International Journal of Networking and Computing – www.ijnc.org, ISSN 2185-2847
Volume 12, Number 1, pages 73–102, January 2022

Terminating Grid Exploration with Myopic Luminous Robots

Shota Nagahama
Nara Institute of Science and Technology, nagahama.shota.nl1@is.naist.jp
Takayama 8916-5, Ikoma 630-0192, Japan

Fukuhito Ooshita
Nara Institute of Science and Technology, f-oosita@is.naist.jp
Takayama 8916-5, Ikoma 630-0192, Japan

Michiko Inoue
Nara Institute of Science and Technology, kounoe@is.naist.jp
Takayama 8916-5, Ikoma 630-0192, Japan

Received: July 23, 2021
Accepted: September 13, 2021
Communicated by Masahiro Shibata

Abstract

We investigate the terminating grid exploration for autonomous myopic luminous robots. Myopic robots mean that they can observe nodes only within a certain fixed distance, and luminous robots mean that they have light devices that can emit colors. First, we prove that, in the semi-synchronous and asynchronous models, three myopic robots are necessary to achieve the terminating grid exploration if the visible distance is one. Next, we give fourteen algorithms for the terminating grid exploration in various assumptions of synchrony (fully-synchronous, semi-synchronous, and asynchronous models), visible distance, the number of colors, and a chirality. Six of them are optimal in terms of the number of robots.

Keywords: Look-Compute-Move, light, limited visibility, exploration, grids

1 Introduction

1.1 Background and motivation

Many studies about cooperation of autonomous mobile robots have been conducted in the field of distributed computing. These studies focus on the minimum capabilities of robots that permit to achieve a given task. To model operations of robots, the *Look-Compute-Move (LCM) model* [21] is commonly used. In the LCM model, each robot repeats cycles of Look, Compute, and Move phases. In the Look phase, the robot observes positions of other robots. In the Compute phase, the robot

[0]A preliminary extended abstract of this paper appears in the proceedings of the 23rd Workshop on Advances in Parallel and Distributed Computational Models (APDCM 2021). We are grateful to Rikuma Tsujimoto for debugging algorithms by constructing simulation programs. This work was partially supported by JSPS KAKENHI Grant Numbers 18K11167 and 20H04140, and JST SICORP Grant Number JPMJSC1806.

executes its algorithm using the observation as its input, and decides whether it moves somewhere or stays idle. In the Move phase, it moves to a new position if the robot decided to move in the Compute phase. To consider minimum capabilities, most studies assume that robots are *identical* (*i.e.*, robots execute the same algorithm and have no identifier), *oblivious* (*i.e.*, robots have no memory to record their past history), and *silent* (*i.e.*, robots do not have communication capabilities). Furthermore, they have *no global compass*, *i.e.*, they do not agree on the directions. Based on the LCM model, previous works clarified solvability of many tasks such as exploration, gathering, and pattern formation in continuous environments (aka two- or three-dimensional Euclidean space) and discrete environments (aka graph networks) (see a survey [17]).

In this paper, we focus on *exploration* in graph networks, which is one of the most central tasks for mobile robots. Two variants of exploration tasks have been well studied: the *perpetual* exploration requires every robot to visit every node infinitely many times, and the *terminating* exploration requires robots to terminate after every node is visited by a robot at least once. During the last decade, many works have considered the perpetual and terminating exploration on the assumption that each robot has unlimited visibility, *i.e.*, it observes all other robots in the network. The perpetual exploration has been studied for rings [1] and grids [2]. The terminating exploration has been studied for lines [15], rings [13,16,18], trees [14], finite grids [10,11], tori [12], and arbitrary networks [6]. However, the capability of the unlimited visibility seems powerful and somewhat contradicts the principle of weak mobile robots. For this reason, some studies consider the more realistic case of *myopic* robots [8,9]. A myopic robot has limited visibility, *i.e.*, it can see nodes (and robots on them) only within a certain fixed distance ϕ. Datta et al. studied the terminating exploration of rings for $\phi = 1$ [8] and $\phi = 2, 3$ [9]. Not surprisingly, since myopic robots are weaker than non-myopic robots, many impossibility results are given for myopic robots.

To improve the task solvability, myopic robots with persistent visible light [7], called myopic *luminous* robots, have attracted a lot of attention. Each myopic luminous robot is equipped with a light device that can emit a constant number of colors to other robots, a single color at a time. The light color is persistent, *i.e.*, it is not automatically reset at the end of each cycle, and hence it can be used as a constant-space memory.

Ooshita and Tixeuil [20] studied the perpetual and terminating exploration of rings for $\phi = 1$ in the synchronous (FSYNC), semi-synchronous (SSYNC), and asynchronous (ASYNC) models. They showed that the number of robots required to achieve the tasks can be reduced compared to non-luminous robots. Nagahama et al. [19] studied the same problem in case of $\phi \geq 2$ and showed that, in the SSYNC and ASYNC models, the number of robots required to achieve the tasks can be reduced compared to the case of $\phi = 1$.

Bramas et al. studied the exploration of an infinite grid with myopic luminous and non-luminous robots in the FSYNC model [3,4]. Here they propose algorithms so that every node of an infinite grid is visited by a robot at least once. In [3] robots agree on a *common chirality*, *i.e.*, robots agree on common clockwise and counterclockwise directions. Bramas et al. [5] also studied the perpetual exploration of a (finite) grid with myopic luminous and non-luminous robots in the FSYNC model on the assumption that robots agree on a common chirality. Algorithms proposed in [5] have additional nice properties: they work even if robots are opaque (*i.e.*, a robot is able to see another robot only if no other robot lies in the line segment joining them), and they are exclusive (*i.e.*, no two robots occupy a single node during the execution). This work also describes the way to extend their algorithms to acheive the terminating exploration and/or to work in the SSYNC and ASYNC models. More concretely, this gives three algorithms to achieve the terminating exploration of a grid in case of a common chirality: algorithms for two robots with $\phi = 1$ and six colors in the FSYNC model, two robots with $\phi = 2$ and five colors in the FSYNC model, and two robots with $\phi = 2$ and six colors in the SSYNC and ASYNC models. However algorithms with a fewer number of colors or no common chirality are not known yet.

1.2 Our contributions

We focus on the terminating exploration of a (finite) grid with myopic luminous and non-luminous robots, and clarify lower and upper bounds of the required number of robots in various assumptions

Table 1: Terminating grid exploration with myopic robots. Bold texts indicate our contributions, and mark $*$ means the algorithm is optimal in terms of the number of robots. Notation ϕ represents the visible distance of a robot, and ℓ represents the number of colors.

Synchrony	ϕ	ℓ	Common chirality	#required robots Lower bound		Upper bound	
FSYNC	2	2	yes	2	[5]	**2***	§ 4.2.1
			no	2	[5]	**3**	§ 4.2.2
		1	yes	3	[5]	**3***	§ 4.2.3
			no	3	[5]	**4**	§ 4.2.4
	1	3	yes	2	[5]	**2***	§ 4.2.5
			no	2	[5]	**4**	§ 4.2.6
		2	yes	3	[5]	**3***	§ 4.2.7
			no	3	[5]	**5**	§ 4.2.8
SSYNC ASYNC	2	3	yes	2	[5]	**2***	§ 4.3.1
			no	2	[5]	**3**	§ 4.3.2
		2	yes	2	[5]	**3**	§ 4.3.3
			no	2	[5]	**4**	§ 4.3.4
	1	3	yes	**3**	§ 3	**3***	§ 4.3.5
			no	**3**	§ 3	**6**	§ 4.3.6

of synchrony, visible distance ϕ, the number of colors, and a chirality. Table 1 summarizes our contributions.

First, we prove that, in the SSYNC and ASYNC models, three myopic robots are necessary to achieve the terminating exploration of a grid if $\phi = 1$ holds. Note that this lower bound also holds for the perpetual exploration because we prove that robots cannot visit some nodes of a grid in this case. Other lower bounds in Table 1 are given by Bramas et al. [5]. They are originally given as impossibility results for the perpetual exploration, however they still hold for the terminating exploration. This is because Bramas et al. prove that, if the number of robots is smaller in each assumption, robots cannot visit some nodes.

Second, we propose fourteen algorithms to achieve the terminating exploration of a grid in various assumptions in Table 1. To the best of our knowledge, they are the first algorithms that achieve the terminating exploration of a grid by myopic robots with at most three colors and/or with no common chirality. In addition, six proposed algorithms are optimal in terms of the number of robots. Every proposed algorithm starts from a designated initial configuration such that all robots form a designated formation on one of four corners. When robots start from the northwest corner, they repeat the following behaviors until they terminate in the south end: 1) Robots go straight to the east, 2) go one step south and turn west, 3) go straight to the west end, and 4) go one step south and turn east.

2 Preliminaries

2.1 System model

The system consists of k mobile robots and a simple connected graph $G = (V, E)$, where V is a set of nodes and E is a set of edges. In this paper, we assume that G is a finite $m \times n$ grid (or a grid, for short) where m and n are two positive integers, *i.e.*, G satisfies the following conditions:

- $V = \{v_{i,j} \mid i \in \{0, 1, \ldots, m-1\},\ j \in \{0, 1, \ldots, n-1\}\}$
- $E = \{(v_{i,j}, v_{i',j'}) \mid v_{i,j}, v_{i',j'} \in V,\ |i - i'| + |j - j'| = 1\}$

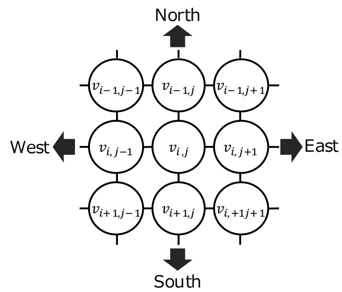


Figure 1: Global directions on a grid

The indices of nodes are used for notation purposes only and robots do not know them. Neither nodes nor edges have identifiers or labels, and consequently robots cannot distinguish nodes and cannot distinguish edges. Robots do not know m or n. Figure 1 shows global directions labeled by North, East, South, and West on a grid. Note that these directions are used only for explanations, and robots cannot access the global directions. Each robot is on a node of G at each instant. When a robot r is on a node v, we say r *occupies* v and v *hosts* r. The distance between two nodes is the number of edges in a shortest path between the nodes. The distance between two robots r_1 and r_2 is the distance between two nodes occupied by r_1 and r_2. Two robots r_1 and r_2 are neighbors if the distance between r_1 and r_2 is one.

Robots we consider have the following characteristics and capabilities. Robots are *identical*, that is, robots execute the same deterministic algorithm and do *not* have unique identifiers. Robots are *luminous*, that is, each robot has a light (or state) that is visible to itself and other robots. A robot can choose the color of its light from a discrete set Col. When the set Col is finite, ℓ denotes the number of available colors (*i.e.*, $\ell = |Col|$). Robots have no other persistent memory and cannot remember the history of past actions. Each robot can communicate by observing positions and colors of other robots (for collecting information), and by changing its color and moving (for sending information). Robots are *myopic*, that is, each robot r can observe positions and colors of robots within a fixed distance ϕ ($\phi > 0$ but $\phi \neq \infty$) from its current position. Since robots are identical, they share the same ϕ. Each robot distinguishes clockwise and counterclockwise directions according to its own *chirality*. The robots agree on a common clockwise direction if and only if they agree on a common chirality.

Each robot executes an algorithm by repeating three-phase cycles: Look, Compute, and Move phases. During the *Look* phase, the robot takes a snapshot of positions and colors of robots within distance ϕ. During the *Compute* phase, the robot computes its next color and movement according to the observation in the Look phase. The robot may change its color at the end of the Compute phase. If the robot decides to move, it moves instantaneously to a neighboring node during the *Move* phase. To model asynchrony of executions, we introduce the notion of *scheduler* that decides when each robot executes phases. When the scheduler makes robot r execute some phase, we say the scheduler activates the phase of r or simply activates r. We consider three types of synchronicity: the FSYNC (fully synchronous) model, the SSYNC (semi-synchronous) model, and the ASYNC (asynchronous) model. In all models, time is represented by an infinite sequence of instants $0, 1, 2, ...$ No robot has access to this global time. In the FSYNC and SSYNC models, all the robots that are activated at an instant t execute a full cycle synchronously and concurrently between t and $t+1$. In the FSYNC model, at every instant, the scheduler activates all robots. In the SSYNC model, at every instant, the scheduler selects a non-empty subset of robots and activates the selected robots. In the ASYNC model, the scheduler activates cycles of robots asynchronously: the time between Look, Compute, and Move phases is finite but unpredictable. Note that in the ASYNC model, a robot r can move based on the outdated snapshot obtained during the previous Look phase. Throughout the paper we assume that the scheduler is *fair*, that is, each robot is activated infinitely often.

2.2 Configuration, view, and algorithm

2.2.1 Configuration

A configuration represents positions and colors of all robots. At instant t, let $Q(t)$ be the set of occupied nodes, and let $M_{i,j}(t)$ be the multiset of colors of robots on node $v_{i,j} \in Q(t)$. A *configuration* $C(t)$ of the system at instant t is defined as $C(t) = \{(v_{i,j}, M_{i,j}(t)) \mid v_{i,j} \in Q(t)\}$. If t is clear from the context, we simply write Q, $M_{i,j}$ and C instead of $Q(t)$, $M_{i,j}(t)$, and $C(t)$, respectively.

