



HAL
open science

Plane Formation by Synchronous Mobile Robots in the Three Dimensional Euclidean Space

Yukiko Yamauchi, Taichi Uehara, Shuji Kijima, Masafumi Yamashita

► **To cite this version:**

Yukiko Yamauchi, Taichi Uehara, Shuji Kijima, Masafumi Yamashita. Plane Formation by Synchronous Mobile Robots in the Three Dimensional Euclidean Space. DISC 2015, Toshimitsu Masuzawa; Koichi Wada, Oct 2015, Tokyo, Japan. 10.1007/978-3-662-48653-5_7. hal-01206131

HAL Id: hal-01206131

<https://hal.science/hal-01206131v1>

Submitted on 28 Sep 2015

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

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

Plane Formation by Synchronous Mobile Robots in the Three Dimensional Euclidean Space*

Yukiko Yamauchi **, Taichi Uehara, Shuji Kijima, and Masafumi Yamashita

Kyushu University, Japan.

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* that requires a swarm of robots moving in the three dimensional Euclidean space to reside in a common plane. The robots are fully synchronous and endowed with visual perception. But they have neither identifiers, access to the global coordinate system, any means of explicit communication with each other, nor memory of past. Though there are plenty of results on the agreement problem for robots in the two dimensional plane, for example, the point formation problem, the pattern formation problem, and so on, this is the first result for robots in the *three dimensional space*. This paper presents a necessary and sufficient condition to solve the plane formation problem. An implication of the result 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 a regular icosahedron), which consists of the same regular polygon faces and the robots on its vertices are “more” symmetric than semi-regular polyhedra.

Keywords. Mobile robots in the three dimensional space, plane formation, rotation group, symmetry breaking

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

* This work was supported by a Grant-in-Aid for Scientific Research on Innovative Areas “Molecular Robotics” (No. No. 24104003 and No. 15H00821) of The Ministry of Education, Culture, Sports, Science, and Technology, Japan.

** Corresponding author. Address: 744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan.
Email: yamauchi@inf.kyushu-u.ac.jp

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 and 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 (i.e., positions of the robots), 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 [9]. Hence symmetry breaking among robots would play a crucial role in our study on 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 subproblem, Suzuki and Yamashita [9] 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 of robots on a plane, where we regard the configuration as a set of points. Then its symmetricity $\rho(P)$ is the order of the cyclic group of P ,

¹ The symmetricity was originally introduced in [10] for anonymous networks to investigate the solvability of some agreement problems.

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 regular $\rho(P)$ -gon in P may not be able to break symmetry among themselves. 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, 9, 11]: 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 measure the symmetry of a configuration in 3D-space with the rotation group of the configuration. In 3D-space, rotation groups with finite order are classified into the cyclic groups, the dihedral groups, the tetrahedral group, the octahedral group, and the icosahedral group. The cyclic groups and the dihedral groups 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 single (or principal) 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 polygon faces and has *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 corresponding robots to have an identical local observation, and the robots may result in an infinite execution, where they keep symmetric movements and never agree on a plane. A vertex-transitive set of points 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* [2]. However, it is sufficient to consider vertex-transitive set of points 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 have chirality and can break the symmetry. Such symme-

try operations consist of rotations around some axes. (See e.g., [1, 2] for more detail.)

Let P and $\gamma(P)$ be a set of points in 3D-space and its rotation group, respectively. Then the points (i.e., the robots) are partitioned into vertex-transitive subsets by the group action of $\gamma(P)$. Hence, for each subset, the robots in it may have the same local observation. We call this decomposition $\gamma(P)$ -decomposition of P . The goal of this paper is to show the following theorem:

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

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

We can rephrase this theorem as follows: Oblivious FSYNC robots cannot form a plane from P if and only if $\gamma(P)$ is a 3D rotation 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 orders of 3D rotation groups, and when the cardinality of a vertex-transitive set of points is 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.

