SIAM J. COMPUT.
Vol. 34, No. 6, pp. 1516–1528

CONVERGENCE PROPERTIES OF THE GRAVITATIONAL ALGORITHM IN ASYNCHRONOUS ROBOT SYSTEMS*

REUVEN COHEN† AND DAVID PELEG†

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. This paper proves the correctness of the gravitational algorithm in the fully asynchronous model. It also analyzes its convergence rate and establishes its convergence in the presence of crash faults.

Key words. robot swarms, autonomous mobile robots, convergence

AMS subject classifications. 68Q22, 70B15

DOI. 10.1137/S0097539704446475

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 [4, 5, 13, 14, 15, 16, 17, 18, 19, 20, 27] and the survey in [6, 7]).

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, 21, 23, 25, 26]. A number of computational models were proposed in the literature for multiple robot systems. In this paper we consider the fully asynchronous model of [8, 9, 11, 22]. 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 from 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 the *center of mass*, the *barycenter*, or the average) of the robot swarm. This approach is easy to analyze in the synchronous model. In the asynchronous

*Received by the editors October 22, 2004; accepted for publication (in revised form) April 18, 2005; published electronically August 31, 2005. Parts of this paper appeared in preliminary form in *Proceedings of the 12th Annual European Symposium on Algorithms*, Bergen, Norway, 2004.

http://www.siam.org/journals/sicomp/34-6/44647.html

†Department of Computer Science and Applied Mathematics, The Weizmann Institute of Science, Rehovot 76100, Israel (r.cohen@weizmann.ac.il, david.peleg@weizmann.ac.il). The work of the first author was supported by the Pacific Theaters Foundation. The work of the second author was supported by the Israel Science Foundation (grant 693/04).

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 causing 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 [25, 26] in a semisynchronous 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 semisynchronous 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 semisynchronous model [26]. In the asynchronous model, an algorithm for gathering $N = 3, 4$ robots is presented in [9, 22], 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 [2, 12]. 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. We prove 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, 21, 23, 25, 26] 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 (instantaneously) a multiset of points $P = \{p_1, \dots, p_N\}$ 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.

This model, which allows the robot to suddenly stop short of reaching its goal point, is henceforth referred to as the *sudden-stop* model. The simpler model where the robot always reaches p_G is henceforth called the *undisturbed-motion* model. For simplicity of presentation, we will prove some of our claims first in the undisturbed-motion model and then extend the proof to the full (sudden-stop) model.

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. [9, 12, 24, 25]), 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 overrestrictive and unrealistic assumption, developing algorithms for the oblivious model still makes sense in various settings for two reasons. First, solutions that rely on nonobliviousness 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. Second, 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 mainly the *fully asynchronous* ($\mathcal{ASYNC}$) 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). This feature is sometimes modeled by incorporating a *Wait* phase in each operation cycle of the robots.

We find it instructive to compare the performance of the gravitational algorithm in this model with its performance in two alternative models studied in the literature, namely, the *semisynchronous* ($\mathcal{SSYNC}$) model (cf. [25]) and the *fully synchronous* ($\mathcal{FSYNC}$) model (cf. [26]). In the fully synchronous model, all robots operate at fixed time cycles, and the Look phase of all robots is simultaneous. In the commonly studied intermediate semisynchronous model, it is again assumed that there are fixed time cycles, and the Look phase of the robots is simultaneous. However, now only a subset of the robots may wake up at every cycle.

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 $t' \le t$, 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 as its goal point 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 goal point $\bar{c}_i[t]$. A formal definition of this algorithm follows.

Algorithm Go_to_COG (code for robot i at time t):
1. Calculate the center of gravity, $\bar{c}_i[t] = \frac{1}{N}\sum_j \bar{r}_j[t]$.
2. Move to the point $\bar{c}_i[t]$.

