Perpetual Torus Exploration by Myopic Luminous Robots⋆

Omar Darwich[a], Ahmet-Sefa Ulucan[c], Quentin Bramas[b], Anissa Lamani[b], Anaïs Durand[d], Pascal Lafourcade[d]

[a] *University of Toulouse, CNRS, LAAS,France*
[b] *University of Strasbourg, CNRS, ICUBE, France*
[c] *University of Strasbourg, France*
[d] *University Clermont Auvergne, CNRS UMR 6158, LIMOS, France*

Abstract

We study perpetual torus exploration for swarms of autonomous, anonymous, uniform, and luminous robots. We consider robots with only few capabilities. They have a finite limited vision (myopic), they can only see robots at distance one or two. We show that the problem is impossible with only two luminous robots and also with three oblivious robots (without light). We then address the problem assuming luminous robots (resp. oblivious) with visibility range one (resp. two). We design optimal solutions with respect to both the number of robots and colors when robots share a common chirality and have, respectively, vision one and two. We also present an optimal solution with respect to the number of robots when they are endowed with vision one and share no common chirality. Finally, we propose a solution for the case in which robots are oblivious, have vision two, and no common chirality that uses one additional robot.

Keywords: Perpetual Exploration, Luminous robots, Torus-shaped network.

1. Introduction

In the last decade, swarm robotics has drawn a lot of attention. Inspired by natural systems, a lot of investigations focused on how to reproduce autonomous behaviors observed in nature within artificial systems. Given a collection of autonomous mobile entities called robots, the main focus is to determine the minimum hypothesis in order for the robots to solve a given task. Robots can evolve either on a continuous 2D plane on which they can freely move or on a discrete universe, generally represented by a graph, where nodes indicate possible locations of the robots and the edges the possibility for the robots to move from one node to another.

In this paper, we assume that the mobile robots are autonomous (*i.e.*, there is no central authority to coordinate their move), anonymous (*i.e.*, they have no identity),

⋆Preliminary results appearing in this paper were presented at SSS2022 [7].

Email addresses: omar.darwich@laas.fr (Omar Darwich), ahmet-sefa.ulucan@etu.unistra.fr (Ahmet-Sefa Ulucan), bramas@unistra.fr (Quentin Bramas), alamani@unistra.fr (Anissa Lamani), anais.durand@uca.fr (Anaïs Durand), pascal.lafourcade@uca.fr (Pascal Lafourcade)

uniform (*i.e.*, they all execute the same algorithm), and luminous (*i.e.*, they are endowed with lights of different colors). Moreover, they cannot communicate directly but are endowed with visibility sensors allowing them to sense their environment within a certain distance called visibility range. We assume myopic robots that can only sense at small distances. Robots operate in the well-known LCM model. That is, they operate in cycles that comprise three phases: Look, Compute, and Move. During the first phase (Look), robots take a snapshot of their environment using their visibility sensors. In the second phase (Compute), based on the taken snapshot, they first decide whether to move or remain idle and then whether they change their color. If they decide to move, they compute a neighboring destination. Similarly, they compute a new color if they decide to change it. Finally, in the last phase (Move), they move to the computed destination (if any) and change their color (if they decided to). We consider the fully synchronous model (FSYNC) in which all robots execute the LCM cycle synchronously and atomically.

In the following, we investigate the case in which the robots have to solve the perpetual exploration problem. In this problem, robots evolve in a discrete universe and have to ensure that each location (node) is visited by at least one robot infinitely often. We are interested in torus-shaped networks and focus on optimal exclusive solutions with respect to both the visibility range and the number of robots. Exclusiveness adds an additional constraint on robots' behavior as they can neither occupy the same node simultaneously nor traverse the same edge at the same time.

2. Related work

The exploration problem is considered one of the benchmarking tasks when it comes to robots evolving on graphs. Various topologies have been considered: lines [15], rings [1, 10, 13, 16, 17], tori [12], grids [2, 4, 5, 11], cuboids [3], and trees [14]. Two variants of the problem have been investigated: (i) the perpetual exploration problem [1, 2, 3, 18], considered in this paper, which requires the robots to visit each node of the graph infinitely often and (ii) the terminating exploration problem [14, 15, 16, 13, 10, 12, 11] which requires the robots to visit each node of the graph at least once and then stop moving.

Most of the investigations consider robots with unlimited visibility range allowing them to observe every node of the system [14, 1, 2, 15, 16, 13, 12, 11]. Robots are in this case oblivious (*i.e.*, they cannot remember past actions) and have to solve the terminating exploration problem. Myopic robots have also been considered in both variants of the problem [17, 10, 4, 6, 9]. When it comes to the perpetual exploration problem, an additional assumption has an impact on the feasibility of the task and the optimality of the proposed solutions. This assumption endows the robots with a common chirality. In fact, chirality is usually assumed when robots evolve in the continuous 2D Euclidean plan but some investigations have also considered it recently in the discrete universe. On finite grids, it has been shown that two (resp. three) synchronous robots with three colors (resp. one color) are sufficient to solve the problem when robots have visibility one and share a common chirality [6]. The case in which robots have no common chirality was investigated in [18]. It was proven that the problem is not solvable with only two robots having a finite number of colors and a finite visibility range. An optimal

Finite grid				
Chirality	Visibility	# Robots	# Colors	Algorithm
yes	finite	1	finite	Impossible [6]
yes	finite	2	1	Impossible [6]
no	1	2	finite	Impossible [18]
yes	1	2	3	[6]
yes	2	2	2	[6]
yes	2	3	1	[6]
no	1	3	3	[18]
no	2	5	1	[18]
Infinite grid				
no	1	finite	1	Impossible [4]

Table 1: Summary of previous results in the finite grid and an impossibility result in the infinite grid that can be extended to the torus case.

solution is also presented using only three robots having a visibility range one, using only three colors. The case in which robots are oblivious and have a visibility range 2 was solved using five robots. Table 1 summaries the previous results on finite grids.

In the case of infinite grids, assuming robots with visibility range one and few colors (O(1)), five (resp. six) synchronous robots are necessary and sufficient to solve the problem with (resp. without) the common chirality assumption [4, 5]. In particular, it has been shown that it is impossible for a finite number of robots having a single light color and a visibility range 1 to travel an arbitrary distance [4], which directly implies the same impossibility results in our model (see Table 1). Finally, in the case of cuboids, it has been shown in [3] that three synchronous robots with a common chirality endowed with five colors are necessary and sufficient to solve the perpetual exploration problem.

Contributions. We first present two impossibility results in the torus: we start by showing that the perpetual torus exploration problem is not solvable with only two robots if the number of colors is finite and their visibility range is limited. This impossibility result has some similarity with the impossibility results in infinite grids, however, the main challenge was to account for the possibility for robots to explore the entire torus by always moving in the same direction. The impossibility comes from the fact that this direction follows a vector with integer values, which is shown not to be sufficient in tori whose size is a multiple of the vector component. We then show that three oblivious robots (*i.e.*, robots with a single light color) are not sufficient to solve the perpetual exploration problem (PTE), using the previous results and the fact that two oblivious robots cannot travel an arbitrary distance.

