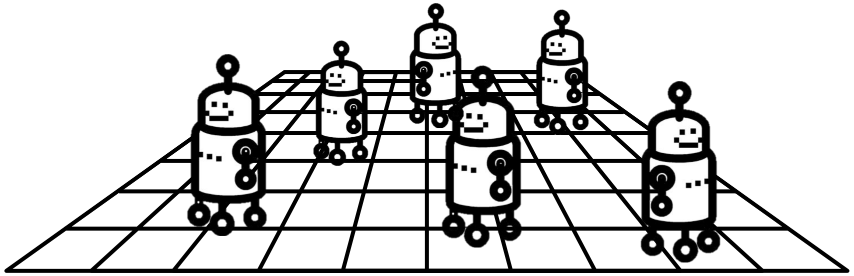
Formation Problems for Synchronous Mobile Robots in the Three-Dimensional Euclidean Space

Yukiko Yamauchi*, Taichi Uehara, and Masafumi Yamashita
(Kyushu University, Japan)

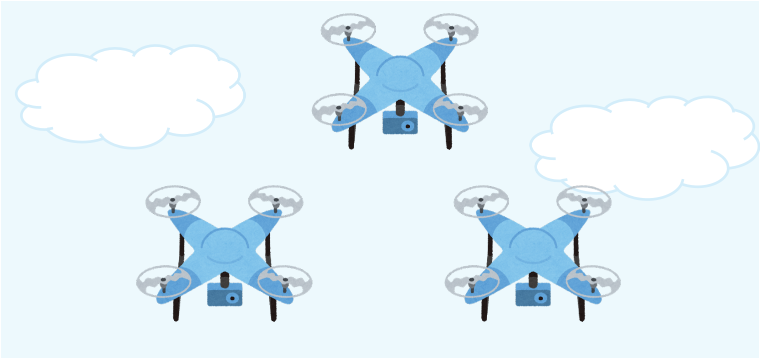
Oct. 22, 2015 GRASTA-MAC 2015@Montreal

Autonomous mobile robot system

- Each robot autonomously moves
  - Observes the positions of other robots
  - Computes next position
  - Then moves



On a 2D-space

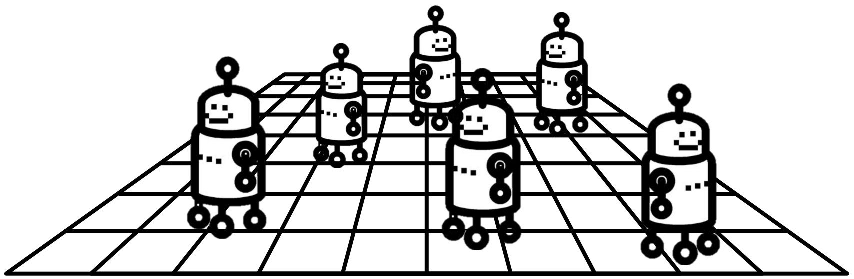


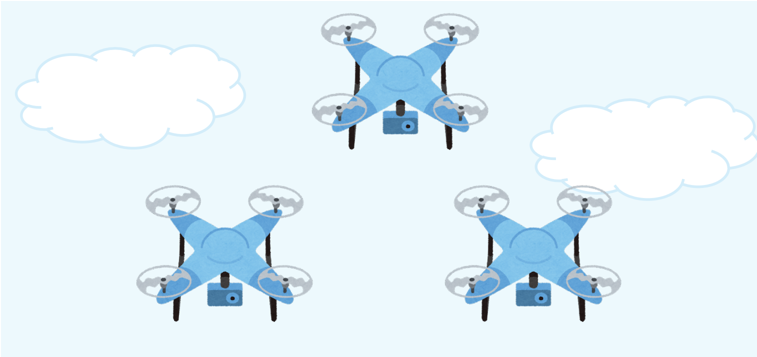
In a 3D-space

- Applications
  - Mobile networks, drones, molecular robots, etc.

Autonomous mobile robot system

- Each robot autonomously moves
  - Observes the positions of other robots
  - Computes next position
  - Then moves



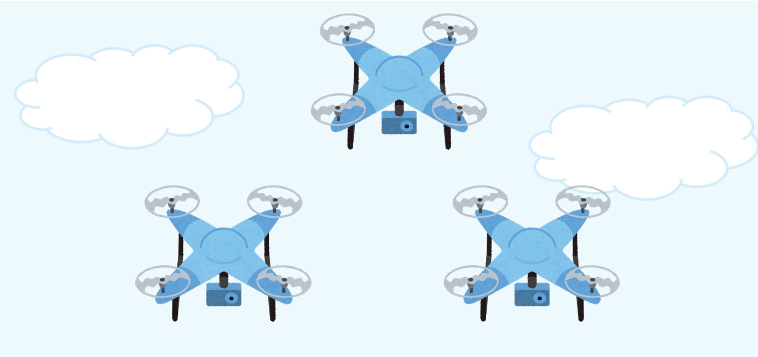


Can we design **distributed algorithm** for robots automatically cooperate?

Gathering, formation, exploration, marching, etc.

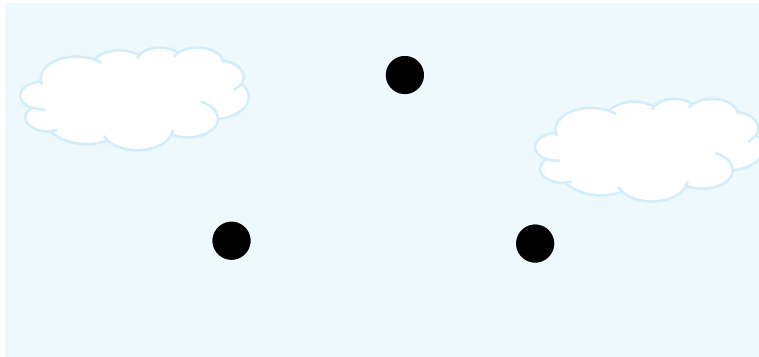
Robot model

- Each robot is a point in a 3D-space
  - NO ID (i.e., anonymous)
  - NO global coordinate system (like GPS)
  - NO communication
  - NO memory of past



Robot model

- Each robot is a point in a 3D-space
  - NO ID (i.e., anonymous)
  - NO global coordinate system (like GPS)
  - NO communication
  - NO memory of past



✓**Observation may be inconsistent**

Robots may not agree vertical axis

✓**Communication = observation of positions**

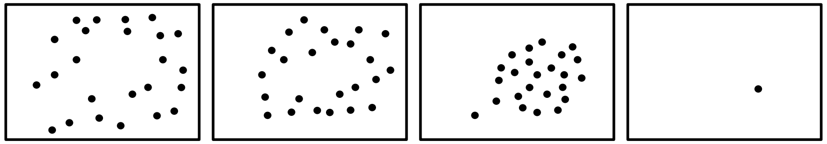
Formation problems

[Suzuki and Yamashita, 1999],
[Flocchini et al., 2008],
[Yamashita and Suzuki, 2010],
[Cieliebak et al., 2012],
[Fujinaga et al., 2015], etc.

