Circle Formation of Weak Mobile Robots *

Yoann Dieudonné[1] Ouiddad Labbani-Igbida[2]
Franck Petit[1]

[1] LaRIA, CNRS FRE 2733
Université de Picardie Jules Verne
France

[2] CREA
Université de Picardie Jules Verne
France

Abstract

The contribution is twofold. We first show the validity of the conjecture of Défago and Konagaya in [DK02], i.e., there exists no deterministic oblivious algorithm solving the Uniform Transformation Problem for any number of robots*. Next, a protocol which solves deterministically the Circle Formation Problem in finite time for any number n of weak robots—$n \notin \{4, 6, 8\}$—is proposed. The robots are assumed to be uniform, anonymous, oblivious, and they share no kind of coordinate system nor common sense of direction.

Keywords: Distributed Computing, Formation of Geometric Patterns, Mobile Robot Networks, Self-Deployment.

1 Introduction

In this paper, we address the class of distributed systems where computing units are *autonomous mobile robots* (also sometimes referred to *sensors* or *agents*), i.e., devices equipped with sensors which do not depend on a central scheduler and designed to move in a two-dimensional plane. Also, we assume that the robots cannot remember any previous observation nor computation performed in any previous step. Such robots are said to be *oblivious* (or *memoryless*). The robots are also *uniform* and *anonymous*, i.e, they all have the same program using no local parameter (such that an identity) allowing to differentiate any of them. Moreover, none of them share any kind of common coordinate mechanism or common sense of direction, and they communicate only by observing the position of the others.

The motivation behind such a weak and unrealistic model is the study of the minimal level of ability the robots are required to have in the accomplishment of some basic cooperative tasks in a deterministic way, e.g., [SS90, SY99, FPSW99, Pre02]. Among them, the *Circle Formation Problem* (CFP) has received a particular attention. The CFP consists in the design of a protocol insuring that starting from an initial arbitrary configuration, all n robots eventually form a circle with equal spacing between any two adjacent robots. In other words, the robots are required to form a *regular n-gon* when the protocol terminated.

*Independantly of our work, in [FPS06], the authors show the validity of the Conjecture of Défago & Konagaya. This part has been removed from the submitted version of this technical report.

Related Works. An informal CFP algorithm is presented in [Deb95] to show the relationship between the class of pattern formation algorithms and the concept of self-stabilization in distributed systems [Dol00]. In [SS96], an algorithm based on heuristics is proposed for the formation of a circle approximation. A CFP protocol is given in [SY99] for non-oblivious robots with an unbounded memory. Two deterministic algorithms are provided in [DK02, CMN04]. In the former work, the robots asymptotically converge toward a configuration in which they are uniformly distributed on the boundary of a circle. This solution is based on an elegant Voronoi Diagram construction. The latter work avoid this construction by making an extra assumption on the initial position of robots. In [DP06], properties on Lyndon words are used to achieve a Circle Formation Protocol (the exact n-gon is eventually built) for a prime number of robots. All the above solutions work in the semi-asynchronous model introduced in [SY96]. The solution in [Kat05] works in a fully asynchronous model, but when n is even, the robots may only achieve a biangular circle—the distance between two adjacent robots is alternatively either α or β.

A common strategy in order to solve a non trivial problem as CFP is to combine subproblems which are easier to solve. In general, CFP is separated into two distinct parts: The first subproblem consists in placing the robots along the boundary of a circle C, without considering their relative positions. The second subproblem, called *uniform transformation problem* (UTP), consists in starting from there, and arranging robots, without them leaving the circle C, evenly along the boundary of C. In [DK02], the authors present an algorithm, for the second subproblem which converges toward a homogeneous distribution of robots, but it does not terminate deterministically. By the way, they conjecture that there is no deterministic solution solving UTP in finite time in the semi-asynchronous model in [SY96]—the robots being uniform, anonymous, oblivious, and none of them sharing any kind of coordinate system or common sense of direction.

Contribution. The contribution is twofold. We first show the validity of the conjecture of Défago and Konagaya in [DK02], i.e., there exists no deterministic oblivious algorithm solving the Uniform Transformation Problem for any number of robots[1]. Next, we propose the first protocol which solves deterministically CFP in finite time for any number n of weak robots, provided that $n \notin \{4, 6, 8\}$. By weak, we mean that the robots are assumed to be uniform, anonymous, oblivious, and they share no kind of coordinate system nor common sense of direction. Our protocol is not based on UTP, but it is based on concentric circles formed by the robots.

Outline of the Paper. In the next section (Section 2), we describe the distributed systems and the model we consider in this paper. In the same section, we present the problem considered in this paper. Section 3 addresses the conjecture of Défago and Konagaya. The algorithm is proposed in Section 4. Finally, we conclude this paper in Section 5.

2 Preliminaries

In this section, we define the distributed system, basic definitions and the problem considered in this paper.

[1]Independantly of our work, in [FPS06], the authors show the validity of the Conjecture of Défago & Konagaya. This part has been removed from the submitted version of this technical report.

Distributed Model. We adopt the model introduced [SY96], in the remainder referred as SSM. The *distributed system* considered in this paper consists of n robots $r_1, r_2, \cdots, r_n$—the subscripts $1, \ldots, n$ are used for notational purpose only. Each robot r_i, viewed as a point in the Euclidean plane, move on this two-dimensional space unbounded and devoid of any landmark. When no ambiguity arises, r_i also denotes the point in the plane occupied by that robot. It is assumed that the robots never collide and that two or more robots may simultaneously occupy the same physical location. Any robot can observe, compute and move with infinite decimal precision. The robots are equipped with sensors allowing to detect the instantaneous position of the other robots in the plane. Each robot has its own local coordinate system and unit measure. The robots do not agree on the orientation of the axes of their local coordinate system, nor on the unit measure. They are *uniform* and *anonymous*, i.e, they all have the same program using no local parameter (such that an identity) allowing to differentiate any of them. They communicate only by observing the position of the others and they are *oblivious*, i.e., none of them can remember any previous observation nor computation performed in any previous step.

