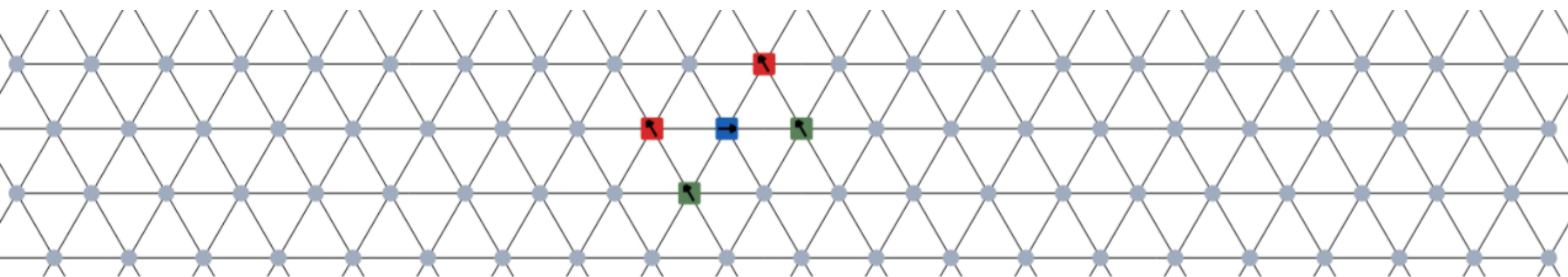
Research proposal

# Exploration of triangular grids by a swarm of luminous robots.

M2 ICS

**Student:** Maurin Gilles
**Supervised by:** Dr. Anaïs Durand

Academic year 2025-2026



# Contents

# 1. Presentation

## 1.1 Introduction

Swarm robotics is the subfield of robotics that aims to solve complexe tasks by robots that are not powerful because of their individual capabilities, but their number and coordination. One of the most obvious tasks we could use such robots for is exploration, and a natural abstraction for this problem is to make it discrete by restraining exploration to a finite grid. In our project, we consider robots with minimal capabilities: they can see around them, move, emit a colour, and change their colour. With no memory nor centralised control, we aim to define sets of rules that will let them autonomously organise to explore the entire grid indefinitely.

## 1.2 Modelisation

By "modelising" we mean that our research is about finding theorical strategies for this exploration rather than putting concrete robots at work. However, the results we aim to reach could benefit real-world applications, as exploration using swarm of robots with minimal capabilities is a strategy with high potential in nano-robotics or extreme conditions for example.

In our modelisation, the grid covering space is a finite graph $G = (V, E)$ whose vertices are to be visited by robots that move along its edges. We discretise time in a way such that at any time $t$ every robot $r_i$ is *visiting* a vertex $u_i \in V$. Robots can choose to move along an edge at each step of time (thereby going to a node of distance one), so at $t + 1$ every robot $r_i$ is visiting a vertex $u'_i \in V$ such that $\forall r_i : 0 \leq d(u_i, u'_i) \leq 1$, where $d(.,.)$ is the distance between two nodes. We are here interested in the particular problem of *perpetual* exploration, where algorithms should ensure that every vertex would be visited an infinite amount of times if it was ran for an infinite period of time. So for a graph to explore $G = (V, E)$, we want for all $u \in V$, for any time $t$, there exists a time $t' \geq t$ such that a robot is visiting $u$ at time $t'$.

For generalisation purposes, we seek algorithms that solves the perpetual exploration problem for families of graphs. Intuitively, a more "general" family may require more capabilities from the robots, while fewer and more minimal robots can be sufficient for families matching more precise properties. For example, previous work in the field was interested in regular *rectangular* grids, made of $\mathcal{L} > 2 \in \mathbb{N}$ lines and $\mathcal{C} > 2 \in \mathbb{N}$ columns, and defined by the graph $G = (V, E)$ with

$$V = \{(i, j) : i \in [0, \mathcal{C} - 1], j \in [0, \mathcal{L} - 1]\}$$

and

$$E = \{\{(i,j),(k,l)\} : (i,j) \in V \wedge (k,l) \in V \wedge |i-k| + |j-l| = 1\}$$

