



HAL
open science

Asynchronous Pattern Formation without Chirality

Quentin Bramas, Sébastien Tixeuil

► **To cite this version:**

Quentin Bramas, Sébastien Tixeuil. Asynchronous Pattern Formation without Chirality. [Research Report] Université Pierre et Marie Curie. 2015. hal-01184532v1

HAL Id: hal-01184532

<https://hal.science/hal-01184532v1>

Submitted on 15 Aug 2015 (v1), last revised 20 Sep 2017 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Asynchronous Pattern Formation without Chirality

Quentin Bramas^{†‡}

Sébastien Tixeuil^{†‡*}

[†]UPMC Sorbonne Universités, LIP6-CNRS 7606, France

[‡]CNRS, LIP6-CNRS 7606, France

*Institut Universitaire de France, France

Abstract

We propose a new probabilistic pattern formation algorithm for oblivious mobile robots that operate in the ASYNC model. Unlike previous work, our algorithm makes no assumptions about the local coordinate systems of robots (the robots do *not* share a common “North” nor a common “Right”), yet it preserves the ability to form any general pattern (and not just patterns that satisfy symmetricity predicates). Our proposal also gets rid of the previous assumption (in the same model) that robots do not pause while moving (so, our robots really are fully asynchronous), and the amount of randomness is kept low – a single random bit per robot per Look-Compute-Move cycle is used. Our protocol consists in the combination of two phases, a probabilistic leader election phase, and a deterministic pattern formation one. As the deterministic phase does not use chirality, it may be of independent interest in the deterministic context. A straightforward extension of our algorithm permits to form patterns with multiplicity points (provided robots are endowed with multiplicity detection), a new feature in the context of pattern formation that we believe is an important feature of our approach.

1 Introduction

We consider a set of mobile robots that move freely in a continuous 2-dimensional Euclidian space. Each robot repeats a Look-Compute-Move (LCM) cycle [10]. First, it *Looks* at its surroundings to obtain a snapshot containing the locations of all robots as points in the plane, with respect to its ego-centered coordinate system. Based on this visual information, the robot *Computes* a destination and then *Moves* towards the destination. The robots are identical, anonymous and oblivious *i.e.*, the computed destination in each cycle depends only on the snapshot obtained in the current cycle (and not on the past history of execution). The snapshots obtained by the robots are not consistently oriented in any manner.

The literature defines three different models of execution: in the fully synchronous (FSYNC) model, robots execute LCM cycles in a lock-step manner, in the semi-synchronous (SSYNC) model, each LCM cycle is supposed atomic, and in the most general asynchronous (ASYNC) model, each phase of each LCM cycle may take an arbitrary amount of time. This last model enables the possibility that a robot observes another robot while the latter is moving (and moving robots appear in the snapshot *exactly the same way static robots do*), and that move actions are based on obsolete observations.

In this particularly weak model it is interesting to characterize which additional assumptions are needed for the robots to cooperatively perform a given task. In this paper, we consider the pattern formation problem in the most general ASYNC model. The robots start in an arbitrary initial configuration where no two robots occupy the same position, and are given the pattern to be formed as a set of coordinates in their own local coordinate system. An algorithm solves the pattern formation problem is within finite time the robots form the input pattern and remain stationary thereafter.

Related Works. The pattern formation problem has been extensively studied in the deterministic setting [3, 2, 8, 9, 5, 11, 12, 6, 7, 10]. The seminal paper on mobile robots [10] presents a deterministic solution to construct general patterns in the SSYNC model, with the added assumption that robots have access to an infinite non-volatile memory (that is, robots are *not* oblivious). The construction was later refined for the ASYNC model by Bouzid *et al.*, still using a finite number of infinite precision variables.

The search for an oblivious solution to the general pattern formation proved difficult [7]. For oblivious deterministic robots to be able to construct any general pattern, it is required that they agree on a common “North” (that is, a common direction and orientation) but also on a common “Right” (that is, a common chirality), so that robots get to all agree on a common coordinate system. If only a “North” (and implicitly if only a “Right”) is available, then some patterns involving an even number of robots cannot be formed. Relaxing the common coordinate system condition led to a characterization of the patterns that can be formed by deterministic oblivious robots [8, 9, 12]. The best deterministic algorithm so far in the ASYNC model without a common coordinate system [12] proves the following: If ρ denotes the geometric symmetricity of a robot configuration, and I and P denote the initial and target configurations, respectively, then P can be formed if and only if $\rho(I)$ divides $\rho(P)$. All aforementioned deterministic solutions assume that both the input configurations and the target configuration do not have multiplicity points (that is, locations hosting more than one robot), and that robots share a common chirality. Overall, oblivious deterministic algorithms either need a common coordinate system or cannot form any general pattern.

To circumvent those impossibility results, the probabilistic path was taken by Yamauchi and Yamashita [13]. The robots are oblivious, operate in the most general ASYNC model, and can form any general pattern from any general initial configuration, without assuming a common coordinate system. However, their approach [13] makes use of three hypotheses that are not proved to be necessary: (i) all robots share a common chirality, (ii) a robot may not make an arbitrary long pause while moving (more precisely, it cannot be observed twice at the same position by the same robot in two different Look-Compute-Move cycles while it is moving), and (iii) infinitely many random bits are required (a robot requests a point chosen uniformly at random in a continuous segment) anytime access to a random source is performed. While the latter two are of more theoretical interest, the first one is intriguing, as a common chirality was also used extensively

in the deterministic case. The following natural open question raises: is a common chirality a necessary requirement for mobile robot *general* gathering ? As the answer is yes in the deterministic [7] case, we concentrate on the probabilistic case.

Our contribution. In this paper, we propose a new probabilistic pattern formation algorithm for oblivious mobile robots that operate in the ASYNC model. Unlike previous work, our algorithm makes no assumptions about the local coordinate systems of robots (they do *not* share a common “North” nor a common “Right”), yet it preserves the ability to form any general pattern (and not just patterns such that $\rho(I)$ divides $\rho(F)$). Besides relieving the chirality assumption, our proposal also gets rid of the previous assumption [13] that robots do not pause while moving (so, they really are fully asynchronous), and the amount of randomness is kept low – a single random bit per robot is used per use of the random source – (vs. infinitely many previously [13]). Our protocol consists in the combination of two phases, a probabilistic leader election phase, and a deterministic pattern formation one. As the deterministic phase does not use chirality, it may be of independent interest in the deterministic context.

A straightforward extension of our algorithm permits to form patterns with multiplicity points (provided robots are endowed with multiplicity detection), a new feature in the context of pattern formation that we believe is an important feature of our approach.

2 Model and Notations

Robots operate in a 2-dimensional Euclidian space. Each robot has its own local coordinate system. For simplicity, we assume the existence (unknown from the robots) of a global coordinate system. Whenever it is clear from the context, we manipulate points in this global coordinate system, but each robot only sees the points in its local system. We say two set of points A and B are similar, denoted $A \approx B$, if B can be obtained from A by translation, scaling, rotation, or symmetry. A *configuration* P is a set of positions of robots at a given time. Each robot that looks at this configuration may see different (but similar) set of points.

Each time a robot is activated it starts a Look/Compute/Move cycle. After the *look* phase, a robot obtains a configuration P representing the positions of the robots in its local coordinate system. After an arbitrary delay, the robot *computes* a path to a destination. Then, it *moves* toward the destination following the previously computed path. The duration of the move phase, and the delay between two phases, are chosen by an adversary and can be arbitrary long. The adversary decides when robots are activated assuming a *fair* scheduling *i.e.*, in any configuration, all robots are activated within finite time. The adversary also controls the robots movement along their target path and can stop a robot before reaching its destination, but not before traveling at least a distance $\delta > 0$ (δ being unknown to the robots).

An execution of an algorithm is an infinite sequence $P(0), P(1), \dots$ of configurations. Let $\mathcal{P}(\psi)$ be the set of all possible configurations in all possible executions of Algorithm ψ . A robot is static when it is not in the moving phase. A configuration is static if all robots are static (note that this information is not known by the robots). A configuration P is said to be *empty* for an algorithm ψ , denoted $\psi(P) = \emptyset$, if ψ does not order any robot to move, otherwise the configuration is said to be *active*. A configuration $P \in \mathcal{P}(\psi)$ is *terminal* (or *stationary*) for ψ if P is static and empty for ψ .

An algorithm ψ forms a pattern F if, for any execution $P(0), P(1), \dots$, there exists a time t such that $P(t') \approx F$ for all $t' \geq t$. In the sequel, F denotes the pattern to form.

Combination of Algorithms. Since robots are oblivious and the scheduling is asynchronous, we cannot explicitly concatenate several algorithms to be executed in a specific order. However, one can simulate the effect of concatenation of two (or more) algorithms by inferring from the current configuration which algorithm to execute. Implementing this technique is feasible if sub-algorithms have disjoint active sets, and robots do not switch sub-algorithms when in a configuration containing moving robots. We say an algorithm ψ satisfies the *termination awareness property* if empty configurations are terminal.

A *combination of algorithms* is a set Ψ of algorithms that each satisfies the termination awareness property, and having disjoint active sets. The fact that one algorithm is executed before results from the relation \rightsquigarrow , where $\psi_1 \rightsquigarrow \psi_2$ holds if an execution of ψ_1 starting with an active configuration of ψ_1 contains an active configuration of ψ_2 . We say that a combination of algorithms Ψ is partially ordered if the transitive closure of the relation \rightsquigarrow is a partial order on the set Ψ . A partially ordered combination $\Psi = \{\psi_1, \dots, \psi_p\}$ is useful because it terminates if and only if each algorithm ψ_i terminates. Moreover, $\text{terminal}(\Psi) = \bigcap_{i=1}^p \text{terminal}(\psi_i)$ and Ψ satisfies the termination awareness property. We denote $\psi|_A$ the restriction of algorithm ψ to a set of configurations A .

Notations. Let P be a set of points, then $C(P)$ denotes the smallest enclosing circle of P . If P is regular (see Definition 1), then $c(P)$ denotes the center of the regular set, otherwise $c(P)$ denotes the center of $C(P)$. The circle of a robot $r \in P$ is the circle centered at $c(P)$ containing r . We say a robot moves on its circle if its trajectory is contained in its circle. A radial movement is a linear movement whose origin and destination are on the same half-line of origin $c(P)$. We say a robot moves radially if it performs a radial movement.

The pattern to form, F , is given to each robot as a set of points in their local coordinate system. However, at each activation, robots can translate and scale their local coordinate system so that $C(P) = C(F)$. Hence, we suppose in the remainder of the paper that $C(P) = C(F)$, and that the radius of $C(P)$ is the common unit distance (unless otherwise mentioned). This is possible because in our case, the configuration where all robots share the same location (that is, are gathered) is not reachable. For two points a and b , $|a|_b = |a - b|$ denotes the distance between a and b . In a n -robot configuration P , as we are often interested in the distance between a point and the center $c(P)$, we simply write $|a|$ instead of $|a|_{c(P)}$. Also, l_P denotes the distance to $c(P)$ of the second closest point of P to $c(P)$.