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 by leaving the center. In the same way, a rotation axis of a 3D rotation group disappears when a robot on it leaves the axis. 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 in a 3D rotation group, the proposed algorithm transforms a configuration whose rotation group is a 3D rotation group into another configuration whose rotation group is 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., [3–7, 9, 11]). Besides fully synchronous (FSYNC) robots, there are two other 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 [9], 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, 9, 11], which are summarized as follows: (1) For non-oblivious FSYNC robots, a pattern F is formable from an initial configuration P if and only if $\rho(P)$ divides $\rho(F)$. (2) Pattern F is formable from P by oblivious ASYNC robots if F is formable from P 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 [9]. 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 [9] for SSYNC robots and then is extended for ASYNC robots in [3]. 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 , since $\rho(F) = n$ when F is a point of multiplicity n , and $\rho(P)$ 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 in Section 2, we introduce the rotation group of points in 3D-space and show some properties of vertex-transitive set of points in Section 3. In Section 4, we then prove Theorem 1. Finally, Section 5 concludes this paper by giving some concluding remarks. Because of the page limitation, we omit detailed proofs. Please see the full version [12].

2 Robot Model

Let $R = \{r_1, r_2, \dots, r_n\}$ be a set of n anonymous robots represented by points in 3D-space. We use the index just for description. 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 .

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 avoids 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.

Instead, each robot r_i has a local x - y - z coordinate system Z_i , where the origin is always its current location and the direction of x - y - z axes and unit distance are arbitrary. However, we assume that Z_0 and all Z_i are right-handed. By $Z_i(p)$ we denote the coordinate 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 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 initial configuration $P(0)$, local coordinate systems Z_i at time 0, and algorithm ψ are fixed.

We say that an algorithm ψ *forms a plane* from an initial configuration $P(0)$, 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 Symmetry 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* [2]. But as briefly argued in Section 1, since all local coordinate systems are right-handed, it is sufficient to consider only direct congruent transformations, and those keeping the center are rotations

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.

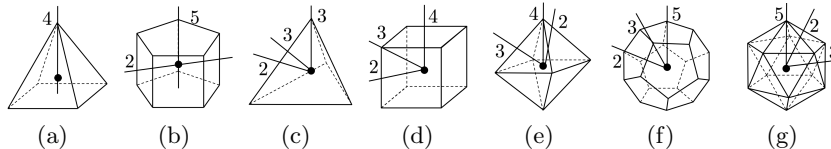


Fig. 1. Rotation groups: (a) the cyclic group C_4 , (b) the dihedral group D_5 , (c) the tetrahedral group T , (d)(e) the octahedral group O , and (f)(g) the icosahedral group I . Figures show only one axis for each fold of axes.

Table 1. Rotation groups T, O and I , and their elements.

Rotation group	2-fold axes	3-fold axes	4-fold axes	5-fold axes	Order
T	3	8	-	-	12
O	6	8	9	-	24
I	15	20	-	24	60

around some axes that contains the center. We thus concentrate on rotation groups with finite order.

A rotation axis is a k -fold axis if the rotation around it is $\pi/k, 2\pi/k, \dots, 2\pi$. There are five kinds of rotation groups of finite order [1, 2]: The cyclic group C_k consists of the single k -fold rotation axis ($k \geq 1$), the dihedral group D_ℓ consists of the single ℓ -fold principal axis and ℓ 2-fold axes ($\ell \geq 2$) perpendicular to the principal axis. The remaining three groups, the tetrahedral group T , the octahedral group O , and the icosahedral group I are called *polyhedral groups*, because they are defined by the rotations of corresponding polyhedra (Figure 1). Table 1 shows for each of the rotation groups T, O , and I , the number of elements around its k -fold rotation axes ($k \in \{2, 3, 4\}$).