As mentioned earlier, the move may terminate before the robot i actually reaches the point $\bar{c}_i[t]$. The point at which the robot does stop its movement is henceforth referred to as its *destination point*. More formally, define the destination point $\bar{\gamma}_i[t]$ of robot i to be the final point of the movement made by i following the last Look performed by i before or at time t. Note that at time t, robot i may not have computed its goal point $\bar{c}_i[t]$ yet, and, even if it had, it has no knowledge of the possibility of sudden stops; hence it is unaware of its destination point $\bar{\gamma}_i[t]$. Nevertheless, for the analysis we may treat this point as given at the moment of the Look action. Recall also that even in case the robot i has not reached $\bar{c}_i[t]$, it must have 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 $N \ge 2$ in the asynchronous model. Notice that in the $\mathcal{ASYNC}$ model, since no guarantees are given as to the behavior of the robot's location and velocity at the duration of the Move phase, the Compute and Wait phases may be assimilated into the Move phase and treated as time periods during the Move phase in which the robot progresses with zero velocity. Hence in the analysis we may consider only the Look and Move phases.

LEMMA 2.1. *If for some time t_0, $\bar{r}_i[t_0]$ and $\bar{\gamma}_i[t_0]$ for all i reside in the interior of a closed convex curve, $\mathcal{P}$, then for every time $t > t_0$, $\bar{r}_i[t]$ and $\bar{\gamma}_i[t]$ also reside in the interior of $\mathcal{P}$ for every $1 \le i \le N$.*

Proof. For the Move operation, it is clear that if for some i, $\bar{r}_i[t_0]$ and $\bar{\gamma}_i[t_0]$ both reside in the interior of a convex $\mathcal{P}$, then for the rest of the Move operation $\bar{\gamma}_i[t] = \bar{\gamma}_i[t_0]$ does not change and $\bar{r}_i[t]$ remains on the segment $[\bar{r}_i[t_0],\bar{\gamma}_i[t_0]]$, which is inside $\mathcal{P}$.

For the Look step, the claim is obvious for the $\bar{r}_i[t]$ values. We first argue that the calculated centers of gravity $\bar{c}_i[t]$, for every i, also reside in the interior of $\mathcal{P}$. If $N = 2$, then the calculated center of gravity is on the line segment connecting both robots and therefore respects convexity. For $N > 2$ robots, the center of gravity is on the line connecting the center of gravity of $N - 1$ robots and the Nth robot, and the claim follows by induction. The lemma now follows since the destination $\bar{\gamma}_i[t]$ of robot i is on the line segment connecting the robot location $\bar{r}_i[t]$ and its calculated center of gravity $\bar{c}_i[t]$ and therefore also respects convexity. ☐

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 $\{\bar{r}_i[t] \mid 1 \le i \le N\} \cup \{\bar{\gamma}_i[t] \mid 1 \le i \le N\}$, namely, the smallest closed interval containing all $2N$ points. Lemma 2.1 yields the following.

COROLLARY 2.2. *For $N \ge 2$ 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 the size of H alone, since it is hard to show that it 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:

