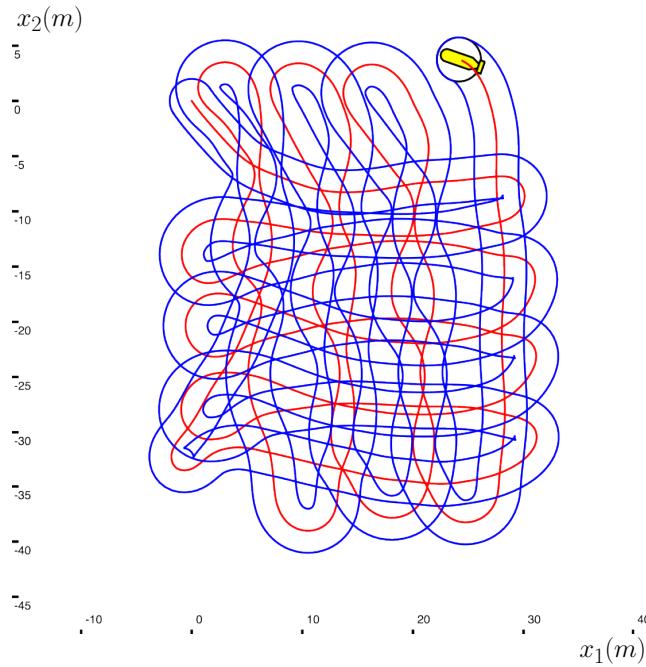


Figure 7.16: The robot is represented at its final pose at the end of the mission. In red its trajectory \boldsymbol{x} and in blue the mission contour γ .

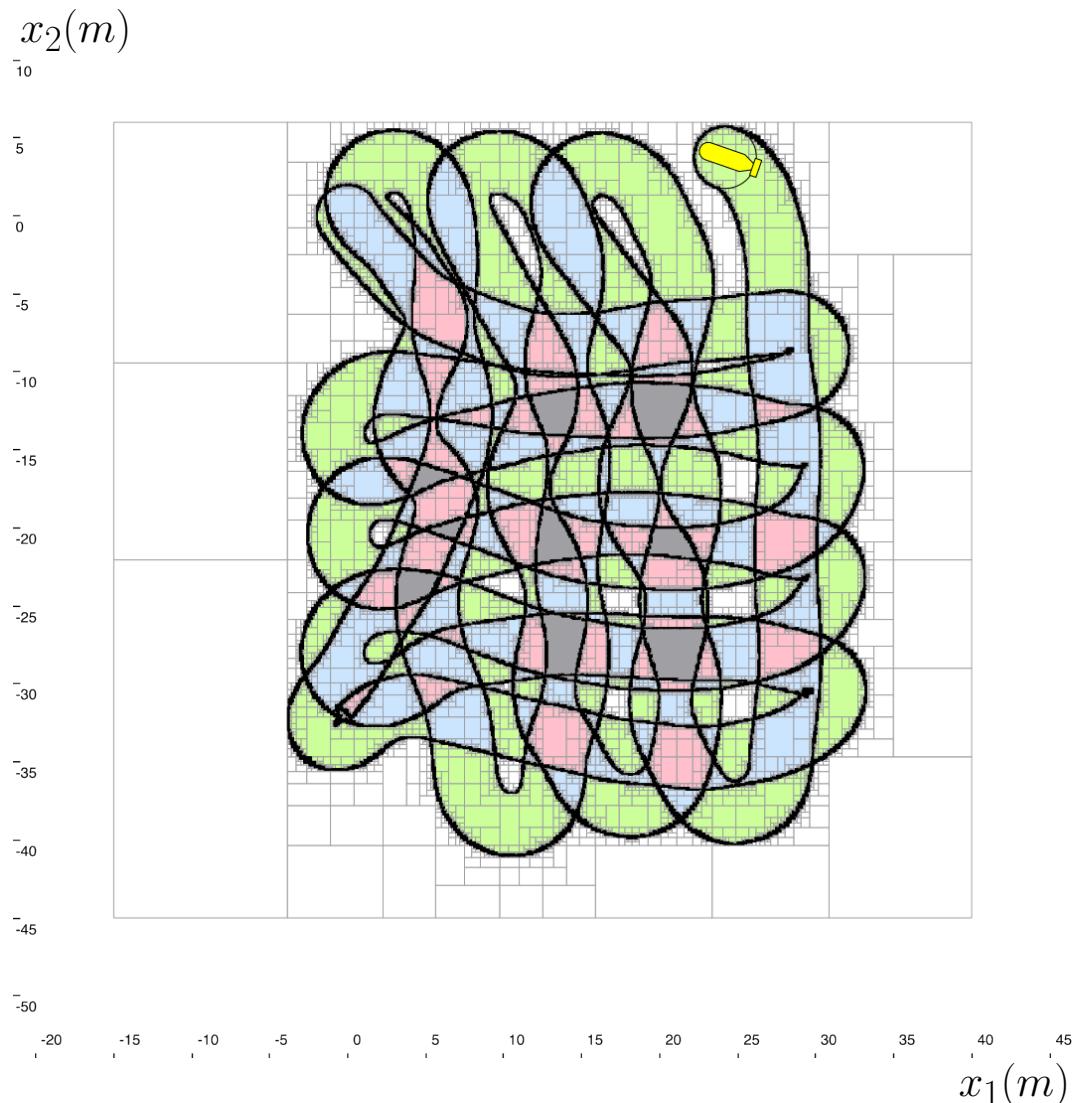


Figure 7.17: Resultant paving of the environment in terms of the coverage measure of its points. Green boxes contain points whose coverage measure is equal to one, blue boxes represent a coverage measure of two, pink boxes a coverage measure of three and gray areas have points with a coverage measure of four.

