Convergence Properties of the Gravitational Algorithm in Asynchronous Robot Systems

Reuven Cohen and David Peleg

Department of Computer Science and Applied Mathematics,
The Weizmann Institute of Science, Rehovot 76100, Israel
E-mail: {r.cohen,david.peleg}@weizmann.ac.il

Abstract. This paper considers the convergence problem in autonomous mobile robot systems. A natural algorithm for the problem requires the robots to move towards their center of gravity. Previously it was known that the gravitational algorithm converges in the synchronous or semi-synchronous model, and that two robots converge in the asynchronous model. The current paper completes the picture by proving the correctness of the gravitational algorithm in the fully asynchronous model for any number of robots. It also analyses its convergence rate, and establishes its convergence in the presence of crash faults.

1 Introduction

1.1 Background and motivation

Swarms of low cost robots provide an attractive alternative when facing various large-scale tasks in hazardous or hostile environments. Such systems can be made cheaper, more flexible and potentially resilient to malfunction. Indeed, interest in autonomous mobile robot systems arose in a variety of contexts (see [20, 14, 21, 15, 5, 17, 16, 18, 19, 4, 20, 21, 28] and the survey in [7, 6]).

Along with developments related to the physical engineering aspects of such robot systems, there have been recent research attempts geared at developing suitable algorithmics, particularly for handling the distributed coordination of multiple robots [3, 8, 9, 22, 24, 26, 27]. A number of computational models were proposed in the literature for multiple robot systems. In this paper we consider the fully asynchronous model of [12, 23, 9, 8]. In this model, the robots are assumed to be identical and indistinguishable, lack means of communication, and operate in Look-Compute-Move cycles. Each robot wakes up at unspecified times, observes its environment using its sensors (capable of identifying the locations of the other robots), performs a local computation determining its next move and moves accordingly.

Much of the literature on distributed control algorithms for autonomous mobile robots has concentrated on two basic tasks, called *gathering* and *convergence*. Gathering requires the robots to occupy a single point within finite time, regardless of their initial configuration. Convergence is the closely related task in which the robots are required to converge to a single point, rather than reach it. More precisely, for every $\epsilon > 0$ there must be a time t_ϵ by which all robots are within a distance of at most ϵ of each other.

A common and straightforward approach to these tasks relies on the robots in the swarm calculating some median position and moving towards it. Arguably the most natural variant of this approach is the one based on using the *center of gravity* (sometimes called also the *center of mass*, the *barycenter* or the average) of the robot group. This approach is easy to analyze in the synchronous model. In the asynchronous model, analyzing the process becomes more involved, since the robots operate at different rates and may take measurements at different times, including while other robots are in movement. The inherent asynchrony in operation might therefore cause various oscillatory effects on the centers of gravity calculated by the robots, preventing them from moving towards each other and possibly even cause them to diverge and stray away from each other in certain scenarios.

Several alternative, more involved, algorithms have been proposed in the literature for the gathering and convergence problems. The gathering problem was first discussed in [26, 27] in a semi-synchronous model, where the robots operate in cycles but not all robots are active in every cycle. It was proven therein that it is impossible to gather *two* oblivious autonomous mobile robots without a common orientation under the semi-synchronous model (although 2-robot convergence is easy to achieve in this setting). On the other hand, there is an algorithm for gathering $N \geq 3$ robots in the semi-synchronous model [27]. In the asynchronous model, an algorithm for gathering $N = 3, 4$ robots is presented in [23, 9], and an algorithm for gathering $N \geq 5$ robots has recently been described in [8]. The gathering problem was also studied in a system where the robots have limited visibility [13, 2]. Fault tolerant algorithms for gathering were studied in [1]. In a failure-prone system, a gathering algorithm is required to successfully gather the nonfaulty robots, independently of the behavior of the faulty ones. The paper presents an algorithm tolerant against a single crash failure in the asynchronous model. For Byzantine faults, it is shown therein that in the asynchronous model it is impossible to gather a 3-robot system, even in the presence of a single Byzantine fault. In the fully synchronous model, an algorithm is provided for gathering N robots with up to f faults, where $N \geq 3f + 1$.

