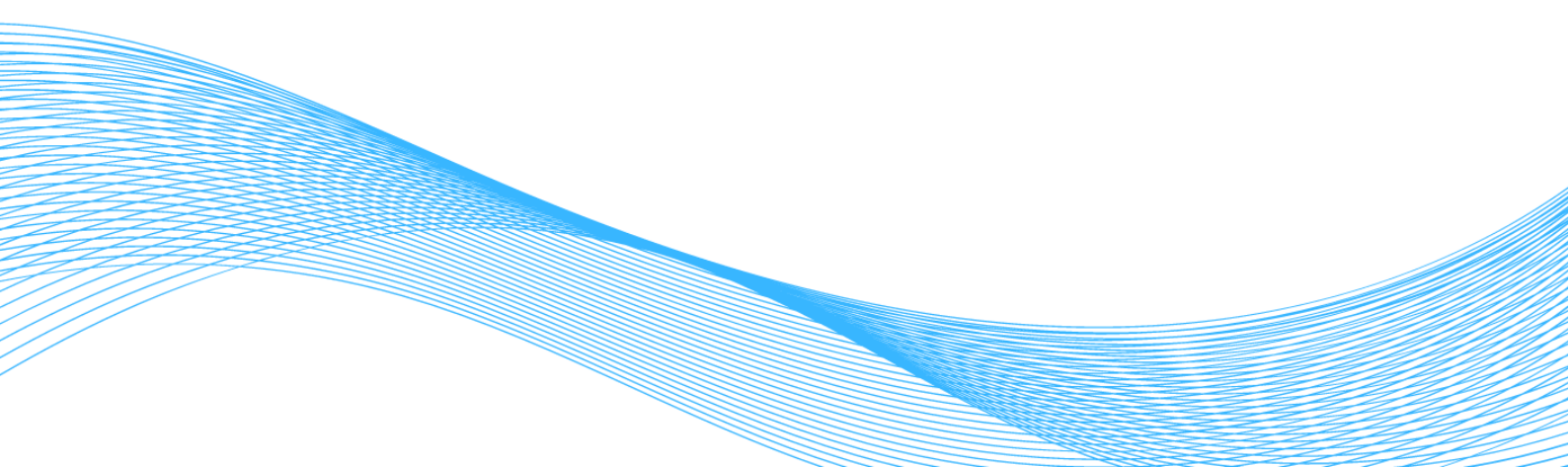

MULTI-ROBOT COORDINATION

AUTONOMOUS EXPLORATION OF GALLERY NETWORKS

Engineering Graduation Project
Seatech 3A - MOCA

Auteur :
Fabien MATHÉ

Enseignant :
M. Mehmet ERSOY

A decorative graphic at the bottom of the page consisting of numerous thin, overlapping blue lines that form a wavy, undulating pattern.

Abstract

Mots-clés:

MOTS-CLÉS

Abstract

Keywords:

KEYWORDS

Remerciements

Contents

Résumé	1
Remerciements	2
Introduction	4
I State of the art	5
I.1 État de l'Art : Exploration Multi-Robot de Cavités	5
I.2 État de l'Art des Méthodes de Communication Inter-Robot dans un Environnement de Cavité	5
II Partie 2	7
II.1 Purpose and range of the simulator	7
II.2 Creation of the map	7
II.3 Robot implementation	9
III Partie 3	10
III.1 Finding the shortest path distance	10
IV Partie 4	12
V Partie 5	13
Conclusion	14
Perspectives	15
References	16
Annexes	17

Introduction

Definition d'un robot, Spécification du type étudier dans ce rapport, les défis de la planification de trajectoire, l'évitement d'obstacle (petit point sur l'optimisation de trajectoire en milieu ouvert pour de la recherche de victime par exemple) Les contraintes extérieures et celle du robot lui même.

L'un des ouvrages de référence utilisés dans ce travail est le livre de S. M. LaValle, *Planning Algorithms* [1], que je remercie particulièrement pour son engagement à rendre ses travaux accessibles au public (<https://lavalle.pl/>).

