Gathering Asynchronous Mobile Robots with Inaccurate Compasses*

Samia Souissi[1], Xavier Défago[1], and Masafumi Yamashita[2]

[1] School of Information Science
Japan Advanced Institute of Science and Technology (JAIST)
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan
{ssouissi, defago}@jaist.ac.jp
[2] Department of Computer Science and Communication Engineering,
Kyushu University, Fukuoka, Japan
mak@csce.kyushu-u.ac.jp

Abstract. This paper considers a system of asynchronous autonomous mobile robots that can move freely in a two-dimensional plane with no agreement on a common coordinate system. Starting from any initial configuration, the robots are required to eventually gather at a single point, not fixed in advance (gathering problem).

Prior work has shown that gathering oblivious (i.e., stateless) robots cannot be achieved deterministically without additional assumptions. In particular, if robots can detect multiplicity (i.e., count robots that share the same location) gathering is possible for *three or more* robots. Similarly, gathering of any number of robots is possible if they share a common direction, as given by compasses, with *no errors*.

Our work is motivated by the pragmatic standpoint that (1) compasses are error-prone devices in reality, and (2) multiplicity detection, while being easy to achieve, allows for gathering in situations with more than two robots. Consequently, this paper focusses on gathering two asynchronous mobile robots equipped with *inaccurate* compasses. In particular, we provide a self-stabilizing algorithm to gather, in a finite time, two oblivious robots equipped with compasses that can differ by as much as $\pi/4$.

1 Introduction

Background. The problem of reaching agreement among autonomous robots has attracted considerable attention within the last few years. One problem of particular interest is the gathering problem, where robots are required to meet at a single location not predetermined in advance, and without agreement on a common coordinate system. This problem has been studied extensively in the literature, under different models and various assumptions [3,4,9,17]. In fact, several factors render this problem difficult to solve. In particular, in all these studies, the problem has been solved only by making some additional assumptions regarding robots' capabilities.

* Work supported by MEXT Grant-in-Aid for Young Scientists (A) (Nr. 18680007).

A. Shvartsman (Ed.): OPODIS 2006, LNCS 4305, pp. 333–349, 2006.

In this paper, we focus on solving the gathering problem in asynchronous models. In the asynchronous model CORDA [12], Prencipe [13] has shown that there exists no deterministic algorithm to solve the gathering problem in finite time with oblivious robots. Cieliebak et al. [4] have introduced multiplicity, and have shown that gathering is possible for three or more robots, when they are able to detect multiple robots at a single point.

Flocchini et al. [9] have solved the gathering problem for any number of robots when they share a common direction, as provided by a compass[1]. However, their result holds when compasses are perfectly consistent (i.e., with no errors). Yet, in practice sensors are error-prone and sensitive to magnetic interference. Consequently, in this paper, we concentrate on the gathering of two asynchronous mobile robots when their compasses are subject to errors.

This work is motivated by the facts that: (1) in practice, compasses are rather inaccurate sensors, and (2) with multiplicity detection, the gathering is solvable only for more than two robots. For example, the accuracy of compasses typically varies from 1 degree to over 10 degrees, depending on sensor quality (cost) and environment conditions. Therefore, our aim is to fill the gap of solving the gathering problem for two robots relying on oblivious computations, and to provide effective answers to the following two questions. First, is it possible to gather two asynchronous mobile robots when their compasses are inaccurate by some unknown angle? Second, what is the bound of that angle?

Contribution. The main contribution of this paper is to study the solvability of the gathering of two asynchronous mobile robots in the face of compass inaccuracies. In particular, we address the problem when robots are oblivious (or memoryless), meaning that they can not remember their previous states, their previous actions or the previous positions of the other robots. While this is a somewhat over-restrictive assumption, developing algorithms in this model is interesting because any algorithm that works correctly for oblivious robots is intrinsically self-stabilizing[2]. We thus provide an algorithm that gathers in a finite number of steps, two asynchronous oblivious mobile robots equipped with compasses that can differ by as much as $\pi/4$.

Difficulty of the problem. In the asynchronous model CORDA, where robots are equipped with inaccurate compasses, it is difficult to gather two robots or compare them in a consistent manner. This is mainly due to the issue of breaking the symmetry between these robots. Let us illustrate this point using a simple example. Assume that there exists a naive algorithm for comparing two asynchronous robots A and B in a consistent manner when their compasses are inaccurate. First, consider that A and B are equipped with accurate compasses, and place them at the two endpoints of a horizontal diameter of a unit circle. Then, a naive algorithm can be based on the comparison of the angles that A and B form respectively with some global North N (i.e., they share the same north) and the

[1] A compass does not only indicate the North direction, but also gives a unified clockwise orientation.

[2] Self-stabilization is the property of a system which, starting in an arbitrary state, always converges toward a desired behavior [7,14].

segment $\overline{AB}$ in clockwise direction. For instance, if the angle is less than or equal to $\pi/2$, the robot wins. Otherwise, if the angle is greater than $\pi/2$, the robot loses. Then, a robot, say A, wins. Then, we rotate the diameter to exchange the positions of A and B. Now B wins. We thus, color the perimeter of the circle by Win and Lose, where at any point which is colored Win or Lose, A wins or loses. Then, there is a point p that is a boundary between a Win and a Lose segment. By introducing error to their compasses, at p, we can derive a contradiction. That is, we can not decide which robot wins, and which one loses.[3]

Related Work. In their SYm model [17], referred to a semi-synchronous model, Suzuki and Yamashita proposed an algorithm to solve the gathering problem deterministically in the case where robots have unlimited visibility. For a system with two robots, they have proven that it is impossible to achieve the gathering of two *oblivious* mobile robots that have no common orientation under their semi-synchronous model, in a finite time. The difficulty of the problem is inherent in breaking the symmetry between the two robots.