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

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

LEMMA 2.3. *For every $t_1 > t_0$, $\phi[t_1] \le \phi[t_0]$.*

Proof. Examine the change in ϕ due to the various robot actions. Suppose a Look operation is performed by robot i at time t. (If two or more robots perform a Look operation simultaneously, then we serialize these operations for the sake of analysis and consider their sequential effects.) Denote with a superscript ^{b} (respectively, ^{a}) the values of the robot locations and centers of gravity and the above quantities just before (resp., after) the (instantaneous) Look operation. Then $\bar{\gamma}_i^b[t] = \bar{r}_i^b[t] = \bar{r}_i^a[t]$. Moreover, $\bar{r}_j^a[t] = \bar{r}_j^b[t]$ and $\bar{\gamma}_j^a[t] = \bar{\gamma}_j^b[t]$ for every $j \ne i$; hence also $\bar{c}^a[t] = \bar{c}^b[t]$. These equalities imply the following. First, denoting the contributions of the robots $j \ne i$ to ϕ_1 and ϕ_2 by $\tilde{\phi}_1[t] = \sum_{j \ne i} |\bar{c}[t] - \bar{\gamma}_j[t]|$ and $\tilde{\phi}_2[t] = \sum_{j \ne i} |\bar{\gamma}_j[t] - \bar{r}_j[t]|$, respectively, these contributions do not change; namely, $\tilde{\phi}_1^a[t] = \tilde{\phi}_1^b[t]$ and $\tilde{\phi}_2^a[t] = \tilde{\phi}_2^b[t]$. Also, the contributions of robot i to ϕ_1^b and ϕ_2^b are $|\bar{c}^b[t] - \bar{\gamma}_i^b[t]| = |\bar{c}^b[t] - \bar{r}_i^b[t]|$ and $\bar{\gamma}_i^b[t] - \bar{r}_i^b[t] = 0$, respectively. Finally, the contributions of robot i to ϕ_1^a and ϕ_2^a are $|\bar{r}_i^a[t] - \bar{\gamma}_i^a[t]|$ and $|\bar{\gamma}_i^a[t] - \bar{c}^a[t]|$. Hence the total contribution of robot i to ϕ^a is the same as its contribution to ϕ^b, since the point $\bar{\gamma}_i^a[t]$ occurs on the segment $[\bar{r}_i^b[t], \bar{c}^b[t]] = [\bar{r}_i^a[t], \bar{c}^a[t]]$. (In the undisturbed-motion case the situation is even simpler, as the robot always reaches its calculated destination and therefore $\bar{\gamma}_i^a[t] = \bar{c}^b[t]$.) 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. ∎

LEMMA 2.4. *ψ is a nonincreasing function of time.*

Proof. By Lemma 2.3, ϕ is nonincreasing. By Corollary 2.2, h is nonincreasing. Therefore their sum is also nonincreasing. ∎

LEMMA 2.5. *For all t, $h[t] \leq \psi[t] \leq 2h[t]$.*

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

We now state a lemma which allows the analysis of the change in $\phi[t]$ (and therefore also $\psi[t]$) in terms of the contributions of individual robots. It is, in general, impossible to separate the change in $\phi[t]$ to contributions of individual robots. However, it is possible to bound the minimum decrease in $\phi[t]$ by the decrease caused by the motion of a single robot.

LEMMA 2.6. *If $H[t_0] = [0,1]$ and at some time $t \geq t_0$ all robots are in the interval $[0, 1/2]$ (i.e., $\bar{r}_i[t] \in [0, 1/2]$ for all i), then there exists a time $t_1 \geq t$, such that $\psi[t_1] \leq \left(1 - \frac{1}{8N^2}\right)\psi[t_0]$.*

Proof. We split the situation into two subcases:

1. At time t all destination points $\bar{\gamma}_i[t]$ resided in the interval $[0, \frac{3}{4}]$. In this case, take t_1 to be the time when each robot has completed at least one cycle. All robots and destinations are now in the interval $[0, \frac{3}{4}]$, and therefore $h \leq \frac{3}{4}$. Now $h[t_1] \leq \frac{3}{4} \leq \frac{3}{4}h[t_0]$, and therefore

$$\psi[t_1] = h[t_1] + \frac{\phi[t_1]}{2N} \leq \frac{3}{4}h[t_0] + \frac{\phi[t_0]}{2N} \leq \frac{7}{8}h[t_0] + \frac{7}{8}\frac{\phi[t_0]}{2N} = \frac{7}{8}\psi[t_0],$$

   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 2.5. The lemma immediately follows in this case.

2. At time t there existed robots with $\bar{\gamma}_i[t] > \frac{3}{4}$. In this case, take k to be the robot with the highest destination point (or one of them) and take t_1 to be the time robot k completes its next Move. Its Move size is at least $\Delta_k \geq \frac{1}{4}$. Suppose at some time interval $[t', t''] \subseteq [t, t_1]$ no Look was performed by any robot and that every robot, i, moved in this time interval by the vector $\bar{\delta}_i$. Now, all robots approached their destinations, so $\phi_2[t''] = \phi_2[t] - \frac{1}{N}\sum_i \delta_i$. The center of gravity was changed by the robots' motions to $\bar{c}[t''] = \bar{c}[t'] + \frac{1}{N}\sum_i \bar{\delta}_i$ and therefore also $\bar{c}[t''] - \bar{c}[t'] \leq \frac{1}{N}\sum_i \delta_i$. Since no Look operations occur in the time interval $[t', t'']$, no $\bar{\gamma}_i$ is changed, and for every i, $|\bar{c}[t''] - \bar{\gamma}_i[t'']| \leq |\bar{c}[t''] - \bar{\gamma}_i[t'']| + \frac{1}{N}\sum_i \delta_i$. For $i = k$,

