Conic Formation in Presence of Faulty Robots

Debasish Pattanayak*

IIT Guwahati, India

Partha Sarathi Mandal

IIT Guwahati, India

Klaus-Tycho Foerster

University of Vienna, Austria

Stefan Schmid

University of Vienna, Austria

ALGOSENSORS 2020

*Visit to University of Vienna is sponsored under SERB OVDF

Motivation

- Self-organization in Robots
- Pattern Formation
- Faults

Related Works

Pattern formation is addressed under various models :

- Arbitrary pattern formation for asynchronous oblivious robots. [Flocchini et al. TCS'08]
- Pattern formation in limited visibility. [Yamauchi et al. SIROCCO'13]
- Sequence of Patterns. [Das et al. Distributed Computing'15]

Special cases:

- Plane formation in 3D. [Yamauchi et al. JACM'13]
- Uniform circle formation. [Flocchini et al. Distributed Computing'17]

Model

Robots are

- Point : dimensionless point robots
- Homogeneous : are indistinguishable
- Oblivious : do not remember past actions
- Anonymous : do not possess identities
- Silent : do not send any messages

Model

Robots follow *look-compute-move* cycle.

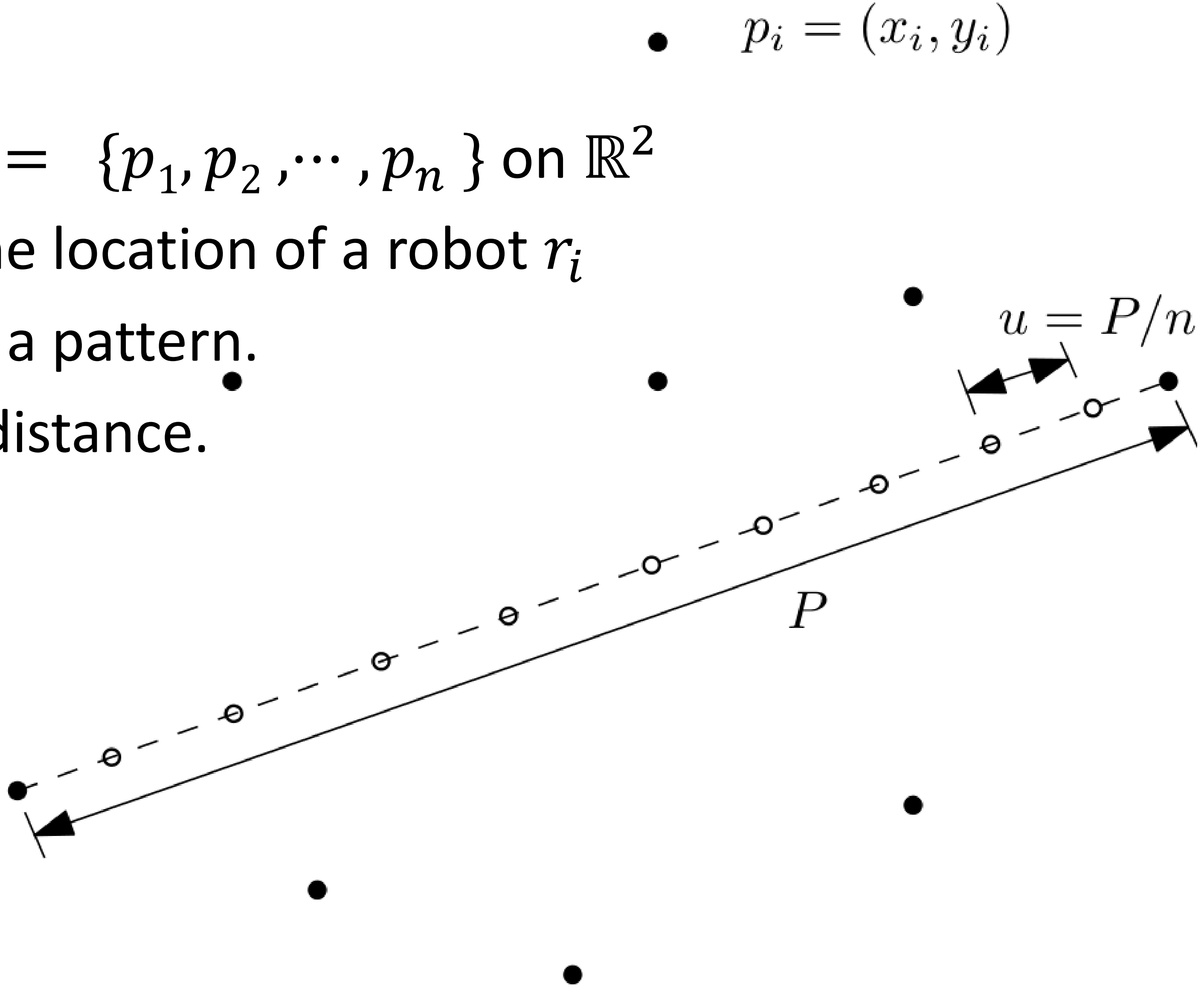
- Look : take a snapshot of the surrounding in local coordinate system
- Compute : compute the destination based on the snapshot
- Move : move to the destination