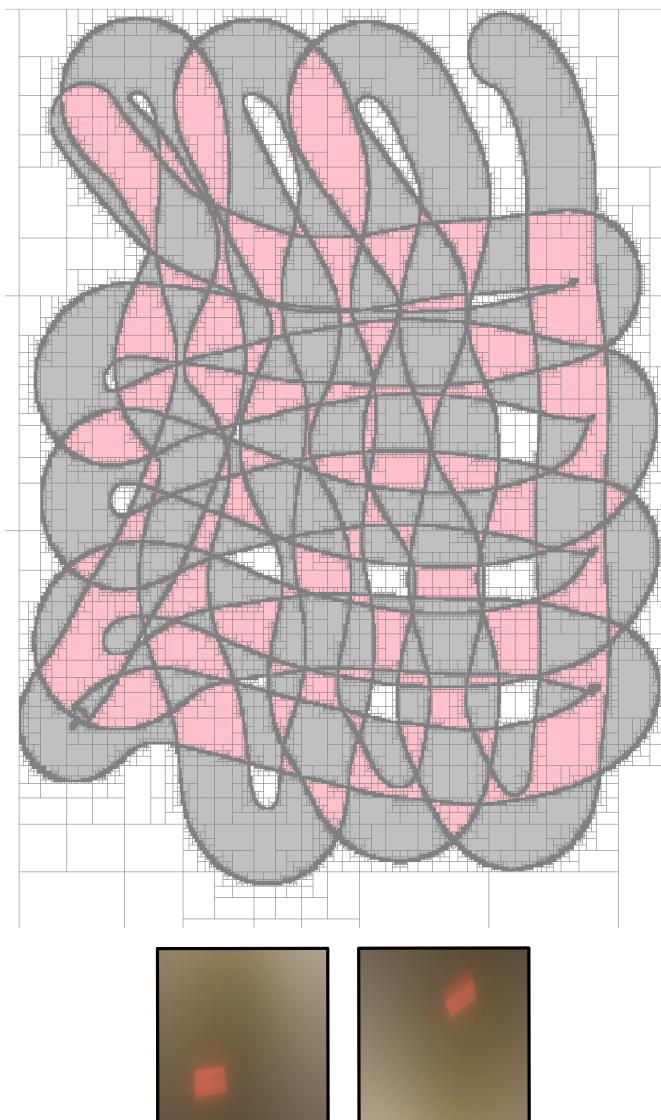


Figure 7.18: Paving of the environment where pink boxes have points for which the coverage measure value is equal to two. The red brick enters entirely in the robot's range of visibility at two different times during the mission. Therefore, the red brick is completely inside one of the highlighted areas in pink.

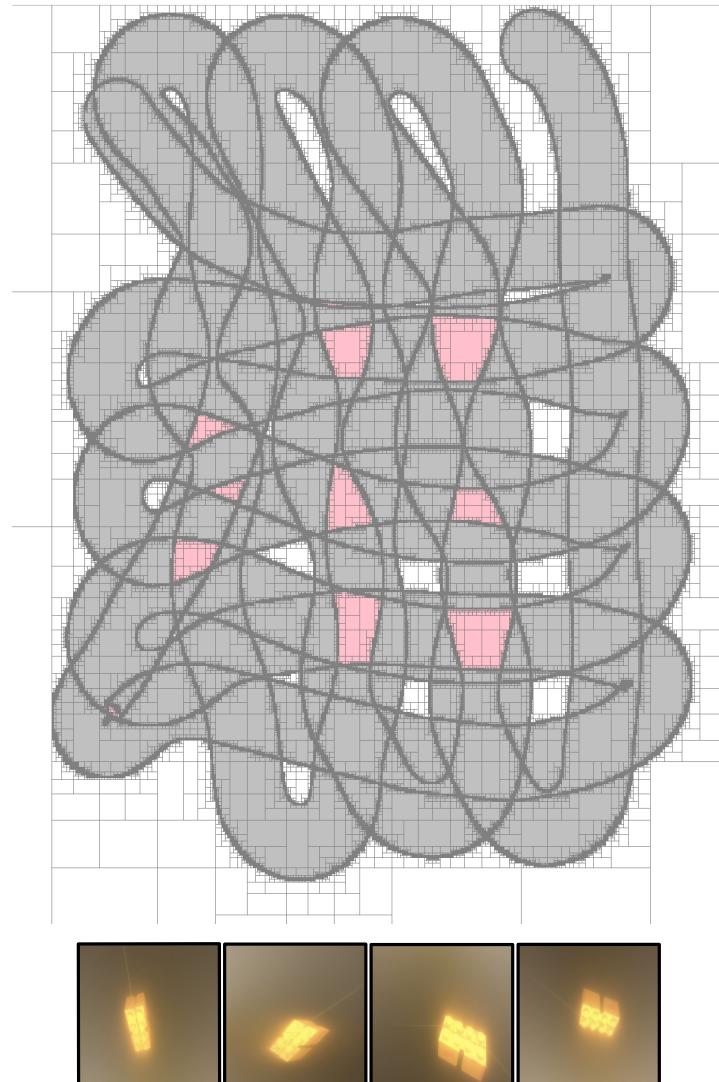


Figure 7.19: Paving of the environment where pink boxes have points for which the coverage measure value is equal to four. The yellow brick enters entirely in the robot's range of visibility at four different times during the mission. Therefore, the yellow brick is completely inside one of the highlighted areas in pink.