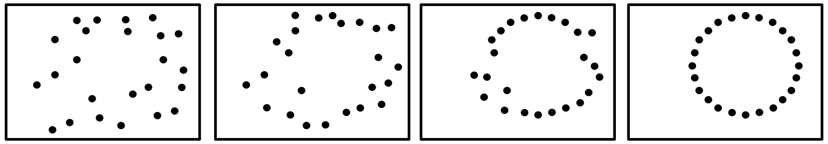
- Robots form a specific pattern

2D-space

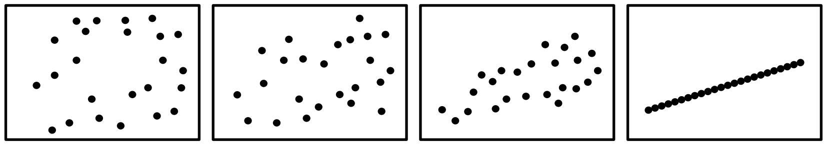
Point formation



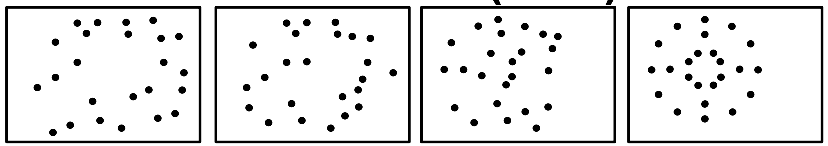
Circle formation



Line formation



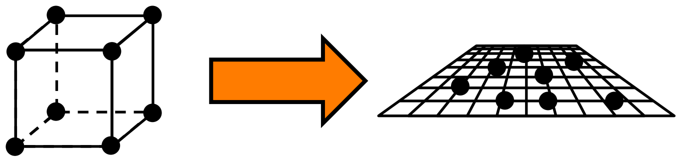
Pattern formation (PTF)



3D-space

[Y. et al., 2015]

Plane formation (PLF)



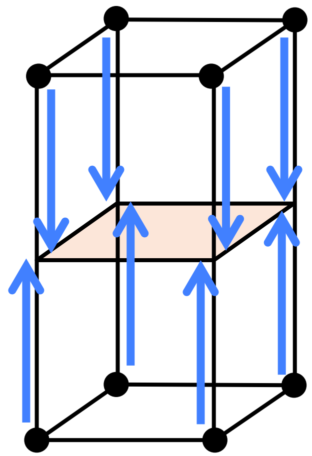
Robots land on a common plane without making any multiplicity

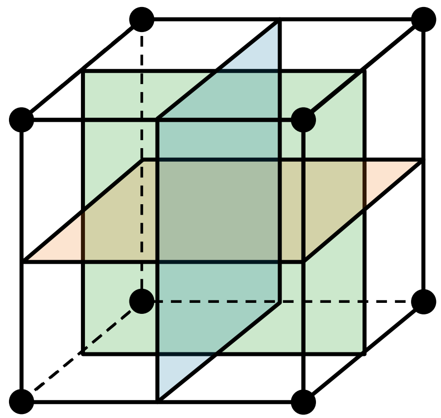
Our results

- Characterization of formation problem in 3D-space
  - Plane formation problem (Rephrase [Y. et al., 2015])
  - Pattern formation problem

Example: Plane formation problem

Consider a convex polyhedron formed by robots …

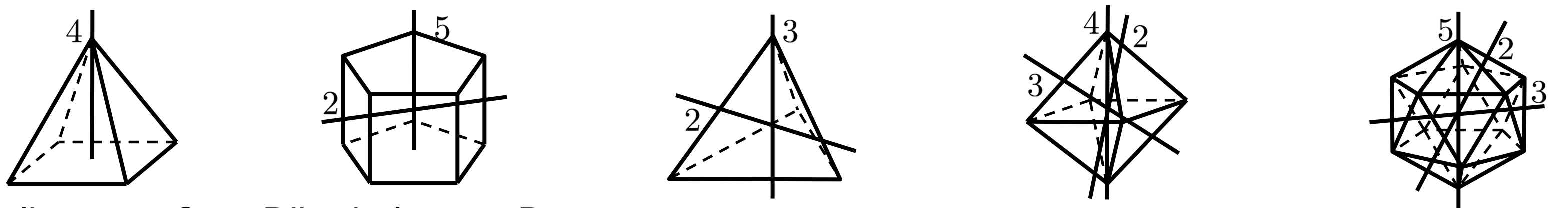




Symmetry among robots is the main difficulty

Main result

- We measure **symmetry** of robots by rotation groups
  - $\rho(P)$: a set of rotation groups that robots cannot eliminate where P is the positions of robots



Cyclic group C_k Dihedral group D_k Tetrahedral group T Octahedral group O Icosahedral group I

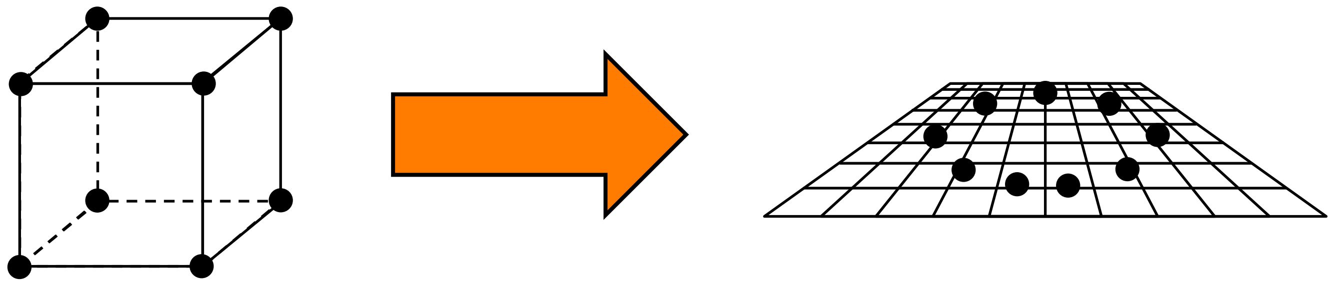
Theorem 1. Let P be an initial configuration of robots. Then oblivious fully-synchronous robots can form a plane from P if and only if $\rho(P)$ consists of cyclic and dihedral groups.