Despite the existence of these elaborate gathering algorithms, the gravitational approach is still very attractive, for a number of reasons. To begin with, it requires only very simple and efficient calculations, which can be performed on simple hardware with minimal computational efforts. It can be applied equally easily to any number of dimensions and to any swarm size. Moreover, the errors it incurs due to rounding are bounded and simple to calculate. In addition, it is oblivious (i.e., it does not require the robots to store any information on their previous operations or on past system configurations). This makes the method both memory-efficient and self-stabilizing (meaning that following a finite number of transient errors that change the states of some of the robots into other (possibly illegal) states, the system returns to a legal state and achieves eventual convergence). Finally, the method avoids deadlocks, in the sense that every robot can move at any given position (unless it has already reached the center of gravity). These advantages may well make the gravitational algorithm the method of choice in many practical situations.

Subsequently, it is interesting to study the correctness and complexity properties of the gravitational approach to convergence. This study is the focus of the current paper. Preliminary progress was made by us in [10], where we proved the correctness of the gravitational algorithm in the semi-synchronous model of [26]. In the asynchronous model, we provided a convergence proof for the special case of a 2-robot system. In the current paper, we complete the picture by proving a general theorem about the convergence of the center of gravity algorithm in the fully asynchronous model. We also analyze the convergence rate of the algorithm. Finally, we establish convergence in the crash fault model. Specifically, we show that in the presence of f crash faults, $1 \leq f \leq N-2$, the $N-f$ nonfaulty robots will converge to the center of gravity of the crashed robots.

1.2 The model

The basic model studied in [3, 8, 9, 22, 24, 26, 27] can be summarized as follows. The N robots execute a given algorithm in order to achieve a prespecified task. Each robot i in the system operates individually, repeatedly going through simple cycles consisting of three steps:

- **Look**: Identify the locations of all robots in i's private coordinate system and obtain a multiset of points $P = \{p_1, \ldots, p_N\}$ defining the current *configuration*. The robots are indistinguishable, so i knows its own location p_i but does not know the identity of the robots at each of the other points. This model allows robots to detect multiplicities, i.e., when two or more robots reside at the same point, all robots will detect this fact. Note that this model is stronger than, e.g., the one of [8].
- **Compute**: Execute the given algorithm, resulting in a goal point p_G.
- **Move**: Move on a straight line towards the point p_G. The robot might stop before reaching its goal point p_G, but is guaranteed to traverse a distance of at least S (unless it has reached the goal). The value of S is not assumed to be known to the robots, and they cannot use it in their calculations.

The "look" and "move" operations are identical in every cycle, and the differences between various algorithms are in the "compute" step. The procedure carried out in the "compute" step is identical for all robots.

In most papers in this area (cf. [25, 26, 13, 9]), the robots are assumed to be rather limited. To begin with, the robots are assumed to have no means of directly communicating with each other. Moreover, they are assumed to be *oblivious* (or memoryless), namely, they cannot remember their previous states, their previous actions or the previous positions of the other robots. Hence the algorithm used in the "compute" step cannot rely on information from previous cycles, and its only input is the current configuration. While this is admittedly an over-restrictive and unrealistic assumption, developing algorithms for the oblivious model still makes sense in various settings, for two reasons. First, solutions that rely on non-obliviousness do not necessarily work in a dynamic environment where the robots are activated in different cycles, or robots might be added to or removed from the system dynamically. Secondly, any algorithm that works correctly for oblivious robots is inherently self-stabilizing, i.e., it withstands transient errors that alter the robots' local states into other (possibly illegal) states.

We consider the *fully asynchronous* timing model (cf. [8, 9]). In this model, robots operate on their own (time-varying) rates, and no assumptions are made regarding the relative speeds of different robots. In particular, robots may remain inactive for arbitrarily long periods between consecutive operation cycles (subject to some "fairness" assumption that ensures that each robot is activated infinitely often in an infinite execution).

To describe the center of gravity algorithm, hereafter named Algorithm `Go_to_COG`, we use the following notation. Denote by $\bar{r}_i[t]$ the location of robot i at time t. Denote the true center of gravity at time t by $\bar{c}[t] = \frac{1}{N}\sum_{i=1}^{N} \bar{r}_i[t]$. Denote by $\bar{c}_i[t]$ the center of gravity as last calculated by the robot i before or at time t, i.e., if the last calculation by i was done at time $t' \leq t$ then $\bar{c}_i[t] = \bar{c}[t']$. Note that, as mentioned before, robot i calculates this location in its own private coordinate system; however, for the purpose of describing the algorithm and its analysis, it is convenient to represent these locations in a unified global coordinate system (which of course is unknown to the robots themselves). This is justified by the linearity of the center of gravity calculation, which renders it invariant under any linear transformation. By convention $\bar{c}_i[0] = \bar{r}_i[0]$ for all i.