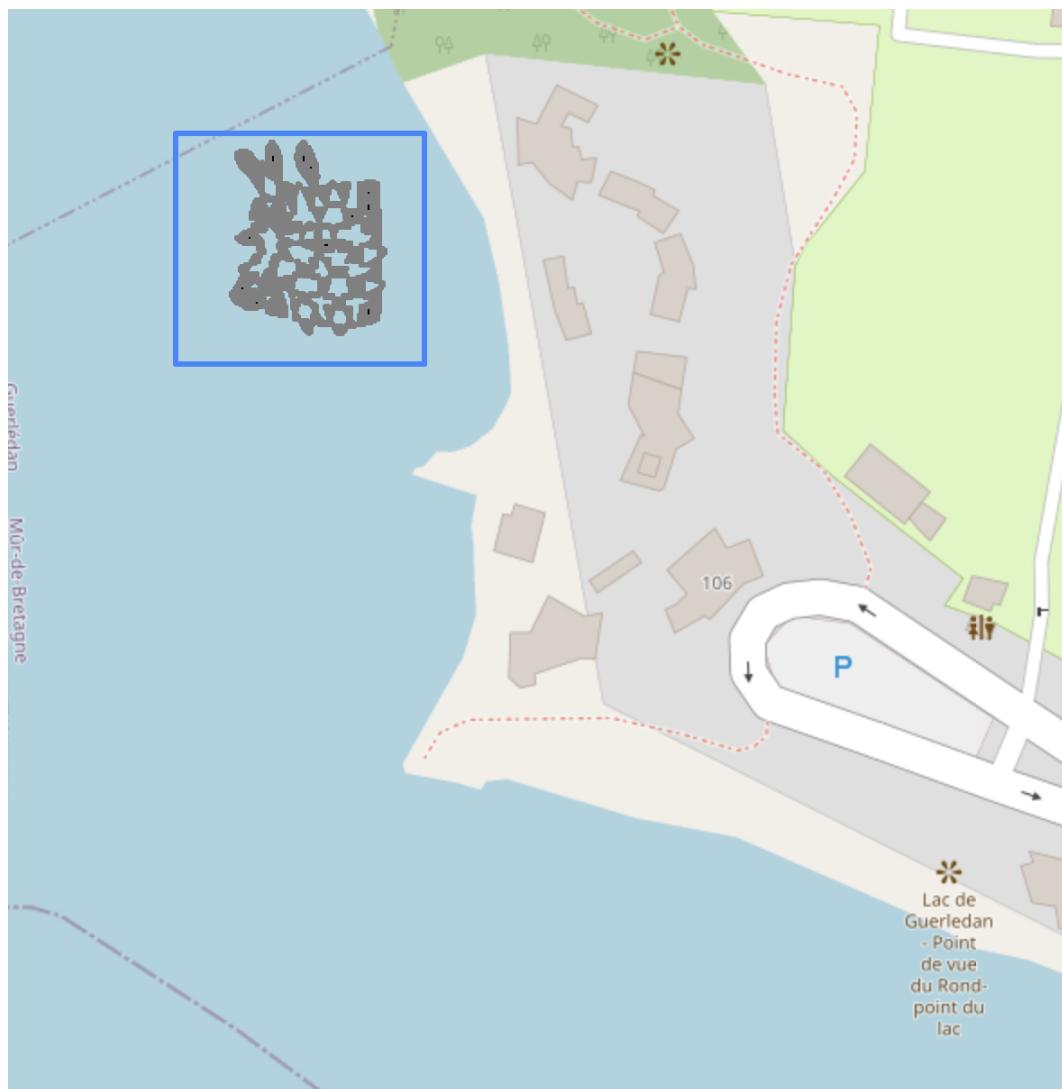


Figure 7.20: Georeferenced representation of the area explored twice on the Guerlédan map. The initial knowledge about the positioning of the red was that it was inside the area surrounded by the blue rectangle. Using the coverage measure estimation this area is contracted to the areas represented in grey on the lake.



Figure 7.21: Georeferenced representation of the area explored four times on the Guerlédan map. The initial knowledge about the positioning of the yellow was that it was inside the area surrounded by the blue rectangle. Using the coverage measure estimation this area is contracted to the areas represented in grey on the lake.

the environment. This allows a refinement on the estimation of the position of the bricks in the lake by contracting the original knowledge about their position, that we considered to be the whole area to be explored. This process is illustrated in Figures from 7.18 to 7.21.

We can conclude that the coverage measure estimation plays a crucial role in refining the initial knowledge about the positioning of targets in the environment. By considering the number of times a specific area has been explored or surveyed, we can narrow down the possibilities for the location of these targets by coupling external information. This leads us to a possible future application of the method into Simultaneous Localization and Mapping (SLAM) algorithms.

SLAM involves the construction of a map of an unknown environment while simultaneously localizing the robot within that environment. While SLAM is a powerful tool, its computational complexity grows with the amount of data the robot needs to process and the size of the environment it is exploring. One of the primary issues with traditional SLAM techniques is that they often treat all parts of the environment equally, leading to exhaustive exploration, which can be extremely time-consuming and resource-intensive. This inefficiency becomes more pronounced when the robot is equipped with sensors that generate high volumes of data, such as LIDAR or high-resolution cameras. In this context, the coverage measure provides a valuable solution to this problem. By integrating the coverage measure into SLAM algorithms, the robot can make informed decisions about where to focus its attention.

7.3 Conclusion

To sum up, this chapter provided practical demonstrations of our algorithms in real-world applications. We showcased the implementation of our area characterization algorithms using real data acquired in these scenarios. These experiments illustrate the value of understanding the coverage measure of the environment. This understanding not only guides mission planning for efficient area coverage, avoiding redundancy, but also offers valuable insights into the data collected during these missions. The following chapter will provide a detailed summary of our work and outline potential directions for future research.

Chapter 8

Conclusions and Prospects

Contents

8.1	Summary	222
8.2	Prospects	222

8.1 Summary

Throughout this document we underscored the importance of determining the area explored during robotics missions. A guaranteed estimation of the explored area serves as a metric for mission success, it validates path-planning algorithms, and overall it ensures the efficiency and safety of missions. Additionally, the concept of a coverage measure is introduced, which quantifies how many times an area has been explored. This measure is valuable for various mission scenarios, including optimizing paths and assessing revisiting missions, for example, for redundancy-based improvements.

In Chapter 5 we formally define the problem at hand. We first define the problem considering a one-dimensional exploration sensor, inspired by side-scan sonars exploration, as presented in Chapter 2. Subsequently, we extend the definitions to accommodate two-dimensional exploration sensors, like cameras. In this chapter we also establish a relation between the topological degree and the coverage measure of points explored by the robot. This relation offers a new framework for developing algorithms for characterizing the explored area that is used in Chapter 6.