Next, we focus on the case in which robots have visibility range one and propose two solutions: $\mathcal{A}_1$ which is optimal with respect to both the number of robots and the number of colors when robots share a common chirality and $\mathcal{A}_2$ which remains optimal with respect to the number of robots for the case in which robots are completely disoriented (they do not share a common chirality). These algorithms explore the torus

Chirality	Visibility	# Robots	# Colors	Algorithm
yes	finite	2	finite	Impossible (Thm. 1)
yes	finite	3	1	Impossible (Thm. 2)
yes	1	3	2	Algorithm $\mathcal{A}_1$
no	1	3	3	Algorithm $\mathcal{A}_2$
yes	2	4	1	Algorithm $\mathcal{A}_3$
no	2	5	1	Algorithm $\mathcal{A}_4$

Table 2: Summary of our results.

row by row, in a way that is similar to the ones used in [6] for the case of finite grids. However, the lack of boundary makes the solutions different: $\mathcal{A}_1$ requires one more robot compared to the best-known algorithm in finite grids and $\mathcal{A}_2$ works with the same number of robots but using a different technic.

Then, we address the case in which robots have visibility range two and propose again two solutions: $\mathcal{A}_3$ which is optimal with respect to both the number of robots and the number of colors when robots share a common chirality and $\mathcal{A}_4$ for the case in which robots are completely disoriented. Table 2 summarizes our contribution. This paper is an extension of the conference paper [7], where we have only considered the case with a common chirality. In this paper, we have new algorithms where the robots do not share a common chirality.

3. Model

We consider a set of $n > 0$ robots located on a *torus*. A graph $G = (V, E)$ is a $(\mathcal{C}, \mathcal{R})$-torus (or torus for short) if $|V| = \mathcal{C} \times \mathcal{R}$ and for any $v_{(i,j)} \in V$; $i \in [0, \mathcal{C} - 1]$, $j \in [0, \mathcal{R} - 1]$:

- $\{v_{(i,j)}, v_{((i+1) \bmod \mathcal{C}, j)}\} \in E$, and
- $\{v_{(i,j)}, v_{(i,(j+1) \bmod \mathcal{R})}\} \in E$.

The order of the nodes of G forms a coordinate system. For example, node $v_{(i,j)}$ is at coordinate (i, j), or, the node is at column i and row j. For simplicity, we note node (i, j) instead of $v_{(i,j)}$. This order/coordinate is used for the analysis only, *i.e.*, robots cannot access it.

At each time instant called a *round*, the robots synchronously perform a *Look-Compute-Move* cycle. In the *Look* phase, a robot gets a snapshot of the subgraph induced by the nodes within distance $\Phi \in \mathbb{N}^*$ from its position. Φ is called the *visibility range* of the robots. The snapshot is not oriented in any way as the robots do not agree on a common North. However, it is implicitly ego-centered since the robot that performs a Look phase is located at the center of the subgraph in the obtained snapshot. Robots agree on a common chirality. Then, each robot *computes* a destination (either Up, Left, Down, Right or Idle) based only on the snapshot it received. Finally, it *moves* towards its computed destination. We also assume that robots are *opaque*, *i.e.*, they obstruct visibility in such a way that if three robots are aligned, the two extremities cannot see each other. We forbid any two robots to occupy the same node simultaneously. A node is *occupied* when a robot is located at this node, otherwise it is *empty*.

Robots may have *lights* with different colors that can be seen by robots within distance Φ from them. We denote by Cl the set of all possible colors. For simplicity, we assume that all tore has dimensions $\mathcal{C} \times \mathcal{R}$ where $\mathcal{C}, \mathcal{R} \geq n\Phi + 1$.

The *state* of a node is either the color of the light of the robot located at this node, if it is occupied, or $\perp$ otherwise. In the Look phase, the snapshot includes the state of the nodes (within distance Φ, including its current node). During the Compute phase, a robot may decide to change the color of its light.

In all our algorithms, we also prevent any two robots from traversing the same edge simultaneously. Since we already forbid them to occupy the same position simultaneously, this means that we additionally prevent robots from swapping their position. Algorithms verifying this property are said to be *exclusive*. However, to be as general as possible, we do not make this additional assumption in our impossibility results.

In the following, we borrow some of the definitions already presented in [18].

Configurations

A *configuration* C in a torus $G(V, E)$ is a set of pairs (p, c), where $p \in V$ is an occupied node and $c \in Cl$ is the color of the robot located at p. A node p is empty if and only if $\forall c$, $(p, c) \notin C$. We sometimes just write the set of occupied nodes when the colors are clear from the context.

Views

We denote by G_r the *globally oriented view* centered at the robot r, *i.e.*, the subset of the configuration containing the states of the nodes at distance at most Φ from r, translated so that the coordinates of r is $(0, 0)$. We use this globally-oriented view in our analysis to describe the movements of the robots: when we say "the robot moves Up", it is according to the globally oriented view. However, since robots do not agree on a common North, they have no access to the globally oriented view. When a robot looks at its surroundings, it instead obtains a snapshot. To model this, we assume that the *local view* acquired by a robot r in the Look phase is the result of an arbitrary *indistinguishable transformation* on G_r. The set $\mathcal{IT}$ of indistinguishable transformations contains the rotations of angle 0 (to have the identity), $\pi/2$, π and $3\pi/2$, centered at r. If the robots do not agree on a common chirality, then $\mathcal{IT}$ also contains a reflection and its combinations with the rotations. Moreover, since robots may obstruct visibility, the function that removes the state of a node u if there is another robot between u and r is *systematically* applied to obtain the local view. Finally, we assume that robots are *self-inconsistent*, meaning that different transformations may be applied at different rounds. In more detail, the adversary can choose a different transformation at each round as opposed to *self-consistent* robots where the transformation applied for a given robot does not change during the execution.

It is important to note that when a robot r computes a destination d, it is relative to its local view $f(G_r)$, which is the globally oriented view transformed by some $f \in \mathcal{IT}$. So, the actual movement of the robot in the *globally oriented view* is $f^{-1}(d)$. For example, if $d = Up$ but the robot sees the torus upside-down (f is the π-rotation), then the robot moves $Down = f^{-1}(Up)$. In a configuration C, $V_C(i,j)$ denotes the globally oriented view of a robot located at (i,j).

Algorithm

An algorithm $\mathcal{A}$ is a tuple $(Cl, Init, T)$ where Cl is the set of possible colors, $Init$ is a mapping from any considered torus to a non-empty set of initial configurations in that torus, and T is the transition function $Views \to \{Idle, Up, Left, Down, Right\} \times Cl$, where $Views$ is the set of local views. When the robots are in Configuration C, a configuration C' obtained after one round satisfies: for all $((i,j),c) \in C'$, there exists a robot in C with color $c' \in Cl$ and a transformation $f \in \mathcal{IT}$ such that one of the following conditions holds:

- $((i,j),c') \in C$ and $f^{-1}(T(f(V_C(i,j)))) = (Idle, c)$,
- $(((i-1) \bmod \mathcal{C}, j), c') \in C$ and $f^{-1}(T(f(V_C((i-1) \bmod \mathcal{C}, j)))) = (Right, c)$,
- $(((i+1) \bmod \mathcal{C}, j), c') \in C$ and $f^{-1}(T(f(V_C((i+1) \bmod \mathcal{C}, j)))) = (Left, c)$,
- $((i, (j-1) \bmod \mathcal{R}), c') \in C$ and $f^{-1}(T(f(V_C(i, (j-1) \bmod \mathcal{R})))) = (Up, c)$, or
- $((i, (j+1) \bmod \mathcal{R}), c') \in C$ and $f^{-1}(T(f(V_C(i, (j+1) \bmod \mathcal{R})))) = (Down, c)$.