2.2.2 View

When a robot takes a snapshot of its environment, it gets a *view* up to distance ϕ. Consider a robot r on node $v_{i,j}$. Let c_r be a color of r. We describe $M_{i',j'} = \perp$ if node $v_{i',j'}$ does not exist, that is, $i' \notin \{0, 1, \ldots, m-1\}$ or $j' \notin \{0, 1, \ldots, n-1\}$ holds. Since r does not know the global direction, it obtains one of the following four views in case of $\phi = 1$ and a common chirality:

- North view: $\mathcal{V}_{1,\nu} = (c_r, M_{i-1,j}, M_{i,j-1}, M_{i,j}, M_{i,j+1}, M_{i+1,j})$
- East view: $\mathcal{V}_{1,e} = (c_r, M_{i,j+1}, M_{i-1,j}, M_{i,j}, M_{i+1,j}, M_{i,j-1})$
- South view: $\mathcal{V}_{1,s} = (c_r, M_{i+1,j}, M_{i,j+1}, M_{i,j}, M_{i,j-1}, M_{i-1,j})$
- West view: $\mathcal{V}_{1,w} = (c_r, M_{i,j-1}, M_{i+1,j}, M_{i,j}, M_{i-1,j}, M_{i,j+1})$

In case of $\phi = 1$ and no common chirality, r obtains one of eight views, which include the above four views and the mirror images of them:

- Mirror image of $\mathcal{V}_{1,\nu}$: $\mathcal{V}_{1,\nu,\mu} = (c_r, M_{i-1,j}, M_{i,j+1}, M_{i,j}, M_{i,j-1}, M_{i+1,j})$
- Mirror image of $\mathcal{V}_{1,e}$: $\mathcal{V}_{1,e,\mu} = (c_r, M_{i,j+1}, M_{i+1,j}, M_{i,j}, M_{i-1,j}, M_{i,j-1})$
- Mirror image of $\mathcal{V}_{1,s}$: $\mathcal{V}_{1,s,\mu} = (c_r, M_{i+1,j}, M_{i,j-1}, M_{i,j}, M_{i,j+1}, M_{i-1,j})$
- Mirror image of $\mathcal{V}_{1,w}$: $\mathcal{V}_{1,w,\mu} = (c_r, M_{i,j-1}, M_{i-1,j}, M_{i,j}, M_{i+1,j}, M_{i,j+1})$

When r obtains one of the views, it cannot recognize which view it obtains, however it can compute other views by rotating and/or flipping the view. Hence, we assume that, in case of a common chirality, r obtains four views $\mathcal{V}_{1,\nu}, \mathcal{V}_{1,e}, \mathcal{V}_{1,s}, \mathcal{V}_{1,w}$ when it takes a snapshot. Note that r does not recognize which view corresponds to each of North, East, South, and West views. Similarly, we assume that, in case of no common chirality, r obtains eight views $\mathcal{V}_{1,\nu}, \mathcal{V}_{1,e}, \mathcal{V}_{1,s}, \mathcal{V}_{1,w}, V_{1,\nu,\mu}, \mathcal{V}_{1,e,\mu}, \mathcal{V}_{1,s,\mu}, \mathcal{V}_{1,w,\mu}$ when it takes a snapshot.

Similarly, in case of $\phi = 2$ and a common chirality, r obtains the following four views.

- North view: $\mathcal{V}_{2,\nu} = (c_r, M_{i-2,j}, M_{i-1,j-1}, M_{i-1,j}, M_{i-1,j+1}, M_{i,j-2}, M_{i,j-1}, M_{i,j}, M_{i,j+1}, M_{i,j+2}, M_{i+1,j-1}, M_{i+1,j}, M_{i+1,j+1}, M_{i+2,j})$
- East view: $\mathcal{V}_{2,e} = (c_r, M_{i,j+2}, M_{i-1,j+1}, M_{i,j+1}, M_{i+1,j+1}, M_{i-2,j}, M_{i-1,j}, M_{i,j}, M_{i+1,j}, M_{i+2,j}, M_{i-1,j-1}, M_{i,j-1}, M_{i+1,j-1}, M_{i,j-2})$
- South view: $\mathcal{V}_{2,s} = (c_r, M_{i+2,j}, M_{i+1,j+1}, M_{i+1,j}, M_{i+1,j-1}, M_{i,j+2}, M_{i,j+1}, M_{i,j}, M_{i,j-1}, M_{i,j-2}, M_{i-1,j+1}, M_{i-1,j}, M_{i-1,j-1}, M_{i-2,j})$
- West view: $\mathcal{V}_{2,w} = (c_r, M_{i,j-2}, M_{i+1,j-1}, M_{i,j-1}, M_{i-1,j-1}, M_{i+2,j}, M_{i+1,j}, M_{i,j}, M_{i-1,j}, M_{i-2,j}, M_{i+1,j+1}, M_{i,j+1}, M_{i-1,j+1}, M_{i,j+2})$

In case of $\phi = 2$ and no common chirality, r obtains eight views, which include the above four views and the mirror images of them:

- Mirror image of $\mathcal{V}_{2,\nu}$: $\mathcal{V}_{2,\nu,\mu} = (c_r, M_{i-2,j}, M_{i-1,j+1}, M_{i-1,j}, M_{i-1,j-1}, M_{i,j+2}, M_{i,j+1}, M_{i,j}, M_{i,j-1}, M_{i,j-2}, M_{i+1,j+1}, M_{i+1,j}, M_{i+1,j-1}, M_{i+2,j})$

- Mirror image of $\mathcal{V}_{2,e}$: $\mathcal{V}_{2,e,\mu} = (c_r, M_{i,j+2}, M_{i+1,j+1}, M_{i,j+1}, M_{i-1,j+1}, M_{i+2,j}, M_{i+1,j}, M_{i,j}, M_{i-1,j}, M_{i-2,j}, M_{i+1,j-1}, M_{i,j-1}, M_{i-1,j-1}, M_{i,j-2})$
- Mirror image of $\mathcal{V}_{2,s}$: $\mathcal{V}_{2,s,\mu} = (c_r, M_{i+2,j}, M_{i+1,j-1}, M_{i+1,j}, M_{i+1,j+1}, M_{i,j-2}, M_{i,j-1}, M_{i,j}, M_{i,j+1}, M_{i,j+2}, M_{i-1,j-1}, M_{i-1,j}, M_{i-1,j+1}, M_{i-2,j})$
- Mirror image of $\mathcal{V}_{2,w}$: $\mathcal{V}_{2,w,\mu} = (c_r, M_{i,j-2}, M_{i-1,j-1}, M_{i,j-1}, M_{i+1,j-1}, M_{i-2,j}, M_{i-1,j}, M_{i,j}, M_{i+1,j}, M_{i+2,j}, M_{i-1,j+1}, M_{i,j+1}, M_{i+1,j+1}, M_{i,j+2})$

2.2.3 Algorithm

An algorithm is described as a set of rules. Each rule is represented as a combination of a label, a guard, and an action. The guard represents possible views obtained by a robot. Recall that robot r obtains several views during the Look phase. If some view of robot r matches a guard in some rule, we say r is enabled. We also say the rule with the corresponding label is enabled. If r is enabled, r can execute the corresponding action (*i.e.*, change its color and/or move to its neighboring node) based on the directions of the matched view during Compute and Move phases. If several views of r match some guard or some view of r matches several guards, one combination of a view and a rule is selected by the scheduler.

2.3 Execution and problem

An execution from initial configuration C_0 is a maximal sequence of configurations $E = C_0, C_1, ..., C_i, ...$ such that, for any $j > 0$, we have *(i)* $C_{j-1} \neq C_j$, *(ii)* C_j is obtained from C_{j-1} after some robots move or change their colors, and *(iii)* for every robot r that moves or changes its color between C_{j-1} and C_j, there exists $0 \leq j' < j$ such that r takes its decision to move or change its color according to its algorithm and its view in $C_{j'}$. The term "*maximal*" means that the execution is either infinite or ends in a *terminal configuration*, *i.e.*, a configuration in which no robot is enabled.

A problem $\mathcal{P}$ is defined as a set of executions: An execution E solves $\mathcal{P}$ if $E \in \mathcal{P}$ holds. An algorithm $\mathcal{A}$ solves problem $\mathcal{P}$ from initial configuration C_0 if any execution from C_0 solves $\mathcal{P}$. We simply say an algorithm $\mathcal{A}$ solves problem $\mathcal{P}$ if there exists an initial configuration C_0 such that $\mathcal{A}$ solves $\mathcal{P}$ from C_0. In this paper, we consider the terminating exploration problem.

Definition 1 (**Terminating exploration problem**). *The terminating exploration is defined as a set of executions E such that 1) every node is visited by at least one robot in E and 2) there exists a suffix of E such that no robots are enabled.*

2.4 Descriptions

For simplicity, we describe a rule in an algorithm with a figure in Fig. 2. Figure 2(a) represents a rule of an algorithm in case of $\phi = 1$. Figure 2(b) represents a rule in case of $\phi = 2$. Each graph in Fig. 2 represents a guard. The guard in Fig. 2(a) represents a view $\mathcal{V}_1 = (c_r, M_{i-1,j}, M_{i,j-1}, M_{i,j}, M_{i,j+1}, M_{i+1,j})$, and similarly the guard in Fig. 2(b) represents a view $\mathcal{V}_2 = (c_r, M_{i-2,j}, M_{i-1,j-1}, M_{i-1,j}, M_{i-1,j+1}, M_{i,j-2}, M_{i,j-1}, M_{i,j}, M_{i,j+1}, M_{i,j+2}, M_{i+1,j-1}, M_{i+1,j}, M_{i+1,j+1}, M_{i+2,j})$. If $M_{i',j'} = \emptyset$ holds, we paint the corresponding node white instead of writing $\emptyset$. If $M_{i',j'} = \perp$ holds, we paint the corresponding node black instead of writing $\perp$. If both $\emptyset$ and $\perp$ are acceptable, we paint the corresponding node gray. If some view of robot r with visible distance ϕ matches $\mathcal{V}_\phi$, r is enabled. In this case, if the scheduler activates r, it executes an action represented by $c_{new}, Movement$. Notation c_{new} represents a new color of the robot. Notation $Movement$ can be $Idle$, $\leftarrow$, $\rightarrow$, $\uparrow$, $\downarrow$ and represents the movement: $Idle$ implies a robot does not move, and $\leftarrow$ (resp., $\rightarrow$, $\uparrow$, $\downarrow$) implies a robot moves toward the node corresponding to $M_{i,j-1}$ (resp., $M_{i,j+1}$, $M_{i-1,j}$, $M_{i+1,j}$) of the guard.

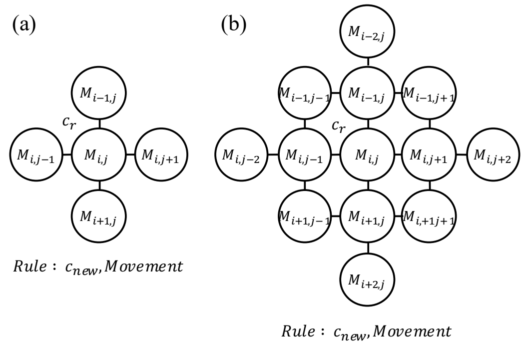


Figure 2: Description of a rule in an algorithm

3 An Impossibility result

In this section, we prove that, in the SSYNC model, two robots cannot achieve the terminating exploration if $\phi = 1$ holds. Since executions in the SSYNC model can happen in the ASYNC model, this impossibility also holds in the ASYNC model. This implies that, in case of $\phi = 1$, at least three robots are necessary to achieve the terminating exploration of grids in the SSYNC and ASYNC models. In the following, we use terms of end nodes and inner nodes. We say node v is an end node if the degree of v is smaller than four. We say node v is an inner node if the distance from v to every end node is at least three.

Theorem 1. *In case of $\phi = 1$ and $k = 2$, no algorithm solves the terminating exploration of grids in the SSYNC model. This holds regardless of the number of colors and a common chirality.*

Proof. For contradiction, we assume that such an algorithm $\mathcal{A}$ exists. Consider an execution $E = C_0, C_1, ...$ of $\mathcal{A}$ in a $m \times n$ grid G that satisfies $m \geq 9$ and $n \geq 9$. Let i be the minimum index such that some robot occupies an inner node at C_i. Let r_1 be a robot that occupies an inner node at C_i and r_2 be another robot. Let d be the distance between r_1 and r_2 at C_i. We consider two cases: (1) $d \geq 2$ and (2) $d \leq 1$.

Consider Case 1, that is, $d \geq 2$ holds. Let v_1 and v_2 be nodes that host r_1 and r_2, respectively, at C_i. We further consider two sub-cases: (1-1) v_2 is not an end node, and (1-2) v_2 is an end node. First assume that v_2 is not an end node (Case 1-1). In this case, we can define nodes v_1' and v_2' such that v_1' is a neighbor of v_1, v_2' is a neighbor of v_2, v_2' is not an end node, the distance between nodes w_1 and w_2 is at least two for any $w_1 \in \{v_1, v_1'\}$ and any $w_2 \in \{v_2, v_2'\}$. Then we can prove that the scheduler makes r_1 and r_2 stay on nodes in $\{v_1, v_1'\}$ and $\{v_2, v_2'\}$, respectively, forever after C_i. Consider configuration C such that r_1 and r_2 stay on nodes in $\{v_1, v_1'\}$ and $\{v_2, v_2'\}$, respectively. Since r_1 and r_2 cannot observe each other and they are not on end nodes, r_x ($x \in \{1, 2\}$) cannot distinguish directions, that is, r_x obtains four identical views when it takes a snapshot. This implies that, when r_x moves, the scheduler can decide which direction r_x moves toward. Hence, if r_1 moves, the scheduler can move r_1 to another node in $\{v_1, v_1'\}$. Similarly, if r_2 moves, the scheduler can move r_2 to another node in $\{v_2, v_2'\}$. This implies that, at the configuration after C, r_1 and r_2 stay on nodes in $\{v_1, v_1'\}$ and $\{v_2, v_2'\}$, respectively. Hence, inductively, after C_i, robots r_1 and r_2 continue to stay on nodes in $\{v_1, v_1'\}$ and $\{v_2, v_2'\}$, respectively. This means that robots can visit at most two inner nodes until C_i and visit at most two other inner nodes after C_i. Since the number of inner nodes in G is at least nine, robots cannot achieve the terminating exploration. Next assume that v_2 is an end node (Case 1-2). Let v_1' be an inner node that is a neighbor of v_1. Similarly to Case 1-1, we can prove that, if r_1 never observes r_2, r_1 continues to stay on nodes in $\{v_1, v_1'\}$. This implies that, to achieve the terminating exploration, r_2 moves toward r_1 or visits the remaining nodes by itself. In any case, r_2 leaves from end nodes, which reduces to Case 1-1.

