Squaring the Circle with Weak Mobile Robots

Yoann Dieudonné, Franck Petit

Squaring the Circle with Weak Mobile Robots

Yoannn Dieudonné* and Franck Petit**

*MIS Lab.
Université Picardie Jules Verne
Amiens

**INRIA / LIP Lab. CNRS UMR 5668
Université de Lyon—ENS Lyon
Lyon

We present two non-trivial deterministic protocols that solve the circle formation problem (CFP) with 4 and 3 robots, respectively. Both solutions do not require that each robot reaches its destination in one atomic step. This paper closes CFP for any number n (> 0) of robots in the semi-synchronous model.

Keywords: Distributed Coordination, (Uniform) Circle Formation, Mobile Robot Networks, Self-Deployment

1 Introduction

Consider a distributed system where the computing units are *mobile weak robots* (*sensors* or *agents*), *i.e.*, devices equipped with sensors and designed to move in a two-dimensional plane. By weak, we mean that the robots are *anonymous*, *autonomous*, *disoriented*, and *oblivious*, *i.e.*, devoid of (1) any local parameter (such that an identity) allowing to differentiate any of them, (2) any central coordination mechanism or scheduler, (3) any common coordinate mechanism or common sense of direction, and (4) any way to remember any previous observation nor computation performed in any previous step. Furthermore, all the robots follow the same program (*uniform* or *homogeneous*), and there is no kind of explicit communication medium. The robots implicitly “communicate” by observing the position of the others robots in the plane, and by executing a part of their program accordingly.

In such a weak model, there has been considerable interest in the design of *deterministic* coordination protocols. One of the common features of these works is the study of the minimal level of ability the robots are required to have to achieve the desired task. The *Circle Formation Problem* (CFP) consists in the design of a protocol insuring that starting from an initial arbitrary configuration (where no two robots is at the same position), 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* in finite time.

The first attempt for formally and deterministically solving the CFP were presented in [1]. It works in the semi-synchronous model (SSM) in which the cycles of all the robots are synchronized and their actions are atomic. They ensure only asymptotical convergence toward a configuration in which the robots are uniformly distributed on the boundary of a circle. In other words, the robots move infinitely often and never reach the desired final configuration. The first solution leading n robots in a regular n-gon in finite time is proposed in [4]. Designed for the fully asynchronous model (CORDA), it is also valid in SSM. It works if $n \geq 5$ only. Moreover, if n is even, the robots may form a *biangular circle* in the final configuration, *i.e.*, the distance between two adjacent robots is alternatively either α or β. A general solution is given in [2]. It works in SSM, for any number n of robots, except 3 and 4. The approach in [2] is based on a technique using tools from combinatorics on words and geometric properties of the *convex hull* formed by the robots. Following this work, both cases $n = 4$ and $n = 3$ remain open problems. Indeed, it is very difficult to maintain a geometric invariant with such a few number of robots, *e.g.*, the smallest enclosing circle, concentric cycles, properties of the convex hull, or a leader. As a matter of fact, due to the high rate of symetric configurations, right now, the problem was suspected to be unsolvable with 4 robots.

In this paper, we first disprove this conjecture by presenting a non-trivial deterministic protocol that solves CFP for the case $n = 4$ (Section 3). Next (Section 4), we present a solution for the case $n = 3$. None of the two solutions requires that each robot reaches its destination in one atomic step. Since a cohort of n

robots trivially always form a regular n-gon if $n \in \{1,2\}$, this paper closes the circle formation problem for any number n (> 0) of robots in SSM.

2 Preliminaries

Model. We adopt the semi-synchronous model, below referred to as *SSM*. The *distributed system* considered in this paper consists of n mobile robots. Each robot, viewed as a point in the Euclidean plane, move on this two-dimensional space unbounded and devoid of any landmark. Any robot can observe, compute and move with infinite decimal precision. The robots are equipped with sensors enabling 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 observable 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. At each time instant t_j ($j \geq 0$), each robot r is either *active* or *inactive*. The former means that, during the computation *step* (t_j, t_{j+1}), using a given algorithm, r computes in its local coordinate system a position $p(t_{j+1})$ depending only on the system configuration at t_j, and moves towards $p(t_{j+1})$. In the latter case, r does not perform any local computation and remains at the same position. In every single activation, the maximum distance traveled by any robot r is bounded by σ_r.

Basic Definitions and Properties. Given a set P of $n \geq 2$ points $p_1, p_2, \cdots, p_n$ on the plane, the convex hull of P, denoted $H(P)$ (H for short), is the smallest polygon such that every point in P is either on an edges of $H(P)$ or inside it. Informally, it is the shape of a rubber-band stretched around $p_1, p_2, \cdots, p_n$. The convex hull is unique and can be computed with time complexity $O(n \log n)$ [3].

A convex hull H is called a (*convex*) *quadrilateral* (respectively, *triangle*) if H forms a polygon with four (resp. three) sides (or edges) and vertices (or, corners). In the sequel, we consider convex quadrilaterals (resp. triangle) only. A quadrilateral is said to be *perpendicular* if and only if its diagonals are perpendicular. Otherwise, it is called a *non-perpendicular* quadrilateral.