We denote by $C \mapsto C'$ the fact that C' can be reached in one round from C (*n.b.*, $\mapsto$ is then a binary relation over configurations). An execution of Algorithm $\mathcal{A}$ in a torus G is then a sequence $(C_i)_{i\in\mathbb{N}}$ of configurations such that $C_0 \in Init(G)$ and $\forall i \geq 0$, $C_i \mapsto C_{i+1}$.

Definition 1 (Perpetual Torus Exploration)**.** *An algorithm $\mathcal{A}$ solves the* Perpetual Torus Exploration *(PTE) problem if in any execution $(C_i)_{i\in\mathbb{N}}$ of $\mathcal{A}$ and for any node $(i,j) \in V$ of the torus and any time t, there exists $t' > t$ such that (i,j) is occupied in $C_{t'}$.*

Notations. $\vec{t}_{(i,j)}(C)$ denotes the translation of the configuration C of vector (i,j).

4. Impossibility results

Lemma 1. *Let $\mathcal{A}$ be an algorithm using a set of $n > 0$ robots. If $\mathcal{A}$ solves the exploration problem for any torus then, there exists a torus such that for any execution $(C_i)_{i\in\mathbb{N}}$ of $\mathcal{A}$ on this torus, there is a configuration C_i such that the distance between the two farthest robots is at least $2\Phi + 3$.*

Proof. We proceed by contradiction. Assume, there is an algorithm $\mathcal{A}$ that solves the PTE problem and let $0 < B$ be the farthest any of the robots will be from each other, in any torus. Let $(C_i)_{i\in\mathbb{N}}$ be the execution of $\mathcal{A}$ on a very large torus $\mathcal{C}, \mathcal{R} \gg B$. When all robots are at distance at most B, then the occupied positions are included in a square sub-grid of size $B \times B$. Since the number of possible configurations included in a sub-grid of size $B \times B$ is finite, there must be two indices t_1 and t_2, when the positions and colors of the robots in the corresponding sub-grids are the same, formally, such that $C_{t_2} = \vec{t}_{(i,j)}(C_{t_1})$ and $t_1 < t_2$ for a given translation $\vec{t}_{(i,j)}$. By making the adversary choose the same rotation, the movements done by the robots in configurations C_{t_1} and C_{t_2} are the same as each robot has the same globally oriented view in both configurations, only their positions on the torus change. Thus $C_{t_2+1} = \vec{t}_{(i,j)}(C_{t_1+1})$ and so on so forth, so that $\forall x$, $C_{t_2+x} = \vec{t}_{(i,j)}(C_{t_1+x})$. We obtain that the configurations are periodic with period $p = t_2 - t_1$, up to translation.

Assume that the torus being explored is of dimensions $\mathcal{C} \times \mathcal{R}$ such that $\mathcal{C} = 3np^3 \max(|i|, 1)$ and $\mathcal{R} = 3np^3 \max(|j|, 1)$. The dimensions of the torus are proportional to the non-null scalar components of translation $\vec{t}_{(i,j)}$ *i.e.*, $i3np^3 \equiv 0 \mod \mathcal{C}$ and $j3np^3 \equiv 0 \mod \mathcal{R}$. This means that,

$$(\vec{t}_{(i,j)})^{3np^3}(C_{t_1}) = \vec{t}_{(i3np^3, j3np^3)}(C_{t_1}) = \vec{t}_{(0,0)}(C_{t_1}) = C_{t_1}.$$

Since translation $\vec{t}_{(i,j)}$ is performed in p rounds, after $p \times 3np^3 = 3np^4$ rounds, all robots will retake their initial positions, so the whole configuration is periodic with period $3np^4$. In this setting, a node is visited infinitely often if and only if it is visited between round t_1 and $t_1 + 3np^4$. Now we have to prove that some nodes are left unvisited between round t_1 and $t_1 + 3np^4$.

Between time t_1 and $t_1 + 3np^4$, each robot visits at most $3np^4$ nodes, hence all the robots visit at most $n \times 3np^4$ nodes after t_1. However, there are at least $9n^2p^6 \leq \mathcal{C} \times \mathcal{R}$ nodes in the torus. Hence, there exist some nodes which are not visited infinitely often, which is a contradiction.

Note that we only proved there are some nodes that are not perpetually visited. Nevertheless, observe that at most nt_1 nodes are visited before t_1 and we can increase arbitrarily the chosen period p by a factor $f \in \mathbb{N}^*$ without changing the result (in particular t_1 does not depend on f). By taking $f \geq 1$ such that $9n^2(fp)^6 - 3n^2(fp)^4 > nt_1$, we have that the number of visited nodes (before or after t_1) is $nt_1 + 3n^2(fp)^4$ and is smaller than the number of nodes in the torus ($9n^2(fp)^6$), hence there is at least one node that is never visited. This implies that the impossibility also holds for non-perpetual algorithms as well (where each node must be visited at most once). □

We restate the following lemma proven in [5].

Lemma 2. *A robot with a self-inconsistent compass that sees no other robot, either stays idle or the adversary can make it alternatively move between two chosen adjacent nodes.*

Theorem 1. *In a torus, it is impossible to solve the exploration problem with two myopic robots equipped with self-inconsistent compasses that agree on a common chirality.*

Proof. By Lemma 1, there is a torus and a configuration where the two robots are at distance $2\Phi + 3$ from each other. In this case, the two robots are isolated. By Lemma 2, the two robots will remain idle or the adversary can make them alternatively move between two nodes, never being in vision from each other and never visiting another node. □

Theorem 2. *In a torus, it is impossible to solve the exploration problem with three anonymous, oblivious, and myopic robots equipped with self-inconsistent compasses that agree on a common chirality.*

Proof. By Lemma 1, there is a torus and a configuration where the distance between the two farthest robots is $2\Phi + 3$ from each other. We have one of the two following possibilities, (i) there are three isolated robots, or (ii) there is an isolated robot and two robots in vision from each other.

In the first case, it is easy to see that the three isolated robots cannot explore the torus because, by Lemma 2, they have to stay idle or the adversary can make them alternatively move between two nodes, never being in vision from each other and never visiting another node.

In the second case, the two robots that see each other cannot travel together in a direction (because they have the same view). All they can do is get either closer to each other or further from each other. Formally, there is a point P at the middle of the two robots and, if they stay in vision, they will always be at the same distance from that point. The two robots can explore a subgrid $\Phi \times \Phi$ centered at a given middle point. This point is at distance at least $\frac{3\Phi}{2} + 2$ from the isolated robots.

If the two robots in vision get isolated from one another, they will be at distance $\frac{\Phi}{2} + 1$ from the middle point. In this case, the closest robot to the originally isolated robot will be at distance $\Phi + 1$. Now the three robots are isolated, and, as in the first case, they cannot explore the torus. □

5. A generic proof for our algorithms

Proving that our algorithms are correct is at the same time very intuitive when looking at the provided animations [8] and sometimes very tricky, especially when it involves algorithms using ambiguous moves, since the adversary could choose a problematic execution that is not easy to construct by hand. To help the reader, we provide in this section a generic proof that we use to prove the correctness of our algorithms.

In the theorem, we consider that the rows of the torus (*i.e.*, the set of nodes having the same y-coordinate) are indexed from 0 to $\mathcal{R} - 1$, and we consider the index modulo $\mathcal{R}$ so that index i and $i + \mathcal{R}$ refers to the same row. Observe that in the following theorem, the value of t_x can be arbitrary. Since a torus is invariant by translation, we can index the rows so that the bottom-most robot is on the row with index 0.