$$\begin{aligned} |\bar{c}[t''] - \bar{\gamma}_k[t'']| &= \left|\bar{c}[t'] + \frac{\bar{\delta}_k}{N} + \frac{1}{N}\sum_{i\neq k}\bar{\delta}_i - \bar{\gamma}_k[t']\right| \\ &\leq \left|\bar{c}[t'] + \frac{\bar{\delta}_k}{N} - \bar{\gamma}_k[t']\right| + \frac{1}{N}\sum_{i\neq k}\delta_j\ . \end{aligned}$$

   Since robot k is approaching $\bar{\gamma}_k$ from the left, it follows that $\bar{r}_k + \bar{\delta}_k \leq \bar{\gamma}_k$. Since by its maximality all robots are to the left of $\bar{\gamma}_k$ at all times, it follows that $\bar{c} = \frac{1}{N}(\sum_{i \neq k} \bar{r}_i + \bar{r}_k) \leq \bar{\gamma}_k - \frac{\delta_k}{N}$. Therefore,

$$|\bar{c}[t''] - \bar{\gamma}_k[t'']| \leq |\bar{c}[t'] - \bar{\gamma}_k[t']| + \frac{1}{N}\left(\sum_i \delta_i - 2\delta_k\right).$$

Look operations do not change ψ, and the total movement of robot k is $\Delta_k \geq \frac{1}{4}$. 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 lemma follows. □

We now give our main lemma. First, we treat the undisturbed-motion model. Note that in this case, the destination points are always the calculated centers of gravity.

LEMMA 2.7. *In the undisturbed-motion model, 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 their destination points 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 earliest time after t_0 when each robot has completed at least one entire Look-Compute-Move cycle. There are now two possibilities.

Case (1). Every destination point $\bar{\gamma}_i[t']$ that was calculated at any time $t' \in [t_0, t^*]$ 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, by Corollary 2.2. 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 2.5, $\psi[t] \leq 2$, and hence $\frac{1}{2N} \geq \frac{1}{4N}\psi[t_0]$. Combined, we get $\psi[\hat{t}] \leq \left(1 - \frac{1}{4N}\right)\psi[t_0]$.

Case (2). For some $t_1 \in [t_0, t^*]$, the destination point (or center of gravity) $\bar{\gamma}_i[t_1] = \bar{c}_i[t_1] = \frac{1}{N}\sum_{j=1}^{N} \bar{r}_j[t_1]$ 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 $[0, \frac{1}{2}]$. The lemma then follows by applying Lemma 2.6. □

To prove the convergence of the gravitational algorithm in the undisturbed-motion model in d-dimensional Euclidean space, we apply Lemma 2.7 to each dimension separately. Observe that by Lemmas 2.7 and 2.5, 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. This proves that in the undisturbed-motion model in d-dimensional Euclidean space, for any $N \geq 2$, N robots performing Algorithm `Go_to_COG` will converge.

We now turn to the full (sudden-stop) model. In this case the following lemma replaces Lemma 2.7.

LEMMA 2.8. *In the full (sudden-stop) model in d dimensions, for every time t_0, there exists some time $\hat{t} > t_0$ such that*

$$\psi[\hat{t}] \leq \max\left\{\left(1 - \frac{1}{8N^2}\right)\psi[t_0]\ ,\ \psi[t_0] - \frac{S}{4N\sqrt{d}}\right\}.$$

Proof. The proof is similar to the proof of Lemma 2.7 with the following changes.