Chapter 6 presents novel algorithms for efficiently computing the winding number of a continuous cycle in the plane with respect to all the points in \mathbb{R}^2 . These algorithms are then used for characterizing the environment and the Waterfall space in terms of the coverage measure of its points.

Finally, the use of these algorithms are illustrated through real world experiments in Chapter 7.

8.2 Prospects

The work developed in this thesis can be extended to different contexts.

We strongly believe that topological concepts can be useful and should be explored for solving problems in robotics. We defined in this work the concept of uncertain topological degrees in order to deal with the uncertainty inherent to the localization process of mobile robots in real world. Studying the link between this uncertain topological degree with persistent homology, as in e.g. [38], is a prospect for extending the works of this thesis.

We are also convinced that the coverage measure can be integrated in SLAM algorithms to reduce the exteroceptive data that has to be compared to find possible feature matching. This would reduce the complexity of the algorithms, and it can minimize false matches in homogeneous environments, such as the underwater environment.

Another possibility is to explore the applicability of the developed method

in collaborative multi-robot systems. The use of set-membership methods in the development of algorithms in this thesis should make this extension straightforward.

Lastly, an additional possibility for extension involves incorporating the coverage measure into path-planning algorithms.

Bibliography

- [1] B. Yamauchi, “A frontier-based approach for autonomous exploration,” *Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom. Towards New Comput. Princ. Robot. Autom. (CIRA)*, pp. 146–151, 07 1997.
- [2] S. Tzafestas. Amsterdam, The Netherlands: Elsevier, 2013.
- [3] D. S. Navare, Y. R. Kapde, S. Maurya, D. Pardeshi, and P. William, “Robotic bomb detection and disposal: Application using arduino,” in *2022 7th International Conference on Communication and Electronics Systems (ICCES)*. IEEE, 2022, pp. 479–483.
- [4] J.-H. Kim and B. Y. Lattimer, “Real-time probabilistic classification of fire and smoke using thermal imagery for intelligent firefighting robot,” *Fire Safety Journal*, vol. 72, pp. 40–49, 2015. [Online]. Available: <https://api.semanticscholar.org/CorpusID:110491792>
- [5] J. Casper and R. R. Murphy, “Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center,” *IEEE Trans Syst Man Cybern B Cybern.*, vol. 3, no. 33, 2003.
- [6] H. Ha, S. Han, and J. Lee, “Fault detection on transmission lines using a microphone array and an infrared thermal imaging camera,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, pp. 267–275, 2012. [Online]. Available: <https://api.semanticscholar.org/CorpusID:3089142>
- [7] K. C. Galloway, K. P. Becker, B. T. Phillips, J. Kirby, S. Licht, D. Tchernov, R. J. Wood, and D. F. Gruber, “Soft robotic grippers for biological sampling on deep reefs,” *Soft Robotics*, vol. 3, pp. 23 – 33, 2016. [Online]. Available: <https://api.semanticscholar.org/CorpusID:330615>
- [8] B. Nguyen, B. M. J. Pizarro, A. Shende, and D. J. Stilwell, “An approach to subsea survey for safe naval transit,” *OCEANS’11 MTS/IEEE KONA*, vol. 19, no. 1, pp. 1–6, 2011.

- [9] S. B. Williams, O. R. Pizarro, M. V. Jakuba, C. R. Johnson, N. S. Barrett, R. C. Babcock, G. A. Kendrick, P. D. Steinberg, A. J. Heyward, P. J. Doherty, I. Mahon, M. Johnson-Roberson, D. Steinberg, and A. Friedman, "Monitoring of benthic reference sites: Using an autonomous underwater vehicle," *IEEE Robotics and Automation Magazine*, vol. 19, no. 1, pp. 73–84, 2012.
- [10] N. Oceanic and A. Administration., "The ocean," <https://globalocean.noaa.gov/the-ocean/#:~:text=The%20ocean%20covers%2071%25%20of,97%25%20of%20the%20Earth%27s%20water.>, [Accessed 16-October-2023].
- [11] E. O. Nasa, "Oxygen factories in the southern ocean," 2016.
- [12] U. S. E. P. Agency, "Climate change indicators: Sea surface temperature," 2023.
- [13] S. S. Myers, M. R. Smith, S. Guth, C. D. Golden, B. Vaitla, N. D. Mueller, A. D. Dangour, and P. Huybers, "Climate change and global food systems: potential impacts on food security and undernutrition," *Annual review of public health*, vol. 38, pp. 259–277, 2017.
- [14] R. Silva, E. Mendoza, I. Mariño-Tapián, M. L. Martínez, and E. Escalante, "An artificial reef improves coastal protection and provides a base for coral recovery," 2016. [Online]. Available: <https://api.semanticscholar.org/CorpusID:133259636>
- [15] R. W. Button, J. Kamp, T. B. Curtin, and J. Dryden, *A Survey of Missions for Unmanned Undersea Vehicles*. Santa Monica, CA: RAND Corporation, 2009.
- [16] C. Gambi, M. Canals, C. Corinaldesi, A. Dell'Anno, E. Manea, A. Pusceddu, A. Sanchez-Vidal, and R. Danovaro, "Impact of resuspended mine tailings on benthic biodiversity and ecosystem processes: The case study of portmán bay, western mediterranean sea, spain," *Environmental Pollution*, vol. 301, p. 119021, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0269749122002354>
- [17] N. Oceanic and A. Administration., "Boldly explore where no one has explored before," <https://www.boem.gov/newsroom/ocean-science-news/boldly-explore-where-no-one-has-explored#:~:text=Although%20the%20ocean%20covers%20about,unmapped%2C%20with%20even%20more%20unexplored.>, [Accessed 16-October-2023].