Using the same model, Ando et al. [2] proposed an algorithm to address the gathering problem in systems wherein robots have limited visibility. Their algorithm *converges* toward a solution to the problem, but it does not solve it deterministically. The gathering problem also has been studied in the presence of faulty robots by Agmon and Peleg [1] in synchronous and asynchronous settings. In particular, they proposed an algorithm that tolerates one crash-faulty robot in a system of three or more robots. They also showed that in an asynchronous environment, it is impossible to perform a successful gathering in a 3-robot system with one Byzantine[4] failure. Later on, Défago et al. [6] strengthen the impossibility of gathering in systems with Byzantine robots by showing that it still holds in stronger models. They also show the existence of randomized solutions for systems with Byzantine-prone robots.

In some of our recent work [15], we introduced the notion of unreliable compasses for robots, and we studied the solvability of the gathering problem in the face of compass instabilities. In particular, we proposed a gathering algorithm that solves the problem in the semi-synchronous model SYm for many robots, with compasses that are eventually stabilizing.

Recently, Cohen and Peleg [5] addressed the issue of analyzing the effect of errors in solving gathering and convergence problems. In particular, they studied imperfections in robot measurements, calculations and movements. They showed that gathering cannot be guaranteed in environments with errors, and illustrated how certain existing geometric algorithms, including ones designed for fault-tolerance fail to guarantee even convergence in the presence of small errors. One of their main positive results is an algorithm for convergence under bounded measurement, movement and calculation errors. However, their work does not relate to compasses.

[3] The argument is similar to the bi-valent argument in the impossibility result of the consensus problem [8].

[4] A robot is said to be Byzantine if it executes arbitrary steps that are not in accordance with its local algorithm [18].

While preparing the print-ready version of this manuscript, it came to our attention that a similar result has been presented by Imasu et al. [10] at a domestic workshop in Japan.

Structure. The remainder of this paper is organized as follows. In Sect. 2, we describe the system model and the basic terminology. Sect. 3 describes the algorithm to gather two asynchronous oblivious mobile robots under compass inaccuracies, and Sect. 4 proves its correctness. Finally, Sect. 5 concludes the paper.

2 System Model and Definitions

2.1 System Model

In this paper, we consider the CORDA model of Prencipe [12,11], which is defined as follows. The system consists of a set of autonomous mobile robots $\mathcal{R} = \{r_1, \cdots, r_n\}$. A robot is modelled as a unit having computational capabilities, and which can move freely in the two-dimensional plane. In addition, robots are equipped with sensor capabilities to observe the positions of other robots, and form a local view of the world. The robots are modelled and viewed as points in the Euclidean plane.[5] The local view of each robot includes a unit of length, an origin and the directions and orientations of the two x and y coordinate axes as given by a compass.

The robots are completely *autonomous*. Moreover, they are *anonymous*, in the sense that they are a priori indistinguishable by appearance, and they do not have any kind of identifiers that can be used during their computations. Furthermore, there is no direct means of communication among them.

We further assume that the robots are *oblivious*, meaning that they keep information neither on previous observations nor on past computations.

The cycle of a robot consists of four states: Wait-Look-Compute-Move.

- *Wait.* In this state, a robot is idle. A robot cannot stay permanently idle (see Assumption 2) below. At the beginning all robots are in Wait state.
- *Look.* Here, a robot *observes* the world by activating its sensors, which will return a snapshot of the positions of all other robots with respect to its local coordinate system. Since each robot is viewed as a point, the positions in the plane are just the sets of robots' coordinates.
- *Compute.* In this state, a robot *performs* a local computation according to its deterministic, oblivious algorithm. The algorithm is the same for all robots, and the result of the *compute* state is a destination point.
- *Move.* The robot *moves* toward its computed destination. If the destination is its current location, then the robot is said to perform a *null movement*; otherwise, it is said to execute a *real movement.* The robot moves toward the computed destination, but the distance it moves is unmeasured; neither infinite, nor infinitesimally small (see Assumption 1). Hence, the robot can only go towards its goal, but the move can end anywhere before the destination.

[5] We assume that there are no obstacles to obstruct vision. Moreover, robots do not obstruct the view of other robots and can "see through" other robots.

The (global) time that passes between two successive states of the same robot is finite, but unpredictable. In addition, no time assumption within a state is made. This implies that the time that passes after the robot starts observing the positions of all others and before it starts moving is arbitrary, but finite. That is, the actual movement of a robot may be based on a situation that was observed arbitrarily far in the past, and therefore it may be totally different from the current situation.

In the model, there are two limiting assumptions related to the cycle of a robot.

Assumption 1. *It is assumed that the distance travelled by a robot r in a move is not infinite. Furthermore, it is not infinitesimally small: there exists a constant $\delta_r > 0$, such that, if the target point is closer than δ_r, r will reach it; otherwise, r will move towards it by at least δ_r.*

Assumption 2. *The time required by a robot r to complete a cycle (Wait-Look-Compute-Move) is not infinite. Furthermore, it is not infinitesimally small; there exists a constant $\epsilon_r > 0$, such that the cycle will require at least ϵ_r time.*

2.2 Definitions

Definition 1 (Absolute north). *An absolute north $\overrightarrow{\mathcal{N}}$ is a vector that indicates a fixed north direction. The absolute north is collocated with an absolute y positive axis.*

It is important to stress that the absolute north is not known to the robots, and is used only for the sake of explanation.