Theorem 3. *Let $\mathcal{A}$ be an algorithm, $t_y \in \mathbb{Z}$ and i a row index. If, in any execution, after a finite number of rounds, the configuration is the same as the initial configuration, but translated by a vector (t_x, t_y) (for some $t_x \in \mathbb{Z}$), and the $|t_y|$ consecutive rows from index i to index $i + |t_y| - 1$ have been visited, then $\mathcal{A}$ solves the PTE problem.*

Proof. Take an arbitrary execution of $\mathcal{A}$. By assumption, after a finite number of rounds, say t, the configuration is the (t_x, t_y)-translation of the initial configuration and rows with indexes $i, \ldots i + |t_y|$ have been visited. Since the topology is a torus, the same property is true from round t. So after a finite number of rounds, say t', the configuration is the same as in round t but translated by (t'_x, t_y) (observe that t' and t'_x are not necessarily equal to t and t_x, respectively, but t_y is fixed regardless of the execution). Also, the rows with indexes $i + |t_y|, \ldots, i + 2|t_y|$ have been visited since the index i does not depend on the execution by assumption, but only on the initial

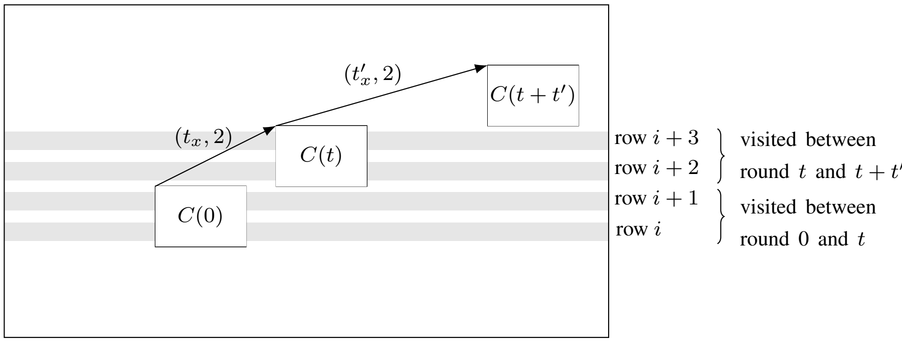


Figure 1: Illustration of the exploration using our generic theorem. $C(t)$ is the same configuration as $C(0)$ but translated by $(t_x, 2)$. An execution starting from configuration $C(t)$ is the same as an execution starting from configuration $C(0)$, but translated by $(t_x, 2)$. Hence, if rows i, $i+1$ are visited between round 0 and t, then rows $i+2$, $i+3$ are visited between round t and t'. Observe that t' and t'_x are not necessarily equal to t and t_x, respectively.

configuration. Hence, if the initial configuration is translated by (t_x, t_y), then the rows that are visited are also translated by (t'_x, t_y) (for some $t'_x \in \mathbb{Z}$, but since we consider the entire rows, so only the y-coordinate is important). Figure 1 illustrates this process.

So the configuration at round $t + t'$ is the $(t_x + t'_x, 2t_y)$-translation of the initial configuration and $2|t_y|$ consecutive rows have been visited from index i to index $2|t_y|$. By repeating the same process $\left\lceil \frac{\mathcal{R}}{|t_y|} \right\rceil$ times, after a finite number of rounds, the entire torus is visited, and the same process continues. This means that the algorithm solves the PTE problem. □

Clearly, the same theorem is true if we swap rows and columns *i.e.*, if after t rounds the configuration is the (t_y, t_x)-translation of the initial configuration and $|t_y|$ consecutive columns have been visited, then $\mathcal{A}$ solves the PTE problem.

6. Visibility range one

We address in this subsection the case in which robots have visibility range one. We propose two algorithms; $\mathcal{A}_1$ which takes advantage of chirality and $\mathcal{A}_2$ which guarantees perpetual exploration even without chirality with a single additional color with respect to $\mathcal{A}_1$.

6.1. The case with chirality

We present an algorithm, denoted by $\mathcal{A}_1$, which assumes a visibility range one and uses three robots and two colors. By Theorem 1, $\mathcal{A}_1$ is optimal w.r.t. the number of robots, and by Theorem 2, $\mathcal{A}_1$ is also optimal w.r.t. the number of colors. Animations are made available online [8] to help the reader visualize the algorithm.

The idea of the algorithm is to make the robots alternate between exploring a row and exploring a column. To explore the whole torus, robots move so that all the

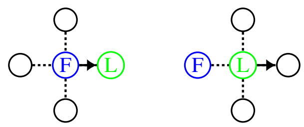


Figure 2: Rules for moving straight.

nodes of the torus are explored infinitely often. More precisely, after exploring row r_i and column c_j, the robots will proceed at exploring row $r_{i-1 \bmod \mathcal{R}}$ and then column $c_{j-1 \bmod \mathcal{C}}$ and so on. Observe that, to apply our generic Theorem 3, we only need to show that we visit an entire row when we reach the same configuration translated by a vector $(t_x, 1)$.

Initially, the robots are co-linear with respectively color L, F, F, [1] as shown in the first configuration in Figure 4. The line of the torus on which they are located is considered as a row. The robot with color F, which does not sense the robot of color L, moves up changing its color to L while the two other robots move along their current row in the following manner: the robot initially with color L moves away from the one with color F and the remaining robot just follows it. This initial sequence of configurations is presented in Figure 4.

To explore a row (resp. column), one robot stays idle while the two others travel in a straight line along the nodes of the row (resp. column) being explored until they reach the idle robot. The idle robot is located on a neighboring row (resp. column). The idle robot has color L and is called the *landmark*. The two robots traveling together in a straight line have different colors. One robot, called the *follower*, has color F and the other robot, called the *leader*, has color L. To explore a row (resp. column), the two robots have to be next to each other on that row (resp. column). The follower always follows the leader and the leader always moves away from the follower. This is done by executing the rules presented in Figure 2.[2]

The tricky part of this algorithm is how robots switch from exploring a column to exploring a row and *vice versa*. Initially, the traveling group visits a row and leaves behind a robot (aka the landmark). When they reach the landmark again (after $\mathcal{C} - 2$ rounds), the landmark is on the left side (from their perspective). By executing the rules presented in Figure 3, the robots make a turn and a new traveling group is formed. The corresponding sequence of configurations is presented in Figure 5. The previous leader becomes the new landmark.

The robots proceed to explore the column and after $\mathcal{R} - 1$ more rounds, the leader reaches the landmark. By executing the rules presented in Figure 7, the robots make a turn and a new traveling group is formed. The corresponding sequence of configurations is presented in Figure 6. At the end of the sequence, the whole configuration is the same as the initial configuration, but translated by vector $(-1, -1)$, so the same

[1]Note that any reachable configuration can be an initial configuration

[2]In all figures, colored letters inside nodes indicate the color of the robots occupying the nodes. Moreover, when a colored letter is given next to a node, it indicates which color the robot will take in the next round.

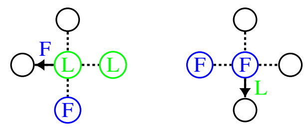


Figure 3: Rules for switching from visiting a row to visiting a column.

process continues forever.