Time is represented as an infinite sequence of time instant $t_0, t_1, \ldots, t_j, \ldots$ Let $P(t_j)$ be the multiset of the positions in the plane occupied by the n robots at time t_j ($j \geq 0$). For every t_j, $P(t_j)$ is called the *configuration* of the distributed system in t_j. $P(t_j)$ expressed in the local coordinate system of any robot r_i is called a *view*, denoted $v_i(t_j)$. At each time instant t_j ($j \geq 0$), each robot r_i is either *active* or *inactive*. The former means that, during the computation step (t_j, t_{j+1}), using a given algorithm, r_i computes in its local coordinate system a position $p_i(t_{j+1})$ depending only on the system configuration at t_j, and moves towards $p_i(t_{j+1})$—$p_i(t_{j+1})$ can be equal to $p_i(t_j)$, making the location of r_i unchanged. In the latter case, r_i does not perform any local computation and remains at the same position.

The concurrent activation of robots is modeled by the interleaving model in which the robot activations are driven by a *fair scheduler*. At each instant t_j ($j \geq 0$), the scheduler arbitrarily activates a (non empty) set of robots. Fairness means that every robot is infinitely often activated by the scheduler.

The Circle Formation Problem. In this paper, the term "*circle*" refers a circle having a radius strictly greater than zero. Consider a configuration at time t_k ($k \geq 0$) in which the positions of the n robots are located at distinct positions on the circumference of a circle C. At time t_k, the *successor* r_j, $j \in 1 \ldots n$, of any robot r_i, $i \in 1 \ldots n$ and $i \neq j$, is the single robot such that no robot exists between r_i and r_j on C in the clockwise direction. Given a robot r_i and its successor r_j on C centered in O:

1. r_i is said to be the *predecessor* of r_j;
2. r_i and r_j are said to be *adjacent*;
3. $\widehat{r_i O r_j}$ denotes the angle centered in O and with sides the half-lines $[O, r_i)$ and $[O, r_j)$ such that no robots (other than r_i and r_j) is on C inside $\widehat{r_i O r_j}$.

Definition 1 (regular n-gon) *A cohort of n robots ($n \geq 2$) forms (or is arranged in) a regular n-gon if the robots take place on the circumference of a circle C centered in O such that for every pair r_i, r_j of robots, if r_j is the successor of r_i on C, then $\widehat{r_i O r_j} = \delta$, where $\delta = \frac{2\pi}{n}$. The angle δ is called the* characteristic angle *of the n-gon.*

The problem considered in this paper, called CFP (*Circle Formation Problem*) consists in the design of a distributed protocol which arranges a group of n ($n > 2$) mobile robots with initial distinct positions into a *regular n-gon* in finite time. (We ignore the trivial cases $n \leq 2$ because in that cases, they always form a regular n-gon.)

3 On The Conjecture of Défago and Konagaya

Definition 2 (UTP Algorithm) *A distributed algorithm A solves the uniform transformation problem (UTP) if and only if, starting from a configuration where the robots are arbitrarily located along the circumference of a circle C, (i) none of the robots leaves the circumference of C during the execution of A and, (ii) all the robots eventually form a regular n-gon.*

In this section, we show the validity of the following conjecture:

Conjecture 3 ([DK02]) *There exists no deterministic oblivious algorithm solving UTP in SYm for any number of robots.*

Definition 4 (Biangular circle) *A cohort of n robots ($n \geq 2$) forms (or is arranged in) a biangular circle if the robots take place on the circumference of a circle C centered in O and there exist two non zero angles α, β such that for every pair r_i, r_j of robots, if r_j is the successor of r_i on C, then $\widehat{r_iOr_j} \in \{\alpha, \beta\}$ and α and β alternate in the clockwise direction.*

Remark 5 *In a biangular circle, $\alpha + \beta = 4\frac{\pi}{n}$.*

Obviously, if $\alpha = \beta$ then, the n robots form a regular n-gon, and n can either odd or even. If $\alpha \neq \beta$, then n must be even ($n = 2p,\ p \geq 1$). In that case, the biangular circle is called a *strict* biangular circle—refer to Figure 1. In that case, there exist two distinct groups G_1 and G_2 such that:

1. $|G1| = |G2| = \frac{n}{2}$;
2. The $\frac{n}{2}$ robots in G_1 (resp. G_2) form a regular $\frac{n}{2}$-gon;
3. The robots do not form a regular n-gon.

Given a configuration $P(t_j)$, if the n robots form a strict biangular circle, then $G_1(t_j)$ (resp. $G_2(t_j)$) indicates the positions of the robots in G_1 (resp. G_2) at time t_j.

The idea of proof is as follows: we show, for each any strategy of the robots, there exists a particular activation schedule foiling it. More precisely, we show that if initially the robots form a strict biangular circle, then they may not eventually form a regular n-gon in a deterministic way because of the unpredictability of the activation schedule. This result holds for the case $n = 2p$ ($p > 1$). So, it proves the general result.

Lemma 6 *Let A be a deterministic oblivious algorithm solving UTP in finite time and a configuration $P(t_j)$ such that the n robots form a strict biangular circle in $P(t_j)$. If any robot r_i becomes active at time t_j, by executing A, it moves toward a position $p(t_{j+1}$ such that $p_i(t_j) \neq p_i(t_{j+1})$.*

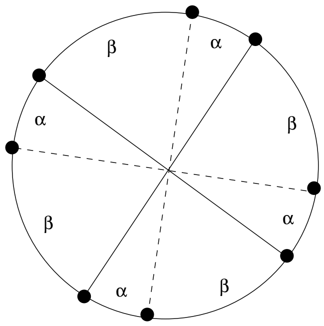


Figure 1: An example showing a strict biangular circle ($\alpha \neq \beta$).