Algorithm `Go_to_COG` is very simple. After measuring the current configuration at some time t, the robot i computes the average location of all robot positions (including its own), $\bar{c}_i[t] = \sum_j \bar{r}_j[t]/N$, and then proceeds to move towards the calculated point $\bar{c}_i[t]$. (As mentioned earlier, the move may terminate before the robot actually reaches the point $\bar{c}_i[t]$, but in case the robot has not reached $\bar{c}_i[t]$, it must has traversed a distance of at least S.)

2 Asynchronous convergence

This section proves our main result, namely, that Algorithm `Go_to_COG` guarantees the convergence of N robots for any $N \geq 2$ in the asynchronous model.

As noted in [10], the convex hull of the robot locations and calculated centers of gravity cannot increase in time. Intuitively, this is because (1) while a robot i performs a Move operation starting at time t_0, $\bar{c}_i$ does not change throughout the move and $\bar{r}_i[t]$ remains on the line segment $[\bar{r}_i[t_0],\bar{c}_i[t_0]]$, which is contained in the convex hull, and (2) when a robot performs a Look step, the calculated center of gravity is inside the convex hull of the N robot locations at that time. Hence we have the following.

Lemma 1. [10] *If for some time t_0, $\bar{r}_i[t_0]$ and $\bar{c}_i[t_0]$ for all i reside in a closed convex curve, $\mathcal{P}$, then for every time $t > t_0$, $\bar{r}_i[t]$ and $\bar{c}_i[t]$ also reside in $\mathcal{P}$ for every $1 \leq i \leq N$.*

Hereafter we assume, for the time being, that the configuration is one-dimensional, i.e., the robots reside on the x-axis. Later on, we extend the convergence proof to d-dimensions by applying the result to each dimension separately.

For every time t, let $H[t]$ denote the convex hull of the points $\{\bar{r}_i[t] \mid 1 \leq i \leq N\} \cup \{\bar{c}_i[t] \mid 1 \leq i \leq N\}$, namely, the smallest closed interval containing all $2N$ points.

Corollary 1. *For $N \geq 2$ robots and times t_1, t_0, if $t_1 > t_0$ then $H[t_1] \subseteq H[t_0]$.*

Unfortunately, it is hard to prove convergence on the basis of h alone, since it is hard to show that h strictly decreases. Other potentially promising measures, such as ϕ_1 and ϕ_2 defined next, also prove problematic as they might sometimes increase in certain scenarios. Subsequently, the measure ψ we use in what follows to prove strict convergence is defined as a combination of a number of different measures. Formally, let us define the following quantities.

$$\begin{aligned}
\phi_1[t] &= \sum_{i=1}^{N} |\bar{c}[t] - \bar{c}_i[t]| \ , \\
\phi_2[t] &= \sum_{i=1}^{N} |\bar{c}_i[t] - \bar{r}_i[t]| \ , \\
\phi[t] &= \phi_1[t] + \phi_2[t] \ , \\
h[t] &= |H[t]| \ , \\
\psi[t] &= \frac{\phi[t]}{2N} + h[t] \ .
\end{aligned}$$

We now claim that ϕ, h and ψ are nonincreasing functions of time.

Lemma 2. *For every $t_1 > t_0$, $\phi[t_1] \leq \phi[t_0]$.*

Proof: Examine the change in ϕ due to the various robot actions. If a Look operation is performed by robot i at time t, then $\bar{c}[t] - \bar{c}_i[t] = 0$ and $|\bar{r}_i[t] - \bar{c}_i[t]| = |\bar{c}_i[t^*] - \bar{c}[t]|$ for any $t^* \in [t', t]$, where t' is the end of the last move performed by robot i. Therefore, ϕ is unchanged by the Look performed.