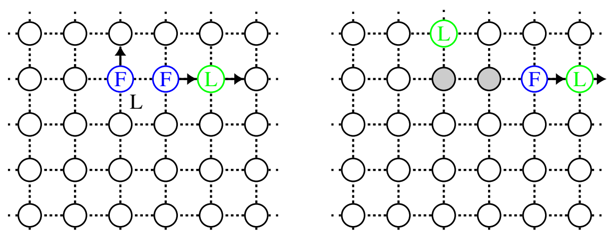


Figure 4: Sequence of configurations executed from the initial configuration. The gray nodes show the visited nodes of the row with index 0, the row where the robots are initially located

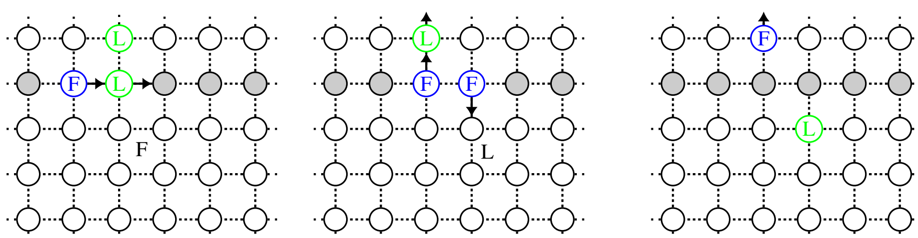


Figure 5: Sequence of configurations when moving from exploring a row to exploring a column. The gray nodes show the visited nodes of the row with index 0, the row where the robots are initially located

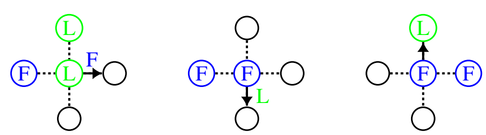


Figure 7: Rules for switching from visiting a column to visiting a row.

It is important to note that every node on a column/row is visited during the exploration of that column/row. Also, the landmark moves two nodes to the left and one node up when going from exploring a column to exploring a row. And, it moves one

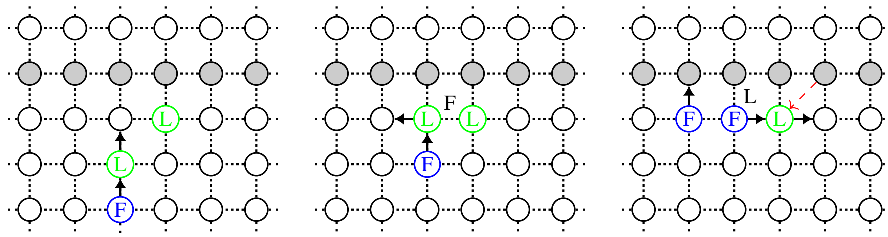


Figure 6: Sequence of configurations when robots move from exploring a column to exploring a row. The gray nodes show the visited nodes of the row with index 0, the row where the robots are initially located. The red dashed arrow highlights the movement of the landmark since the initial configuration.

node to the right and two nodes downward when going from exploring a row to exploring a column. This means that between two consecutive columns (rows) exploration, the landmark moves one node to the left and one node downward.

Theorem 4. $\mathcal{A}_1$ *solves the PTE problem with three robots and two colors.*

Proof. As we saw, starting from an initial configuration, the traveling group always visits the entire row they started from and reaches the landmark again after $\mathcal{C}-2$ rounds. Indeed, this is true regardless of the size of the torus as the traveling group moves in straight line until it reaches the landmark. After a turn (Fig. 5), the traveling group visit a column and, after $\mathcal{R}$ rounds, the robots reach the same initial configuration, translated by $(-1, -1)$. Since, in finite number of rounds, the robots have explored the row they started on (*i.e.*, the row with index 0, shown in gray in the figures) and reached a configuration that is the $(-1, -1)$-translation of the initial configuration, by Theorem 3, $\mathcal{A}_1$ solves the PTE problem in a torus of size $\mathcal{C} \times \mathcal{R}$, with $\mathcal{C}, \mathcal{R} \geq 4$. □

6.2. *The case without chirality*

We present an algorithm, denoted by $\mathcal{A}_2$, which assumes a visibility range one and uses three robots and three colors. By Theorem 1, $\mathcal{A}_2$ is optimal w.r.t. the number of robots. Animations are made available online [8] to help the reader visualize the algorithm.

Initially, robots are co-linear with, respectively, color F, F, B[3]. The row of the torus on which the three robots are co-located initially is considered as a column, as in the first configuration in Figure 10. Let $\ell_1, \ell_2, \ell_3, \ldots, \ell_{\mathcal{R}}$ be the sequence of consecutive rows in the initial configuration such that $\mathcal{R}$ is the number of rows, ℓ_1 is the row which hosts the robot with color F having a unique adjacent node and $\ell_{\mathcal{R}-1}$ (resp. $\ell_{\mathcal{R}}$) is the row on which the robot with color B (resp. the second robot with color F) is located.

The idea of $\mathcal{A}_2$ is to explore the torus row by row from ℓ_1 to ℓ_n infinitely often. This is achieved by repeating infinitely often two consecutive phases: *row-change* and *row-exploration* described below:

[3]recall that any reachable configuration can be an initial configuration

1. Phase *row-change*. This phase starts when robots are co-linear on a column with respectively F, F, B (as in the initial configuration). The three robots execute the rules shown in Figure 8. That is, after one round, two robots are located in the same row. The row-exploration phase is then started.

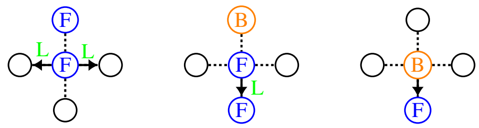


Figure 8: Rules for switching rows.

2. Phase *row-exploration*. This phase follows Phase row-change. The two robots on the same row simply explore all the nodes of the row by executing the rules of Figure 9 while the third robot, referred to as a landmark, remains idle. When the robot with color L meets the landmark, it moves to the next row followed by the robot with color F. Phase row-change is then initiated.

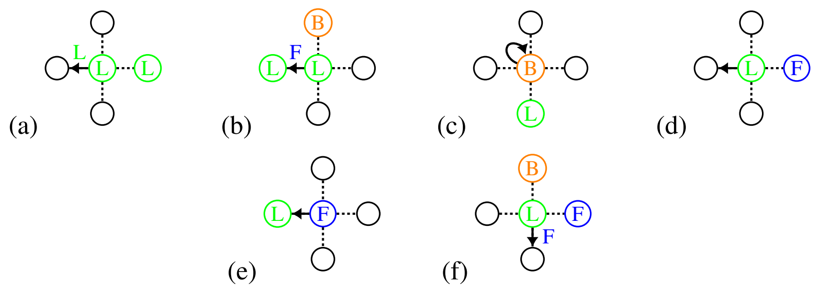


Figure 9: Rules for exploring a row.

Figure 10 presents the sequence of configurations during both the row-change and row-exploration phases.