Theorem 2. Let P be an initial configuration of robots. Then oblivious fully-synchronous robots can form a target pattern F from P if and only if $\rho(P)$ is a subset of $\rho(F)$.

Theorem 1

Theorem 1. Let P be an initial configuration of robots. Then oblivious fully-synchronous robots can form a plane from P if and only if $\rho(P)$ consists of cyclic and dihedral groups.

- Intuitively …
  - Robots can form a plane from a cubic configuration



  - From initial configuration forming a uniform polyhedron

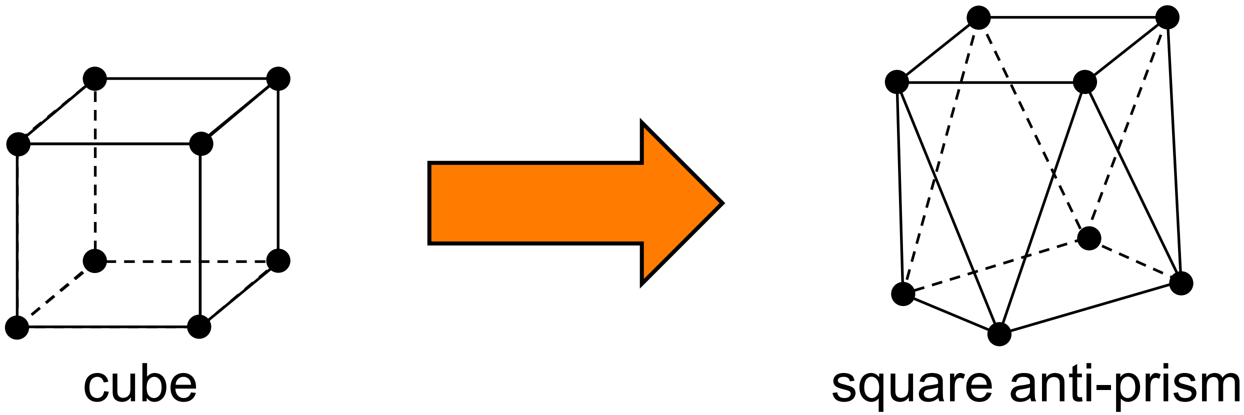
5 Regular polyhedra

13 Semi-regular polyhedra

Theorem 2

Theorem 2. Let P be an initial configuration of robots. Then oblivious fully-synchronous robots can form a target pattern F from P if and only if $\rho(P)$ is a subset of $\rho(F)$.

- Intuitively …
  - Robots can form a square anti-prism from a cube

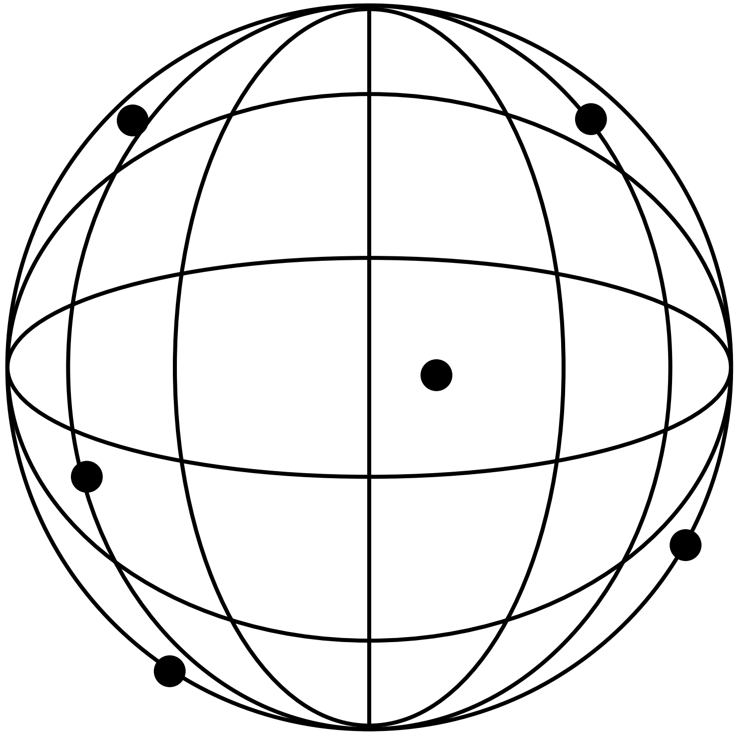


✓Robots can break the symmetry of a cube

System model

Robot

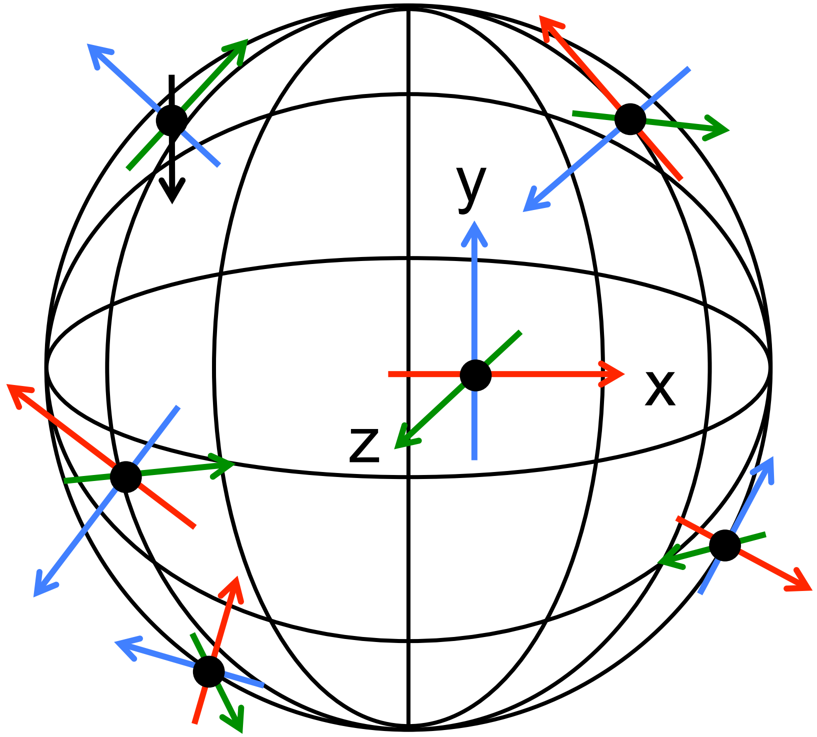
- **Anonymous** point in 3D-space
- **Synchronously** repeats a **Look-Compute-Move cycle**
  - Robots are fully-synchronous (FSYNC)



Robots may reside
in the interior of their
smallest enclosing ball

Look phase

- Each robot observes the positions of other robots in its **local x-y-z coordinate system**
  - Arbitrary direction and unit distance
  - Right handed system with the origin being its current position



