Vol.2015-AL-153 No.16
2015/6/13

Agreement in the Three Dimensional Space: Plane Formation by Synchronous Mobile Robots

Yukiko Yamauchi[1,a)] Taichi Uehara[1,b)] Shuji Kijima[1,c)] Masafumi Yamashita[1,d)]

Abstract: Creating a swarm of mobile computing entities frequently called robots, agents or sensor nodes, with self-organization ability is a contemporary challenge in distributed computing. Motivated by this, this paper investigates the *plane formation problem* for a swarm of robots moving in the three dimensional Euclidean space. Specifically, we examine the existence of a distributed algorithm that works on each robot so that all robots eventually reside in a common plane. Symmetry breaking has been shown to be a key to form an agreement among robots in the two dimensional space. The *symmetricity* of a set of points in the two dimensional space is the order of the cyclic group of the set. In this paper, we generalize the concept of symmetricity to the three dimensional space based on rotation groups, and completely characterize the set of initial configurations P from which a plane is formable in terms of the symmetricity of P. An implication of this characterization is somewhat counter-intuitive: The robots *cannot* form a plane from most of the semi-regular polyhedra, while they *can* from every regular polyhedron (except an icosahedron), which consists of the same regular polygon and contains "more" symmetric robots than semi-regular polyhedra.

1. Introduction

Self-organization in a swarm of mobile computing entities frequently called robots, agents or sensor nodes, has gained much attention as sensing and controlling devices are developed and become cheaper. It is expected that mobile robot systems perform patrolling, sensing, and exploring in a harsh environment such as disaster area, deep sea, and space. For robots moving in the *three dimensional* Euclidean space (3D-space), we investigate the *plane formation problem*, which is a fundamental self-organization problem that requires robots to occupy distinct positions on a common plane from initial positions, mainly motivated by an obvious observation: Robots on a plane would be easier to control than those deployed in 3D-space.

In this paper, a mobile robot system consists of autonomous robots that move in 3D-space, and cooperate with each other to accomplish their tasks without any central control. A robot is represented by a point in 3D-space and repeats executing the "Look-Compute-Move" cycle, during which, it observes, in *Look phase*, the positions of all robots by taking a snapshot, which we call a *local observation* in this paper, computes the next position based only on the snapshot just taken and using a given deterministic algorithm in *Compute phase*, and moves to the next position in *Move phase*. This definition of Look-Compute-Move cycle implies that it has *full vision*, i.e., the vision is unrestricted, the algorithm is *oblivious*, i.e., it does not depend on a snapshot of the past, and the move is an atomic action, i.e., each robot does not stop en route to the next position but we do not care which route it takes. A robot has no access to the global x-y-z coordinate system, and all actions are done in terms of its local x-y-z coordinate system. We assume that it has *chirality*, which means that it has the sense of clockwise and counter-clockwise directions. In particular, we assume that local coordinate systems are right-handed.

The robots can see each other, but do not have direct communication capabilities; communication among robots must take place solely by moving and observing robots' positions, tolerating possible inconsistency among the local coordinate systems. The robots are *anonymous*; they have no unique identifiers and are indistinguishable by their looks, and execute the same algorithm. Finally, they are fully synchronous (FSYNC); they all start the i-th Look-Compute-Move cycle simultaneously, and synchronously execute each of its Look, Compute and Move phases.

The purpose of this paper is to show a necessary and sufficient condition for the solvability of the plane formation problem. The *line formation problem* in the two dimensional Euclidean space (2D-space or plane), is the counter-part of the plane formation problem in 3D-space, and is *unsolvable* from an initial configuration P if P is a regular polygon, intuitively because anonymous robots forming a regular polygon cannot break symmetry among themselves, and lines they propose are also symmetric, so that they cannot agree on one line from them [8]. Hence symmetry breaking among robots would play a crucial role in our study of the plane formation in 3D-space, too.

The *pattern formation problem* requires robots to form a target pattern from an initial configuration, and our plane formation problem is a subproblem of the pattern formation problem in 3D-space. To investigate the pattern formation problem in 2D-space, which contains the line formation problem as a subprob-

[1] Kyushu University, 744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan
[a)] yamauchi@inf.kyushu-u.ac.jp
[b)] uehara@tcslab.csce.kyushu-u.ac.jp
[c)] kijima@inf.kyushu-u.ac.jp
[d)] mak@inf.kyushu-u.ac.jp

lem, Suzuki and Yamashita [8] used the concept of *symmetricity* to measure the degree of symmetry of a configuration consisting of the robots' positions on the plane. *1 Let P be a configuration. Then its symmetricity $\rho(P)$ is the order of the cyclic group of P, where its rotation center o is the center of the smallest enclosing circle of P, if $o \notin P$. That is, its rotational symmetry is $\rho(P)$ and $\rho(P)$ is the number of angles such that rotating P by θ ($\theta \in [0, 2\pi)$) around o produces P itself, which intuitively means that the $\rho(P)$ robots forming a $\rho(P)$-gon in P may not be able to break symmetry among them. However, when $o \in P$, the symmetricity $\rho(P)$ is defined to be 1, independently of its rotational symmetry. This is the crucial difference between the rotational symmetry and the symmetricity and reflects the fact that the robot at o *can* break the symmetry in P by leaving o. Then the following result has been obtained [7], [8], [10]: A target pattern F is formable from an initial configuration P, if and only if $\rho(P)$ divides $\rho(F)$.