A triangle is said to be *equilateral* if all its sides are of equal length. An *isosceles* triangle has two sides of equal length. A triangle having all sides of different lengths is said to be *scalene*. A *trapezoid* is a quadrilateral with at least one pair of opposite sides parallel. An *isosceles trapezoid* is a trapezoid whose the diagonals are of equal length. A *parallelogram* is a quadrilateral with both pairs of opposite sides parallel. A *rectangle* is defined as a parallelogram where all four of its angles are right angles. A *square* is a rectangle perpendicular quadrilateral.

3 Four Robots

In this section, we present our algorithm that leads 4 mobile robots to eventually form a square. We refer to Figure 1 to explain our scheme.

Consider the convex hull H formed by the robots on the plane. If the 4 robots belong to the same Line L, then H is reduced to the segment of line linking the 4 points (Figure 1, Case *sL*). Otherwise (the 4 robots are not aligned), there are only two possible forms for H : H forms either a quadrilateral or a triangle. If H forms a triangle, then there is a robot r being located either inside H (Case *nD-T*) or between two of the three corners of the triangle (Cases *nP-D* and *P-D*). In the latter case, three out of the four robots q, r, and s are aligned on a line L (r belonging to the segment $[q,s]$), whereas the fourth robot t does not. Such a configuration is called a (*arbitrary*) *delta*. If the line L' passing through r and t is perpendicular to L, then the delta is said to be *perpendicular* (Case *P-D*).

Our propotol is made of two steps : (1) Starting from an arbitrary configuration, move the robots to eventually form an arbitrary perpendicular quadrilaterals; (2) Starting from an arbitrary perpendicular quadrilateral, the robots eventually form a square.

Indeed, when the convex hull H forms an arbitrary perpendicular quadrilateral (Figure 1, Case *PQ*), the diagonals of H are perpendicular in O. The system eventually forms a square by sliding the closest

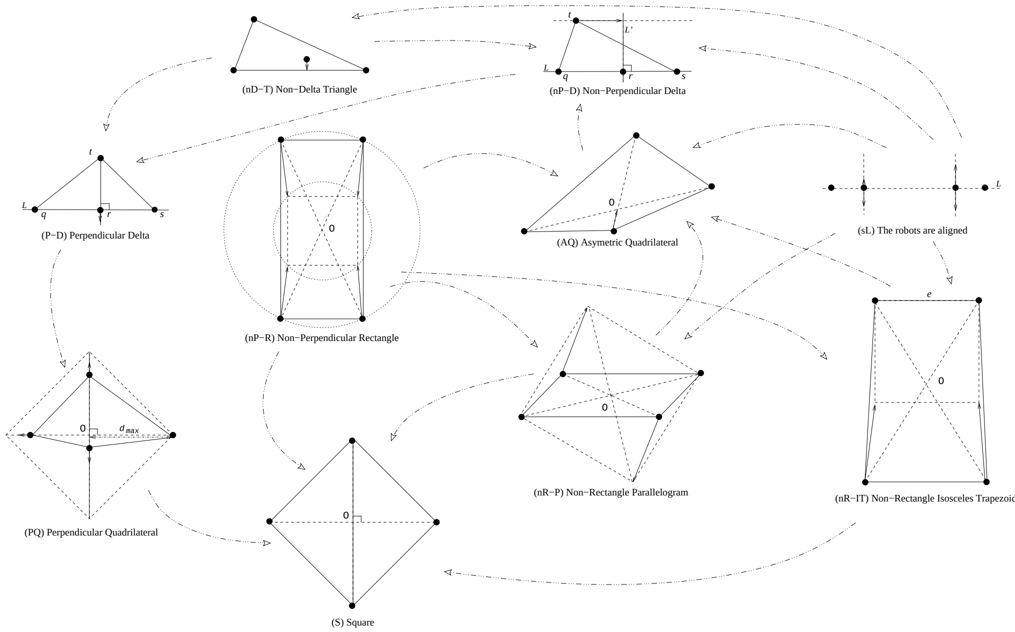

FIG. 1: General Scheme with 4 robots.

robots away from O along their diagonal until they reach the positions being at the same distance from O than the farthest ones (d_{max} in Figure 1). To reach such an arbitrary perpendicular quadrilateral, we aim to bring the system into an arbitrary delta. Starting from a perpendicular delta, Case *P-D* in Figure 1 (q, r, and s are aligned one a line L, the line L' passing through r and t is perpendicular to L), in one step, the system becomes an arbitrary perpendicular quadrilateral by sliding r on L' in the opposite direction of t. Starting from a non-perpendicular delta (Case *nP-D*), the system eventually becomes a perpendicular delta by moving t along L'', the line passing through t that is parallel to L, until L' and L become perpendicular.