Compute phase

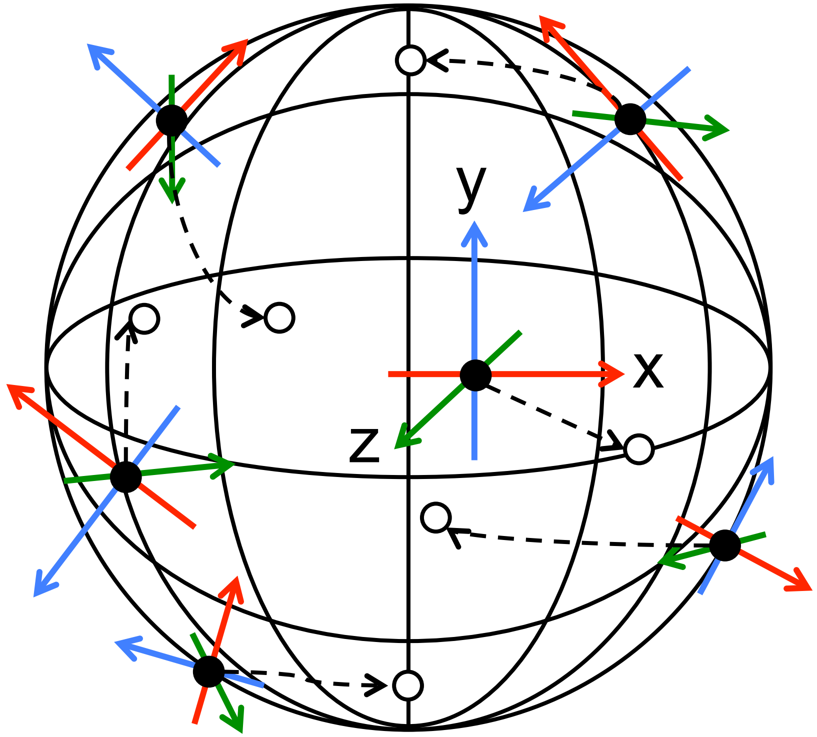
- Each robot computes next position with a **common algorithm**

- We consider **<u>oblivious</u>** algorithms
  - Input is the observation of the current phase
  - Does not use any memory of past
    - Past observations, computation, etc.

  - Important for **fault-tolerance**
    - Robots can tolerate memory failure (self-stabilization)

Move phase

- Each robot moves to the new position
  - Without stopping en route (i.e., rigid movement)



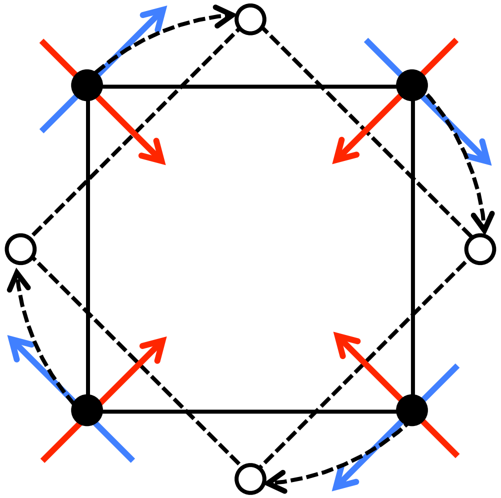
Symmetricity in 3D-space

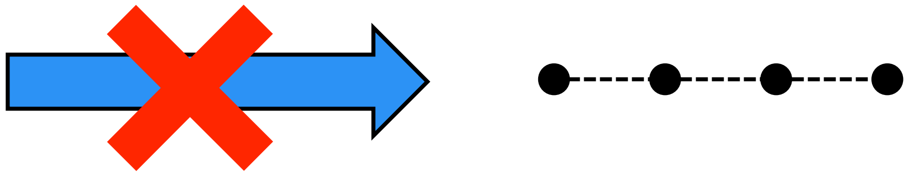
Symmetry among robots

Example. Robots on a plane

- Positions
- Local coordinate systems
- Local observations

Common algorithm outputs symmetric positions



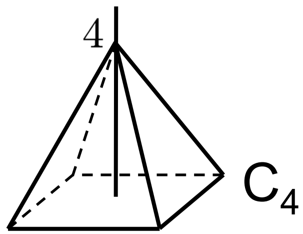


Robots cannot break rotational symmetry

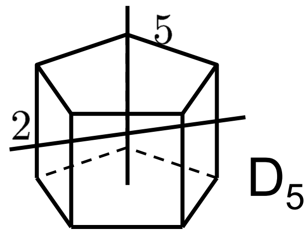
Known for 2D-space e.g. [Suzuki and Yamashita, 1999], [Yamashita and Suzuki, 2010], [Fujinaga et al., 2015]

Rotation groups in 3D-space

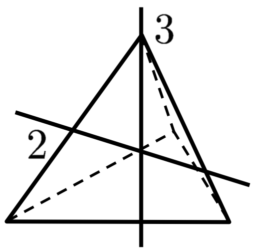
- Rotations form 5 kinds rotation groups
  - Defined by set of rotation axes and their arrangement



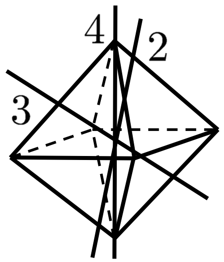
Cyclic group C_k
(Order k)



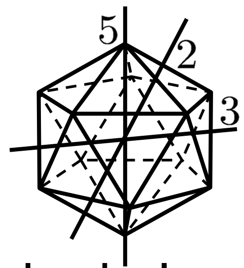
Dihedral group D_k
(Order 2k)



Tetrahedral group T
(Order 12)



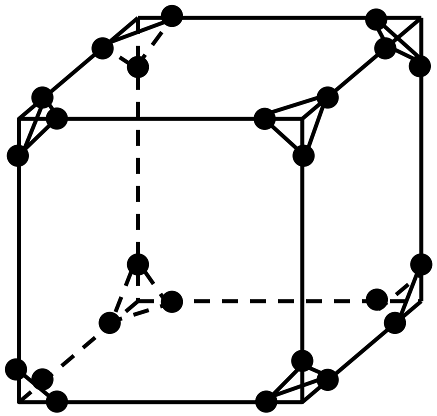
Octahedral group O
(Order 24)



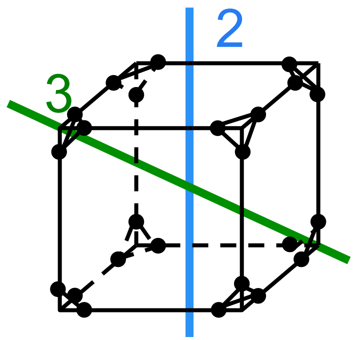
Icosahedral group I
(Order 60)