In order to investigate the plane formation problem (in 3D-space), we extend the concept of symmetricity defined for points in 2D-space to 3D-space using the concept of rotation group. In 3D-space, rotation groups with a finite order are classified into the cyclic group, the dihedral group, the tetrahedral group, the octahedral group, and the icosahedral group. The cyclic group and the dihedral group are said to be *two-dimensional* (2D), in the sense that the plane formation problem is obviously solvable, since there is a single rotation axis or a single principal rotation axis, and all robots can agree on a plane perpendicular to the axis and containing the center of the smallest enclosing ball of themselves. Then FSYNC robots can easily solve the plane formation problem by moving onto the agreed plane.

The other three rotation groups are defined by the rotations of corresponding regular polyhedra, and these rotation groups are called polyhedral groups. A regular polyhedron consists of regular polygons as its faces and have *vertex transitivity*, that is, there are rotations that replace any two vertices with keeping the polyhedron unchanged as a whole. For example, we can rotate a cube around any axis containing two opposite vertices, any axis containing the centers of opposite faces, and any axis containing the midpoints of opposite edges. For each regular polyhedron, rotations applicable to the polyhedron form a group, and, in this way, the three rotation groups, i.e., the tetrahedral group, the octahedral group and the icosahedral group, are defined. We call them *three-dimensional* (3D) rotation groups.

When a configuration has a 3D rotation group, the robots are not on any plane. In addition, the vertex-transitivity among the robots may allow all of them to have an identical local observation, and the robots may result in an infinite execution, where they keep symmetric movements (in 3D-space), and never agree on a plane. A vertex-transitive point set is in general obtained by specifying a seed point and a set of symmetry operations, which consists of rotations around an axis, reflections for a mirror plane (*bilateral symmetry*), reflections for a point (*central inversion*), and *rotation-reflections* [3]. However, it is sufficient to consider vertex-transitive point sets constructed from transformations that preserve the center of the smallest enclosing ball of robots, and keep Euclidean distance and handedness, in other words, direct congruent transformations, since otherwise, the robots can break the symmetry in a vertex-transitive point set (because they have chirality). Such symmetry operations consist of rotations around some axes. (See e.g., [1], [3] for more detail.)

We define the *symmetricity* of a configuration in 3D-space as the rotation group of the configuration, when we regard the configuration as a set of points (see Section 3 for a formal definition). Let P and $\varrho(P)$ be a set of points in 3D-space and its symmetricity, respectively. Then robots are partitioned into vertex-transitive subsets with symmetricity $\varrho(P)$, so that for each subset, the robots in it can have the same local observation. We call this decomposition $\varrho(P)$-*decomposition* of P. The goal of this paper is to show the following theorem:

Theorem 1 Let $P(0)$ and $\{P_1, P_2, \ldots, P_m\}$ be an initial configuration and the $\varrho(P(0))$-decomposition of $P(0)$, respectively. Then oblivious FSYNC robots can form a plane from $P(0)$ if and only if (i) $\varrho(P(0))$ is a 2D group, or (ii) $\varrho(P(0))$ is a 3D group and there exists a subset P_i such that $|P_i| \notin \{12, 24, 60\}$.

We can rephrase this theorem as follows: Oblivious FSYNC robots cannot form a plane from $P(0)$ if and only if $\varrho(P(0))$ is a 3D group and $|P_i| \in \{12, 24, 60\}$ for each P_i. The impossibility proof is by a construction based on the decomposition of the robots. Obviously $12, 24$ and 60 are the cardinalities of 3D rotation groups, and when a vertex-transitive set has a cardinality in $\{12, 24, 60\}$, the corresponding rotation group enables "symmetric" local coordinate systems that imposes an infinite execution, where the robots' positions keep the axes of the rotation group. We will show this fact by constructing the worst-case local coordinate systems.

Theorem 1 implies the following, which is somewhat counter-intuitive: The plane formation problem is *solvable*, even if $P(0)$ is a regular polyhedron (except an icosahedron), i.e., even if the robots initially occupy the vertices of a regular polyhedron (except a regular icosahedron), while it is *unsolvable* for most of the semi-regular polyhedra.

For the possibility proof, we present a plane formation algorithm that breaks regular polyhedra for solvable cases. In the 2D-space, the symmetricity of a configuration is defined to be 1 when a robot is on the rotation axis of the cyclic group, because the robot on the center can break the symmetry in the configuration by leaving the position. In a similar way, a rotation axis of a 3D group disappears when a robot on it leaves the position. Fortunately, there is always a robot on a rotation axis, if the cardinality of a vertex-transitive robots is not in $\{12, 24, 60\}$ (and we can use it to reduce the number of rotation axes). Although there are multiple rotation axes, we present an algorithm that transforms a configuration that yields a 3D rotation group into another configuration yielding a 2D rotation group, by reducing the number of rotation axes.

Related works. We roughly review some of works on robots in 2D-space, since there is few research on robots in 3D-space, although an autonomous mobile robot system in 2D-space has been extensively investigated (see e.g., [2], [4], [5], [6], [7], [8], [10]). Besides fully synchronous (FSYNC) robots, there are two other

*1 The symmetricity was originally introduced in [9] for anonymous networks to investigate the solvability of some agreement problems.

types of robots, semi-synchronous (SSYNC) and asynchronous (ASYNC) robots. The robots are SSYNC if some robots do not start the i-th Look-Compute-Move cycle for some i, but all of those who have started the cycle synchronously execute their Look, Compute and Move phases [8], and they are ASYNC if no assumptions are made on the execution of Look-Compute-Move cycles [5]. The book by Flocchini et al. [4] contains almost all results on ASYNC robots up to year 2012.