For a set S , $|S|$ denotes its cardinal. $\mathcal{D}(a)$ denotes the open disc of radius a centered at $c(P)$. For an open disc D , \overline{D} denotes the closure of D , and $\text{Circum}(D) = \text{Circum}(\overline{D})$ denotes its circumference. The interior and the exterior of a disc or a circle do not include the circumference. We said that a set of points A (or simply a point) *holds* $C(P)$ if $C(P \setminus B) \neq C(P)$, for a subset $B \subset A$.

The angle formed by three points u, v and w is denoted $\text{ang}(u, v, w) \in [0, 2\pi)$ and the orientation depends on the context. If the orientation is not given, it is either clockwise or counterclockwise, but it does not vary for a given robot during a cycle. Then, $\text{ang}_{\min}(u, v, w) \in [0, \pi)$ denotes the minimum angle among both orientations. The angle formed by two half-lines h_1 and h_2 with same origin o , denoted $\text{ang}(h_1, h_2)$, is the angle $\text{ang}(u_1, o, u_2)$ where $u_1 \in h_1$ and $u_2 \in h_2$.

Let M be a set of points, $H_c(M)$ denotes the set of half lines starting at a point c that contains at least one robot in M . Let $p \in M$, $\alpha_{\min, c}(p, M)$ denotes the minimum, not null, angle $\text{ang}_{\min}(p, c, m)$ where $m \in M$. Then, $\alpha_{\min, c}(M)$ denotes the minimum $\alpha_{\min, c}(p, M)$ for all p in M (i.e., the minimum angle between two half lines in $H_c(M)$). If c is clear from the context (especially, if $c = c(M)$), it is omitted. Finally, we say that a robot r is *selected* if $r \in \overline{\mathcal{D}}(l_F/2)$ and r is the only robot in $\mathcal{D}(2|r|)$ (see Figure 1a).

Local View and Symmetricity. Let P be a configuration and $r \in P$. If $r \neq c(P)$, we define the local view Z_r of robot r as the set of positions of robots in P in the polar coordinate system centered at $c(P)$ where r is at coordinate $(1, 0)$ and with the orientation that maximizes the view (using the lexicographic order over the set of coordinates). Of course, if both orientations give the same view, r is on an axis of symmetry. Two robots are *equivalent* if they have the same view with the same orientation. It is known that all equivalency classes have the same cardinal, which is the *rotational symmetry factor* $\rho(P)$ (also called the symmetricity of P). Moreover, if $\rho(P) > 1$, each equivalency classes forms a $\rho(P)$ -gon centered at $c(P)$. However, If $\rho(P) = 1$, the configuration can have an axis of symmetry. In this case, for all $r \in P$, either r has a symmetric robot that has the same view with a different orientation, or r 's view is invariant with both orientations (r is on the axis of symmetry). Moreover, if there is one axis of symmetry, then there are $\rho(P)$ axes of symmetry, and the robots cannot agree on a common orientation.

Regular Set. We now give several definitions to characterize in a configuration what symmetry information can be retained from past configurations. In more details, starting from a symmetric configuration, we want to move a subset of robots while keeping the information that the initial configuration was symmetric. To do so, we give the definition of *the* m -regular set of a configuration. Every symmetric configuration contains a unique m -regular set and if a configuration contains a m -regular set and we know that only the robots in this subset have moved, then the previous configuration contains the same m -regular set. First, we give the conditions for a set to be m -regular (see Figures 1b and 1c).

Definition 1. A set of robots M , with $m = |M| \geq 2$, is m -regular (resp. $m/2$ -regular) if there is a point c and an angle α (resp. two angles α and β) such that $H_c(M) = \{h_0, \dots, h_{m-1}\}$ and for all $0 \leq i < m$, $\text{ang}(h_i, h_{i+1}) = \alpha$ (resp. $\text{ang}(h_i, h_{i+1}) = \alpha$ if i is even, β otherwise) were $h_m = h_0$. If M is $m/2$ -regular, we say it is bi-angled, and the lines that cut equally the angle formed by h_i and h_{i+1} are called the virtual axes of symmetry of M .

The center and the angles α and β of a m -regular set are unique and be computed in linear time [1]. Moreover, a m -regular set remains m -regular (with the same center) upon radial moves of the robots in M . Indeed, the center of a m -regular set is its Weber point [1], which is unique and invariant under straight line movement towards it. Regular sets were defined by Bouzid *et al.* [3], but we use here a simpler definition that only captures equiangular and biangular configurations.

We now define *the* regular set of a configuration, which represent intuitively the regular set that is closest to the center. In the following definition **it is crucial** to observe that, in order for a configuration to have a m -regular set, it is not sufficient to have a subset that is m -regular. Indeed, the entire configuration must satisfy additional requirements to be coherent with the m -regular subset.

Let P be a n -robot configuration (with $c(P) \notin P$). We consider the increasing sequence of subset of robots Q_1, \dots, Q_k ($k \leq n$) where Q_i contains the i robots in P with greatest local view that does not hold $C(P)$. Since the local views are uniquely defined and the fact whether a robot hold $C(P)$ is computed the same way by all robots, then the sequence $(Q_i)_{1 \leq i \leq k}$ is also uniquely defined. For each Q_i in this sequence, every robot can check (a) whether Q_i is m -regular with center $c(P)$. If so, it can check (b) whether m divides $\rho(P \setminus Q_i)$, and (c), in the case when Q_i is bi-angled, whether the virtual axis of symmetry of Q_i are axes of symmetry of $P \setminus Q_i$. Among all Q_i that satisfy those three properties, let Q_{\max} be the largest one (or the empty set if no such Q_i exists). Again, Q_{\max} is uniquely defined.

Definition 2 (The regular set $\text{reg}(P)$ of a configuration P). Let P be a n -robot configuration (with $c(P) \notin P$). If P is a m -regular set (with $m = n$, or $m = n/2$), then $\text{reg}(P) = P$. Otherwise, $\text{reg}(P) = Q_{\max}$. If $\text{reg}(P) = \emptyset$, we say that the configuration P does not contain a regular set.

Property 1. Let P be a n -robot configuration. If $\rho(P) > 1$, or if P contains an axis of symmetry, then P contains a regular set.

By construction, the regular set of a configuration is unique and, if $P \neq \text{reg}(P)$, its center is the center of $C(P)$. For our purpose, we need to be able to elect a robot in a regular set. This is possible by shifting a single robot *i.e.*, by moving it on its circle, so that it is the only robot that breaks the regularity. In this case we say the configuration contains a shifted regular set (see Figure 1d).

Definition 3. Let P be a n -robot configuration. Then, P contains a ε -shifted- m -regular set (with $0 < \varepsilon \leq 1/4$) if there exist $r \in P$ and $r' \in \mathbb{R}^2 \setminus P$ such that $P' = P - \{r\} \cup \{r'\}$ contains a m -regular set of center c satisfying: (a) $\text{ang}_{\min}(r, c, r') = \varepsilon \alpha_{\min}(P')$, (b) $\alpha_{\min}(r, P) < \alpha_{\min}(r', P')$, and (c) $|r| = |r'| = \min_{u \in P} |u|$. In this case, ε is the shift, r is the shifted robot, and $\text{reg}(P) = \text{reg}(P') - \{r'\} \cup \{r\}$ is the shifted regular set of P . The associated regular set of $\text{reg}(P)$ is $\text{reg}(P')$.

A shifted-regular set is obtained from a regular set Q by moving one of the robot that is closest to the center. This robot moves on its circle in the direction that decreases the minimum angle with the other

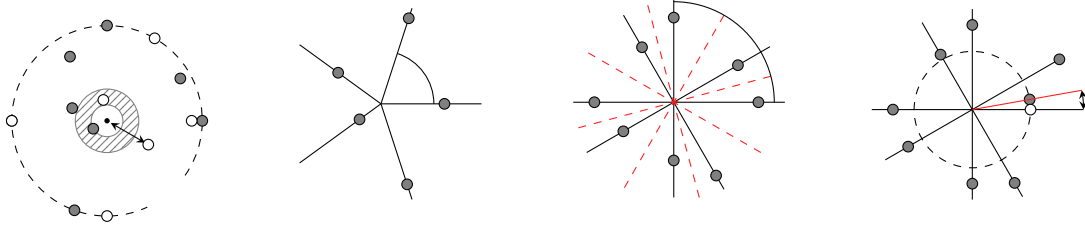
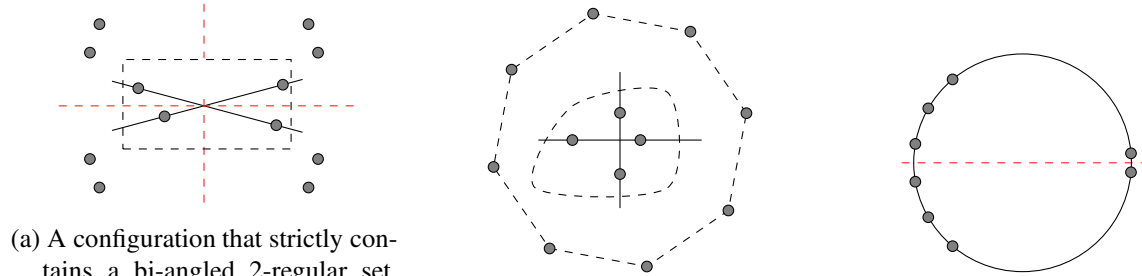


Figure 1: (a) A configuration of robots that contains a selected robot and the pattern to form (white nodes), (b) a 5-regular set, (c) a bi-angled 4-regular set and (d) a bi-angled shifted-4-regular set



(a) A configuration that strictly contains a bi-angled 2-regular set. $2 = \rho(P \setminus \text{reg}(P))$ and the virtual axis of $\text{reg}(P)$ are axis of symmetry of $P \setminus \text{reg}(P)$.

(b) A configuration that strictly contains a 4-regular set. We see that 4 divides $8 = \rho(P \setminus \text{reg}(P))$.

(c) r_1 and r'_1 have the greatest view but hold $C(P)$ so $\text{reg}(P) = \{r_2, r'_2\}$.

Figure 2: Examples of configurations that strictly contain a regular set.

robots. The shift ε is the factor between angle $\text{ang}(r, c(P), r')$ and the minimum angle between every two half-lines centered at $c(P)$. The robots can check if a the configuration contains a ε -shifted- m -regular set; it is unique if $|P| \geq 7$.

