

Circle Formation by Asynchronous Opaque Fat Robots on an Infinite Grid

Pritam Goswami^a, Manash Kumar Kundu^b, Satakshi Ghosh^a and Buddhadeb Sau^a

^aDepartment of Mathematics, Jadavpur University, Kolkata-700032, India; ^b Gayeshpur Government Polytechnic, Department of Science and Humanities, Kalyani, West Bengal - 741234, India

ARTICLE HISTORY

Compiled August 22, 2023

ABSTRACT

This study addresses the problem of "Circle Formation on an Infinite Grid by Fat Robots" (*CF_FAT_GRID*). Unlike prior work focused solely on point robots in discrete domain, it introduces fat robots to circle formation on an infinite grid, aligning with practicality as even small robots inherently possess dimensions. The algorithm, named *CIRCLE_FG*, resolves the *CF_FAT_GRID* problem using a swarm of fat luminous robots. Operating under an asynchronous scheduler, it achieves this with five distinct colors and by leveraging one-axis agreement among the robots.

KEYWORDS

Swarm Robot Algorithm; Circle Formation; Asynchronous; Infinite Grid; Luminous Robots; LCM Cycle

1. Introduction

Distributed algorithm has nowadays become a very emerging and interesting topic. The reasons behind this are very clear. Distributed systems offer a multitude of advantages over centralized systems, making them a preferred choice for various applications. One key benefit is enhanced reliability and fault tolerance. Additionally, distributed systems often exhibit superior scalability by allowing for easy expansion through the addition of new nodes, thus accommodating growing workloads and user demands seamlessly. This work focuses on swarm robot algorithm. A system consisting of swarm robots is an example of a distributed system. A robot in a swarm is a small computing unit with the capability to move. These robots are considered to be autonomous (i.e., there is no central control), anonymous (i.e., the robots do not have any unique identifiers), homogeneous (i.e., all robots execute the same distributed algorithm), and identical (i.e., all robots are physically indistinguishable). In a swarm robot system, a collection of such robots are considered on an environment (i.e., euclidean plane, circle, discrete network, etc.) and their aim is to execute some tasks (e.g. Gathering [1–3], Dispersion [4,5], Exploration [6,7], Pattern Formation[8,9] etc.) for which the mobility of the robots are needed. Swarm robotics has a huge application in many different scenarios such as patrolling, area coverage, network maintenance, etc.

Now one main direction of research in this field is to examine the optimal model of a robot in terms of different parameters. Memory and communication is one such parameter. There are mainly four robot models in the literature based on memory and communication. These four models are,

- *OBLLOT* Model: In this model, the robots do not have any persistent memory and there is no means of explicit communication between any two robots in the system.
- *FSTA* Model: In this robot model, the robots do not have any means of explicit communication between themselves. However, the robots have finite bits of persistent memory to remember their previous states.
- *FCOM* Model: In this model, the robots do not have any persistent memory but they can explicitly communicate with other robots using finite bit messages.
- *LUMI* Model: In this model, the robots have finite persistent memory and also can communicate with other robots using finite bit messages.

In every robot variant (excluding the *OBLLOT* model), each individual robot comes equipped with a light that possesses a finite range of colors. These colors serve as a method of both communication and memory among the robots. In the *FSTA* model, communication is absent, as robots cannot perceive the lights of other robots. However, a robot can observe its own light, which functions as its personal memory. In the *FCOM* model, robots lack the ability to view their own light, yet they can perceive the lights of their counterparts. Lastly, in the *LUMI* model, a robot has the capacity to see both its own light and the lights of other robots, allowing for comprehensive communication and memory to remember finitely many previous states.

Another parameter is robot vision. Categorized by visibility, there exist two distinct classifications of robot models: the *Non-Restricted Visibility Model* and the *Restricted Visibility Model*. In the *Non-Restricted Visibility Model*, a robot possesses the ability to observe the entire surroundings without any hindrance caused by other robots, and although extensively employed in existing literature ([1,8,9]) its practical feasibility is limited. Hence, the *Restricted Visibility Model* emerges. This model encompasses two potential visibility constraints. Firstly, a robot's visual range might be constrained, allowing it to only perceive objects within a certain distance, rather than the entire environment ([3,10,11]). Furthermore, visibility might be obstructed by the presence of other robots in the vicinity ([12–14]). If robots do not interfere with the visibility of other robots, then these types of robots are called *Transparent Robots*, otherwise, the robots are called *opaque Robots*.

Another aspect of the robot model considers the dimension of the robots. This perspective divides robots into two primary classifications. These two divisions encompass robots without physical dimensions and those possessing specific dimensions. In the dimensionless robot model, a robot is conceptualized as a mere point in space. Conversely, within the model involving robots with dimensions, these entities are represented as disks with some radius. While the point robot model prevails in academic discourse ([1,8,10,15]), it lacks suitability for real-world implementations. On the contrary, robots with dimensions, often referred to as *Fat robots*, prove more pragmatic for practical applications([16–18]).

At any moment in time, a robot can either be in idle state or in non-idle state. To move from the idle state to a non-idle state a robot first gets activated. The non-idle state of a robot is divided into three phases namely LOOK phase, COMPUTE phase, and MOVE phase in this order. After activation, a robot first executes LOOK phase where it takes a snapshot of its surroundings to get the positions of other visible robots

according to its own local coordinate system. Then it executes COMPUTE phase, where it runs a distributed algorithm with the information from LOOK phase as input. Then as the output of the algorithm, the robot gets a position. After this, it executes the MOVE phase, where the robot moves to the position of the output. If the output position is the same as the current position, then it does not move. After executing the MOVE phase, the robot returns to the idle state until it activates again and performs the LOOK-COMPUTE-MOVE phases. This is known as the LCM cycle.

Now since one execution of the algorithm depends on the position of other robots, activation of the robots plays a huge role in designing swarm robot algorithms. The activation of a robot is controlled by an entity called a scheduler. Mainly there are three types of scheduler models used massively in the literature. These three types of schedulers are,

- *Fully Synchronous Scheduler (FSYNC)*: In fully synchronous scheduler, the time is divided into rounds of equal lengths. Each round is subdivided into three intervals for LOOK, COMPUTE, and MOVE phase. These subdivisions are equal for each robot. Also, all robots are activated by an *FSYNC* scheduler at the beginning of each round.
- *Semi-Synchronous Scheduler (SSYNC)*: Semi synchronous scheduler is somewhat similar to the fully synchronous scheduler. The only difference here is that at the beginning of each round, an *SSYNC* scheduler activates a subset of robots. Note that if the subset is equal to the whole set of robots for each round, then we get the *FSYNC* scheduler. So *SSYNC* is more general than an *FSYNC* scheduler.
- *Asynchronous scheduler (ASYNC)*: In the asynchronous scheduler, there is no notion of rounds. In a particular moment, a robot can be either idle or can be executing any one of LOOK, COMPUTE or, MOVE phase. The duration of any of these phases is finite but unpredictable. This is the most general scheduler model which is also very feasible in terms of practical implementation.

In this work, we are interested in the problem of pattern formation. Now there are two types of pattern formation problems on which there is a vast literature. The two types of pattern formation problems are as follows:

- *Arbitrary Pattern Formation (APF)* where a set of robots, deployed in an environment, are provided with target pattern coordinates with respect to some global coordinate system. The robots do not have agreement to any global coordinate system however they each have their own local coordinate system. The aim is to provide a distributed algorithm for the robots so that first they can agree on a global coordinate and according to that coordinate system embed the target pattern and move to the target positions to form the pattern. ([8,16,17])
- *Geometric Shape Formation* (e.g., Line Formation, Circle Formation, Uniform Circle Formation etc.). In this problem, the robots only know which shape to form but they are not provided with specific coordinates of the target according to some global coordinate system. For example, in circle formation problems the robots know that they have to form a circle but they have no agreement initially about the radius and center of the circle. These are decided autonomously by executing the algorithm ([18–21]).

In this work, the main focus is this *Geometric Shape Formation* problem on a discrete domain. To be more specific, this work investigates the problem of *circle formation on an infinite grid by opaque fat robots (CF_FAT_GRID)* under an asynchronous sched-

uler. In the following subsection, we aim to offer a brief overview of how research on this issue has progressed, tracing its evolution from its inception to the contemporary landscape.

1.1. Related Works

Circle Formation is a well-known problem in distributed computing. It has been examined within both continuous and discrete contexts. The origin of the circle formation problem dates back to the work of Sugihara and Suzuki [19]. While they proposed a heuristic algorithm, it produced an approximate circle. Subsequently, another more accurate approximation algorithm was introduced by Tanaka et al. [15].

The problem, of strategically positioning robots equidistantly along the circle's circumference is known as Uniform Circle Formation. This notion was first explored by Suzuki and Yamashita [22]. Later, Defago and Konogaya [23] developed a circle formation algorithm that did not require robot orientation.

Defago and Konogaya's later work [20] presented a deterministic algorithm for non-uniform circle formation, where robots converge to an evenly spaced boundary configuration. Flocchini et al. [24] contributed an alternative uniform circle formation algorithm, removing certain assumptions, albeit restricted to cases where $n \neq 4$.

It is important to note that these aforementioned studies focused on the point robot model, with transparent robots operating in an Euclidean plane.

Feletti et al. [14] first considered opaque robot models for solving uniform circle formation. In their work, opaque luminous point robots are initially placed on a plane, equipped with a light possessing 5 distinct colors under the \mathcal{FSYNC} scheduler. Building on their prior research, their recent work [13] extended the solutions to incorporate asynchrony in the scheduler, using 19 colors for luminous, opaque point robots on a plane.

The utilization of opaque fat robots for circle formation was pioneered by the work in [18]. This algorithm tackled circle formation under a limited visibility model while considering global coordinate agreement.

Within discrete domain, the circle formation problem emerged in [12]. Here, the authors considered luminous opaque point robots on an infinite grid, functioning under an asynchronous scheduler. An algorithm was introduced by them where the robots are equipped with a light possessing seven colors and have one axis agreement. In [25], Ito et al. improved upon this concept by enhancing the existing algorithm for circle formation on an infinite grid. They achieved uniform circle formation with a diameter of $O(n)$ using five colors, and an alternative algorithm formed a uniform circle with a diameter of $O(n^2)$ with complete visibility, using just four colors. In a recent work [16], fat robots are introduced in a discrete domain (infinite grid) for solving arbitrary pattern formation problem

To our current knowledge, the application of fat robots in a discrete domain to solve the circle formation problem remains unexplored. Thus, our research focuses on investigating this challenge by employing opaque fat robots on an infinite grid. The subsequent subsection provides a concise problem description and outlines the contributions of this work.

1.2. Problem Description and Our Contribution

This work considers the problem of circle formation on an infinite grid by luminous opaque fat robots under an asynchronous scheduler using only five colors. In a discrete environment, robots may not be able to form an exact circle. Thus, an approximated circle is considered (Definition 2.1) to translate the circle formation problem from the Euclidean plane into the discrete domain (here infinite grid).

The robots are considered to be opaque disks of radius $rad \leq \frac{1}{2}$. The center of the disk is considered to be the position of the robot. The robots operate in a LOOK-COMPUTE-MOVE cycle under an asynchronous scheduler. Each robot has a light possessing five distinct colors. The robots do not agree on any global coordinate system; however, they agree on a common x -axis (i.e., a common left-right agreement). The problem requires the robots to agree on a circle CIR and then move to the circumference of CIR and terminate.

The main challenge here is the visibility. Since the robots are opaque, a robot can obstruct the visibility of another robot. As a result, a robot may not have the total information of the current situation. This is also the case for point opaque robots. But the visual hindrance becomes more severe if the robots have dimension. For example, considering the opaque point robot model, a robot r can not see another r' iff there is another robot r_o such that r, r' and r_o is colinear. But if we consider the opaque fat robots, a robot may not see another robot even when no three robots are colinear (Figure 1). In this work, the provided algorithm carefully handles this issue.

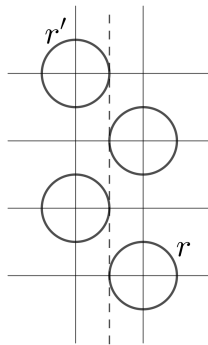


Figure 1. r can not see r' even though no three robots are colinear.

Another challenge here is avoiding collisions. In the Euclidean plane, there are infinitely many paths between any two points, whereas, in a discrete domain, there are only finitely many paths between any two vertices. Thus, avoiding collision is easier in the Euclidean plane compared to the discrete domain. This challenge has been dealt with through carefully sequenced movements of robots. To simulate sequential movements in an asynchronous environment, we have used luminous robots.

The algorithm provided in this work is called the *CIRCLE_FG* algorithm. The *CIRCLE_FG* algorithm solves the *CF_FAT_GRID* problem, considering a swarm of luminous, opaque fat robots with one-axis agreement and five distinct colors, from any initial configuration within finite time. A comparison table is provided so that readers can compare this work with some previously similar works.

Paper	Domain	Scheduler	Axis Agreement	Vision	Dimension	Circle Type	# colors
[13]	Plane	\mathcal{FSYNC}	No	Opaque	Point	Uniform	5
[13]	Plane	\mathcal{ASYNC}	No	Opaque	Point	Uniform	19
[21]	Grid	\mathcal{ASYNC}	One axis	Opaque	Point	Non-Uniform	7
[25]	Grid	\mathcal{ASYNC}	One axis	Opaque	Point	Uniform	5
This Paper	Grid	\mathcal{ASYNC}	One axis	Opaque	Fat	Non-Uniform	5

2. Model and Definitions

In this section, we first describe the models considered. Then, some definitions and notations are provided that have been used throughout the entire paper.

2.1. Model

Infinite Grid: An infinite grid is an infinite geometric graph $\mathcal{G} = (V, E)$ where vertices are points placed on \mathbb{R}^2 with coordinates $\{(a, b) : a \in \mathbb{Z} \text{ and } b \in \mathbb{Z}\}$. Also, two vertices are adjacent iff the Euclidean distance between them is one unit.

Robot Model: This work considers a set $\mathfrak{R} = \{r_1, r_2, \dots, r_k\}$ of k robots that are initially placed arbitrarily on k vertices of an infinite grid \mathcal{G} . The robots are considered to be autonomous, anonymous, homogeneous and identical. The robots are not point. In this work, the robots are considered to be disks of radius at most $\frac{1}{2}$ unit.

Axis and Unit Length Agreement: Each robot has a local coordinate system where origin is the position of itself. There is no global agreement on the coordinates however the robots agree on the direction and orientation of the x -axis which is parallel to one of the grid lines and also on unit length. That implies the robots agree on left, right and on the distance between any two points, but do not agree on up and down.

Visibility Model: In this work, the robots have unlimited but obstructed visibility. A robot r_i can see another robot r_j if and only if there is a point on the perimeter of r_i , say p_i and another point on the boundary of r_j , say p_j such that the line segment $\overline{p_i p_j}$ does not intersect at a point on any other robot. From this visibility model, it follows that if r_i can see r_j then r_j also sees r_i .

Memory and communication: In this work, the robots are considered to be of the \mathcal{LUMI} model. Thus the robots are able to remember some finite previous states and can communicate with visible robots using finite bit messages broadcasted using lights. Each of the robots are equipped with a light that can have 5 distinct colors from the set $Col = \{\text{off}, \text{chord}, \text{moving1}, \text{diameter}, \text{done}\}$ one at a time. Upon activation from idle state, a robot executes according to the LOOK-COMPUTE-MOVE cycle (LCM cycle) as described below.

LCM cycle: Upon activation with a color $C_1 \in Col$, a robot r first executes the LOOK phase. During this phase, the robot takes a snapshot of its surroundings and gets the positions of other visible robots according to its own coordinate system. Using this information as input, the robot then executes the algorithm in the COMPUTE phase. As an output of the algorithm, the robot gets a color $C_2 \in Col$ and a grid point at most one hop away from its current position. During the MOVE phase, the robot first changes its color to C_2 from C_1 (if different) and then moves to the new grid point (if different). After the MOVE phase is executed, the robot again returns to the idle state until it is activated again. A robot can move only along the edges of the grid, i.e., a robot can only move to one of the four adjacent vertices of its current position by

moving once. The move is also considered to be rigid and instantaneous, i.e., a robot is always seen on the grid points.

Scheduler Model: The scheduler considered in this work is the most general asynchronous scheduler ($\mathcal{ASYN}\mathcal{C}$). There is no agreement on rounds. The time taken by a robot to execute the LOOK phase, COMPUTE phase, MOVE phase, and the time a robot remains idle is finite but unpredictable.