As for the pattern formation problem in 2D-space, which includes the line formation problem as a subproblem, the solvable cases are determined for each of the FSYNC, SSYNC and ASYNC models [7], [8], [10], which are summarized as follows: (1) For non-oblivious FSYNC robots, a pattern F is formable from an initial configuration $P(0)$ if and only if $\rho(P(0))$ divides $\rho(F)$. (2) Pattern F is formable from $P(0)$ by oblivious ASYNC robots if F is formable from $P(0)$ by non-oblivious FSYNC robots, except for F being a point of multiplicity 2.

This exceptional case is called the rendezvous problem. Indeed, it is trivial for two FSYNC robots, but is unsolvable for two SSYNC (and hence ASYNC) robots [8]. Therefore it is a bit surprising to observe that the point formation problem for more than two robots is solvable even for ASYNC robots. The result first appeared in [8] for SSYNC robots and then is extended for ASYNC robots in [2]. As a matter of fact, except the existence of the rendezvous problem, the point formation problem (for more than two robots) is the easiest problem in that it is solvable from any initial configuration $P(0)$, since $\rho(F) = n$ when F is a point of multiplicity n, and $\rho(P(0))$ is always a divisor of n by the definition of the symmetricity, where n is the number of robots.

The other easiest case is a regular n-gon (frequently called the circle formation problem), since $\rho(F) = n$. A circle is formable from any initial configuration, like the point formation problem for more than two robots. Recently the circle formation problem for n robots ($n \neq 4$) is solved without chirality [6].

Organization. After explaining the model and a proof scenario in Section 2, we introduce the symmetricity for points in 3D-space and show some properties of vertex-transitive point sets in Section 3. In Section 4, we then prove Theorem 1. Finally, Section 5 concludes this paper by giving some concluding remarks.

2. Robot Model

Let $R = \{r_1, r_2, \dots, r_n\}$ be a set of n robots represented by points in 3D-space. Without loss of generality, we can assume $n \geq 4$, since all robots are already on a plane when $n \leq 3$. By Z_0 we denote the global x-y-z coordinate system. Let $p_i(t) \in \mathbb{R}^3$ be the position of r_i at time t in Z_0, where $\mathbb{R}$ is the set of real numbers. A *configuration* of R at time t is denoted by $P(t) = \{p_1(t), p_2(t), \dots, p_n(t)\}$. We assume that the robots initially occupy distinct positions, i.e., $p_i(0) \neq p_j(0)$ for all $1 \leq i < j \leq n$. In general, $P(t)$ can be a multiset, but it is always a set throughout this paper since the proposed algorithm avoids any multiplicity. [*2] The robots have no access to Z_0. Instead, each robot r_i has a local x-y-z coordinate system Z_i, where the origin is always its current location, while the direction of each positive axis and the magnitude of the unit distance never change. We assume that Z_0 and all Z_i are right-handed. By $Z_i(p)$ we denote the coordinates of a point p in Z_i.

We investigate fully synchronous (FSYNC) robots in this paper. They all start the t-th Look-Compute-Move cycle simultaneously, and synchronously execute each of its Look, Compute and Move phases. We specifically assume without loss of generality that the $(t+1)$-th Look-Compute-Move cycle starts at time t and finishes before time $t+1$. At time t, r_i (and all other robots simultaneously) looks and obtains a set $Z_i(P(t)) = \{Z_i(p_1(t)), Z_i(p_2(t)), \dots, Z_i(p_n(t))\}$.[*3] We sometimes call $Z_i(P(t))$ the *local observation* of r_i at t. Next, r_i computes its next position using an algorithm ψ, which is common to all robots. Formally, ψ is a total function from $\mathcal{P}_n^3$ to $\mathbb{R}^3$, where $\mathcal{P}_n^3 = (\mathbb{R}^3)^n$ is the set of all configurations (which may contain multiplicities). Finally, r_i moves to $\psi(Z_i(P(t)))$ in Z_i before time $t+1$. An infinite sequence of configurations $\mathcal{E} : P(0), P(1), \dots$ is called an *execution* from an *initial configuration* $P(0)$. Observe that the execution $\mathcal{E}$ is uniquely determined, once local coordinate systems Z_i at time 0, algorithm ψ, and initial configuration $P(0)$ are fixed.

We say that an algorithm ψ *forms a plane* from an initial configuration $P(0) \in \mathcal{P}_n^3$, if, regardless of the choice of initial local coordinate systems Z_i of $r_i \in R$, the execution $P(0), P(1), \dots$ eventually reaches a configuration P_f that satisfies the following three conditions:

(a) P_f is contained in a plane,
(b) $|P_f| = n$, i.e., all robots occupy distinct positions, and
(c) Once the system reaches P_f, the robots do not move anymore.