Theorem 5. *$\mathcal{A}_2$ solves the PTE problem with three robots and three colors.*

Proof. Starting from an initial configuration, we saw that, after two rounds, the traveling group starts exploring a row (the row indexed 0, *i.e.*, where the bottom-most robot was located initially). After $\mathcal{R}-2$ more rounds, the leader sees the landmark, and after one more round the configuration is the same as the initial configuration, but translated by vector $(0,-1)$ (see Fig. 10). Since one row has been visited, by Theorem 3, $\mathcal{A}_2$ solves the PTE problem. □

7. Visibility range two

7.1. The case with chirality

We present an algorithm, denoted by $\mathcal{A}_3$, which assumes a visibility range two and uses four oblivious robots. $\mathcal{A}_3$ is optimal w.r.t. the number of colors. By Theorem 2,

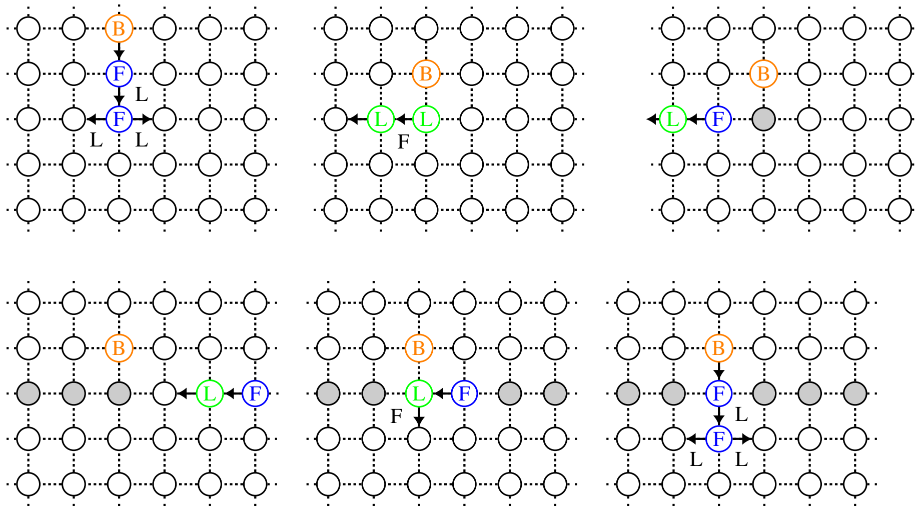


Figure 10: $\mathcal{A}_2$: Sequence of configurations when moving along a row, reaching the same configuration than the initial one, but translated. The gray nodes show the visited nodes of the row with index 0, the row of the initial bottom-most robot.

$\mathcal{A}_3$ is optimal w.r.t. the number of robots, for oblivious robots. Animations are made available online [8] to help the reader visualize the algorithm.

The idea of the algorithm is again to make the robots explore the torus row by row in a given direction. This is achieved as follows: three robots, referred to as the traveling group, move to explore three adjacent rows at the same time, and one robot is left behind to be used as their landmark. When the traveling group reaches the landmark, all four robots perform a three rounds sequence to move to the next rows to be explored.

Initially, the robots are placed in the configuration shown in Figure 11. The three robots on the right form a $>$ shape and are the traveling group. That is, two robots are located on the same column separated by one empty node. The goal of the traveling group is to explore the whole row by executing the rules defined in Figure 12.

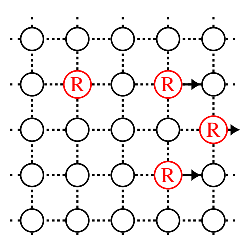


Figure 11: Initial configuration of Algorithm $\mathcal{A}_3$

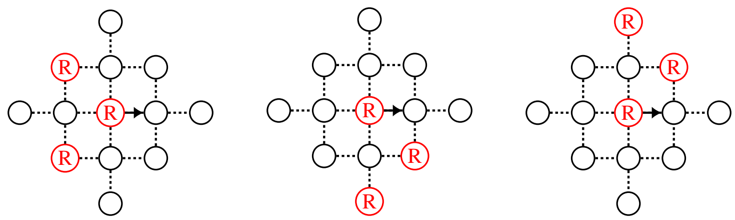


Figure 12: Rules for three robots moving straight.

The landmark is left behind so that the traveling group knows when they are done exploring the current rows and they have to move to the next one. Note that the landmark is on the same row as the topmost robot. When that robot is one node away from the landmark it goes down. The same is done by the landmark. The bottom robot keeps going right because it does not see the landmark, and the center robot stays idle. The rules executed by the robots are presented in Figure 13. After one round, the robots form a T-shape.

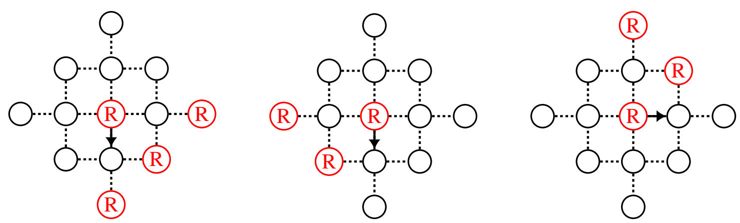


Figure 13: Rules executed when robots initiate rows change, the last two are the same rules are previously, but are shown here with another orientation to help the reader.

From the T-shape, the robots move to create a reverse L shape *i.e.*, the two robots in the center of the T-shape move down while the robot on the right goes left. Figure 14 presents the rules executed during this process.

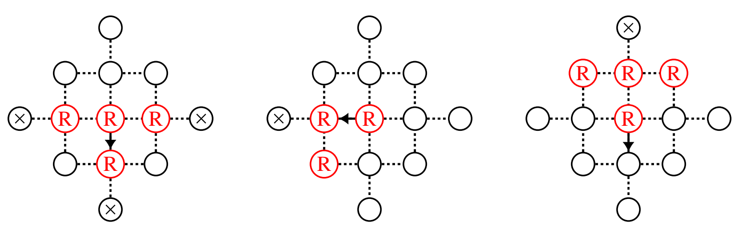


Figure 14: Rules for the creation of the reverse L shape.

Within the reverse L shape, three robots are co-linear (the ones located on the long side). Among these robots, the one in the middle moves to the right to recreate the $>$ shape while all the other robots remain idle. Refer to the rule presented Figure 15. That is, after three rounds the robots change rows and the $>$ shape is built again.

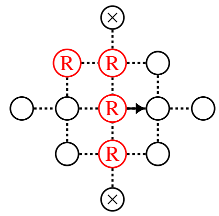


Figure 15: Rule for restoring the $>$ shape.

Now the three robots on the right form the new traveling group. The robots repeat the same behavior and hence start moving right until they reach the landmark once more. There are two more rules to tell the topmost robot in the traveling group to keep following the group even if it sees the landmark at the back. These rules are presented in Figure 16.

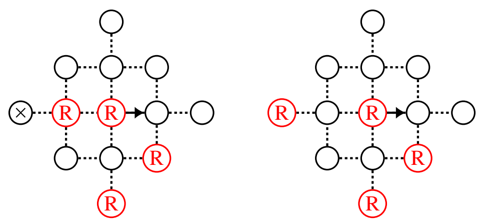


Figure 16: Rules for the topmost robot to keep traveling with the group.

It is important to note that the landmark changes its position to two nodes to the right and one node down. The fact that it moves down makes the robots always explore a new row. Figure 17 presents the sequence of configuration during this process.

Theorem 6. *$\mathcal{A}_3$ solves the PTE problem with four oblivious robots.*

Proof. Starting from an initial configuration, the traveling group moves right until it reaches the landmark. Hence, regardless of the size of the torus, after $\mathcal{R} - 4$ rounds, the traveling group reaches the landmark, after visiting one row (the row with index 0, shown in gray in the figure, assuming the bottom-most robot is initially located on the row with index 0). After four more rounds (see Figure 17), the obtained configuration is the same as the initial configuration, but translated by a vector $(-2, -1)$. By Theorem 3, the algorithm solves the PTE problem. $\square$

7.2. The case without chirality

We present an algorithm, denoted by $\mathcal{A}_4$, which assumes a visibility range two and uses five oblivious robots. Animations are made available online [8] to help the reader visualize the algorithm.