2.2. Definitions and Notations

Definition 2.1 (Grid Circumference). Let \mathcal{CIR} be a circle on the plane on which the infinite grid is embedded. Let L_H be a horizontal grid line which intersects the circle \mathcal{CIR} on at most two points A and A' . Now,

- If A and A' are grid points, we say that only A and A' on L_H are on the circumference of \mathcal{CIR} .
- Otherwise, if A and A' are not grid points, we call a grid point (a, b) on the circumference of \mathcal{CIR} on L_H if the line joining the grid points (a, b) , $(a + 1, b)$ or (a, b) , $(a - 1, b)$ contains exactly one of A and A' .

The set of all such grid points on the circumference of the circle is called the Grid Circumference.

For simplicity by the term “on the circumference of the circle” we will always mean on some grid point which is in the set Grid Circumference.

Now, let us define the problem ($\mathcal{CF_FAT_GRID}$) formally .

Definition 2.2 (Problem Statement of $\mathcal{CF_FAT_GRID}$). Let a finite set of fat robots of same radius are initially located on distinct vertices of an infinite grid \mathcal{G} . We say that the Circle formation by Fat robots on infinite grid ($\mathcal{CF_FAT_GRID}$) is solved if there is an algorithm \mathcal{A} such that after executing \mathcal{A} , for finite time the following conditions are satisfied.

- (1) \mathcal{A} terminates. That is there exists a time t such that at time t all robots are terminated and does not execute \mathcal{A} anymore.
- (2) At time t , all robots are located on the grid circumference of the same circle, say \mathcal{CIR} .

Definition 2.3 (Configuration). Let $\mathcal{G} = (V, E)$ be an infinite grid. Let $f : V \rightarrow \{0, 1\} \times (Col \cup \{NULL\})$ be a function such that

$$f(a, b) = \begin{cases} (0, NULL) & \text{if there is no robot on } (a, b) \\ (1, c) & \text{if a robot is on } (a, b) \text{ with color } c \in Col \end{cases}$$

Then we call the pair (\mathcal{G}, f) a configuration which is denoted as \mathcal{C} . A configuration at time t is denoted as $\mathcal{C}(t)$.

We have used some notations throughout the paper. A list of these notations is mentioned in the following table.

Notation	Description
\mathcal{L}_1	First vertical line on left that contains at least one robot.
\mathcal{L}_i	i -th vertical line on the right starting from \mathcal{L}_1 .
$\mathcal{L}_V(r)$	The vertical line on which the robot r is located.
$\mathcal{L}_H(r)$	The horizontal line on which the robot r is located.
$\mathcal{L}_I(r)$	The left immediate vertical line of robot r which has at least one robot on it.
$\mathcal{R}_I(r)$	The right immediate vertical line of robot r which has at least one robot on it.
$H_L^O(r)$	Left open half for the robot r .
$H_L^C(r)$	Left closed half for the robot r (i.e $H_L^O(r) \cup \mathcal{L}_V(r)$).
$H_B^O(r)$	Bottom open half for the robot r .
$H_B^C(r)$	Bottom closed half for the robot r (i.e $H_B^O(r) \cup \mathcal{L}_H(r)$).
$H_U^O(r)$	Upper open half for the robot r .
$H_U^C(r)$	Upper closed half for the robot r (i.e $H_U^O(r) \cup \mathcal{L}_H(r)$).

Definition 2.4 (Phase 1 Final Configuration (P1FC)). We call a configuration a Phase 1 Final Configuration if the following conditions hold

- (1) \mathcal{L}_2 has exactly two robots, say r_1 and r_2 , with color **diameter**. All other robots are on either \mathcal{L}_1 or \mathcal{L}_3 with color **chord**.
- (2) All robots on \mathcal{L}_1 and \mathcal{L}_3 are strictly between $\mathcal{L}_H(r_1)$ and $\mathcal{L}_H(r_2)$.

Definition 2.5. Let t_b be the time when a robot moves first from the initial configuration. We say that \mathcal{L}_1 is *fixed* at a time $t_f > t_b$ if both of the following conditions hold:

- (1) $\forall t \in [t_b, t_f)$, $\mathcal{C}(t)$ has a robot that is not on \mathcal{L}_1 .
- (2) From time t_f onwards, no robot from \mathcal{L}_1 moves left until all robots move to \mathcal{L}_1 .

Definition 2.6 (Terminal Robot). In a configuration \mathcal{C} , a robot r is called a terminal robot on $\mathcal{L}_V(r)$ if there is no robot either above or below r on $\mathcal{L}_V(r)$.

3. CIRCLE_FG Algorithm

In this section, we propose an algorithm called *CIRCLE_FG* that solves the *CF_FAT_GRID* problem in finite time. First, we discuss the outline of the algorithm and then discuss it in detail later in this section.

Outline of the algorithm: The algorithm works in two phases. In Phase 1, the robots first form a vertical line, say L . The two extreme robots on the line change their color to **diameter**. The other robots on line L move according to the function *CHORDMOVE* to the vertical lines that are one hop away from L thus forming a P1FC (Figure 2). Note that all robots that now see both robots of color **diameter** agree on the diameter of the circle, and consequently, on the circle itself. So in Phase 1, the diameter is formed on which all robots can agree. Then in Phase 2, the robots with color **chord** first move to the vertical lines that are $\frac{d}{2}$ distance away from the agreed diameter on both sides of it (d is the length of the diameter) and change color to **off**. From this configuration, all robots change their color to **moving1** and move towards the diameter by forming a line on each vertical grid line in between. Let

L_m be a vertical line where all robots of color `moving1` to the left (or right) of the agreed diameter have formed a line. Starting from L_m , all robots that are strictly inside the agreed circle move further from the diameter after changing their color to `done` in order to reach their corresponding positions on the circle. Following this, the remaining robots on L_m move towards the agreed diameter and establish a new line on the next vertical grid line towards the diameter. In this manner, all robots move to their designated positions on the circle and terminate their movement. This entire process is illustrated in Figure 3. This is a very brief description of the algorithm. In the following two subsections, we will provide detailed descriptions of Phase 1 and Phase 2, along with explanations of the correctness of each phase.

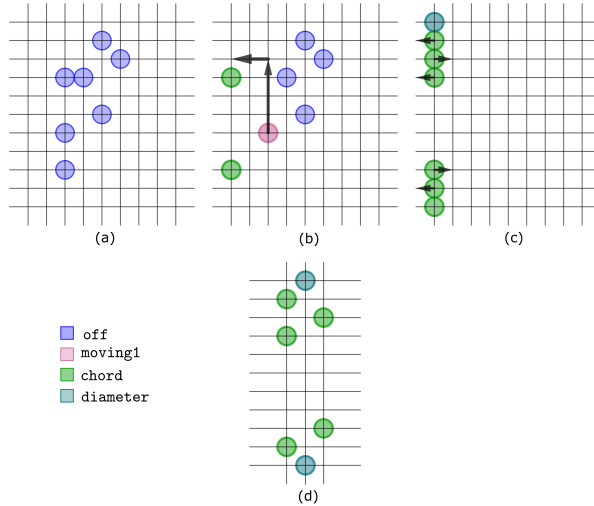


Figure 2. Outline of Phase 1: (a) Initial configuration $\mathcal{C}(0)$. (b) Two terminal robots of \mathcal{L}_1 in $\mathcal{C}(0)$ moved two hop and changes color to `chord`. A terminal robot of color `off` on nearest vertical line of \mathcal{L}_1 changes color to `moving1` and moves to \mathcal{L}_1 . (c) All robots of color `off` moves to \mathcal{L}_1 in a similar way and forms a single line. The terminal robots on this line changes color to `diameter` eventually.(here in this specific example, due to asynchrony only one terminal robot changes color to `diameter` first). Next all non terminal robots move either left or right. (d) P1FC is formed.

3.1. Phase 1

3.1.1. Brief description of Phase 1

In Phase 1, initially all the robots have color `off` and are placed arbitrarily on the grid. There can be at most two robots that are terminal on \mathcal{L}_1 of the initial configuration. Upon activation these robots will see their left open half is empty and one of their upper or bottom halves of their corresponding vertical line is empty. For this view the robot change their color to `moving1` from `off` and moves left shifting the line \mathcal{L}_1 . Except the terminal robots on \mathcal{L}_1 of the initial configuration other robots of color `off` change their color to `moving1` only when they are terminal on their corresponding vertical line and sees all robots on their left immediate vertical line has color `chord`. So, unless the terminal robots on \mathcal{L}_1 of the initial configuration change their color to `chord` no other robots do anything even if they are activated.

Let r_1 and r_2 be two terminal robots on \mathcal{L}_1 in the initial configuration. If \mathcal{L}_1 of the initial configuration contains more than two robots then there must be another robot except r_1 and r_2 , say r , with color `off`. Now upon activation let r_1 has moved left once.

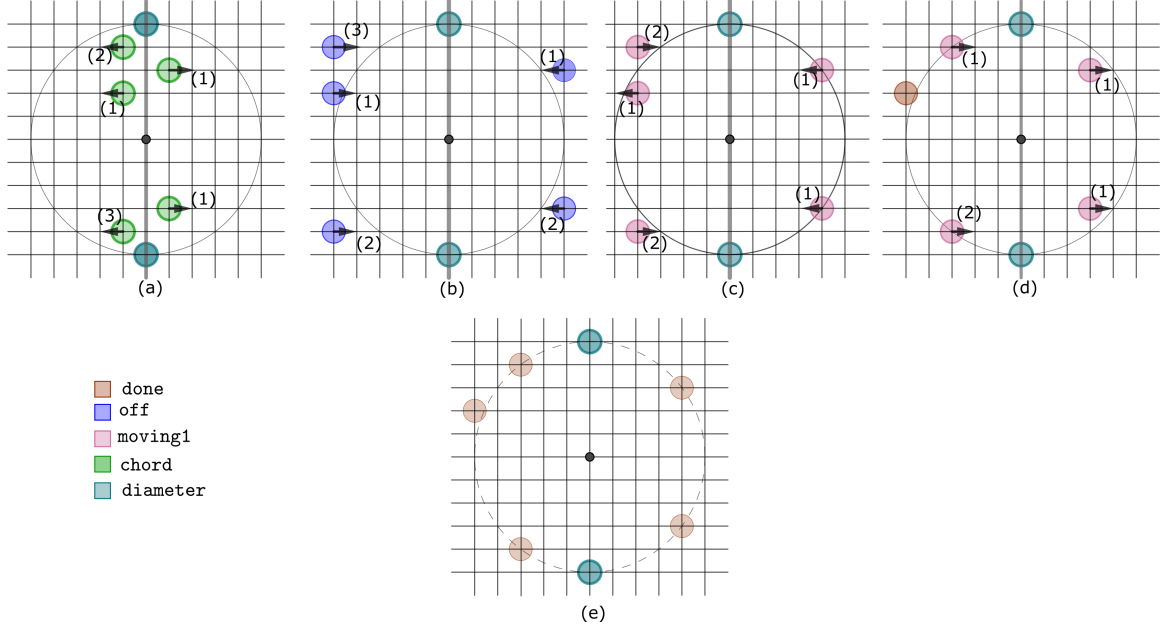


Figure 3. Outline of Phase 2: (a) From P1FC, all robots with color `chord` from one vertical line moves to another vertical line further from the diameter until it reaches $\frac{d}{2}$ distance from diameter where d is the diameter length. The number (i) associated with a robot denotes that it moved after $i - 1$ robots has already moved. (b) Before reaching the vertical line at distance $\frac{d}{2}$, all robots change their color to `off`. Now all robots of color `off` moves one hop to the vertical line towards diameter after changing the color to `moving1`. (c)-(d) When all robots of color `moving1` are on a line the robots that are strictly inside the circle moves away from diameter after changing the color to `done` and the other robots of color `moving1` moves to the next vertical line towards diameter after all inside robots moved. (e) Circle is formed.

Now when it is activated again, it will see r on its $\mathcal{R}_I(r_1)$ having color `off` and also $\mathcal{R}_I(r_1)$ is one hop away from $\mathcal{L}_V(r_1)$. In this view if $H_L^O(r_1)$ is empty then, r_1 moves left again shifting the \mathcal{L}_1 further left. The target of this move is to make the distance between \mathcal{L}_1 and the first vertical line that contains a robot of color `off` more than one. Let when r_1 moved left further r_2 was on $\mathcal{L}_V(r)$ with a pending movement due to asynchrony. Then, it will move left and see all robots on $\mathcal{L}_I(r_2)$ has color `moving1` and it is singleton on $\mathcal{L}_V(r_2)$. For this view r_2 moves to left again and moves to \mathcal{L}_1 along with r_1 . A robot with color `moving1` can change its color to `chord` for two possible views. For the first one it has to be on \mathcal{L}_1 and has to see another robot with color `chord` on \mathcal{L}_1 . For the other one it has to see its right immediate vertical line, which is at least two hop away, has at least a robot of color `off`. So when r_1 moved left twice from its initial position upon its next activation activation, if r_2 is still in $\mathcal{L}_v(r)$ and has not yet changed its color to `moving1`, r_1 changes its color to `chord` otherwise r_2 reaches $\mathcal{L}_V(r_1)$ and eventually both of them change their color to `chord` before any other robot does anything. Now a robot, say r' , which has changed its color to `moving1` from `off` on \mathcal{L}_k by seeing all robots of color `chord` on $\mathcal{L}_I(r')$ must be terminal on \mathcal{L}_k for some $k > 1$. Now, if upon activation r' sees all visible robots on $\mathcal{L}_I(r')$ has color `chord` and no robot of color `chord` on $\mathcal{L}_V(r')$ (This condition is to stop robots of color `moving1` to move further left from \mathcal{L}_1 , then it can have either both of $H_U^C(r') \cap \mathcal{L}_I(r')$ and $H_B^C(r') \cap \mathcal{L}_I(r)$ non empty or, empty. For this case, if there is another robot on $\mathcal{L}_V(r')$, then r' moves along $\mathcal{L}_V(r')$ opposite to that robot, otherwise it moves along its positive Y-axis. After finite moves one of $H_U^C(r') \cap \mathcal{L}_I(r')$ and $H_B^C(r') \cap \mathcal{L}_I(r)$ must

become empty for r' when it moves left. Note that all robots of color **off** decides that it is in Phase 1 when they see no robots of color **diameter**. A robot with color **moving1** decides it is in Phase 1 when it sees no robot of color **diameter** or when it sees a robot of color **diameter** on its own vertical line. If a robot of color **moving1** sees a robot with color **diameter** on its own vertical line, it changes its color to **diameter**.

Now a robot with color **chord** changes its color to **diameter** when it sees at least one robot of color **chord** on its corresponding vertical line and sees no other robot on its left and right open halves and one of upper or bottom closed halves. A robot, say r_c , with color **chord** executes CHORDMOVE when it is not terminal on $\mathcal{L}_V(r_c)$ and sees at least one robot of color **diameter** on $\mathcal{L}_V(r_c)$. On the otherhand if r_c is terminal on $\mathcal{L}_V(r_c)$ and sees a robot of color **diameter** on $\mathcal{L}(r_c)$ then it changes its color to **diameter** from **chord**. a robot with color **chord** distinguishes Phase 1 when it does not see any robot of color **diameter** and also sees at least one robot with **diameter** color on its own vertical line. We now describe the CHORDMOVE subroutine.

CHORDMOVE Subroutine: A robot r with color **chord** executes the subroutine CHORDMOVE when it sees at least one robot with color **diameter** and is not terminal on $\mathcal{L}_V(r)$. while executing CHORDMOVE, if a robot, say r , sees only one robot r_1 with color **diameter** then it checks if there is any other robot between the horizontal lines passing through r_1 and r i.e., $\mathcal{L}_H(r_1)$ and $\mathcal{L}_H(r)$ respectively. If there is no robots in the above mentioned region then r moves left. Now if there are robots between the mentioned region, then the following procedure takes place. If r_N be the nearest of r which is in the mentioned region and if r_N is on $H_L^O(r)$ then r moves right otherwise r moves left. Now, if a robot r sees both the robots, say r_1 and r_2 , with color **diameter** then, r finds its direction to move as stated above considering both the regions between $\mathcal{L}_H(r_1)$, $\mathcal{L}_H(r)$ and $\mathcal{L}_H(r_2)$, $\mathcal{L}_H(r)$. If for both the direction considering both the regions are same then r moves according to that direction otherwise it moves left. Note that after this procedure is complete, $\mathcal{L}_V(r_1)$ has only two robots r_1 and r_2 both having the color **diameter** and $\mathcal{L}_I(r_1)$ and $\mathcal{R}_I(r_1)$ contains all the robots with color **chord**. Also observe that, difference between number of robots on $\mathcal{L}_I(r_1)$ and $\mathcal{R}_I(r_1) \leq 2$. Note that, a robot, say r , with color **chord** does nothing after it has already moved once executing CHORDMOVE until the whole procedure is complete. This is because after r has moved once, it sees r_1 or r_2 not on $\mathcal{L}_V(r)$ and $\mathcal{L}_V(r_1)$ has other robots except r_1 and r_2 until all non terminal robots on $\mathcal{L}_V(r_1)$ executes this subroutine exactly once.