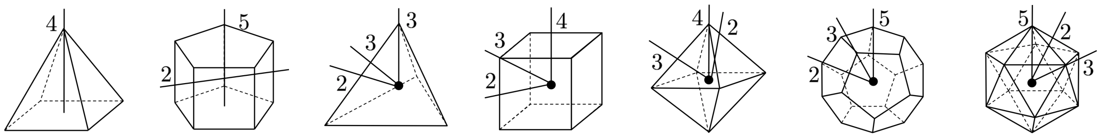
3. Symmetricity in 3D-Space

In 3D-space, we consider the smallest enclosing ball and the convex hull of the positions of robots, i.e., robots are vertices of a convex polyhedron. We do not care for non-convex polyhedra. A *uniform polyhedron* is a polyhedron consisting of regular polygons and all its vertices are congruent. The family of uniform polyhedra contains the regular polyhedra (Platonic solids) and the semi-regular polyhedra (Archimedean solids). Any uniform polyhedron is *vertex transitive*, i.e., for any pair of vertices of the polyhedron, there exists a symmetry operation that moves one vertex to the other with keeping the the polyhedron as a whole.

In general, symmetry operations on a polyhedron consists of rotations around an axis, reflections for a mirror plane (*bilateral symmetry*), reflections for a point (*central inversion*), and *rotation-reflections* [3]. But as briefly argued in Introduction, since all local coordinate systems are right-handed, it is sufficient to consider only direct congruent transformations, and those keeping the center are rotations around some axes that contains the center. We thus concentrate on rotation groups with finite order.

[*2] It is impossible to break up multiple oblivious FSYNC robots (with the same local coordinate system) on a single position as long as they execute the same algorithm, and thus our algorithm is designed to avoid any multiplicity. However, we need to take into account any algorithm that may lead R to a configuration with multiplicities, when proving the impossibility result by reduction to the absurd.

[*3] Since Z_i changes whenever r_i moves, notation $Z_i(t)$ is more rigid, but we omit parameter t to simplify its notation.



(a) (b) (c) (d) (e) (f) (g)

Fig. 1 Rotation groups: (a) the cyclic group C_4, (b) the dihedral group D_5, (c) the tetrahedral group T, (d)(e) the octahedral group O, and (f)(g) the icosahedral group I. Figures show only one axis for each type and its fold.

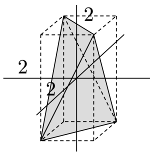


Fig. 2 A sphenoid consisting of 4 congruent isosceles triangles. Its rotation group is D_2. Since the vertices are not placed equidistant positions from the three axes, we can distinguish an axis as the principal axis from the others.

There are five kinds of rotation groups of finite order [1], [3]: The cyclic group C_k with the single k-fold rotation axis ($k \geq 1$), the dihedral group D_ℓ with the single ℓ-fold principal axis and ℓ 2-fold axes ($\ell \geq 2$), the tetrahedral group T, the octahedral group O, and the icosahedral group I. The three groups T, O and I are called *polyhedral groups*. See Figure 3.

In the group theory, we cannot distinguish the principal 2-fold axes of D_2 from the other two 2-fold axes. Since we consider the symmetry of a point set in 3D-space, we add one more rotation group D_2^-, which is essentially D_2, but the robots can distinguish a principal axis from the others by the points' positions (i.e., the robots' positions). Consider a sphenoid consisting of 4 congruent isosceles triangles. Figure 2 illustrates such a sphenoid, in which each of its rotation axes contains the midpoints of opposite edges. Symmetry operations on such a sphenoid is D_2, however we can recognize, for example, the vertical 2-fold axis from the others by their lengths (between the midpoints connecting). Actually, the family of vertex-transitive point sets on which D_2 can act are a line, a square, a rectangle, a regular tetrahedron and the family of such sphenoids. But T can also act on a regular tetrahedron. Thus the point sets which is not contained in a plane and to which only D_2 can act have a primal axis. We use D_2^- to distinguish these cases. Later we will show that the robots can form a plane if they can recognize a single rotation axis or a principal axis. Based on these observations, we say that the cyclic groups C_k and the dihedral groups D_ℓ (including D_2^- and D_2) are *two-dimensional* (2D), while the polyhedral groups are *three-dimensional* (3D) since polyhedral groups cannot act on point sets on a plane. [*4]

We now define the symmetricity of a set of points in 3D-space. Let $\mathbb{S} = \{C_k, D_2^-, D_\ell, T, O, I \mid k = 1, 2, \ldots, and\ \ell = 2, 3, \ldots\}$ be the set of rotation groups, where C_1 is the rotation group with order 1; its unique element is the identity element (i.e., 1-fold rotation). We first define a transitive relation $\succ$ on $\mathbb{S}$. Very intuitively, $A \succ B$ ($A, B \in \mathbb{S}$) means that A has "higher" symmetry than B. Specifically, we define, for all $k(\geq 1)$ and $\ell(\geq 3)$, $C_k \prec D_2^- \prec D_\ell \prec D_2 \prec T \prec O \prec I$, $C_k \prec C_{k+1}$ and $D_\ell \prec D_{\ell+1}$. For any $P \in \mathcal{P}_n^3$, by $B(P)$ and $b(P)$, we denote the smallest enclosing ball of P and its center, respectively. Now the *symmetricity* $\varrho(P)$ of P is defined as follows: If P is on a plane, then $\varrho(P) = \rho(P) \in \{C_k : k = 1, 2, \ldots\}$; otherwise,

$$\varrho(P) = \begin{cases} C_1 & \text{if } b(P) \in P, \\ \text{the rotation group of } P & \text{otherwise.} \end{cases}$$

Recall that $\rho(P)$ is the symmetricity of a point set P in 2D-space. It is worth noting that robots r_i can obviously calculate $\varrho(P)$ from P (more specifically, from its local observation $Z_i(P)$), by checking all rotation axes that keep P unchanged.