In the group theory, we do not distinguish the principal axes of D_2 from the other two 2-fold axes. Consider a sphenoid consisting of 4 congruent isosceles triangles (Figure 2). Rotation operations on such a sphenoid are those of D_2 , however we can recognize, for example, the vertical 2-fold axis from the others by their lengths (between the midpoints connecting). The family of sets of points on which only D_2 can act are lines, rectangles, such sphenoids, and their compositions. Actually, we can easily show that a set of points to which D_2 can act but we cannot distinguish the principal axis have four 3-fold rotation axes, thus T can also act on the set. Hence, the sets of points to which only D_2 can act have the principal axis. 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 this, we say that the cyclic groups and the dihedral groups are *two-dimensional* (2D), while the polyhedral groups are *three-dimensional* (3D) since polyhedral groups cannot act on a set of points on a plane.

Let $\mathbb{S} = \{C_k, D_\ell, T, O, I \mid k = 1, 2, \dots, \text{and } \ell = 2, 3, \dots\}$ 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). When G' is a subgroup of G ($G, G' \in \mathbb{S}$),

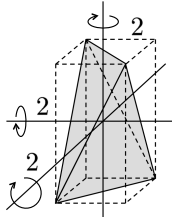


Fig. 2. A sphenoid consisting of 4 congruent isosceles triangles. Its rotation group is D_2 . Since the vertices are not placed equidistant positions from the three axes, we can distinguish an axis as the principal axis from the others.

we denote it by $G' \preceq G$. If G' is a proper subgroup of G (i.e., $G \neq G'$), we denote it by $G' \prec G$. For example, we have $D_2 \prec T$, $T \prec O, I$, and if $G \in \mathbb{S}$ has a k -fold rotation axis, $C_{k'} \preceq G$ if k' divides k .

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. We now define the rotation group of a set of points in 3D-space. For a set of points $P \in \mathcal{P}_n^3$, the rotation group that acts on P and no proper supgroup of it acts on P is uniquely determined. We call such group the *rotation group* of P and denote it by $\gamma(P)$. Hence, even when the points of P are on one plane, its rotation group is chosen from cyclic groups and dihedral groups. For example, the rotation group of four points forming a square is D_4 .⁴ It is worth noting that each robot r_i can obviously calculate $\gamma(P)$ 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. For a set of points P , when all points of P are on $B(P)$, we say P is *spherical*. We say that a set of points 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 set of points is always spherical.

Given a set of points P , $\gamma(P)$ 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 a divides b . For example, we can embed T to O , and T to I , but cannot embed O to I . In fact, group G can be embedded to an arrangement of group G' if $G \preceq G'$.

⁴ This is the major difference between the rotation group of robots in 3D-space and the symmetricity of robots on 2D-plane. In our context, existing works assume that robots agree on the “top” direction against the plane where robots resides and their symmetricity is chosen from cyclic groups [7, 9, 11].

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

Proof. (Sketch.) For any point $p \in P$, let $Orb(p) = \{g * p \in P : g \in \gamma(P)\}$ be the orbit of the group action of $\gamma(P)$ through p . By definition $Orb(p)$ is vertex-transitive regarding $\gamma(P)$. Let $\{Orb(p) : p \in P\} = \{P_1, P_2, \dots, P_m\}$ be its orbit space. Then $\{P_1, P_2, \dots, P_m\}$ is obviously a partition which satisfies the first part of the statement. Such a decomposition is unique as a matter of fact.

Then, we can show that there exists a translation of a local observation of a robot to a “local view” that satisfies the following two properties:

1. All robots in P_i have the same local view for $i = 1, 2, \dots, m$.
2. Any two robots, one in P_i and the other in P_j , have different local views, for all $i \neq j$.