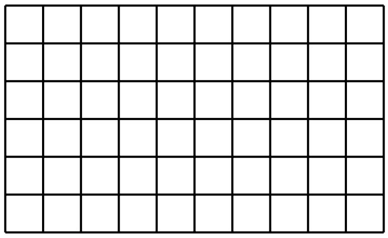


Figure 1.1: Rectangular grid

In our case, we are interested in triangular grids, that can be informally imagined as grids made of triangles instead of squares. Multiple families of triangular grids can be isolated depending on the "global shape" of the grids. For example, the family of triangular grids shaping a global triangle is defined by a size $\mathcal{N} > 0 \in \mathbb{N}$ and the graph $G = (V,E)$ with

$$V = \{(i,j) : i \in [0,\mathcal{N}], j \in [0,i]\}$$

and

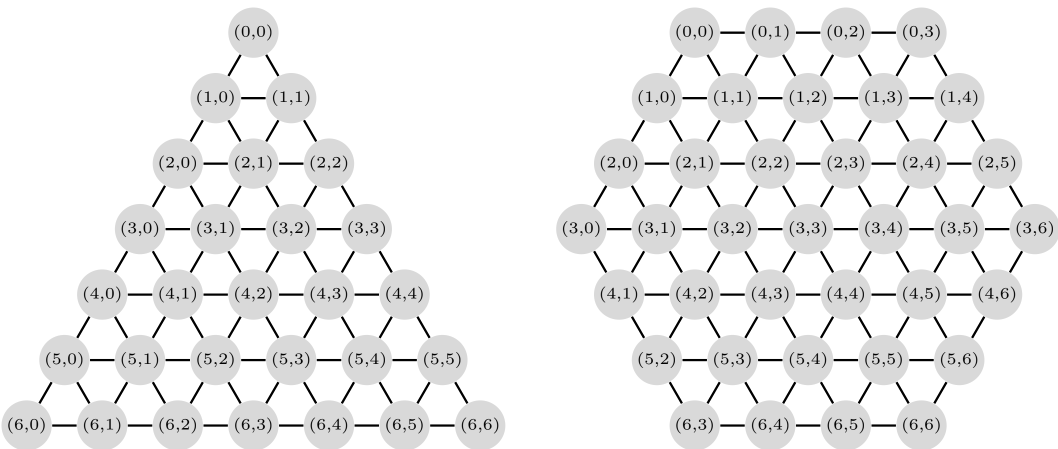
$$E = \{\{(i,j),(k,l)\} : (i,j) \in V \wedge (k,l) \in V \wedge max(|i-k|,|j-l|) \leq 1 \wedge i+j \neq k+l\}$$

We can also define triangular grids shaping a global hexagon of size $\mathcal{N} > 0 \in 2\mathbb{N}$ defined by the same set of rules for the edges applied to the set of vertices

$$V = \{(i,j) : i \in [0,\mathcal{N}], j \in [max(0, i-(\mathcal{N}/2)), min(i+(\mathcal{N}/2),\mathcal{N})]\}$$

Eventually, we call Rectangle Enclosed Triangular Grid (RETG) grids of size $\mathcal{N} \times \mathcal{M}$ with $\mathcal{N} > 2 \in \mathbb{N}$ and $\mathcal{M} > 2 \in \mathbb{N}$ where the same set of rules for the edges is applied to the set of vertices

$$V = \{(i,j) : i \in [0,\mathcal{N}], j \in [\lceil i/2 \rceil, \lceil i/2 \rceil + \mathcal{M}]\}$$




(a) Global triangle
$\mathcal{N} = 6$

(b) Global hexagon
$\mathcal{N} = 6$

Figure 1.2: Types of triangular grid.