Definition 2 (Compass). *A compass is a function of robots and time. The function outputs a relative north direction $\overrightarrow{N_r}(t)$ for some robot r at time t.*

Definition 3 (γ^*-Inaccurate compasses). *Informally, compasses are γ^*-*Inaccurate *iff., for every robot r, the absolute difference between the north indicated by the compass of r and $\overrightarrow{\mathcal{N}}$ is at most γ^* at any time t (also referred to as* error *of the compasses). In addition, for every robot r, the error of its compass is consistent or invariant, i.e., the error of the compass does not fluctuate over time. In other words, a pair of γ^*-Inaccurate compasses can differ by as much as $2\gamma^*$ at any time t, and the difference is invariant. The special case when $\gamma^* = 0$ represents perfect compasses.*

*Formally, compasses are γ^*-*Inaccurate *iff., the following two properties are satisfied:*

1. *γ^*-Inaccuracy: $\forall r \in \mathcal{R}, \forall t, |\measuredangle \overrightarrow{\mathcal{N}}\overrightarrow{N_r}(t)| \leq \gamma^*$,*
2. *Invariance: $\forall r, \forall t, t', \overrightarrow{N_r}(t) = \overrightarrow{N_r}(t')$.*

2.3 Notations

Given some robot r, $r(t)$ is the position of r at a time t. Let A and B be two points, with $\overline{AB}$, we indicate the segment starting at A and terminating at B, and $\|\overline{AB}\|$ is the length of such a segment. Given three distinct points A, B, and C, we denote by $\triangle(A, B, C)$, the triangle having them as corners, and by $\widehat{BAC}$, the angle formed by A, B and C, and centered at A. Finally, given a region $X(t)$ at time t, we denote by $|X(t)|$, the number of robots in that region at time t. The parameter t is omitted whenever clear from the context.

3 Gathering with Inaccurate Compasses

The basic intuition behind the algorithm is to break the symmetry between two robots, that is, to forbid symmetric configurations of two robots. More precisely, with a perfect compass, it is easy to break the symmetry between two robots. For instance, by making one robot move and the other remain stationary. However, with inaccurate compasses, it is difficult to design an algorithm that breaks the symmetry between the two, as they can end up in a situation in which neither do move, which results in a deadlock situation or in situation inc which both move in such a way they cycle forever. In order to avoid such situations, it is first necessary to ensure that the two robots do not see each other on the same zone.

The main idea of our algorithm is to make each robot partition the plane into four different zones, so that two similar zones for two different robots should not overlap. Then, depending on the different possible configurations (resulting from the partitions) of the two robots, we design their movements such that a configuration is transformed to gathering, or to an intermediate configuration leading to the gathering, without introducing cycles between configurations or deadlock situations.

Before we describe the algorithm in more detail, we will first explain how robots divide the plane.

3.1 Partitions

First, a robot needs to partition the plane into four sectors that do not overlap, namely the *North*, *South*, *East* and *West* sectors. Let α_N, α_S, α_E and α_W be the respective angular measurements of these sectors. Also, by Λ_N, Λ_S, Λ_E and Λ_W, we denote the rays delimiting these sectors, respectively (refer to Fig. 1).

Now, let us assume there exits a constant $\gamma^* \geq 0$ that represents the maximum angle inaccuracy between the relative north $\overrightarrow{N_r}$ of some robot r and the absolute north $\overrightarrow{\mathcal{N}}$. Then, the following conditions must be satisfied in order to avoid a situation in which both robots see each other in the same sector because of compass inconsistencies.

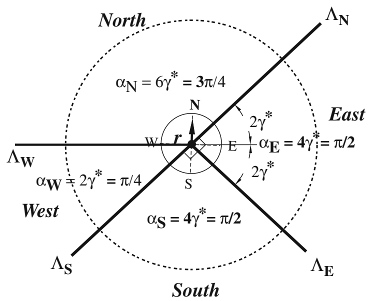


Fig. 1. The four sectors *North*, *South*, *East* and *West* for robot r

$$\alpha_N \leq \pi - 2\gamma^* \tag{1}$$
$$\alpha_S \leq \pi - 2\gamma^* \tag{2}$$
$$\alpha_E \leq \pi - 2\gamma^* \tag{3}$$
$$\alpha_W \leq \pi - 2\gamma^* \tag{4}$$

We further set the following conditions on the sectors. These conditions will help to avoid the occurrence of infinite executions, i.e., having robots looping in the same configuration.

$$\alpha_E + \alpha_S \leq \pi \tag{5}$$
$$\alpha_N + \alpha_W \leq \pi \tag{6}$$

By summation of Equation (1) and Equation (5), we get:

$$\begin{aligned} \alpha_N + \alpha_E + \alpha_S &\leq 2\pi - 2\gamma^* \quad \text{then,} \\ \alpha_N + \alpha_E + \alpha_S + \alpha_W &\leq 2\pi - 2\gamma^* + \alpha_W \\ 2\pi &\leq 2\pi - 2\gamma^* + \alpha_W \\ 2\gamma^* &\leq \alpha_W \end{aligned}$$

After finding the condition in the *West* sector, we choose the minimal value for α_W. That is, $\alpha_W = 2\gamma^*$. Then, by summation of Equation (1), and Equation (2), we get:

$$\begin{aligned} \alpha_N + \alpha_S &\leq 2\pi - 4\gamma^* \quad \text{then,} \\ \alpha_N + \alpha_S + \alpha_E &\leq 2\pi - 4\gamma^* + \alpha_E \\ \text{By hypothesis,} \quad \alpha_N + \alpha_S + \alpha_E &\leq 2\pi \quad \text{then, by subtraction, we get:} \\ 0 &\leq -4\gamma^* + \alpha_E \quad \text{then,} \\ 4\gamma^* &\leq \alpha_E \end{aligned}$$

Thus, we choose $\alpha_E = 4\gamma^* = \alpha_S = \pi/2$ (From Equation (5)). This means that $\gamma^* = \pi/8$. It follows that, $\alpha_W = 2\gamma^* = \pi/4$. Finally, from Equation (1), and the fact that the sum of the four sectors is equal to 2π, we get, $\alpha_N = \pi - 2\gamma^* = 3\pi/4$.

We have derived the condition that $\gamma^* \leq \pi/8$. Thus, in the remainder of the paper, we consider the largest inaccuracy value of γ^*, i.e., $\gamma^* = \pi/8$.