I State of the art

I.1 État de l'Art : Exploration Multi-Robot de Cavités

L'exploration des cavités est essentielle pour la cartographie souterraine, la recherche scientifique et les interventions d'urgence, tout en posant des défis uniques pour la robotique autonome.[2] Leur exploration multi-robot est un domaine de recherche en plein essor dans le domaine de la robotique, avec des applications dans des environnements variés tels que l'exploration spatiale, l'exploration de grottes, les missions de sauvetage dans des environnements urbains sinistrés, ou encore l'extraction minière.[3, 4] Ces environnements, souvent complexes et dynamiques, présentent de nombreux défis pour la planification de mission, notamment des obstacles imprévisibles, des zones inaccessibles pour les robots mobiles, et l'absence de signal GPS. De ce fait, l'exploration multi-robot permet de tirer parti de la coopération entre robots pour surmonter ces difficultés.

I.1.1 Défis dans l'exploration multi-robot de cavités

L'exploration de cavités avec plusieurs robots implique plusieurs défis techniques. En particulier, la gestion de la coopération entre robots et la gestion de l'information dans des environnements complexes sont deux aspects fondamentaux de ce type de mission. Les défis peuvent être classés en plusieurs catégories :

I.2 État de l'Art des Méthodes de Communication Inter-Robot dans un Environnement de Cavité

La communication inter-robot (CIR) dans un environnement de cavité est un défi majeur pour les systèmes multi-robots, en raison des conditions particulières que ces environnements présentent, telles que des espaces confinés, des obstacles physiques et des perturbations qui peuvent affecter les signaux de communication. Les méthodes de CIR peuvent être classées en différentes catégories selon la technologie utilisée et la stratégie de communication adoptée. Les approches classiques reposent principalement sur des réseaux sans fil, tels que la communication par radiofréquence (RF), qui est couramment utilisée pour des applications de communication à longue portée mais qui peut souffrir de limitations dans les cavités où les signaux sont atténués par des parois solides. Dans ce contexte, des méthodes de communication ad hoc sont souvent employées, où les robots créent un réseau dynamique de relais pour échanger des informations. Une approche courante est l'utilisation de la *communication par maillage*, dans laquelle chaque robot agit comme un relais, permettant ainsi une couverture étendue et une transmission de données entre robots même lorsque les obstacles interfèrent avec les signaux directs. La *communication acoustique* ou *ultrasonique*, qui repose sur des ondes sonores, est une alternative viable dans les environnements de cavité, où elle peut transmettre des informations de manière robuste, notamment pour des distances plus courtes ou dans des environnements très confinés. De plus, la communication optique (par exemple, la *communication par lumière visible ou infrarouge*) devient de plus en plus populaire dans des applications de haute précision dans des cavités étroites, car elle est peu sensible aux interférences électromagnétiques et permet des transmissions rapides et sécurisées. Une autre stratégie consiste à utiliser des protocoles de communication coopérative, où les robots collaborent pour optimiser le flux d'informations. Des techniques de *routage adaptatif* et de *partitionnement dynamique de réseau* sont souvent utilisées pour gérer les obstacles et les interférences dans les réseaux multi-robots. Enfin, les méthodes de *communication opportuniste*, basées sur des échanges de données ponctuels lorsque la ligne de visée est dégagée, sont particulièrement adaptées aux environnements où la connectivité est intermittente et où les robots doivent s'organiser pour minimiser les interruptions dans

le flux de données. L'un des principaux défis reste de maintenir une communication fiable et efficace, notamment en optimisant les stratégies de répartition de la bande passante, d'allocation des ressources et de gestion des interférences dans des espaces très contraints. Des approches récentes cherchent à intégrer des systèmes hybrides combinant ces technologies pour assurer une meilleure résilience face aux conditions environnementales changeantes et aux mouvements des robots.

1.2.1 Planification de chemin, de trajectoire et évitement d'obstacle

La planification de chemin pour un système mécatronique constitue le fondement de tous les systèmes mobiles autonomes, qu'il s'agisse de drones ou de bras de manutention et d'assemblage. Le principe de la planification de chemin ou de trajectoire est de déterminer une solution - un chemin ou une trajectoire - reliant un point de départ à un point cible.

La distinction entre planification de chemin et planification de trajectoire réside dans le fait qu'un chemin planifié n'est pas nécessairement réalisable par un robot. En effet, la planification de chemin ne prend pas toujours en compte la faisabilité physique ou cinématique pour un robot mobile.

L'évitement d'obstacles est une contrainte essentielle dans ces deux approches. Les obstacles définissent les zones inaccessibles, et comme nous le verrons par la suite, leur nature - mobile ou immobile - détermine en grande partie la méthode à employer pour résoudre le problème de planification.