Clearly, the above scheme does not cover all the possible cases. In particular, it gives no details about the "arbitrary" configurations considered in the above first item. In fact, we can detail the different classes of such "arbitrary" configurations and the corresponding moves as follows :

1. *The convex hull H forms an arbitrary quadrilateral that is not perpendicular, a rectangle, an isosceles trapezoid, nor a parallelogram.* In the sequel, such a configuration is called an *asymmetric* quadrilateral (Case *AQ*, in Figure 1). In that case, we show that there always exists a robot r being either the unique closest or the unique farthest robot from the center O of the quadrilateral. By moving either r or the opposite robot (w.r.t. to O) along its diagonal toward O, the moving robot eventually reaches O. By the way, it crosses one side of the triangle formed by the 3 other robots. The system then becomes a non-perpendicular delta, and from this point on, adopts the above behavior.

2. *The convex hull H of the* 4 *robots forms a symmetric non-perpendicular quadrilateral that is not reduced to a line segment*. In that case, H forms either an isosceles trapezoid (Case *nR-IT*) or a parallelogram (Case *nR-P*) — note that H can be a non-perpendicular rectangle (Case $nP-R$) if it is both an isosceles trapezoid and a parallelogram. In these cases, the robots move trying to form a square in one step. Clearly, if they move synchronously and reach their respective positions to form a square, then they succeed. Therefore, in every executing starting from either a non-rectangle isosceles trapezoid or a non-rectangle parallelogram, the four robots eventually form either an asymmetric quadrilateral, or a square. Starting from a non-square rectangle, the four robots eventually form either an asymmetric quadrilateral, a non-rectangle isosceles trapezoid, a non-rectangle parallelogram, or a square—refer to Figure 1.

3. *The convex hull H forms a triangle that is not an arbitrary delta*. So, one of the four robots is located inside the triangle (Case *nD-T* in Figure 1). In that case, the robot r inside the triangle moves toward the closest side of the triangle — if r is at the center of the triangle, then it arbitrarily chooses one side to move

on. Again, the system reaches a configuration where the cohort of robots forms a delta.
4. *The* 4 *robots are aligned on the same Line* L (Case sL). In that case, both robots r_1 and r_2 located between the two extremities of the segment formed by the 4 robots are able to move perpendicularly to L. With respect to the asynchrony, there are 5 possible resulting configurations : either a non-perpendicular quadrilateral (possibly, an isosceles trapezoid or a parallelogram) or a triangle (possibly, a non-perpendicular delta).

4 Three Robots

In this section, we show that starting from an arbitrary configuration, 3 robots can form an equilateral triangle in finite time. As for the case $n = 4$, consider the convex hull H formed by the robots on the plane. If the 3 robots belong to the same Line L, then H is reduced to the segment of line linking the 3 points. Otherwise, H forms a non-aligned triangle. (In the following, when we consider a non-aligned triangle, we will omit the term "non-aligned".)

Let us consider the three following cases :
1. *The three robots form an isosceles triangle.* In that case, if the triangle is also equilateral, then the problem is solved. If the triangle is not equilateral, then let r be the unique robot being placed at the unique angle different from the two others robots s and t. Let p be the position of L, the perpendicular bisector of $[s,t]$, such that p, s, and t form an equilateral triangle. Since H forms an isosceles triangle, r belongs to L. So, it can move along L toward p. Clearly, while the triangle is not equilateral — *i.e.*, r does not reach p —, r remains the single robot allowed to move. By fairness, the equilateral triangle is formed in finite time.
2. *The three robots are on the same line L.* Let s and t be the two robots located at the extremities of the segment formed by the three robots. Let r be the median robot and $d(s,r)$ (respectively $d(t,r)$) denotes the distance between s and r (resp., r and t). There are two cases to consider :
a. $d(s,r) = d(t,r)$ — r is located at the middle of $[s,t]$. In that case, r can move on any position on the perpendicular bisector of $[s,t]$. After one step, the robots form an isosceles triangle, and the system behaves as in the previous case.
b. $d(s,r) \neq d(t,r)$. Then, r can move toward the position p such that $d(s,p) = d(t,p)$. Clearly, r reaches p in finite time, and the three robots behaves as above thereafter.
3. *The three robots form a scalene triangle.* Since the three robots form a scalene triangle, the three internal angles are all different. Let r be the robots corresponding to the greatest internal angle. Then, r can move toward the intersection between the opposite side formed by the two others robots and the line passing through r that is perpendicular to the opposite side of the triangle. While the robots are not on the same line, r remains the only robots allowed to move because its internal angle increases whereas the two others internal angles decrease. By fairness, the three robots are eventually on the same line. Then, they behave as above.

5 Conclusion

We closed the circle formation problem for any number $n(n > 0)$ of robots in SSM. We proposed two non-trivial deterministic protocols solving CFP for 4 and 3 robots, respectively. The proposed solutions do not require that each robot reaches its destination in one atomic step. In a future work, we would like to address and solve the problem for any number of robots in CORDA.

Références

[1] X Defago 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.

[2] Y Dieudonné, O Labbani-Igbida, and F Petit. Circle formation of weak mobile robots. *ACM Transactions on Autonomous and Adaptive Systems*, 3(4), 2008.

[3] R.L Graham. An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters*, 1 :132–133, 1972.

[4] 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.