We now describe in more detail the features of each sector, as follows:

- *East*(r) sector: it is centered at r, has the East direction (indicated by its compass) $\overrightarrow{E_r}$ as bisector, and its angular sector α_E is equal to $4\gamma^*$, which is $\pi/2$. Note that *East*(r) is delimited by $\Lambda_N(r)$ and $\Lambda_E(r)$. However, only $\Lambda_E(r)$ is a part of *East*(r).
- *South*(r) sector: it is adjacent to *East*(r) in clockwise direction, and its angular sector α_S is equal to α_E, which is equal to $4\gamma^*$ (i.e., $\pi/2$). Note that *South*(r) is delimited by $\Lambda_E(r)$ and $\Lambda_S(r)$. However, only $\Lambda_S(r)$ is included in *South*(r).
- *West*(r) sector: it is adjacent to *South*(r) in clockwise direction and its angular sector α_W is equal to $2\gamma^*$, that is $\pi/4$. Note that *West*(r) is delimited by $\Lambda_W(r)$ and $\Lambda_N(r)$. However, only $\Lambda_W(r)$ is a part of *West*(r) sector.
- *North*(r) sector: this is the remaining sector, and its angular sector α_N is equal to $6\gamma^*$, that is $3\pi/4$. Note that *North*(r) is delimited by $\Lambda_N(r)$ and $\Lambda_W(r)$. However, only $\Lambda_N(r)$ is included in *North*(r) sector.

In the following, we will describe the possible configurations of the two robots, given the above partitions.

3.2 Valid Configurations

We consider two robots r and r' that are equipped with compasses that can diverge by as much as $2\gamma^*$, that is $\pi/4$. Let r and r' divide the plane as described in Sect. 3.1. Then, r and r' can only be in one of the following valid configurations, or a symmetric configuration:

1. Configuration *North/South*: $r' \in$ *South*(r) (i.e., robot r sees r' on its *South* sector) and $r \in$ *North*(r'), or vice versa.
2. Configuration *North/East*: $r' \in$ *East*(r) and $r \in$ *North*(r'), or vice versa.
3. Configuration *North/West*: $r' \in$ *West*(r) and $r \in$ *North*(r'), or vice versa.
4. Configuration *East/West*: $r' \in$ *West*(r) and $r \in$ *East*(r'), or vice versa.
5. Configuration *East/South*: $r' \in$ *South*(r) and $r \in$ *East*(r'), or vice versa.

Based on the partitions described in Sect. 3.1, Table 1 summarizes possible and impossible configurations when robots's compasses are inaccurate by at most $\gamma^* = \pi/8$, with respect to some global north. By design, the partitions prevent the occurrence of some undesirable configurations, such as *North/North*, that could lead to a deadlock situation by using the algorithm[6](see Sect. 3.3).

[6] It is important to mention that when γ^* is equal to zero, i.e., when the compasses of r and r' are consistent or, the configurations *East/South* and *North/West* are impossible.

Algorithm 1. Gathering Two Robots with $\pi/8$-Inaccurate Compasses

1: Robot r divides the plane into four sectors: *North*, *South*, *East* and *West* (see Sect. 3.1);
2: r' := the other robot visible to r at some time t;
3: **if** (r sees only itself) **then** {*gathering terminated*}
4: Do_nothing();
5: **else**
6: **if** ($|South(r)| > 0$) **then** {r' is to the *South*: *direct move*}
7: Move(r');
8: **else if** ($|East(r)| > 0$) **then** {r' is to the *East*: *side move up*}
9: $\Psi_E(r)$:= the parallel axis to $\Lambda_E(r)$ passing through r';
10: $H := \Lambda_N(r) \cap \Psi_E(r)$ (see Fig. **??**);
11: $Goal := p \in \Lambda_N(r)$ such that $\|\overline{rGoal}\| > \|\overline{rH}\|$ and $r\widehat{Goal r'} \geq r\widehat{r' Goal}$;
12: Move($Goal$);
13: **else if** ($|West(r)| > 0$) **then** {r' is to the *West*: *side move down*}
14: $\Psi_W(r)$:= the parallel axis to $\Lambda_W(r)$ passing through r';
15: $H' := \Lambda_S(r) \cap \Psi_W(r)$ (see Fig. **??**);
16: $Goal := p \in \Lambda_S(r)$ such that $\|\overline{rGoal}\| > \|\overline{rH'}\|$ and $r\widehat{Goal r'} \geq r\widehat{r' Goal}$;
17: Move($Goal$);
18: **else** {r' is to the *North*: *no movement.*}
19: Do_nothing();
20: **end if**
21: **end if**

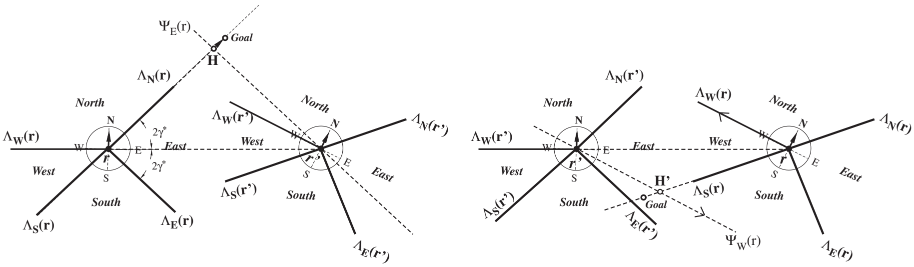
Table 1. Different configurations and movements of robot r and r' ($\gamma^* = \pi/8$)

	Robot r			
Robot r'	*North (no movement)*	*South (direct move)*	*East (side move up)*	*West (side move down)*
North (no movement)	no	○	○	○
South (direct move)	○	no	○	no
East (side move up)	○	○	no	○
West (side move down)	○	no	○	no

3.3 Movements

The algorithm is given in Algorithm 1, and Table 1 summarizes the different movements of robot r and r' (the table is symmetrical). Let us consider the movement of robot r. First, robot r creates the four sectors, and then it decides its movement based on the sector in which it has locates robot r', as follows:

- *No movement* (Algorithm1:line 18): If $r' \in North(r)$, then r does not move. That is, if r sees r' in its *North* sector, it remains stationary.
- *Direct move* (Algorithm1:line 6): If $r' \in South(r)$, then r moves directly in a linear movement to r'.
- *Side move up* (Algorithm1:line 8): If $r' \in East(r)$, then r performs a *side move up*. The need for such a move is explained as follows: given the valid



(a) Side move up on $\Lambda_N(r)$: $r' \in East(r)$, then r performs a *side move up* to *Goal*.