Robots begin to execute Phase 2 from a Phase 1 Final Configuration. So, we have to ensure that by executing Phase 1 robots will eventually form a Phase 1 Final Configuration. This is ensured in the subsequent correctness section. The pseudo code of Phase 1 is presented in Algorithm 1.

3.1.2. Correctness of Phase 1

Phase 1 is divided into two parts. In the first part, the robots first form a line where all robots have either the color **chord** or **moving1**. To be specific only terminal robots can have the color **moving1** on that line. Now at least one of the terminal robots eventually change its color to **diameter**. Then in the second part, the non terminal robots with color **chord** moves left or right once, reaching either \mathcal{L}_1 or \mathcal{L}_3 and thus forming a P1FC eventually.

So, first we have to show that all robots must move to a single line eventually where all robots have color either **chord** or **moving1**. We prove this by ensuring that all robots with color **off** eventually change its color to **moving1** (Lemma 3.12). Then all

Algorithm 1: Phase 1

```
1 Procedure PHASE1()
2    $r \leftarrow \text{myself}$ 
3   if  $r.\text{color} = \text{off}$  then
4     if  $r$  sees no robot with color diameter then
5       if There is no robot  $i \in H_L^O(r)$  and  $r$  is terminal then
6          $r.\text{color} \leftarrow \text{moving1}$ ;
7         move left;
8       else if  $\mathcal{L}_I(r)$  is at least two hop away and all visible robots and all visible
9         robots on  $\mathcal{L}_I(r)$  has color chord then
10        if  $r$  is terminal then
11           $r.\text{color} \leftarrow \text{moving1}$ ;
12    else if  $r.\text{color} = \text{moving1}$  then
13      if  $r$  sees no robot with color diameter on  $\mathcal{L}_V(r)$  then
14        if all visible robots on  $\mathcal{L}_I(r)$  has color chord and  $r$  sees no robot of color
15        chord on  $\mathcal{L}_V(r)$  then
16          if  $H_U^C(r) \cap \mathcal{L}_I(r)$  and  $H_B^C(r) \cap \mathcal{L}_I(r)$  both are non empty then
17            if There is a robot  $r'$  on  $\mathcal{L}_V(r)$  then
18              move opposite to  $r'$ 
19            else
20              move according to positive  $Y$ - axis;
21          else
22            move left;
23        else if  $r$  is singleton on  $\mathcal{L}_V(r)$  and all visible robots on  $\mathcal{L}_I(r)$  has color
24        moving1 or, (distance of  $\mathcal{R}_I(r)$  having a robot with color off = 1 and
25         $H_L^O(r)$  is empty. then
26          move left;
27        else if  $H_L^O(r)$  is empty then
28          if sees a robot with color chord on  $\mathcal{L}_V(r)$  then
29             $r.\text{color} \leftarrow \text{cord}$ ;
30          else if distance of  $\mathcal{R}_I(r)$  having a robot of color off  $\geq 2$  then
31             $r.\text{color} \leftarrow \text{chord}$ ;
32      else
33         $r.\text{color} \leftarrow \text{diameter}$ ;
34        terminate;
35    else if  $r.\text{color} = \text{chord}$  then
36      if  $r$  sees no robot with color diameter then
37        if There is a robot with color chord on  $\mathcal{L}_V(r)$ , there is no robot on
38         $H_L^O(r), H_R^O(r)$  and  $H(r)$  where  $H(r) \in \{H_B^C(r), H_U^C(r)\}$  then
39           $r.\text{color} \leftarrow \text{diameter}$ ;
40          terminate;
41      else if  $r$  sees a robot with color diameter on  $\mathcal{L}_V(r)$  then
42        if  $r$  is terminal then
43           $r.\text{color} \leftarrow \text{diameter}$ ;
44          terminate;
45        else
46          Execute CHORDMOVE();
```

robot with color `moving1` moves to \mathcal{L}_1 (Lemma 3.11). For this, we also ensured that \mathcal{L}_1 becomes fixed after a finite time (Lemma 3.7), otherwise a potential livelock situation may occur.

In the following, we have stated some observations proved some claims which will be needed to prove the above mentioned lemmas.

Observation 3.1. *A robot r can see all robots of $\mathcal{L}_I(r)$ (resp. $\mathcal{R}_I(r)$) if $\mathcal{L}_I(r)$ (resp. $\mathcal{R}_I(r)$) is at least two hop away from r .*

Observation 3.2. *If r be a robot of color `off` executing Phase 1 such that $H_R^O(r)$ is non empty, then all robots on $H_R^O(r)$ must have color `off`*

Observation 3.3. *After a move by any robot from the initial configuration and before any robot changes its color to `chord`, \mathcal{L}_1 can have at most two robots of color `moving1` and all other robots has color `off`.*

Claim 3.4. *Let r be a robot with color `moving1` such that $\mathcal{R}_I(r)$ is one hop away from $\mathcal{L}_V(r)$ and there is at least one robot with color `off` on $\mathcal{R}_I(r)$. If r is activated in this configuration, then r always sees a robot having color `off` on $\mathcal{R}_I(r)$ in Phase 1.*

Proof. Let r be a robot with color `moving1`. Let $\mathcal{R}_I(r)$ be one hop away from $\mathcal{L}_V(r)$ which has a robot, say r' of color `off` on it. If possible r does not see any robot with color `off` on $\mathcal{R}_I(r)$. That is r does not see r' upon activation, say at a time $t > 0$. Let $\mathcal{L}_H(r)$ and $\mathcal{L}_H(r')$ be two horizontal lines passing through r and r' respectively. Then the above assumption is true only when $\mathcal{L}_V(r)$ and $\mathcal{L}_V(r')$ each contains at least one robot between $\mathcal{L}_H(r)$ and $\mathcal{L}_H(r')$. Let r_1 and r_2 be two such robots on $\mathcal{L}_V(r)$ and $\mathcal{L}_V(r')$ respectively. without loss of generality, let r can see r_2 and r' can see r_1 . Note that r_2 must be of color `moving1` at time t . Also r_1 either have color `off` or color `moving1` at time t (r_1 can not be of color `chord` at time t as r' is on $\mathcal{R}_I(r_1)$ having color `off` at time t).

Now, if r_1 has color `off` then in the interval $(0, t)$, $\mathcal{L}_V(r_1)$ has not changed. Now in this interval, all robots on $\mathcal{L}_V(r_1)$ must have color `moving1` or `off`. So in the interval, $(0, t)$, r_2 can never see all robots with color `chord` on $\mathcal{L}_I(r_2)$ and thus can not change its initial color `off` to `moving1` in the mentioned interval. So, r_2 can not be of color `moving1` at time t which is a contradiction to our assumption.

So, let r_1 has color `moving1` at time t . Now, since at time t , r_2 has color `moving1`, there is a time $t_1 < t$ when r_2 has color `off`, is terminal on $\mathcal{L}_V(r_2)$ and either sees $H_L^O(r_2)$ is empty or sees all robots on $\mathcal{L}_I(r_2)$ having color `chord`. This implies r and r_1 must have moved to $\mathcal{L}_I(r_2)$ after time t_1 . Thus, at time t_1 all of r, r_1 and r_2 were along with r' on $\mathcal{L}_V(r_2) = \mathcal{L}_V(r')$. Now since, r and r_1 moves to $\mathcal{L}_I(r_2)$, they must have changed their color to `moving1` from initial color `off`. There are three cases. Firstly, let both r and r_1 get activated and see that they are terminal on $\mathcal{L}_V(r')$ and change their color to `moving1` on or after time t_1 . This is not possible as in this case r, r_1 and r_2 all have to be terminal on $\mathcal{L}_V(r')$ at time t_1 which is not possible. Secondly, let both r and r_1 be activated and see themselves terminal on $\mathcal{L}_V(r')$ and change their color to `moving1` before t_1 . This case is also impossible as at time t_1 since r and r_1 are still on $\mathcal{L}_V(r')$, r_2 can not see itself as a terminal robot which is a contradiction. So first, let us assume r changed its color before time t_1 and r_1 changed its color after time t_1 and before time t . We claim that, for r_1 to change its color after time t_1 , r must have to move left from $\mathcal{L}_V(r')$. This is because, at time t_1 , r and r_2 both are terminal on $\mathcal{L}_V(r')$ so until r moves r_1 can not become terminal and thus can not change its

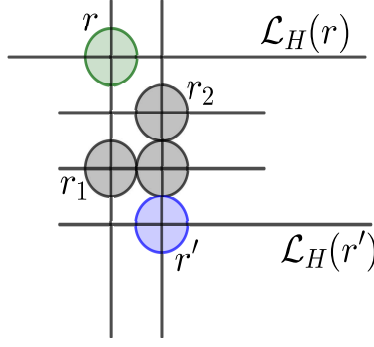


Figure 4. r has color `moving1`, r' has color `off` at time t and r can not see r' .

color. Now after time t_1 if r_1 is activated before time t , even if it is terminal now it will not change its color to `moving1` as $\mathcal{L}_I(r_1)$ is exactly one hop away (less than two hop away). Thus, even if r_1 is activated it does not change its color from `off` until time t . So at time t , r can see r_2 with color `off` contrary to our assumption. Now, if r_1 had changed its color to `moving1` before time t_1 and r after time t_1 , then by similar argument it can be said that at time t , r stays at $\mathcal{L}_V(r')$. So, in this case, if $\mathcal{R}_I(r)$ is non-empty, all robots on $\mathcal{R}_I(r)$ will have color `off` (Observation 3.2). Thus, r must see at least one robot having color `off` on $\mathcal{R}_I(r)$ even if it is one hop away contrary to our assumption. Thus, for a robot r if $\mathcal{R}_I(r)$ has a robot with color `off` and $\mathcal{R}_I(r)$ is one hop away then r always sees at least one robot of color `off` on $\mathcal{R}_I(r)$. \square

Claim 3.5. *Before \mathcal{L}_1 is fixed, no robot in the configuration can have color `chord`.*

Proof. Let there is a time t when \mathcal{L}_1 is not fixed but the configuration at time t has a robot, say r with color `chord`. without loss of generality let r be the first robot that changes its color from `moving1` to `chord`. So, there must be a time $t_0 < t$ when r is activated on \mathcal{L}_1 with color `moving1` and changes its color to `chord`. Note that at time t_0 , \mathcal{L}_1 is not fixed. Also, since there are more than 2 robots in the system, there must be at least one robot with color `off` which is not on \mathcal{L}_1 for the whole duration $[t_b, t_0)$. Thus there must be a robot on \mathcal{L}_1 with color `moving1` which moves left after t_0 . Let r' be that robot. Note that r' can not be r as a robot with color `chord` does not move in Phase 1. Now, if r' is activated after time t_0 for executing the LCM cycle where it moves left from \mathcal{L}_1 then, upon activation, it must have seen a robot r_1 with color `off` on $\mathcal{R}_I(r') = \mathcal{L}_2$ which is one hop away from \mathcal{L}_1 . But this is not possible because if r_1 is on \mathcal{L}_2 after time t_0 it must have been there at time t_0 also. So at time t_0 , r does not change its color to `chord` upon activation contrary to our assumption. So, let r' be activated at a time $t' < t_0$ for executing the LCM cycle where it moves left from \mathcal{L}_1 after time t_0 . This is only possible if at time t' , $\mathcal{R}_I(r') = \mathcal{L}_2$ had a robot, say r_1 , with color `off`. Now since r' moves left after time t_0 , at time t_0 upon activation r must have seen a robot with color `off` on $\mathcal{R}_I(r) = \mathcal{L}_2$ which is one hop away from \mathcal{L}_2 . Thus r doesn't change its color to `chord` upon activation at time t_0 . This is also a contradiction. Thus, before \mathcal{L}_1 is fixed, no robot changes their color to `chord`. \square

Claim 3.6. *In Phase 1, between the time of first move by any robot from the initial configuration and the time when all robots move to a single line for the first time, a*

robot on \mathcal{L}_1 have color either *moving1* or *chord*.

Proof. Let $t_b > 0$ be the time when the first move by a robot happened from the initial configuration. Also, let $t_f > t_b$ be the time, when all robots move to a single line after time t_b for the first time (t_f can be infinite if all robots never move to a single line). Now, \mathcal{L}_1 can not have a robot with color *diameter* in the time interval (t_b, t_f) as, in this interval no robot sees both its left and right open halves empty. Also, since at time t_b at least one leftmost terminal robot moves left after changing its color to *moving1*, \mathcal{L}_1 also shifts left at time t_b . Note that, after this move, \mathcal{L}_1 does not have any robot with color *off*. So in the interval (t_b, t_f) , \mathcal{L}_1 can not have any robot with color *off* as robots with color *off* never moves left to reach \mathcal{L}_1 (algorithm 1) and no robot of different color change their color to *off* in Phase 1. So, within the time interval (t_b, t_f) , \mathcal{L}_1 can have robots of color either *moving1* or of color *chord*. \square

Lemma 3.7. \mathcal{L}_1 can not shift left infinitely often without all robots being on \mathcal{L}_1 .

Proof. Let t_b be the time when the first robot moves from the initial configuration. After t_b , suppose \mathcal{L}_1 shifts left infinitely often, while there remains at least one robot not positioned on \mathcal{L}_1 after t_b . This implies there is a robot, say r , which moves left from \mathcal{L}_1 infinitely often. Notably, robot r must have the color *moving1*, and it retains this color without change. Now r can move left from \mathcal{L}_1 only if it sees a robot, say r' , of color *off* on $\mathcal{R}_I(r)$ which is one hop away from \mathcal{L}_1 (Figure 5). Let $t_0 > t_b$ be a time when r is activated on \mathcal{L}_1 and observes r' on $\mathcal{R}_I(r)$, situated at a distance of one unit from \mathcal{L}_1 . In this case, r moves left and shifts \mathcal{L}_1 to left along with it. Note that after this move is completed, no robot on \mathcal{L}_1 will ever see another robot of color *off* on \mathcal{L}_2 (as robots with color *off* never move left in Phase 1). Consequently, for all subsequent times $t_1 > t_0$, if r is reactivated at t_1 on \mathcal{L}_1 with the color *moving1*, it will not perceive any robot with the color *off* on \mathcal{L}_2 . As a result, it will not move left, contradicting our initial assumption. Hence, the leftward shift of \mathcal{L}_1 cannot continue infinitely without all robots being positioned on the same line. \square

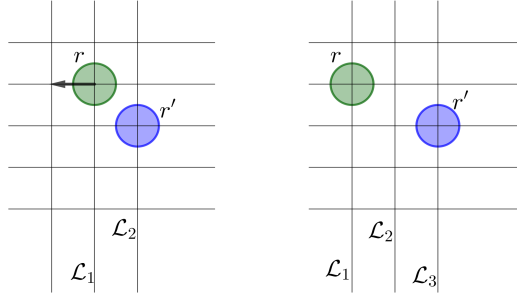


Figure 5. in the configuration on left, r can see r' with color *off* on \mathcal{L}_2 so it moves left. in the configuration on right, r can not see any robot on \mathcal{L}_2 with color *off* so it does not move

Claim 3.8. Excluding \mathcal{L}_1 there can not be more than two robots with color *moving1* in Phase 1.

Proof. Suppose there are three robots, denoted as r_1, r_2 , and r_3 , all with color *moving1*, positioned on $\mathcal{L}_{k_1}, \mathcal{L}_{k_2}$, and \mathcal{L}_{k_3} at time t , where $1 < k_1 \leq k_2 \leq k_3$. Initially, r_1, r_2 , and r_3 must all be located on the same vertical line. Otherwise, the rightmost robot among them in the initial configuration cannot transition its color from *off* to

moving1, unless the leftmost robot(s) among the others reach \mathcal{L}_1 and alter their color to chord. However, this contradicts the observation that all three robots have the color moving1 at time t . Therefore, let's assume that initially all of them are positioned on \mathcal{L}_k with the color off. It is important to note that none of them moves left from \mathcal{L}_k unless all of them are activated for their corresponding LCM cycle, in which they change their color to moving1. This implies that there exists a time $t' < t$ when all of r_1, r_2 , and r_3 are located on \mathcal{L}_k , and each of them either possesses the color moving1 or becomes activated in their corresponding LCM cycle, during which they change their color to moving1. This implies r_1, r_2 and r_3 all are terminal on \mathcal{L}_k at time t' but this is not possible. Hence, excluding \mathcal{L}_1 , there can not be more than two robots with color moving1 in Phase 1. \square

Claim 3.9. *A robot with color moving1 on \mathcal{L}_1 must be terminal on \mathcal{L}_1 in Phase 1.*

Proof. From Observation 3.3 and Claim 3.5 it is evident that before \mathcal{L}_1 is fixed, it can have at most two robots and those are of color moving1. So before \mathcal{L}_1 is fixed, any robot with color moving1 on \mathcal{L}_1 must be terminal on \mathcal{L}_1 .