Now consider some time interval $[t'_0, t'_1] \subseteq [t_0, t_1]$, such that no look operations were performed during $[t'_0, t'_1]$. Suppose that during this interval each robot i moved a distance Δ_i (where some of these distances may be 0). Then ϕ_2 decreased by $\sum_i \Delta_i$, the maximum change in the center of gravity is $|\bar{c}[t_1] - \bar{c}[t_0]| \leq \sum_i \Delta_i / N$, and the robots' calculated centers of gravity have not changed. Therefore, the change in ϕ_1 is at most $\phi_1[t_1] - \phi_1[t_0] \leq \sum_i \Delta_i$. Hence, the sum $\phi = \phi_1 + \phi_2$ cannot increase. ∎

By Lemma 2 and Cor. 1, respectively, ϕ and h are nonincreasing. Hence we have:

Lemma 3. *ψ is a nonincreasing function of time.*

Lemma 4. *For all t, $h \leq \psi \leq 2h$.*

Proof: The lower bound is trivial. For the upper bound, notice that ϕ is the sum of $2N$ summands, each of which is at most h (since they all reside in the segment). ∎

We now state a lemma which allows the analysis of the change in ϕ (and therefore also ψ) in terms of the contributions of individual robots.

Lemma 5. *If by the action of a robot i separately, in the time interval $[t_0, t_1]$ its contribution to ϕ is δ_i, then $\phi[t_1] \leq \phi[t_0] + \delta_i$.*

Proof: Lemma 2 implies that "look" actions have no effect on ϕ and therefore can be disregarded. A robot moving a distance Δ_i will always decrease its term in ϕ_2 by Δ_i, and the motions of other robots have no effect on this term. Denote by Δ_j the motions of the other robots. Notice that

$$|\bar{c} + \frac{\Delta_i}{N} + \frac{1}{N}\sum_{j \neq i} \Delta_j - \bar{c}_k| \;\leq\; |\bar{c} + \frac{\Delta_i}{N} - \bar{c}_k| + \frac{1}{N}\sum_{j \neq i} |\Delta_j|.$$

The function ϕ_1 contains N summands, each of which contains a contribution of at most $\frac{1}{N}|\Delta_j|$ from every robot $j \neq i$. Therefore, the total contribution of each robot to ϕ_1 is at most $|\Delta_j|$, which is canceled by the negative contribution of $|\Delta_j|$ to ϕ_2. ∎

Theorem 1. *For every time t_0, there exists some time $\hat{t} > t_0$ such that*

$$\psi[\hat{t}] \leq \left(1 - \frac{1}{8N^2}\right) \psi[t_0].$$

Proof: Assume without loss of generality that at time t_0, the robots and centers of gravity resided in the interval $H[t_0] = [0, 1]$ (and thus $h[t_0] = 1$ and $\psi[t_0] \leq 2$). Take t^* to be the time after t_0 when each robot has completed at least one entire Look–Compute–Move cycle. There are now two possibilities:

1. Every center of gravity $\bar{c}_i[t']$ that was calculated at time $t' \in [t, t^*]$ resided in the segment $(\frac{1}{2N}, 1]$. In this case at time t^* no robot can reside in the segment $[0, \frac{1}{2N}]$ (since every robot has completed at least one cycle operation, where it has arrived at its calculated center of gravity outside the segment $[0, \frac{1}{2N}]$, and from then on it may have moved a few more times to its newly calculated centers of gravity, which were also outside this segment). Hence at time $\hat{t} = t^*$ all robots and centers of gravity reside in $H[\hat{t}] \subseteq [\frac{1}{2N}, 1]$, so $h[\hat{t}] \leq 1 - \frac{1}{2N}$, and $\phi[\hat{t}] \leq \phi[t_0]$. Therefore,

   $$\psi[\hat{t}] \;=\; \frac{\phi[\hat{t}]}{2N} + h[\hat{t}] \;\leq\; \frac{\phi[t_0]}{2N} + 1 - \frac{1}{2N} \;=\; \psi[t_0] - \frac{1}{2N}\;.$$

   Also, by Lemma 4, $\psi[t] \leq 2$, hence $\frac{1}{2N} \geq \frac{1}{4N}\psi[t_0]$. Combined, we get $\psi[\hat{t}] \leq \left(1 - \frac{1}{4N}\right) \psi[t_0]$.