- [18] C. J. Ashton, A. S. Bruce, G. Colledge, and M. Dickinson, “The search for mh370,” *Journal of Navigation*, vol. 68, pp. 1 – 22, 2014. [Online]. Available: <https://api.semanticscholar.org/CorpusID:129463592>
- [19] K. Picard, B. Brooke, and M. Coffin, “Geological insights from malaysia airlines flight mh370 search,” 2017.
- [20] A. Tal, I. Klein, and R. Katz, “Inertial navigation system/doppler velocity log (ins/dvl) fusion with partial dvl measurements,” *Sensors (Basel, Switzerland)*, vol. 17, 2017. [Online]. Available: <https://api.semanticscholar.org/CorpusID:2913970>
- [21] J. Wang, T. Xu, and Z. Wang, “Adaptive robust unscented kalman filter for auv acoustic navigation,” *Sensors (Basel, Switzerland)*, vol. 20, 2019. [Online]. Available: <https://api.semanticscholar.org/CorpusID:209435233>
- [22] Y. Wu, X. Ta, R. Xiao, Y. Wei, D. An, and D. Li, “Survey of underwater robot positioning navigation,” *Applied Ocean Research*, vol. 90, p. 101845, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0141118718305546>
- [23] M. Ferrera, V. Creuze, J. Moras, and P. Trouv -Peloux, “Aqualoc: An underwater dataset for visual–inertial–pressure localization,” *The International Journal of Robotics Research*, vol. 38, no. 14, pp. 1549–1559, 2019.
- [24] A. I. Mourikis and S. I. Roumeliotis, “A multi-state constraint kalman filter for vision-aided inertial navigation,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 3565–3572.
- [25] H. Moravec and A. Elfes, “High resolution maps from wide angle sonar,” in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 116–121.
- [26] C. Luo and S. X. Yang, “A bioinspired neural network for real-time concurrent map building and complete coverage robot navigation in unknown environments,” *IEEE Transactions on Neural Networks*, vol. 19, pp. 1279–1298, 2008. [Online]. Available: <https://api.semanticscholar.org/CorpusID:9307717>
- [27] A. C. Kapoutsis, S. A. Chatzichristofis, and E. B. Kosmatopoulos, “Darp: Divide areas algorithm for optimal multi-robot coverage path planning,”

- Journal of Intelligent & Robotic Systems*, vol. 86, pp. 663–680, 2017. [Online]. Available: <https://api.semanticscholar.org/CorpusID:25496234>
- [28] A. K. Lakshmanan, R. E. Mohan, B. Ramalingam, A. V. Le, P. Veerajagadeeshwar, K. Tiwari, and M. Ilyas, “Complete coverage path planning using reinforcement learning for tetromino based cleaning and maintenance robot,” *Automation in Construction*, vol. 112, p. 103078, 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:214475070>
 - [29] B. Desrochers and L. Jaulin, “Computing a guaranteed approximation of the zone explored by a robot.” *IEEE Transactions on Automatic Control*, vol. 62, pp. 425–430, 2017.
 - [30] V. Drevelle, L. Jaulin, and B. Zerr, “Guaranteed characterization of the explored space of a mobile robot by using subpavings,” in *IFAC Symposium on Nonlinear Control Systems*, 2013. [Online]. Available: <https://api.semanticscholar.org/CorpusID:3134550>
 - [31] D. P. Williams and J. Groen, “Multi-view target classification in synthetic aperture sonar imagery,” 2009. [Online]. Available: <https://api.semanticscholar.org/CorpusID:18815823>
 - [32] I. Quidu, J.-P. Malkasse, G. Burel, and P. Vilbé, “Mine classification based on a multiview characterisation,” 2001. [Online]. Available: <https://api.semanticscholar.org/CorpusID:17420884>
 - [33] B. Ding and G. Wen, “Exploiting multi-view sar images for robust target recognition,” *Proceedings of the 3rd International Conference and Exhibition on Underwater Acoustic Measurements: Technologies and Results*, vol. 9, 2017.
 - [34] H. Su, S. Maji, E. Kalogerakis, and E. G. Learned-Miller, “Multi-view convolutional neural networks for 3d shape recognition,” *CoRR*, vol. abs/1505.00880, 2015. [Online]. Available: <http://arxiv.org/abs/1505.00880>
 - [35] D. P. Williams, F. Baralli, M. Micheli, and S. Vasoli, “Adaptive underwater sonar surveys in the presence of strong currents,” *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2604–2611, 2016. [Online]. Available: <https://api.semanticscholar.org/CorpusID:15911335>

- [36] Y. Baryshnikov and R. Ghrist, “Target enumeration via euler characteristic integrals,” *SIAM Journal on Applied Mathematics*, vol. 70, no. 3, pp. 825–844, 2009. [Online]. Available: <https://doi.org/10.1137/070687293>
- [37] V. De Silva and R. Ghrist, “Coordinate-free coverage in sensor networks with controlled boundaries via homology,” *The International Journal of Robotics Research*, vol. 25, no. 12, pp. 1205–1222, 2006. [Online]. Available: <https://doi.org/10.1177/0278364906072252>
- [38] V. De Silva and G. R., “Coverage in sensor networks via persistent homology,” *Algebraic & Geometric Topology*, vol. 7, no. 1, pp. 339 – 358, 2007. [Online]. Available: <https://doi.org/10.2140/agt.2007.7.339>
- [39] F. Stenger, “Computing the topological degree of a mapping in R^n ,” *Numerische Mathematik*, vol. 25, pp. 23–38, 1975.
- [40] P. Franek, “Effective topological degree computation based on interval arithmetic,” *Mathematics of Computation*, vol. 84, no. 293, p. 1265–1290, 2014.
- [41] N. Oceanic and A. Administration., “What is sonar?” <https://oceanservice.noaa.gov/facts/sonar.html>, [Accessed 15-June-2023].
- [42] M. Ludvigsen and A. J. Sørensen, “Towards integrated autonomous underwater operations for ocean mapping and monitoring,” *Annual Reviews in Control*, vol. 42, pp. 145–157, 2016.
- [43] D. D. Sternlicht, J. E. Fernandez, R. Holtzapple, D. P. Kucik, T. C. Montgomery, and C. M. Loeffler, “Advanced sonar technologies for autonomous mine countermeasures,” *OCEANS*, pp. 1–5, 2011.
- [44] D. of Sound in the Sea, “The first studies of underwater acoustics: The 1800s,” <https://dosits.org/people-and-sound/history-of-underwater-acoustics/the-first-studies-of-underwater-acoustics-the-1800s/>, [Accessed 15-June-2023].
- [45] D. M. McLean, “Confronting technological and tactical change: Allied antisubmarine warfare in the last year of the battle of the atlantic.” *Naval War College Review*, vol. 47, no. 1, pp. 87–104, 1994.
- [46] P. Blondel and B. Murton, *Handbook of Seafloor Sonar Imagery*. New York: Wiley, 1997.