Let \mathcal{L}_1 becomes fixed at a time t_0 . Now let us assume r be a robot with color moving1 on \mathcal{L}_1 that is not terminal on \mathcal{L}_1 at a time $t > t_0$. This implies there must be another robot, say r' , which without loss of generality is directly below r on \mathcal{L}_1 at the time t and r has color moving1. Now, r and r' must have moved to \mathcal{L}_1 from \mathcal{L}_2 . Note that, in the initial configuration, r and r' can not be on two different vertical lines otherwise the rightmost robot among them can not change its color from off to moving1 until the other one reaches \mathcal{L}_1 and change color to chord. Also, even when they are on the same vertical line and one of them, say r , already moves left before the other one i.e., r' wakes to change its color from off to moving1, it can't do so unless r reaches \mathcal{L}_1 and change its color to chord. So, without loss of generality let us assume before one of r and r' moves from \mathcal{L}_k , the other robot must have been activated and seen \mathcal{L}_1 where all robots have color chord. This ensures that r and r' will change their color to moving1 from off. Let in the initial configuration, r and r' were on the same line which is the line \mathcal{L}_k at the time t_0 and $k \geq 3$ (\mathcal{L}_2 at time t_0 can not have any robot in the initial configuration). Also observe that, for r and r' to reach \mathcal{L}_1 they must move there from \mathcal{L}_2 (The lines \mathcal{L}_i are denoted for the time when \mathcal{L}_1 becomes fixed). Since in the initial configuration r and r' were not on \mathcal{L}_2 of the current configuration, they must have color moving1 while on \mathcal{L}_2 . Suppose r is located on \mathcal{L}_2 while r' is on \mathcal{L}_j with $j \geq 2$ and has color either moving1 or is in the transitional LCM cycle where it would change its color to moving1 on \mathcal{L}_j . Now for the former case, if r is activated and it moves to \mathcal{L}_1 while r' is idle then upon activation r' can not move to \mathcal{L}_{j-1} until r changes its color to chord. For the latter case also, before the next activation of r' , if r moves to \mathcal{L}_1 , r' can not move to \mathcal{L}_{j-1} unless r changes its color to chord. In either of these cases, when r' reaches \mathcal{L}_1 , r must be of color chord, which contradicts our assumption.

Therefore, for both r and r' to reach \mathcal{L}_1 with the color moving1, there must exist a time $t_1 > t_0$ and $t_1 < t$ when both of them are on \mathcal{L}_2 with the color moving1. Furthermore, neither of them moves to \mathcal{L}_1 while the other one is inactive. Note that r' must have changed its color to moving1 from off after seeing all robots with color chord as \mathcal{L}_1 of the initial configuration can not have more than two terminal robots of color off. This implies \mathcal{L}_1 must have at least one robot of color chord at the time when r' moves on it after time t_0 . Now since both r and r' are activated on \mathcal{L}_2 before any one of them moves left, they must see each other on \mathcal{L}_2 and move opposite of

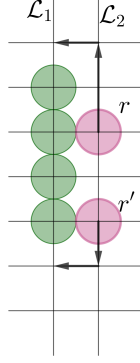


Figure 6. r and r' from \mathcal{L}_2 moves to \mathcal{L}_1 in such a way that at \mathcal{L}_1 all other robots are between r and r' . Thus r' can not be directly below or above r .

each other on \mathcal{L}_2 until one robot has moved above $\mathcal{L}_H(r_1)$ and the other moves below $\mathcal{L}_H(r_2)$, where r_1 is the uppermost robot on \mathcal{L}_1 and r_2 is the lowest robot on \mathcal{L}_1 before r and r' moves on to \mathcal{L}_1 . Thus now when they move left on \mathcal{L}_1 they must have at least one robot with color **chord** between them. So at time t , r' can not be directly below r . Thus A robot with color **moving1** on \mathcal{L}_1 must be terminal on \mathcal{L}_1 . \square

Claim 3.10. *In Phase 1, if the fixed \mathcal{L}_1 has a robot r with color **moving1** which sees at least one robot that is not on \mathcal{L}_1 upon activation then r changes its color to **chord** eventually.*

Proof. If \mathcal{L}_1 has more than two robots then r must see a robot with color **chord** on $\mathcal{L}_V(r) = \mathcal{L}_1$ upon activation (Claim 3.9 and Claim 3.6) (Figure. 7(a)). So, it must change its color to **chord**. Therefore, let \mathcal{L}_1 have at most two robots when r is activated on the fixed \mathcal{L}_1 at a time, say t_0 . If another robot has color **chord** on \mathcal{L}_1 when r is activated at t_0 , by the same reason r changes its color to **chord**. So let us assume that if there is another robot, say r' , on \mathcal{L}_1 at time t_0 then it has color **moving1** (Figure 7(b)). In this configuration, there is no robot of color **chord**. So all other robots except r and r' are of color **off** in this configuration, thus they are at their initial positions. For this case, upon activation, r must see at least one robot with color **off** on $\mathcal{R}_I(r)$ which is at least two hop away from \mathcal{L}_1 and thus r changes its color to **chord**. Now at time t_0 upon activation if r is singleton on \mathcal{L}_1 then there can be at most another robot with color **moving1** on \mathcal{L}_j ($j > 1$) (Figure. 7(c)). Now if there is no other robot with color **moving1** at time t_0 then r sees $\mathcal{R}_I(r)$ has a robot of color **off** and it is at least 2 hop away from \mathcal{L}_1 . Thus, in this case, r changes its color to **chord**. Otherwise, the other robot, say r' with color **moving1** is either singleton on some \mathcal{L}_j or not ($j > 1$). Also in between \mathcal{L}_1 and \mathcal{L}_j , there is no other robot at time t_0 . This is because, \mathcal{L}_j at time t_0 must be either the vertical line \mathcal{L}_1 of the initial configuration or is strictly left of \mathcal{L}_1 of the initial configuration and since there is no robot with color **chord** on \mathcal{L}_1 , except r and r' all robots has color **off** at time t_0 (this ensures all robots except r and r' never moved from their initial position until time t_0). If, at time t_0 , r' is singleton on some \mathcal{L}_j ($j > 1$) then upon activation r' moves left to \mathcal{L}_{j-1} . Note that if r' is not on \mathcal{L}_1 , r on \mathcal{L}_1 does nothing even when activated. So r' eventually reaches \mathcal{L}_1 . Now when r' reaches \mathcal{L}_1 , by the above argument when r activates next, it changes the color to **chord** as it sees $\mathcal{R}_I(r)$ has a robot with color **off** which is at least two hop away from \mathcal{L}_1 . Now let at time t_0 , r' was not singleton on \mathcal{L}_j , then there must be another

robot, say r_1 , with color `off` on \mathcal{L}_j which is seen by r at time t_0 . For this case also, r changes its color to `chord`. \square

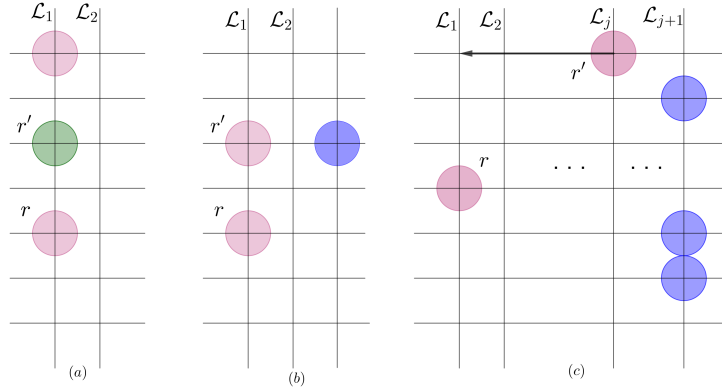


Figure 7. (a) \mathcal{L}_1 has more than two robots including r . Then r sees r' of color `chord` on \mathcal{L}_1 . (b) There are exactly two robots, r and r' on \mathcal{L}_1 and both of them has color `moving1`. (c) r is singleton on \mathcal{L}_1 with color `moving1` and r' is another robot with color `moving1` on \mathcal{L}_j . r' moves to \mathcal{L}_1 and transforms into case (b). Here the blue color denotes color `off`

Lemma 3.11. *In Phase 1, a robot with color `moving1` on \mathcal{L}_j eventually moves to \mathcal{L}_{j-1} where $j \geq 2$ if \mathcal{L}_1 is fixed.*

Proof. We will prove this using mathematical induction on j .

Base case: In the base case we first establish that a robot r with color `moving1` on \mathcal{L}_2 moves to \mathcal{L}_1 eventually. For that, let r be a robot on \mathcal{L}_2 with color `moving1` at a time t . If possible, let r never reaches \mathcal{L}_1 . Note that there can be at most one robot, say r' , other than r which is not on \mathcal{L}_1 and has color `moving1` at time t (Claim 3.8). Now let at time t , r' is on \mathcal{L}_k for some $k \geq 2$.

Case 1: Suppose r' is on \mathcal{L}_2 along with r (i.e., $k = 2$) at time t . Note that \mathcal{L}_2 can not have any other robot of color `off` as it is strictly to the left of the \mathcal{L}_1 in the initial configuration. So all robots on right of \mathcal{L}_2 must be of color `off` and does not do anything even if they are activated as whenever they are activated they never see any robot with color `chord` on their left immediate vertical lines. Now at time t , \mathcal{L}_1 either has all robot with color `chord` or, has all robots of color `chord` except at most two robots on the terminal. For the later case, by Claim 3.10 all robot on \mathcal{L}_1 will have color `chord` eventually at a time, say t' , where $t' > t$ (Figure 8(a)). Until then r and r' does nothing even if they are activated. after t' when r' activates next it sees \mathcal{L}_1 has all robot with color `chord` and thus move left to \mathcal{L}_1 . with similar argument after moving to \mathcal{L}_1 eventually r' end up with color `chord`. next when r is activated on \mathcal{L}_2 it must see that all robot on \mathcal{L}_1 has color `chord` and thus moves left contrary to the assumption.

Case 2: Next suppose r is singleton on \mathcal{L}_2 and r' is singleton on some \mathcal{L}_k where $k > 2$. Note that between \mathcal{L}_2 and \mathcal{L}_k there is no other robots (Claim 3.8 and Observation 3.2). Note that all other robot on the right of \mathcal{L}_k at time t has color `off` and upon activation they do nothing from time t onwards as they never see all robots with color `chord` on their left immediate vertical line as r never moves to \mathcal{L}_1 . In this case whenever r' is activated it sees r singleton on \mathcal{L}_2 and moves left to \mathcal{L}_{k-1} (Figure 8(b)). This way eventually r' reaches \mathcal{L}_2 along with r . This is same configuration as described in case 1. So, again it would reach a contradiction.

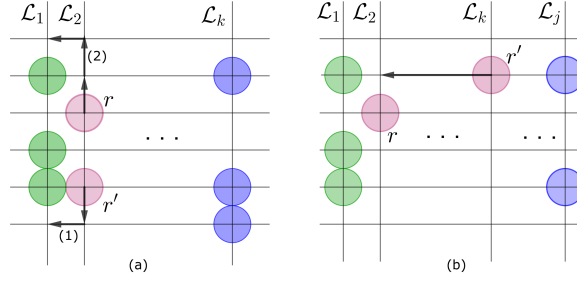


Figure 8. (a) r and r' are on \mathcal{L}_2 with color `moving1`. If r do not move left, r' moves left upon seeing all robots on \mathcal{L}_1 of color `chord` and in the next activation changes color to `chord`. Next r moves to \mathcal{L}_1 . (b) r with color `moving1` is singleton on \mathcal{L}_2 and r' with color `moving1` is also singleton on \mathcal{L}_k ($k > 2$). If r do not move to \mathcal{L}_1 , r' moves and reaches \mathcal{L}_2 and converts to the case in (a).

Case 3: Let, r is singleton on \mathcal{L}_2 and r' is not singleton on \mathcal{L}_k at time t where $k > 2$. Then if r' has a pending move, it moves to \mathcal{L}_{k-1} , otherwise if r' activates after time t , it does nothing. For the former case, after r' moves to \mathcal{L}_{k-1} , it either falls into the case 1 or, the case 2. Now, for the later case, if r' does nothing on \mathcal{L}_k then r always remains singleton on \mathcal{L}_2 and no new robot moves onto \mathcal{L}_1 from time t on wards. So, at time t if not all robots on \mathcal{L}_1 is of color `chord`, eventually they all becomes of color `chord` (Claim 3.10). So, upon next activation r must move to \mathcal{L}_1 . This is again a contradiction to our assumption.

Case 4: In this case, let us assume r is only robot in the configuration of color `moving1` on \mathcal{L}_2 at time t . If from t onwards no new robot changes its color to `moving1` then eventually there will be a time, say $t' > t$ when all robots on \mathcal{L}_1 have color `chord`. Also no new robot moves to \mathcal{L}_1 after t' . Thus, when r activates next it must move to \mathcal{L}_1 . So, let us now assume there is a time when a new robot r' changes its color to `moving1`. Then as described in the previous cases, r will move to \mathcal{L}_1 contradicting our assumption. So, There will be a time when r moves to \mathcal{L}_1 from \mathcal{L}_2 .

Hypothesis: For some $j > 2$ and for any $i \leq j$, a robot having a color `moving1` on \mathcal{L}_i moves to \mathcal{L}_{i-1} where $i \geq 2$.

Inductive step: Let r be a robot on \mathcal{L}_{j+1} with color `moving1` at a time t , where $j \geq 2$. Now by Claim 3.8, there can be at most another robot, say r' with color `moving1` on \mathcal{L}_k at time t ($k > 1$). If possible let r never move to \mathcal{L}_j . Now there are two cases.

Case 1: For the first case let us assume $k \leq j + 1$ (Figure 9(a)). First assume that r is singleton on \mathcal{L}_{j+1} at time t . Then $k < j + 1$. By Observation 3.2 between \mathcal{L}_1 and \mathcal{L}_{j+1} there are no other robot except r' on \mathcal{L}_k at time t . Note that by induction hypothesis r' eventually moves to \mathcal{L}_1 and changes its color to `chord`, say at a time $t' > t$. Before that no robot on $H_R^Q(r)$ does anything even if they are activated as they can not change their color from `off` to `moving1` due to the fact that they can never see \mathcal{L}_1 as their left immediate vertical line due to r being on \mathcal{L}_{j+1} . So, after t' , whenever r activates it sees $\mathcal{L}_I(r) = \mathcal{L}_1$ where all robots have color `chord`. Thus eventually it moves left to \mathcal{L}_j . Also, before t' if r is activated and sees r' on some $\mathcal{L}_{k'}$ where $1 < k' \leq k$ then it moves left to \mathcal{L}_j contrary to the assumption. So, let r is not singleton on \mathcal{L}_{j+1} at time t . Let there are p robots on \mathcal{L}_{j+1} at time t (Figure 9(a)). Now if r' is on some \mathcal{L}_k where $1 < k < j + 1$ then, it must be singleton on \mathcal{L}_k . Now, until r' reaches \mathcal{L}_1 and changes its color to `chord`, no other robot on the right of r' does anything. By induction hypothesis, it can be ensured that r' will reach \mathcal{L}_1 and eventually all robot on it would change their color to `chord` at a time say, t' . Now since $p > 1$, there must exist another robot, say r_1 , having color `off` on \mathcal{L}_{j+1} which

is terminal on it at time t' . Also at time t' there is no other robot in between \mathcal{L}_1 and \mathcal{L}_{j+1} and no other robot except r_1 moves in between \mathcal{L}_1 and \mathcal{L}_{j+1} from time t' onwards unless r_1 reaches \mathcal{L}_1 and changes its color to **chord**. This is because only r_1 can change its color to **moving1** from **off**, after time t' and until it reaches \mathcal{L}_1 and changes color to **chord** (by Claim 3.8). This implies, after t' whenever r_1 is activated it will change its color to **moving1** from **off** and will eventually move to \mathcal{L}_j seeing all robots on \mathcal{L}_1 having color **chord**. Now by the hypothesis r_1 will reach \mathcal{L}_1 and change its color to **chord** eventually at a time, say t_1 . Now at time t_1 , \mathcal{L}_{j+1} has $p-1$ robots. if $p-1 = 1$ then r becomes singleton on \mathcal{L}_{j+1} and as described above it will eventually move to \mathcal{L}_j . otherwise there always will be a robot on \mathcal{L}_{j+1} which is terminal and has color **off**. For this case the terminal robot will eventually reach \mathcal{L}_1 as described above and will change its color to **chord**. This implies eventually number of robots on \mathcal{L}_{j+1} will decrease until only r remains. For this case, r moves to \mathcal{L}_j eventually, contrary to the assumption.