We will show the idea of the translation. Let $L(P)$ be the *largest empty ball* that is centered at $b(P)$, contains no point of P in its interior, and contains at least one point of P on its sphere. Intuitively, the local view of $r_i \in R$ is constructed by considering $L(P)$ as the earth and line $p_i b(P)$ as the earth’s axis, where p_i is the position of r_i . Then, the positions of each robot is represented by its amplitude, longitude, and latitude. This local view does not depend on any local coordinate systems, and each robot can compute the local view of other robots. Then, the robots can agree on the total ordering of the subsets. \square

We call $\{P_1, P_2, \dots, P_m\}$ the $\gamma(P)$ -*decomposition* of P . The robots can agree on the decomposition and the ordering of the subsets, and each robot can recognize which subset it resides. In the following, we assume that $\{P_1, P_2, \dots, P_m\}$ is ordered in this ordering, thus P_1 is on $L(P)$ and P_m is on $B(P)$.

We go on to the analysis of the structure of a set of points that is vertex-transitive regarding a 3D rotation group. Any vertex-transitive (spherical) set of points 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 . Not necessarily $|G| = |Orb(s)|$ holds. For any $p \in P$, we call $\mu(p) = |\{g \in G : g * p = p\}|$ the *multiplicity* of p .⁵ We of course count the identity element of G for $\mu(p)$, and $\mu(p) \geq 1$ holds for all $p \in P$. We can show that the multiplicity of $p \in P$ is identical, and $\mu(p) > 1$ when it is on the $\mu(p)$ -fold rotation axis of G .

For a set of points $P \in \mathcal{P}_n^3$ and its $\gamma(P)$ -decomposition $\{P_1, P_2, \dots, P_m\}$, if $\gamma(P)$ is a 3D rotation group, each P_i is one of the polyhedra shown in Table 2.

4 Proof of Theorem 1

This section proves Theorem 1. In Subsection 4.1, we show the necessity of Theorem 1 by showing that any algorithm for oblivious FSYNC robots cannot form

⁵ The word “multiplicity” is also used for a multiset. Here, the multiplicity of a point p is the size of the stabilizer of G respect to p [8]. Readers can identify the meaning clearly from the context.

Table 2. Vertex-transitive sets of points generated by 3D rotation groups: rotation group, order, multiplicity, and cardinality.

Rotation group	Order	Multiplicity	Cardinality	Polyhedron
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

a plane 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 set of points P is a vertex-transitive set of points regarding a 3D rotation group, there is an arrangement of local coordinate systems of robots forming P such that the execution from P keeps a 3D rotation group forever, no matter which algorithm they obey.

Lemma 1. *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 regarding a 3D rotation group.*

Proof. (Sketch.) The idea of the proof is to show that we can construct local coordinate systems in $P(0)$ that keep the rotation axes of group G forever in the execution of any algorithm, where G is given as follows:

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

We construct a set of symmetric local coordinate systems based on the fact that $P(0)$ is vertex-transitive regarding G . If $G = \gamma(P(0))$, this property clearly holds. The only case where $G \neq \gamma(P(0))$ is when $G = T$ and $\gamma(P(0)) \in \{O, I\}$, but we can show that there exists an embedding of T to $\gamma(P(0))$ such that no robot is on the rotation axes of T . With the fact that $|P(0)| = |T|$, $P(0)$ is vertex-transitive regarding T .

Let $P(0) = \{p_1, p_2, \dots, p_n\}$ where p_i is the 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, the

positions of $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$ of Z_1 in Z_0 . Then, because for each $r_i \in R$ there exists a distinct element $g_i \in G$ such that $p_i = g_i * p_1$, we obtain the local coordinate system of r_i by applying g_i to Z_1 . The local coordinate systems of the robots are symmetric regarding G , local observations of the robots are identical, and the output of the algorithm that the robots execute are identical at the robots, i.e., the destination of robots are symmetric regarding G . After the movement, the positions and local coordinate systems of the robots are still symmetric regarding G . Let $P(1)$ be this new configuration. In the same way, in $P(1)$, the next destinations are symmetric regarding G . In this way, robots repeat symmetric movement regarding G forever and any configuration that appears in the execution keeps G . \square

From Lemma 1, the plane formation problem is unsolvable from each of the semi-regular polyhedra except an icosidodecahedron consisting of 30 robots. Some of the minimum unsolvable instances are a regular icosahedron, a truncated tetrahedron, and a cuboctahedron, each of which consists of 12 robots.