- In each round we treat only the dimension for which h is largest (or one such dimension if there exist more than one). We refer to this dimension as the x-dimension and denote quantities in this dimension by ${}^{(x)}$.
- In Case (1), the moving robots may not complete their move and therefore may not leave the segment $[0, \frac{1}{2N}]$ even if no center of gravity was calculated in this segment. We split this case into two cases. If at the time of the last Look no robot resided in the segment $[0, \frac{1}{4N}]$, then no robot will reside there also at the end of the Move, and h must have decreased by at least $\frac{1}{4N}$. Otherwise, the distance between each of the robots in the segment $[0, \frac{1}{4N}]$ to its observed center of gravity in the x-dimension is at least $\frac{1}{4N}$, since by assumption $\bar{c}[t] \geq \frac{1}{2N}$ for all $t \in [t_0, \hat{t}]$. Recalling that d is the number of dimensions, we conclude that the total distance from these robots to the center of gravity is at most $\sqrt{d}$, since by the maximality of h in dimension x over every other dimension ν, $h^{(\nu)}[t_0] \leq h^{(x)}[t_0] = 1$. Thus, the ratio of the x-component of the vector $\bar{r}_i[t] - \bar{c}[t]$ to the vector size is at least $\frac{1}{4N\sqrt{d}}$. Therefore, the projection of a vector of length at least S in this direction on the x-axis is at least $\frac{S}{4N\sqrt{d}}$, and $h^{(x)}$ decreases by at least this amount.
- Case (2) still holds, since Lemma 2.6 is true even with sudden stops. ∎

Lemma 2.8 yields the following.

THEOREM 2.9. *In the full (sudden-stop) $\mathcal{ASYNC}$ model for any $N \geq 2$, in d-dimensional Euclidean space, N robots performing Algorithm `Go_to_COG` will converge.*

Proof. Denote $\Psi[t] = \sum_{\nu=1}^{d} \psi^{(\nu)}[t]$. Then

$$\Psi[t_0] \ \leq\ 2\sum_{\nu=1}^{d} h^{(\nu)}[t_0] \ \leq\ 2dh^{(x)}[t_0] \ \leq\ 2d\psi^{(x)}[t_0].$$

Therefore, the value of ψ for the largest dimension is $\psi^{(x)}[t_0] \geq \Psi[t_0]/2d$. By Lemma 2.8 the value of ψ for the largest dimension decreases after every two complete cycles of the robot swarm by an additive or multiplicative constant. The theorem follows. ∎

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.

3.1. The undisturbed-motion model. First, we discuss the convergence rate in the undisturbed-motion model.

LEMMA 3.1. *In the undisturbed-motion model, for every time interval* $[t_0, t_1]$*,*

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

Proof. Consider the different cases analyzed in the proof of Lemma 2.7. 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 (2), we have two possibilities. If Case (1) of Lemma 2.6 holds, then 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 (2) of Lemma 2.6. 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 (2) is repeated for $\lfloor \frac{t_1 - t_0}{2} \rfloor$ times. ∎

By the two inequalities of Lemma 2.5 we have that $h[t_1] \leq \psi[t_1]$ and $\psi[t_0] \leq 2h[t_0]$, respectively. Lemma 3.1 now yields the following lemma.

LEMMA 3.2. *In the undisturbed-motion $\mathcal{ASYNC}$ model, for every time interval* $[t_0, t_1]$*,*

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

COROLLARY 3.3. *In any execution of the gravitational algorithm in the undisturbed-motion $\mathcal{ASYNC}$ 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 3.4. *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 $\bar{\gamma}_i[t]$ 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 movement to its perceived center of gravity $\bar{\gamma}_i[t]$. 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, $h[t+1] \approx (1 - \frac{1}{N-1})h[t]$. It follows that $O(N)$ steps are needed to reduce the interval size to half 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 Corollary 3.3 and Lemma 3.4.

It is interesting to compare these bounds with what happens in the fully synchronous ($\mathcal{FSYNC}$) model. Here all robots operate at fixed time cycles, and the Look phase of all robots is simultaneous. In this model, after the first step all robots are at the center of gravity. Therefore, we have the following.