- [47] A. Burguera and G. Oliver, "High-resolution underwater mapping using side-scan sonar," *PLoS One*, vol. 11, no. 1, 2016.
- [48] C. P. and C. De Moustier, "Sidescan sonar image processing techniques," *IEEE Journal of Oceanic Engineering*, vol. 18, no. 2, pp. 108–122, 1993.
- [49] D. J. and C. R., "Resolution measurement for synthetic aperture sonar," *OCEANS 2019*, pp. 1–6, 2019.
- [50] H. J. Flowers and J. E. Hightower, "A novel approach to surveying sturgeon using side-scan sonar and occupancy modeling," *Marine and Coastal Fisheries: Dynamics, Management, and Ecosystem Science*, vol. 5, no. 1, pp. 211–223, 2013.
- [51] M. Bouvet and B. Murton, *Traitements des signaux pour les systèmes sonar*. Paris: Elsevier-masson, 1997.
- [52] A. Penko, J. Calantoni, and B. T. Hefner, "Modeling and observations of sand ripple formation and evolution during trex13," *IEEE Journal of Oceanic Engineering*, vol. 42, no. 2, pp. 260–267, 2017.
- [53] K. A. Kastens and M. B. Cita, "Tsunami-induced sediment transport in the abyssal mediterranean sea," *GSA Bulletin*, vol. 92, no. 11, pp. 845–857, 1981.
- [54] G. T., S. K., and Y. Petillot, "Three-dimensional reconstruction of underwater objects using wide-aperture imaging sonar," *Journal of Field Robotics*, vol. 35, no. 6, pp. 890–905, 2018.
- [55] C. Brown, A. Hewer, W. Meadows, D. Limpenny, K. Cooper, and H. Rees, "Mapping seabed biotopes at hastings shingle bank, eastern english channel. part 1. assessment using sidescan sonar," *Journal of the Marine Biological Association of the UK*, vol. 84, no. 3, pp. 481–488, 2004.
- [56] T. Sheffer and H. Guterman, "Geometrical correction of side-scan sonar images," *IEEE International Conference on the Science of Electrical Engineering in Israel (ICSEE)*, pp. 1–5, 2018.
- [57] M. F. Fallon, M. Kaess, H. Johannsson, and J. J. Leonard, "Efficient auv navigation fusing acoustic ranging and side- scan sonar," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–8, 2011.

- [58] S. Reed, Y. Petillot, and B. J., “An automatic approach to the detection and extraction of mine features in sidescan sonar.” *IEEE Journal of Oceanic Engineering.*, vol. 28, no. 1, pp. 90–105, 2003.
- [59] M. Legris, K. Lebart, F. Fohanno, and b. zerr, “Les capteurs d’imagerie en robotique sous-marine: tendances actuelles et futures,” *Traitemet du Signal*, vol. 20, pp. 137–164, 01 2003.
- [60] B. D. Van Veen and K. M. Buckley, “Beamforming: A versatile approach to spatial filtering,” *IEEE ASSP Magazine*, 1988.
- [61] M. Magazines, “The journey to helping auvs think: How marine roboticists are turning auv sight into perception,” <https://www.maritimemagazines.com/marine-technology/202010/the-journey-to-helping-auvs-think/>, [Accessed 15-June-2023].
- [62] X. K. Li, L. Xie, and Y. Qin, “Underwater target feature extraction using hilbert-huang transform,” *Journal of Harbin Engineering University*, p. 542–546, 2009.
- [63] R. Baran and J. Coughlin, “A neural network for target classification using passive sonar.” In *Proceedings of the Conference on Analysis of Neural Network Applications*, p. 188–198, 1991.
- [64] D. Singh and M. Valdenegro-Toro, “The marine debris dataset for forward-looking sonar semantic segmentation,” *CoRR*, vol. abs/2108.06800, 2021. [Online]. Available: <https://arxiv.org/abs/2108.06800>
- [65] A. Sethuraman and K. A. Skinner, “Towards sim2real for shipwreck detection in side scan sonar imagery,” *3rd Workshop on Closing the Reality Gap in Sim2Real Transfer for Robotics*, 2022.
- [66] D. Neupane and J.-H. Seok, “A review on deep learning-based approaches for automatic sonar target recognition,” *Electronics*, vol. 9, p. 1972, 2020.
- [67] H. Peyvandi, M. Farrokhooy, H. Roufarshbaf, and S. J. Park, *SONAR Systems and Underwater Signal Processing: Classic and Modern Approaches*. London: IntechOpen, 2011.
- [68] S. Hozyn, “A review of underwater mine detection and classification in sonar imagery.” *Electronics*, vol. 10, p. 2943, 2021.