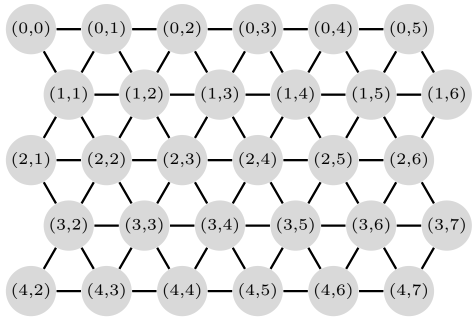


Figure 1.3: RETG
$\mathcal{N} = 4, \mathcal{M} = 5$

Regarding the capabilities of the robots, they are luminous, deaf-mute, myopic and opaque. "Luminous" refers to the fact that all robots have a colour. Robots know their own colour, they can change it, and they can recognise other robots' colours. "Deaf-mute" clarifies that colours are the only mean of communication robots can use. "Myopic" suggests that robots have a limited visibility range. In our graph context, it corresponds to a maximum length $\phi$ such that a robot only knows the state of the nodes it can reach by a path of maximum length $\phi$. Finally, robots are said "opaque" because other robots cannot "see" through them. In other words, the previous definition of the visibility range should be completed as follows: a robot only knows the state of the nodes it can reach by a path of maximum length $\phi$ made of empty nodes.

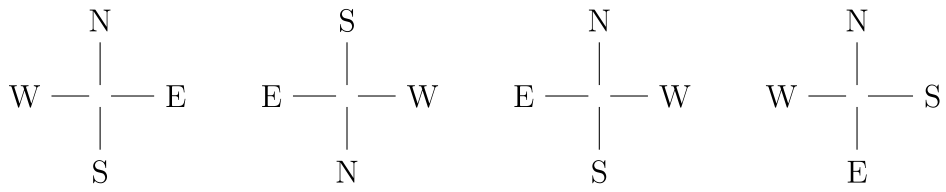
Formally, we can define the set of all the nodes a robot visiting a node $r$ can see as follows:

$$\{(i,j) \in V, d(r,(i,j)) \leq \phi,$$
$$\exists (l_k, l_{k+1})_{k=1...K-1}, K \leq \phi, (l_k, l_{k+1}) \in E, \forall k,$$
$$l_1 = r, l_K = (i,j), l_k \in V \wedge \sigma(l_k) = \perp, \forall k = 2...K-1\}$$

Where $d(r,(i,j))$ is the distance between the nodes $r$ and $(i,j)$, and $\sigma(l_k)$ is the state of the node $l_k$: either the colour of the robot visiting the node, or $\perp$ if the node is empty.

Another crucial parameter influencing robots' capabilities is their "sense of direction". We distinguish three degrees of sense of direction depending on whether or not robots have a global compass and a common chirality. Very informally, a global compass allows robots to "distinguish the North and the South", and a common chirality only allows them to "distinguish their left from their right", where "left" and "right" have the same meaning for all robots.
More precisely, if we represent what the robot sees in a 2D plan centered on the robot, the operations making this representation unchanged for the robot are none if it has a compass, rotations if it has chirality, and reflections with respect to an axis passing through the robot in addition to rotations without chirality.

To better represent the idea: on the following figure, robots with a global compass are able to make the difference between cases $a$, $b$, $c$ and $d$; robots without global compass but agreeing on a common chirality could not make the difference between cases $a$ and $b$, but would recognise cases $c$ and $d$; and robots without common chirality would only be able to distinguish case $d$ from the others.




(a) Case *a* (b) Case *b* (c) Case *c* (d) Case *d*

Figure 1.4: Distinguishable cases depending on the level of sense of direction.

Robots evolve on the grid by steps of Look-Compute-Move (LCM) cycles. In a single such cycle, a robot captures a representation of his surrounding with respect to its visibility range (Look), it uses this representation to decide where it should go and what colour it should be next (Compute), and it executes this decision (Move). Robots have no persistent memory, so the information captures during the Look phase is all they can use to compute their next move.
In the typical configuration, we consider robots operating synchronously, so at any moment in time they have executed the same number of cycles and are ready to execute the same phase (Look, Compute, or Move) next. However, the asynchronous configuration can also be studied. In this case, the time a robot takes to execute each phase of the cycle and the number of cycles a robot remains inactive between two cycles is unpredictable (chosen by an adversary in theorical work such as the present one) yet unbounded.