(b) Side move down on $\Lambda_S(r)$: $r' \in West(r)$, then r performs a *side move down* to *Goal*.

Fig. 2. Principle of the algorithm

configurations described in Sect. 3.2, if $r' \in East(r)$, then $r \in South(r')$ or $r \in North(r')$ or $r \in West(r')$. Robot r (also r') cannot figure out in which configuration they are, for instance the *East*/*South* or *North*/*East* configuration. Then, if we let robot r make a *direct move* toward r', then in the case when both robots are in the configuration *East*/*South*, they will swap their positions endlessly. Also, if we make robot r stay still, then, if both robots are in the configuration *North*/*East*, none of the robots will ever move and they will always remain in a deadlock situation. Therefore, the aim of this *side move up* is to bring both robots eventually into the configuration *North*/*South*, where one robot can move and the other remains stationary, which can lead to gathering by our algorithm.

A *side move up* is computed by robot r as follows: let H be the intersection of $\Lambda_N(r)$ and the axis $\Psi_E(r)$, with $\Psi_E(r)$ parallel to $\Lambda_E(r)$ passing through robot r'. Then, the destination *Goal* of robot r is any point that belongs to $\Lambda_N(r)$, such that the distance $\|\overline{r\,Goal}\| > \|\overline{rH}\|$, and the angle $\widehat{r\,Goal\,r'}$ is greater than or equal to the angle $\widehat{rr'\,Goal}$ (refer to Fig. 2(a)).

– *Side move down* (Algorithm1:line 13): If $r' \in West(r)$, then r performs a *side move down*. The aim of this move is similar to the *side move up*, and it is computed by robot r as follows: let H' be the intersection of $\Lambda_S(r)$ and the axis $\Psi_W(r)$, with $\Psi_W(r)$ parallel to $\Lambda_W(r)$ passing through robot r' (refer to Fig. 2(b)). Then, the destination *Goal* of robot r is any point that belongs to $\Lambda_S(r)$, such that the distance $\|\overline{r\,Goal}\| > \|\overline{rH'}\|$, and the angle $\widehat{r\,Goal\,r'}$ is greater than or equal to the angle $\widehat{rr'\,Goal}$ (refer to Fig. 2(b)).

4 Correctness

In this section, we will prove that our algorithm solves the problem of gathering two robots in a finite time, assuming $\pi/8$-Inaccurate compasses. Due to space limitations, we only give the complete proof of two lemmas that are central to

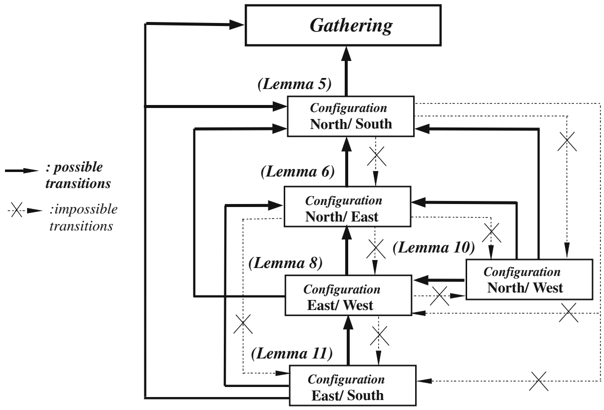


Fig. 3. Different configurations allowed by Algorithm 1, and their transformation to gathering

the paper. For all other lemmas, we give an outline of the idea behind the proof. All the complete proofs can be found in the technical report version [16]. We first state some lemmas, to illustrate that some incompatible configurations are ruled out by the algorithm. Second, we show how any possible configuration by the algorithm is transformed into gathering in a finite time. Fig. 3 summarizes the different possible configurations, and their transformation to gathering.

Under the partitions described in Sect. 3.1 and by considering $\gamma^* = \pi/8$, trivially, we derive the following two lemmas:

Lemma 1. *Under the partitions, and assuming* $\pi/8$*-Inaccurate compasses, the system can not be in the configuration North/North or East/East or South/South or West/West at any time* t*.*

Lemma 2. *Under the partitions, and assuming* $\pi/8$*-Inaccurate compasses, the system can not be in the configuration West/South at any time* t*.*

From the above two lemmas, we derive the following theorem:

Theorem 1. *By the algorithm, the possible configurations are North/South, North/East, North/West, East/West and East/South, and their symmetric ones (i,e. South/North, East/North, West/North, West/East and South/East).*

Lemma 3. *Given a robot* r *and its target point* H *with* $r \neq H$*,* r *reaches its target in a finite number of steps.*

Proof (Lemma 3). The proof derives from Assumption 1. In one cycle, r travels at least $\delta_r > 0$ of the desired distance. Besides, by Assumption 2, the cycle of a robot is finite. Thus, the number of steps required for robot r to reach its destination H is at most $\lceil \|\overline{rH}\|/\delta_r \rceil$, which is finite, and the lemma holds.