A point on the sphere of a ball is said to be *on* the ball, and we assume that the *interior* or the *exterior* of a ball does not include its sphere. When all robots are on $B(P)$, we say the point set (configuration) is *spherical*. We say that a point set P is *vertex-transitive* regarding a rotation group G, if (i) for any two points $p, q \in P$, $g * p = q$ for some $g \in G$, and (ii) $g * p \in P$ for all $g \in G$ and $p \in P$, where $*$ denotes the group action. Note that a vertex-transitive point set is always spherical.

Given a point set P, $\varrho(P)$ determines the arrangement of its rotation axes. We thus use the name of a rotation group and the arrangement of rotation axes interchangeably. We define an embedding of a rotation group to another rotation group. For two groups $G, G' \in \mathbb{S}$, an *embedding* of G to G' is an embedding of each rotation axis of G to one of the rotation axes of G' so that any k-fold axis of G overlaps a k'-fold axis of G' satisfying $k|k'$ with keeping the arrangement of the axes of G, where $a|b$ denotes that b is a multiple of a. For example, we can embed T to O, and T to I, but not O to I. In fact, a group G can be embedded to a group G' if G is a subgroup of G'.

Theorem 2 Let $P \in \mathcal{P}_n^3$ be any configuration. Then P can be decomposed into subsets $P_1, P_2, \ldots, P_m$ in such a way that each P_i is vertex-transitive regarding $\varrho(P)$. Furthermore, the robots can agree on a total ordering among the subsets.

Such a decomposition is unique as a matter of fact, and we call this decomposition of P into $\{P_1, P_2, \ldots, P_m\}$ the $\varrho(P)$-*decomposition* of P.

We will show a sketch of the proof. For any point $p \in P$, let $Orb(p) = \{g * p \in P : g \in \varrho(P)\}$ be the orbit of the group action of $\varrho(P)$ through p. By definition $Orb(p)$ is vertex-transitive regarding $\varrho(P)$. Let $\{Orb(p) : p \in P\} = \{P_1, P_2, \ldots, P_m\}$ be its orbit space. Then $\{P_1, P_2, \ldots, P_m\}$ is obviously a partition (since $p \in Orb(p)$), which satisfies the property of the theorem.

Additionally, by defining an appropriate "local view", robots can agree on the ordering of the subsets. Intuitively, robot r_i translates its local observation $Z_i(P)$ with geocentric longitude,

[*4] Group D_2^- deserves to be called a 2D group. We will justify why we classify D_2 in a 2D group by the end of this section.

Vol.2015-AL-153 No.16
2015/6/13

Table 1 Vertex-transitive point sets in 3D-space (i.e., polyhedra) characterized by rotation group, order, multiplicity and cardinality.

Rotation group	Order	Multiplicity	Cardinality	Polyhedra
D_2	4	1	4	Regular tetrahedron, (Infinitely many sphenoids)
T	12	3	4	Regular tetrahedron
		2	6	Regular octahedron
		1	12	Infinitely many polyhedra
O	24	4	6	Regular octahedron
		3	8	Cube
		2	12	Cuboctahedron
		1	24	Infinitely many polyhedra
I	60	5	12	Regular icosahedron
		3	20	Regular dodecahedron
		2	30	Icosidodecahedron
		1	60	Infinitely many polyhedra

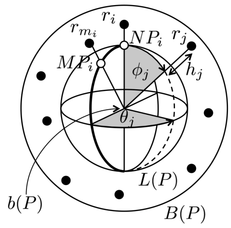


Fig. 3 Amplitude, longitude and latitude calculated from r_i's local observation. The prime meridian for r_i is drawn by bold arc. The position of r_j is now represented by a triple $p_j^* = (h_j, \theta_j, \phi_j)$.

latitude and altitude. The position of a robot $r_j \in R$ is now represented by the *altitude* h_j in $[0, 1]$, *longitude* θ_j in $[0, 2\pi)$, and *latitude* ϕ_j in $[0, \pi]$. (See Figure 3.) Such local view does not depend on a local coordinate system, and we can show the following lemma.

Lemma 3 Let $P \in \mathcal{P}_n^3$ be any configuration, and let $\{P_1, P_2, \ldots, P_m\}$ be the $\varrho(P)$-decomposition of P. Then we have the following two properties:
(1) All robots in P_i have the same local view for $i = 1, 2, \ldots, m$.
(2) Any two robots, one in P_i and the other in P_j, have different local views, for all $i \neq j$.
Then the robots can agree on a total ordering among these subsets using a lexicographic ordering of local views.

We go on the analysis of the structure of a spherical point set that is vertex-transitive regarding a 3D rotation group. (Recall that a vertex-transitive point set is spherical.) Any vertex-transitive (spherical) point set P is specified by a rotation group G and a seed point s as the orbit $Orb(s)$ of the group action of G through s, so that $G = \varrho(P)$ holds. Not necessarily $|G| = |Orb(s)|$ holds. For any $p \in P$, we call $\mu(p) = |\{g \in G : g * s = p\}|$ the *multiplicity* of p. We of course count the identity element of G for $\mu(s)$, and $\mu(p) \geq 1$ holds for all $p \in P$. Then, we have the following property.