Proof. Since the robots have no common coordinate system and sense of direction, then all of them may have the same view at t_j, i.e., $v_i(t_j) = v_k(t_j)$, for all r_i,r_k. Such a configuration is shown in Figure 2. Assume by contradiction that, there exists a robot r_i which becomes active at t_j and move toward a position $p(t_{j+1})$ such that $p_i(t_j) \neq p_i(t_{j+1})$. Since A is a deterministic algorithm, if all the robots are the same view, then all the active robots choose the same behavior, i.e., $\forall r_i$ such that r_i is active at t_j, r_i move to a position $p(t_{j+1})$ such that $p_i(t_j) \neq p_i(t_{j+1})$. From the model, all the inactive robots remain at the same position at time t_{j+1}. So, $P(t_j) = P(t_{j+1}$. Since the robots are oblivious and A is a deterministic algorithm, we can easily deduce by induction (starting from t_j) that the robots always form a stric biangular circle by executing A. □

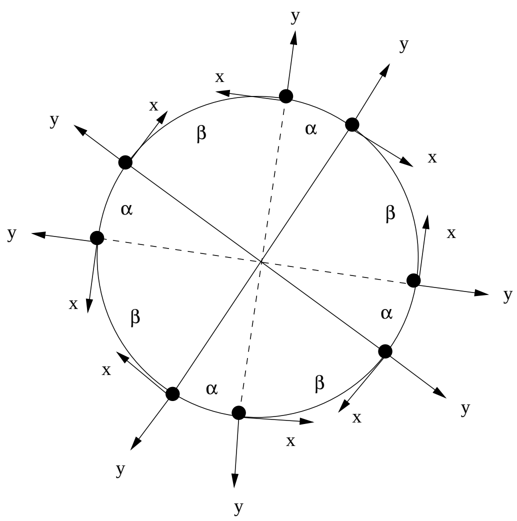


Figure 2: An example showing the initial configuration in the proof of Lemma 6.

Lemma 7 *Let A be a deterministic oblivious algorithm solving UTP in finite time and a configuration $P(t_j)$ such that the n robots form a strict biangular circle in $P(t_j)$. If any robot r_i becomes active at time t_j, by executing A, it moves toward a position $p_i(t_{j+1})$ such that $p_i(t_{j+1}) \neq p_k(t_j)$ for all $r_k \neq r_i$.*

Proof. By contradiction, assume that there exists r_i, r_k such that r_i moves toward a position $p_i(t_{j+1})$ such that $p_i(t_{j+1}) = p_k(t_j)$. Clearly, if r_k is inactive at time t_j, r_i and r_k have the same position at time t_{j+1}. Assume that r_i and r_k have the same coordinate system. So, they share the same view, i.e., $v_i(t_{j+1}) = v_k(t_{j+1})$. Assume that from t_{j+1} on, r_i and r_k are always active at the same time. So, from t_{j+1} on, for any move that r_i makes by executing A, r_k makes the same move as r_i. Therefore, at time t_{j+2}, r_i and r_k are again located at the same point and they share the same view. By induction, starting from t_{j+1}, it could be impossible to separate r_i and r_k in a deterministic manner. Hence, the n-gon cannot be eventually formed and A is not a deterministic oblivious algorithm solving UTP in finite time. □

Lemma 8 *There exists no algorithm deterministic oblivious algorithm A solving UTP in SYm starting from a configuration where the robots form a strict biangular circle.*

Proof. Assume by contradiction that there exists a deterministic oblivious algorithm A solving UTP in Sym starting from a configuration where the robots form a strict biangular circle. Assume that the n robots that, initially, the robots in G_1 (resp in G_2) of the strict biangular circle have the same view. Note that the view of the robots in G_1 may be different than the view of the robots in $G2$. In such a configuration, the robots are said to be in a *special biangular circle*. In the following of the proof, we also assume that if one robot in G_1 (resp. G_2) becomes active at time t_j, then all the robots in G_1 (resp. G_2) are active in t_j.

By fairness, at least one robot r_i becomes active at time t_j. Without loss of generality, assume that $r_i \in G_1$. By assumption, all the robots in G_1 are active in t_j. There are only two cases:

1. The robots $\in G_1$ move such that $G_1(t_{j+1}) \cup G_2(t_j)$ do not form a regular n-gon. Then, assuming that no robot in G_2 is active at t_j, $G_2(t_j) = G_2(t_{j+1})$. So, $G_1(t_{j+1}) \cup G_2(t_{j+1})$ do not form a regular n-gon.

2. The robots $\in G_1$ move such that $G_1(t_{j+1}) \cup G_2(t_j)$ form a regular n-gon. Then, assume that all the robots in G_2 are active at t_j. Clearly, the only possibility that at time t_{j+1}, the robots form a regular n-gon is that $G_2(t_{j+1})$ coincides with $G_2(t_j)$. This contradicts Lemma 6 and 7. Thus, $G_1(t_{j+1}) \cup G_2(t_{j+1})$ do not form a regular n-gon.

So, in both cases, $G_1(t_{j+1}) \cup G_2(t_{j+1})$ do not form a regular n-gon. Since all the robots in G_1 (resp. G_2) share the same view and execute the same deterministic algorithm A, every robot r_i in G_1 (resp. G_2) moves in the exact same way at the same time along the boundary of a same circle. Thus, either $G_1(t_{j+1}) \neq G_2(t_{j+1})$ or $G_1(t_{j+1}) = G_2(t_{j+1})$. In the former case, the robots form a biangular circle at t_{j+1}. The latter case, it would be impossible to separate G_1 and G_2 in a deterministic manner. The lemma is proven by induction. □

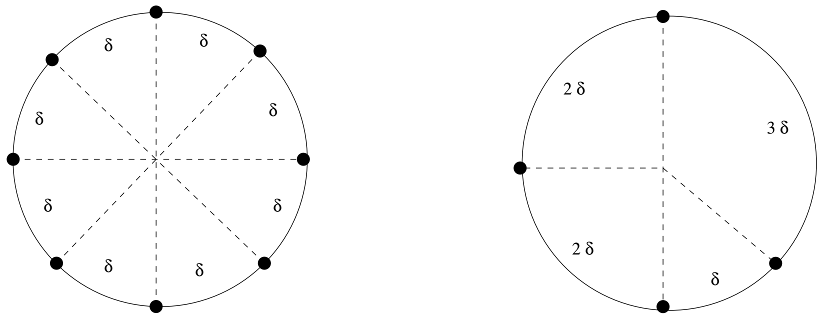
The proof of Conjecture 3 directly follows from Lemma 8.