Parmi les méthodes déterministe, on trouve une large variété de méthodes principalement basées sur trois approches différentes. Les approches par graphes, celle de décomposition cellulaire et enfin les celles utilisant des champs potentiels.[5, 6]

Les méthodes des graphes consistent à construire une carte des chemins empruntable en partant des obstacles de la scène. Parmi ces méthodes utilisant des graphes, on peut distinguer quatre types différents : Les graphes de visibilité[7], les diagrammes de Voronoï [8] ou encore la méthode des Silhouettes[6].

Les méthodes associées à la décomposition cellulaire consistent à diviser l'espace libre du robot en régions simples, appelées cellules, où il est facile de générer un chemin entre deux configurations. Un graphe représentant les relations d'adjacence entre les cellules est ensuite construit et exploré.[5, 9, 10, 11]

Une autre méthode repose sur une subdivision fine de l'espace afin de repérer les zones libres. La méthode des champs potentiels s'appuie sur cette idée en définissant des potentiels qui traduisent des forces d'attraction, dirigées vers les coordonnées cibles, et de répulsion, correspondant par exemple aux obstacles. Le chemin est ensuite déterminé en suivant l'opposé du gradient du potentiel total ainsi calculé.[5, 12]

Des approches alternatives ont été développées dans les années 1990 et 2000, notamment les méthodes stochastiques de planification de chemin RPP (*Random Path Planners*) et PRM (*Probabilistic Roadmap Planners*). Ces méthodes consistent à échantillonner l'espace de manière aléatoire afin de créer un graphe de chemins possibles (*roadmap*) dans cet espace.[13, 14, 15]

L'un des principaux atouts de cette méthode de planification de chemin est qu'elle ne dépend ni de la structure de l'espace, ni du nombre d'obstacles, ni de leur disposition. Une fois la *roadmap* créée, il suffit d'utiliser une méthode de recherche de chemin dans un graphe pour trouver le chemin menant au point objectif.[16]

II Partie 2

II.1 Purpose and range of the simulator

The simulator is designed to test various algorithms and methods for multi-robot exploration of caves. The robots operate in the air while moving along the floor. To simplify the problem, we approximate the floor as a 2D plane without any surface irregularities.

II.2 Creation of the map

The map consist in succession of straight lines generate by a simple cellular automata.

A cellular automaton is a grid of cells, where each cell can exist in different states based on predefined rules. The concept was developed by Stanislaw Ulam and John von Neumann in the 1940s.

The most famous cellular automaton, which helped popularize its use, was developed by John Conway: *Conway's Game of Life*. This model follows a set of four simple rules, which can be found here. The rules are based on the properties of neighboring cells.

There are two commonly used neighborhood types in cellular automata:

- **Von Neumann neighborhood**, which considers the four direct neighbors (left, top, right and bottom).
- **Moore neighborhood**, an extension that includes all eight surrounding cells, both diagonal and direct neighbors.

The beauty of this system lies in the complexity that emerges from such simple rules. Beginning with the configuration shown in Figure II.1a, the system evolves into the states illustrated in Figure II.1 at generations 87 and 263.



Figure II.1: Conway's Game of Life

Programmers, computer scientists, and mathematicians began cataloging all the patterns they encountered and constructed remarkably complex machines.

In our case, the rules are defined as follows:

- If there are more than 4 activated cells in the Moore neighborhood, the cell activates.
- Otherwise, the cell deactivates.

I implement the method on a Cartesian grid and then transform it into a triangular mesh, as shown in Figure II.2. The maps generated after this transformation improved in quality.

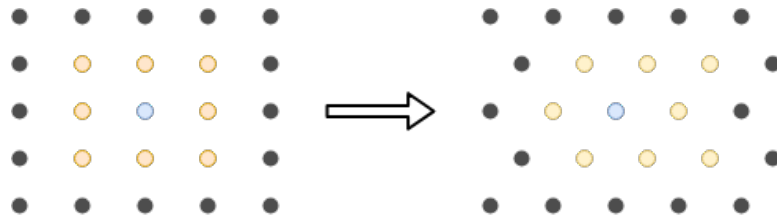


Figure II.2: Grid transformation

Using the Marching Squares method, I draw the boundaries between occupied cells (red dots) and unoccupied cells (green dots).

Marching Squares is a technique for generating the contours of a two-dimensional grid, which, in our case, is a triangular mesh. As we traverse the domain, the boundaries are drawn accordingly. Figure II.3 illustrates all possible states of an element composed of three cells.