2. For some $t_1 \in [t, t^*]$, the center of gravity $\bar{c}_i[t_1] = \frac{1}{N}\sum_{j=1}^{N} \bar{r}_j[t_1]$ calculated by some robot i at time t_1 resided in $[0, \frac{1}{2N}]$. Therefore, at time t_1 all robots resided in the segment $[0, \frac{1}{2}]$ (by Markov inequality [11]). We split again into two subcases:
   (a) At time t_1 all centers of gravity, $\bar{c}_i[t_1]$ resided in the interval $[0, \frac{3}{4}]$. In this case, take $\hat{t} = t^*$, $h[\hat{t}] \leq h[t_1] \leq \frac{3}{4} \leq \frac{3}{4}h[t_0]$, and therefore

$$\psi[\hat{t}] = h[\hat{t}] + \frac{\phi[\hat{t}]}{2N} \leq \frac{3}{4}h[t_0] + \frac{\phi[t_0]}{2N} \leq \frac{7}{8}h[t_0] + \frac{7}{8}\frac{\phi[t_0]}{2N} = \frac{7}{8}\psi[t_0],$$

   where the last inequality is due to the fact that $\frac{\phi[t_0]}{2N} \leq h[t_0]$ as argued in the proof of Lemma 4. The theorem immediately follows.
   (b) At time t_1 there existed robots with $\bar{c}_i[t_1] > \frac{3}{4}$. In this case, take k to be the robot with the highest center of gravity (or one of them) and take $\hat{t}$ to be the time robot k completes its next move. Its move size is at least $\Delta_k \geq \frac{1}{4}$, hence its motion decreased $|\bar{r}_k - \bar{c}_k|$ by $\Delta_k \geq \frac{1}{4}$, and ϕ_2 is decreased by Δ_k. Also, The robots and calculated centers of gravity are now restricted to the segment $[0, \bar{c}_k]$. Since the true center of gravity must be to the left of $\bar{c}_k$, $\bar{c} - \bar{c}_k$ is decreased by Δ_k/N. The sum of all other summands in ϕ_1 may increase by at most $(N-1)\Delta_k/N$. By Lemma 5 ϕ can be bounded from above by the contribution of a single robot. Therefore, ϕ decreased by at least $2\Delta_k/N$, and the term $\frac{\phi}{2N}$ in ψ decreased by at least $\frac{2\Delta_k/N}{2N} = \frac{\Delta_k}{N^2} \geq \frac{1}{4N^2}$. As $\psi[t_0] \leq 2$ and h is nonincreasing, ψ decreased by at least $\frac{1}{4N^2} \geq \frac{\psi[t_0]}{8N^2}$. The theorem follows.
   ∎

To prove the convergence of the gravitational algorithm in d-dimensional Euclidean space, apply Theorem 1 to each dimension separately. Observe that by Theorem 1 and Lemma 4, for every $\epsilon > 0$ there is a time t_ϵ by which $h[t_\epsilon] \leq \psi[t_\epsilon] \leq \epsilon$, hence the robots have converged to an ϵ-neighborhood.

Theorem 2. *For any $N \geq 2$, in d-dimensional Euclidean space, N robots performing Algorithm* `Go_to_COG` *will converge.*

3 Convergence rate

To bound the rate of convergence in the fully asynchronous model, one should make some normalizing assumption on the operational speed of the robots. A standard type of assumption is based on defining the maximum length of a robot cycle during the execution (i.e., the maximum time interval between two consecutive Look steps of the same robot) as one time unit. For our purposes it is more convenient to make the slightly modified assumption that for every time t, during the time interval $[t, t+1]$ every robot has completed at least one cycle. Note that the two assumptions are equivalent up to a constant factor of 2. Note also that this assumption is used only for the purpose of complexity analysis, and was not used in our correctness proof.

Lemma 6. *For every time interval* $[t_0, t_1]$,

$$\psi[t_1] \;\leq\; \left(1 - \frac{1}{8N^2}\right)^{\lfloor \frac{t_1 - t_0}{2} \rfloor} \psi[t_0].$$

Proof: Consider the different cases analyzed in the proof of Theorem 1. By our timing assumption, we can take $t^* = t_0 + 1$. In case 1, ψ is decreased by a factor of $1 - \frac{1}{4N}$ by time $\hat{t} = t^*$, i.e., within one time unit. In case 2a, ψ is decreased by a factor of $\frac{7}{8}$ by time $\hat{t} = t^*$, i.e., within one time unit again. The slowest convergence rate is obtained in case 2b. Here, we can take $\hat{t} = t^* + 1 = t_0 + 2$, and conclude that ψ is decreased by a factor of $1 - \frac{1}{8N^2}$ after two time units. The lemma follows by assuming a worst case scenario in which during the time interval $[t_0, t_1]$, the "slow" case 2b is repeated for $\lfloor \frac{\hat{t} - t_0}{2} \rfloor$ times. ∎

By the two inequalities of Lemma 4 we have that $h[t_1] \leq \psi[t_1]$ and $\psi[t_0] \leq 2h[t_0]$, respectively. Lemma 6 now yields the following.