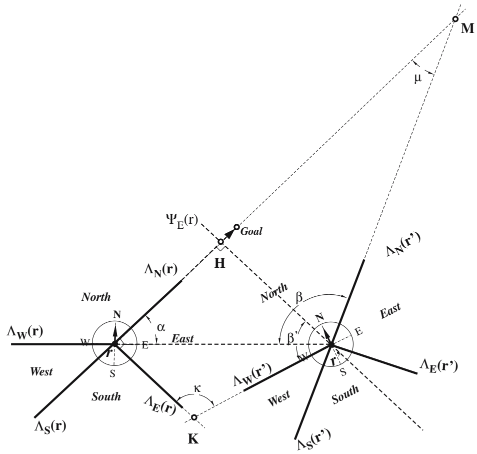


Fig. 4. Transformation of *North*/*East* configuration

Lemma 4. *Given two robots r and r' that are in the configuration North/East or East/West or East/South at some time t_0, with $r' \in East(r)$ and r is either in $North(r')$ or $West(r')$ or $South(r')$. Then, the destination Goal computed by robot r (resulting from its side move up) is on the $North(r')$.*

Proof (Lemma 4).

We will prove the *North*/*East* configuration only. The *East*/*West* and *East*/*South* configurations can be proved in a similar way.

Assume that $r' \in East(r)$ and $r \in North(r')$ at time t_0. First, observe that if $\Lambda_N(r) \cap \Lambda_N(r') = \emptyset$ (i.e., $\Lambda_N(r)$ and $\Lambda_N(r')$ are parallel or do not intersect), then $Goal \in North(r')$ because $r \in North(r')$, and $Goal \in \Lambda_N(r)$.

Now assume that, $\Lambda_N(r) \cap \Lambda_N(r') = M$. Let $H = \Psi_E(r) \cap \Lambda_N(r)$ (refer to Fig. 4). To show that $Goal \in North(r')$, we will show that, always, $Goal \in \triangle(r, r', M)$. In other words, we need to show that $H \in \triangle(r, r', M)$ and the distance $\|\overline{HM}\| \neq 0$.

Consider the triangle $\triangle(r, r', M)$. Let α, β, and μ denote the angles at r, r' and M that are within the triangle $\triangle(r, r', M)$, respectively. First, if all three angles α, β, and μ are acute, then obviously the foot H of the perpendicular starting from r' is inside $\triangle(r, r', M)$, and $\|\overline{HM}\| \neq 0$. Second, if the angle β at r' is obtuse, then again the foot H of the perpendicular starting from r' is inside $\triangle(r, r', M)$, and $\|\overline{HM}\| \neq 0$. Now consider the angle α at r. By hypothesis, α_E is equal to $\pi/2$. This means that α cannot be an obtuse angle, and it is at most $\pi/2$. In this later case where $\alpha = \pi/2$, we have the foot H of the perpendicular starting from r' equal to r (in this case $\Lambda_E(r)$ passes by r'), and the triangle $\triangle(r', r, M)$ has a right angle at r. Consequently, $\|\overline{rM}\| = \|\overline{HM}\| \neq 0$ and $Goal \in \triangle(r, r', M)$.

Now, we will prove that the angle μ at M can not be an obtuse angle (because if μ is an obtuse angle, H is outside $\triangle(r, r', M)$). Let $K = \Lambda_E(r) \cap \Lambda_W(r')$ and

κ be the angle at K. We also denote by β' the angle at r' formed by $\Psi_E(r)$ and $\Lambda_W(r')$. Consider the quadrilateral formed by r, H, r' and K. Then, we have: (1) $\kappa+\beta' = \pi$ since the respective angles at r and H are equal to $\pi/2$. Consider now the quadrilateral formed by r, K, r' and M. Then, we have: (2) $\kappa + \mu = 3\pi/4$ since $\alpha_E(r)$ is equal to $\pi/2$, and $\alpha_N(r')$ is equal to $3\pi/4$ by hypothesis. By subtraction of (1) from (2), we get: (3) $\beta' - \mu = \pi/4$. By assumption, $\beta' < 3\pi/4$ because $\Psi_E(r)$ can not be equal to $\Lambda_N(r')$ as $\Lambda_N(r')$ can not be perpendicular to $\Lambda_N(r)$ by the partitions described in Sect. 3.1. Consequently, the angle μ at M is less than $\pi/2$. Thus, μ can not be an obtuse angle. As a result, in all cases the foot H of the perpendicular starting from r' is inside the triangle $\triangle(r, r', M)$, and $\|\overline{HM}\| \neq 0$. Then, $\forall p \in \overline{HM}$, $p \in North(r')$. We have by the algorithm, $\widehat{rGoalr'} \geq \widehat{rr'Goal}$. Since μ is not an obtuse angle and $\widehat{rr'M}$ can be an obtuse angle, then the triangle $\triangle(r, r', Goal)$ is included in $\triangle(r, r', M)$. This proves that $Goal \in \triangle(r, r', M)$, and thus $Goal \in North(r')$. This completes the proof.

In the following, we will show the different possible transitions that each valid configuration can take, in order to reach gathering in a finite time. The impossible transitions can be derived implicitly, so we do not prove them explicitly.

4.1 Transition of *North/South* Configuration to Gathering

Lemma 5. *Let r and r' be two robots that are in the configuration North/South with $r' \in South(r)$ at some time t_0. Then, there is a time $\bar{t} > t_0$ when r and r' gather at the same point. Moreover, r and r' can not shift to any other configuration except gathering.*

Proof (Lemma 5). By the algorithm, r will perform a *direct move* toward r'. Also, during the movement of r, r' is unable to move. Consequently, by Lemma 3, r reaches r' in a finite time. This terminates the proof.

4.2 Transition of *North/East* Configuration to Gathering

Lemma 6. *Let r and r' be two robots that are in the configuration North/East with $r' \in East(r)$, and $r \in North(r')$ at some time t_0. Then, there is a finite time $\bar{t}$ at which this configuration is transformed into North/South configuration with $r' \in South(r)$. Moreover, r and r' can not shift to any other configuration except the North/South configuration.*

Proof (Lemma 6). The proof is a direct consequence from Lemma 4. Let $Goal$ be the destination of r. Initially, $r \in North(r')$. Besides, by Lemma 4, $\forall p \in \overline{rGoal}$, $p \in North(r')$. Then, r' is unable to move during the movement of r to $Goal$. When r reaches its destination $Goal$, $\Lambda_E(r)$ is above r', thus $r' \in South(r)$. Consequently, r and r' enter the configuration $North/South$ in a finite time.