4 Circle Formation Protocol

In this section, we present the main result of this paper. We first provide particular configurations of the system which we use for simplifying the design and proofs of the protocol. Next, the protocol is presented.

4.1 Definitions and Basics properties

Definition 9 (regular (k,n)-gon) *A cohort of k robots ($0 < k \leq n$) forms a regular (k,n)-gon if their positions coincide with a regular n-gon such that $n-k$ robots are missing.*



(a) A 8-gon ($\alpha = \frac{\pi}{4}$). (b) A $(4,8)$-gon obtained removing 4 robots from the corresponding 8-gon.

Figure 3: An example showing a (k,n)-gon.

An example of a (k,n)-gon is given in Figure 3. Given a (k,n)-gon such that $k \geq 2$, if p robots are missing (w.r.t. the corresponding n-gon) between two adjacent robots, then $\widehat{rOr'} = (p+1)\frac{2\pi}{n}$. Given a $(1,n)$-gon, then number of missing robot is equal to $n-1$. Remark that since the uniqueness of any circle is guaranteed by passing through 3 points only, there is an infinity of circles passing through 1 or 2 robots. So, if $k \leq 2$, then there is an infinity of (k,n)-gon passing through k robots.

Let C_1 and C_2 be two circles having their radius greater than 0. C_1 and C_2 are said to be *concentric* if they share the same center but their radius are different. Without lost of generality, in the remainder, given a pair (C_1, C_2) of concentric circles, C_1 (resp. C_2) indicates the circle with the greatest radius (resp. smallest radius).

Definition 10 (Concentric Configuration) *The system is said to be in a concentric configuration if there exists a pair of concentric circles (C_1, C_2) and a partition of the n robots into two subsets A and B such that every robot of A (respectively B) is located on C_1 (resp. C_2).*

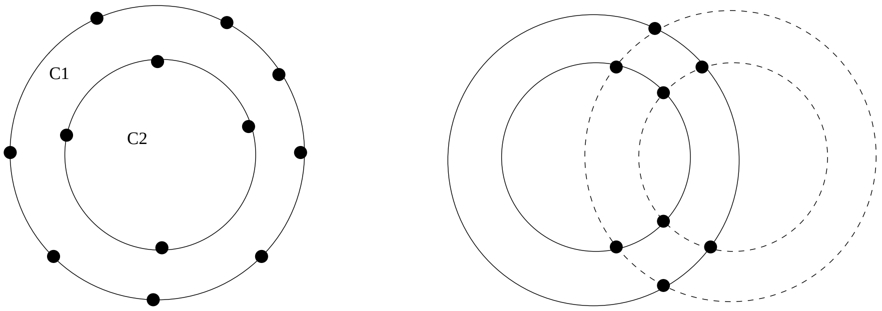
Remark 11 *$A \neq \emptyset$ and $B \neq \emptyset$.*

Remark 12 *If $n \leq 8$, then the pair (C_1, C_2) may not be unique.*

An example illustrated Remark 12 is given in Figure 4.

Lemma 13 *If the system is in a concentric configuration and if $n > 8$, then there exists a single pair (C_1, C_2) in which all the robots are located.*

Proof. Assume by contradiction, that the system is in a concentric configuration, $n > 8$ and there exists two pairs $\gamma = (C_1, C_2)$ and $\gamma' = (C'_1, C'_2)$ such that $\gamma \neq \gamma'$ (i.e., $C_1 \neq C'_1$, $C_1 \neq C'_2$, $C_2 \neq C'_1$ and $C_2 \neq C'_2$) and in which all the robots are located. Since two different circles share at most two



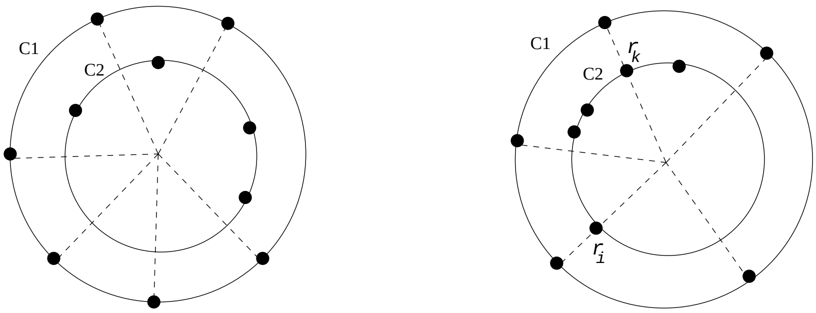
(a) An example of a concentric configuration with $n = 12$.

(b) An example showing that the pair of concentric circle may not be unique with $n \leq 8$.

Figure 4: Examples of concentric configurations.

points, the pairs γ can share at most eight robots with γ' (refer to Case (b) in Figure 4). Since by assumption $n \geq 9$, there exists at least one robot which is located on either C_1 or C_2, but which is located on neither C'_1 nor C'_2. This contradicts the fact that each robot is located either on C'_1 or on C'_2. □

So, from Lemma 13, when the system is in a concentric configuration and $n \geq 9$, the pair (C_1, C_2) is unique. In such a configuration, given a robot r, $proj(r)$ denotes the projection of r on C_1, i.e., the intersection between the half-line $[c, r)$ and C_1, where c is the center of (C_1, C_2). Obviously, if r is located on C_1, then $proj(r) = r$. We denote by Π the projection set of the n robots. In a concentric configuration, if $|\Pi| = n$, then the radii passing through the robots on $C1$ split up the disk bounded by C_1 into *sectors*.



(a) (b)

Figure 5: The concentric configuration shown in Case (a) is split up into sectors, whereas the one in Case (b) is not because the some robots on $C1$ are located on the projections of r_i and r_k.