A last constraint is generally taken into consideration for algorithms solving the perpetual exploration problem: exclusivity. It forbids two robots visiting the same node at the same time or moving along the same edge (it thereby prevents two robots swapping positions). This constraint is especially relevant when thinking about real-world applications, because it prevents two robots from colliding.

The robots are autonomous and identical: they cannot be manipulated by other means that the embedded set of rules, and all the robots obey to the same set of rules. Therefore, to find an algorithm solving the perpetual exploration problem can be reduced to finding a set of rules that lets robots explore the grid when applied. We assume that one has minimal control on the initial position: one can choose how the robots are located with resepect to each other, but not their global orientation nor their position inside the grid (for example one cannot choose to start from a specific border or corner). Thanks to this helpful relaxation, a set of rules solving the exploration problem when starting from a chosen locally-defined configuration is a sufficient result.

## 1.3 Representation

Since robots have no sense of the global grid, it is important to think rules at a local level. We therefore introduce the *views*, that define what a robot sees and are logically centered on the robot.

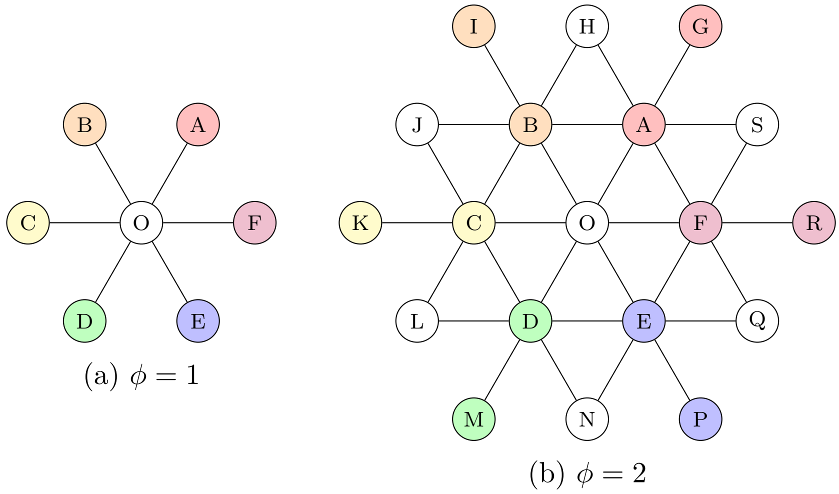



Figure 1.5: Robot views on triangular grids.

Because of the absence of global compass, a view actually corresponds to 6 view corresponding of rotations of $(k\pi)/3$ radians, $k = 0...5$.

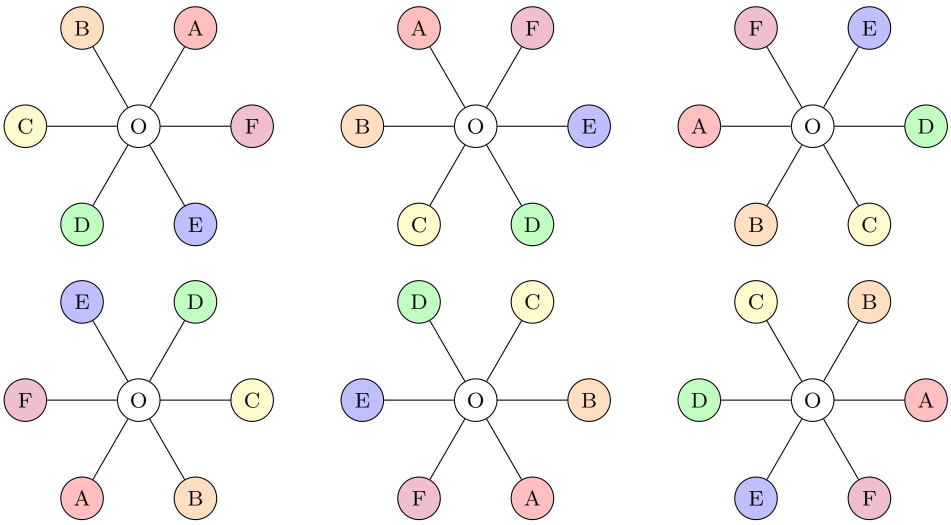



Figure 1.6: Views unchanged by rotation.