When an initial configuration P is not vertex-transitive, we obtain the following theorem by applying Lemma 1 to each of the subsets of the $\gamma(P)$ -decomposition of P .

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

4.2 Sufficiency

This subsection proves the following theorem by showing a plane formation algorithm for oblivious FSYNC robots.

Theorem 4. *Let P and $\{P_1, P_2, \dots, P_m\}$ be an initial configuration and the $\gamma(P)$ -decomposition of P , respectively. Then oblivious FSYNC robots can form a plane from P if either (i) $\gamma(P)$ is a 2D rotation group, or (ii) $\gamma(P)$ is a 3D rotation group and there exists a subset P_i such that $|P_i| \notin \{12, 24, 60\}$.*

A very rough idea behind the plane formation algorithm is the following: Let $P(0)$ and $\{P_1, P_2, \dots, P_m\}$ be an initial configuration and the $\gamma(P(0))$ -decomposition of $P(0)$.

If $\gamma(P(0))$ is a 2D rotation group, 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 $\gamma(P(0))$ is a 3D rotation group. Then there is a subset P_i such that $|P_i| \notin \{12, 24, 60\}$. That is, $|P_i| < |\gamma(P(0))|$ ($\gamma(P(0)) = \gamma(P_i)$), and all robots in P_i are on some rotation axes of $\gamma(P(0))$. The proposed symmetry breaking algorithm moves the robots of P_i so that none of them will be on any rotation axes of $\gamma(P(0))$. This move cannot maintain $\gamma(P(0))$, otherwise

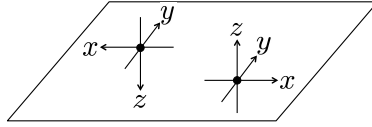


Fig. 3. Two robots on a plane do not agree on the clockwise direction even when they have chirality.

these robots form a vertex-transitive set of points with multiplicity one regarding $\gamma(P(0))$, thus $|P_i| = |\gamma(P(0))|$ which is a contradiction. Specifically, such P_i forms a regular tetrahedron, a cube, a regular octahedron, a regular dodecahedron, or an icosidodecahedron from Table 2. Our symmetry breaking algorithm breaks the symmetry of these (semi-)regular polyhedral configurations, and as a result configuration $P(1)$ yields such that $\gamma(P(1))$ is a 2D rotation group.

In the following, we assume that $b(P(0)) \notin P(0)$ because the robots trivially can translate a configuration $P(0)$ with $b(P(0)) \in P(0)$ to another configuration $P(1)$ with $\gamma(P(1)) = C_1$ by the robot on $b(P(0))$ leaving the center.

The proposed plane formation algorithm consists of three phases. The first phase selects P_s with the smallest index among the subsets whose size is not in $\{12, 24, 60\}$, and shrinks P_s so that it becomes the innermost subset in the next configuration $P(1)$, i.e., only the robots that formed P_s is on $L(P(1))$ and these robots form the same (semi-)regular polyhedron as P_s . This phase is necessary to keep the center of the smallest enclosing circle of robots.

The second phase breaks the (semi-)regular polyhedron formed by the robots on $L(P(1))$ and configuration $P(2)$ yields whose rotation group $\gamma(P(2))$ is a 2D rotation group. We call this phase “go-to-center” phase. Intuitively, this phase makes the robots on $L(P(1))$ select an adjacent face of polyhedron that they form and approach the center, but stop ϵ before the center. We will show that the destinations of robots do not have any 3D rotation group, and the robots succeeds in breaking their symmetry.