Theorem 1. *Let $n \geq 7$ and P be a n -robot configuration that contains a ε -shifted- m -regular set Q of center c . Then Q , c , m , ε and the shifted robot are unique.*

Algorithm overview The algorithm consists of several phases, described in pseudo-code in algorithm `formPattern`. Line 6 corresponds to randomized symmetry breaking algorithm ψ_{RSB} described in section 3. Lines 7 to 17 correspond to the deterministic pattern formation algorithm ψ_{DPF} described in section 4. Each function calls correspond to a phase that is executed if and only its phase condition is not verified. If the condition is verified, the next phase is considered. Each time a robot is activated, it must find the first phase with a condition that is not verified and follow the corresponding instructions. Each phase is done not to break the previous phase conditions. The condition line 3 is checked before because the movement line 4 breaks the condition of the other phases.

3 Randomized Symmetry Breaking Algorithm

In this section we describe Algorithm ψ_{RSB} . A n -robot configuration P is in the active set of ψ_{RSB} if (i) P does not have a selected robot, (ii) if does not have a unique robot with maximal view, or (iii) if the set P from which we remove the robot with maximum view forms the pattern F from which we remove one of the point with maximal view. The goal of ψ_{RSB} is, starting from any active configuration, to obtain a configuration with a selected robot (and thus is not in the active set of ψ_{RSB}). We partition the active set of ψ_{RSB} in two sets Q and Q^c , depending whether the configuration contains a regular set or not, and we study the two algorithms $\psi_{RSB|_Q}$ and $\psi_{RSB|_{Q^c}}$ separately. We show that both algorithms satisfy the termination awareness

Algorithm formPattern: main algorithm that forms a pattern F

```

1 ClosestF ← {f s.t. f has a maximal view and f does not hold C(F)}
2 ClosestP ← {r s.t. r has a maximal view}
3 if ClosestP = {r} and ∃f ∈ ClosestF s.t. P - {r} ≈ F - {f} then
4   | r moves toward f
5 else
6   | r_s ← selectARobot()
7   | Z ← createGlobalCoordinateSystem(r_s)
8   | P' ← P - {r_s}
9   | F' ← F - {f_max} (where f_max is a robot with a maximal view)
10  | Let C_1, C_2, ..., C_m be the m circles centered at c(P) with decreasing radius, each containing at least one point in F'. For each
11  | 1 ≤ i ≤ m, let m_i = |C_i ∩ F'| > 0
12  | if |C_i ∩ F'| = 2 then
13  |   | fixEnclosingCircle()
14  |   for i = 1, 2, ..., m do
15  |     | cleanExterior(i)
16  |     | locateEnoughRobots(i)
17  |     | removeRobotsInExcess(i)
18  |   rotateRobotOnCircle()

```

property, and terminate with probability one. Also, $\psi_{RSB|_{Q^c}} \rightsquigarrow \psi_{RSB|_Q}$, so that $\psi_{RSB} = \{\psi_{RSB|_Q}, \psi_{RSB|_{Q^c}}\}$ is a partially ordered combination of algorithm. The pseudo-code of this phase is given in the procedure **selectARobot** called line 6 of the main algorithm. The phase is ignored if the *phase condition* is verified.

3.1 The current configuration P is in Q .

In this subsection, Q denotes the m -regular set (shifted or not) of P . Also, all movements ordered by $\psi_{RSB|_Q}$ are done so that any configuration that is immediately reachable from P also contains a regular set (possibly shifted) with the same robot and the same center as Q (see property 2 in the appendix). This feature is important to ensure that, when a robot becomes selected, all robots in P are static so that $\psi_{RSB|_Q}$ satisfies the termination awareness property.

If the regular set is not shifted, the goal is to elect a robot among robots in Q . Once a robot is elected (see below how the probabilistic election process takes place), it moves on its circle to create a shift $\varepsilon = 1/8$. If the regular set is shifted with a shift $1/8 < \varepsilon < 1/4$ such that the other robots in Q are not on the same circle as the shifted robot, then the shifted robot moves on its circle to create a shift $\varepsilon = 1/8$. The shifted robot may be activated multiple times before reaching a configuration with a $1/8$ -shifted regular set. If another robot in Q is activated during the movement of the shifted robot (*i.e.*, when the shift is not exactly $1/8$), it chooses not to move.

If the configuration P contains a shifted regular set with a shift $\varepsilon = 1/8$, the shifted robot does not move and the other robot in the shifted regular set move radially to reach the circle of the shifted robot. When it is done, and when the shifted robot is activated, it knows that the other robots have reached their destination and are now static. Thus it can start its last movement to become selected. It first moves on its circle so that the shift is $\varepsilon = 1/4$, and then moves radially toward $c(P)$ until it becomes selected. During this last movement, all the other robots are static so that when the shifted robot become selected, the whole configuration is static. So the algorithm $\psi_{RSB|_Q}$ satisfies the termination awareness property. The probabilistic, and most challenging, part of the algorithm $\psi_{RSB|_Q}$ is therefore the robot election.

Robot Election. If the current configuration P contains a regular set Q , the goal is to elect a robot. A robot r_e is *elected* when $|r_e| < \frac{7}{8} \min_{r \in Q - \{r_e\}} |r|$ where distances are taken from $c(P)$. A robot is aware that it is elected if it is elected during its look phase. When a robot is aware it is elected, it moves on its circle to create a $1/8$ -shifted-regular set. To elect a robot, each robot r in Q proceeds in the following way. If there is another robot in Q that is strictly closer to the center, then r does not move. If r is not elected and

Algorithm selectARobot: select a robot

Phase Condition: There exists a selected robot r_s
Returned Value: r_s

```

if  $P$  contains a  $\varepsilon$ -shifted regular set  $Q$  then
   $r_e \leftarrow$  the shifted robot
   $S \leftarrow \{r \in P \mid |r| > |r_e|\}$ 
  if  $S \neq \emptyset$  and  $\varepsilon \neq 1/8$  then
     $r_e$  moves on its circle to create a  $1/8$ -shifted regular set
  else if  $S \neq \emptyset$  and  $\varepsilon = 1/8$  then
    for  $r \in S$  do
       $r$  moves radially at distance  $|r|$  from  $c(P)$ 
  else if  $\varepsilon < 1/4$  then
     $r_e$  moves on its circle to create a  $1/4$ -shifted regular set
  else
     $r_e$  moves radially toward  $c(P)$  to become selected
else if  $P$  contains a regular set  $Q$  then
  if  $P \setminus Q = \emptyset$  then
     $d \leftarrow \min_{r' \in P \setminus Q} |r'|$ 
  else
     $d \leftarrow \infty$ 
  handlePartiallyFormedPattern()
  for  $r \in P$  do
    if  $|r| < \frac{7}{8} \min_{r \in Q - \{r\}} |r|$  then
       $r$  moves on its circle to create a  $1/8$ -shifted regular set
    else if  $\{r' \neq r \text{ s.t. } |r'| < |r|\} = \emptyset$  then
       $c \leftarrow 1$  with probability  $1/2$ ,  $0$  otherwise
      if  $c$  then
         $r$  moves a distance  $|r|/8$  toward  $c(P)$ 
      else
         $r$  moves a distance  $\min(\frac{1}{2}(d - |r|), \frac{1}{7}|r|)$  away from  $c(P)$ 
  else
     $r_{max} \leftarrow$  unique robot with maximum view that does not hold  $C(P)$ 
    if  $\exists r \in [r_{max}, c(P)]$ ,  $P \cup \{r\} - \{r_{max}\}$  is regular then
       $r_{max}$  moves toward  $r$ 
    else
       $r_{max}$  moves toward  $c(P)$ 
  
```

is one of the closest robot (unique or not), then r chooses randomly (each choice with probability $1/2$) to go toward or away from the center $c(P)$. If r chooses to move toward the center, it moves a distance $|r|/8$. If r chooses to move away from the center, it moves a (possibly null) distance $\min(\frac{1}{2}(d - |r|), \frac{1}{7}|r|)$, where d is the minimum distance to the center among robots in $P \setminus Q$ (and $d = \infty$ if $P \setminus Q = \emptyset$). This ensures that robots in Q does not move outside D_{\max} defined in Property 2, so that it is a (M1) movement. We observe that during that phase, no robot becomes *selected*. Indeed if r is the closest robot to the center and $l_Q = \min_{r' \in Q - \{r\}} |r'|$, then $l_Q \geq |r| \geq \frac{3}{4}l_Q > \frac{1}{2}l_Q$, so that other robots are in $\mathcal{D}(2|r|)$. Also, to ensure that $n - 1$ robots cannot form part of the pattern, a special case is handled by calling the function `handlePartiallyFormedPattern()` (which is discribed in the appendix). So that the configuration cannot leave the active set of $\psi_{RSB|_Q}$ without being aware of it. The following properties hold: (i) eventually one robot is aware it is elected with probability one, and (ii) once a robot is aware it is elected, another robot cannot be elected.

Lemma 1. *The following properties hold: (i) eventually one robot is aware it is elected with probability one, and (ii) once a robot is aware it is elected, another robot cannot be elected.*

Proof. i) Suppose first that robots always reach their destination. Initially the configuration is in a state where there are at least two robots whose location (or destination for those that are moving) are the closest to the center. Let r be the first (or one of the first) activated robot among them. With probability $1/2$, it chooses to move toward the center. Let r' be another robot among them. If r' is activated after r begins

its movement, r' does not move, otherwise it moves away from the center with probability $1/2$. So with probability greater than $1/2^n$, r is the only robot to move toward the center. After that, if another robot is activated before the next activation of r , it does not move. At the next activation of r , with probability $1/2$, r chooses to move toward the center and becomes elected (and r is aware it is elected when it is next activated since the other robots are static). If this does not happen, *i.e.*, if the first or the second choice of r is to move away the center or if another robot chose to move toward the center, then the configuration gets back to its initial state. So, we have infinitely often a probability greater than $1/2^{n+1}$ to have an elected robot. So, eventually there is a single elected robot with probability one.

If robots does not always reach their destinations, the probability that r moves by a distance d toward the center (to become elected), is $1/2^{\lceil \frac{d}{\delta} \rceil}$, instead of $1/4$ in the first case. Indeed, in the first case r needs to chose two times to move toward the center, and here r needs to choose $\lceil \frac{d}{\delta} \rceil$ times to move toward the center. The probability that one of the other robot chooses to move away (and stay still while it is not one of the closest robot to the center) is still $1/2$. So that there is again a non null probability that a robot is elected, which implies that a robot is eventually elected with probability one.

ii) Once a robot r_e is elected, the other robots are currently either moving away, not moving, or moving toward the center by a distance at most the eighth of their distance to the center. In each case, when another robot looks again (after it finishes its movement) it sees that the robot r_e is the only closest robot to the center, and then it chooses not to move. Indeed, if another robot r is moving toward the center, it is by a distance at most $|r|/8$. Since we have $|r_e| < \frac{7}{8}|r|$ when r started its movement, we have $|r_e| < |r|$ after r finishes it. After r 's movement, r_e is maybe no longer elected, but since r_e was aware it was elected, it already chooses to move on its circle to create a $1/8$ -shifted-regular set (and it moves by a non null distance, so that in the next look phase, it is shifted). \square

We shown that if P is in \mathcal{Q} , then eventually a robot become selected with probability one. As soon as the elected robot is selected, the configuration is static. Finally, until the elected robot is selected, the configuration remains in \mathcal{Q} and cannot reach a configuration in \mathcal{Q}^c . Thus we have the following lemma:

Lemma 2. $\psi_{RSB|_{\mathcal{Q}}}$ terminates in finite time with probability one, $\psi_{RSB|_{\mathcal{Q}}} \rightsquigarrow \psi_{DPF}$, $\psi_{RSB|_{\mathcal{Q}^c}} \Leftarrow \psi_{RSB|_{\mathcal{Q}}}$, and $\psi_{RSB|_{\mathcal{Q}}}$ satisfies the termination awareness property.