Consider Case 2, that is, $d \leq 1$ holds. Let v_1 be a node that hosts r_1. Let v_2 be a node that hosts r_2 if $d = 1$, and a neighbor of v_1 if $d = 0$. We can prove that, as long as each robot moves toward another robot or stays on its current node, robots continue to stay on nodes in $\{v_1, v_2\}$: if two robots stay on different nodes, they can only move toward another node, and if two robots stay on a single node v_1 or v_2, the scheduler can move them to another node in $\{v_1, v_2\}$. Hence, eventually a robot moves to another node, say v_3, when the distance between two robots is one. In this moment, the scheduler activates only this robot. After the movement, the distance between r_1 and r_2 is two. Similarly to Case 1, after the configuration, robots can visit only two other inner nodes. This implies that robots can visit at most two inner nodes (v_1 and v_2) until C_i and visit at most three other inner nodes (v_3 and two other inner nodes) after C_i. Since the number of inner nodes in G is at least nine, they cannot achieve the terminating exploration.

This is a contradiction. □

Note that this impossibility result also holds for the perpetual exploration, where every robot must visit every node infinitely many times, because the proof of Theorem 1 shows that robots cannot visit some nodes in this case.

4 Terminating Grid Exploration Algorithms

4.1 Overview

In this subsection, we give the overview of our algorithms. All of our algorithms make robots start exploration from one of four corners. Without loss of generality, we assume robots start from the northwest corner in this manuscript. In this case, robots explore the grid according to the arrow in Fig. 3. In other words, robots start exploration from the northwest corner and repeat the following behaviors:

1. Proceed east: Robots go straight to the east end.

2. Turn west: They go one step south and turn west.

3. Proceed west: Robots go straight to the west end.

4. Turn east: They go one step south and turn east.

In each algorithm, we implement the behaviors of proceeding and turning. While proceeding, robots recognize their forward direction by their form. In the FSYNC model, since all robots are activated at every instant, they move forward at every instant and keep their initial form. The robots repeat this behavior until they reach the end of the grid. On the other hand, in the SSYNC and ASYNC models, not all robots are activated at the same time. For this reason, we propose the way to make robots move forward by moving a single robot at every instant.

The difficult part is to implement the behaviors of turning. Since robots do not know global directions, they must understand the south direction from the local information. We realize this in two different approaches. The first approach is to keep robots in two rows when proceeding east or west. By making different forms in north and south rows, robots distinguish the two directions. Mainly we use this approach in the case of no common chirality. The second approach is used only in the case of a common chirality. In this approach, robots change their form of proceeding depending on the directions. That is, robots distinguish the east and west directions by their form. In the case of a common chirality, robots can go south by turning right (resp. left) when they proceed east (resp. west). In the second approach, robots do not have to keep themselves in two rows when proceeding. This is the main reason why we can reduce the number of robots in the case of a common chirality.

4.2 Algorithms for the FSYNC model

In this subsection, we give terminating grid exploration algorithms for the FSYNC model.

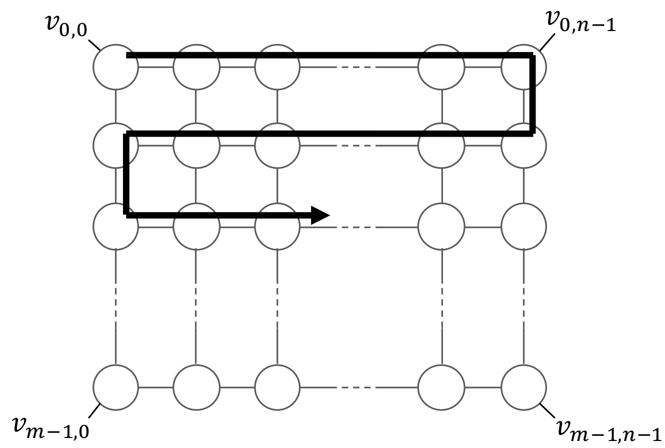


Figure 3: Route of grid exploration with our algorithm

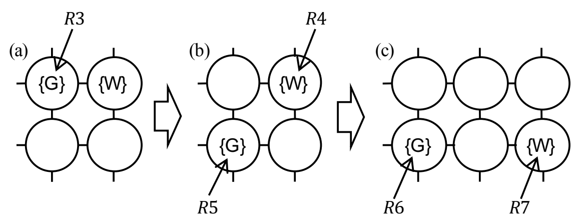


Figure 4: Turning west in an execution of Algorithm 1

4.2.1 FSYNC, $\phi = 2$, $\ell = 2$, a common chirality, and $k = 2$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 2$, $\ell = 2$, a common chirality, and $k = 2$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}\}$. The algorithm is given in Algorithm 1.

Proceeding east. From the initial configuration, robots with color G and W can execute rules $R1$ and $R2$, respectively. Hence, they proceed east while keeping the form.

Turning west. The process of turning west is shown in Fig. 4. After robots proceed east, they reach the east end of the grid (Fig. 4(a)). From this configuration, the robot with color G moves south by rule $R3$, and hence the configuration becomes one in Fig. 4(b). From this configuration, the robot with color W moves south by rule $R4$. At the same time, the robot with color G moves west by rule $R5$. Hence, the configuration becomes one in Fig. 4(c).

Proceeding west. From the configuration in Fig. 4(c), the robot with color G and the robot with color W can execute rules $R6$ and $R7$, respectively. Hence, they proceed west while keeping the form.

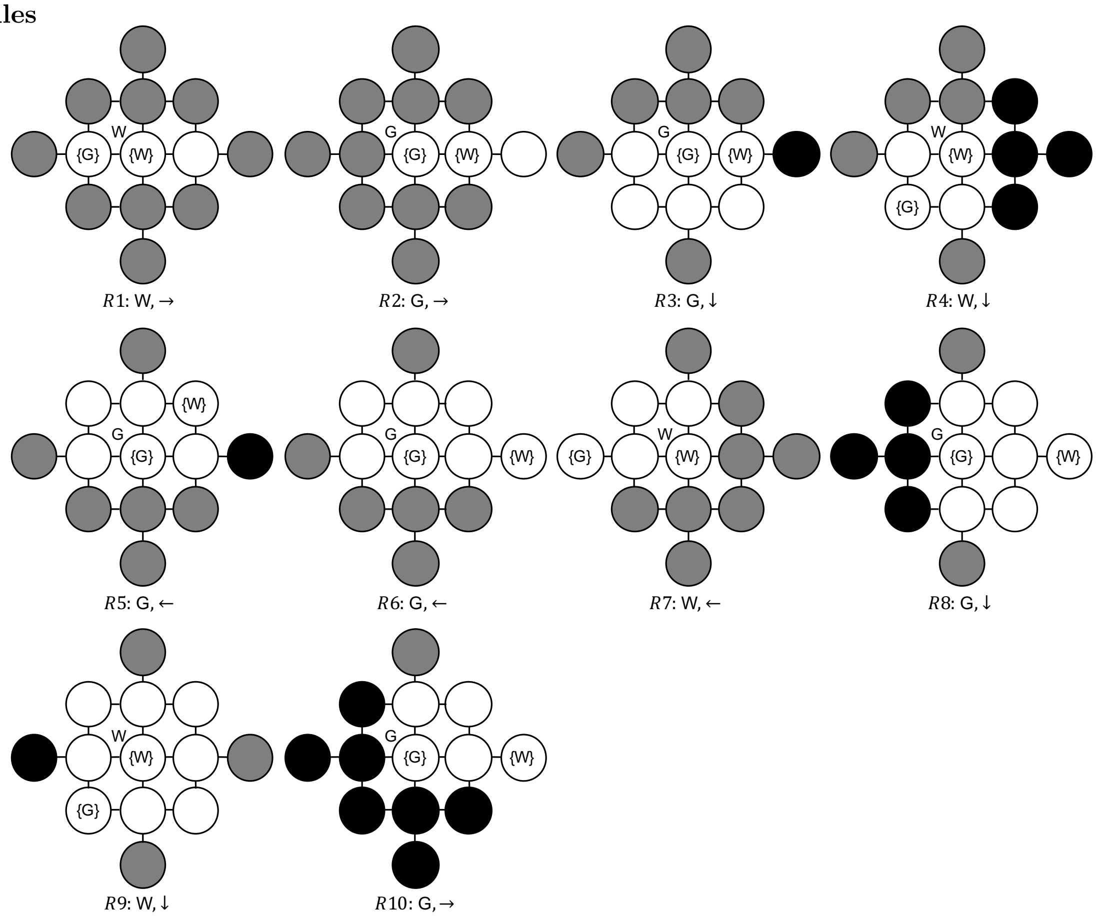
Turning east. The process of turning east is shown in Fig. 5. After robots proceed west, they reach the west end of the grid (Fig. 5(a)). From this configuration, the robot with color G moves south by rule $R8$. At the same time, the robot with color W moves by rule $R7$. Hence, the configuration becomes one in Fig. 5(b). From this configuration, the robot with color W moves south by rule $R9$, and hence the configuration becomes one in Fig. 5(c). From this configuration, two robots can proceed east again.

Algorithm 1 Fully Synchronous Terminating Exploration for $\phi = 2$, $\ell = 2$, $k = 2$ with a Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\})\}$

Rules



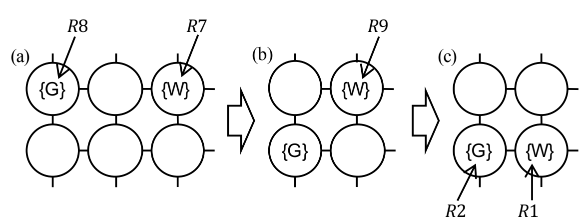


Figure 5: Turning east in an execution of Algorithm 1

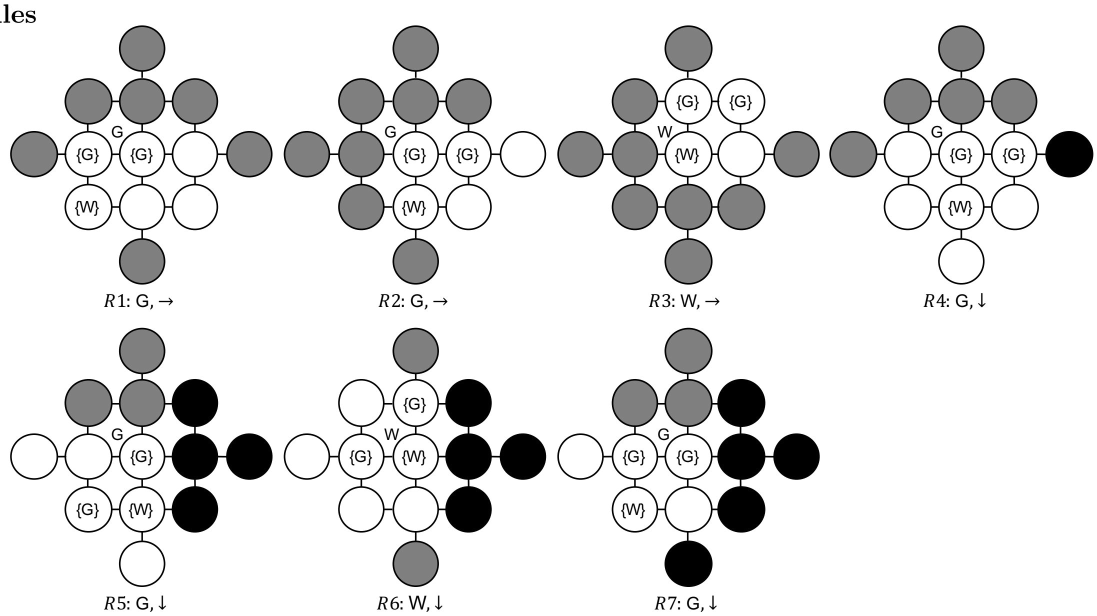
End of exploration. After robots visit all nodes and reach a south corner of the grid, the configuration becomes terminal. In case that m is odd, two robots visit the south end nodes while proceeding east, and hence they reach the southeast corner. Immediately after node $v_{m-1,n-1}$ is visited, the configuration is $\{(v_{m-1,n-2}, \{\mathsf{G}\}), (v_{m-1,n-1}, \{\mathsf{W}\})\}$. At this configuration, no robots are enabled. In case that m is even, two robots visit the south end nodes while proceeding west, and hence they reach the southwest corner. Immediately after node $v_{m-1,0}$ is visited, the configuration is $\{(v_{m-1,0}, \{\mathsf{G}\}), (v_{m-1,2}, \{\mathsf{W}\})\}$. From this configuration, robots with colors G and W move by rules $R10$ and $R7$, respectively. Hence, the configuration becomes $\{(v_{m-1,1}, \{\mathsf{G}, \mathsf{W}\})\}$. At this

Algorithm 2 Fully Synchronous Terminating Exploration for $\phi = 2$, $\ell = 2$, $k = 3$ with No Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{G}\}), (v_{1,0}, \{\mathsf{W}\})\}$

Rules



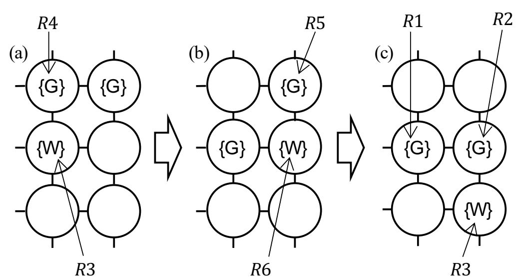


Figure 6: Turning west in an execution of Algorithm 2

configuration, no robots are enabled.