From Lemma 5 and Lemma 6, we conclude that:

Theorem 2. *Any North/East configuration of two robots equipped with $\pi/8$-Inaccurate compasses is transformed after a finite time to gathering.*

4.3 Transition of *East/West* Configuration to Gathering

Lemma 7. *Given two robots r and r' at some time t_0, where r and r' are in the configuration East/West, with $r \in West(r')$ and $r' \in East(r)$, then the destination Goal$'$ of r' (resulting from its side move down) belongs to $East(r)$ or $South(r)$.*

Proof (Lemma 7). Let $H' = \Psi_W(r') \cap \Lambda_S(r')$. Consider the triangle $\triangle(r, r', Goal')$, and let α, α' and β be the angles at r, r' and $Goal'$, respectively. By hypothesis, $\alpha' \leq \alpha_W = \pi/4$. Then, $\alpha + \beta \leq 3\pi/4$. By the algorithm, $\alpha \leq \beta$. Thus, $\alpha \leq 3\pi/8 < \pi/2$. Let $M = \Lambda_E(r) \cap \Lambda_S(r')$. Then, the angle $\widehat{r'rM} \leq \pi/4$ since r and r' are in the configuration *East/West*. It follows that if $Goal' \in \overline{H'M}$, then $Goal' \in East(r)$. Otherwise, $Goal' \in South(r)$.

Lemma 8. *Let r and r' be two robots that are in the configuration East/West, with $r' \in East(r)$, and $r \in West(r')$ at some time t_0. Then, there is a finite time $\bar{t}$ in which this configuration is transformed into North/East or North/South configuration. Moreover, r and r' cannot enter any other configuration except the North/East or North/South configuration.*

Proof (Lemma 8). We distinguish several cases depending on the movement of each robot. We assume that both r and r' always reach their final destinations. All other cases where r or r' end their moves before destination are easy to deduce from previous lemmas.

1. **r moves/ r' does not move:** By the algorithm, r will perform a *side move up*. Let *Goal* be the destination of r and $\bar{t}$ be the time when r reaches its target. At $\bar{t}$, we have $r' \in South(r)$ (since at $\bar{t}$, r' becomes below $\Lambda_E(r)$). In addition, by Lemma 4, $Goal \in North(r')$. Then, at $\bar{t}$, $r \in North(r')$. Consequently, r and r' become in the configuration *North/South* in a finite time.
2. **r' moves/ r does not move:** By the algorithm, r' will perform a *side move down*. Let $Goal'$ be its destination and $\bar{t'}$ be the time when r' reaches $Goal'$. At time $\bar{t'}$, r is above $\Lambda_W(r')$, thus $r \in North(r')$. In addition, by Lemma 7, $r' \in East(r)$ or $r' \in South(r)$ at $\bar{t'}$. Consequently, r and r' leave the configuration *East/West* in a finite number of steps, and become in the configuration *East/North* or *North/South*.
3. **both r and r' move:** By the algorithm, r will perform a *side move up* and r' will perform a *side move down*. Let *Goal* and $Goal'$ be their respective destinations and $\bar{t}$ and $\bar{t'}$ be the time when they end their moves, respectively. At $\bar{t}$, $\forall p$ that is below $\Lambda_E(r(\bar{t}))$, $p \in South(r)$. Since, at $\bar{t}$, $r' \in \overline{r'\,Goal'}$, and by Lemma 7, $Goal' \in East(r(t_0))$ or $Goal' \in South(r(t_0))$. Thus, $r' \in South(r)$ at $\bar{t}$ because $\Lambda_E(r(\bar{t}))$ is above $Goal'$ and r'.
   When r' reaches $Goal'$, r is above $\Lambda_W(r')$. Consequently, at $\bar{t'}$, $r \in North(r')$. Since, r and r' reach their respective target in a finite time, we hence conclude that they become in the configuration *North/South* in a finite time.

From Lemma 8, Lemma 5 and Theorem 2, we conclude:

Theorem 3. *Any East/West configuration of two robots equipped with $\pi/8$-Inaccurate compasses is transformed after a finite time to the gathering.*

4.4 Transition of *North/West* Configuration to Gathering

Lemma 9. *Given two robots r and r' at some time t_0, where r and r' are in the configuration North/West, with $r \in West(r')$ and $r' \in North(r)$, then the destination $Goal'$ of r' (resulting from its side move down) belongs to $East(r)$.*

The proof is very similar to the proof of Lemma 7, thus omitted here.

Lemma 10. *Let r and r' be two robots that are in the configuration North/West, with $r \in West(r')$, and $r' \in North(r)$ at some time t_0. Then, there is a finite time $\bar{t}$ in which this configuration is transformed into North/East or East/West or North/South configuration. Moreover, r and r' can not enter any other configuration except the North/East or East/West or North/South configuration.*

Proof (Lemma 10).
By the algorithm, r' will make a *side move down*. Let $Goal'$ be its destination. Then, by Lemma 9, $Goal' \in East(r)$. As long as $r' \in North(r)$, r remains stationary. While r' is moving toward its target, it crosses $East(r)$ sector. Then, r and r' become in the configuration *East/West* if $\Lambda_W(r')$ is still above r. Otherwise, they enter the configuration *North/East*, with $r \in North(r')$ if r' reaches $Goal'$ and r still did not move. Finally, r and r' enter the configuration *North/South* if r performs a look operation when $r' \in East(r)$, and moves to it destination. From Lemma 3, these transformations are done in a finite time, and the lemma holds.