3.2 The current configuration P is in \mathcal{Q}^c .

If $P \in \mathcal{Q}^c$, then, by Property 1, $\rho(P) = 1$ and P does not have an axis of symmetry. So, all robots have distinct views and there exists a unique robot r_{\max} with maximum view, among the robots that do not hold $C(P)$. If the robot with maximum view holds $C(P)$, then all robots are on $C(P)$. Since the algorithm $\psi_{RSB|_{\mathcal{Q}^c}}$ only orders r_{\max} to move toward the center $c(P)$, as soon as r_{\max} moves it becomes the unique robot with maximum view, and remains the only robot ordered to move by $\psi_{RSB|_{\mathcal{Q}^c}}$.

First, $\psi_{RSB|_{\mathcal{Q}^c}}$ checks if there is a position in the segment $[r_{\max}, c(P))$ such that the whole configuration is a regular set (shifted or not). If it is the case, the first such point in the path becomes the destination of r_{\max} . Otherwise, r_{\max} is ordered to move toward the center until it becomes selected. The resulting configuration cannot strictly contains a regular set since P does not contain one and r_{\max} performs a radial movement. Starting from a static initial configuration, once r_{\min} is selected, or once the configuration is regular, all the robots are static, so we have the following lemma:

Lemma 3. $\psi_{RSB|_{\mathcal{Q}^c}}$ terminates in finite time, $\psi_{RSB|_{\mathcal{Q}^c}} \rightsquigarrow \psi_{DPF}$, $\psi_{RSB|_{\mathcal{Q}^c}} \rightsquigarrow \psi_{RSB|_{\mathcal{Q}}}$, and $\psi_{RSB|_{\mathcal{Q}^c}}$ satisfies the termination awareness property.

4 Deterministic Pattern Formation without Chirality

We now present a new deterministic pattern formation algorithm ψ_{DPF} that forms an arbitrary pattern F from an initial configuration that contains a selected robot. The initial n -robot configuration P must either contain a selected robot r_s or have a unique robot with maximum view and the other robots form the pattern F from which we remove a robot with maximal view. In order to be part of the main pattern formation algorithm Ψ , the algorithm ψ_{DPF} ensures that this initial condition remains true during the entire execution. When the set $P - \{r_s\}$ match the pattern, then r_s moves to its final destination (line 4). The destination of r_s is a point f_s in F that has maximal view, among points that do not hold $C(F)$ (which always exists if $|F| \geq 4$). The selected robot remains the unique robot with maximum view during this last move. So first, the robots in $P' = P - \{r_s\}$ have to form the pattern $F' = F - \{f_s\}$. Even if f'_s is another point with maximal view (that does not hold $C(F)$), $F - \{f_s\} \approx F - \{f'_s\}$ so that the choice of the point with maximal view is not important.

The algorithm does not assume that the robots are aware of a global orientation (chirality). For that, the algorithm ensures that the configuration is never axially symmetric, except maybe in the initial configuration and in the terminal configuration. Also, since there is a selected robot, the symmetricity is one. So, all robots can agree on a common global oriented coordinate system at any time of the algorithm. The algorithm ψ_{DPF} consists in three phases. The first phase (line 7) creates the global oriented coordinate system. The second phase (line 14, 15 and 16) moves robots so that they are at the right distance to the center $c(P)$. The last phase (line 17) moves robots on their circle to create the pattern.

Algorithm createGlobalCoordinateSystem: create the global coordinate system

Phase Condition: There exists a robot r_{\max} such that:

- i) $|r_{\max}| = \min_{r \in P - \{r_s\}} |r|$
- ii) $ang_{\min}(r_s, c(P), r_{\max}) = \min_{r \in P - \{r_s\}} ang_{\min}(r_s, c(P), r)$
- iii) $|r_{\max}| \leq |f_{\max}|$
- iv) $2ang_{\min}(r_s, c(P), r_{\max}) < \theta_{F'}$ with

$$\theta_{F'} = \min(\{\pi\} \cup \{ang_{\min}(f_{\max}, c(F), f) \mid f \in F' - \{f_{\max}\}, |f| = |f_{\max}|\})$$

(where f_{\max} is any robot in F' that maximizes the view)

Returned Value: the polar coordinate system of center $c(P)$, vector $\overrightarrow{c(P)r_{\max}}$, and the orientation that maximizes the coordinates of r_s

if $r_s = c(P)$ **then**

$r_{\max} \leftarrow \min_{r \in P - \{r_s\}} |r|$
 $d \leftarrow \min(|f|, \min_{r \in P'} |r|)/2$
 r_s moves at distance d from $c(P)$ such that i), ii) and iv) are true

else

if $\exists r_{\max}$ such that i), ii) and iv) are true **then**
 r_{\max} moves toward $c(P)$ at distance $|f_{\max}|$ from $c(P)$
else
 r_s moves toward $c(P)$

Phase 1. In this phase (described in algorithm **createGlobalCoordinateSystem**), only two robots move in order to create the global coordinate system used in the remainder of Algorithm ψ_{DPF} . We need a unique robot r_{\max} in $P - \{r_s\}$ that satisfies the following conditions (distances are taken from $c(P) = c(F)$): i) $|r_{\max}| = \min_{r \in P - \{r_s\}} |r|$; ii) $ang_{\min}(r_s, c(P), r_{\max}) = \min_{r \in P - \{r_s\}} ang_{\min}(r_s, c(P), r)$; iii) $2ang_{\min}(r_s, c(P), r_{\max}) < \theta_{F'}$ with

$$\theta_{F'} = \min(\{\pi\} \cup \{ang_{\min}(f_{\max}, c(F), f) \mid f \in F' - \{f_{\max}\}, |f| = |f_{\max}|\})$$

(where f_{\max} is any robot in F' that maximizes the view). In particular, r_{\max} maximizes the view among robots in $P - \{r_s\}$. If r_{\max} does not exist initially, the selected robot r_s moves to the center $c(P)$ and then moves to create a unique r_{\max} . Indeed, once r_s is at $c(P)$, it can choose any robot $r \in P - \{r_s\}$ that is the closest to $c(P)$ and moves a little in a direction so that the angle $ang_{\min}(r_s, c(P), r)$ is small enough for r to satisfy the last two conditions. Finally, if $|f_{\max}| < |r_{\max}|$, then r_{\max} moves radially at the same distance from

$c(P)$ as f_{\max} . The phase ends when r_{\max} exists and is at most at distance $|f_{\max}|$ from the center $c(P)$. Note that r_s can still be in movement, but the definition of the global coordinate system does not depend on the distance $|r_s|$.

The existence of r_{\max} $\xrightarrow{\text{implies}}$ that we can define a global polar coordinate system $Z : \mathcal{R} \rightarrow \mathbb{R}^+ \times [0, 2\pi)$, with center $c(P)$, vector $c(P)r_{\max}$, and the orientation that maximizes the coordinates of r_s . If r_s , r_{\max} , and $C(P)$ do not change, then Z remains invariant. We already have $C(F') = C(P)$ and from now, F' is mirrored so that the orientation of a point with maximal view f_{\max} coincides with the orientation of Z , and rotated so that the points f_{\max} and r_{\max} are on the same half-line of origin $c(P)$. Again the choice of f_{\max} and the choice of the orientation (if both orientations of f_{\max} have a maximal view) are not important since the resulting coordinates of points in F' in Z are identical. So, from now on, robots see the points in F' in the same way in the global coordinate system.

Phase 2. Thanks to this global coordinates system Z , we can strictly order the robots of $P' = P - \{r_s\}$, $r_{\max} = r_{n-1} < r_{n-2} < \dots < r_1$ using the lexicographic order on their polar coordinates. We keep this ordering unchanged during the remainder of the algorithm.

This phase consists in moving robots so that there is the right number of robots on each circle centered at $c(P)$. Let C_1, C_2, \dots, C_m be the m circles centered at $c(P)$ with decreasing radius, each containing at least one point in F' . For each $1 \leq i \leq m$, let $m_i = |C_i \cap F'| > 0$. We have $\sum_{i=1}^m m_i = |F'| = n - 1$.

Before beginning this phase we move the robots that have a null angle (except r_{\max}) on their circle in the direct orientation while preserving the order (*i.e.*, without reaching another robot), so that no robot has a null angle (except r_{\max}). This is required for proper operation of action *ii*), defined below. Also, if $m_1 = 2$, since two robots cannot move on $C(P)$ synchronously to hold $C(P)$, we need to execute a pre-phase to ensure that the two robots are located at the two points of $C(P) \cap F$, keeping $C(P)$ unchanged. Informally, this pre-phase moves a robot on $C(P)$ if there are only two robot on it, then the two greatest robot reach their destination point in $F \cap C(P)$. Then the other robots can leave safely $C(P)$. So now we suppose that if $m_1 = 2$, then C_1 already contains two robots located at their corresponding point in F .

Recursively we move robots such that each circle C_i contains exactly m_i robots. We define the following procedure for a given i , $1 \leq i \leq m$. The procedure executes four actions sequentially and assumes, if $i > 1$, that $|interior(C_{i-1}) \cap P'| = \sum_{j=i}^m m_j$.

cleanExterior(i) If $i > 1$ and $|interior(C_{i-1}) \cap exterior(C_i) \cap P'| > 0$, then the smallest robot in $exterior(C_i)$ moves to C_i while it remains greater than robots already in C_i . To do so it can move a little toward $c(P)$, so that there is no other robot in its circle, then moves on its circle so that its angle is greater than the angles of robots in C_i , and finally moves radially toward $c(P)$ to reach C_i . If $i = m$, we also ensure that its angle is less than $2\pi - \text{ang}(r_s, c(P), r_{\max})$, so that the second property of r_{\max} defined in phase 1 remains true. We repeat this process until there are no more robots between C_{i-1} and C_i .