The idea of the algorithm is to make the robots alternate between exploring rows and then columns in such a way that they explore infinitely often all the nodes of the

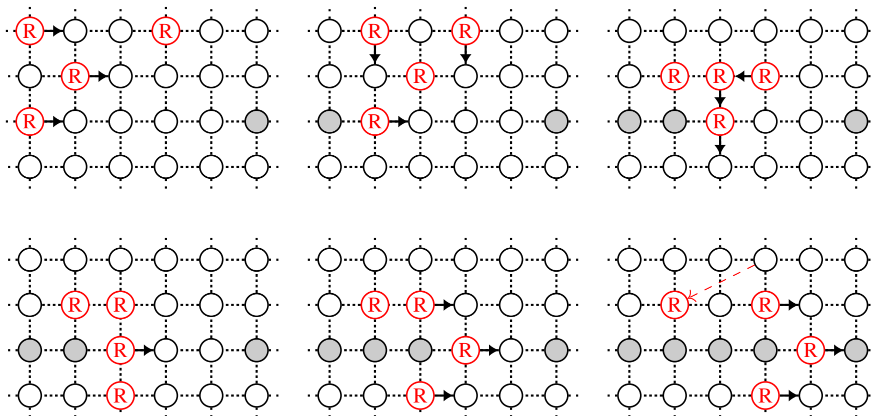


Figure 17: $\mathcal{A}_3$: Sequence of configurations when changing rows. The gray nodes show the visited nodes of the row with index 0, the row of the initial bottom-most robot (the first gray node is where the bottom-most robot was located initially). The red dashed arrow highlights the movement of the landmark.

torus. After visiting three consecutive columns and three consecutive rows, the configuration is translated diagonally. By doing so infinitely often, perpetual exploration is performed.

For this purpose, robots are divided into two teams throughout the execution of $\mathcal{A}_4$: *the explorers* and *the landmark*. The explorer team consists of four robots placed in a perfect T shape and are in charge of the exploration while the landmark is just a single robot used to guide the explorers so that they can keep track of the exploring direction. Note that during the execution of $\mathcal{A}_4$, some robots may change their respective role.

Initially, robots are placed so that the explorers are ready to explore three columns[4]. More precisely, four robots form a T shape, and the landmark is located diagonally above (refer to Figure 18). The explorer team executes the rules presented in Figure 19 in order to move in a straight line. Initially, they visit the three columns simultaneously and reach the landmark from above. When reaching the landmark, they perform a right turn (from their perspective *i.e.*, they were going down and are now heading left), using the additional rule presented in Figure 20. The corresponding sequence of configurations is shown in the first 5 configurations of Figure 21. After the turn, the isolated robot in the reached configuration is the landmark while the four other robots are in the explorer team.

The explorers now move in a straight line and visit the three rows simultaneously. After that; they reach the landmark again but this time the landmark is on the other side so the same sequence as before is performed but in a mirrored way *i.e.*, they perform a left turn (again from their perspective, from going left to going down). The sequence of configurations during this process is presented in the last 5 configurations of Figure 21, which is the same as the first right turn but in a mirrored way. After visiting these three

[4]Again, we recall that every reachable configuration can be initial.

rows (gray nodes in the figures), the configuration is similar to the initial configuration but translated diagonally (3 nodes to the East and 3 nodes to the North). By repeating the same process the robots will explore infinitely often all the nodes of the torus.

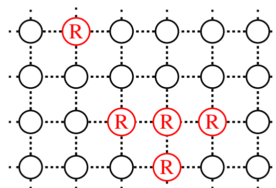


Figure 18: Instance of an initial configuration of Algorithm $\mathcal{A}_4$

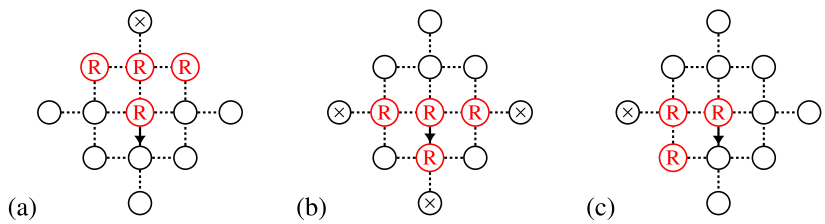


Figure 19: Rules to move in a straight line when forming a T shape.

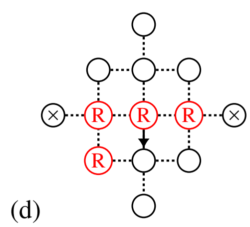


Figure 20: Rules for performing a U-turn.

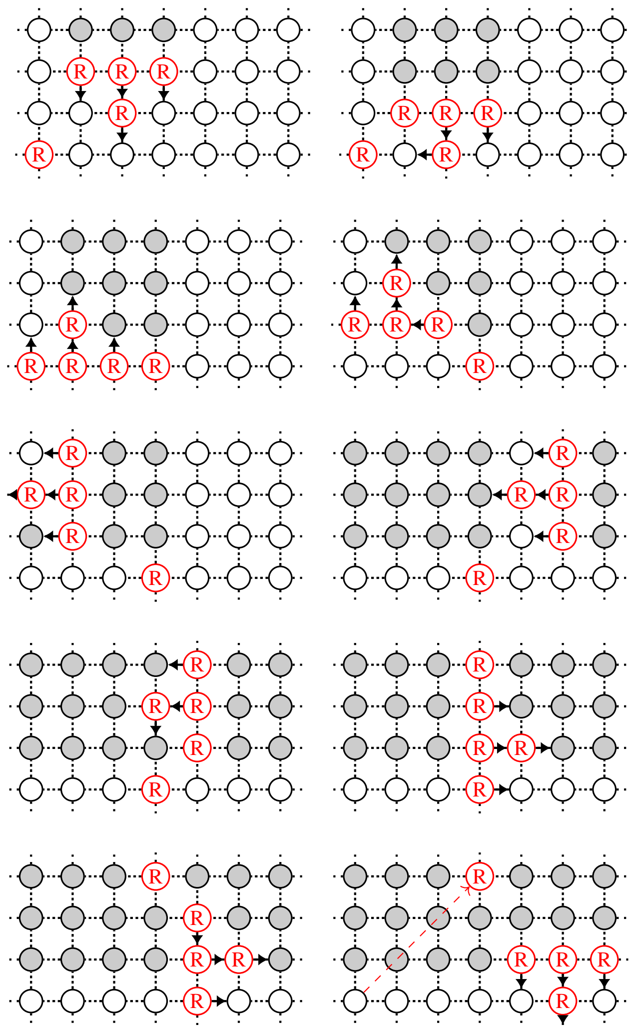
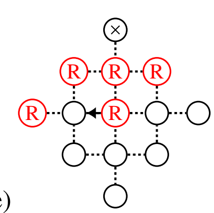


Figure 21: $\mathcal{A}_4$: Sequence of configurations, after visiting three columns, the robots make a turn, visit three consecutive rows (shown in gray), make a turn, and reach the same configuration as the initial one but translated by a vector $(3, 3)$. The red dashed arrow highlights the movement of the landmark



(e)

Figure 22: Rules for performing a U-turn.