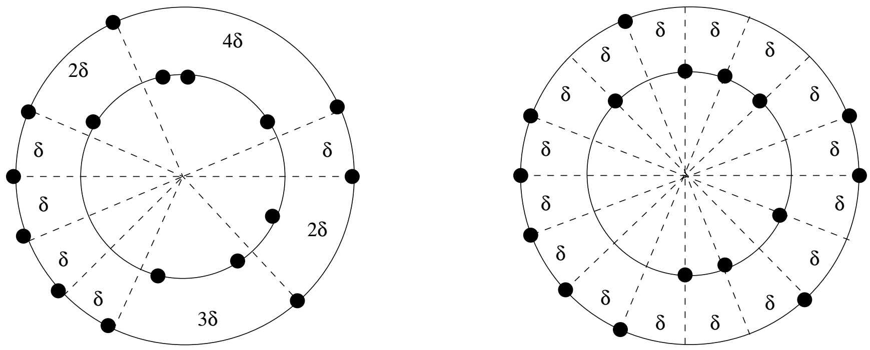
Definition 14 (quasi n-gon) *A cohort of n robots ($n \geq 9$) forms an (arbitrary) quasi n-gon iff the three following conditions hold:*

1. *The robots form a concentric configuration divided into sectors;*

2. *The robots on C_1 form a regular (k, n)-gon;*

3. *In each sector, if p robots are missing on C_1 to form a regular n-gon, then p robots are located on C_2 in the same sector.*

A quasi n-gon is said to be *aligned* iff Pi coincide with a regular n-gon. Two quasi n-gon are shown in Figure 6, the first one is arbitrary, the other one is aligned.



(a) An arbitrary quasi n-gon. (b) An aligned quasi n-gon.

Figure 6: Two quasi n-gon with $n = 16$.

4.2 The Protocol

Let us consider the overall scheme of our protocol presented in Algorithm 1. It is mainly based on the particular configurations presented in the previous subsection.

As mentioned in the introduction, the proposed scheme is combined with the protocol presented in [Kat05] which leads a cohort of n robots from an arbitrary to a biangular configuration, with $n \geq 2$. In the remainder, we refer to the protocol in [Kat05] as Procedure $< A \rightsquigarrow B >$—from an *A*rbitrary configuration to a *B*iangular configuration. The model used in [Kat05], called Corda [Pre02], allows more asynchrony among the robots than the semi-asynchronous model used in this paper—let us call it SSM. However, we borrow the following result from [Pre02]:

Theorem 15 *[Pre02] Any algorithm that correctly solves a problem P in Corda, correctly solves P in SSM.*

The above result means that Procedure $< A \rightsquigarrow B >$ can be used in SSM. Obviously, Procedure $< A \rightsquigarrow B >$ trivially solves the CFP if the number of robots n is odd. So, to solve CFP for any number of robots, it remains to deal with a system in a strict biangular configuration when n is even.

In the remainder, we consider that the system is in an arbitrary configuration if the robots do not form either (1) a regular n-gon, (2) a quasi n-gon, or (3) a strict biangular circle. Let us describe the general scheme provided by Algorithm 1.

Procedure $< A \rightsquigarrow B >$ excluded, the protocol mainly consists of three procedures. The first one, called Procedure $< \texttt{a}Q \rightsquigarrow N\texttt{gon} >$ is used when the system form an aligned quasi n-gon. It leads the system into a regular n-gon. The aim of Procedure $< Q \rightsquigarrow \texttt{a}Q >$ is to transform the cohort from an arbitrary quasi n-gon into an aligned quasi n-gon. The last procedure, Procedure $< B \rightsquigarrow Q >$, is used when the robots form a biangular circle and arranges them into either a regular n-gon or an arbitrary quasi n-gon, depending on the synchrony of the robots. The details of those procedures are given in the remainder of this section.

Let us explain how the procedures are used by giving the overall scheme of Algorithm 1. Starting from an arbitrary configuration, using Procedure $<A \rightsquigarrow B>$, the system is eventually in a biangular circle. If n is odd, then the robots form a regular n-gon and the system is done. Otherwise (n is even), the robots form either a regular n-gon or a strict biangular circle. Starting from the latter case, each robot executes Procedure $<B \rightsquigarrow Q>$. As mentioned above, the resulting configuration can be either a regular n-gon or a quasi n-gon. ¿From a quasi n-gon, the robots execute either Procedure $<\mathtt{a}Q \rightsquigarrow N\mathtt{gon}>$ or Procedure $<Q \rightsquigarrow \mathtt{a}Q>$, depending on whether the quasi n-gon is aligned or not.

Both procedures $<\mathtt{a}Q \rightsquigarrow N\mathtt{gon}>$ and $<Q \rightsquigarrow \mathtt{a}Q>$ require no ambiguity on the concentric configuration forming the quasi n-gon, i.e $n \geq 9$. However, since $<\mathtt{a}Q \rightsquigarrow N\mathtt{gon}>$ and $<Q \rightsquigarrow \mathtt{a}Q>$ are called when n is even only, only the cases $n = 4, 6$ and 8 are not solved by our algorithm. So, in the remainder, we assume that $n \notin \{4, 6, 8\}$. Finally, starting from an aligned quasi n-gon, the resulting configuration of the execution of Procedure $<\mathtt{a}Q \rightsquigarrow N\mathtt{gon}>$ is a regular n-gon. Otherwise, the quasi n-gon becomes aligned by executing Procedure $<Q \rightsquigarrow \mathtt{a}Q>$.

---

n:= the number of robots;
if n is even
then **if** the robots do not form a regular n-gon
  then **if** the robots form a quasi n-gon
  then **if** the robots form an aligned quasi n-gon
    then Execute $<\mathtt{a}Q \rightsquigarrow N\mathtt{gon}>$;
    else Execute $<Q \rightsquigarrow \mathtt{a}Q>$;
  else **if** the robots form a strict biangular circle
    then Execute $<B \rightsquigarrow Q>$;
    else Execute $<A \rightsquigarrow B>$;
else Execute $<A \rightsquigarrow B>$;

Algorithm 1: Procedure $<A \rightsquigarrow N\mathtt{gon}>$ for any r_i in a cohort of n robots ($n \neq 4$, 6, or 8).

Theorem 16 *Procedure $<A \rightsquigarrow N\mathtt{gon}>$ is a deterministic Circle Formation Protocol for any number n of robots such that $n \notin \{4, 6, 8\}$.*

The above theorem follows from Procedure $<A \rightsquigarrow N\mathtt{gon}>$ (Algorithm 1, [Kat05], Lemmas 17, 19, and 22. In the remainder of this section, the procedures and the proofs of the three above lemmas are presented in separate paragraphs.