Finally, in the third phase, the robots agree on the plane F perpendicular to the single rotation axis (or the principal axis) and containing $b(P(2))$. Then, they land distinct positions of F . Each robot selects the foot of the perpendicular line from its current position to F as its destination. Let $P_1'', P_2'', \dots, P_\ell''$ be the $\gamma(P(2))$ -decomposition of $P(2)$. For each P_i'' , at most two robots select the same destination, however, these robots can easily select new different destinations. Because all local coordinate systems are right-handed, if the negative z -axis of the two robots points to F , the clockwise directions (e.g., rotation from the positive y -axis to positive x -axis) of the two robots are different (Figure 3). By using this property and $b(P(2))$ as a reference point, the two robots select different points on a small circle centered at their common foot as their destinations.

Because of the page restriction, we focus on the second phase. The proposed symmetry breaking algorithm is shown in Algorithm 4.1.

Algorithm 4.1 Symmetry breaking algorithm for robot $r_i \in R$

Notation

P : Current configuration with $\gamma(P) \in \{T, O, I\}$ observed in Z_i .
 $\{P_1, P_2, \dots, P_m\}$: $\gamma(P)$ -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 and determined by using the radius of $B(P)$.
 p_i : Current position of r_i (i.e., the origin).

Algorithm

```
If  $p_i \in P_1$  then
  If  $P_1$  forms an icosidodecahedron then
    Select an adjacent regular pentagon face.
    Destination  $d$  is the point  $\epsilon$  before the center of the face
    on the line from  $p_i$  to the center.
  Else
    //  $P_1$  forms 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  $\epsilon$  before the center of the face
    on the line from  $p_i$  to the center.
  Endif
  Move to  $d$ .
Endif
```

Lemma 2. *Let P be a configuration such that $\gamma(P)$ is 3D rotation group and $|P_1| \notin \{12, 24, 60\}$ where $\{P_1, P_2, \dots, P_m\}$ is the $\gamma(P)$ -decomposition of P . Then the robots execute Algorithm 4.1 at P and suppose that a configuration P' yields as the result. Then $\gamma(P')$ is a 2D rotation group.*

Proof. (Sketch.) Let $\{P_1, P_2, \dots, P_m\}$ be the $\gamma(P)$ -decomposition of P . Because of the assumption, we have $|P_1| \notin \{12, 24, 60\}$. Thus, P_1 is either a regular tetrahedron, a regular octahedron, a cube, a regular dodecahedron or an icosidodecahedron by Table 2.

In Algorithm 4.1, 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 $pc(F)$, with a restriction that p needs to select a regular pentagon if P_1 is an icosidodecahedron. Note that ϵ is common to all robots in P_1 . Then, letting D be the set of points consisting of the candidates for d (for $p \in P_1$), D forms one of the 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.

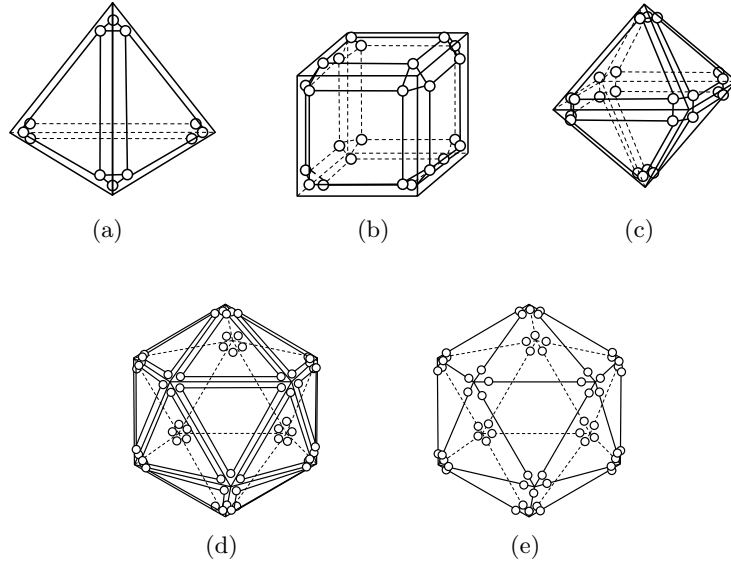


Fig. 4. Candidate set D . White circles are the points of D . (a) ϵ -cantellated tetrahedron, (b) ϵ -cantellated cube, (c) ϵ -cantellated octahedron, (d) ϵ -cantellated icosahedron, and (e) ϵ -truncated icosahedron.