4.2.2 FSYNC, $\phi = 2$, $\ell = 2$, no common chirality, and $k = 3$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 2$, $\ell = 2$, no common chirality, and $k = 3$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}\}$. The algorithm is given in Algorithm 2.

Proceeding east and turning west. At the initial configuration, the robot on $v_{0,1}$ can execute rule $R1$, the robot on $v_{0,0}$ can execute rule $R2$, and the robot on $v_{1,0}$ can execute rule $R3$. By repeatedly executing those rules, robots proceed east while keeping the form. The process of turning west is shown in Fig. 6.

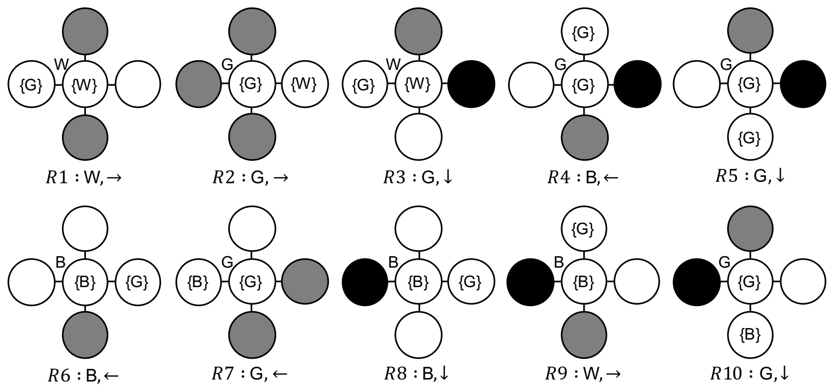
Proceeding west and turning east. The form of robots in Fig. 6(c) is a mirror image of the one that robots make to proceed east. Hence, robots proceed west and turn east with the same rules as

Algorithm 3 Fully Synchronous Terminating Exploration for $\phi = 1$, $\ell = 3$, $k = 2$ with Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\})\}$

Rules



proceeding east and turning west, respectively.

End of exploration. In case that m is odd, robots visit the south end nodes while proceeding west. Eventually, the configuration becomes $\{(v_{m-2,0}, \{\mathsf{G}\}), (v_{m-2,1}, \{\mathsf{G}\}), (v_{m-1,1}, \{\mathsf{W}\})\}$. Node $v_{m-1,0}$ has not been visited yet. From this configuration, robots on $v_{m-2,0}$ and $v_{m-1,1}$ move to $v_{m-1,0}$ by rules $R7$ and $R3$, respectively, and hence the configuration becomes $\{(v_{m-1,0}, \{\mathsf{G}, \mathsf{W}\}), (v_{m-2,1}, \{\mathsf{G}\})\}$. At this configuration, no robots are enabled. In case that m is even, robots terminate the algorithm similarly to the odd case.

4.2.3 FSYNC, $\phi = 2$, $\ell = 1$, a common chirality, and $k = 3$

In executions of Algorithm 1, robots do not change their colors and robots with different colors do not occupy a single node. Therefore, by representing the robot of color W in Algorithm 1 with two robots of color G, we can construct a terminating exploration algorithm in case of $\phi = 2$, $\ell = 1$, a common chirality, and $k = 3$.

4.2.4 FSYNC, $\phi = 2$, $\ell = 1$, no common chirality, and $k = 4$

In executions of Algorithm 2, robots do not change their colors and robots with different colors do not occupy a single node. Therefore, by representing the robot of color W in Algorithm 2 with two robots of color G, we can construct a terminating exploration algorithm in case of $\phi = 2$, $\ell = 1$, no common chirality, and $k = 4$.

4.2.5 FSYNC, $\phi = 1$, $\ell = 3$, a common chirality, and $k = 2$

We give a terminating exploration algorithm for $m \times n$ grids $(m \geq 2, n \geq 3)$ in case of $\phi = 1$, $\ell = 3$, a common chirality, and $k = 2$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}, \mathsf{B}\}$. The algorithm is given in Algorithm 3.

Proceeding east and turning west. From the initial configuration, robots with colors W and G can execute rules $R1$ and $R2$, respectively. Hence, they proceed east while keeping the form. The process of turning west is shown in Fig. 7.

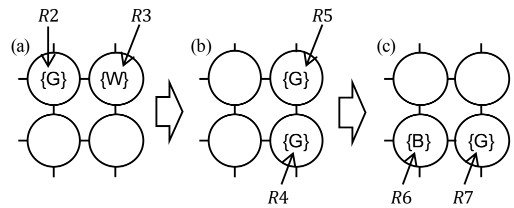


Figure 7: Turning west in an execution of Algorithm 3

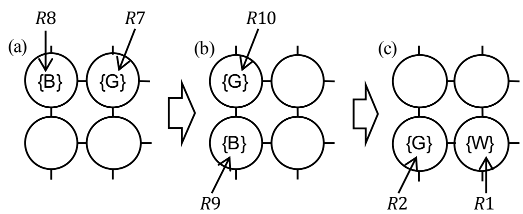


Figure 8: Turning east in an execution of Algorithm 3

Proceeding west and turning east. From the configuration in Fig. 7(c), the robot with color B and the robot with color G can execute rules $R6$ and $R7$, respectively. Hence, they proceed west while keeping the form. The process of turning east is shown in Fig. 8.

End of exploration. In case that m is odd, two robots visit the south end nodes while proceeding east, and hence they reach the southeast corner. Immediately after node $v_{m-1,n-1}$ is visited, the configuration is $\{(v_{m-1,n-2}, \{\mathsf{G}\}), (v_{m-1,n-1}, \{\mathsf{W}\})\}$. From this configuration, the robot with color G moves, and hence the configuration becomes $\{(v_{m-1,n-1}, \{\mathsf{G}, \mathsf{W}\})\}$. At this configuration, no robots are enabled. In case that m is even, two robots visit the south end nodes while proceeding west, and hence they reach the southwest corner. Immediately after node $v_{m-1,0}$ is visited, the configuration is $\{(v_{m-1,0}, \{\mathsf{B}\}), (v_{m-1,1}, \{\mathsf{G}\})\}$. From this configuration, the robot with color G moves by rule $R7$, and hence the configuration becomes $\{(v_{m-1,0}, \{\mathsf{G}, \mathsf{B}\})\}$. At this configuration, no robots are enabled.

4.2.6 FSYNC, $\phi = 1$, $\ell = 3$, no common chirality, and $k = 4$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 1$, $\ell = 3$, no common chirality, and $k = 4$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}, \mathsf{B}\}$. The algorithm is given in Algorithm 4.

Proceeding east and turning west. At the initial configuration, the robot on $v_{0,1}$ can execute rule $R1$, the robot on $v_{0,0}$ can execute rule $R2$, the robot on $v_{1,1}$ can execute rule $R3$, and the robot on $v_{1,0}$ can execute rule $R4$. By repeatedly executing those rules, robots proceed east while keeping the form. The process of turning west is shown in Fig. 9.

Proceeding west and turning east. The form of robots in Fig. 9(c) is a mirror image of the one that robots make to proceed east. Hence, robots proceed west and turn east with the same rules as proceeding east and turning west, respectively.

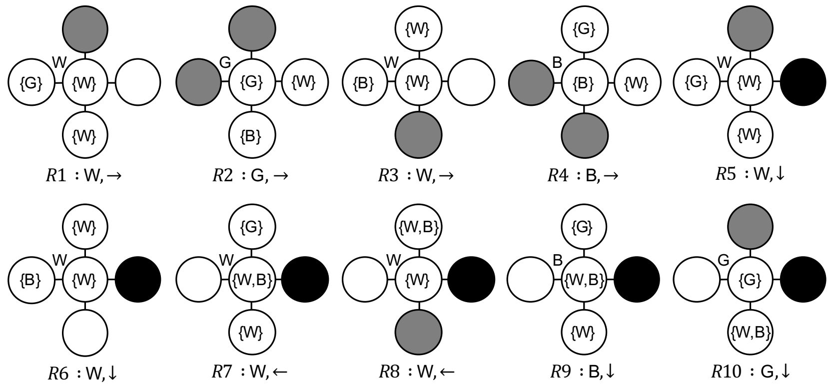
End of exploration. In case that m is odd, robots visit the south end nodes while proceeding west, and hence they reach the southwest corner. Immediately after node $v_{m-1,0}$ is visited,

Algorithm 4 Fully Synchronous Terminating Exploration for $\phi = 1$, $\ell = 3$, $k = 4$ with No Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\}), (v_{1,0}, \{\mathsf{B}\}), (v_{1,1}, \{\mathsf{W}\})\}$

Rules



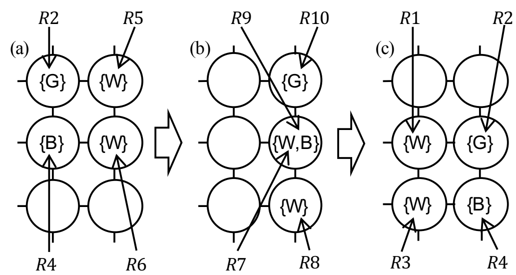


Figure 9: Turning west in an execution of Algorithm 4

the configuration is $\{(v_{m-2,0}, \{\mathsf{W}\}), (v_{m-2,1}, \{\mathsf{G}\}), (v_{m-1,0}, \{\mathsf{W}\}), (v_{m-1,1}, \{\mathsf{B}\})\}$. From this configuration, the robot on $v_{m-2,0}$ moves to $v_{m-1,0}$ by rule $R5$. At the same time, robots with colors G and B move west by rules $R2$ and $R4$, respectively. Hence, the configuration becomes $\{(v_{m-2,0}, \{\mathsf{G}\}), (v_{m-1,0}, \{\mathsf{W}, \mathsf{W}, \mathsf{B}\})\}$. At this configuration, no robots are enabled. In case that m is even, robots terminate the algorithm similarly to the odd case.

4.2.7 FSYNC, $\phi = 1$, $\ell = 2$, a common chirality, and $k = 3$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 1$, $\ell = 2$, a common chirality, and $k = 3$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}, \mathsf{B}\}$. The algorithm is given in Algorithm 5.

Proceeding east and turning west. At the initial configuration, the robot on $v_{0,1}$ can execute rule $R1$, the robot on $v_{0,0}$ can execute rule $R2$, and the robot on $v_{1,0}$ can execute rule $R3$. By repeatedly executing those rules, robots proceed east while keeping the form. The process of turning west is shown in Fig. 10.

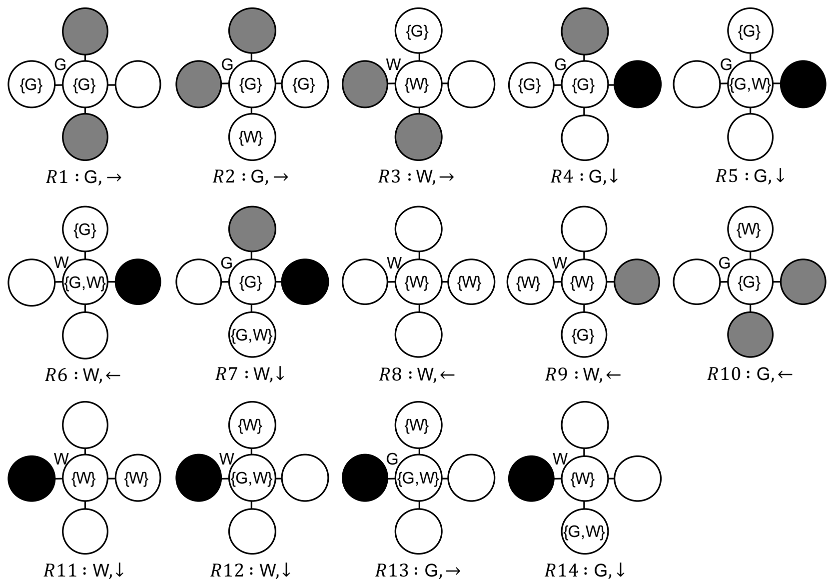
Proceeding west and turning east. At the configuration in Fig. 10(c), the robot on a west node can execute rule $R8$, the robot with color W on a east node can execute rule $R9$, and the robot with color G can execute rule $R10$. Hence, they proceed west while keeping the form. The process of turning east is shown in Fig. 11.

Algorithm 5 Fully Synchronous Terminating Exploration for $\phi = 1$, $\ell = 2$, $k = 3$ with Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{G}\}), (v_{1,0}, \{\mathsf{W}\})\}$

Rules



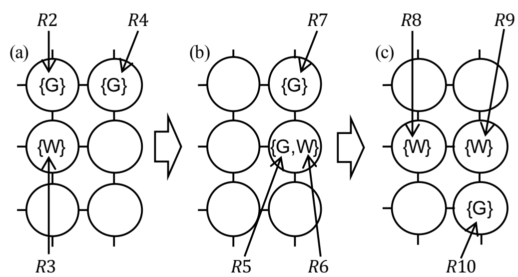


Figure 10: Turning west in an execution of Algorithm 5

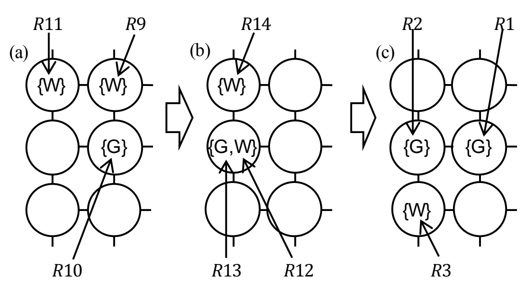


Figure 11: Turning east in an execution of Algorithm 5

End of exploration. In case that m is odd, robots visit the south end nodes while proceeding west. Eventually, the configuration becomes $\{(v_{m-2,0}, \{\mathsf{W}\}), (v_{m-2,1}, \{\mathsf{W}\}), (v_{m-1,1}, \{\mathsf{G}\})\}$. Node $v_{m-1,0}$ has not been visited yet. From this configuration, the robot on $v_{m-2,0}$ moves to $v_{m-1,0}$ by rule $R11$.