Procedure $<\mathtt{a}Q \rightsquigarrow N\mathtt{gon}>$. Starting from an aligned quasi n-gon, each robots on C_2 needs to move toward its projection on C_1 whereas it is required that any robot on C_1 remains at the same position because it is located on its projection. This obvious behavior is made of the following single instruction:

$$\text{move to } proj(r_i)$$

Since we have $n \geq 9$ in quasi n-gon, from Lemma 13, the pair (C_1, C_2) is unique. Moreover, it remains unchanged while the regular n-gon is not formed. So, the following result holds:

Lemma 17 *Starting from an aligned quasi n-gon, Procedure $< \mathtt{a}Q \rightsquigarrow N\mathtt{gon} >$ solves the Circle Formation Problem.*

Procedure $< Q \rightsquigarrow \mathtt{a}Q >$. The idea behind Procedure $< Q \rightsquigarrow \mathtt{a}Q >$ consists in changing a quasi n-gon into an aligned quasi n-gon by arranging the robots on C_2 in each sector—refer to Figure 6.

In the following of the paragraph, denote a quasi n-gon by the corresponding pair of concentric circles (C_1, C_2). Two quasi n-gons (C_1^α, C_2^α) and (C_1^β, C_2^β) are said to be *equivalent* if $C_1^\alpha = C_1^\beta$, $C_2^\alpha = C_2^\beta$ and the positions of the robots on C_1^α and C_1^β are the same ones. In other words, the only allowed possible difference between two equivalent quasi n-gons (C_1^α, C_2^α) and (C_1^β, C_2^β) is different positions of robots between C_2^α and C_2^β in each sector.

Procedure $<Q \rightsquigarrow \mathtt{a}Q>$ is shown Algorithm 2. This procedure assumes that the initial configuration is an arbitrary quasi n-gon. In such a configuration, we build, a partial order among the robots on C_2 belonging to a common sector to eventually form an aligned quasi n-gon.

$C_1 :=$ greatest concentric circle; $C_2 :=$ smallest concentric circle;
if r_i are located on C_2
then $MySector :=$ sector wherein r_i is located;
    $PS := FindFinalPos(Mysector)$;
    $FRS :=$ set of robots in MySector which are not located on a position in PS;
    if $FRS \neq \emptyset$
    then $EFR := ElectFreeRobots(FRS)$;
        if $r_i \in EFR$ **then** move to Position $Associate(r_i)$;

Algorithm 2: Procedure $<Q \rightsquigarrow \mathtt{a}Q>$ for any robot r_i in an arbitrary quasi n-gon

---

Let $p_1, \ldots, p_s$ be the final positions on C_2 in the sector S in order to form the aligned quasi n-gon. Let B_1, B_2 the two points located on C_2 at the boundaries of S. Of course, if only one robot is located on C_1 (i.e. there exists only one sector), then $B_1 = B_2$. For each $i \in 1 \ldots s$, p_i is the point on C_2 in S such that $\widehat{B_1 O p_i} = \frac{2k\pi}{n}$, $p_i \neq B_1$ and $p_i \neq B_2$. Clearly, while the distributed system remains in an equivalent quasi n-gon, all the final positions remain unchanged for every robot. A final position p_i, $i \in 1 \ldots s$, is said to be *free* if no robot takes place at p_i. Similarly, a robot r_i on C_2 in S is called a *free* robot if its current position does not belong to $\{p_1, \ldots, p_s\}$.

Define Function $FindFinalPos(S)$ which returns the set of final positions on C_2 in S with respect to B_1. Clearly, in S all the robots compute the same set of final positions, stored in PS. Each robot also temporarily stores the set of free robots in the variable called FRS. Of course, since the robots are oblivious, each active robot on C_2 re-compute PS and FRS each time Procedure $<Q \rightsquigarrow \mathtt{a}Q>$ is executed. Basically, if $FRS = \emptyset$ all the robots occupy a final position in the sector S. Otherwise, the robots move in waves to the final positions in their sector following the order defined by Function $ElectFreeRobots()$. In each sector, the elected robots are the closest free robots from B_1 and B_2. Clearly, the result of Function $ElectFreeRobots()$ return the same set of robots for every robot in the same sector. Also, the number of elected robots is at most equal to 2, one for each point B_1 and B_2. Note that it can be equal to 1 when there is only one free robot, i.e., when only one robot in S did not reach the last free position.

Function $Associate(r)$ assigns a unique free position to an elected robot as follows:
If $ElectFreeRobots()$ returns only one robot r_i, then r_i is associated to the single free remaining position p_i in its sector. This allows r_i to move to p_i. If $ElectFreeRobots()$ returns a pair of robots

$\{r_i, r_{i'}\}$ ($r_i \neq r_{i'}$), then the closest robot to B_1 (respectively, B_2) is associated with the closest position to B_1 (resp., B_2) in S. Note that, even if the robots may have opposite clockwise directions, r_i, $r_{i'}$, and their associated positions are the same for every robot in S.

Lemma 18 *According to Procedure $<Q \rightsquigarrow \mathtt{a}Q>$, if the robots are in a quasi n-gon at time t_j ($j \geq 0$), then at time t_{j+1}, the robots are in an equivalent quasi n-gon.*

Proof. By assumption, at each time instant t_j, at least one robot is active. So, by fairness, starting from a quasi n-gon, at least one robot executes Procedure $<Q \rightsquigarrow \mathtt{a}Q>$. Assume first that no robot executing Procedure $<Q \rightsquigarrow \mathtt{a}Q>$ moves from t_j to t_{j+1}. In that case, since the robots are located on the same positions at t_j and at t_{j+1}, the robots are in the same quasi n-gon at t_{j+1}. Hence, the robots remains in an equivalent quasi n-gon seeing that any quasi n-gon is equivalent to itself. So, at least one robot moves from t_j to t_{j+1}. However, in each sector at most two robots are allowed to move toward distinct free positions on C_2 only inside their sector. Thus, the robots remains in an equivalent quasi n-gon. □