Case 2: Let $k > j + 1$ (Figure 9(b)). This implies r is singleton on \mathcal{L}_{j+1} also, there are no robots between \mathcal{L}_1 and \mathcal{L}_k except r . First assume r' is singleton on \mathcal{L}_k (Figure 9(b)). Then upon seeing only r on $\mathcal{L}_I(r')$ it will first move left until it reaches \mathcal{L}_{j+1} . Now as argument-ed for the previous case 1, r' will reach \mathcal{L}_1 eventually and change its color to **chord**. Note that since r is singleton on \mathcal{L}_{j+1} , no robot from $H_R^O(r)$ does anything upon activation. Also, there are no robots in between \mathcal{L}_1 and \mathcal{L}_{j+1} and all robots on \mathcal{L}_1 has color **chord**. Note that now when r will be activated it will always see $\mathcal{L}_I(r) = \mathcal{L}_1$ where all robots have color **chord** and thus eventually it will move to \mathcal{L}_j . This is again a contradiction. Thus, let us assume at time t , r' is not singleton on \mathcal{L}_k . For this case r' will not do anything even if it is activated. Thus, eventually r will see all robots with color **chord** on $\mathcal{L}_I(r)$ (i.e., \mathcal{L}_1). Thus r will move to \mathcal{L}_j .

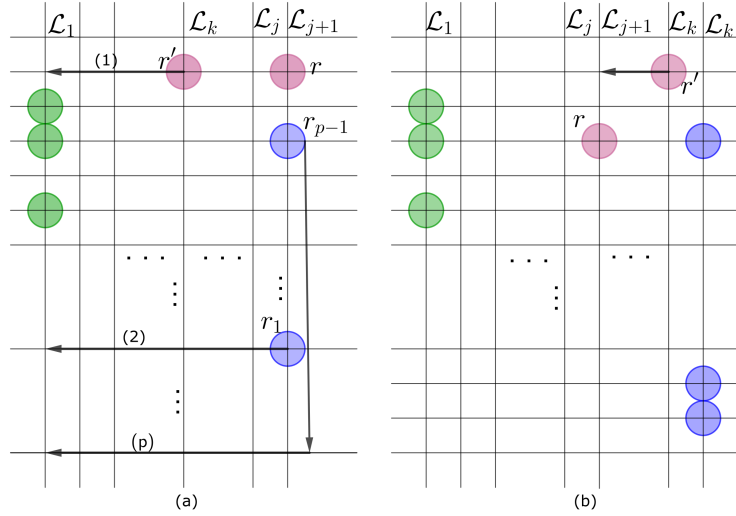


Figure 9. (a) r' moves to \mathcal{L}_1 first and changes color to **chord**. Next r_1 changes color to **moving1** and do same as r' . Eventually r_{p-1} also moves to \mathcal{L}_1 and changes color to **chord**. Next whenever r activates it moves and eventually reaches \mathcal{L}_j . (b) Here if r does not move, singleton r' moves left. After reaching \mathcal{L}_{j+1} it transforms into case I.

So considering another robot, r' having color **moving1** on \mathcal{L}_k ($k > 1$) at time t we always reach a contradiction. So, let at time t , r is the only robot with color **moving1** on \mathcal{L}_{j+1} ($j \geq 2$). For some pending move, another robot may change its color to **moving1** later, say at $t' > t$. Now before t' if \mathcal{L}_1 has all robot with color **chord** and r is

activated, it will move to \mathcal{L}_j . Otherwise, we reach the same configuration described in case 1 and case 2. So, for this case also, we have a contradiction. Thus, r will eventually move to \mathcal{L}_j . \square

Lemma 3.12. *After \mathcal{L}_1 is fixed, let \mathcal{L}_k be the first vertical line with a robot r of color *off* at a time t . Then r eventually changes its color to *moving1* in Phase 1.*

Proof. Let after \mathcal{L}_1 is fixed, t be a time when \mathcal{L}_k is the first vertical line that contains a robot r having color *off*. This implies, all robots between \mathcal{L}_1 and \mathcal{L}_k (if exists) has color *moving1* at time t . Now let from t onwards r never changes its color to *moving1*. Let there be $p \geq 1$ robots at time t on \mathcal{L}_k . Now we have two cases.

Case 1: Let $p = 1$. That is r is singleton on \mathcal{L}_k at time t (Figure 10(a)). Then robots on $H_R^O(r)$ must be of the color *off* at time t (Observation 3.2) and does nothing upon activation from time t onwards as r does not change its color. Now, by lemma 3.11, all robots in between \mathcal{L}_1 and \mathcal{L}_k eventually moves to \mathcal{L}_1 and change their color to *chord* at a time, say t_0 . And within this time no new robot changes its color to *moving1* from *off*. Thus from t_0 onwards, if r does not change its color, the configuration remains unchanged. Now, when r is activated after time t_0 , it sees all robots on $\mathcal{L}_I(r) = \mathcal{L}_1$ has color *chord* and thus changes its color to *moving1*, contrary to our assumption.

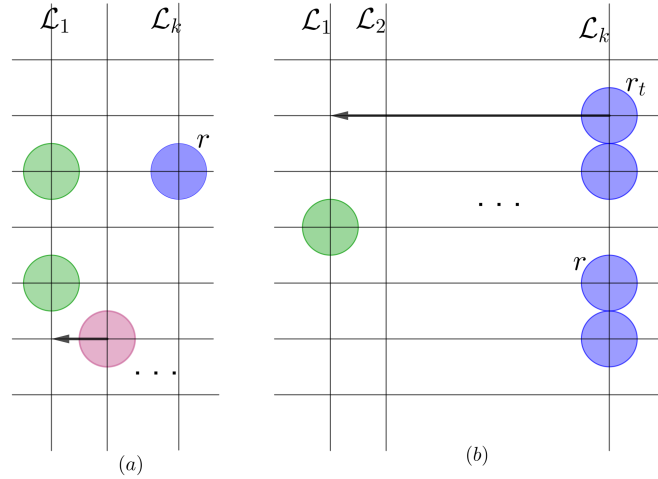


Figure 10. (a) r with color *off* is singleton on \mathcal{L}_k the robots between \mathcal{L}_1 and \mathcal{L}_k moves to \mathcal{L}_1 and changes color to *chord*. Next, whenever r activates it changes color to *moving1*. (b) r is not singleton on \mathcal{L}_k . All robots between \mathcal{L}_1 and \mathcal{L}_k has already moved to \mathcal{L}_1 and changed color to *chord*. Next, whenever r_t activates it changes color *moving1* and moves left decreasing the number of robots on \mathcal{L}_k and eventually transforming into the case shown in (a).

Case 2: Let $p > 1$ (Figure 10(b)). For this case, we will show that number of robots on \mathcal{L}_k eventually decreases until it becomes one. Which we already discussed in case 1. First observe that no robot on $H_R^O(r)$ at time t , moves to \mathcal{L}_k as they will remain of color *off* (observation 3.2) from time t onwards if r does not change its color to *moving1*. So the number of robots on \mathcal{L}_k never increase from time t onwards. Now let the number of robots on \mathcal{L}_k never decrease i.e., it remains the same. This implies, no robot from \mathcal{L}_k changes its color to *moving1* from time t onwards. Now by Lemma 3.11 all robots in between \mathcal{L}_1 and \mathcal{L}_k have color *moving1* at time t and they will eventually move to \mathcal{L}_1 and change their color to *chord* at a time, say $t_1 \geq t$. Now let r_t be the terminal robot on \mathcal{L}_k which is activated first after time t_1 . Now upon activation,

it must see all robots on $\mathcal{L}_I(r_t) = \mathcal{L}_1$ has color `chord` and thus changes its color to `moving1` contrary to our assumption. Thus the number of robots on \mathcal{L}_k will decrease and eventually have only r . So according to case 1, we will again reach a contradiction. As for both cases contradictions are achieved, our assumption that r never changes its color to `moving1` from time t onwards is false. So there is a time when r changes its color to `moving1`. \square

So using Lemma 3.7 we guarantee that within a finite time after the first move by any robot from the initial configuration \mathcal{L}_1 will be fixed. Let \mathcal{L}_k be the first vertical line on the right of \mathcal{L}_1 where a robot of color `off` exists ($k > 2$). So all robots in between \mathcal{L}_1 and \mathcal{L}_k has color `moving1`. Thus by Lemma 3.11 eventually all these robots will move to \mathcal{L}_1 . Now observe that all robots on \mathcal{L}_k will have color either `moving1` or `off`. Now using Lemma 3.11 it can be said that all robots of color `moving1` on \mathcal{L}_k will move to \mathcal{L}_1 eventually. Also by Lemma 3.12 all robots of color `off` on \mathcal{L}_k change their color to `moving1` and move to \mathcal{L}_1 eventually. Thus Within finite time all robots of \mathcal{L}_k move to \mathcal{L}_1 . Now if there are no other robots on the right of \mathcal{L}_k then we are done. Otherwise the first vertical line on the right of \mathcal{L}_1 containing a robot of color `off`, shifts right. Eventually, there will be one such line that does not have any other robot on its right and all other robots will be on \mathcal{L}_1 . Now as described above all other robots of that line will also eventually move to \mathcal{L}_1 . In this moment all non terminal robots on \mathcal{L}_1 will have color `chord` and the terminal robots either will have color `chord` or `moving1` (Claim 3.9). From this above discussion, we can have the following theorem.

Theorem 3.13. *There exists a time t when all robots move to a single line with non terminal robots having color `chord` and terminal robots having color either `chord` or, `moving1` by executing the algorithm 1: Phase 1 from any initial configuration assuming one axis agreement, under asynchronous scheduler.*

We now proof the following theorem which states that A configuration where all robots are on a single line with all non terminal robots have color `chord` and terminal robots have color either `chord` or, `moving1` will eventually change into a Phase 1 Final Configuration. This theorem ensures the termination of Phase 1.

Theorem 3.14. *Let in a configuration \mathcal{C} , all robots are on a single line where each non terminal robots have color `chord` and the terminal robots have color either `chord` or `moving1`. Then In finite time the configuration will change into a Phase 1 Final Configuration (P1FC).*

Proof. By the theorem 3.13, there exists a time t when all robots will be on a single line. All robots on that line which are not terminal must have color `chord` at time t and the terminal robots can have color either `moving1` or, `chord` at time t . Let both terminal robots, say r_1 and r_2 , has color `moving1` at time t . Then among r_1 and r_2 , whichever is activated first, say r_1 without loss of generality, must see a robot of color `chord` on $\mathcal{L}_V(r_1) = \mathcal{L}_1$ and changes its color to `chord`. The guarantee that r_1 will see a robot of color `chord` comes from the fact that the non terminal robots with color `chord` on \mathcal{L}_1 do not move out of $\mathcal{L}_V(r_1)$ until they see a robot with color `diameter` on $\mathcal{L}_V(r_1)$. So there must exist a time $t_1 > t$ when all robots are on \mathcal{L}_1 and all but at most one terminal robot say r_2 has color `chord` and r_2 has either color `chord` or `moving1`. Now there are two cases. In the first case, we assume all robots have color `chord` at t_1 , and in the second case we assume all robots except r_2 have color `chord` and r_2 has color `moving1`.

Case I: Let us consider the case when all robots on \mathcal{L}_1 has color **chord** at time t_1 . Now after t_1 whichever robot of r_1 and r_2 is activated first, changes its color to **diameter**. Note that both terminal robots can have color **diameter** too. So, there exists a time $t_2 > t_1$ when all robots are on \mathcal{L}_1 , all non terminal robots have color **chord** and at least one terminal robot has color **diameter**.

Case I(a): Now if both the terminal robots r_1 and r_2 have color **diameter** at time t_2 (Figure 11(a)), then the non terminal robots on \mathcal{L}_1 that can see r_1 or r_2 with color **diameter** moves either left or right once by executing CHORDMOVE subroutine. After a robot moves to the left or right of $\mathcal{L}_V(r_1)$ by executing CHORDMOVE, the next non terminal robot on $\mathcal{L}_V(r_1)$ can now see at least one robot of color **diameter** and thus execute the CHORDMOVE subroutine. This way all non terminal robots on $\mathcal{L}_V(r_1)$ will execute CHORDMOVE at least once. Note that a robot that has executed CHORDMOVE once does not move again until Phase 1 Final configuration is achieved. This is because the robot does not execute Phase 1 as it will see a robot of color **diameter** which is not on its own vertical line. Also, it will not execute Phase 2 as it never sees only two robots with color **diameter** on its right immediate (or, left immediate) vertical line until Phase 1 Final Configuration is achieved. Now Let t_3 be a time the last non terminal robot on $\mathcal{L}_V(r_1)$ moved right or left after executing CHORDMOVE. Then we can ensure that only r_1 and r_2 are on \mathcal{L}_2 with color **diameter** at time t_3 and all other robots are on \mathcal{L}_1 and \mathcal{L}_3 with color **chord** and also they are strictly between $\mathcal{L}_H(r_1)$ and $\mathcal{L}_H(r_2)$. Hence at t_3 , the configuration becomes a Phase 1 Final Configuration.

Case I(b): Now let at time t_2 , only r_1 has color **diameter** and r_2 has color **chord** (Figure 11(b)). Now if r_2 is activated before any of the non terminal robots move by executing CHORDMOVE it changes its color to **diameter**. For this case, we can show that eventually Phase 1 Final Configuration will be achieved (using a similar argument as Case I(a)). So, let before r_2 is activated a robot which is not terminal on \mathcal{L}_1 and sees r_1 executes CHORDMOVE and moves left. Then r_2 never changes its color to **diameter** until all non terminal robots move left or right of $\mathcal{L}_V(r_1) = \mathcal{L}_V(r_2)$ by executing CHORDMOVE. When all such robots move, r_2 then can see r_1 with color **diameter** and then it changes its color to **diameter**. Note that after r_2 changes its color **diameter** the configuration becomes a Phase 1 Final Configuration.

Case II: Let at time t_1 all but one terminal robot, say r_2 (without loss of generality), on \mathcal{L}_1 has color **chord**. Let r_2 has color **moving1** at time t_1 . Now, if r_1 is activated and sees a robot of color **chord** on its vertical line and sees no robot on both of its left and right open halves then r_1 changes its color to **diameter**. Let this happens at a time $t_4 > t_1$. Then similar to the above discussion all non terminal robots on $\mathcal{L}_V(r_1)$, that see r_1 after time t_4 , on their own vertical line executes CHORDMOVE and move left or right. If at least one such non terminal robot has already executed CHORDMOVE then even if r_2 is activated after that, with color **moving1** it does not change its color as it sees its left open half non-empty.