Algorithm cleanExterior(i): remove robots outside C_i

Phase Condition: $i = 1$ or $|interior(C_{i-1} \cap exterior(C_i) \cap P'| = 0$
 $r \leftarrow$ smallest robot in $exterior(C_i)$
 $C \leftarrow$ circle centered at $c(P)$ that contains r
if $|C \cap P'| > 1$ **then**
 | r moves toward $c(P)$ without reaching the circle of another robot nor C_i
else
 | $a \leftarrow \max_{r' \in C_i} \text{ang}(r_{\max}, c(P), r')$ **if** $\text{ang}(r_{\max}, c(P), r) > a$ **then**
 | r moves toward $c(P)$ to reach C_i
 | **else**
 | r moves on C_i in the direct orientation to have an angle $(2\pi + a)/2$

locateEnoughRobots(i) If $|C_i \cap P'| < m_i$, then we have $|interior(C_i) \cap P'| \geq 1$. Indeed, if $i = 1$, $|interior(C_1) \cap P'| = m_1 - |C_1 \cap P'| \geq 1$, otherwise, there are by hypothesis at least m_i robots inside C_{i-1} , and after performing

action i), these robots are not between C_i and C_{i-1} . We order the greatest robot in $interior(C_i)$ to move to C_i while remaining smaller than robots already in C_i . To do so, it can move a little away from $c(P)$ so that there is no other robot in its circle, then moves on its circle so that its angle is smaller than the angles of robots in C_i (but not null) and finally moves radially away from $c(P)$ to reach C_i . We can repeat this process until there are exactly m_i robots on C_i .

Algorithm locateEnoughRobots(i): locate enough robots on C_i

Phase Condition: $|C_i \cap P'| \geq m_i$
 $r \leftarrow$ greatest robot in $interior(C_i)$
 $C \leftarrow$ circle centered at $c(P)$ that contains r
if $|C \cap P'| > 1$ **then**
 r moves away from toward $c(P)$ without reaching the circle of another robot nor C_i
else
 $a \leftarrow \min_{r' \in C_i} \text{ang}(r_{\max}, c(P), r')$
 if $\text{ang}(r_{\max}, c(P), r) < a$ **then**
 r moves away from $c(P)$ to reach C_i
 else
 r moves on C_i in the indirect orientation to have an angle $a/2$

removeRobotsInExcess(i) If $i > 1$ and $|C_i \cap P'| > m_i$, then the smallest robot in C_i moves a little toward $c(P)$ (here, “a little” means a small distance such that the order is preserved *i.e.*, the robot does not reach the circle of another robot nor C_{i+1}). We repeat this process until there are exactly m_i robots on C_i .

If $i = 1$ and $|C_1 \cap P'| > m_1$, then we cannot do the exact same thing because we have to ensure that $C(P)$ does not change. However, we know that $m_1 \geq 3$. The m_1 greatest robots r_1, \dots, r_{m_1} remain on C_1 , and have to be the only robots to hold $C(P)$. To do so, the angles formed by two consecutive robots in $\{r_1, \dots, r_{m_1}\}$ have to be smaller than or equals to π . This is obtained by moving the robots on C_1 , while preserving the ordering and $C(P)$, such that r_1, \dots, r_{m_1} form the regular m_1 -gon that have the line $c(P)r_{\max}$ as an axis of symmetry. At the same time, if the m_1 -gon is not formed yet, other robots in C_1 move on C_1 to be evenly distributed in the arc between angle 0 and π/m_1 (the blue arc in Figure 4), again while preserving the ordering and $C(P)$. Overall, each robot on C_1 has a deterministic (and non-blocking) destination. Once the m_1 -gon is formed (even if some other robots are still moving), the smallest robot in $C(P) \cap P$ moves a little toward $c(P)$. This is repeated until only r_1, \dots, r_{m_1} remain on C_1 .

Algorithm removeRobotsInExcess(i): remove robot in excess on C_i

Phase Condition: $|C_i \cap P'| = m_i$
// Where $\text{Poly}(a, b)$ denotes the set of vertice of the regular a -gon centered at $c(P)$ that have the line $c(P)r_{\max}$ as axis of symmetry union b points evenly distributed in the arc between angle 0 and π/a
if $i > 1$ **then**
 $r \leftarrow$ smallest robot on C_i
 r moves toward $c(P)$ without reaching the circle of another robot
else
 if robots the m_1 greatest robots on C_1 forms $\text{Poly}(m_1, 0)$ **then**
 $r \leftarrow$ smallest robot on C_1
 r moves toward $c(P)$ without reaching the circle of another robot
 else
 robots on C_1 form $\text{Poly}(m_1, |P \cap C_1| - m_1)$

After executing the above procedure for a given i , we have m_i robots on C_i and $|interior(C_i) \cap P'| = |interior(C_{i-1}) \cap P'| - m_i = \sum_{j=i+1}^m m_j$, so that we can execute the same procedure with $i + 1$. If $i < m$ it is important to observe that some robots (those ordered to move in the last two cases) may still be in movement, but since they are now strictly between C_i and C_{i+1} , they receive new order with deterministic destinations when executing the procedure with $i + 1$. Hence, at the end of the procedure with $i = m$, all robots are static. Also, if $i = m$, r_{\max} may be ordered to move radially to f_{\max} , but if it is the case, only the distances of Z

change, so the ordering is preserved.

After executing those actions for $i = 1, 2, \dots, m$, each circle contains the right number of robots.

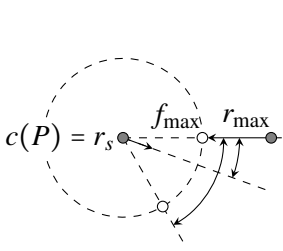


Figure 3.a: Step 1: r_s create a unique robot r_{\max} ($\theta = \theta_{F'}$), then r_{\max} move to reach the circle of f_{\max} .

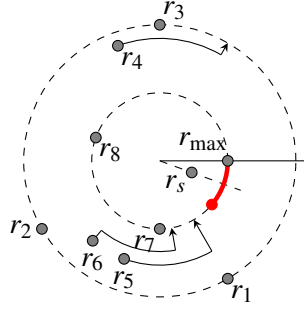


Figure 3.b: Resolution of the case i) and ii)

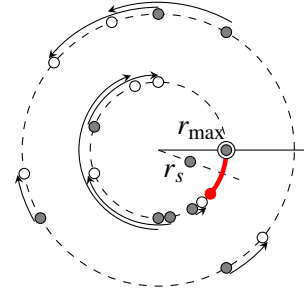


Figure 3.c: Second phase of the pattern formation algorithm

Phase 3. Let $i \in [1, m]$, C_i now contains m_i robots and m_i destinations for those robots. The robots and the destinations are ordered so that each robot is aware of its corresponding destination. They can all move toward their destination, while remaining on C_i and preserving the robots ordering (*i.e.*, without reaching another robot position). When a robot r is active and another robot is on the way, r choose on the circle half the distance to this robot. There cannot be a deadlock since there is no cycle in the waiting relation. Indeed, robots on C_i are ordered by angle so that they behave like they are on a finite segment of length 2π .

If $i = 1$, during their movement, robots also ensure that $C(P)$ remains unchanged. To do so, if a robot $r \in C_1$ is active and detects that its movement can modify $C(P)$, then it moves the most it can without changing $C(P)$.

At the end of the phase, robots in $P - \{r_s\}$ form F' and r_s can move directly to f_s . There is no robot on the path to f_s because f_s is one of the closest point to the center. Also, $C(P)$ does not change so that the robots can easily check whether they are in this last step. After this movement, the robots form F .

Lemma 4. ψ_{RSB} terminates in finite time, $\psi_{RSB} \Leftarrow \psi_{DPF}$ and ψ_{DPF} satisfies the termination awareness property.

From Lemmas 2, 3, and 4, we can state our main result:

Theorem 2. The combination of algorithms $\{\psi_{RSB}, \psi_{DPF}\}$ forms any pattern F from any initial configuration I (that do not contain multiplicity points) with probability 1, provided $n \geq 7$.

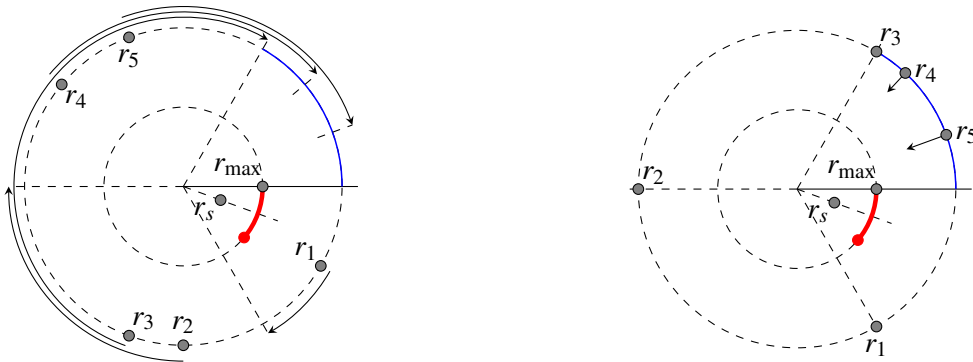


Figure 4: Resolution of the case iv) (with $m_1 = 3$). When the 3 greatest robots form the regular 3-gon, r_5 moves toward $c(P)$. Then r_4 moves a little toward $c(P)$.

Algorithm rotateRobotOnCircle: move the robots on their circle to reach their final destination

Phase Condition: $F' = P'$

Let r_1, \dots, r_{n-1} be the robots in P' in the lexicographic order of their polar coordinates in the global coordinate system.

Let d_1, \dots, d_{n-1} be the point of F' in the lexicographical order of their polar coordinates in the global coordinate system.

for $i = 1, \dots, n - 1$ **do**

$A \leftarrow$ the arc of the circle of r_i delimited by r_i and d_i that does not contains the point of angle 0

if $A \cap P' \neq \emptyset$ **then**

$c \leftarrow$ closest robot in $A \cap P'$

$d \leftarrow$ point of A in the middle r_i and c

$A \leftarrow$ the arc of the circle of r_i delimited by r_i and d

if $r_i \in C(P)$ **and then**

$d \leftarrow$ the farthest point on A so that $C(P)$ does not change

r_i moves on A toward d

5 Extensions and Perspectives

Similarly to previous work [13], neither the initial configuration nor the target pattern should contain multiplicity point for our approach to work. We show that if robots are endowed with multiplicity detection, a straightforward extension of our algorithm permits to form any general pattern that can contains points of multiplicity. However, the case of handling initial configurations with points of multiplicity requires algorithmic tools that are yet to be developed.

The case of patterns that contain points of multiplicity. If we assume that robots are endowed with multiplicity detection, the ordering of robots in the last phase of algorithm ψ_{DPF} remains unchanged except for robots located at points of multiplicity. Indeed, we can still compute the number of robots greater than a robot r , not located at a point of multiplicity, which gives its unique index. However, robots at points of multiplicity are incomparable. If we only allow robots that have the same destination point to form points of multiplicity, then the pattern can be formed by the same algorithm. The only case that need a small change in the algorithm is when $c(F)$ is a point of multiplicity. In this case we first form the pattern \tilde{F} where the destinations at $c(F)$ are replaced by the middle g_f between $c(F)$ and f_{\max} . Once \tilde{F} is formed, the robots at g_f move to $c(F)$. More details can be found in Appendix C.