dron, which corresponds to the candidate set D when P_1 is an icosidodecahedron. We would like to emphasize the difference between ϵ -cantellated icosahedron and an ϵ -truncated icosahedron.

Let $S \subset D$ be any set such that $|S| = |P_1|$. Then it is sufficient to show that $\gamma(S)$ is a 2D rotation group. To derive a contradiction, suppose that there is an S such that $\gamma(S)$ is a 3D rotation group. We first claim $b(S) = b(D)$, otherwise $\gamma(S)$ is a 2D rotation group because the intersection of two balls is either a point, a circle, or a ball. For each of the polyhedra that P_1 can be, we can show that $\gamma(S)$ is a 2D rotation group for each of the five cases by contradiction.

First, when P_1 forms a regular tetrahedron, D is an ϵ -cantellated tetrahedron (Figure 4(a)). If $\gamma(S)$ is a 3D rotation group, S must be a regular tetrahedron, since $|S| = |P_1| = 4$. Since S is a regular tetrahedron, one point should be selected from the four faces of P_1 . Then we can show the non-existence of a desirable S by checking, for each candidates for S in an exhaustive way, its inconsistency.

Second, when P_1 forms a regular octahedron, D is an ϵ -cantellated cube (Figure 4(b)). If $\gamma(S)$ is a 3D rotation group, because $|S| = 6$, S must be a regular octahedron, since otherwise S was the union of a regular tetrahedron and a 2-set, and $\gamma(S)$ would be a 2D rotation group. Obviously S cannot be a regular octahedron, since D is an ϵ -cantellated cube and all vertices are around vertices of a cube.

The above two cases show the two basic techniques that we use for the remaining three cases. When P_1 forms a cube, a regular dodecahedron, or an icosidodecahedron, S must contain either a regular tetrahedron, a regular octahedron or a cube as a subset. However, we can show that D does not contain any of these regular polyhedra and we conclude that $\gamma(S)$ is a 2D rotation group for any $|P_1|$ -subset of D , which implies that $\gamma(P')$ is a 2D rotation group. \square

Finally, as already mentioned, from a configuration P' , robots can agree on a common plane and land distinct points on it if $\gamma(P')$ is 2D rotation group. Consequently, we obtain Theorem 4.

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 used the rotation group of a set of points in 3D-space, and presented a necessary and sufficient condition for the plane formation problem. Since real systems work in a 3D-space, many natural problems would arise from practical applications.

References

1. H.S.M. Coxeter, Regular polytopes, Dover Publications, 1973.
2. P. Cromwell, Polyhedra, University Press, 1997.
3. 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.
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.*, 44, 3, pp.740–785m 2015.
8. J.J. Rotman, An introduction to the theory of groups (4th ed.), Springer-Verlag New York, Inc., 1994.
9. I. Suzuki and M. Yamashita, Distributed anonymous mobile robots: Formation of geometric patterns, *SIAM J. on Comput.*, 28, 4, pp.1347–1363, 1999.
10. 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.
11. M. Yamashita and I. Suzuki, Characterizing geometric patterns formable by oblivious anonymous mobile robots, *Theor. Comput. Sci.*, 411, pp.2433–2453, 2010.
12. Y. Yamauchi, T. Uehara, S. Kijima, and M. Yamashita, Plane formation by synchronous mobile robots in the three dimensional Euclidean space, *arXiv:1505.04546*, 2015.