Rotation group of robots' positions

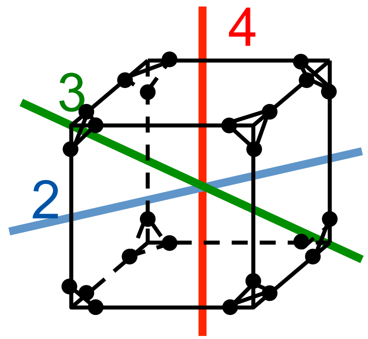
- **Rotation group** $\gamma(P)$ of set of points P
  - Rotation group acts on P
  - No supergroup of $\gamma(P)$ acts on P



Configuration P



T acts on P

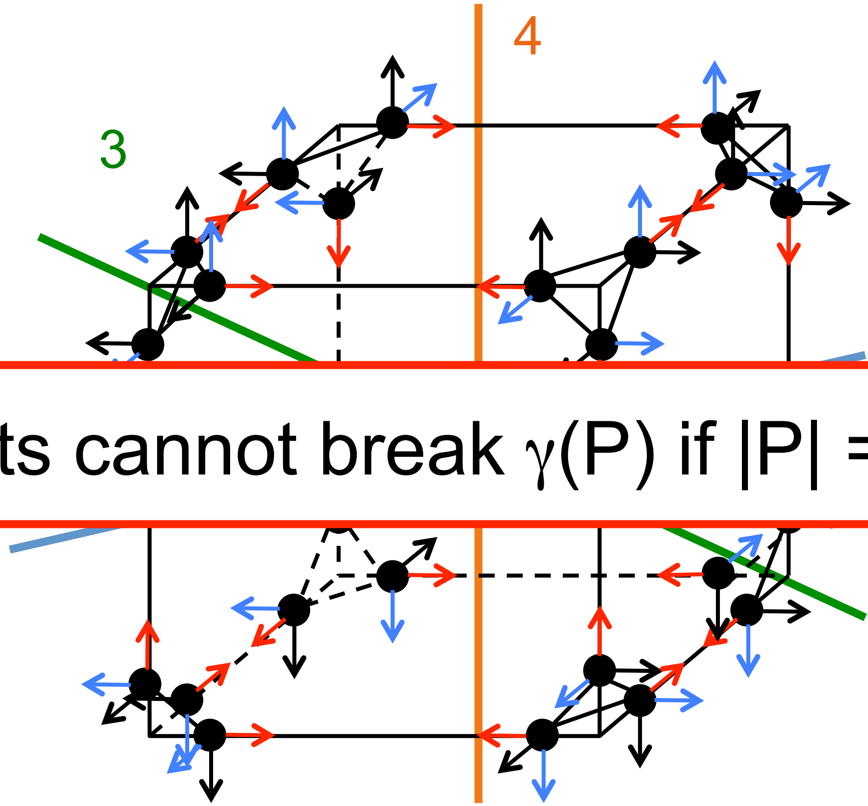


O acts on P

$\gamma(P)$ = O

Generating symmetric situation

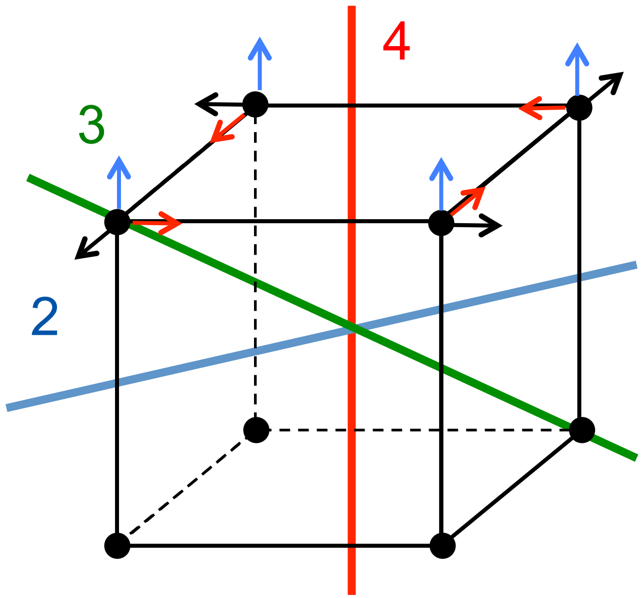
- For a set of points P
  - Select a local coordinate system of one robot
  - Apply each rotation of $\gamma(P)$



Robots cannot break $\gamma(P)$ if $|P| = |\gamma(P)|$.

Generating symmetric situation (Contd.)

- For a set of points P
  - Select a local coordinate system of one robot
  - Apply each rotation of $\gamma(P)$

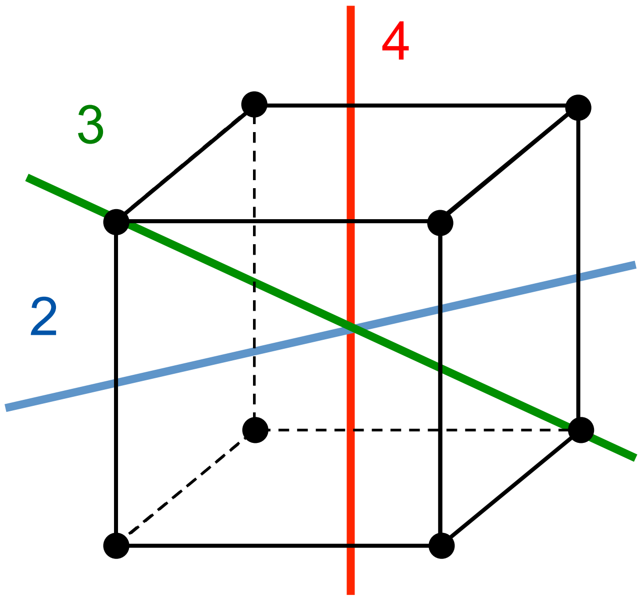


$\gamma(P)=O$

When a rotation axis is **occupied** by a robot, $\gamma(P)$ **cannot generate** symmetric local coordinate systems.