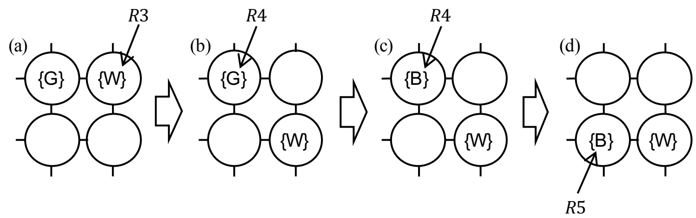


Figure 12: Turning west in an execution of Algorithm 6

At the same time, the other robots move west by rules $R9$ and $R10$, and hence the configuration becomes $\{(v_{m-2,0}, \{\mathsf{W}\}), (v_{m-1,0}, \{\mathsf{G}, \mathsf{W}\})\}$. From this configuration, the robot on $v_{m-2,0}$ moves to $v_{m-1,0}$ by rule $R14$, and hence the configuration becomes $\{(v_{m-1,0}, \{\mathsf{G}, \mathsf{G}, \mathsf{W}\})\}$. At this configuration, no robots are enabled. In case that m is even, robots visit the south end nodes while proceeding east. Eventually, the configuration becomes $\{(v_{m-2,n-2}, \{\mathsf{G}\}), (v_{m-2,n-1}, \{\mathsf{G}\}), (v_{m-1,n-2}, \{\mathsf{W}\})\}$. Node $v_{m-1,n-1}$ has not been visited yet. From this configuration, the robot on $v_{m-2,n-1}$ moves to $v_{m-1,n-1}$ by rule $R4$. At the same time, the other robots move east by rules $R2$ and $R3$, and hence the configuration becomes $\{(v_{m-2,n-1}, \{\mathsf{G}\}), (v_{m-1,n-1}, \{\mathsf{G}, \mathsf{W}\})\}$. From this configuration, the robot on $v_{m-2,n-1}$ moves to $v_{m-1,n-1}$ by rule $R7$, and hence the configuration becomes $\{(v_{m-1,n-1}, \{\mathsf{G}, \mathsf{W}, \mathsf{W}\})\}$. At this configuration, no robots are enabled.

4.2.8 FSYNC, $\phi = 1$, $\ell = 2$, no common chirality, and $k = 5$

In executions of Algorithm 4, robots do not change their colors and robots with colors G and B do not occupy a single node. Therefore, by representing the robot of color B in Algorithm 4 with two robots of color G, we can construct a terminating exploration algorithm in case of $\phi = 1$, $\ell = 2$, no common chirality, and $k = 5$.

4.3 Algorithms for the ASYNC model

In this subsection, we give terminating exploration algorithms for the ASYNC model. Clearly robots can achieve terminating exploration with those algorithms also in the SSYNC and FSYNC models.

4.3.1 ASYNC, $\phi = 2$, $\ell = 3$, a common chirality, and $k = 2$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 2$, $\ell = 3$, a common chirality, and $k = 2$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}, \mathsf{B}\}$. The algorithm is given in Algorithm 6.

Proceeding east. From the initial configuration, the robot with color W moves east by rule $R1$, and hence the configuration becomes $\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,2}, \{\mathsf{W}\})\}$. From this configuration, the robot with color G moves east by rule $R2$, and hence the configuration becomes $\{(v_{0,1}, \{\mathsf{G}\}), (v_{0,2}, \{\mathsf{W}\})\}$. After that, robots proceed east while keeping the form by repeatedly executing those rules.

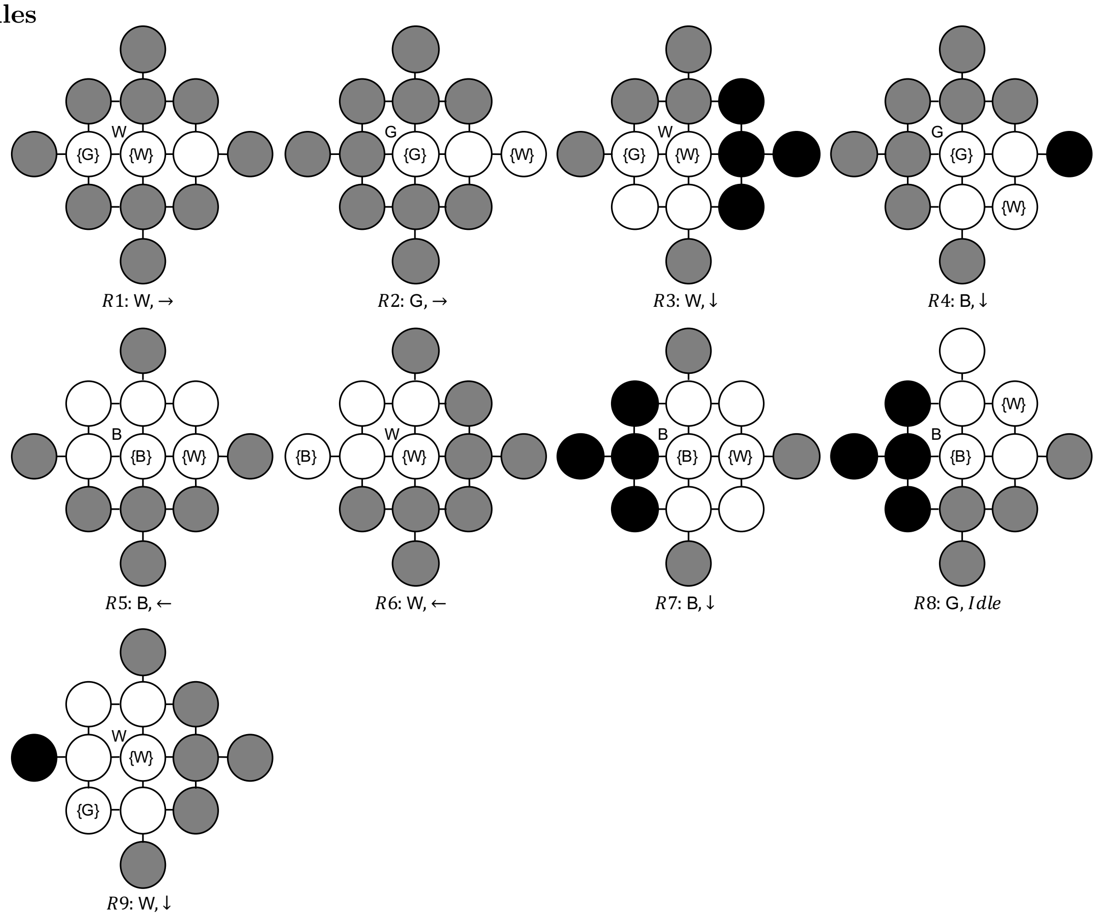
Turning west. The process of turning west is shown in Fig. 12. After robots proceed east, they reach the east end of the grid (Fig. 12(a)). From this configuration, the robot with color W moves south by rule $R3$, and hence the configuration becomes one in Fig. 12(b). From this configuration, the robot with color G changes its color to B and moves south by rule $R4$. In the ASYNC model, after the robot with color G changes its color, the other robot may observe the intermediate configuration (Fig. 12(c)). However, there are no rules that the other robot can execute in the intermediate configuration. Consequently, the configuration becomes one in Fig. 12(d).

Algorithm 6 Asynchronous Terminating Exploration for $\phi = 2$, $\ell = 3$, $k = 2$ with Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\})\}$

Rules



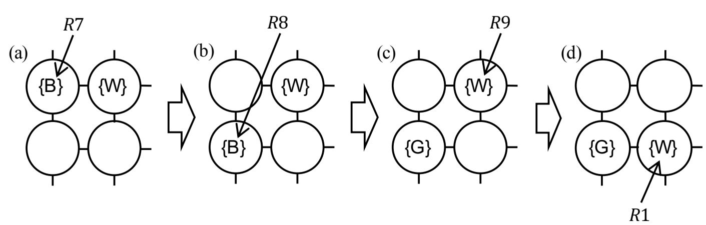


Figure 13: Turning east in an execution of Algorithm 6

Proceeding west. From the configuration in Fig. 12(d), the robot with color B moves west by rule $R5$. Next, the robot with color W moves west by rule $R6$. After that, robots proceed west while keeping the form by repeatedly executing those rules.

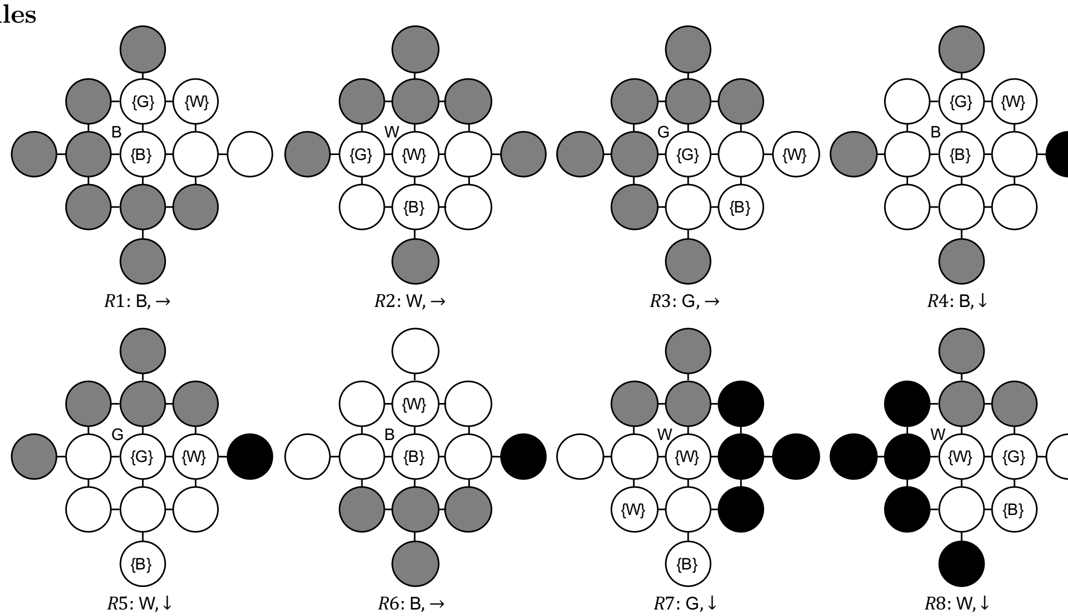
Turning east. The process of turning east is shown in Fig. 13. After robots proceed west, they reach the west end of the grid (Fig. 13(a)). From this configuration, the robot with color B moves south by rule $R7$, and hence the configuration becomes one in Fig. 13(b). From this configuration,

Algorithm 7 Asynchronous Terminating Exploration for $\phi = 2$, $\ell = 3$, $k = 3$ with No Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\}), (v_{1,0}, \{\mathsf{B}\})\}$

Rules



the robot with color B changes its color to G by rule $R8$, and hence the configuration becomes one in Fig. 13(c). From this configuration, the robot with color W moves south by rule $R9$, and hence the configuration becomes one in Fig. 13(d). From this configuration, two robots can proceed east again.

End of exploration. In case that m is odd, two robots visit the south end nodes while proceeding east, and hence they reach the southeast corner. Immediately after node $v_{m-1,n-1}$ is visited, the configuration is $\{(v_{m-1,n-2}, \{\mathsf{G}\}), (v_{m-1,n-1}, \{\mathsf{W}\})\}$. At this configuration, no robots are enabled. In case that m is even, two robots visit the south end nodes while proceeding west, and hence they reach the southwest corner. Immediately after node $v_{m-1,0}$ is visited, the configuration is $\{(v_{m-1,0}, \{\mathsf{B}\}), (v_{m-1,1}, \{\mathsf{W}\})\}$. At this configuration, no robots are enabled.

4.3.2 ASYNC, $\phi = 2$, $\ell = 3$, no common chirality, and $k = 3$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 2$, $\ell = 3$, a common chirality, and $k = 2$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}, \mathsf{B}\}$. The algorithm is given in Algorithm 7.

Proceeding east and turning west. From the initial configuration, the robot with color B moves by rule $R1$, and hence the configuration becomes $\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\}), (v_{1,1}, \{\mathsf{B}\})\}$. From this configuration, the robot with color W by rule $R2$, and hence the configuration becomes $\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,2}, \{\mathsf{W}\}), (v_{1,1}, \{\mathsf{B}\})\}$. From this configuration, the robot with color G by rule $R3$, and hence the configuration becomes $\{(v_{0,1}, \{\mathsf{G}\}), (v_{0,2}, \{\mathsf{W}\}), (v_{1,1}, \{\mathsf{B}\})\}$. After that, robots proceed east while keeping the form by repeatedly executing those rules. The process of turning west is shown in Fig. 14.