From Lemma 10, Theorem 2 and Theorem 3, we conclude:

Theorem 4. *Any North/West configuration of two robots equipped with $\pi/8$-Inaccurate compasses is transformed after a finite time to gathering.*

4.5 Transition of *East/South* Configuration to Gathering

Lemma 11. *Let r and r' be two robots that are in the configuration East/South at some time t_0, with $r' \in East(r)$ and $r \in South(r')$ Then, there is a finite time t in which this configuration is transformed into North/South or North/East or East/West or the gathering.*

Proof (Lemma 11). By the algorithm, r' will make a *direct move* toward r, and r will make a *side move up*. Then, we distinguish several cases, depending on where each robot sees the other one, and where it ends its move. By using similar arguments as in previous lemmas, it is easy to show that r and r' shift to the *North/South* or *North/East* or *East/West* configuration or the gathering in a finite time.

From Lemma 5, Lemma 11, Theorem 2 and Theorem 3, we conclude that:

Theorem 5. *Any East/South configuration of two robots equipped with $\pi/8$-Inaccurate compasses is transformed in a finite time to gathering.*

Theorem 6. *In a system, with 2 anonymous, oblivious mobile robots relying on inaccurate compasses, the gathering problem is solvable in a finite time for $\pi/8$-Inaccurate compasses.*

Proof (Theorem 6).
Theorem 1 states the different valid configurations by the algorithm. Also, from Lemma 5, Theorem 2, Theorem 3, Theorem 4 and Theorem 5, any valid configuration is transformed into gathering in a finite time (see Fig. 3), thus the theorem holds.

5 Conclusion

In this paper, we concentrate on the gathering of autonomous mobile robots when their compasses are subject to errors. In particular, we have studied the solvability of the gathering of two asynchronous mobile robots in the face of compass inaccuracies, and relying on oblivious computations. We thus provided an algorithm that gathers in a finite number of steps, two asynchronous oblivious mobile robots equipped with compasses that can differ by as much as $\pi/4$.

The benefit of our algorithm is that we solve the problem with inaccurate compasses. Moreover, our algorithm is self-stabilizing and tolerates any number of transient errors. We can also argue that even with weaker compasses that fluctuate for some arbitrary periods, and eventually they become constant with bounded errors that are less than or equal to $\pi/4$, our algorithm is still valid and solves the problem in a finite time.

Acknowledgments

We are especially grateful to Hirotaka Ono, Matthias Wiesmann and Rami Yared for their insightful comments regarding this work.

References

1. Agmon, N., Peleg, D.: Fault-tolerant gathering algorithms for autonomous mobile robots. In: Proc. 15th Annual ACM-SIAM Symp. on Discrete Algorithms (SODA'04), Philadelphia, PA, USA (2004) 1070–1078
2. Ando, H., Oasa, Y., Suzuki, I., Yamashita, M.: Distributed memoryless point convergence algorithm for mobile robots with limited visibility. IEEE Trans. on Robotics and Automation **15**(5) (1999) 818–828
3. Cieliebak, M.: Gathering non-oblivious mobile robots. In: Proc. 6th Latin American Symp. on Theoretical Informatics (LATIN'04). (2004) 577–588
4. Cieliebak, M., Flocchini, P., Prencipe, G., Santoro, N.: Solving the robots gathering problem. In: Proc. Intl. Colloquium on Automata, Languages and Programming (ICALP'03). (2003) 1181–1196

5. Cohen, R., Peleg, D.: Convergence of Autonomous Mobile Robots With Inaccurate Sensors and Movements. In: Proc. 23rd International Symposium on Theoretical Aspects of Computer Science (STACS'06). LNCS (2006)
6. Défago, X., Gradinariu, M., Messika, S., Raipin-Parvédy, P.: Fault-tolerant and self-stabilizing mobile robots gathering. In: Proc. 20th Intl. Symp. on Distributed Computing (DISC'06). LNCS 4167 (2006) 46–60
7. Dolev, S.: Self-Stabilization. MIT Press (2000)
8. Fisher, M., J., Lynch, N., A., Paterson, M. S.: Impossibility of distributed consensus with one faulty process. J. ACM. **32**(2) (1985) 374–382
9. Flocchini, P., Prencipe, G., Santoro, N., Widmayer, P.: Gathering of asynchronous robots with limited visibility. Theor. Comput. Sci. **337**(1–3) (2005) 147–168
10. Imazu, H., Itoh, N., Katayama, Y., Inuzuka, N., Wada, K.: A Gathering Problem for Autonomous Mobile Robots with Disagreement in Compasses (in Japanese). In: 1st Workshop on Theoretical Computer Science in Izumo, Japan (2005) 43–46
11. Prencipe, G.: Instantaneous Actions vs. Full Asynchronicity: Controlling and Coordinating a Set of Autonomous Mobile Robots. In Proc. 7th Italian Conference on Theoretical Computer Science (ICTCS'01). (2001) 154–171
12. Prencipe, G.: Corda: Distributed coordination of a set of autonomous mobile robots. In: Proc. 4th European Research Seminar on Advances in Distributed Systems (ERSADS'01), Bertinoro, Italy (2001) 185–190
13. Prencipe, G.: On the feasibility of gathering by autonomous mobile robots. In: Proc. Colloquium on Structural Information and Communication Complexity (SIROCCO'05). (2005) 246–261
14. Schneider, M.: Self-stabilization. ACM Computing Surveys **25**(1) (1993) 45–67
15. Souissi, S., Défago, X., Yamashita, M.: Using Eventually Consistent Compasses to Gather Oblivious Mobile Robots with Limited Visibility. In: Proc. 8th Intl. Symp. on Stabilization, Safety, and Security of Distributed Systems (SSS'06). LNCS 4280 (2006) 471–487
16. Souissi, S., Défago, X., Yamashita, M.: Gathering Asynchronous Mobile Robots with Inaccurate Compasses. Research Report (JAIST), IS-RR-2006-014, Hokuriku, Japan (2006)
17. Suzuki, I., Yamashita, M.: Distributed anonymous mobile robots: Formation of geometric patterns. SIAM Journal on Computing **28**(4) (1999) 1347–1363
18. Tel, G.: Introduction to Distributed Algorithms. In: Cambridge University Press (2001)