Case II(a): If r_2 is activated before any robot moves from $\mathcal{L}_V(r_1) = \mathcal{L}_V(r_2)$ then it sees a robot with color **chord** on $\mathcal{L}_V(r_2)$ and sees $H_L^O(r_2)$ empty. Thus r_2 in this case changes its color to **chord** at a time say $t_5 > t_1$. In the configuration, if $t_5 > t_4$ then at time t_5 , all robots have color **chord** except r_1 , which has color **diameter**. This is similar to the case I(b). Thus for this case eventually the configuration will become a Phase 1 Final Configuration. So, let $t_5 \leq t_4$. If $t_5 < t_4$, then at t_5 , all robots will be on a single line with color **chord**. This is similar to case I and thus eventually the configuration will change to a Phase 1 Final Configuration. Now if $t_5 = t_4$, then at t_5 , r_1 will have color **diameter** and r_2 will have color **chord** which is similar to the case I(b). Thus again the configuration will eventually become a Phase 1 Final

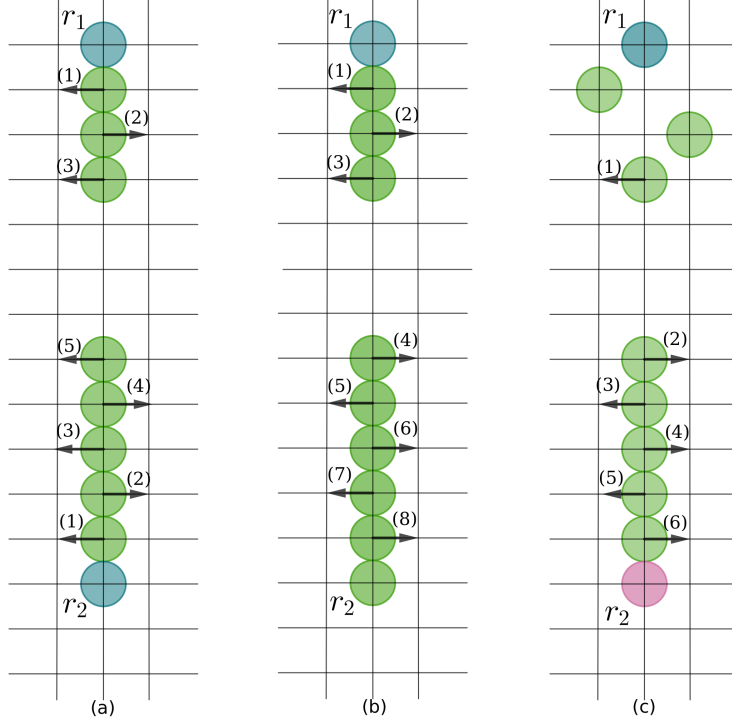


Figure 11. (a) Both terminal robots have color **diameter** in $\mathcal{C}(t_2)$. (b) r_1 has color **diameter** and r_2 has color **chord** in $\mathcal{C}(t_2)$. r_2 does not change color **diameter** until all non terminal robots of color **chord** execute CHORDMOVE exactly once and moves to left or right. (c) r_2 still has color **moving1** while some non terminal robots from the line already executed CHORDMOVE. In this case also r_2 changes color to **diameter** when all non terminal robots moves left or right. The numbers in the bracket denotes the order in which the robots move.

Configuration.

Case II(b): Now let after t_1 , r_2 is activated for the first time when at least one non terminal robot already executed CHORDMOVE (Figure 11(c)). Then r_2 does not change its color even if it is activated as it does not have $H_L^O(r_2)$ non-empty. Now When all non terminal robots execute CHORDMOVE once then on $\mathcal{L}_V(r_2)$ there are only two robots r_1 with color **diameter** and r_2 with color **moving1**. Now when r_2 is activated again, it sees r_1 with color **diameter** on $\mathcal{L}_V(r_2)$ and changes its color to **diameter**. After this, the configuration again becomes a Phase 1 Final Configuration.

For all of the above cases, it can be ensured that A configuration where all robots are on a single line with all non terminal robots having color **chord** and terminal robots having color either **chord** or, **moving1** will change into a Phase 1 Final Configuration within finite time. \square

3.2. Phase 2

Phase 1 terminates when the configuration is a P1FC. So, the initial configuration in Phase 2 is a P1FC where r_1 and r_2 be the only robots on \mathcal{L}_2 and they have color **diameter**. Note that these two robots are already terminated in Phase 1 after changing color to **diameter**. Now, these two robots will help the other robots to agree on the circle to be formed. The line segment of $\mathcal{L}_V(r_1)$ between r_1 and r_2 will be agreed by other robots as a diameter of the circle to be formed. So, if a robot sees both r_1 and

r_2 it knows the circle, say \mathcal{CIR} .

Now, observe that in a P1FC, each horizontal line between $\mathcal{L}_H(r_1)$ and $\mathcal{L}_H(r_2)$ contains at most one robot. Now each of these horizontal lines has exactly two grid points on the circumference of the circle \mathcal{CIR} , one on the left of the agreed diameter and the other on the right. So, for all $r \in \mathfrak{R}$, if r terminates at a grid point on $\mathcal{L}_H(r)$ which is on the circumference of \mathcal{CIR} , we say that circle is formed. In Phase 2 of algorithm *CF_FAT_GRID*, any robot r terminates at the grid point on $\mathcal{L}_H(r)$ which is on the circumference of \mathcal{CIR} and on the left (resp. right) of the agreed diameter if in the P1FC r is on the left (resp. right) of the agreed diameter.

3.2.1. Brief Description of Phase 2

Phase 2 starts when there are exactly two robots with color **diameter**, say r_1 and r_2 on the same vertical line, and all other robots have color **chord** on $\mathcal{L}_I(r_1)$ and $\mathcal{R}_I(r_1)$. Difference between number of robots on $\mathcal{L}_I(r_1)$ and $\mathcal{R}_I(r_1)$ is at most two. Note that the vertical line $\mathcal{L}_V(r_1)$ divides the circle on two halves $H_L^O(r_1)$ and $H_R^O(r_1)$. Here, we describe Phase 2 for the robots only in $H_L^O(r_1)$. The algorithm for the robots in $H_R^O(r_1)$ will be similar.

Note that due to the procedure **CHORDMOVE** in Phase 1, all robots with color **chord** must be strictly between the horizontal lines passing through r_1 and r_2 . Observe that, in Phase 1, if a robot r of color **chord** sees at least one robot of color **diameter** then there are two possibilities. Either r sees at least one robot of color **diameter**, say r_1 , on $\mathcal{L}_V(r)$ or, it sees r_1 on $\mathcal{R}_I(r)$ (resp. $\mathcal{L}_I(r)$) while there must be another robot of color **chord** on $\mathcal{R}_I(r)$ (resp. $\mathcal{L}_I(r)$). In Phase 2 we have shown that (Lemma 3.19) a robot r of color **chord** always sees at least one robot of color **diameter** which can not be on $\mathcal{L}_V(r)$ (as r never moves to the vertical line where the robots of color **diameter** are located). Now if r sees a robot r_1 of color **diameter** on $\mathcal{R}_I(r)$ (resp. $\mathcal{L}_I(r)$) then, it must see the other robot r_2 of color **diameter** on $\mathcal{R}_I(r)$ (resp. $\mathcal{L}_I(r)$) too. But the difference from Phase 1 is that, here $\mathcal{L}_V(r_1)$ does not have any other robots of color **chord**. Thus a robot of color **chord** can always distinguish between Phase 1 and Phase 2.

Now, a robot which is terminal on $\mathcal{L}_I(r_1)$, say r , must see both the robots r_1 and r_2 . In this case, it moves toward its left. Before this move r changes its color to **off** only if the horizontal distance of r from r_1 or r_2 is $\lceil \frac{d}{2} \rceil - 1$. Observe that due to this rule after a finite time all robots which were initially on $\mathcal{L}_I(r_1)$ with color **chord**, reach a vertical line which is $\lceil \frac{d}{2} \rceil$ distance away from r_1 and r_2 with color **off**. Now we claim that a robot with color **off** always sees at least one of r_1 or r_2 (Lemma 3.20). By this condition, a robot with color **off**, can identify whether it is in Phase 1 or Phase 2. Let us name the vertical line $\mathcal{L}_V(r_1)$ as v_0 , and v_i be the i -th vertical line on the left of v_0 . So, after a finite time, all robots will be on $v_{\lceil \frac{d}{2} \rceil}$ strictly between $\mathcal{L}_H(r_1)$ and $\mathcal{L}_H(r_2)$ having color **off**. In this configuration, all the robots can see both r_1 and r_2 and thus can calculate the point c which is equidistant from both r_1 and r_2 on v_0 . There can be at most two robots on $v_{\lceil \frac{d}{2} \rceil}$ which are nearest to c . Also, if there are two such robots then there can be no other robots strictly between them on $v_{\lceil \frac{d}{2} \rceil}$. Thus a nearest robot to c on $v_{\lceil \frac{d}{2} \rceil}$ can understand if it is nearest to c . Such a robot, say r , first moves right after changing the color to **moving1** and does not move further until it sees no other robots on its left. Now, after r moves, another robot, say r' , on $v_{\lceil \frac{d}{2} \rceil}$ moves right only if it sees r with color **moving1** on its immediate right vertical line $v_{\lceil \frac{d}{2} \rceil - 1}$ and sees no other robot on the left of v_0 strictly between $\mathcal{L}_H(r)$ and $\mathcal{L}_H(r')$. Thus there exists a

time when all robots on the left of v_0 are on $v_{\lceil \frac{d}{2} \rceil - 1}$ and have color **moving1**. We call this configuration a $(\lceil \frac{d}{2} \rceil - 1)$ -Left Sub Circle Configuration ($(\lceil \frac{d}{2} \rceil - 1)$ -LSCC). More formally,

Definition 3.15 (j -Left Sub Circle Configuration Left (j -LSCC)). A configuration where r_1 and r_2 with color **diameter** be the only two robots on v_0 is called a j -Left sub circle configuration (Figure 12) if

- (1) All robots of color **moving1** on the left of v_0 are the only robots on vertical line v_j .
- (2) There are no robots strictly between v_j and v_0 .
- (3) All robots on the left of v_0 are strictly between $\mathcal{L}_H(r_1)$ and $\mathcal{L}_H(r_2)$ and each horizontal line contains at most one robot.
- (4) All robots on left of v_j (if any) must be of color **done**

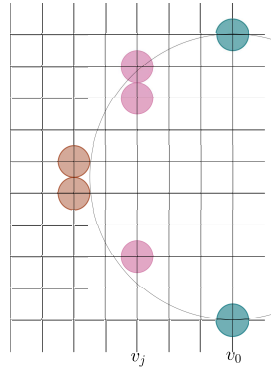


Figure 12. j -LSCC where on v_0 there are only two robots of color **diameter**, all robots of color **moving1** in the configuration are on v_j and all other robots are on left of v_j with color **done**.

Let r be a robot on the left of v_0 that sees both r_1 and r_2 with color **diameter**. Then it can find out the point c which is equidistant from r_1 and r_2 on v_0 . Let \mathcal{CIR} be the circle with center at c and radius $\frac{d}{2}$ where d is the length between r_1 and r_2 along v_0 . Let $\mathcal{L}_\perp(r)$ be the line through r and perpendicular to v_0 . Let C_r be the point of intersection of \mathcal{CIR} and $\mathcal{L}_\perp(r)$. We say that a robot r is on \mathcal{CIR} if $0 \leq \text{distance}(r, C_r) < 1$ and r is on left of C_r on $\mathcal{L}_\perp(r)$.

In a j -LSCC configuration we can divide the robots of color **moving1** in two classes namely, \mathcal{IN}_j and \mathcal{OUT}_j . We say that a robot of color **moving1** in a j -LSCC is in \mathcal{IN}_j if it is strictly inside the circle \mathcal{CIR} otherwise, it is in \mathcal{OUT}_j (i.e., when the robot is either strictly outside or on the circle \mathcal{CIR}).

Now a robot of color **moving1** can distinguish Phase 2 from Phase 1 from the fact that a robot of color **moving1** always sees at least one robot of color **diameter** not on its vertical line in Phase 2 (Lemma 3.23). Let r be a robot of color **moving1** in Phase 2. If r on some v_j ($j > 1$), sees both r_1 and r_2 with color **diameter** on $\mathcal{R}_I(r)$ and no robot of color **off** or **moving1** visible to r on $\mathcal{L}_I(r)$, also if r is nearest to c , then r moves left after changing its color to **done** only if r is strictly inside the circle \mathcal{CIR} . Otherwise, if r is on or strictly outside the circle then, it moves right. Now if $j = 1$ then r moves left after changing the color to **done** when r is strictly inside the circle, nearest to c and sees no robot of color **off** or **moving1** on $\mathcal{L}_I(r)$. Otherwise, if it is on the circle, it changes its color to **done** and terminates on $\mathcal{L}_V(r) = v_1$. If r with color **moving1** sees another robot r' of color **moving1** on its right then it only moves right

when it sees there is no other robot inside the rectangle bounded by the lines $\mathcal{L}_H(r)$, $\mathcal{L}_H(r')$, $\mathcal{L}_V(r)$ and $\mathcal{L}_V(r')$.

We now have the following observations. Also, the pseudo code of Phase 2 is presented in Algorithm 2.

Observation 3.16. *In Phase 2, there can not be a configuration where two robots exists such that one is of color **chord** and another is color **moving1**.*

Observation 3.17. *In Phase 2, if a configuration has a robot r_c of color **chord** and a robot r_o of color **off** on the left (respectively right) of v_0 then the r_o must have to be on $H_L^C(r_c)$ (respectively $H_R^C(r_c)$).*

Observation 3.18. *In Phase 2, if a configuration has a robot r_m of color **moving1** and a robot r_o of color **off** on the left (respectively right) of v_0 then the r_o must have to be on $H_L^C(r_m)$ (respectively $H_R^C(r_m)$).*

3.2.2. Correctness of Phase 2

To prove the correctness of Phase 2 we have to prove two things.

- (1) For a robot r on the left (resp. right) of the agreed diameter i.e., v_0 , if r terminates, then it terminates on the grid point on the circumference of the agreed circle on left (resp. right) of v_0 which is on $\mathcal{L}_H(r)$ (Lemma 3.25)
- (2) Except r_1 and r_2 (the robots of color **diameter**), all other robots terminate in Phase 2. (Lemma 3.26).

For r_1 and r_2 , they are already terminated on Phase 1 on the two endpoints of the agreed diameter, so they are also on the circle. Now to prove these two things we have some other lemmas that will be useful for the proof. In the following, we prove these lemmas along with the two above-mentioned results and summarize the main result in Theorem 3.27.

Lemma 3.19. *In Phase 2, a robot r with color **chord** always sees a robot of color **diameter**.*

Proof. Let r_1 and r_2 be the only two robots of color **diameter** on the vertical line v_0 at a time, say t in Phase 2. Let at time t , r be a robot of color **chord** on v_{i_1} on the left of v_0 that can not see both r_1 and r_2 . Then there must exist two robots r'_1 and r'_2 strictly inside the rectangles bounded by v_{i_1} , $\mathcal{L}_H(r)$, $\mathcal{L}_H(r_1)$, v_0 and v_{i_1} , $\mathcal{L}_H(r)$, $\mathcal{L}_H(r_2)$, v_0 respectively (Figure 13). Note that r'_1 and r'_2 must have color **chord** (due to Observation 3.16 and Observation 3.17). Let $\mathcal{L}_V(r'_1)$ is the vertical line v_{i_2} and $\mathcal{L}_V(r'_2)$ is the vertical line v_{i_3} where $i_1 > i_2 \geq i_3 > 0$. We claim that $i_2 = i_3$. Otherwise, let at time t , $i_2 > i_3$. Now note that, r must have moved left from v_{i_2} before t as $i_1 > i_2$. Thus, there must exist a time $t_1 < t$ when r gets activated on v_{i_2} and sees it is terminal on v_{i_2} and sees r_1 and r_2 on $\mathcal{R}_I(r)$ and then moves left to v_{i_1} . Thus, at t_1 , $\mathcal{R}_I(r) = v_0$. This implies r'_2 must be on the left or on v_{i_2} at time t_1 . This is not possible as r'_2 is on v_{i_3} at time $t > t_1$ and a robot of color **chord** on the left of v_0 never moves right. Thus at time t , r'_1 and r'_2 must be on same vertical line say v_{i_2} . Now at t , r is on v_{i_1} which is on the left of v_{i_2} . This implies there exists a time before t when r moved left from v_{i_2} . Thus there exists a time $t_2 < t$ when r is activated on v_{i_2} and sees it is terminal on v_{i_2} and there is no robots between v_{i_2} and v_0 . This implies at t_2 , r'_1 and r'_2 must be on v_{i_2} . So at t_2 , r can not be terminal on v_{i_2} . The contradiction arises because our assumption that r can not see both r_1 and r_2 is false. Hence, r must see at least one

Algorithm 2: Phase 2

```
1 Procedure PHASE2()
2    $r \leftarrow$  myself
3    $d =$  distance between two robots with color diameter
4   robot with color diameter is on right (resp. left) open half of  $r$ 
5    $c =$  midpoint of two robots with color diameter
6   if  $r.color = chord$  then
7     if  $r$  sees only two robots with color diameter on  $\mathcal{R}_I(r)$  (resp.  $\mathcal{L}_I(r)$ ) then
8       if  $r$  is terminal then
9         if horizontal distance from robot with color diameter  $= \lceil \frac{d}{2} \rceil - 1$  then
10           $r.color = off$ 
11          move horizontally away from robot with color diameter
12        else
13          move horizontally away from robot with color diameter
14      else if  $r.color = off$  then
15        if  $r$  sees at least one robot of color diameter then
16          if  $r$  sees exactly two robots on  $\mathcal{R}_I(r)$  (resp.  $\mathcal{L}_I(r)$ ) with color diameter
17            and  $r$  is nearest to the center  $c$  then
18               $r.color = moving1$ ;
19              move towards the robot of color diameter;
20          else if  $r$  sees  $r'$ , a robot of color moving1 on  $\mathcal{R}_I(r)$  (resp.  $\mathcal{L}_I(r)$ ) such that
21            no robots between  $\mathcal{L}_H(r)$  and  $\mathcal{L}_H(r')$  then
22               $r.color = moving1$ ;
23              move horizontally towards the robot of color diameter;
24      else if  $r.color = moving1$  then
25        if  $r$  sees at least one robot of color diameter not on  $\mathcal{L}_V(r)$  then
26          if  $\mathcal{R}_I(r)$  (resp.  $\mathcal{L}_I(r)$ ) has exactly two robots of color diameter then
27            if  $\mathcal{R}_I(r)$  (resp.  $\mathcal{L}_I(r)$ ) is more than one hop away from  $\mathcal{L}_V(r)$  then
28              if  $r$  is nearest to  $c$  then
29                if there is no robot of color off or moving1 on  $\mathcal{L}_I(r)$  (resp.
30                   $\mathcal{R}_I(r)$ ) then
31                  if  $r$  is strictly inside of CIR then
32                     $r.color = done$ ;
33                    move left (right resp.);
34                  else
35                    move right (left resp.);
36            else
37              if  $r$  is nearest to  $c$  then
38                if there is no robot of color off or moving1 on  $\mathcal{L}_I(r)$  (resp.
39                   $\mathcal{R}_I(r)$ ) then
40                  if  $r$  is strictly inside of CIR then
41                     $r.color = done$ ;
42                    move left (right resp.);
43                  else
44                     $r.color = done$ 
45          else if  $\mathcal{R}_I(r)$  ( $\mathcal{L}_I(r)$  resp.) has a robot  $r'$  of color moving1 such that no
46            robots on or strictly inside the rectangle bounded by  $\mathcal{L}_H(r)$ ,  $\mathcal{L}_H(r')$ ,  $\mathcal{L}_V(r)$ 
47            and  $\mathcal{L}_V(r')$  except  $r$  and  $r'$  then
48              move right (left resp.);
49      else if  $r.light = done$  then
50        terminate;
```

robot of color **diameter** in Phase 2.