Property 4 Let $P \in \mathcal{P}_n^3$ and $\{P_1, P_2, \ldots, P_m\}$ be a configuration and the $\varrho(P)$-decomposition of P, respectively. Then if $\varrho(P)$ is 3D, P_i is one of the polyhedra shown in Table 1 for $i = 1, 2, \ldots, m$.

4. Proof of Theorem 1

This section proves Theorem 1. In Subsection 4.1, we first show the necessity of Theorem 1 by showing that any algorithm for oblivious FSYNC robots cannot form a plane from a configuration if an initial configuration does not satisfy the condition in Theorem 1. In Subsection 4.2, we show the sufficiency by presenting a plane formation algorithm for oblivious FSYNC robots.

4.1 Necessity

Provided $|P| \in \{12, 24, 60\}$, we first show that when a point set P is a vertex-transitive set whose symmetricity is 3D, there is an arrangement of local coordinate system Z_i for each robot $r_i \in R$ such that the execution from P keeps 3D symmetricity forever, no matter which algorithm they obey.

Lemma 5 Assume $n = |R| \in \{12, 24, 60\}$. Then the plane formation problem is unsolvable from an initial configuration $P(0)$ for oblivious FSYNC robots, if $P(0)$ is a vertex-transitive set of points whose symmetricity is 3D.

We will show the sketch of the proof. The idea of the proof is to show that we can construct the local coordinate systems in P that keep the rotation axes of group G forever in the execution of any algorithm, where G is given as follows depending on n:

$$G = \begin{cases} T & \text{if } n = 12, \\ O & \text{if } n = 24, \\ I & \text{if } n = 60. \end{cases}$$

The only case where $G \neq \varrho(P)$ is when $G = T$ and $\varrho(P) \in \{O, I\}$, but we can show that no robot in R is on the rotation axes of G embedded in $\varrho(P)$, and they keep this G. Let $P = \{p_1, p_2, \ldots, p_n\}$ where p_i is be position of $r_i \in R$. We fix a local coordinate system Z_1 arbitrarily for $r_1 \in R$, that is fixed by the origin, $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$ in Z_0. Then, because for each $r_i \in R$ there exists a distinct element $g_i \in G$ such that $p_i = p_1 * g_i$, we obtain the local coordinate system of p_i by the action g_i on Z_1. The local coordinate systems are symmetric regarding G, and any algorithm outputs symmetric next positions that keep the rotation axes of G.

By applying Lemma 5 to each of the subset of $\varrho(P(0))$-decomposition of an initial configuration $P(0)$, we obtain the following theorem.

Theorem 6 Let $P(0)$ and $\{P_1, P_2, \ldots, P_m\}$ be an initial configuration and the $\varrho(P(0))$-decomposition of $P(0)$, respectively. Then the plane formation problem is unsolvable from $P(0)$ for oblivious FSYNC robots, if $\varrho(P(0))$ is 3D, and $|P_i| \in \{12, 24, 60\}$ for $i = 1, 2, \ldots, m$.

4.2 Sufficiency

This subsection proves the following theorem.

Theorem 7 Let $P(0)$ and $\{P_1, P_2, \ldots, P_m\}$ be an initial configuration and the $\varrho(P(0))$-decomposition of $P(0)$, respectively. Then oblivious FSYNC robots can form a plane from $P(0)$ if either (i) $\varrho(P(0))$ is a 2D group, or (ii) $\varrho(P(0))$ is a 3D group and there is a subset P_i such that $|P_i| \notin \{12, 24, 60\}$.

To prove Theorem 7, we designed an algorithm for oblivious FSYNC robots to solve the plane formation problem from an arbitrary initial configuration $P(0)$ that satisfies the conditions in the theorem. The algorithm solves the plane formation problem in at most three rounds, and $P(3)$ is contained in a plane. As mentioned, once the robots have reached an agreement on a common plane, they can move to some points on the plane in a round (since they are FSYNC). An agreement is reached in the second round using a symmetry breaking algorithm after a preparation step in the first round. The condition of the theorem guarantees that there exists P_s with smallest s that satisfies $|P_s| \notin \{12, 24, 60\}$. The first round shrinks P_s with keeping $\varrho(P(0))$ so that robots in P_s become P'_1 in $P(1)$ where $P'_1, P'_2, \ldots, P'_m$ is the $\varrho(P(1))$-decomposition of $P(1)$. The landing algorithm we use in the third round is conceptually easy, but contains some technical subtleties to land the robots to distinct positions on the plane. Basically, the destination of a robot is the foot of the perpendicular line from its current position to the agreed plane. For each P'_i, there are at most two robots with the same destination, however, we can resolve such collision easily intuitively because even when robots have the chirality, they do not agree on the clockwise direction if they are put on a plane with their negative z-axis perpendicular to the plane and pointing to the opposite directions. Then, if two robots have the same foot on the agreed plane, they deterministically choose distinct new destinations. We can assign distinct destinations to the robots by the above collision resolution procedure in the order of $P'_1, P'_2, \ldots, P'_m$.

A very rough idea behind the plane formation algorithm is the following: If $\varrho(P(0))$ is 2D, since there is a single rotation axis or a principal axis, which is obviously recognizable by the robots, they can agree on the plane perpendicular to this axis and containing $b(P(0))$, and indeed the robots can select distinct landing points on the plane.