Symmetricity of robots' positions

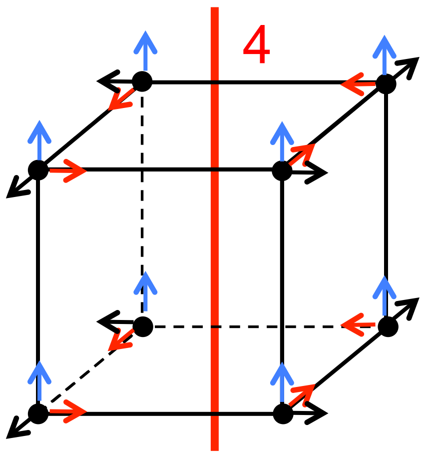
- **Symmetricity** $\rho(P)$ of a set of points
  - The set rotation groups formed by unoccupied rotation axes of $\gamma(P)$



$\gamma(P) = O$

$\rho(P) = \{C_1, C_2, C_4, D_2, D_4\}$

We can generate symmetric situation for any G in $\rho(P)$



✓ **Robots cannot break any G in $\rho(P)$**

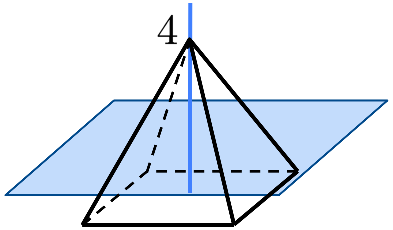
23

Plane formation problem

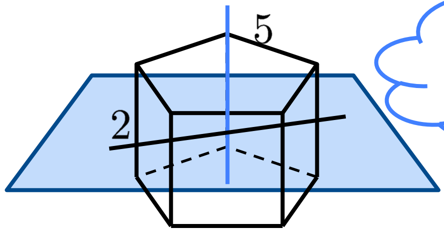
Rotation groups in 3D-space

- Rotations form 5 kinds rotation groups
  - Defined by set of rotation axes and their arrangement

2D groups



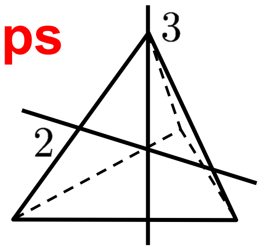
Cyclic group C_k
(Order k)



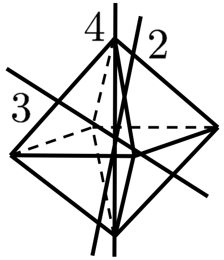
Dihedral group D_k
(Order 2k)

PLF is easy!

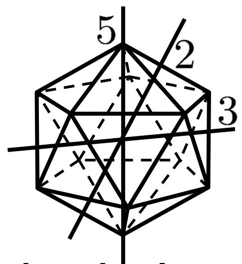
3D groups



Tetrahedral group T
(Order 12)



Octahedral group O
(Order 24)



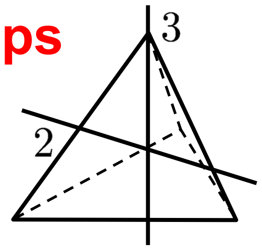
Icosahedral group I
(Order 60)

Rotation groups in 3D-space

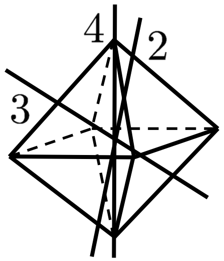
- Rotations form 5 kinds rotation groups
  - Defined by set of rotation axes and their arrangement

Lemma 1.
Let P be an initial configuration of robots. Then oblivious FSYNC robots cannot form a plane from P if $\rho(P)$ contains 3D-rotation group.

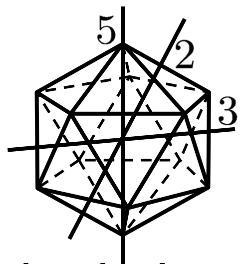
3D groups



Tetrahedral group T
(Order 12)



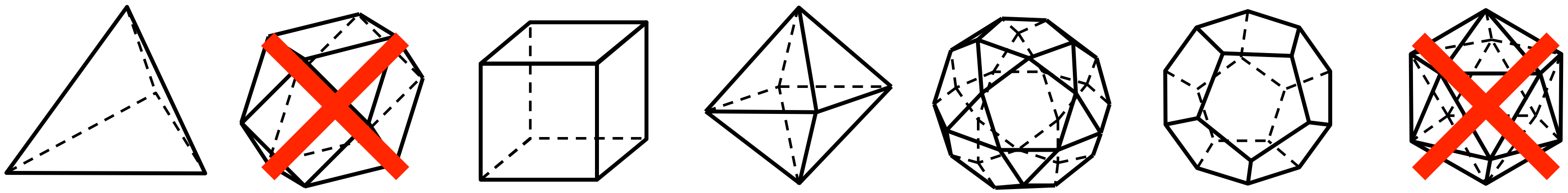
Octahedral group O
(Order 24)



Icosahedral group I
(Order 60)

Seven polyhedra

- Points on rotation axis of T, O, or I form 7 polyhedra
  - Their symmetricity consists of proper subgroup of their rotation group

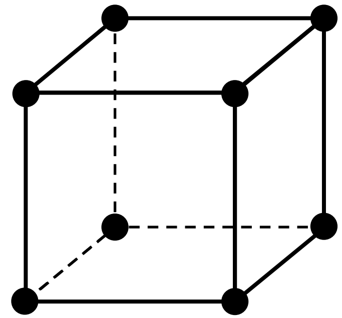


Theorem 1.
Let P be an initial configuration of robots. Then oblivious FSYNC robots can form a plane from P
if and only if $\rho(P)$ consists of cyclic and dihedral groups.

✓ Robots **can** break 3D rotation group of 5 polyhedra.
✓ Robots **cannot** break 3D rotation group of 2 polyhedra.

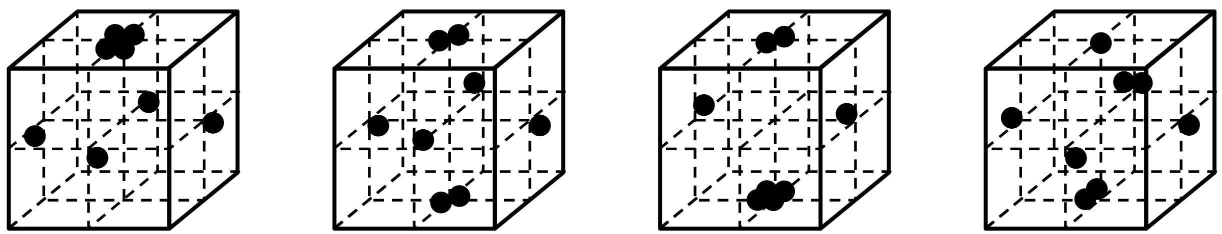
Symmetry breaking

- Example: Cube
  - 8 vertices (robots)
  - 6 faces
  - If 8 robots select a face, it is not uniform



Go-to-center algorithm

Robot selects an adjacent face and approaches the center, but stops ε before the center.