The following lemma follows from Lemma 18 and fairness:

Lemma 19 *Procedure $<Q \rightsquigarrow \mathtt{a}Q>$ is a deterministic algorithm transforming an arbitrary quasi n-gon into an aligned n-gon in finite time.*

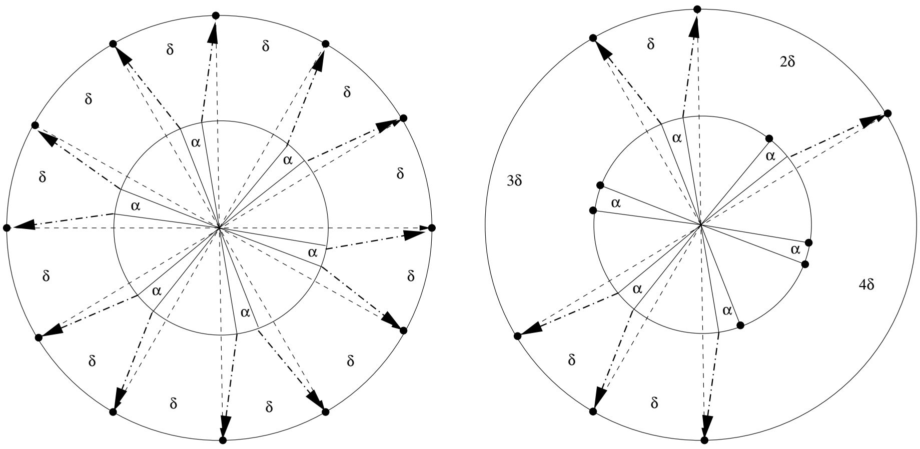
Procedure $<B \rightsquigarrow Q>$. We assume that initially, the robots from a strict biangular circle. In such a configuration, every active robots r_i apply the following scheme:

1. Robot r_i computes the concentric circle C' whose the radius is twice the radius of the strict biangular circle C;
2. Robot r_i considers its neighbor $r_{i'}$ such that $\widehat{r_iOr_{i'}} = \alpha$ and r_i moves away from r'_i to the position $p_i(t_{j+1})$ on C' with an angle equal to $\frac{\pi}{n} - \frac{\alpha}{2}$. More precisely, $\widehat{p_i(t_{j+1})Op_i(t_j)} = \frac{\pi}{n} - \frac{\alpha}{2}$ and $\widehat{p_i(t_{j+1})Op_{i'}(t_j)} = \frac{\pi}{n} + \frac{\alpha}{2}$ —refer to Figure 7.

Let us consider two possible behaviors depending on the synchrony of the robots.

1. Assume that every robot in the strict biangular circle is active at time t_j. In that case, at t_{j+1}, the robots form a regular n-gon—see Case (a) in Figure 7. Indeed, there are two cases:
   (a) If $\widehat{p_i(t_j)Op_{i'}}(t_j) = \alpha$, then $\widehat{p_i(t_{j+1})Op_{i'}}(t_{j+1}) = \alpha + 2(\frac{\pi}{n} - \frac{\alpha}{2})$.
   So, in that case, $\widehat{p_i(t_{j+1})Op_{i'}}(t_{j+1}) = 2\frac{\pi}{n}$.
   (b) If $\widehat{p_i(t)Op_{i'}}(t_j) = \beta$, then $\widehat{p_i(t_{j+1})Op_{i'}}(t_{j+1}) = \beta - 2(\frac{\pi}{n} - \frac{\alpha}{2})$.
   So, in that case, $\widehat{p_i(t_{j+1})Op_{i'}}(t_{j+1}) = \beta - 2\frac{\pi}{n} + \alpha$, which also equal to $\beta - 4\frac{\pi}{n} + \alpha + 2\frac{\pi}{n}$. From Remark 5, we know that $\beta = 4\frac{\pi}{n} - \alpha$. Hence, $\widehat{p_i(t_{j+1})Op_{i'}}(t_{j+1}) = \beta - \beta + 2\frac{\pi}{n}$, which is equal to $2\frac{\pi}{n}$.

   Note (1) the trajectories of the robots do not cross between them, and (2) all the angles α (resp. β) increases up (resp. decrease down) to $\frac{2\pi}{n}$.



(a) If all the robots are active at t_j, then the robots form a regular n-gon at t_{j+1}.

(b) If some robots are inactive at t_j, then the robots form a quasi n-gon at t_{j+1}.

Figure 7: An example showing the principle of Procedure $<B\rightsquigarrow Q>$.

2. Assume that some robots, in the strict biangular circle, are not active at time t_j. In that case, only a subset of robots move toward C' from t_j to t_{j+1}. Then, the robots form a quasi n-gon at time t_{j+1}—see Case (b) in Figure 7. Indeed at t_{j+1}, the robots are in a concentric configuration where C_1 is C' and C_2 is the initial circle C (i.e the biangular circle at time t_j). Furthermore on C_1, the robots form a regular (k, n)-gon where $n - k$ represent the subset of robots which remain inactive at time t_j.

To show that, if the system eventually do not form a regular n-gon, we need to prove that it eventually form a quasi n-gon. Following the above explanations, in remains to show that, in the above second case, the configuration is sliced into sectors at time t_{j+1} such that, in each sector, the missing robots on C_1 are located on C_2.

Lemma 20 *Using Procedure $< B\rightsquigarrow Q >$, if all the robots are in strict biangular circle at time t_j, then the configuration is sliced into sectors at t_{j+1} when the n-gon is not formed.*

Proof. As already stated previously, the robots are in concentric configuration at time t_{j+1}. Moreover, at t_j, the robots are in a strict biangular circle such that $\alpha + \beta = \frac{4\pi}{n}$. Since the biangular circle is strict, without loss of generality, we can assume that $\alpha < \beta$ with $0 < \alpha < \frac{2\pi}{n}$ and $\frac{2\pi}{n} < \beta < \frac{4\pi}{n}$.