The case of initial configurations that contain points of multiplicity. In the case where the initial configuration contains points of multiplicity, a convenient solution would be to reuse known pattern formation algorithms (such as ours) and run a preliminary phase where multiplicity points are eliminated. This task is known as the scattering task in the literature [4]. However, even the most recent developments [4] only considers the SSYNC model. Of course, as our protocol (and its multiplicity extension) also performs correctly in SSYNC, it is possible to combine the two to obtain a protocol in SSYNC that manages multiplicities both in I and in F : indeed, it is easy to combine protocols in SSYNC because moves are always aware of the latest configuration, so for all configurations that have multiplicities that do not belong to a legitimate path toward the target pattern, the scattering phase is run, until robots either reach a configuration where there is no point of multiplicity or a configuration that makes progress toward the target pattern. Extending this scheme to the ASYNC model requires to solve the open problem of ASYNC scattering, and making sure the combinations of protocols is feasible, which is a challenging path for future research.

References

- [1] Luzi Anderegg, Mark Cieliebak, and Giuseppe Prencipe. The weber point can be found in linear time for points in biangular configuration. Technical report, 2003.
- [2] Zohir Bouzid, Shlomi Dolev, Maria Potop-Butucaru, and Sébastien Tixeuil. Robocast: Asynchronous communication in robot networks. In Chenyang Lu, Toshimitsu Masuzawa, and Mohamed Mosbah,

- editors, *Principles of Distributed Systems - 14th International Conference, OPODIS 2010, Tozeur, Tunisia, December 14-17, 2010. Proceedings*, volume 6490 of *Lecture Notes in Computer Science*, pages 16–31. Springer, 2010.
- [3] Zohir Bouzid and Anissa Lamani. Robot networks with homonyms: The case of patterns formation. In Xavier Défago, Franck Petit, and Vincent Villain, editors, *Stabilization, Safety, and Security of Distributed Systems*, volume 6976 of *Lecture Notes in Computer Science*, pages 92–107. Springer Berlin Heidelberg, 2011.
- [4] Quentin Bramas and Sébastien Tixeuil. The random bit complexity of mobile robots cattering. In *AD HOC NOW*, *Lecture Notes in Computer Science*, page to appear, Athens, Greece, July 2015. Springer Berlin Heidelberg.
- [5] Shantanu Das, Paola Flocchini, Nicola Santoro, and Masafumi Yamashita. On the computational power of oblivious robots: Forming a series of geometric patterns. In *Proceedings of the 29th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, PODC '10, pages 267–276, New York, NY, USA, 2010. ACM.
- [6] Yoann Dieudonné, Franck Petit, and Vincent Villain. Leader election problem versus pattern formation problem. In Nancy A. Lynch and Alexander A. Shvartsman, editors, *Distributed Computing, 24th International Symposium, DISC 2010, Cambridge, MA, USA, September 13-15, 2010. Proceedings*, volume 6343 of *Lecture Notes in Computer Science*, pages 267–281. Springer, 2010.
- [7] Paola Flocchini, Giuseppe Prencipe, Nicola Santoro, and Peter Widmayer. Arbitrary pattern formation by asynchronous, anonymous, oblivious robots. *Theor. Comput. Sci.*, 407(1-3):412–447, 2008.
- [8] Nao Fujinaga, Hirotaka Ono, Shuji Kijima, and Masafumi Yamashita. Pattern formation through optimum matching by oblivious corda robots. In Chenyang Lu, Toshimitsu Masuzawa, and Mohamed Mosbah, editors, *Principles of Distributed Systems*, volume 6490 of *Lecture Notes in Computer Science*, pages 1–15. Springer Berlin Heidelberg, 2010.
- [9] Nao Fujinaga, Yukiko Yamauchi, Shuji Kijima, and Masafumi Yamashita. Asynchronous pattern formation by anonymous oblivious mobile robots. In MarcosK. Aguilera, editor, *Distributed Computing*, volume 7611 of *Lecture Notes in Computer Science*, pages 312–325. Springer Berlin Heidelberg, 2012.
- [10] Ichiro Suzuki and Masafumi Yamashita. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM J. Comput.*, 28(4):1347–1363, 1999.
- [11] Masafumi Yamashita and Ichiro Suzuki. Characterizing geometric patterns formable by oblivious anonymous mobile robots. *Theor. Comput. Sci.*, 411(26-28):2433–2453, 2010.
- [12] Yukiko Yamauchi and Masafumi Yamashita. Pattern formation by mobile robots with limited visibility. In Thomas Moscibroda and Adele A. Rescigno, editors, *Structural Information and Communication Complexity - 20th International Colloquium, SIROCCO 2013, Ischia, Italy, July 1-3, 2013, Revised Selected Papers*, volume 8179 of *Lecture Notes in Computer Science*, pages 201–212. Springer, 2013.
- [13] Yukiko Yamauchi and Masafumi Yamashita. Randomized pattern formation algorithm for asynchronous oblivious mobile robots. In Fabian Kuhn, editor, *Distributed Computing*, volume 8784 of *Lecture Notes in Computer Science*, pages 137–151. Springer Berlin Heidelberg, 2014.

A Robot Election Pre-phase

Algorithm handlePartiallyFormedPattern: executed before the robot election to handle configuration that can create a configuration that verifies in line 3 of the main algorithm

```

if  $\{F_r \cap F_r^c\}$  is a partition of  $F$  such that  $F_r^c \approx P \setminus Q$  and
 $|Q| - 1$  robots in  $Q$  are located on an halfline  $[c(P), f)$ , with  $f \in F_r$  then
   $d_1 \leftarrow$  radius of  $C(F_r)$ 
   $d_2 \leftarrow \min\{d \mid \mathcal{D}(d_1) \cap \text{exterior}(\mathcal{D}(d)) \cap F_r = \emptyset\}$ 
   $d \leftarrow (d_1 + d_2)/2$ 
  if  $\exists r \in Q$  s.t.  $|r| > d$  then
    if  $\exists r \in Q$  s.t.  $|r| > d_1$  then
      for  $r \in Q$  s.t.  $|r| > d_1$  do
         $r$  moves radially at distance  $|d_1|$  from  $c(P)$ 
      else
        for  $r \in Q$  s.t.  $|r| > d$  do
           $r$  moves radially at distance  $|d|$  from  $c(P)$ 
    exit

```

This pre-phase is executed before the robot election, *i.e.*, when the current configuration P contains a regular set Q , with $m = |Q|$. Before executing the robot election algorithm, a robot checks if the current configuration satisfies the following conditions:

- i) the pattern can be rotated so that robots in $P \setminus Q$ are located at points in F ,
- ii) among the m remaining points of F , denoted F_r , at least $m - 1$ are on $m - 1$ half lines, each containing exactly one robot in Q .

If those conditions are not both satisfied, the robot election is performed as previously described. Otherwise three cases can happen. Let d_1 be the radius of the smallest circle enclosing F_r . If $\mathcal{D}(d_1) \cap F_r \neq \emptyset$, let d_2 be the smallest radius such that $\mathcal{D}(d_1) \cap \text{exterior}(\mathcal{D}(d_2)) \cap F_r = \emptyset$, otherwise let $d_2 = d_1$. Also, let $d = (d_1 + d_2)/2$.

In the first case, at least one robot r satisfies $|r| > d_1$. Then, all such robots move toward $c(P)$ to reach the circle of radius d_1 . After each robot reaches its destination, either the whole configuration forms F , or P from which we remove the robot with maximum view from F from which we remove a point with maximal view (the configuration is no more in the active set of ψ_{RSB}), or the configuration is still in the active set of $\psi_{RSB|_Q}$ and satisfies the second or the third case. If the configuration is no more in the active set of $\psi_{RSB|_Q}$, then the configuration is static, so that $\psi_{RSB|_Q}$ still satisfies the termination awareness property.

In the second case, at least one robot r satisfies $|d_1| \geq |r|$ and $|r| > |d|$. Then, all such robots move toward $c(P)$ to reach the circle of radius d . After each robot reaches its destination, the configuration satisfies the third case. During this phase the configuration remains in the active set of $\psi_{RSB|_Q}$ since no robot reaches a point in F_r .

In the third case, the robots in Q are at most at distance d from $c(P)$. Then the robot election proceeds as previously described, except that a robot with destination p such that $|p| \geq d$ does not move. During this phase the configuration remains in the active set of $\psi_{RSB|_Q}$. Indeed, if $d_1 \neq d$, then there is at least one point in F_r that does not contain a robot (and it is not a point with maximum view since some points are closer to $c(P)$), otherwise (all points in F_r are at distance d to $c(P)$), there are at least two robots inside $\mathcal{D}(d)$ (because the whole configuration is active for $\psi_{RSB|_Q}$) and they cannot reach the circle of radius d .

Algorithm fixEnclosingCircle: locate the robot of $C(P) \cap P'$ on $C(P) \cap F'$ when $|C(F) \cap F'| = 2$

Phase Condition: $|C(F) \cap F'| \neq 2$ or there are only two robot in $C(P)$ located on the two point of $C(F) \cap F'$

```

if  $|C(P) \cap P'| = 2$  then
   $r \leftarrow$  greatest robot in interior  $C(P)$ 
   $C \leftarrow$  circle centered at  $c(P)$  that contains  $r$ 
  if  $|C \cap P| > 1$  then
     $r$  moves away from toward  $c(P)$  without reaching the circle of another robot nor  $C(P)$ 
  else
     $a \leftarrow \min_{r' \in C(P)} \text{ang}(r_{\max}, c(P), r')$ 
    if  $\text{ang}(r_{\max}, c(P), r) < a$  then
       $r$  moves away from  $c(P)$  to reach  $C(P)$ 
    else
       $r$  moves on  $C(P)$  in the indirect orientation to have an angle  $a/2$ 
  else
     $r \leftarrow$  greatest robot in  $C(P)$ 
     $r' \leftarrow$  smallest robot in  $C(P)$ 
    if  $r$  and  $r'$  are located the points of  $C(P) \cap F'$  then
       $r'' \leftarrow$  second smallest robot in  $C(P) \cap P'$ 
       $r''$  moves toward  $c(P)$  without reaching the circle of another robot
    else
      Let  $r_1, \dots, r_k$  be the other robots in  $C(P) \cap P'$  in the lexicographical order of their polar coordinates
      // perform the following movements while preserving  $C(P)$  and the ordering of robots
       $r$  moves on  $C(P)$  toward the greatest point in  $C(P) \cap F'$ 
       $r'$  moves on  $C(P)$  toward the smallest point in  $C(P) \cap F'$ 
      for  $i = 1, \dots, j$  do
         $a \leftarrow \text{ang}(r_{\max}, c(P), r') + i \times (\text{ang}(r_{\max}, c(P), r') + \text{ang}(r_{\max}, c(P), r)) / (j + 1)$ 
         $r_i$  moves on  $C(P)$  toward the point in  $C(P) \cap F'$  with angle  $a$ 

```

B Pattern Formation when $|C(F) \cap F'| = 2$

We execute this special phase before executing Phase 2 of Algorithm ψ_{DPF} , if $|C(F) \cap F'| = 2$. Once r_{\max} exists, if $|C(F) \cap F'| = 2$ and there are not exactly two robots in $C(P)$ located at the two points in $C(F) \cap F'$, then the following is executed.