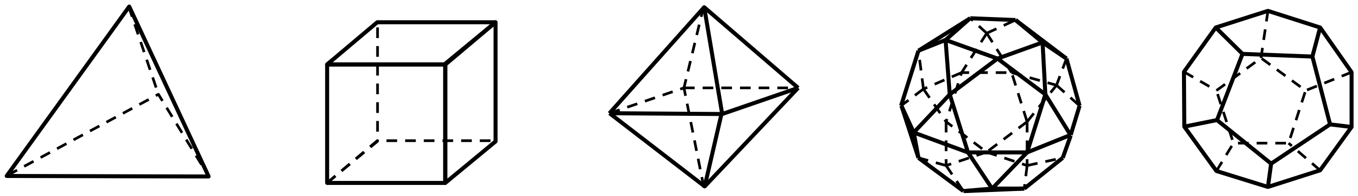


Any resulting configuration P' satisfies γ(P') is in ρ(P)

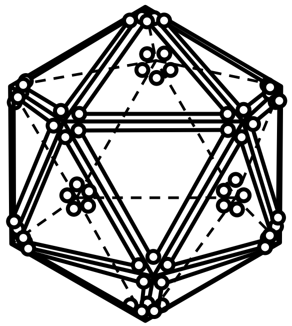
Go-to-center algorithm

Go-to-center algorithm

Robot selects an adjacent face and approaches the center, but stops ε before the center.



- **Any resulting configuration has rotation group in ρ(P)**
  - From the candidates of destinations, check all possible choices
  - E.g. For regular dodecahedron, 60 candidate destinations form ε-cantellated icosahedron

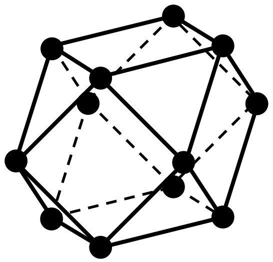


Two exceptions

Cuboctahedron

$\gamma(P) = O$

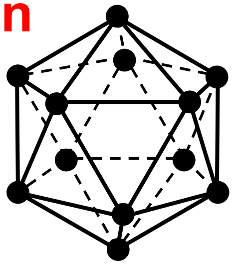
$|P|=12$

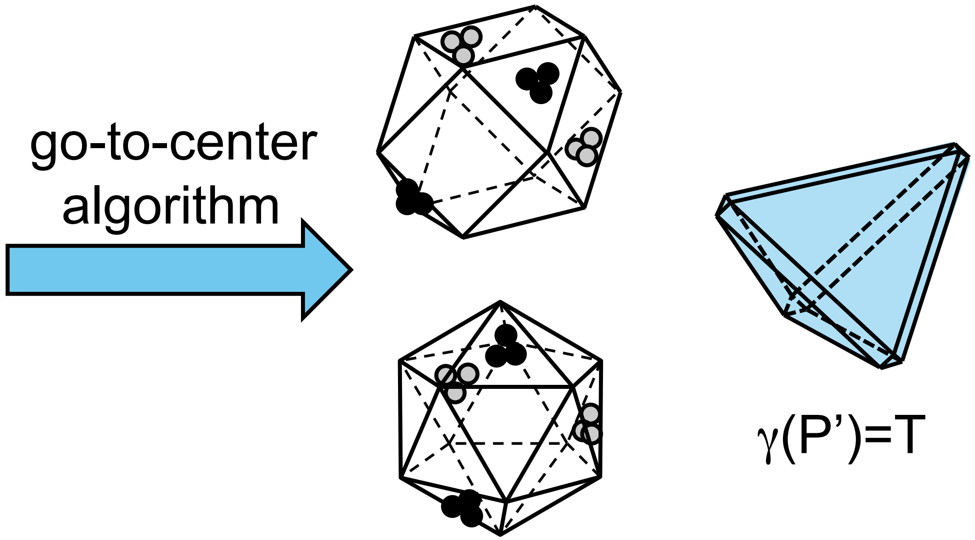


Reg. icosahedron

$\gamma(P) = I$

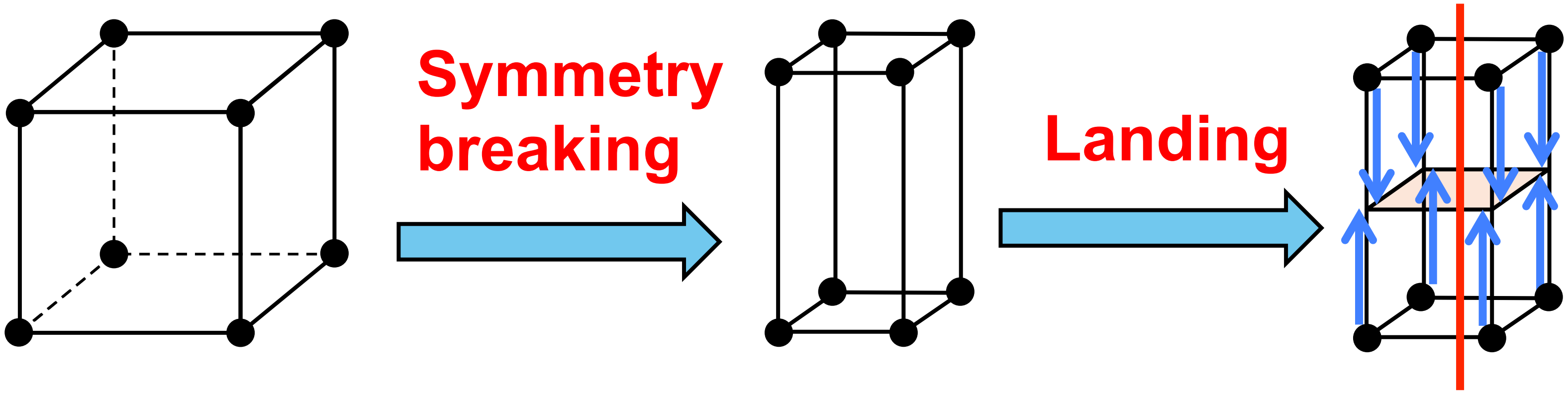
$|P|=12$





- T is in $\rho(P)$ for these two polyhedra
  - T generates 12 symmetric local coordinate systems
    - Robots can never break T, thus never form a plane
  - There are infinitely many such polyhedra

Plane formation algorithm

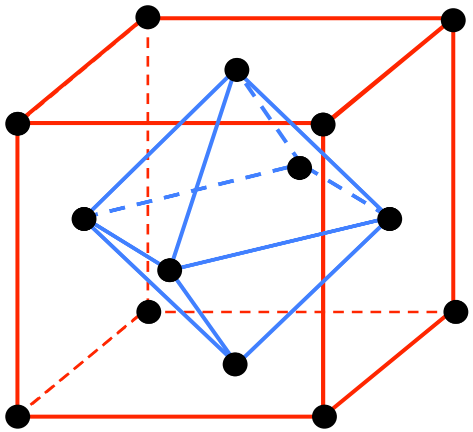


- Symmetry breaking phase
  - Break the five polyhedra by the go-to-center algorithm
  - Rotation group becomes 2D-group

- Landing phase
  - Robots agree on a plane and select distinct destinations

Composite initial configuration

- Rotation group of entire robots decomposes them
  - Each subgroup forms some "vertex-transitive" polyhedron
  - Robots can concentrate on one subgroup
    - To execute the go-to-center algorithm



Theorem 1.
Let P be an initial configuration of robots. Then oblivious FSYNC robots can form a plane from P if and only if $\rho(P)$ consists of cyclic and dihedral groups.