Theorem 3. *For every time interval* $[t_0, t_1]$,

$$h[t_1] \;\leq\; 2\left(1 - \frac{1}{8N^2}\right)^{\lfloor \frac{t_1 - t_0}{2} \rfloor} h[t_0].$$

Corollary 2. *In any execution of the gravitational algorithm in the asynchronous model, over every interval of* $O(N^2)$ *time units, the size of the d-dimensional convex hull of the robot locations and centers of gravity is halved in each dimension separately.* ∎

An example for slow convergence is given by the following lemma.

Lemma 7. *There exist executions of the gravitational algorithm in which* $\Omega(N)$ *time is required to halve the convex hull of* N *robots in each dimension.*

Proof: Initially, and throughout the execution, the N robots are organized on the x axis. The execution consists of phases of the following structure. Each phase takes exactly one time unit, from time t to time $t+1$. At each (integral) time t, robot 1 is at one endpoint of the bounding segment $H[t]$ while the other $N-1$ robots are at the other endpoint of the segment. Robot 1 performs a look at time t and determines its perceived center of gravity $\bar{c}_i[t]$ to reside at a distance $\frac{h[t]}{N-1}$ from the distant endpoint. Next, the other $N-1$ robots perform a long sequence of (fast) cycles, bringing them to within a distance ϵ of robot 1, for arbitrarily small ϵ. Robot 1 then performs its move to its perceived center of gravity $\bar{c}_i[t]$. Hence the decrease in the size of the bounding interval during the phase is $h[t] - h[t+1] = \frac{h[t]}{N-1} + \epsilon$, or in other words, at the end of the phase $h[t+1] \approx \left(1 - \frac{1}{N-1}\right) h[t]$. It follows that $O(N)$ steps are needed to reduce the interval size to a $1/e$ fraction of its original size. ∎

Note that there is still a linear gap between the upper and lower bounds on the convergence rate of the gravitational algorithm as stated in Cor. 2 and Lemma 7.

It is interesting to compare these bounds with what happens in a variant of the fully synchronous model (cf. [27]), in which all robots operate at fixed time cycles, and the Look phase of all robots is simultaneous. It is usually assumed that the robots do not necessarily reach their desired destination. However, there is some constant minimum distance, S, which the robots are guaranteed to traverse at every step. Therefore, if $H[0]$ is the convex hull of the N robots at time 0 and $h[0]$ is the maximum width of $H[0]$ in any of the d dimensions, then we have the following.

Lemma 8. *In any execution of the gravitational algorithm in the fully synchronous model, the robots achieve gathering in at most* $\lceil 4h[0]d^{3/2}/S \rceil$ *time.*

Proof: If the distance of each robot from the center of gravity is at most S, then at the next step they will all gather. Suppose now that there exists at least one robot whose distance from the center of gravity is greater than S. Since the center of gravity is within the convex hull, the largest dimension is at least $h[0] \geq S/\sqrt{d}$. Without loss of generality, assume that the projection of the hull on the maximum width dimension is on the interval $[0, a]$, and that the projection of the center of gravity $\bar{c}[0]$ is in the interval $[\frac{a}{2}, a]$. Then in each step, every robot moves by a vector $\min\{\bar{r}_i - \bar{c}, S' \frac{\bar{r}_i - \bar{c}}{|\bar{r}_i - \bar{c}|}\}$ for some $S' \geq S$. By assumption, a is the width of the largest dimension and therefore $a \geq |\bar{r}_i - \bar{c}|/\sqrt{d}$. For every robot in the interval $[0, \frac{a}{4}]$, the distance to the current center of gravity will decrease in the next step by at least $\min\{\frac{a}{4}, S\frac{a/4}{a\sqrt{d}}\} \geq \frac{S}{4\sqrt{d}}$. Thus, the width of at least one dimension decreases by at least $\frac{S}{4\sqrt{d}}$ in each step. Therefore, gathering is achieved after at most $\lceil 4h[0]d^{3/2}/S \rceil$ cycles, independently of N. ∎

4 Fault tolerance

In this section we consider the behavior of the gravitational algorithm in the presence of possible robot failures.