- [69] K. Ghorbal, E. Goubault, and S. Putot, “A logical product approach to zonotope intersection,” *CoRR*, vol. abs/1002.2236, 2010. [Online]. Available: <http://arxiv.org/abs/1002.2236>
- [70] E. Walter and H. Piet-Lahanier, “Exact recursive polyhedral description of the feasible parameter set for bounded-error models,” *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 911–915, 1989.
- [71] A. Rauh and L. Jaulin, “A novel thick ellipsoid approach for verified outer and inner state enclosures of discrete-time dynamic systems,” *IFAC-PapersOnLine*, vol. 54, no. 87, pp. 601–606, 2021.
- [72] L. Jaulin and E. Walter, “Set inversion via interval analysis for nonlinear bounded-error estimation,” *Automatica*, vol. 29, no. 4, pp. 1053–1064, 1993.
- [73] L. Jaulin, I. Braems, and E. Walter, “Interval methods for nonlinear identification and robust control.” *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 4, pp. 4676–4681, 2002.
- [74] H. Schichl and A. Neumaier, “Interval analysis on directed acyclic graphs for global optimization.” *J Glob Optim*, vol. 33, no. 1, p. 541–562, 2005.
- [75] P. Saraev, “Numerical methods of interval analysis in learning neural network.” *Autom Remote Control*, vol. 73, p. 1865–1876, 2012.
- [76] L. Jaulin and F. Le Bars, “An interval approach for stability analysis: Application to sailboat robotics.” *IEEE Transactions on Robotics*, vol. 29, no. 1, pp. 282–287, 2013.
- [77] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis*. London: Springer London, 2001.
- [78] R. E. Moore and C. Yang, “Interval analysis i. technical document lmsd-285875,” *Lockheed Missiles and Space Division, Sunnyvale, CA, USA.*, 1959.
- [79] P. Bouron, “Méthodes ensemblistes pour le diagnostic, l'estimation d'état et la fusion de données temporelles.” *PhD thesis*, 2002.
- [80] J. Trevor and J. Palsberg, “Type inference in systems of recursive types with subtyping,” *Available on authors' web page*, 1999.
- [81] P. Hell and J. Nešetřil, “Colouring, constraint satisfaction, and complexity,” *Computer Science Review*, vol. 2, no. 3, pp. 143–163, 2008.

- [82] H. Simonis, “Sudoku as a constraint problem.” *CP Workshop on modeling and reformulating Constraint Satisfaction Problems*, vol. 12, pp. 13–27, 2005.
- [83] V. Kumar, “Algorithms for constraint-satisfaction problems: A survey.” *AI magazine*, vol. 13, no. 1, pp. 32–32, 1992.
- [84] L. Jaulin, “Solving set-valued constraint satisfaction problems.” *Computing*, vol. 94, pp. 297–311, 2012.
- [85] ———, “Range-only slam with occupancy maps: A set-membership approach.” *IEEE Transactions on Robotics*, vol. 27, pp. 1004–1010, 2011.
- [86] G. Chabert and L. Jaulin, “Contractor programming,” *Artificial Intelligence*, vol. 173, no. 11, pp. 1079–1100, 2009. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0004370209000381>
- [87] S. Rohou, “Reliable robot localization: a constraint programming approach over dynamical systems,” *PhD thesis*, 2017.
- [88] F. Benhamou, F. Goualard, L. Granvilliers, and J. Puget, “Revising hull and box consistency,” *Int. Conf. on Logic Programming*, p. 230–244, 1999.
- [89] B. Desrochers and L. Jaulin, “A minimal contractor for the polar equation: Application to robot localization.” *Engineering Applications of Artificial Intelligence*, vol. 55, p. 83–92, 2016.
- [90] L. Jaulin and B. Desrochers, “Introduction to the algebra of separators with application to path planning,” *Engineering Applications of Artificial Intelligence*, vol. 33, no. 1, pp. 141–147, 2014.
- [91] B. Desrochers, “Simultaneous localization and mapping in unstructured environments.” *Universite Bretagne Loire*, 2018.
- [92] J. Sliwka, F. Le Bars, O. Reynet, and L. Jaulin, “Using interval methods in the context of robust localization of underwater robots,” *Annual Meeting of the North American Fuzzy Information Processing Society*, pp. 1–6, 2011.
- [93] R. Guyonneau, S. Lagrange, L. Hardouin, and P. Lucidarme, “Guaranteed interval analysis localization for mobile robots,” *Advanced Robotics*, vol. 28, no. 16, pp. 1067–1077, 2014.

- [94] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features.” *In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*, vol. 1, p. I–511–I–518, 2001.
- [95] ———, “A note on the computation of high-dimensional integral images.” *Pattern Recognition Letters.*, vol. 32, no. 2, p. 197–201, 2011.
- [96] B. Desrochers, S. Lacroix, and L. Jaulin, “Set-membership approach to the kidnapped robot problem.” *In IEEE International Conference on Intelligent Robots and Systems*, p. 3715–3720, 2015.
- [97] L. Jaulin and E. Walter, “Set inversion via interval analysis for nonlinear bounded-error estimation,” *Automatica*, vol. 29, pp. 1053–1064, 1993.
- [98] D. Dubois, L. Jaulin, and H. Prade, *Thick Sets, Multiple-Valued Mappings and Possibility Theory*, 2021, pp. 101–109.
- [99] S. Rohou, B. Desrochers *et al.*, “The Codac library – Constraint-programming for robotics,” 2022, <http://codac.io>.
- [100] K. Apt, “The essence of constraint propagation.” *Theoretical Computer Science*, vol. 221, no. 1-2, p. 179–210, 1998.
- [101] J. Yao, Y. Yao, V. Kreinovich, P. P. da Silva, S. Starks, G. Xiang, and H. T. Nguyen, “Towards more adequate representation of uncertainty: From intervals to set intervals, with the possible addition of probabilities and certainty degrees.” *Proceedings of the IEEE World Congress on Computational Intelligence*, p. 983–990, 2008.
- [102] B. Desrochers and L. Jaulin, “Thick set inversion.” *Artificial Intelligence*, vol. 249, pp. 1–18, 2017.
- [103] A. B. Kurzhanski and T. F. Filippova, “On the theory of trajectory tubes - a mathematical formalism for uncertain dynamics, viability and control.” *Advances in Nonlinear Dynamics and Control: A Report from Russia*, p. 122–188, 1993.
- [104] F. Le Bars, J. Sliwka, L. Jaulin, , and O. Reynet, “Set-membership state estimation with fleeting data.” *Automatica*, vol. 48, no. 2, p. 381–387, 2012.
- [105] A. Bethencourt and L. Jaulin, “Solving non-linear constraint satisfaction problems involving time-dependant functions.” *Mathematics in Computer Science*, vol. 8, no. 3, p. 503–523, 2014.