Assume, by contradiction, that there exists one robot r_i on C_2 located on the radius passing through any robot $r_{i'}$ on C_1 at t_{j+1}. This implies that at t_j, $\widehat{r_iOr_{i'}} = \frac{\pi}{n} - \frac{\alpha}{2}$, i.e., the angle whose $r_{i'}$ moved away from r_i on C' from t_j to t_{j+1}. Furthermore, at t_j, $r_{i'}$ is active and r_i is inactive. Note that $\widehat{p_i(t_j)Op_{i'}}(t_j)$ is either equal to α or β. Thus, either $\frac{\pi}{n} - \frac{\alpha}{2} = \alpha$ or $\frac{\pi}{n} - \frac{\alpha}{2} = \beta$. However, $\frac{\pi}{n} - \frac{\alpha}{2} < \frac{2\pi}{n}$, and $\frac{2\pi}{n} < \beta < \frac{4\pi}{n}$. Hence, $\frac{\pi}{n} - \frac{\alpha}{2} = \alpha$, and then $\widehat{p_i(t_j)Op_{i'}}(t_j) = \alpha$. By executing Procedure $<B\rightsquigarrow Q>$, $r_{i'}$ moves away from r_i with an angle $\frac{\pi}{n} - \frac{\alpha}{2}$, where $0 < \frac{\pi}{n} - \frac{\alpha}{2} < \frac{2\pi}{n}$. Since r_i is inactive we have $\widehat{p_i(t_{j+1})Op_{i'}}(t_{j+1}) = (\frac{\pi}{n} - \frac{\alpha}{2}) + \alpha$. Furthermore, Procedure $<B\rightsquigarrow Q>$ is called only when $n \geq 9$, and thus, we have $0 < (\frac{\pi}{n} - \frac{\alpha}{2}) + \alpha < \frac{2\pi}{9} + \frac{2\pi}{9} = \frac{4\pi}{9}$ and $0 < \widehat{p_i(t_{j+1})Op_{i'}}(t_{j+1}) < \frac{4\pi}{9}$. Thus, at t_{j+1}, r_i and $r_{i'}$ are not on the same radius. A contradiction. □

Lemma 21 *Using Procedure $<B\rightsquigarrow Q>$, if all the robots form a strict biangular circle at time t_j, then in each sector, the missing robots on C_1 are located on C_2 at t_{j+1} when the n-gon is not formed.*

Proof. Clearly, when all the robots are active and move simultaneously by applying our method, the trajectories do not cross between them (see Figure 7). Assume by contradiction, that at time t_{j+1}, there exists any sector with one extra robot r. If all the robots have been active at time t_j, r would have crossed any other trajectory in order to form a regular n-gon. A contradiction. □

The following lemma directly follows from the algorithm, Lemmas 20 and 21:

Lemma 22 *Procedure $<B\rightsquigarrow Q>$ is a deterministic algorithm transforming a biangular circle into either a regular n-gon or quasi n-gon in finite time.*

5 Concluding Remarks

In this paper, we studied the problem of forming a regular n-gon with a cohort of n robots (CFP). We first shown that it is impossible to obtain a regular n-gon in a deterministic way only by moving the robots along the circle on which all of them take place. Next, we presented a new approach for this problem based on concentric circles formed by the robots. Combined with the solution in [Kat05], our solution works with any number of robots n except if $n = 4$,6 or 8. The main reasons that n must be different from 4, 6 or 8 comes from the fact that the robots may confuse in the recognition of the particular configurations if n is lower than 9. The CFP remains open for these three special cases. In a future work, we would like to investigate CFP in a weakest model such that Corda.

References

[CMN04] I Chatzigiannakis, M Markou, and S Nikoletseas. Distributed circle formation for anonymous oblivious robots. In *3rd Workshop on Efficient and Experimental Algorithms*, pages 159–174, 2004.

[Deb95] X A Debest. Remark about self-stabilizing systems. *Communications of the ACM*, 38(2):115–117, 1995.

[DK02] X Défago and A Konagaya. Circle formation for oblivious anonymous mobile robots with no common sense of orientation. In *2nd ACM International Annual Workshop on Principles of Mobile Computing (POMC 2002)*, pages 97–104, 2002.

[Dol00] S. Dolev. *Self-Stabilization*. The MIT Press, 2000.

[DP06] Y Dieudonné and F Petit. Circle formation of weak robots and Lyndon words. Technical Report TR 2006-05, LaRIA, CNRS FRE 2733, Université of Picardie Jules Verne, Amiens, France, 2006. http://hal.ccsd.cnrs.fr/ccsd-00069724, submitted for publication.

[FPS06] P Flocchini, G Prencipe, and N Santoro. Self-deployment algorithms for mobile sensors on a ring. In *2nd International Workshop on Algorithmic Aspects of Wireless Sensor Networks (Algosensors 2006).*, 2006. To appear.

[FPSW99] P Flocchini, G Prencipe, N Santoro, and P Widmayer. Hard tasks for weak robots: The role of common knowledge in pattern formation by autonomous mobile robots. In *10th Annual International Symposium on Algorithms and Computation (ISAAC 99)*, pages 93–102, 1999.

[Kat05] B Katreniak. Biangular circle formation by asynchronous mobile robots. In *12th International Colloquium on Structural Information and Communication Complexity (SIROCCO 2005)*, pages 185–199, 2005.

[Pre02] G Prencipe. Distributed coordination of a set of autonomous mobile robots. Technical Report TD-4/02, Dipartimento di Informatica, University of Pisa, 2002.

[SS90] K Sugihara and I Suzuki. Distributed motion coordination of multiple mobile robots. In *IEEE International Symosium on Intelligence Control*, pages 138–143, 1990.

[SS96] K Sugihara and I Suzuki. Distributed algorithms for formation of geometric patterns with many mobile robots. *Journal of Robotic Systems*, 3(13):127–139, 1996.

[SY96] I Suzuki and M Yamashita. Agreement on a common x-y coordinate system by a group of mobile robots. *Intelligent Robots: Sensing, Modeling and Planning*, pages 305–321, 1996.

[SY99] I Suzuki and M Yamashita. Distributed anonymous mobile robots - formation of geometric patterns. *SIAM Journal of Computing*, 28(4):1347–1363, 1999.