Let us first consider a model allowing only *transient* failures. Such a failure causes a change in the states of some robots, possibly into illegal states. Notice that Theorem 2 makes no assumptions about the initial positions and centers of gravity, other than that they are restricted to some finite region. It follows that, due to the oblivious nature of the robots (and hence the algorithm), the robots will converge regardless of any finite number of transient errors occurring in the course of the execution.

We now turn to consider the *crash* fault model. This model, presented in [1], follows the common crash (or "fail-stop") fault model in distributed computing, and assumes that a robot may fail by halting. This may happen at any point in time during the robot's cycle, i.e., either during the movement towards the goal point or before it has started. Once a robot has crashed, it will remain stationary indefinitely.

In [1], it is shown that in the presence of a single crash fault, it is possible to *gather* the remaining (functioning) robots to a common point. Here, we avoid the gathering requirement and settle for the weaker goal of convergence. We show that the Go_to_COG algorithm converges for every number of crashed robots. In fact, in a sense convergence is easier in this setting since the crashed robots determine the final convergence point for the nonfaulty robots. We have the following.

Theorem 4. *Consider a group of N robots that execute Algorithm Go_to_COG. If $1 \leq M \leq N-2$ robots crash during the execution, then the remaining $N-M$ robots will converge to the center of gravity of the crashed robots.*

Proof: Consider an execution of the gravitational algorithm by a group of N robots. Without loss of generality assume that the crashed robots were $1, \ldots, M$, and their crashing times were $t_1 \leq \ldots \leq t_M$, respectively. Consider the behavior of the algorithm starting from time t_M. For the analysis, a setting in which the M robots crashed at general positions $\bar{r}_1, \ldots, \bar{r}_M$ is equivalent to one in which all M crashed robots are concentrated in the center of gravity $\frac{1}{M}\sum_{i=1}^{M} \bar{r}_i$. Assume, without loss of generality, that this center of gravity is at 0.

Now consider some time $t_0 \geq t_M$. Let $H[t_0] = [a, b]$ for some $a \leq 0 \leq b$. By Corollary 1, the robots will remain in the segment $[a, b]$ at all times $t \geq t_0$. The center of gravity calculated by any nonfaulty robot $M+1 \leq j \leq N$ at time $t \geq t_0$ will then be

$$\bar{c}_j[t] \;=\; \frac{1}{N}\sum_{i=1}^{N} \bar{r}_i \;=\; \frac{1}{N}\left(M \cdot 0 + \sum_{i=M+1}^{N} \bar{r}_i[t]\right).$$

Hence, all centers of gravity calculated hereafter will be restricted to the segment $[a', b']$ where $a' = \frac{N-M}{N} \cdot a$ and $b' = \frac{N-M}{N} \cdot b$. Consequently, denoting by $\hat{t}$ the time by which every nonfaulty robot has completed a Look-Compute-Move cycle, we have that $H[\hat{t}] \subseteq [a', b']$, hence $h[\hat{t}] \leq \frac{N-M}{N} \cdot h[t_0]$. Again, the argument can be extended to any number of dimensions by considering each dimension separately. It follows that the robots converge to a point. ∎

References

1. N. Agmon and D. Peleg. Fault-tolerant gathering algorithms for autonomous mobile robots. In *Proc. 15th ACM-SIAM Symp. on Discrete Algorithms*, pages 1063–1071, January 2004.
2. H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita. A distributed memoryless point convergence algorithm for mobile robots with limited visibility. *IEEE Trans. Robotics and Automation*, 15:818–828, 1999.
3. H. Ando, I. Suzuki, and M. Yamashita. Formation and agreement problems for synchronous mobile robots with limited visibility. In *Proc. IEEE Symp. of Intelligent Control*, pages 453–460, August 1995.