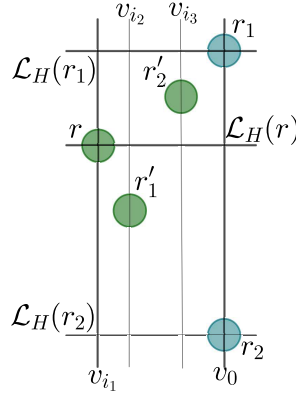


Figure 13. r_1 and r_2 is on v_0 with color **diameter**, r is on v_{i_1} with color **chord** which can not see r_1 and r_2 due to r'_1 on v_{i_2} and r'_2 on v_{i_3} .

□

Lemma 3.20. *In Phase 2, a robot of color **off** always sees a robot of color **diameter**.*

Proof. Let r_1 and r_2 be the only two robots of color **diameter** on the vertical line v_0 at a time, say t in Phase 2. Let r be a robot of color **off** in Phase 2 that does not see r_1 and r_2 at t . So at time t , r must be on $v_{\lceil \frac{d}{2} \rceil}$ and there must be two robots r'_1 and r'_2 strictly inside the rectangles bounded by $\mathcal{L}_H(r)$, $\mathcal{L}_H(r_1)$, $v_0, v_{\lceil \frac{d}{2} \rceil}$ and $\mathcal{L}_H(r)$, $\mathcal{L}_H(r_2)$, $v_0, v_{\lceil \frac{d}{2} \rceil}$ respectively (Figure 14). Now, r'_1 and r'_2 both can be of color either **chord** or of color **moving1** (by Observation 3.16 Observation 3.17, Observation 3.18). Now, if both of r'_1 and r'_2 is of color **chord**, then we reach contradiction by arguing similarly as in Lemma 3.19. So, let us consider the case where both r'_1 and r'_2 are of color **moving1** at time t .

Let at t , r is on $v_{\lceil \frac{d}{2} \rceil}$. Also, r'_1 and r'_2 are on right of $v_{\lceil \frac{d}{2} \rceil}$. $\mathcal{L}_H(r)$ divides the grid in two halves. Let the half where r'_1 is located at time t be denoted as the upper half and the half where r'_2 is located at t be denoted as the lower half. Now there exists a time $t' < t$ such that all robots on the left of v_0 have color **off** and are on $v_{\lceil \frac{d}{2} \rceil}$. The first robot that moves right from $v_{\lceil \frac{d}{2} \rceil}$ after changing color to **moving1** must be nearest to c at time t' . Let r_{i_1} be such a robot. Note that r_{i_1} is not r . So without loss of generality let it is in the upper half at t' . Now there can be at most another robot, say r_{i_2} , which is also nearest to c at time t' . If this is the case then r_{i_2} must be in the upper half at time t' as otherwise r becomes nearer to c than r_{i_1} and r_{i_2} contrary to the assumption. So, the first robot, say r_f , from lower half that moves right from $v_{\lceil \frac{d}{2} \rceil}$ must have moved after seeing a robot, say r'_f of color **moving1** on $\mathcal{R}_I(r_f)$ such that there is no other robots strictly between $\mathcal{L}_H(r_f)$ and $\mathcal{L}_H(r'_f)$ at some time t_1 where $t > t_1 > t'$. Note that r'_f must be on the upper half at t_1 . So, at t_1 , the region strictly between $\mathcal{L}_H(r_f)$ and $\mathcal{L}_H(r'_f)$ can not be empty as r is there. So, we arrive at a contradiction due to the wrong assumption that there exists a time t such that in the configuration at time t there exist a robot of color **off** that can not see both r_1 and r_2 of color **diameter**. Thus any robot of color **off** always sees at least one robot of color **diameter**.

□

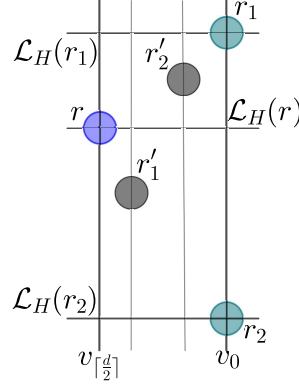


Figure 14. r_1 and r_2 is on v_0 with color **diameter**, r is on $v_{\lceil \frac{d}{2} \rceil}$ with color **off** which can not see r_1 and r_2 due to r'_1 and r'_2 .

Now, we have to prove that a robot of color **moving1** always sees at least one robot of color **diameter** in Phase 2 which is not on its own vertical line. This will ensure that a robot of color **moving1** distinguishes Phase 2 from Phase 1. This is because in Phase 1 even if a robot of color **moving1** sees a robot of color **diameter** it must be on its own vertical line. To prove this we have to prove the following lemma first.

Lemma 3.21. *Let at a time t , a configuration is a j -LSCC, where $j > 1$ and $OUT_j \neq \phi$. Then,*

- (1) *there exists a time $t_1 \geq t$ such that at time t_1 , the configuration is again a j -LSCC where $IN_j = \phi$ and OUT_j at time $t_1 = OUT_j$ at time t .*
- (2) *Moreover, there is another time $t_2 > t_1$ such that at t_2 , the configuration is a $(j-1)$ -LSCC and $IN_{j-1} \cup OUT_{j-1}$ at time $t_2 = OUT_j$ at time t_1 .*

Proof. Let at time t the configuration is a j -LSCC where $j > 1$. In this case if $IN_j = \phi$ then we have nothing to prove for the first part as $t_1 = t$. So, let at time t , $IN_j \neq \phi$. Now to prove the first part we have to show that there exists a time $t' > t$ such that at time t' the configuration is again a j -LSCC configuration where $|IN_j|$ at $t' < |IN_j|$ at t and between $[t, t']$ no robot in OUT_j at time t moves even if it is activated. Let at t , $IN_j = \{r_{i_1}, r_{i_2}, \dots, r_{i_p}\}$ where $p \geq 1$. Note that, distance between r_{i_k} and c is strictly less than distance between r and c for any $r \in OUT_j$ and any $k \in \{1, 2, \dots, p\}$ at time t . So at t , a robot nearest to c must be from IN_j . Let r_{i_n} be one such robot. Then upon activation, it moves left after changing the color to **done**. Let t' be the first time instance such that for all $t_x \in [t, t')$, r_{i_n} is on v_j in $\mathcal{C}(t_x)$ and in $\mathcal{C}(t')$, r_{i_n} is on v_{j+1} with color **done**. So, $|IN_j|$ at $t' < |IN_j|$ at t . Now we only have to show that any robot of color **moving1** in OUT_j at time t stays on v_j in the time interval $[t, t']$. If possible let some robots that were in OUT_j at time t move right, on or, before t' . Let r_o be the first such robot to move right. Then r_o must have been activated at some time $t'_1 < t'$ when it is nearest to c . But since $t'_1 < t'$, in $\mathcal{C}(t'_1)$, r_{i_n} was on v_j . Thus in $\mathcal{C}(t'_1)$, r_o can not be nearest to c . So at t' , the configuration is again a j -LSCC configuration where $|IN_j|$ at $t' < |IN_j|$ at t and between $[t, t']$ no robot in OUT_j at time t moves even if it is activated. So, $|OUT_j|$ at time t' remains same to $|OUT_j|$ at time t . This proves eventually there is a time t_1 when the configuration is a j -LSCC with $IN_j = \phi$ and OUT_j at time $t_1 = OUT_j$ at time t .

Now, for the second part, we have to prove that there exists a time $t_2 > t_1$ when

all robots on v_j at t_1 are on v_{j-1} at time t_2 . Thus we have to show that a robot after reaching v_{j-1} from v_j does not do anything until all robots of v_j at time t_1 reach v_{j-1} . If possible let a robot r after reaching v_{j-1} moves again before all robots of v_j move. This implies there exists a time $t'_2 > t_1$ when there are robots on v_j with color `moving1` but r does not see any robot of color `moving1` on v_j from v_{j-1} at time t'_2 . This implies there must exist at least one robot, say r' , on v_j which does not have color `moving1` and which obstructs r from seeing any robot of color `moving1` on v_j at time t'_2 (Figure 15). Note that r' must be of color `done` and it must have moved from v_{j-1} to v_j after changing its color to `done` from `moving1`. This is because at t_1 , $\mathcal{IN}_j = \phi$, so r' has not changed its color to `done` on v_j . So, at t'_2 there must be at least one robot of color `done` on v_j which has moved to v_j from v_{j-1} after t_1 . Without

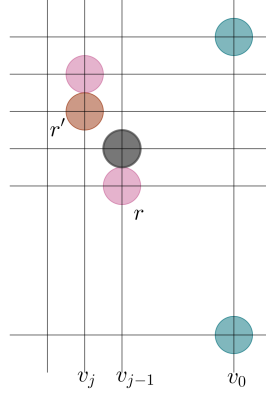


Figure 15. Configuration $\mathcal{C}(t'_2)$. Here r on v_{j-1} has no visible robot of color `moving1` on v_j in spite of v_j having such a robot. r' actually obstructs the view of r .

loss of generality let r' be the first robot that moves to v_j from v_{j-1} after changing its color to `done` from `moving1`. Let r' is activated on v_{j-1} with color `moving1` at some time t'_4 where $t'_2 > t'_4 > t_1$. This implies r' also has not seen any robot on v_j of color `moving1` at time t'_4 . Since $t'_2 > t'_4 > t_1$, there must exist a robot of color `moving1` on v_j at time t'_4 . r' does not see that robot at t'_4 implies there must exist another robot, say r_s of color `done` on v_j at t'_4 . Also, r_s must have moved to v_{j-1} to v_j for the same reason described above. So r' can not be the first robot that moved to v_j from v_{j-1} after changing its color to `done` from `moving1` after t_1 . So no robot that reached v_{j-1} from v_j after t_1 , moves until all robots of v_j at time t_1 moves to v_{j-1} . Let t_2 be the time when the last robot of v_j reaches v_{j-1} after t_1 . Note that all robots of color `moving1` at time t_2 are on v_{j-1} and they are the only ones on v_{j-1} . Also, there are no robots between v_{j-1} and v_0 and all robots on the left of v_{j-1} have color `done`. So this configuration at time t_2 is a $(j-1)$ -LSCC. Also, all robots that were on v_j at time t_1 are now on v_{j-1} at time t_2 and no other robots moved onto v_{j-1} . So, the set of all robots on v_{j-1} at time $t_2 = \mathcal{IN}_{j-1} \cup \mathcal{OUT}_{j-1}$ at time $t_2 = \mathcal{OUT}_j$ at time t_1 . □

We have seen in the description that, in Phase 2 the configuration becomes a $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC configuration. Now this lemma tells that for all $j > 1$ and $j \leq \lceil \frac{d}{2} \rceil - 1$ there is a time when the configuration becomes a $(j-1)$ -SLCC from j -SLCC. So starting from $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC we will eventually have $(\lceil \frac{d}{2} \rceil - 2)$ -LSCC then $(\lceil \frac{d}{2} \rceil - 3)$ -LSCC and so on until we have a 1-LSCC. Also note that from i -LSCC, j -LSCC can not be formed if $j > i$ as robots of color `moving1` that are on the left of v_0 only moves right

in Phase 2. Moreover, we can have the following corollary.

Corollary 3.22. *After $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC is formed, for all $j \in \{1, 2, \dots, \lceil \frac{d}{2} \rceil - 2\}$ and for any $i < j$, a configuration can not have robots on v_i until j -LSCC is formed.*

Proof. Let us fix a $j \in \{1, 2, \dots, \lceil \frac{d}{2} \rceil - 2\}$. Let at time t which is after $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC is formed, the configuration is either a p -LSCC or, t is between t_1 and t_2 such that at t_1 the configuration is a p -LSCC and at t_2 it becomes $(p - 1)$ -LSCC where $j < p \leq \lceil \frac{d}{2} \rceil - 1$. Now by Lemma 3.21, at time t there can be no robots on v_s where $s < p - 1$. Now for any $i < j$, we have $i < j < p \implies i < j \leq p - 1$. Hence at t , there can be no robots on v_i where $i < j$. \square

Now we prove the following lemma that ensures that a robot of color always sees at least one robot of color **diameter** not on its own vertical line in Phase 2.