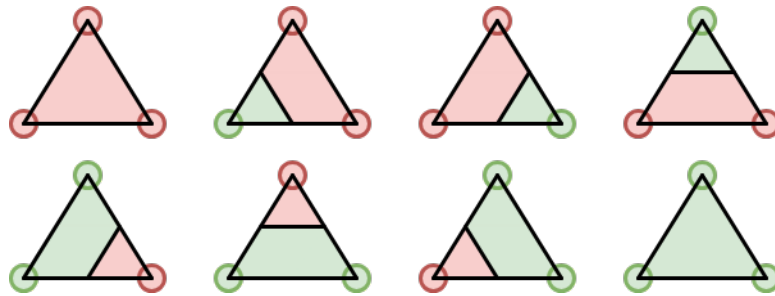


Figure II.3: Isolines, possible states of a triangular element

At the end of both steps—transformation and Marching Squares—we obtain the following schematic representation:

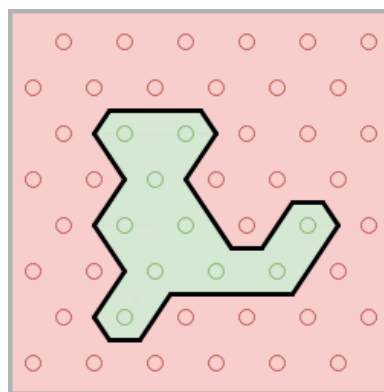


Figure II.4: Exemple scheme of map generated

In the simulator, maps are generated based on a seed and a random generator. The shape of each map is controlled by three parameters: the step size along the x-axis, the step size along the y-axis, and the overall map size.

This allows for a vast variety of map configurations. Figure II.5 illustrate some examples.

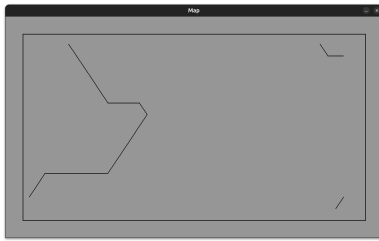
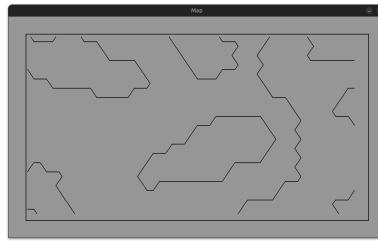
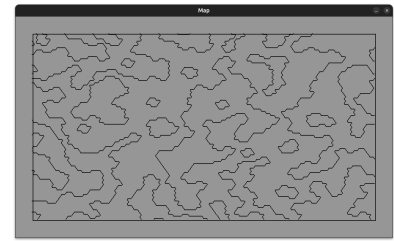
(a) $\delta x = 100 \text{ mu}$ (b) $\delta x = 40 \text{ mu}$ (c) $\delta x = 10 \text{ mu}$

Figure II.5: Different maps made with equilateral triangles

TO BE DONE

II.3 Robot implementation

II.3.1 Sensors

The robot is equipped with 2 sensors by default, an accelerometer that calculates acceleration in x and y axis and the rotation along z

Simultaneous Localization and Mapping (SLAM) is a key problem in robotics, enabling a robot to determine its position while simultaneously building a map of its environment. Various techniques have been developed to tackle this challenge. For instance, EKF-SLAM uses a Kalman filter to estimate both the localization and the map but faces scalability limitations in large environments. FastSLAM, which relies on a particle filter, enhances scalability by handling world features through multiple hypotheses.

Graph-based approaches, such as Graph-SLAM, are effective for large-scale optimization problems but require complex data management. Visual SLAM (V-SLAM) leverages cameras to estimate localization and construct maps, whereas LiDAR-based SLAM relies on laser sensors for precise depth measurements, making it particularly useful for outdoor environments. Dynamic SLAM variants manage environments with moving objects by excluding them from map updates.[17]

In this report, I used LiDAR-based SLAM,

III Partie 3

III.1 Finding the shortest path distance

To find the shortest path, I like to take inspiration from nature by simulating a wave propagating through a medium. This way, the shortest path naturally emerges as the one the wave follows.