4. T. Balch and R. Arkin. Behavior-based formation control for multi-robot teams. *IEEE Trans. on Robotics and Automation*, 14, December 1998.
5. G. Beni and S. Hackwood. Coherent swarm motion under distributed control. In *Proc. DARS'92*, pages 39–52, 1992.
6. Y.U. Cao, A.S. Fukunaga, and A.B. Kahng. Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, 4(1):7–23, March 1997.
7. Y.U. Cao, A.S. Fukunaga, A.B. Kahng, and F. Meng. Cooperative mobile robots: Antecedents and directions. In *Proc. Int. Conf. of Intel. Robots and Sys.*, pages 226–234, 1995.
8. M. Cieliebak, P. Flocchini, G. Prencipe, and N. Santoro. Solving the robots gathering problem. In *Proc. 30th Int. Colloq. on Automata, Languages and Programming*, pages 1181–1196, 2003.
9. M. Cieliebak and G. Prencipe. Gathering autonomous mobile robots. In *Proc. 9th Int. Colloq. on Structural Information and Communication Complexity*, pages 57–72, June 2002.
10. R. Cohen and D. Peleg. Robot convergence via center-of-gravity algorithms. In *Proc. 11th Colloq. on Structural Information and Communication Complexity*, volume LNCS 3104, pages 79–88. Springer-Verlag, 2004.
11. W. Feller. *An introduction to Probability Theory and its Applications.* Wiley, New York, 1968.
12. P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer. Hard tasks for weak robots: The role of common knowledge in pattern formation by autonomous mobile robots. In *Proc. 10th Int. Symp. on Algorithms and Computation*, pages 93–102, 1999.
13. P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer. Gathering of autonomous mobile robots with limited visibility. In *Proc. 18th Symp. on Theoretical Aspects of Computer Science*, pages 247–258, February 2001.
14. D. Jung, G. Cheng, and A. Zelinsky. Experiments in realising cooperation between autonomous mobile robots. In *Proc. Int. Symp. on Experimental Robotics*, June 1997.
15. Y. Kawauchi, M. Inaba, and T. Fukuda. A principle of decision making of cellular robotic system (CEBOT). In *Proc. IEEE Conf. on Robotics and Automation*, pages 833–838, 1993.
16. M.J. Mataric. *Interaction and Intelligent Behavior.* PhD thesis, MIT, 1994.
17. S. Murata, H. Kurokawa, and S. Kokaji. Self-assembling machine. In *Proc. IEEE Conf. on Robotics and Automation*, pages 441–448, 1994.
18. L.E. Parker. Designing control laws for cooperative agent teams. In *Proc. IEEE Conf. on Robotics and Automation*, pages 582–587, 1993.
19. L.E. Parker. On the design of behavior-based multi-robot teams. *J. of Advanced Robotics*, 10, 1996.
20. L.E. Parker and C. Touzet. Multi-robot learning in a cooperative observation task. In *Distributed Autonomous Robotic Systems 4*, pages 391–401, 2000.
21. L.E. Parker, C. Touzet, and F. Fernandez. Techniques for learning in multi-robot teams. In T. Balch and L.E. Parker, editors, *Robot Teams: From Diversity to Polymorphism.* A. K. Peters, 2001.
22. G. Prencipe. CORDA: Distributed coordination of a set of atonomous mobile robots. In *Proc. 4th European Research Seminar on Advances in Distributed Systems*, pages 185–190, May 2001.
23. G. Prencipe. *Distributed Coordination of a Set of Atonomous Mobile Robots.* PhD thesis, Universita Degli Studi Di Pisa, 2002.

24. K. Sugihara and I. Suzuki. Distributed algorithms for formation of geometric patterns with many mobile robots. *Journal of Robotic Systems*, 13(3):127–139, 1996.
25. I. Suzuki and M. Yamashita. Agreement on a common x-y coordinate system by a group of mobile robots. In *Proc. Dagstuhl Seminar on Modeling and Planning for Sensor-Based Intelligent Robots*, September 1996.
26. I. Suzuki and M. Yamashita. Distributed anonymous mobile robots - formation and agreement problems. In *Proc. 3rd Colloq. on Structural Information and Communication Complexity*, pages 313–330, 1996.
27. I. Suzuki and M. Yamashita. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM J. on Computing*, 28:1347–1363, 1999.
28. I.A. Wagner and A.M. Bruckstein. From ants to a(ge)nts. *Annals of Mathematics and Artificial Intelligence*, 31, special issue on ant-robotics:1–5, 1996.