Lemma 3.23. *In Phase 2, a robot r of color **moving1** can always see a robot of color **diameter** that is not on $\mathcal{L}_V(r)$.*

Proof. Let t be a time when a robot, say r , of color **moving1** on v_j can not see both of r_1 and r_2 , the two robots of color **diameter** on v_0 . Without loss of generality let r be on the left of v_0 . This implies at time t , there must exist two robots r'_1 and r'_2 strictly inside the rectangles bounded by $\mathcal{L}_H(r), \mathcal{L}_H(r_1), v_0, v_j$ and $\mathcal{L}_H(r), \mathcal{L}_H(r_2), v_0, v_j$ respectively (Figure 16) i.e., r'_1 and r'_2 are on v_{i_1} and v_{i_2} where $j > i_1 \geq i_2$. Also, $j > 1$ and $j < \lceil \frac{d}{2} \rceil$ as even if a robot can have color **moving1** on $v_{\lceil \frac{d}{2} \rceil}$ it performs first look phase as a robot of color **moving1** after it moves right. Also from the description $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC is formed first until then no robot moves to v_i where $i < \lceil \frac{d}{2} \rceil - 1$. So until $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC is formed all robots of color **moving1** must see both the robots r_1 and r_2 from $v_{\lceil \frac{d}{2} \rceil - 1}$. So t must be a time after $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC is formed. Observe that, r'_1 and r'_2 must be of color **moving1**. Now since at time t , there are robots on v_{i_1} and v_{i_2} where $i_1, i_2 < j$, there exists a time $t' < t$ when the configuration was a j -LSCC such that $\mathcal{IN}_j = \phi$ (Lemma 3.21 and Corollary 3.22) (Figure 16). Also in $\mathcal{C}(t')$, r, r'_1, r'_2 were on v_j . Now similar to lemma 3.20, let us denote the half where r'_1 is located as the upper half and the other one as the lower half. Now the first robot that moves to v_{j-1} from v_j must be nearest to c at time t' . Without loss of generality let it be on the upper half at time t' . There can be at most another such robot which is also nearest to c at time t' . If another such robot exists then it also has to be on the upper half at time t' otherwise r becomes nearer to c at time t and moves to v_{j-1} before any other robots and thus at t r can not be at v_j as assumed. Also at time t , there is at least one robot of color **moving1** on the lower half. So let r_l be the first robot from the lower half at time t' that has moved to v_{j-1} from v_j . Then it must have moved from v_j after seeing a robot, say r'_l , of color **moving1** on $\mathcal{R}_I(r_l) = v_{j-1}$ such that there is no other robots except r_l and r'_l on or between the rectangle, say R , bounded by $v_j, v_{j-1}, \mathcal{L}_H(r_l)$ and $\mathcal{L}_H(r'_l)$ at a time $t'_1 > t'$ and $t > t'_1$. Now since r_l is the first robot from the lower half to move after t' , r'_l must be from the upper half. Thus at time t'_1 , R must contain r other than r_l and r'_l which is a contradiction. Hence, a robot of color **moving1** always sees a robot of color **diameter**. And according to the algorithm, since r never reaches v_0 in Phase 2, it can not see r_1 and r_2 on its own vertical line. \square

Let r be a robot in Phase 2 on the left of v_0 . Note that $\mathcal{L}_H(r)$ intersects the circle \mathcal{CIR} exactly once at a point C_r on the left of v_0 . Thus, there exists exactly one grid point denoted as, $C_T(r)$ on $\mathcal{L}_H(r)$ such that either $\text{dist}(C_r, C_T(r)) = 0$ or $C_T(r)$ is

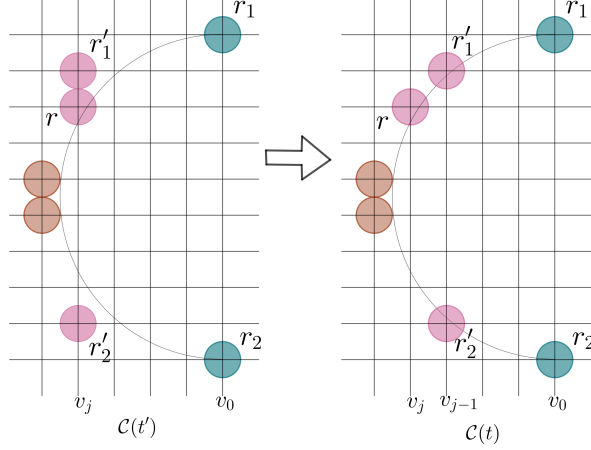


Figure 16. $\mathcal{C}(t')$ is a j -LSCC where all robots on v_j are in \mathcal{OUT}_j . From $\mathcal{C}(t')$, $\mathcal{C}(t)$ is formed where r is still on v_j but r'_1 and r'_2 are on right of v_j obstructing r from seeing r_1 and r_2 . Here r_1 and r_2 are the robots of color `diameter` on v_0 .

on the left of C_r such that $\text{dist}(C_r, C_T(r)) < 1$. Then we define $C_T(r)$ to be the *Terminating Point* of r . We can now have the following observation

Observation 3.24. *In a 1-LSCC if a robot $r \in \mathcal{OUT}_1$, then r must be on $C_T(r)$.*

For r , $C_T(r) = \mathcal{L}_H(r) \cap v_j$ for some $j \geq 1$. Now, in a 1-LSCC if $r \in \mathcal{OUT}_1$, then $j \leq 1$. Hence, $j = 1$.

We now first ensure that if a robot r terminates, it does not terminate on a grid point that is not on $C_T(r)$.

Lemma 3.25. *A robot r can only terminate on $C_T(r)$.*

Proof. For this, first note that a robot can only terminate after $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC is formed. Now let for a robot r , $C_T(r)$ is the grid point $\mathcal{L}_H(r) \cap v_j$ for some $j \in [1, \lceil \frac{d}{2} \rceil] \cap \mathbb{N}$. If $j = \lceil \frac{d}{2} \rceil$ then in $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC, $r \in \mathcal{IN}_{\lceil \frac{d}{2} \rceil - 1}$. Then eventually r moves to $v_{\lceil \frac{d}{2} \rceil}$ after changing the color to `done`. When it is activated next it terminates on $C_T(r) = v_{\lceil \frac{d}{2} \rceil} \cap \mathcal{L}_H(r)$. So if $j = \lceil \frac{d}{2} \rceil$ then r can not terminate on any other position except $C_T(r)$. Now let us consider the case $j < \lceil \frac{d}{2} \rceil$. If possible let r terminates on $\mathcal{L}_H(r) \cap v_i$ where $i > j \geq 1$. Then there must exist a time when r is on v_{i-1} . Then by Corollary 3.22 there exists a time when the configuration is i -LSCC. Here $r \in \mathcal{OUT}_i$ and it is strictly outside the circle \mathcal{CIR} as $i > j$. So, again by lemma 3.21 there exists a time when the configuration becomes a $(i-1)$ -LSCC and r is on v_{i-1} . Here $i-1 \geq j$ i.e $r \in \mathcal{OUT}_{i-1}$. Now if $i-1 = 1$ then r eventually changes the color to `done` and terminates on $v_j = v_1$ as $C_T(r) = \mathcal{L}_H(r) \cap v_1$ for this case (Observation 3.24). This is contrary to our assumption that r terminates on v_i where $i > j = 1$. So, let $i-1 > 1$. Then eventually $(i-2)$ -LSCC will be formed and r will be on v_{i-2} . Now note that according to the algorithm for Phase 2, after $(\lceil \frac{d}{2} \rceil - 1)$ -LSCC is formed, only a robot of color `moving1` can move further from v_0 after changing the color to `done`. So even if a robot moves further from v_0 it can move in such a way only once as after that it terminates. So if r with color `moving1` is on v_{i-2} at some time in Phase 2, it can not move back to v_i and terminate. Thus we reach a contradiction assuming r terminates

on v_i where $i > j$ where $C_T(r) = \mathcal{L}_H(r) \cap v_j$ for some $j \in [1, \lceil \frac{d}{2} \rceil] \cap \mathbb{N}$. Thus r must terminate either on $C_T(r) = \mathcal{L}_H(r) \cap v_j$ or on $\mathcal{L}_H(r) \cap v_i$ where $i < j$. If possible let r terminates on $\mathcal{L}_H(r) \cap v_i$ where $i < j$. This implies there exists a time t when the configuration is a j -LSCC. In this configuration r is on $C_T(r)$ thus $r \in \mathcal{OUT}_j$. Thus eventually $(j-1)$ -LSCC will be formed where r is on v_{j-1} . Note that in this configuration $r \in \mathcal{IN}_{j-1}$. So, eventually, r will change its color to `done` and move to v_j . So r terminates on $\mathcal{L}_H(r) \cap v_j = C_T(r)$ contrary to our assumption. Hence if r terminates it must terminate at $C_T(r)$. \square

Now we will prove that all robots that are not on v_0 terminates

Lemma 3.26. *All robot that are not on v_0 terminates eventually during Phase 2.*

Proof. If possible let r be a robot on the left of v_0 that never terminates in Phase 2. Let $C_T(r) = \mathcal{L}_H(r) \cap v_j$ for some $j \in \{1, 2, \dots, \lceil \frac{d}{2} \rceil\}$. For $j > 1$, r can terminate after it moves to $C_T(r)$ from v_{j-1} of a $(j-1)$ -LSCC after changing the color to `done`. Now as it is assumed that r does not terminate, either $(j-1)$ -LSCC is never formed or, even if it is formed r is not on v_{j-1} in $(j-1)$ -LSCC. Now by Lemma 3.21 if there is at least one robot that has not terminated $(j-1)$ -LSCC will be formed eventually. Thus if r never terminates there can be only one possibility that when $(j-1)$ -LSCC is formed r is not there on v_{j-1} . This implies r must be on some v_i where $i > j$ with color `done`. This implies r terminates at $v_i \cap \mathcal{L}_H(r)$ for some $i > j$. This is impossible due to Lemma 3.25. Hence r must terminate if $j > 1$. Now similarly, for $j = 1$, r never terminates implies, When 1-LSCC is formed, r is not on v_1 . Again this is impossible due to similar reasons as above. Thus all robots that are not on v_0 must terminate in Phase 2 \square

Now using Lemma 3.25 and Lemma 3.26 and the fact that the robots of color `diameter` terminates after changing its color to `diameter` either from `chord` or from `moving1` we can state the following theorem.

Theorem 3.27. *From any initial configuration within finite time, all opaque fat robots with one axis agreement can terminate after forming a circle on an infinite grid under asynchronous scheduler by executing the algorithm `CIRCLE_FG`.*

4. Conclusion

The problem of circle formation is widely studied in the field of swarm robotics. It has been studied under various assumptions on plane. But in the discrete domain, the work is limited. With obstructed visibility model, this problem has been considered on the plane and infinite grid using luminous opaque robots. But using fat robots (i.e. robots with certain dimensions), it was only done in a plane. In this paper, we have taken care of this. We have shown that with a swarm of luminous opaque fat robots having one-axis agreement on an infinite grid, a circle of diameter $O(n)$ (n is the number of robots in the system) can be formed from any initial configuration using one light having 5 distinct colors. For future courses of research, one way is to find out the optimal number of colors needed to solve this problem. Also, it would be interesting to see if the same problem can be solved using disoriented robots.

Acknowledgements: The First author is supported by UGC [1132/(CSIR-UGC NET JUNE 2017)], the Government of India and the third author is supported by the West Bengal State government Fellowship Scheme e [P-1/RS/23/2020].

Disclosure Statement: The authors report there are no competing interests to declare.

References

- [1] Agmon N, Peleg D. Fault-tolerant gathering algorithms for autonomous mobile robots. *SIAM J Comput.* 2006;36(1):56–82. Available from: <https://doi.org/10.1137/050645221>.
- [2] Cicerone S, Stefano GD, Navarra A. Gathering of robots on meeting-points: feasibility and optimal resolution algorithms. *Distributed Comput.* 2018;31(1):1–50. Available from: <https://doi.org/10.1007/s00446-017-0293-3>.
- [3] Goswami P, Sharma A, Ghosh S, et al. Time optimal gathering of myopic robots on an infinite triangular grid. In: Devismes S, Petit F, Altisen K, et al., editors. *Stabilization, Safety, and Security of Distributed Systems - 24th International Symposium, SSS 2022, Clermont-Ferrand, France, November 15-17, 2022, Proceedings; (Lecture Notes in Computer Science; Vol. 13751)*. Springer; 2022. p. 270–284. Available from: https://doi.org/10.1007/978-3-031-21017-4_18.
- [4] Molla AR, Jr WKM. Dispersion of mobile robots. In: *ICDCN '22: 23rd International Conference on Distributed Computing and Networking, Delhi, AA, India, January 4 - 7, 2022*. ACM; 2022. p. 217–220. Available from: <https://doi.org/10.1145/3491003.3493373>.
- [5] Kaur T, Mondal K. Distance-2-dispersion: Dispersion with further constraints. In: Mohaisen D, Wies T, editors. *Networked Systems - 11th International Conference, NETYS 2023, Benguerir, Morocco, May 22-24, 2023, Proceedings; (Lecture Notes in Computer Science; Vol. 14067)*. Springer; 2023. p. 157–173. Available from: https://doi.org/10.1007/978-3-031-37765-5_12.
- [6] Bramas Q, Devismes S, Lafourcade P. Infinite grid exploration by disoriented robots. In: Georgiou C, Majumdar R, editors. *Networked Systems - 8th International Conference, NETYS 2020, Marrakech, Morocco, June 3-5, 2020, Proceedings; (Lecture Notes in Computer Science; Vol. 12129)*. Springer; 2020. p. 129–145. Available from: https://doi.org/10.1007/978-3-030-67087-0_9.
- [7] Darwich O, Ulucan A, Bramas Q, et al. Perpetual torus exploration by myopic luminous robots. In: Devismes S, Petit F, Altisen K, et al., editors. *Stabilization, Safety, and Security of Distributed Systems - 24th International Symposium, SSS 2022, Clermont-Ferrand, France, November 15-17, 2022, Proceedings; (Lecture Notes in Computer Science; Vol. 13751)*. Springer; 2022. p. 164–177. Available from: https://doi.org/10.1007/978-3-031-21017-4_11.
- [8] Flocchini P, Prencipe G, Santoro N, et al. Arbitrary pattern formation by asynchronous, anonymous, oblivious robots. *Theor Comput Sci.* 2008;407(1-3):412–447. Available from: <https://doi.org/10.1016/j.tcs.2008.07.026>.
- [9] Ghosh S, Goswami P, Sharma A, et al. Move optimal and time optimal arbitrary pattern formations by asynchronous robots on infinite grid. *Int J Parallel Emergent Distributed Syst.* 2023;38(1):35–57. Available from: <https://doi.org/10.1080/17445760.2022.2124411>.
- [10] Flocchini P, Prencipe G, Santoro N, et al. Gathering of asynchronous robots with limited visibility. *Theor Comput Sci.* 2005;337(1-3):147–168. Available from: <https://doi.org/10.1016/j.tcs.2005.01.001>.
- [11] Luna GAD, Uehara R, Viglietta G, et al. Gathering on a circle with limited visibility by anonymous oblivious robots. In: Attiya H, editor. *34th International Symposium on Distributed Computing, DISC 2020, October 12-16, 2020, Virtual Conference; (LIPIcs; Vol. 179)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik; 2020. p. 12:1–12:17. Available from: <https://doi.org/10.4230/LIPIcs.DISC.2020.12>.
- [12] Adhikary R, Kundu MK, Sau B. Circle formation by asynchronous opaque

- robots on infinite grid. *Computer Science*. 2021 Feb;22(1). Available from: <https://journals.agh.edu.pl/csci/article/view/3840>.
- [13] Feletti C, Mereghetti C, Palano B. Uniform circle formation for fully, semi-, and asynchronous opaque robots with lights. *Applied Sciences*. 2023;13(13). Available from: <https://www.mdpi.com/2076-3417/13/13/7991>.
- [14] Feletti C, Mereghetti C, Palano B. Uniform circle formation for swarms of opaque robots with lights: 20th international symposium, sss 2018, tokyo, japan, november 4–7, 2018, proceedings; 2018. p. 317–332.
- [15] Tanaka O. Forming a circle by distributed anonymous mobile robots. Bachelor thesis, Department of Electrical Engineering, Hiroshima University, Hiroshima, Japan. 1992;.
- [16] Kundu MK, Goswami P, Ghosh S, et al. Arbitrary pattern formation by opaque fat robots on infinite grid. *Int J Parallel Emergent Distributed Syst*. 2022;37(5):542–570. Available from: <https://doi.org/10.1080/17445760.2022.2088750>.
- [17] Bose K, Adhikary R, Kundu MK, et al. Arbitrary pattern formation by opaque fat robots with lights. In: Changat M, Das S, editors. *Algorithms and Discrete Applied Mathematics - 6th International Conference, CALDAM 2020, Hyderabad, India, February 13-15, 2020, Proceedings*; (Lecture Notes in Computer Science; Vol. 12016). Springer; 2020. p. 347–359. Available from: https://doi.org/10.1007/978-3-030-39219-2_28.
- [18] Dutta A, Gan Chaudhuri S, Datta S, et al. Circle formation by asynchronous fat robots with limited visibility. 02; 2012. p. 83–93.
- [19] Sugihara K, Suzuki I. Distributed motion coordination of multiple mobile robots. *Proceedings 5th IEEE International Symposium on Intelligent Control 1990*. 1990;:138–143 vol.1 Available from: <https://api.semanticscholar.org/CorpusID:61599595>.
- [20] Défago X, Souissi S. Non-uniform circle formation algorithm for oblivious mobile robots with convergence toward uniformity. *Theor Comput Sci*. 2008;396(1-3):97–112. Available from: <https://doi.org/10.1016/j.tcs.2008.01.050>.
- [21] Adhikary R, Kundu MK, Sau B. Circle formation by asynchronous opaque robots on infinite grid. *Comput Sci*. 2021;22(1). Available from: <https://doi.org/10.7494/csci.2021.22.1.3840>.
- [22] Suzuki I, Yamashita M. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM Journal on Computing*. 1999;28(4):1347–1363. Available from: <https://doi.org/10.1137/S009753979628292X>.
- [23] Défago X, Konagaya A. Circle formation for oblivious anonymous mobile robots with no common sense of orientation. In: *Proceedings of the Second ACM International Workshop on Principles of Mobile Computing*; New York, NY, USA. Association for Computing Machinery; 2002. p. 97–104; POMC '02. Available from: <https://doi.org/10.1145/584490.584509>.
- [24] Flocchini P, Prencipe G, Santoro N, et al. Distributed computing by mobile robots: uniform circle formation. *Distributed Comput*. 2017;30(6):413–457. Available from: <https://doi.org/10.1007/s00446-016-0291-x>.
- [25] Ito Y, Kim Y, Katayama Y. Brief announcement: Mutually-visible uniform circle formation by asynchronous mobile robots on grid plane. In: *International Symposium on Stabilizing, Safety, and Security of Distributed Systems*; 2022. p. 352–357.