- [106] R. D. Clément Aubry and L. Jaulin, “Loop detection of mobile robots using interval analysis,” *Automatica*, vol. 49, no. 2, pp. 463–470, 2013.
- [107] C. Gorini, “Geometry for the artist: An interdisciplinary consciousness-based course,” *SSRN Electronic Journal*, vol. 3, 2018.
- [108] R. Ghrist, *Elementary Applied Topology*, 2014. [Online]. Available: <https://www2.math.upenn.edu/~ghrist/notes.html>
- [109] J. M. Lee, *Introduction to Topological Manifolds*, 1st ed., ser. Graduate Texts in Mathematics. Springer, 2000, vol. 202.
- [110] A. Hatcher, *Algebraic Topology*, ser. Algebraic Topology. Cambridge University Press, 2002. [Online]. Available: <https://pi.math.cornell.edu/~hatcher/AT/AT.pdf>
- [111] M. W. Hirsch, *Differential Topology*, ser. Graduate Texts in Mathematics. Springer New York, NY, 1976.
- [112] J. R. Munkres, *Topology*, 2nd ed. Prentice Hall, Inc., 2000.
- [113] M. A. Armstrong, *Basic Topology*. McGraw-Hill Book Company, Maidenhead, UK, 1979.
- [114] S. Willard, *General Topology*. Dover Publications, 2004.
- [115] R. B. Melrose, “Differential analysis on manifolds with corners,” 1996.
- [116] D. Joyce, “On manifolds with corners,” 2010.
- [117] N. Jacobson, *Basic Algebra I*. Dover Publications, 2009.
- [118] E. H. Spanier, *Algebraic Topology*. Springer, 1996.
- [119] E. Outerelo and J. M. Ruiz, *Mapping Degree Theory*. American Mathematical Soc., 2009, vol. 108.
- [120] W. J. Kaczor and M. T. Nowak, *Problems in Mathematical Analysis: Real numbers, sequences, and series*. American Mathematical Society, 2000, vol. 1.
- [121] I. Kurzhanski, A. ad Valyi. Boston, MA: Birkhäuser, 1997.
- [122] S. Rohou, L. Jaulin, L. Mihaylova, F. Le Bars, and S. M. Veres, “Guaranteed computation of robot trajectories,” *Robotics and Autonomous Systems*, vol. 93, pp. 76–84, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0921889016304006>

- [123] F. Le Bars, J. Sliwka, L. Jaulin, and O. Reynet, “Set-membership state estimation with fleeting data,” *Automatica*, vol. 48, no. 2, pp. 381–387, 2012. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0005109811005322>
- [124] H. Agarwal and R. G. Tiwari, “Exploiting sensor fusion for mobile robot localization,” *Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pp. 463–466, 2019.
- [125] M. Kam, X. Zhu, and P. Kalata, “Sensor fusion for mobile robot navigation,” *Proceedings of the IEEE*, vol. 85, no. 1, pp. 108–119, 1997.
- [126] D. Alciatore and R. Miranda, “A winding number and point-in-polygon algorithm,” 1995.
- [127] J. W. Alexander, “Topological invariants of knots and links,” *Transactions of the American Mathematical Society*, vol. 30, p. 275–306, 1928.
- [128] A. Kulowski, “Optimization of a point-in-polygon algorithm for computer models of sound field in rooms,” *Applied Acoustics*, vol. 35, pp. 63–74, 1992.
- [129] M. Shimrat, “Algorithm 112: Position of point relative to polygon,” *Communications of the ACM*, vol. 5, no. 8, p. 434, 1962.
- [130] L. Guibas, L. Ramshaw, and J. Stolfi, “A kinetic framework for computational geometry,” in *24th Annual Symposium on Foundations of Computer Science (sfcs 1983)*, 1983, pp. 100–111.
- [131] D. Sunday, “Inclusion of a point in polygon,” https://web.archive.org/web/20130126163405/http://geomalgorithms.com/a03-_inclusion.html, [Accessed 20-Septembre-2023].
- [132] A. Möbius, “Über die bestimmung des inhaltes eines polyëders,” *Berichte über die Verhandlungen der Königlich Sächsischen Gesellschaft der Wissenschaften, Mathematisch-Physische Klasse*, vol. 17, pp. 31–68, 1865.
- [133] M. McIntyre and G. Cairns, “A new formula for winding numbers,” *Geometriae Dedicata*, vol. 46, 1993.
- [134] S. Rohou, P. Franek, C. Aubry, and L. Jaulin, “Proving the existence of loops in robot trajectories,” *The International Journal of Robotics Research*, vol. 37, no. 12, pp. 1500–1516, 2018. [Online]. Available: <journals.sagepub.com/doi/full/10.1177/0278364918808367>

- [135] C. Li, W.-H. Zhang, and X.-Q. Jin, “Convergence and uniqueness properties of gauss-newton’s method,” *Computers & Mathematics with Applications*, vol. 47, no. 6-7, pp. 1057–1067, 2004.
- [136] Itseez, “Open source computer vision library,” <https://github.com/itseez/opencv>, 2015.
- [137] G. Berg, W. Julian, R. Mines, and F. Richman, “The constructive jordan curve theorem,” *The Rocky Mountain Journal of Mathematics*, pp. 225–236, 1975.
- [138] F. Bars, R. Sanchez, L. Jaulin, S. Rohou, and A. Rauh, “An online interval-based inertial navigation system for control purposes of autonomous boats,” *Frontiers in Control Engineering*, vol. 2, p. 786188, 03 2022.
- [139] “Hydrographes et roboticiens explorent guerlédan,” <https://guerledan.ensta-bretagne.fr>, accessed: 2023-08-24.