Proceeding west and turning east. The form of robots in Fig. 14(g) is a mirror image of the one that robots make to proceed east. Hence, robots proceed west and turn east with the same rules

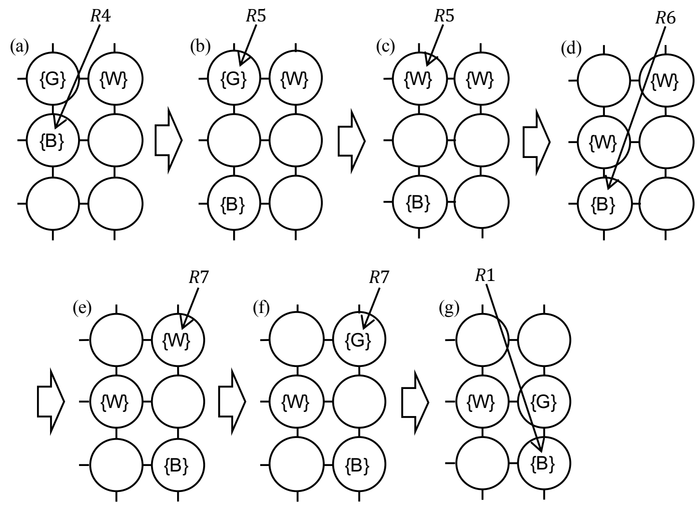


Figure 14: Turning west in an execution of Algorithm 7

as proceeding east and turning west, respectively.

End of exploration. In case that m is odd, robots visit the south end nodes while proceeding west. Eventually, the configuration becomes $\{(v_{m-2,0}, \{\mathsf{W}\}), (v_{m-2,1}, \{\mathsf{G}\}), (v_{m-1,1}, \{\mathsf{B}\})\}$. Node $v_{m-1,0}$ has not been visited yet. From this configuration, the robot with color W moves to $v_{m-1,0}$ by rule $R8$, and hence the configuration becomes $\{(v_{m-2,1}, \{\mathsf{G}\}), (v_{m-1,0}, \{\mathsf{W}\}), (v_{m-1,1}, \{\mathsf{B}\})\}$. At this configuration, no robots are enabled. In case that m is even, robots terminate the algorithm similarly to the odd case.

4.3.3 ASYNC, $\phi = 2$, $\ell = 2$, a common chirality, and $k = 3$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 2$, $\ell = 2$, a common chirality, and $k = 3$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}\}$. The algorithm is given in Algorithm 8.

Proceeding east and turning west. From the initial configuration, the robot with color W moves east by rule $R1$, and hence the configuration becomes $\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,2}, \{\mathsf{W}\}), (v_{1,0}, \{\mathsf{G}\})\}$. From this configuration, the robot on $v_{0,0}$ moves east by rule $R2$, and hence the configuration becomes $\{(v_{0,1}, \{\mathsf{G}\}), (v_{0,2}, \{\mathsf{W}\}), (v_{1,0}, \{\mathsf{G}\})\}$. From this configuration, the robot on $v_{1,0}$ moves east by rule $R3$, and hence the configuration becomes $\{(v_{0,1}, \{\mathsf{G}\}), (v_{0,2}, \{\mathsf{W}\}), (v_{1,1}, \{\mathsf{G}\})\}$. After that, robots proceed east while keeping the form by repeatedly executing those rules. The process of turning west is shown in Fig. 15.

Proceeding west and turning east. From the configuration in Fig. 15(f), the robot with color W on a west node moves west by rule $R9$. Next, the robot with color G moves west by rule $R10$. Then, the robot with color W on a east node moves west by rule $R11$. After that, robots proceed west while keeping the form by repeatedly executing those rules. The process of turning east in an execution of Algorithm 8 is shown in Fig. 16.

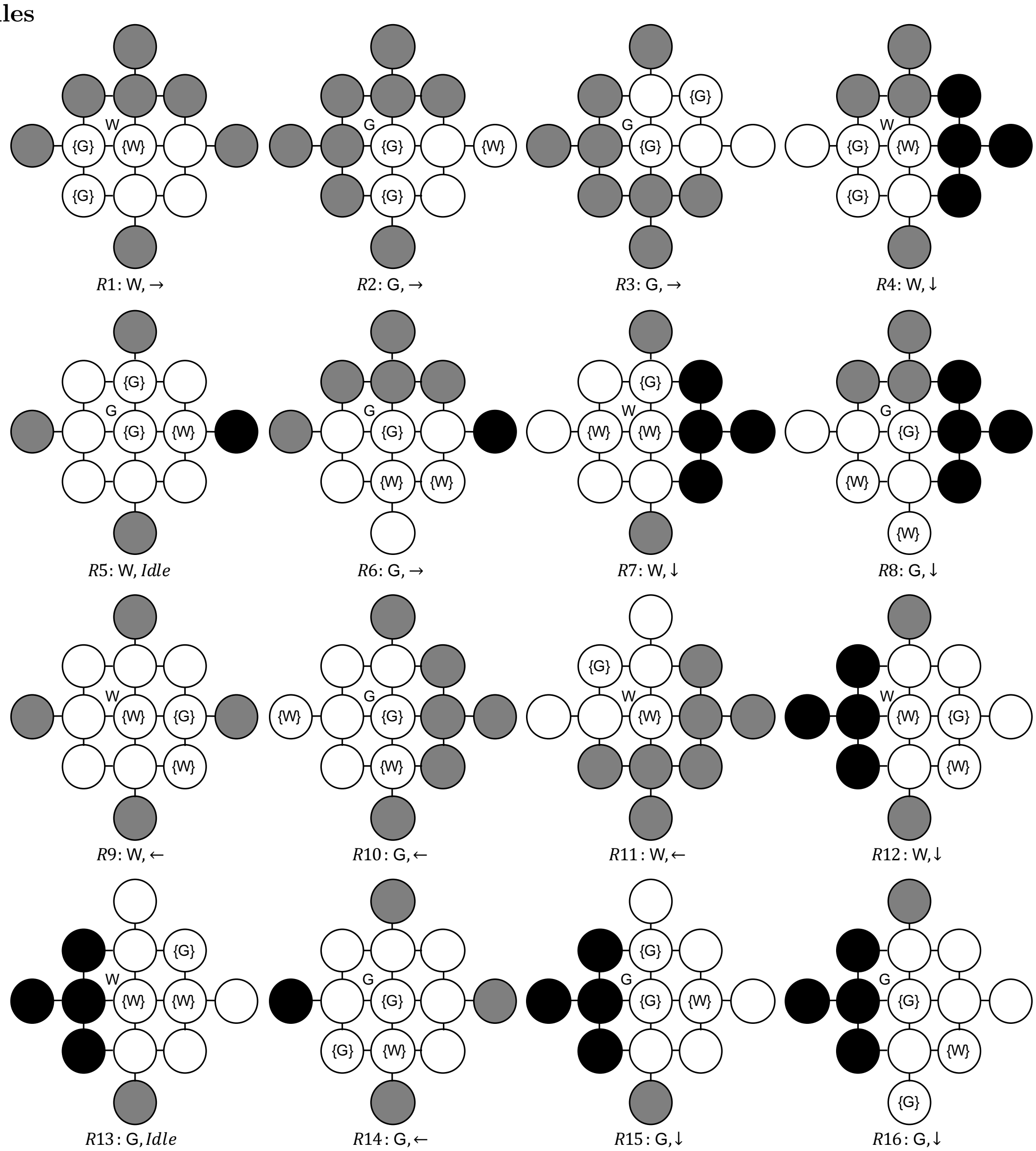
End of exploration. In case that m is odd, robots visit the south end nodes while proceeding west. Eventually, the configuration becomes $\{(v_{m-2,0}, \{\mathsf{W}\}), (v_{m-2,1}, \{\mathsf{G}\}), (v_{m-1,1}, \{\mathsf{W}\})\}$. Node $v_{m-1,0}$ has not been visited yet. From this configuration, the robot on $v_{m-2,0}$ moves to $v_{m-1,0}$ by rule $R12$, and hence the configuration becomes $\{(v_{m-2,1}, \{\mathsf{G}\}), (v_{m-1,0}, \{\mathsf{W}\}), (v_{m-1,1}, \{\mathsf{W}\})\}$.

Algorithm 8 Asynchronous Terminating Exploration for $\phi = 2$, $\ell = 2$, $k = 3$ with Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\}), (v_{1,0}, \{\mathsf{G}\})\}$

Rules



At this configuration, no robots are enabled. In case that m is even, robots visit the south end nodes while proceeding east. Eventually, the configuration becomes $\{(v_{m-2,n-2}, \{\mathsf{G}\}), (v_{m-2,n-1}, \{\mathsf{W}\}), (v_{m-1,n-2}, \{\mathsf{G}\})\}$. Node $v_{m-1,n-1}$ has not been visited yet. From this configuration, the robot on $v_{m-2,n-1}$ moves to $v_{m-1,n-1}$ by rule $R4$, and hence the configuration becomes $\{(v_{m-2,n-2}, \{\mathsf{G}\}), (v_{m-1,n-2}, \{\mathsf{G}\}), (v_{m-1,n-1}, \{\mathsf{W}\})\}$. At this configuration, no robots are enabled.

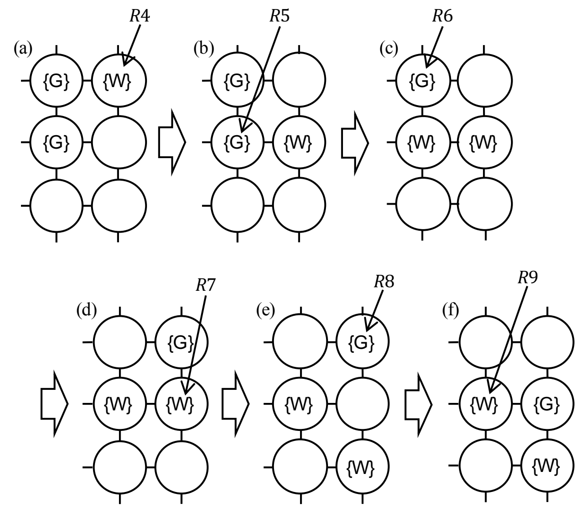


Figure 15: Turning west in an execution of Algorithm 8

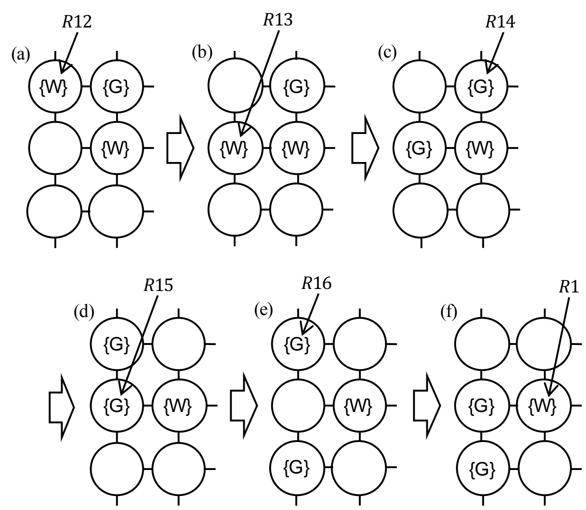


Figure 16: Turning east in an execution of Algorithm 8

4.3.4 ASYNC, $\phi = 2$, $\ell = 2$, no common chirality, and $k = 4$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 2$, $\ell = 2$, no common chirality, and $k = 4$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}\}$. The algorithm is given in Algorithm 9.

Proceeding east and turning west. The process of proceeding east is shown in Fig. 17, and the process of turning west is shown in Fig. 18.

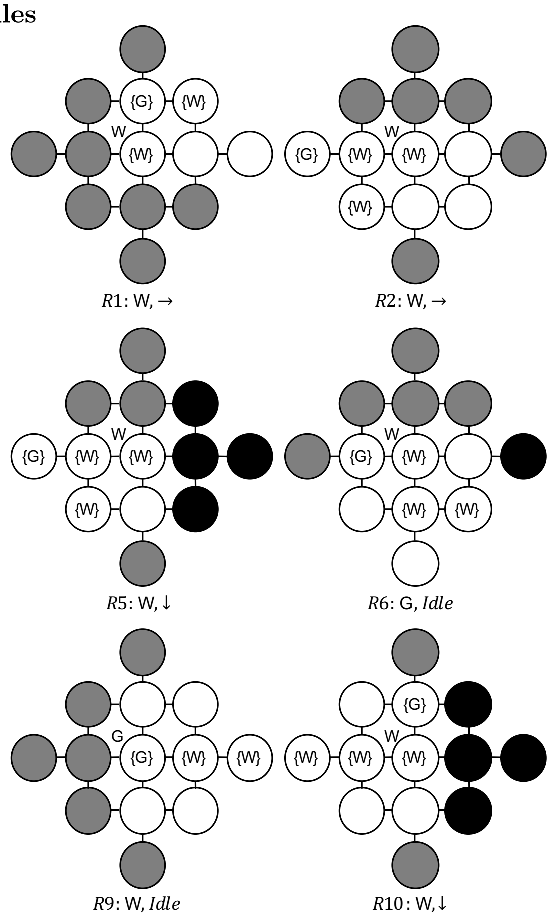
Proceeding west and turning east. The form of robots in Fig. 18(h) is a mirror image of the one that robots make to proceed east. Hence, robots proceed west and turn east with the same rules as proceeding east and turning west, respectively.