Without common chirality, a view covers 6 additional views equivalent by symetry. With the labels we use in our examples, they correspond to the views where nodes are labelled clockwise instead of counterclockwise.

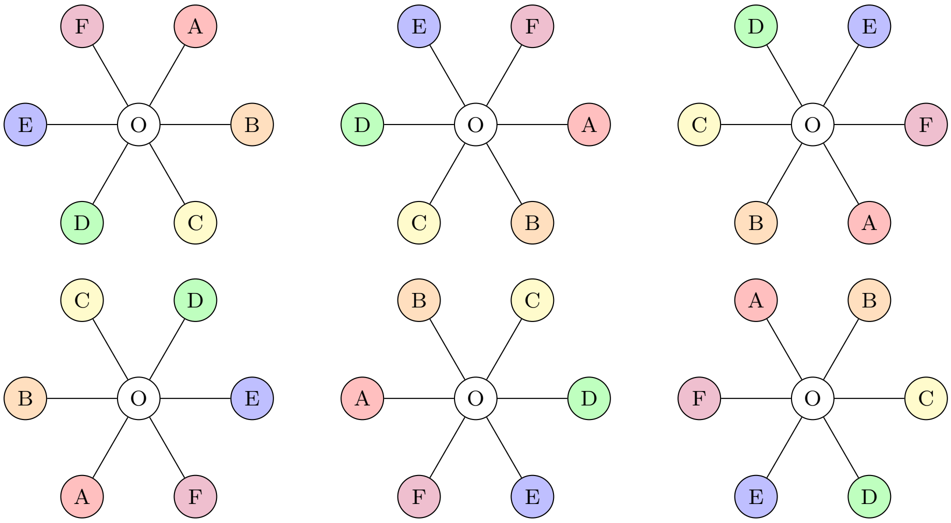
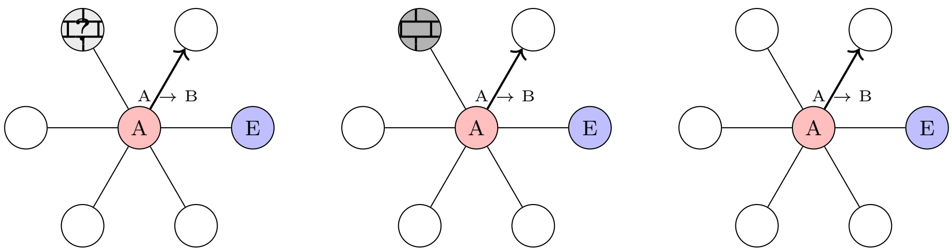



Figure 1.7: Views unchanged by symetry.

In our representations, each colour a robot can have will be associated with a letter, and the algorithms will be described refering to these letters rather than actual colours. An algorithm is also likely to require special behaviours when robots are near the borders of the grid. This will be handled by creating rules for views including the border as "wall" nodes. Note that such nodes are used only to represent views, and do not correspond to vertices in the graph giving the formal definition of a grid.

Sometimes, being near the border may not influence the expected behaviour of the robots in some cases. If this happens, to avoid creating multiple equivalent rules, we will add another special node to our representations (a wall of a lighter gray colour and an interogation mark), that can interchangeably be a border or an empty node.

For example, rule I of figure 1.8 covers both rules II and III, and corresponds therefore to 12 different scenarios in the global oriented grid (24 without common chirality).




(a) Rule I (b) Rule II (c) Rule III

Figure 1.8: Views including walls.

A rule is composed of a view, a node within this view to go next, and a colour to change for. In the previous example on figure 1.8 for example, the rule could be verbosely described as follows: *"When the robot's colour is A, and it has a robot of colour E among its neighbours with an empty node on the left and three empty nodes on the right, then the robot should move towards the left of the E-coloured robot and switch to colour B."*.