The wave equation describes how information spreads at a certain speed, but challenges arise—how to model refraction, how to ensure the wave propagates at a constant speed. To tackle this, I used a cellular automaton once again. Made for propagating information, they are an interesting tool for this approach.[18]

Differently from the part **FIGURE ...**, I will use cellular automata as a computational tool for simulating phenomenon. Application were found in various domain from physics to biology. Lattice Gaz Cellular Automata (LGCA) for instance are used to simulate gaz fluid flows, it is the precursor of the lattice Boltzman Method (LBM)[19]

Base on the work of *Calvo Tavia* and *al.*[18], the process is describe below:

Given a lattice $\Lambda = \{(i, j) \in \mathbb{N}^2 : 1 \leq i, j \leq L\}$ with L the size of the lattice, we define the following sets:

- $A_t = \{(i, j) \in \Lambda : a_t(i, j) > 0\}$: the set of activated cells at time t
- \mathcal{B} : the set of obstacles
- $\Gamma \subset \Lambda$: the set of secondary wave sources
- E_t : the set of empty spaces
- $\mathcal{M}_{ij} = \{(k, l) \in \Lambda : \|(k - i, l - j)\|_\infty = 1\}$: the Moore neighborhood of a cell (i, j)

Each cell of the lattice carries 2 variables:

- $a_t(i, j)$: the state of the cell (i, j) at time t
- $z_t(i, j)$: the distance vector of the wave from the source to the cell (i, j) at time t , defined as:

$$z_t(i, j) = \left(\begin{array}{c} \text{Total number of steps taken to arrive at cell } (i, j) \\ \text{Number of diagonal steps among them} \end{array} \right)$$

To update z_t , we introduce the following variable that track the distance of the wave from the source to the cell (i, j) at time t :

$$r_{t_{ij}}(k, l) = \begin{cases} (0, 0) & \text{if } (i, j) \in E_t \text{ or } (k, l) \notin A_t \\ z(k, l) + (1, \mathbb{1}_{D_{ij}}(k, l)) & \text{otherwise} \end{cases}$$

where $\mathbb{1}_{D_{ij}}$ the diagonal function indicator, i.e. 1 if (k, l) is a diagonal neighbor of (i, j) and 0 otherwise.

$$D_{ij} = \{(k, l) \in \Lambda : |k - i| + |l - j| = 1\} \subset \mathcal{M}_{ij}$$

Note that $\|r_{t_{ij}}\|_2$ is the distance measurement from the wave source to the cell (i, j) .

Moreover, we define the set of cells in the Moore neighborhood that could be a source of activation for the cell (i, j) at time t as:

$$W_t = \{(k, l) \in M_{ij} : t < a_t(k, l) + \|r_{t_{ij}}\|_2 \leq t + 1\}$$

Finally, we define the pair (k, l) where the distance $\|r_{t_{ij}}\|_2$ is minimal if the set of potential source of activation for the cell (i, j) at time t is not empty, i.e. $W_t \neq \emptyset$:

$$(i_t^*, j_t^*) = \begin{cases} \operatorname{argmin}\{\|r_{t_{ij}}(k, l)\|_2 : (k, l) \in W_{t_{ij}}\} & \text{if } W_t \neq \emptyset \\ (i, j) & \text{otherwise} \end{cases}$$

After defining all the sets and variables above, the wave propagation is governed by the following rules:

At each iteration of time, we compute the two variables $a_t(i, j)$ and $z(i, j)$ for each cell $(i, j) \in \Lambda$ as follows:

$$a_t(i, j) = \begin{cases} t + 1 & \text{if } (i, j) \in \Gamma \text{ or } (M_{ij} \cap A_t) \neq \emptyset \\ a_t(i_t^*, j_t^*) & \text{otherwise} \end{cases}$$

$$z_t(i, j) = \begin{cases} z_t(i, j) & \text{if } (i, j) \notin E_t \setminus \Gamma \text{ or } W_t = \emptyset \\ r_t(i_t^*, j_t^*) & \text{otherwise} \end{cases}$$

This implementation does not allow to track the refraction of the waves, these are wrongly reflected by the obstacles. To prevent front breaking in wave propagation near obstacles, the concept of "additional secondary" wave sources is introduced, inspired by Huygens' principle, where selected cells on the obstacle boundary generate secondary waves. An algorithm determines these sources based on geometric conditions, ensuring correct wave propagation by maintaining the expected wavefront shape even when interacting with obstacles. This simple algorithm is describe in the work of *Calvo Tavia and al.*[18]. We will not go into the details of the algorithm here.