Algorithm 9 Asynchronous Terminating Exploration for $\phi = 2$, $\ell = 2$, $k = 4$ Without Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\}), (v_{0,2}, \{\mathsf{W}\}), (v_{1,0}, \{\mathsf{W}\})\}$

Rules



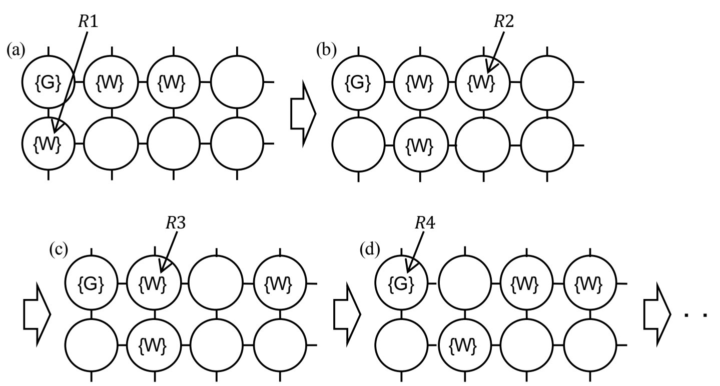


Figure 17: Proceeding east in an execution of Algorithm 9

End of exploration. In case that m is odd, robots visit the south end nodes while proceeding west. Eventually, the configuration becomes $\{(v_{m-2,0}, \{\mathsf{W}\}), (v_{m-2,1}, \{\mathsf{W}\}), (v_{m-2,2}, \{\mathsf{G}\}), (v_{m-1,1}, \{\mathsf{W}\})\}$. Node $v_{m-1,0}$ has not been visited yet. From this configuration, the robot

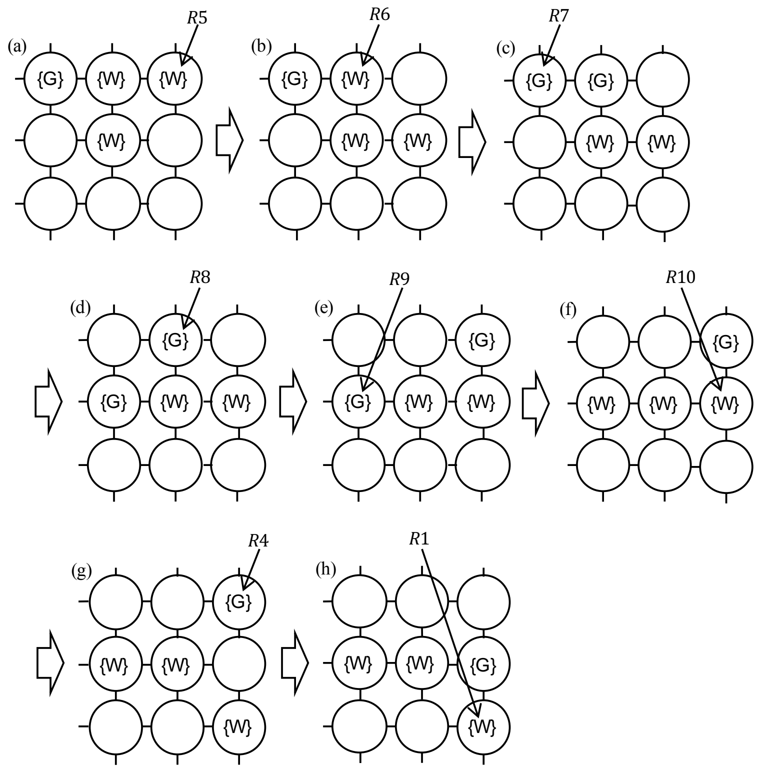


Figure 18: Turning west in an execution of Algorithm 9

on $v_{m-2,0}$ moves to $v_{m-1,0}$ by rule $R5$, and hence the configuration becomes $\{(v_{m-2,1}, \{\mathsf{W}\}), (v_{m-2,2}, \{\mathsf{G}\}), (v_{m-1,0}, \{\mathsf{W}\}), (v_{m-1,1}, \{\mathsf{W}\})\}$. At this configuration, no robots are enabled. In case that m is even, robots terminate the algorithm similarly to the odd case.

4.3.5 ASYNC, $\phi = 1$, $\ell = 3$, a common chirality, and $k = 3$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 2, n \geq 3$) in case of $\phi = 1$, $\ell = 3$, a common chirality, and $k = 3$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}, \mathsf{B}\}$. The algorithm is given in Algorithm 10.

Proceeding east and turning east The process of proceeding east is shown in Fig. 19. This is the procedure that is proposed as a ring exploration algorithm in [20]. The process of turning west is shown in Fig. 20.

Proceeding west and turning east The process of proceeding west is similar to that of proceeding east. Robots with colors W and B for proceeding west move in the same way as robots with colors G and W for proceeding east, respectively. The form in Fig. 20(h) corresponds to one in Fig. 19(b). Rules $R7$, $R8$, and $R9$ for proceeding west correspond to rules $R1$, $R2$, and $R3$ for proceeding east, respectively. Hence, robots proceed west keeping the form by repeatedly executing those rules. The process of turning east is shown in Fig. 21.

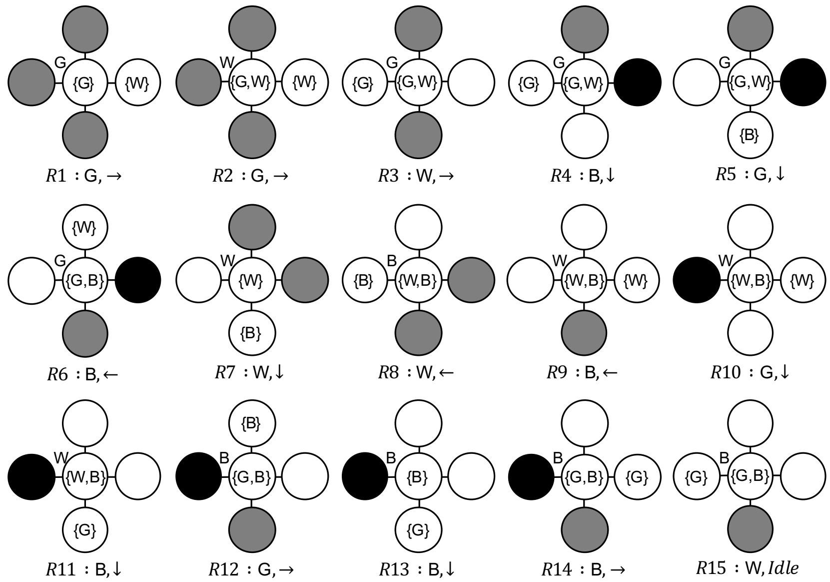
End of exploration. In case that m is odd, robots visit the south end nodes while proceeding east. Eventually, the configuration becomes $\{(v_{m-1,n-2}, \{\mathsf{G}\}), (v_{m-1,n-1}, \{\mathsf{G}, \mathsf{W}\})\}$. At this configuration, no robots are enabled. In case that m is even, robots visit the south end nodes while proceeding west.

Algorithm 10 Asynchronous Terminating Exploration for $\phi = 1$, $\ell = 3$, $k = 3$ with Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\}), (v_{0,2}, \{\mathsf{W}\})\}$

Rules



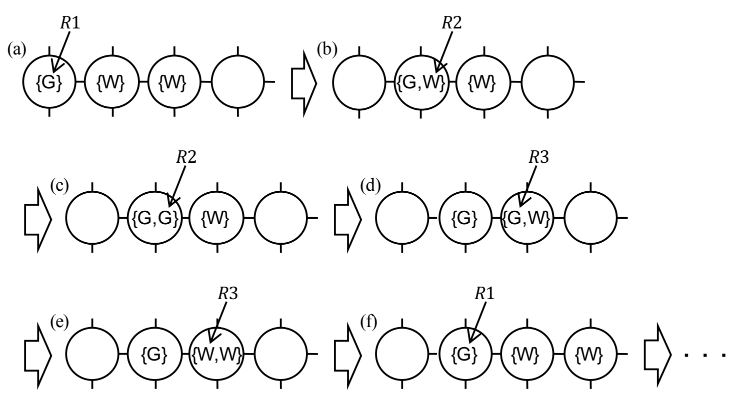


Figure 19: Proceeding east in an execution of Algorithm 10

Eventually, the configuration becomes $\{(v_{m-1,0}, \{\mathsf{W}, \mathsf{B}\}), (v_{m-1,1}, \{\mathsf{W}\})\}$. At this configuration, no robots are enabled.

4.3.6 ASYNC, $\phi = 1$, $\ell = 3$, no common chirality, and $k = 6$

We give a terminating exploration algorithm for $m \times n$ grids ($m \geq 3, n \geq 3$) in case of $\phi = 1$, $\ell = 3$, no common chirality, and $k = 6$. A set of colors is $Col = \{\mathsf{G}, \mathsf{W}, \mathsf{B}\}$. The algorithm is given in Algorithm 11.

Proceeding east. The process of proceeding east is shown in Fig. 22 and Fig. 23. At the initial configuration or at a configuration immediately after turning east, robots make the form in Fig. 22(a). After that, robots change the form from Fig. 22(a) to Fig. 22(h). Consider the configuration in Fig. 23(h) (identical to Fig. 22(h)). At this configuration, let r_1 be the robot with color W on a

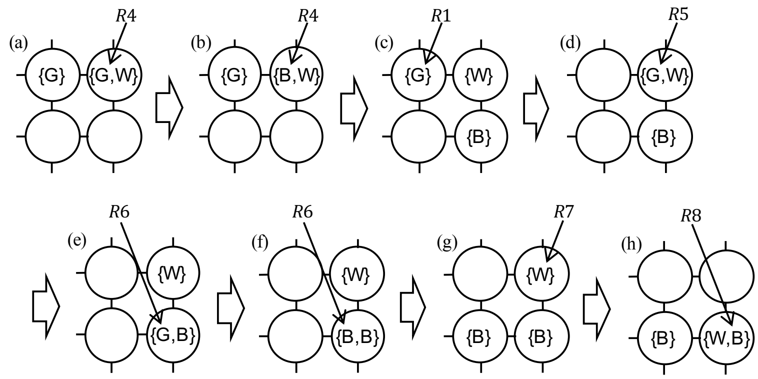


Figure 20: Turning west in an execution of Algorithm 10

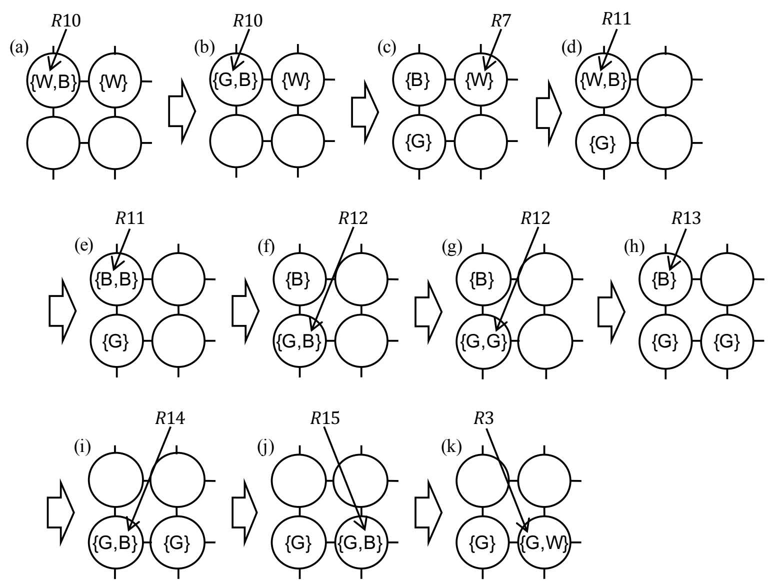


Figure 21: Turning east in an execution of Algorithm 10

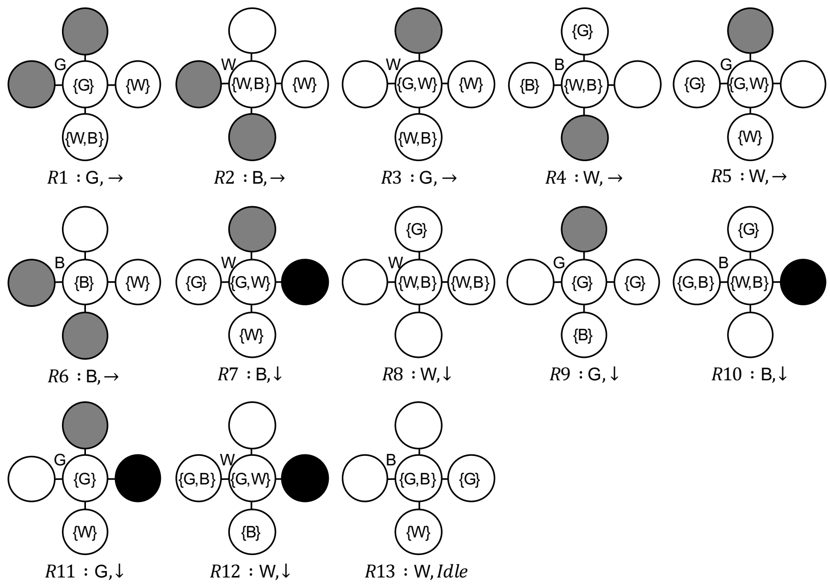
northeast node and let r_2 be the robot with color B. Then, r_1 can execute rule $R5$, and r_2 can execute rule $R6$. This implies that different executions exist after this configuration in the ASYNC model. However we can observe that the configuration eventually becomes one in Fig. 23(m) regardless of the scheduler. If r_2 finishes $R6$ before r_1 finishes the compute phase of $R5$, the configuration becomes one in Fig. 23(i). If r_1 finishes the compute phase of $R5$ before r_2 finishes $R6$, the configuration becomes one in Fig. 23(j). If r_1 finishes the compute phase of $R5$ and r_2 finishes $R6$ at the same time, the configuration becomes one in Fig. 23(k). At the configurations in Fig. 23(i) and Fig. 23(k), robots cannot execute rules except $R5$, and hence the configuration eventually becomes one in Fig. 23(m). At the configuration in Fig. 23(j), robots cannot execute rules except $R5$ and $R6$. From this configuration, if r_2 finishes $R6$ before r_1 finishes $R5$, the configuration becomes one in Fig. 23(k). If r_1 finishes $R5$ before r_2 finishes $R6$, the configuration becomes one in Fig. 23(l). If r_1 finishes $R5$ and r_2 finishes $R6$ at the same time, the configuration becomes one in Fig. 23(m). At the configurations in Fig. 23(l), robots cannot execute rules except $R6$, and hence the configuration eventually becomes one in Fig. 23(m). From the above discussion, the configuration eventually becomes one in Fig. 23(m) in any case.