If there are only two robots on $C(P)$, then the greatest robot in $\text{interior}(C(P))$ reaches $C(P)$, while remaining smaller than robots in $C(P)$ (see Action *ii*). Now, there are at least three robots on $C(P)$. The greatest robot r in $C(P)$ moves toward the greatest robot in $C(F)$, the smallest r' moves toward the other point in $C(P) \cap F'$, and the other robots choose evenly distributed destinations between r and r' . Those movements are done while keeping $C(P)$ and the ordering unchanged (like in Phase 3). The smallest robot is chosen for r' instead of the second greatest so that no robot can prevent r' to reach the smallest point in $C(P) \cap F'$, especially if it has a null angle. Once r and r' reach their destination, the other robots can leave $C(P)$, starting from the smallest. Those last movements change the ordering of r , so that it becomes the second greatest robot.

C The case of patterns that contain points of multiplicity

In this section, we assume that robots are endowed with multiplicity detection. If F contains points of multiplicity, but $c(F)$ is not a point of multiplicity, then l_f exists if $n \geq 4$. Indeed, a point in F with multiplicity greater than one does not hold $C(F)$, in the sense that if we decrease its multiplicity by one $C(F)$ remains unchanged. In this case our algorithm works in the same way. Indeed, in Phase 2, robots in a circle can move to their destination while preserving the ordering until they reach their destination. After reaching their destination they are incomparable, but it is not important since they do not move anymore.

The only case we need to study separately is when $c(F)$ is a point of multiplicity m . In the case F is not just a point of multiplicity n , we consider $\tilde{F} = F - \{(c(F), m)\} \cup \{(g_F, m)\}$ i.e., \tilde{F} is the original multiset

F , from which we remove the points at $c(F)$ and we add the point g_F with multiplicity m , where g_F is the middle between $c(F)$ and the point distinct from $c(F)$ with maximal view. Again g_F is not uniquely defined, but for two possible choices of g_F , the two resulting multisets \tilde{F} are similar. Then we use our algorithm using \tilde{F} instead of F . Once \tilde{F} is formed, the m closest robots to $c(\tilde{F})$ *i.e.*, robots at g_F , move to $c(\tilde{F})$. Again, the robots are aware that they are doing this last movement, by checking in the current configuration if the m closest robots to the center *are on the same half line starting at $c(F)$* , and the other robots form the pattern $F - \{(c(F), m)\}$. Such a configuration cannot be obtained when executing ψ_{RSB} so the previous lemmas are still true.

In the case F is a point of multiplicity n , we consider the set $\tilde{F} = F - \{(c(F), 1)\} \cup \{(g, 1)\}$ *i.e.*, \tilde{F} is the original multiset F , from which we decrease by one the multiplicity of $c(F)$ and we add an arbitrary point g (for a different point g , the resulting \tilde{F} is similar). Once \tilde{F} is formed *i.e.*, once a point of multiplicity $n - 1$ exists, then the unique robot that is not at a point of multiplicity, moves toward the point of multiplicity $n - 1$.

D Omitted Proofs

D.1 Proof of Property 1

Property 1 (restated). *Let P be a n -robot configuration. If $\rho(P) > 1$ or if P contains an axis of symmetry, then P contains a regular set.*

Proof. It is sufficient to show that at least one set Q in the increasing sequence Q_1, \dots, Q_k satisfies the three properties (a), (b), and (c). Let $\rho(P) = m$. If P does not contain an axis of symmetry, then Q_m contains the m robots with maximum identical view (with the same orientation). Indeed the other robots are not closer to $c(P)$ and form n -gons so that the m robots with maximum view do not hold $C(P)$. Then, Q_m forms a m -gon centered at $c(P)$, and P deprived of Q_m can still be partitioned in m -gons so that $\rho(P \setminus Q_m)$ is a multiple of m . In this case, $Q_{\max} = Q_m$.

Similarly, if P contains an axis of symmetry, Q_{2m} satisfies the three properties. □

D.2 Proof of Theorem 1

We start by several lemmas related to the Weber point. Let P be a n -robot configuration with $n \geq 7$.

Lemma 5. *Let $P' = P - \{r\} \cup \{r'\}$. If P is regular, then*

$$\text{ang}_{\min}(W(P'), W(P), r) \in \left[\frac{\pi}{4}, \frac{3\pi}{4} \right].$$

Proof. Let w be the symmetric of $W(P')$ with respect to $W(P)$. Let $\text{ang}_P(u) = \text{ang}_{\min}(w, W(P), u)$.

On the one hand, we know that:

$$\begin{aligned} \cos(\text{ang}_P(r)) + \sum_{p \in P - \{r\}} \cos(\text{ang}_P(p)) &= 0 \\ \cos(\text{ang}_{P'}(r')) + \sum_{p \in P - \{r'\}} \cos(\text{ang}_{P'}(p)) &= 0 \end{aligned}$$

So,

$$\cos(\text{ang}_P(r)) - \cos(\text{ang}_{P'}(r')) + \sum_{p \in P - \{r'\}} [\cos(\text{ang}_P(p)) - \cos(\text{ang}_{P'}(p))] = 0$$

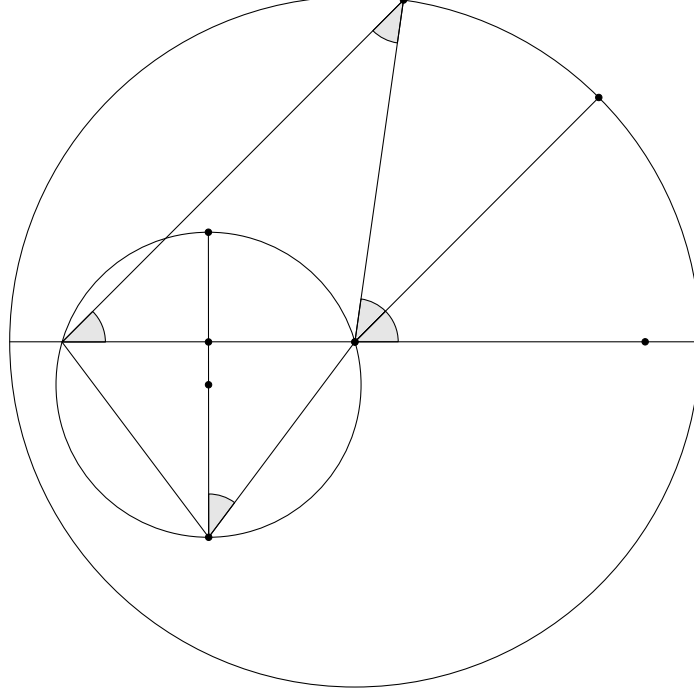


Figure 5: Worst case for the angle $ang_{\min}(W(P), p, W(P'))$

Also,

$$\forall p \in P - \{r\} \quad \cos(ang_P(p)) - \cos(ang_{P'}(p)) < 0 \text{ and } \cos(ang_P(r)) - \cos(ang_{P'}(r')) > 0$$

So,

$$\forall p \in P - \{r\} \quad \cos(ang_{P'}(p)) - \cos(ang_P(p)) \leq \cos(ang_P(r)) - \cos(ang_{P'}(r')) \quad (1)$$

The fact that P is regular with $|P| \geq 7$ implies that there is at least one robot in every cone centered at $W(P)$ of angle $\pi/2$. Let p in $P - \{r\}$ be a point such that $ang_P(p) \in [\frac{\pi}{4}, \frac{3\pi}{4}]$. Suppose for the purpose of contradiction that $ang_P(r) \notin [\frac{\pi}{4}, \frac{3\pi}{4}]$. Then $|ang_P(p) - \pi/2| < |ang_P(r) - \pi/2|$, which contradicts Equation (1). \square

Lemma 6. *Let $P' = P - \{r\} \cup \{r'\}$, with $ang_{\min}(r, W(P), r') = \theta \leq \alpha_{\min}(P)$, and $|r' - W(P)| = |r - W(P)| = \min_{p \in P} |p - W(P)|$. We have $\forall p \in P$, $ang_{\min}(W(P), p, W(P')) \leq 2\theta$.*

Proof. With the same notations as in the previous lemma, we have $ang_P(r) \in [\pi/4, 3\pi/4]$. First, we suppose $|W(P') - W(P)| < |r - W(P)|$. The worst case happens when $ang_{P'}(r') = ang_P(r) = \pi/4$ (see Figure 5). Without loss of the generality, we suppose $|W(P) - r| = 1$. In this case, by the law of sines,

$$|B - W(P)| = \frac{\sin(\theta)}{\sqrt{2}}.$$

Then,

$$|F - W(P)| = \frac{1}{\sqrt{2}}$$

Since $\theta < \pi/4$ (because $|W(P') - W(P)| < |r - W(P)|$), we obtain

$$|F - H| = \frac{1}{\cos(\theta)\sqrt{2}} < 1.$$

This implies that the disc of diameter FH is strictly included in the disc centered at $W(P)$ of radius 1, and since a point p such that $\text{ang}_{\min}(W(P), p, W(P')) < 2\theta$ must be inside this disc, then $|p - W(B)| < |r - W(B)|$.