Theorem 7. *$\mathcal{A}_4$ solves the PTE problem with five oblivious robots.*

Proof. Assuming initially the bottom-most robot is on the row with index 0, then the landmark is on the row with index 3 (see the initial configuration in Figure 18). After $\mathcal{R} - 2$ rounds (while visiting three columns), the traveling group reaches the landmark. From there, they perform a turn and start exploring three consecutive rows located above the landmark (see the first 5 configurations in Figure 21), having index 4, 5, and 6. After $\mathcal{C} + 1$ more rounds, the robots reach again the landmark and start performing another turn (see the last 5 configurations of Figure 21). After three more rounds, the configuration is the same as the initial configuration, but translated by a vector $(3, 3)$. Since they have visited three consecutive rows with index 4, 5, and 6, by Theorem 3, $\mathcal{A}_4$ solves the PTE problem. □

8. Conclusion

We presented two optimal solutions for the PTE problem with respect to both the number of robots and the number of colors when robots share a common chirality and have visibility one and two respectively. Indeed, we have shown that three robots endowed with two colors are necessary and sufficient to solve the problem when robots have visibility one and four oblivious robots are necessary and sufficient to solve the problem when robots have visibility two.

We also addressed the case in which robots are completely disoriented *i.e.*, robots do not share a common chirality. We proposed two algorithms to solve the PTE problem one which is optimal with respect to the number of robots in the case of visibility range one and another one for the case in which robots have visibility range two.

A direct extension to this work would be to show the optimality of the proposed solutions when robots are disoriented. Indeed, we conjuncture that three colors are necessary when robots have a visibility range one and that five robots are necessary with oblivious robots when the visibility range is two.

Another interesting extension would be to consider $(\mathcal{C}, \mathcal{R})$-tori such that $\mathcal{C}, \mathcal{R} < n\Phi + 1$. Ad-hoc solutions might be needed in this case as robots might observe the same robots on different sides of the torus.

Finally, it would be interesting to see how the proposed solutions could be adapted to solve the terminating exploration problem.

References

[1] Lélia Blin, Alessia Milani, Maria Potop-Butucaru, and Sébastien Tixeuil. Exclusive perpetual ring exploration without chirality. In Nancy A. Lynch and Alexander A. Shvartsman, editors, *Distributed Computing, 24th International Symposium, DISC 2010, Cambridge, MA, USA, September 13-15, 2010. Proceedings*, volume 6343 of *Lecture Notes in Computer Science*, pages 312–327, Boston, Massachusetts, USA, September 2010. Springer.

[2] François Bonnet, Alessia Milani, Maria Potop-Butucaru, and Sébastien Tixeuil. Asynchronous exclusive perpetual grid exploration without sense of direction.

In Antonio Fernández Anta, editor, *Proceedings of International Conference on Principles of Distributed Systems (OPODIS 2011)*, number 7109 in Lecture Notes in Computer Science (LNCS), pages 251–265, Toulouse, France, December 2011. Springer Berlin / Heidelberg.

[3] Quentin Bramas, Stéphane Devismes, Anaïs Durand Pascal Lafourcade, and Anissa Lamani. Beedroids: How luminous autonomous swam of uavs can save the world? In *11th International Conference on Fun with Algorithms (FUN 2022)*, LIPIcs, pages 7:1–7:21, 2022.

[4] Quentin Bramas, Stéphane Devismes, and Pascal Lafourcade. Finding Water on Poleless using Melomaniac Myopic Chameleon Robots. In *FUN 2020, 10th International Conference on Fun with Algorithms*, pages 6:1–6:19, Favignana, Sicily, Italy, September 28-30 2020. LiPICs.

[5] Quentin Bramas, Stéphane Devismes, and Pascal Lafourcade. Infinite grid exploration by disoriented robots. In *International Conference on Networked Systems*, pages 129–145. Springer, 2020.

[6] Quentin Bramas, Stéphane Devismes, and Pascal Lafourcade. Optimal Exclusive Perpetual Grid Exploration by Luminous Myopic Opaque Robots with Common Chirality. In *ICDCN'21: International Conference on Distributed Computing and Networking, Virtual Event*, pages 76–85, Nara, Japan, 5-8 January 2021. ACM.

[7] Omar Darwich, Ahmet-Sefa Ulucan, Quentin Bramas, Anissa Lamani, Anaïs Durand, and Pascal Lafourcade. Perpetual torus exploration by myopic luminous robots. In Stéphane Devismes, Franck Petit, Karine Altisen, Giuseppe Antonio Di Luna, and Antonio Fernández Anta, editors, *Stabilization, Safety, and Security of Distributed Systems - 24th International Symposium, SSS 2022, Clermont-Ferrand, France, November 15-17, 2022, Proceedings*, volume 13751 of *Lecture Notes in Computer Science*, pages 164–177. Springer, 2022.

[8] Omar Darwich, Ahmet-Sefa Ulucan, Quentin Bramas, Anissa Lamani, Anaïs Durand, and Pascal Lafourcade. Perpetual Torus Exploration by Myopic Luminous Robots: The Animations, April 2022. https://doi.org/10.5281/zenodo.6482690.

[9] Ajoy Kumar Datta, Anissa Lamani, Lawrence L. Larmore, and Franck Petit. Ring exploration by oblivious agents with local vision. In *IEEE 33rd International Conference on Distributed Computing Systems, ICDCS 2013, 8-11 July, 2013, Philadelphia, Pennsylvania, USA*, pages 347–356. IEEE Computer Society, 2013.

[10] Ajoy Kumar Datta, Anissa Lamani, Lawrence L. Larmore, and Franck Petit. Enabling ring exploration with myopic oblivious robots. In *2015 IEEE International Parallel and Distributed Processing Symposium Workshop, IPDPS 2015*, pages 490–499, Hyderabad, India, May 25-29, 2015 2015. IEEE Computer Society.

[11] Stéphane Devismes, Anissa Lamani, Franck Petit, Pascal Raymond, and Sébastien Tixeuil. Terminating exploration of a grid by an optimal number of asynchronous oblivious robots. *Comput. J.*, 64(1):132–154, 2021.

[12] Stéphane Devismes, Anissa Lamani, Franck Petit, and Sébastien Tixeuil. Optimal torus exploration by oblivious robots. *Computing*, 101(9):1241–1264, 2019.

[13] Stéphane Devismes, Franck Petit, and Sébastien Tixeuil. Optimal probabilistic ring exploration by semi-synchronous oblivious robots. *Theoretical Computer Science (TCS)*, 498:10–27, 2013.

[14] Paola Flocchini, David Ilcinkas, Andrzej Pelc, and Nicola Santoro. Remembering without memory: Tree exploration by asynchronous oblivious robots. *Theor. Comput. Sci.*, 411(14-15):1583–1598, 2010.

[15] Paola Flocchini, David Ilcinkas, Andrzej Pelc, and Nicola Santoro. How many oblivious robots can explore a line. *Inf. Process. Lett.*, 111(20):1027–1031, 2011.

[16] Paola Flocchini, David Ilcinkas, Andrzej Pelc, and Nicola Santoro. Computing without communicating: Ring exploration by asynchronous oblivious robots. *Algorithmica*, 65(3):562–583, 2013.

[17] Fukuhito Ooshita and Sébastien Tixeuil. Ring exploration with myopic luminous robots. In Taisuke Izumi and Petr Kuznetsov, editors, *Stabilization, Safety, and Security of Distributed Systems - 20th International Symposium, SSS 2018*, volume 11201 of *Lecture Notes in Computer Science*, pages 301–316, Tokyo, Japan, November 4-7 2018. Springer.

[18] Arthur Rauch, Quentin Bramas, Stéphane Devismes, Pascal Lafourcade, and Anissa Lamani. Optimal exclusive perpetual grid exploration by luminous myopic robots without common chirality. In *Networked Systems - 9th International Conference, NETYS 2021, Virtual Event, May 19-21, 2021, Proceedings*, volume 12754 of *Lecture Notes in Computer Science*, pages 95–110. Springer, 2021.