IV Partie 4

V Partie 5

Conclusion

Perspectives

ROS (Robot Operating System) implementation coupled with Rviz or Gazebo. Implementation on robot either to have both simulation and experimentation

References

- [1] S. M. LaValle, Planning Algorithms. Cambridge, U.K.: Cambridge University Press, 2006, available at <http://planning.cs.uiuc.edu/>.
- [2] N. Geographic, "Greenland's secret caves: Inside the hidden world of ice," 2025, consulté le 10 janvier 2025. [Online]. Available: <https://www.nationalgeographic.com/science/article/greenland-secret-caves-exploration>
- [3] H. t. Dang, "Underwater robots for karst and marine exploration : A study of redundant auvs," Ph.D. dissertation, 2021, thèse de doctorat dirigée par Lapierre, Lionel SYAM - Systèmes Automatiques et Micro-Électroniques Montpellier 2021. [Online]. Available: <http://www.theses.fr/2021MONT038>
- [4] P. Kambesis, "The importance of cave exploration to scientific research," Journal of Cave and Karst Studies, vol. 69, 04 2007.
- [5] L. Jean-Claude, Robot Motion Planning. Boston, MA: Springer US, 1991. [Online]. Available: <http://link.springer.com/10.1007/978-1-4615-4022-9>
- [6] B. Aneeta, S. Ekta, and D. Bhaskar, "Robot path planning using silhouette method," in 13th National Conference on Mechanisms and Machines, January 2008, pp. 12–13.
- [7] L.-P. Tomás and W. M. A., "An algorithm for planning collision-free paths among polyhedral obstacles," Communications of the ACM, 1979.
- [8] G. Santiago, M. Luis, A. Mohamed, and M. Fernando, "Path planning for mobile robot navigation using voronoi diagram and fast marching," in 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 2376–2381.
- [9] Z. David and L. Jean-Claude, "New heuristic algorithms for efficient hierarchical path planning," IEEE Transactions on Robotics and Automation, vol. 7, pp. 9–20, 1991. [Online]. Available: <https://api.semanticscholar.org/CorpusID:21438079>
- [10] K. K. and S. M., "An efficient motion-planning algorithm for a convex polygonal object in two-dimensional polygonal space," Discrete and Computational Geometry, vol. 5, no. 1, pp. 43–76, 1990. [Online]. Available: <http://eudml.org/doc/131106>
- [11] A. Francis, B. Jean-Daniel, and F. Bernard, "A practical exact motion planning algorithm for polygonal objects amidst polygonal obstacles," in Proceedings of the 1988 IEEE International Conference on Robotics and Automation, vol. 3, 1988, pp. 1656–1661. [Online]. Available: <https://api.semanticscholar.org/CorpusID:37779065>
- [12] K. Y. and B. J., "Potential field methods and their inherent limitations for mobile robot navigation," in Proceedings of the 1991 IEEE International Conference on Robotics and Automation, vol. 2, 1991, pp. 1398–1404.
- [13] N. M. Amato and Y. Wu, "A randomized roadmap method for path and manipulation planning," in Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). IEEE, 1996, pp. 113–120.
- [14] D. Hsu, R. Kindel, J.-C. Latombe, and S. Rock, "Randomized kinodynamic motion planning with moving obstacles," The International Journal of Robotics Research, vol. 21, no. 3, pp. 233–255, 2002.
- [15] C. Nissoux, T. Siméon, and J.-P. Laumond, "Visibility based probabilistic roadmaps," Advanced Robotics, vol. 13, no. 2, pp. 223–244, 1999.
- [16] A. Gasparetto, P. Boscariol, A. Lanzutti, and R. Vidoni, "Path planning and trajectory planning algorithms: A general overview," Mechanisms and Machine Science, vol. 29, pp. 3–27, 03 2015.
- [17] S. Ding, T. Zhang, M. Lei, H. Chai, and F. Jia, "Robust visual-based localization and mapping for underwater vehicles: A survey," Ocean Engineering, vol. 312, p. 119274, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S002980182402612X>
- [18] C. Calvo Tapia, J. Villacorta-Atienza, V. Mironov, V. Gallego, and V. Makarov, "Waves in isotropic totalistic cellular automata: Application to real-time robot navigation," Advances in Complex Systems, vol. 19, p. 1650012, 12 2016.
- [19] S. Chen and G. D. Doolen, "Lattice Boltzmann Method for Fluid Flows," Annual Review of Fluid Mechanics, vol. 30, pp. 329–364, Jan. 1998.

Annexes