# 2. State of the art

## 2.1 Other types of grids

The problem of perpetual grid exploration by a swarm of robots has already been adressed in various settings. The closest to the work we plan to conduct deals with synchronous robots evolving on rectangular grids, with common chirality [1] or not [2]. Similar scenarios have been studied for asynchronous models in [3]. Related work also deals with other shapes of grids such as torus [5] and 3D rectangular grids [4].

The work on 2D rectangular grids is likely to provide helpful insights for the problem on triangular grids by two means. Firstly, it demonstrates impossibility results for some parameters with proofs that can – for some of them – easily be transposed to triangular grids. Secondly, it provides optimal algorithms based on strategies that can inspire algorithms to solve the problem on triangular grids. This is the topic of section 2.3.

Table 2.1 summarises existing impossibility results and optimal algorithms for rectangular grids with and without common chirality, depending on three parameters: the range of visibility, the number of robots, and the number of colours robots can have. Note that impossibility results with chirality naturally still stand without chirality and are therefore not repeated.

| | Visibility | Robots | Colours | |
|---|---|---|---|---|
| With Chirality | $< \infty$ | 1 | $< \infty$ | *Impossible [1]* |
| | $< \infty$ | 2 | 1 | *Impossible [1]* |
| | 1 | 2 | 2 | *Impossible [1]* |
| | 1 | 3 | 1 | *Impossible [1]* |
| | 1 | 2 | 3 | *Possible [1]* |
| | 2 | 2 | 2 | *Possible [1]* |
| | 2 | 3 | 1 | *Possible [1]* |
| Without Chirality | 1 | 2 | $< \infty$ | *Impossible [2]* |
| | 1 | 3 | 3 | *Possible [2]* |
| | 2 | 5 | 1 | *Possible [2]* |

Table 2.1: Impossible and optimal parameters for rectangular grids.

## 2.2 Triangular grids

To our knowledge, perpetual triangular grid exploration has only been studied by Das et al. in [6], in a synchronous setting and for the specific case of rectangle-enclosed triangular grids. Also, a gap remains between the parameters they prove unsufficient and the parameters used in the algorithms they propose, so optimality might not have been reached yet.

| | Visibility | Robots | Colours | |
|---|---|---|---|---|
| With Chirality | $< \infty$ | 1 | $< \infty$ | *Impossible [6]* |
| | 2 | 2 | 1 | *Impossible [6]* |
| | 0 | 2 | $< \infty$ | *Impossible [6]* |
| | 2 | 2 | 2 | *Possible [6]* |
| Without Chirality | 1 | 2 | 3 | *Possible [6]* |

Table 2.2: Impossible and optimal parameters for RETGs.

## 2.3 Strategies

Most algorithms proposed for rectangular grids share the same core strategy. The robots "sweep" the grid, as in the examples of figure 2.1. When the robots finish sweeping in one direction, depending on the algorithm and the configuration, they either continue their exploration applying a similar pattern in the opposite direction, or they rotate along the corner and continue their exploration in a perpendicular direction.

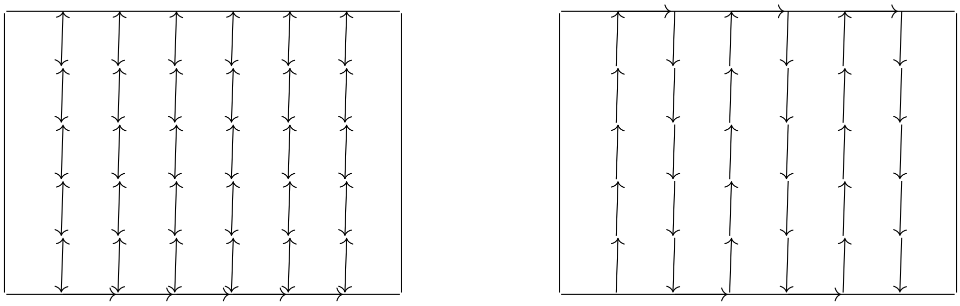


Figure 2.1: Two sweeping strategies.