Suppose otherwise that $\varrho(P(0))$ is 3D. Then there is a P_i such that $|P_i| \notin \{12, 24, 60\}$. That is, $|P_i| \neq |\varrho(P_i)|$, which implies that $|P_i| < |\varrho(P_i)|$, and all robots in P_i are on some rotation axes of $\varrho(P_i)$. We propose a symmetry breaking algorithm that moves the robots so that no robots will be on the rotation axes of $\varrho(P_i)$. This move cannot maintain $\varrho(P_i)$, since otherwise if $\varrho(P_i)$ was maintained at the current configuration, the multiplicity of any point would be 1 regarding $\varrho(P_i)$ (since no robots are on the rotation axes of $\varrho(P_i)$), and thus $|P_i| = |\varrho(P_i)|$ would hold. Specifically, such P_i forms a regular tetrahedron, a cube, a regular octahedron, a regular dodecahedron, or a icosidodecahedron, by Table 1. Our symmetry breaking algorithm breaks the symmetry of these (semi-)regular polyhedral configurations, and as a result a configuration $P(1)$ yields such that $\varrho(P(1))$ is 2D.

Because of the preparation round (preparation round) and the third round (landing round) are simple and straightforward, in this paper we focus on the second round (symmetry breaking round). The algorithm is shown in Algorithm 1.

Algorithm 1 Symmetry breaking algorithm for robot r_i

Notation

P: Current configuration with $\varrho(P) \in \{T, O, I\}$ observed in Z_i
$P_1, P_2, \ldots, P_m$: $\varrho(P)$ decomposition of P where $|P_1| \notin \{12, 24, 60\}$
ϵ: an arbitrarily small distance compared to the distance
between any two centers of the faces of P_1

Algorithm

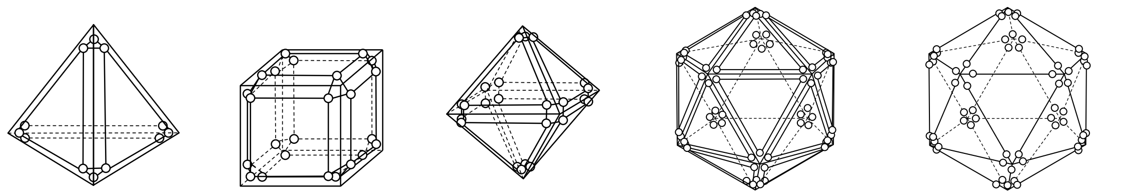
If $p_i \in P_1$ **then**
  If P_1 is an icosidodecahedron **then**
    Select an adjacent regular pentagon face.
    Destination d is the point ϵ before the center of the face
    on the line from p_i to the center.
  Else // P_1 is a regular tetrahedron, a regular octahedron,
    // a cube or a regular dodecahedron.
    Select an adjacent face of the regular polyhedron.
    Destination d is the point ϵ before the center of the face
    on the line from p_i to the center.
  Endif
  Move to d.
Endif

We assume that $\varrho(P)$ is 3D group and and $|P_1| \notin \{1, 12, 24, 60\}$ where $P_1, P_2, \ldots, P_m$ is the ordered $\varrho(P)$-decomposition of configuration P, i.e., $b(P) \notin P$. This is because for any initial configuration P', the robots can trivially translate P' to another configuration P that satisfies the two conditions because the robots are FSYNC. (This preparation is done in the first round.) Remember that for configuration $P = \{p_1, p_2, \ldots, p_n\}$, p_i is the position of $r_i \in R$. Algorithm 1 make robots in P_1 to select one of the adjacent faces and to move toward the center of the face, however robots stop ϵ before the center.

Lemma 8 Let P be a configuration such that $\varrho(P)$ is 3D group and $|P_1| \notin \{1, 12, 24, 60\}$ where $P_1, P_2, \ldots, P_m$ is the ordered $\varrho(P)$-decomposition of P. Then the robots execute Algorithm 1 at P and suppose that a configuration P' yields as the result. Then $\varrho(P')$ is 2D group.

We will show a sketch of the proof. Let $\{P_1, P_2, \ldots, P_m\}$ be the $\varrho(P)$-decomposition of P. Because of the assumption, we have $|P_1| \notin \{1, 12, 24, 60\}$. Thus, as mentioned, P_1 is either a regular tetrahedron, a regular octahedron, a cube, a regular dodecahedron or an icosidodecahedron by Table 1.

In Algorithm 1, only the robots in P_1 move. Each robot $p \in P_1$ selects a face F of P_1 incident on p, and moves to d which is at distance ϵ from the center $c(F)$ of F on line segment $\overline{pc(F)}$, with a restriction that p needs to select a regular pentagon if P_1 is an icosidodecahedron, i.e., when $|P_1| = 30$. Letting D be the set of points consisting of the candidates for d (for $p \in P_1$), D is one of the uniform polyhedra shown in Figure 4. Specifically, Figure 4(a) illustrates an ϵ-cantellated tetrahedron, which corresponds to the candidate set D when P_1 is a regular tetrahedron. Figure 4(b) illustrates an ϵ-cantellated cube, which corresponds to the candidate set D when P_1 is a regular octahedron. Figure 4(c) illustrates an ϵ-cantellated octahedron, which corresponds to the candidate set D when P_1 is a cube. Figure 4(d) illustrates an ϵ-cantellated icosahedron, which corresponds to the candidate set D when P_1 is a regular dodecahedron. Finally, Figure 4(e) illustrates an ϵ-truncated icosahedron, which corresponds to the



(a) ϵ-cantellated tetrahedron (b) ϵ-cantellated cube (c) ϵ-cantellated octahedron (d) ϵ-cantellated icosahedron (e) ϵ-truncated icosahedron

Fig. 4 Candidate set D corresponding to P_1.

candidate set D when P_1 is an icosidodecahedron.