In the case $|W(P') - W(P)| \geq |r - W(P)|$, since $\text{arg}_{P'}(r') \geq \text{arg}_P(r)$ (because otherwise the half lines $]W(P), r$ and $]W(P'), r'$ would intersect, which is impossible), then $\theta \geq \pi/4$. Since $\theta \leq \alpha_{\min}(P) \leq 2\pi/7$, this is not possible if $|P| \geq 7$. \square

Lemma 7. *Let the robots be indexed in the clockwise order $r = r_1, r_2, \dots, r_n$ around the Weber point $W(P)$. Let $P' = P - \{r\} \cup \{r'\}$, with $\text{ang}_{\min}(r, W(P), r') = \theta \leq \alpha_{\min}(P')/4$ and $|r - W(P)| = \min_{i \in [1, n]} |r_i - W(P)|$. Then, the robots in P' are ordered in the same way as in P around $W(P')$.*

Proof. For a vector \vec{u} , let $\text{ang}_{\vec{u}}(p) = \text{ang}\left(\overrightarrow{W(P)p}, \vec{u}\right)$, and $\text{ang}'_{\vec{u}}(p) = \text{ang}\left(\overrightarrow{W(P')p}, \vec{u}\right)$. We see that r_1, r_2, \dots, r_n are ordered around $W(P)$ is equivalent to:

$$i \in [1, n-1], \quad \text{ang}_{\vec{u}}(r_i) < \text{ang}_{\vec{u}}(r_{i+1})$$

with \vec{u} a vector such that $\text{ang}_{\vec{u}}(r)$ is minimal among robots in P , an $\text{ang}'_{\vec{u}}(r')$ is minimal among robots in P' (this is possible since $\text{ang}_{\min}(r, W(P), r') \leq \alpha_{\min}(P')/4$). From lemma 6, we have

$$\forall p \in P, \quad |\text{ang}_{\vec{u}}(p) - \text{ang}'_{\vec{u}}(p)| = \text{ang}_{\min}(W(P), p, W(P')) < \alpha_{\min}(P')/2.$$

That implies that for all $i \in [1, n-1]$, $\text{ang}'_{\vec{u}}(r_i) < \text{ang}'_{\vec{u}}(r_{i+1})$. \square

Lemma 8. *Let a, b , and c be distinct points in \mathbb{R}^2 . Let θ_1 and θ_2 in \mathbb{R}^* . There is at most one point p such that:*

$$\begin{cases} \text{ang}(a, p, b) = \theta_1 \leq \pi \\ \text{ang}(b, p, c) = \theta_2 \leq \pi \\ \text{ang}(c, p, a) \leq \pi \end{cases}$$

Proof. The set of points p such that $\text{ang}(a, p, b) = \theta_1$ is a circular arc $C_{a,b}$ passing through a and b . Also the set of points p such that $\text{ang}(b, p, c) = \theta_2$ is a circular arc $C_{b,c}$ passing through b and c . If there are three distinct points in $C_{a,b} \cap C_{b,c}$, then $C_{a,b}$ and $C_{b,c}$ are two arcs of the same circle. Yet, for any point p in this circle, either $\text{ang}(a, p, b) > \pi$ or $\text{ang}(b, p, c) > \pi$ or $\text{ang}(c, p, a) > \pi$, depending on the position of p in the circle. Hence, a contradiction. So, there is at most one point distinct from b in $C_{a,b} \cap C_{b,c}$. \square

Lemma 9. *Let P be a n -robot shifted regular configuration. If $n \geq 7$, the shifted (elected) robot is unique.*

Proof. Let P be a n -robot shifted regular configuration with shifted robot r (resp., r'), associated to the m -regular set $P_{\text{reg}} = P - \{r\} \cup \{r_{\text{reg}}\}$ (resp., associated to the m' -regular set $P'_{\text{reg}} = P - \{r'\} \cup \{r'_{\text{reg}}\}$). From Lemma 7, the ordering is unchanged between P , P_{reg} and P'_{reg} . So, let r_1, \dots, r_n be a counterclockwise ordering of robots in P around $W(P)$. An ordering of robots in P_{reg} (resp., P'_{reg}) is obtained by replacing $r = r_{i_0}$ by r_{reg} (resp., $r' = r'_{i'_0}$ by r'_{reg}).

• If $n = m = m'$, since $n \geq 6$, there exist three robots r_i, r_j , and r_k ($i < j < k$) in $P_{\text{reg}} \cap P'_{\text{reg}} = P - \{r\} - \{r'\}$ such that

$$\text{ang}(r_i, c(P_{\text{reg}}), r_j) \leq \pi \quad \wedge \quad \text{ang}(r_j, c(P_{\text{reg}}), r_k) \leq \pi \quad \wedge \quad \text{ang}(r_k, c(P_{\text{reg}}), r_i) \leq \pi$$

The angle between two robots is uniquely determined by the difference between their indexes (for instance $\text{ang}(r_i, c(P_{\text{reg}}), r_j) = 2(j - i)\pi/n$). Since the ordering around $c(P'_{\text{reg}})$ is the same, we have:

$$\begin{cases} \text{ang}(r_i, c(P_{\text{reg}}), r_j) = \text{ang}(r_i, c(P'_{\text{reg}}), r_j) \\ \text{ang}(r_j, c(P_{\text{reg}}), r_k) = \text{ang}(r_j, c(P'_{\text{reg}}), r_k) \\ \text{ang}(r_k, c(P_{\text{reg}}), r_i) = \text{ang}(r_k, c(P'_{\text{reg}}), r_i) \end{cases}$$

By Lemma 8, $c(P_{reg}) = c(P'_{reg})$. Once we know the center is the same, since $n \geq 3$, the shifted robot is uniquely determined.

- If $m = n/2$ or $m' = n/2$, the configuration can be divided in two subsets of $n/2$ robots that are both $n/2$ -regular, so that if $n \geq 8$ we can find three robots as before in one of the $n/2$ -regular subsets to find the common center and the shifted robot. If $m = 3$, it is possible to have $c(P_{reg}) \neq c(P'_{reg})$, so that n must be greater than 7. \square

Theorem 1 (restated). *Let $n \geq 7$, and P be a n -robot configuration that contains a ε -shifted- m -regular set Q of center c . Then Q , c , m , ε and the shifted robot are unique.*

Proof. Let P be a n -robot configuration that contains a ε -shifted- m -regular set Q . If $\varepsilon = 0$ and $Q = P$ i.e., the configuration is regular, then the center is unique since it is the Weber point. If $\varepsilon \neq 0$ and $Q = P$ i.e., the configuration is shifted regular, Lemma 9 implies that the center and the shifted robot are unique.

Now suppose $Q \subsetneq P$. This implies that the possible shifted regular sets we consider have the same center $c(P)$. In the construction of Q_{\max} , the increasing sequence of views is unique since each local view is unique. So, the sequence Q_i is also unique. Since all the conditions on each Q_i can be checked the same way by each robot, the maximum set Q_i that satisfies all the conditions, that is Q_{\max} , is unique. Therefore, if $\varepsilon = 0$, then the regular set of P is unique.

Now suppose that $\varepsilon > 0$ and Q is the shifted regular set of P . The shifted robot r_e is the only robot that minimizes the angle $\alpha_{\min}(r_e, P)$ with the other robots. Indeed, since the robots in $P \setminus Q$ form a m -gon centered at $c(P)$, robots in $Q - \{r_e\}$ all have the same (and greater than $\alpha_{\min}(r_e, P)$) minimum angle with the other robots. \square

D.3 Proof of property 2

The following property gives precisely what moves robots are allowed to make while maintaining a configuration with a shifted regular set (with the same center and shifted robot) that contains the same robots. In the sequel, we abusively use 0-shifted- m -regular set instead of m -regular set.

Property 2. *Let P be a n -robot configuration ($n \geq 7$) that contains a ε -shifted- m -regular set Q (with $0 \leq \varepsilon \leq 1/4$). Let D_{\max} be the largest open disc centered at $c(P)$ that does not contain any robot in $P \setminus Q$. After one of the following movements, the configuration contains a ε' -shifted- m -regular set formed with the same robots, with the same center and, if $\varepsilon > 0$, the same shifted robot as in Q :*

- (M1) $\varepsilon = 0$ and a robot in Q moves radially in $D_{\max} - \{c(P)\}$;
- (M2) $\varepsilon = 0$ and one of the robots that is closest to $c(P)$ moves on its circle, creating a shift $0 < \varepsilon' \leq 1/4$;
- (M3) $\varepsilon > 0$ and the shifted robot moves on, or inside, its circle, preserving a shift $0 < \varepsilon' \leq 1/4$;
- (M4) $\varepsilon > 0$ and a robot in $Q - \{r_e\}$ (with r_e the shifted robot) moves radially in $D_{\max} \setminus \mathcal{D}(|r_e|)$.

Proof. First, if $reg(P) = P$ then the property is given by the uniqueness of the shifted regular set.

Case (M1): we suppose that P contains a m -regular set $Q \neq P$ with $|Q| = q$ ($q = m$ or $q = 2m$). Let Q' be the robots in Q after the movement defined in (M1). To prove the property, it is sufficient to show that (i) Q' contains the q robots with greatest view, that (ii) Q' satisfies the three property given in definition 1, and that (iii) Q' is the largest such set.

(i): The views of robots in Q' that are inside D are greater than the views of robot in $P \setminus Q$, and the robots in Q' that are in $Circum(D)$ did not moved (because robots are not allowed to move from inside D to its circumference). So their views are still greater than the views in $P \setminus Q$. Moreover, since robots in Q do not hold $C(P)$, then robots in Q' do not hold $C(P)$ either.

(ii): After the movement, Q' is clearly a m -regular set. Since robots $P \setminus Q$ have not moved, m still divides $\rho(P \setminus Q)$ and if Q is bi-angled, then Q' is bi-angled and the virtual axis of Q' are still axes of symmetry of $P \setminus Q$.

(iii): Since the three property are invariant by radial movement, the properties violated by a superset of Q are also violated by the same superset of Q' containing the same robots.

Case (M2): Let r' be the position of r after its movement described in (M2). Then, Q' is a ε -shifted- m -regular set with associated regular set $Q = Q' - \{r'\} + \{r\}$.

Case (M3): Since only the shifted robot moves, by definition of the associated regular set, the associated regular set before and after the movement is the same.

Case (M4): Let $Q_{reg} = Q - \{r_e\} + \{r'_e\}$ be the regular set associated with Q . Let Q'_{reg} be the position of robots in Q_{reg} after the movement described in (M4). By applying Case (M1), $P' - \{r_e\} + \{r'_e\}$ contains Q'_{reg} as m -regular set formed by the same robots, with the same center as Q_{reg} . Since no robots moved closer to the center than r_e , P' contains a ε -shifted- m -regular set that has Q'_{reg} as associated regular set. \square

D.4 Termination of phase 3

Lemma 10. *During phase 3, each robot in $P - \{r_s\}$ reaches its destination in F' and $C(P)$ remains unchanged.*

First we show that there is no deadlock. We have $r_1 > r_2 > \dots > r_m$ and $d_1 > d_2 > \dots > d_m$. Now, r_k has destination d_k and moves toward it (staying on C_i) in the direct orientation if $r_k > d_k$ and in the indirect orientation otherwise. If $r_{k+1} \geq d_k$ (resp., $r_{k-1} \leq d_k$), then r_k cannot reach d_k until $r_{k+1} < d_k$ (resp., until $r_{k-1} > d_i$). There cannot be a deadlock because if $r_m \geq d_i$ (resp., $r_1 \leq d_k$), nothing can prevent r_m (resp. r_1) to move to d_m (resp., d_1), which would imply $r_m < d_k$ (resp. $r_1 > d_1$).

Now suppose for the purpose of contradiction that $C(P)$ is modified. That means that there are two robots r and r' on C_1 that form an angle greater than π . Before $C(P)$ is modified, they form an angle of at most π , so that one robot's movement on C_1 in the direct orientation, and the other's movement on C_1 in the indirect orientation. This is possible only if there is no point in F on C_1 between r and r' , which is a contradiction with the fact that $C(P) = C(F)$.