To implement a such strategy, an algorithm must be able to handle:

- Two configurations for when the robots move in straight line, to be able to distinguish two directions.
- A different behaviour for each of these configurations when the robots reach a wall, that must lead to the robots forming the other configuration and going the opposite direction.
- A special behaviour when an angle is reached, to start a new "phase" of sweeping.

In algorithms $Vone^2_3$ and $Vtwo^2_2$ from [1], only two robots are used, with respective roles of "Leader" and "Follower". The robots are thereby gathered thanks to the follower always moving towards the leader, which moves in front. In $Vone^2_3$, robots have a visibility range of 1, so the two "straight line" configurations are distinguished thanks to two colours for the leader. In $Vtwo^2_2$, only two colours are used but the visibility range is 2, so the "straight line" configurations are distinguished by keeping or not one empty node between the follower and the leader.

In $Vtwo_3^1$, only one colour is used so robots' role is are fuzzier, but thee "straight line" configurations are distinguishable thanks to two shapes (called "$<$" and "L" in the paper) made of three robots.

Without chirality, more robots or colours are necessary, but the core idea behind the algorithms remain identical [2]. The two algorithms proposed in [6] for triangular grids with and without chirality both use a strategy made of "Leader" and "Follower" roles.

# 3. Work planning

## 3.1 Goals and intuitions

Much is yet to be done for triangular grids, so the plan for our project will be to start with the easiest cases and explore further if the speed of progression allows it. The first goal is to find optimal algortihms (and prove them optimal) for all types of triangular grids with synchronous robots and common chirality. Then, we should seek optimal algorithms without chirality. Eventually, if time allows it, we may start studying the asynchronous case.

So far, no results exist to our knowledge for the sets of parameters presented in table 3.1. The case with visibility 1, 2 robots and 2 colours and the case with visibility 1, 3 robots and 1 colour deserve a particular attention because if they are reachable, they would be an improvement from the current known optimals for rectangular and rectangle-enclosed triangular grids.

| | Visibility | Robots | Colours | In rectangular grids |
|---|---|---|---|---|
| With Chirality | **1** | **2** | **2** | *Impossible* |
| | **1** | **3** | **1** | *Impossible* |
| | 1 | 2 | 3 | *Possible* |
| | 2 | 3 | 1 | *Possible* |
| | 1 | 3 | 2 | *?* |
| Without Chirality | **1** | **2** | **2** | *Impossible* |
| | **1** | **3** | **1** | *Impossible* |
| | 1 | 3 | 2 | *?* |

Table 3.1: Sets of parameters to study.

Also, since the known results for triangular grids apply only to RETGs, we have to check wether the impossibility results apply to other types of triangular grids, or if some of them can be explored with fewer parameters, or on the contrary require more of them. It should also been checked whether a unique algorithm manages to solve the perpetual exploration problem for every type of triangular grid, or a different algorithms must be found for each global shape. In this second case, an algorithm with more parameters but solving more types of grids can be another interesting perspective to study.

## 3.2 Found algorithm

Our first intuitions lead us to an algorithm that appears to solve the perpetual exploration of triangular grids shaping a global triangle with two synchronous robots with visibility range 1, 2 colours and common chirality; however, its efficiency has not been rigourously proven yet. This results is particularly encouraging since it reaches a sobriety in parameters that outclasses the feasible optimal in rectangular grids and the currently known optimal for RETGs. However, this algorithm cannot be used for other types of triangular grids, and we tend to think that similar results could not be reached for RETGs and grids shaping global hexagons.

The algorithm is highly inspired by $Vone_3^2$ from [1], but benefits from the fact that walls in a triangular grid are more easily distinguishable than they are in rectangular grids. In rectangular grids, robots arrive perpendicularly to walls, so the only way to know if they had to turn right or left was by saving this information with a third colour or a changing distance between the robots. On the other hand, in triangular grids will reach the walls at an angle that provides enough information to react accordingly. Special rules in the angles ensure that the "sweeping" phases alternate in the three directions, and the perpetual exploration is therby solved.