Algorithm 11 Asynchronous Terminating Exploration for $\phi = 1$, $\ell = 3$, $k = 6$ Without Common Chirality

Initial configuration

$\{(v_{0,0}, \{\mathsf{G}\}), (v_{0,1}, \{\mathsf{W}\}), (v_{0,2}, \{\mathsf{W}\}), (v_{1,0}, \{\mathsf{W}, \mathsf{B}\}), (v_{1,1}, \{\mathsf{W}\})\}$

Rules



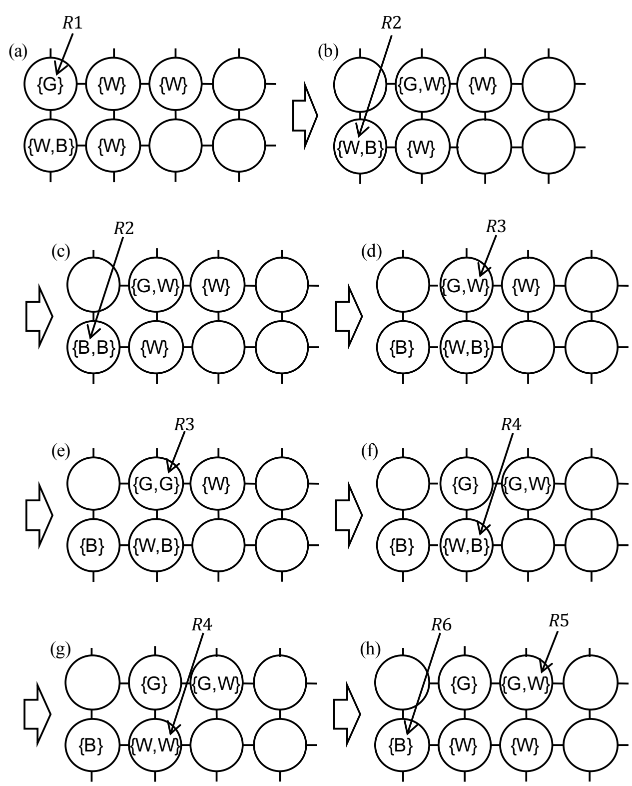


Figure 22: Proceeding east in executions of Algorithm 11 (I)

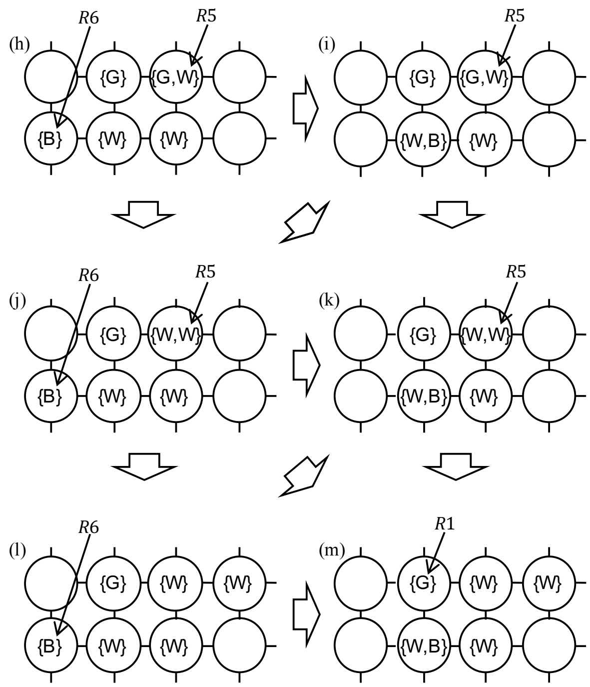


Figure 23: Proceeding east in executions of Algorithm 11 (II)

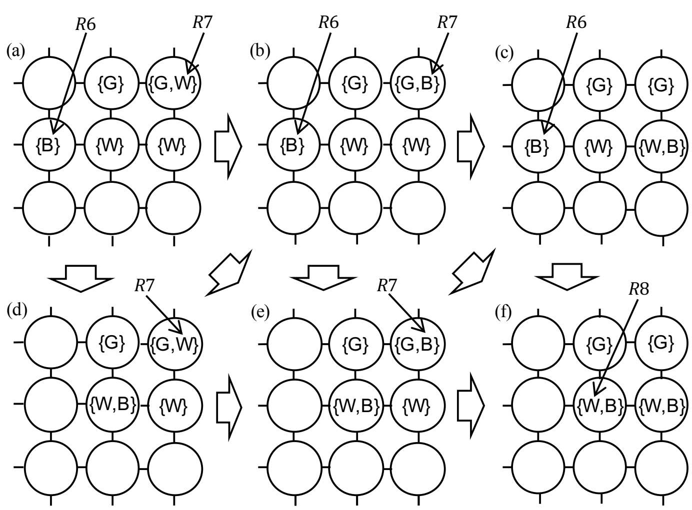


Figure 24: Turning west in an execution of Algorithm 11 (I)

Turning west. The process of turning west is shown in Fig. 24 and Fig. 25. After robots proceed east, they reach the east end of the grid, and the configuration becomes one in Fig. 24(a). At this configuration, let r_1 be the robot with color B, and let r_2 be the robot with color G on a northeast node. Then, r_1 can execute rule $R6$, and r_2 can execute rule $R7$. Although different executions exist after this configuration, we can observe that the configuration eventually becomes one in Fig. 24(f). After that, robots change the form from Fig. 25(f) (identical to Fig. 24(f)) to Fig. 25(n).

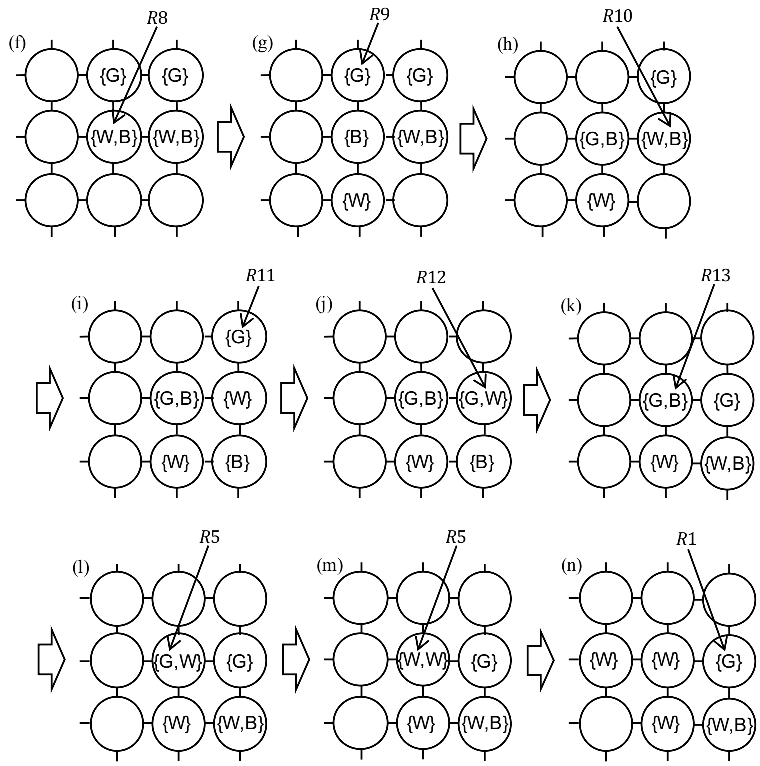


Figure 25: Turning west in an execution of Algorithm 11 (II)

Proceeding west and turning east. The form of robots in Fig. 25(n) is a mirror image of the one that robots make to proceed east. Hence, robots proceed west and turn east with the same rules as proceeding east and turning west, respectively.

End of exploration. In case that m is odd, robots visit the south end nodes while proceeding west. Eventually, the configuration becomes $\{(v_{m-2,0}, \{\mathsf{G}\}), (v_{m-2,1}, \{\mathsf{G}\}), (v_{m-1,0}, \{\mathsf{W}, \mathsf{B}\}), (v_{m-1,1}, \{\mathsf{W}, \mathsf{B}\})\}$. At this configuration, no robots are enabled. In case that m is even, robots terminate the algorithm similarly to the odd case.

5 Conclusions

In this paper, we have investigated terminating exploration algorithms for myopic robots in finite grids. First, we have proved that, in the SSYNC and ASYNC models, three myopic robots are necessary to achieve the terminating exploration of a grid if $\phi = 1$ holds. Second, we have proposed fourteen algorithms to achieve the terminating exploration of a grid in various assumptions of synchrony, visible distance, the number of colors, and a chirality. To the best of our knowledge, they are the first algorithms that achieve the terminating exploration of a grid by myopic robots with at most three colors and/or with no common chirality. In addition, six proposed algorithms are optimal in terms of the number of robots.

For the future work, it is interesting to close the gap between the lower and upper bounds of the number of required robots. Another interesting problem is to allow more general initial configurations so that robots can start exploration inside a grid. It is also interesting to consider other tasks and topologies with myopic luminous robots.

References

[1] L. Blin, A. Milani, M. Potop-Butucaru, and S. Tixeuil. Exclusive perpetual ring exploration without chirality. In *24th International Symposium on Distributed Computing*, pages 312–327, 2010.

[2] F. Bonnet, A. Milani, M. Potop-Butucaru, and S. Tixeuil. Asynchronous exclusive perpetual grid exploration without sense of direction. In *15th International Conference on Principles of Distributed Systems*, pages 251–265, 2011.

[3] Q. Bramas, S. Devismes, and P. Lafourcade. Infinite grid exploration by disoriented robots. In *8th International Conference on Networked Systems*, pages 129–145, 2020.

[4] Q. Bramas, P. Lafourcade, and S. Devismes. Finding water on poleless using melomaniac myopic chameleon robots. In *10th International Conference on Fun with Algorithms*, volume 157, pages 6:1–6:19, 2020.

[5] Q. Bramas, P. Lafourcade, and S. Devismes. Optimal exclusive perpetual grid exploration by luminous myopic opaque robots with common chirality. In *International Conference on Distributed Computing and Networking*, pages 76–85, 2021.

[6] J. Chalopin, P. Flocchini, B. Mans, and N. Santoro. Network exploration by silent and oblivious robots. In *36th International Workshop on Graph Theoretic Concepts in Computer Science*, pages 208–219, 2010.

[7] S. Das, P. Flocchini, G. Prencipe, N. Santoro, and M. Yamashita. Autonomous mobile robots with lights. *Theoretical Computer Science*, 609:171–184, 2016.

[8] A. K. Datta, A. Lamani, L. L. Larmore, and F. Petit. Ring exploration by oblivious agents with local vision. In *IEEE 33rd International Conference on Distributed Computing Systems*, pages 347–356, 2013.

[9] A. K. Datta, A. Lamani, L. L. Larmore, and F. Petit. Ring exploration by oblivious robots with vision limited to 2 or 3. In *15th International Symposium on Stabilization, Safety, and Security of Distributed Systems*, pages 363–366, 2013.

[10] S. Devismes, A. Lamani, F. Petit, P. Raymond, and S. Tixeuil. Optimal grid exploration by asynchronous oblivious robots. In *14th International Symposium on Stabilization, Safety, and Security of Distributed Systems*, pages 64–76, 2012.

[11] S. Devismes, A. Lamani, F. Petit, P. Raymond, and S. Tixeuil. Terminating exploration of a grid by an optimal number of asynchronous oblivious robots. *The Computer Journal*, 64(1):132–154, 2021.

[12] S. Devismes, A. Lamani, F. Petit, and S. Tixeuil. Optimal torus exploration by oblivious robots. *Computing*, 101(9):1241–1264, 2019.

[13] S. Devismes, F. Petit, and S. Tixeuil. Optimal probabilistic ring exploration by semi-synchronous oblivious robots. *Theoretical Computer Science*, 498:10–27, 2013.

[14] P. Flocchini, D. Ilcinkas, A. Pelc, and N. Santoro. Remembering without memory: Tree exploration by asynchronous oblivious robots. *Theoretical Computer Science*, 411(14-15):1583–1598, 2010.

[15] P. Flocchini, D. Ilcinkas, A. Pelc, and N. Santoro. How many oblivious robots can explore a line. *Information Processing Letters*, 111(20):1027–1031, 2011.

[16] P. Flocchini, D. Ilcinkas, A. Pelc, and N. Santoro. Computing without communicating: Ring exploration by asynchronous oblivious robots. *Algorithmica*, 65(3):562–583, 2013.

[17] P. Flocchini, G. Prencipe, and N. Santoro, editors. *Distributed Computing by Mobile Entities, Current Research in Moving and Computing*, volume 11340 of *Lecture Notes in Computer Science*. Springer, 2019.

[18] A. Lamani, M. Potop-Butucaru, and S. Tixeuil. Optimal deterministic ring exploration with oblivious asynchronous robots. In *International Colloquium on Structural Information and Communication Complexity*, pages 183–196, 2010.

[19] S. Nagahama, F. Ooshita, and M. Inoue. Ring exploration of myopic luminous robots with visibility more than one. In *International Symposium on Stabilizing, Safety, and Security of Distributed Systems*, pages 256–271, 2019.

[20] F. Ooshita and S. Tixeuil. Ring exploration with myopic luminous robots. *Information and Computation*, 2021.

[21] I. Suzuki and M. Yamashita. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM Journal on Computing*, 28(4):1347–1363, 1999.