Model

- Robots are activated in a *fully-synchronous* manner : all robots are activated at once.
- Robots do not have any agreement on the coordinate system.
- Robots follow *rigid* movement : they move to the destination point in the same round
- The *faulty* robots *cannot move*.

Notations

- Configuration $C = \{p_1, p_2, \cdots, p_n\}$ on $\mathbb{R}^2$
- $p_i = \{x_i, y_i\}$ is the location of a robot r_i
- P is the length of a pattern.
- u is the uniform distance.



Problem

The objective is to form a conic pattern with all the robots (faulty and non-faulty).

The points on a conic pattern satisfy

$$a_1x^2 + a_2y^2 + a_3xy + a_4x + a_5y + a_6 = 0$$

Problem

We form the following conic patterns corresponding to the number of faults.

- $f = 1$: point
- $f = 2$: line
- $f = 3$: circle
- $f = 4$: parabola
- $f = 5$: parabola, hyperbola or ellipse

Assumptions

- Number of faulty robots f is known beforehand
- All the faulty robots have crashed initially
- All initial configurations are asymmetric
- All robots occupy initially distinct positions
- The faulty robots form a convex polygon

$f = 1$: Gathering

- The robots gather at a point.
- If a robot sees only one robot : move to that robot's position.
- All the robots move to the center of the smallest enclosing circle.

Lower Bound (Rounds)

Theorem: For every $f \geq 2$ and every $n \geq f + 3$ holds: Any deterministic algorithm needs more than one round to make a pattern passing through all f faulty robots.

Proof Idea: Since a non-faulty robot is indistinguishable from the faulty robots, a deterministic algorithm cannot choose the target pattern based on their location in one round.

Lower Bound (Robots)

Theorem: At least $2f + 1$ robots are required to form a pattern passing through f faulty robots.

Proof Idea: The robots that move to a target pattern should be a majority among the total robots, otherwise the adversary can select the robots moving to a different pattern to be faulty.

Configurations

Definition (Terminal Configuration). A configuration is a terminal configuration if all the robots form the target pattern corresponding to f faulty robots.

Definition (Type I Configuration). If exactly n−f robots are in the target pattern corresponding to f faulty robots, then it is a Type I configuration.

Definition (Type O Configuration). If a configuration is not Terminal or Type I, then it is a Type O configuration.

Configurations

Definition (Uniform Configuration). A configuration is a uniform configuration if the distance between all consecutive pairs of robots in the configuration along the pattern is the same.

Definition (Quasi-Uniform Configuration). If a uniform configuration with m uniform positions is occupied by n robots, where $n \leq m \leq 2n$, then it is a quasi-uniform configuration.

Configurations

Lemma. An asymmetric configuration is orderable.

Configuration can be split in three types: initial, transitional and final, denoted henceforth as C_0, C_1 and C_2.

Algorithm

Two steps of the algorithm are:

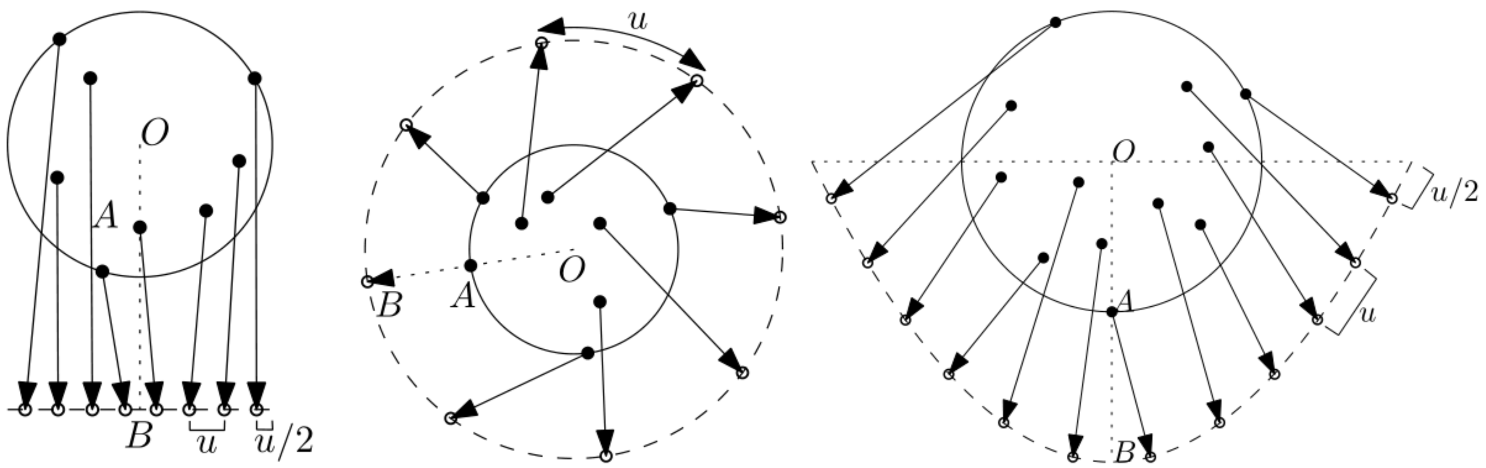
- Determine the faulty robot
- Move to the pattern containing faulty robots

Since the algorithm does not know which robots are actually faulty, in each round it does the following.

- Determine the target pattern.
- Move to the target pattern.

Algorithm

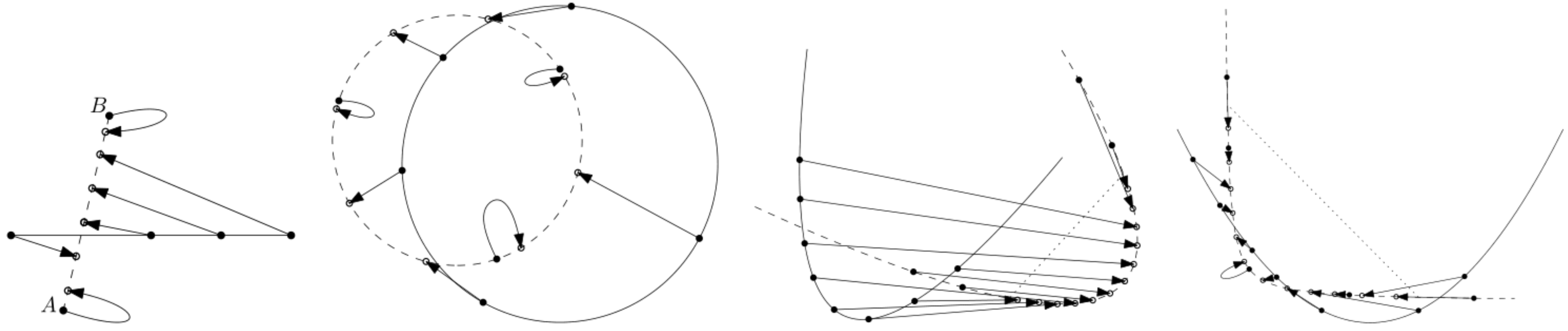
Type O configuration:



(a) $f = 2$ (b) $f = 3$ (c) $f = 4$ and $f = 5$

Algorithm

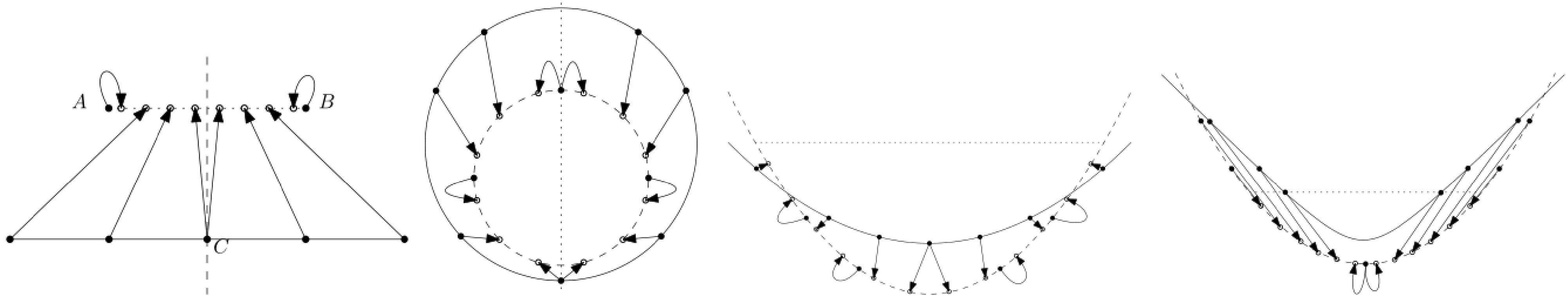
Asymmetric Type I configuration:



(a) $f = 2$ (b) $f = 3$ (c) $f = 4$ (d) $f = 5$

Algorithm

Symmetric Type I configuration:

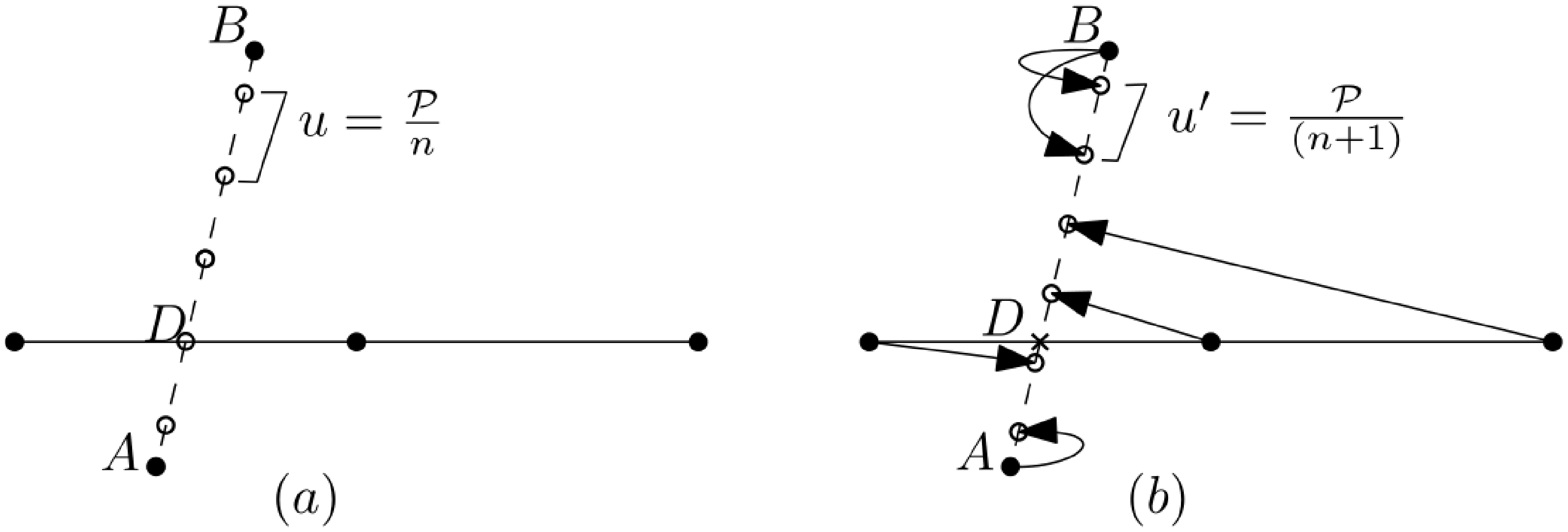


(a) $f = 2$ (b) $f = 3$ (c) $f = 4$ (d) $f = 5$

Algorithm

Choosing Non-overlapping Points

- Determine the pattern length P
- Choose points at uniform distance $u' = P/(n+1)$ if points at uniform distance $u = P/n$ overlaps with points existing pattern.



Correctness

Lemma. The destinations of all robots are distinct.

Lemma. Given configuration C and a destination set C', we have $C \cap C' = \phi$.

Lemma. It takes one round to determine all the faulty robots for a Type O configuration for $\mathrm{f} \in \{2,3,4,5\}$.

Lemma. It takes one round to determine all the faulty robots for a Type I configuration for $f \in \{2,3,4,5\}$.

Lemma. The target pattern passing through the faulty robots in C_1 can be uniquely determined.

Correctness

Theorem: Starting from any initial asymmetric configuration, the algorithm terminates in at most two rounds.

Corollary. Starting from a configuration other than the terminal configuration, the non-faulty robots are at uniform pattern points in the terminal configuration.

Corollary. The faulty robots can be determined from a terminal configuration unless it is the initial configuration.

Relaxation of Assumptions

We can extend our results in the following directions:

- Upper bound on the number of faults
- Initial configuration with reflective symmetry
- Lower order patterns for higher number of faults

Conclusion and Future Work

- We present a framework for pattern formation with faults.
- Our algorithm is optimal with respect to the number of rounds and number of robots.

Future Work:

- Extend to semi-synchronous
- Generalize the number of faults