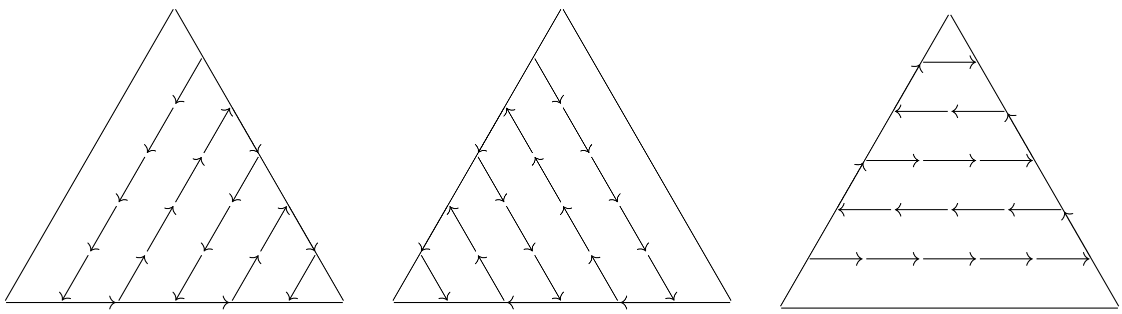


Figure 3.1: Rotating directions in a global triangle grid.

## 3.3 The simulator

The goal when we find an algorithm is to prove rigourously that it solves the problem, usually by testing it empirically on small grids and generalising to larger ones belonging to the same family. However, this can be a tedious work to do by hand, especially given the necessity to test multiples sizes, initial configurations, and in our case different shapes of grids. Therefore, we will take great advantage of a helpful tool designed Anaïs Durand that easily allows to change parameters of the grid and robots, initial configurations, and save and load sets of rules.

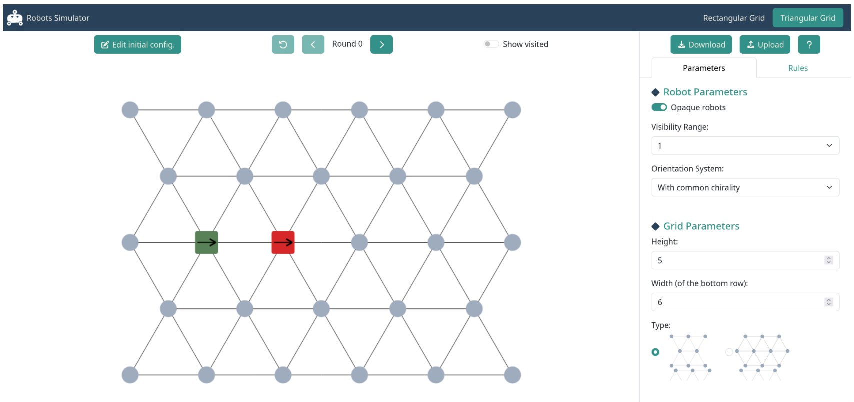



Figure 3.2: Simulator

# Bibliography


[1] Quentin Bramas, Pascal Lafourcade, and Stéphane Devismes. “Optimal exclusive perpetual grid exploration by luminous myopic opaque robots with common chirality”. In: *Proceedings of the 22nd International Conference on Distributed Computing and Networking*. 2021, pp. 76–85.

[2] Arthur Rauch et al. “Optimal exclusive perpetual grid exploration by luminous myopic robots without common chirality”. In: *International Conference on Networked Systems*. Springer. 2021, pp. 95–110.

[3] Quentin Bramas et al. “Optimal Asynchronous Perpetual Grid Exploration”. In: *International Symposium on Stabilizing, Safety, and Security of Distributed Systems*. Springer. 2024, pp. 89–105.

[4] Quentin Bramas et al. “Beedroids: How Luminous Autonomous Swarms of UAVs Can Save the World?” In: *FUN: Conference on Fun with Algorithms*. 2022.

[5] Omar Darwich et al. “Perpetual torus exploration by myopic luminous robots”. In: *Theoretical Computer Science* 976 (2023), p. 114143.

[6] Raja Das et al. “Rectangle Enclosed Triangular Grid Exploration with Myopic Luminous Robots”. In: *Proceedings of the 26th International Conference on Distributed Computing and Networking*. 2025, pp. 249–253.