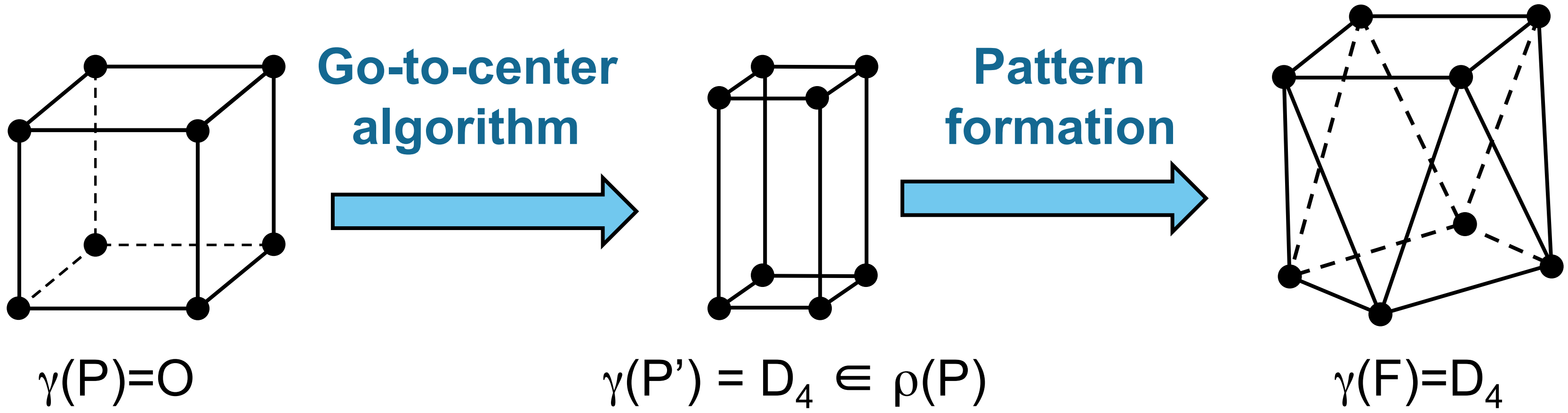
32

Pattern formation problem

Theorem 2

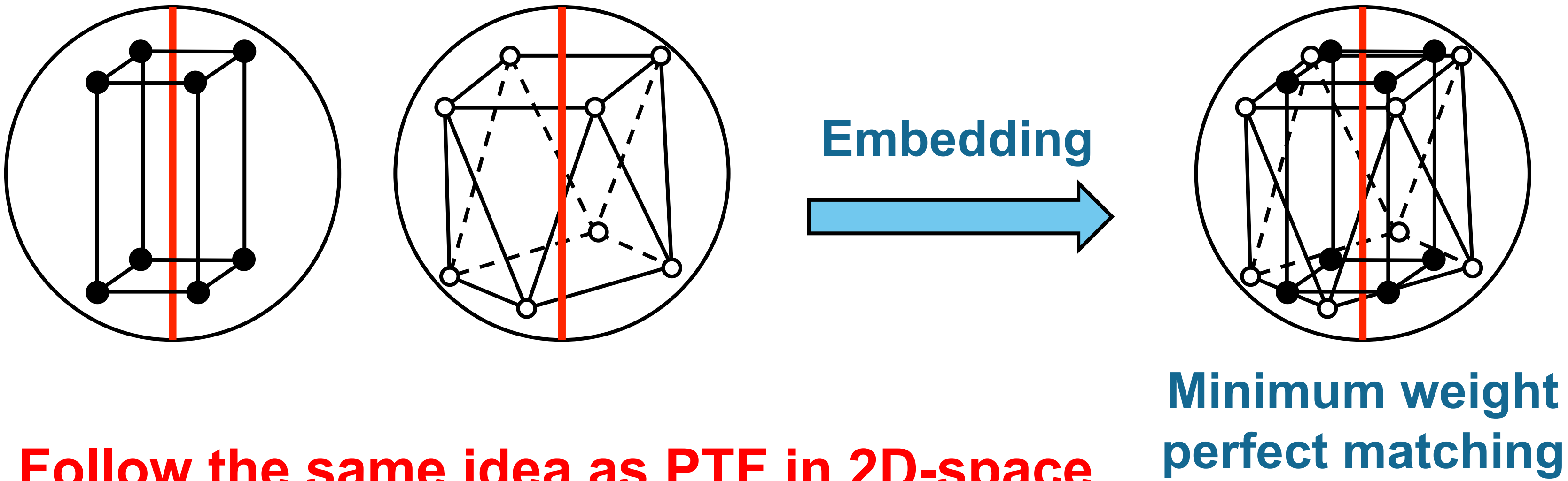
Theorem 2. Let P be an initial configuration of robots. Then oblivious fully-synchronous robots can form a target pattern F from P if and only if $\rho(P)$ is a subset of $\rho(F)$.

- Impossibility is clear from definition of symmetricity
- Pattern formation algorithm for solvable instances



✓ **$\rho(P)$ guarantees $\gamma(P')$ is a subgroup of $\gamma(F)$**

Pattern formation algorithm



Follow the same idea as PTF in 2D-space

- Embedding phase
  - Overlap the SEBs of P' and F and axes of γ (P') and γ(F)
- Matching phase
  - Robots agree on a minimum weight perfect matching between P' and F

Summary

- Symmetricity of robots in 3D-space
  - Symmetry of moving points is different from rotation group of points (i.e., group theory)
  - Characterization of PLF and PTF
    - Generalization of existing results in 2D-space

- Ongoing work: Weaker models
  - Semi-synchronous robots (Semi-synchronous cycles)
    - [Uehara et al., 2015]
  - Non-rigid movement (Robots may stop en route)
  - Without chirality (Symmetry by mirror planes)

Open problems

- Straightforward problems from 2D-plane
  - Problems
    - Gathering, scattering, marching, etc.
  - Robot models
    - Limited visibility, local memory, heterogeneous robots, etc.
  - Algorithm
    - Randomization
  - Information
    - Agreement in local coordinate systems, lights, etc.
  - Limited moving area
    - Sphere of a ball

- Clear separation between 2D and 3D space?