Let $S \subset D$ be any set such that $|S| = |P_1|$. Then it is sufficient to show that $\varrho(S)$ is a 2D group. To derive a contradiction, suppose that there is an S such that $\varrho(S)$ is 3D.

(A) Regular Tetrahedron: See Figure 4(a). If $\varrho(S)$ is 3D, S must be a regular tetrahedron, since $|S| = |P_1| = 4$. Since S is a regular tetrahedron, a point q_F must be selected from each of U_F, where $F \in \mathcal{F}$ and $\mathcal{F}$ is the set of four faces of P_1. By definition $c(\mathcal{F})$ is a regular tetrahedron, and each of its faces contains exactly one element in S, otherwise obviously S would not be a regular tetrahedron. Then we can show the non-existence of a desirable S by checking, for each candidates for S in an exhaustive way, its inconsistency, (e.g., by using a development diagram).

(B) Regular Octahedron: See Figure 4(b). Point set D forms an ϵ-cantellated cube. If $\varrho(S)$ is 3D, because $|S| = 6$, S must be a regular octahedron, since otherwise S was the union of a regular tetrahedron and a 2-set, and $\varrho(S)$ would be 2D. Obviously S cannot be a regular octahedron, since D is an ϵ-cantellated cube and all vertices are around vertices of a cube.

(C) Cube: See Figure 4(c). Point set D forms an ϵ-cantellated octahedron. If $\varrho(S)$ is 3D, because $|S| = 8$, S must contain either a regular tetrahedron, a regular octahedron or a cube as a subset. Since all vertices of D are around vertices of a regular octahedron, S cannot contain a regular tetrahedron and a cube. Furthermore, like (B), S cannot contain a regular octahedron.

(D) Regular Dodecahedron: See Figure 4(d). Point set D forms an ϵ-cantellated icosahedron. If $\varrho(S)$ is 3D, because $|S| = 20$, S must contain either a regular tetrahedron, a regular octahedron or a cube as a subset. Since all vertices of D are around the vertices of a regular icosahedron, S cannot contain a regular tetrahedron, a regular octahedron and a cube.

(E) Icosidodecahedron: See Figure4(e). Point set D forms an ϵ-cantellated icosahedron. If $\varrho(S)$ is 3D, because $|S| = 30$, S must contain a regular tetrahedron, a regular octahedron, a cube, or a regular dodecahedron as a subset. Since all vertices of D are around the vertices of a regular icosahedron, S cannot contain a regular tetrahedron, regular octahedron, cube or a regular dodecahedron.

Now we conclude that $\varrho(S)$ is 2D for any $|P_1|$-subset S of D, which implies that $\varrho(P')$ is 2D. As already mentioned, from a configuration P' robots can agree on a common plane and land distinct points on it. Then, we obtain Theorem 7.

5. Conclusion

In this paper, we have investigated the plane formation problem for anonymous oblivious FSYNC robots in 3D-space. To analyze it, we have defined the symmetricity of a set of points in the space in terms of its rotation group, and we present a necessary and sufficient condition for the plane formation problem.

References

[1] H.S.M. Coxeter, Regular polytopes, Dover Publications, 1973.
[2] M. Cieliebak, P. Flocchini, G. Prencipe, and N. Santoro, Distributed computing by mobile robots: gathering, *SIAM J. of Comput.,* 41, 4, pp.829–879, 2012.
[3] P. Cromwell, Polyhedra, University Press, 1997.
[4] P. Flocchini, G. Prencipe, and N. Santoro, Distributed Computing by Oblivious Mobile Robots, Morgan & Claypool, 2012.
[5] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer, Arbitrary pattern formation by asynchronous, anonymous, oblivious robots, *Theor. Comput. Sci.,* 407, pp.412–447, 2008.
[6] P. Flocchini, G. Prencipe, N. Santoro, and G. Viglietta, Distributed computing by mobile robots: Solving the uniform circle formation problem, *Proc. OPODIS'14*, pp.217-232, 2014.
[7] N. Fujinaga, Y. Yamauchi, H. Ono, S. Kijima, and M. Yamashita, Pattern formation by oblivious asynchronous mobile robots, *Siam J. Comput.* (accepted).
[8] I. Suzuki and M. Yamashita, Distributed anonymous mobile robots: Formation of geometric patterns, *SIAM J. on Comput.*, 28, 4, pp.1347–1363, 1999.
[9] M. Yamashita and T. Kameda, Computing on anonymous networks: Part I-Characterizing the solvable cases, *IEEE Trans. Parallel Distrib. Syst*, 7, 1, pp.69-89, 1996.
[10] M. Yamashita and I. Suzuki, Characterizing geometric patterns formable by oblivious anonymous mobile robots, *Theor. Comput. Sci*, 411, pp.2433–2453, 2010.