LEMMA 3.5. *In any execution of the gravitational algorithm in the undisturbed-motion $\mathcal{FSYNC}$ model, the robots gather after a single step.*

We now turn to the intermediate semisynchronous ($\mathcal{SSYNC}$) model. In this model there are fixed time cycles, and the Look phase of the robots is simultaneous, but only a subset of the robots may wake up at every cycle. We define, for the sake of time analysis, the notion of a (possibly long) time unit so that at every time unit, every robot is awakened at least once. (Here a "time unit" may compose of many cycles.)

LEMMA 3.6. *In any execution of the gravitational algorithm in the undisturbed-motion $\mathcal{SSYNC}$ model, over every interval of $O(N)$ time units, the size of the d-dimensional convex hull of the robot locations and centers of gravity is halved in each dimension separately.*

Proof. Again, we appeal to the proof of Lemma 2.7. However, in the semisynchronous model we have the advantage that at the beginning of the Look phase, each of the robots resides at the location of its last calculated center of gravity. This implies that Case (2) of the analysis of Lemma 2.6 is impossible. Thus, it is guaranteed that at every time step t, $h[t]$ is decreased by at least $\frac{h[t]}{2N}$. Therefore, we conclude that, over every interval of $O(N)$ time units, the size of the d-dimensional convex hull of the robot locations and centers of gravity is halved in each dimension separately. ∎

3.2. The full (sudden-stop) model. We now discuss the full (sudden-stop) model. We give the analogues of the above theorems in this case.

First, we discuss the fully synchronous model $\mathcal{FSYNC}$. In the following, $H[0]$ 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. We have the following lemma.

LEMMA 3.7. *In any execution of the gravitational algorithm in the full (sudden-stop) $\mathcal{FSYNC}$ 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, t, every robot moves by a vector $\min\{\bar{r}_i[t] - \bar{c}[t], S' \frac{\bar{r}_i[t]-\bar{c}[t]}{|\bar{r}_i[t]-\bar{c}[t]|}\}$ for some $S' \geq S$. By assumption, a is the width of the largest dimension, and therefore $a \geq |\bar{r}_i[t] - \bar{c}[t]|/\sqrt{d}$. 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. ∎

LEMMA 3.8. *In any execution of the gravitational algorithm in the full (sudden-stop) $\mathcal{SSYNC}$ model, over every interval of $O\left(N\left\lceil\frac{h}{S}\right\rceil\right)$ time units, the size of the d-dimensional convex hull of the robot locations and centers of gravity is halved in each dimension separately.*

Proof. As in the proof of Lemma 3.6, Case (2) of the analysis of Lemma 2.6 is impossible here. This guarantees that at every time step t, $h[t]$ is decreased by at least $\min\{\frac{S}{4N\sqrt{d}}, \frac{h[t]}{2N}\}$. Therefore, over every interval of $O(N\left\lceil\frac{h}{S}\right\rceil)$ time units, the size of the d-dimensional convex hull of the robot locations and centers of gravity is halved in each dimension separately. ∎

Finally, we present the result for the $\mathcal{ASYNC}$ model. The proof is similar to Corollary 3.3, with the additional option of sudden stops, as discussed in the proof of Lemma 2.8. We obtain the following.

THEOREM 3.9. *In any execution of the gravitational algorithm in the full (sudden-stop) $\mathcal{ASYNC}$ model, over every interval of $O\left(N^2 + \frac{Nh}{S}\right)$ time units, the size of the d-dimensional convex hull of the robot locations and centers of gravity is halved in each dimension separately.*

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

Let us first observe that the gravitational algorithm achieves *self-stabilization* (cf. [10]) in a model allowing only *transient* failures. Such a failure causes a change in the states of some robots, possibly into illegal states. That is, a measurement, calculation, or movement error may occur, causing the robot to move to a location other than the center of gravity. In essence, self-stabilization is the requirement that if the system *quiets* at some time t_0 (namely, no more transient errors occur), then from time t_0 on, the algorithm will achieve convergence, despite starting from a potentially illegal state. Notice that Theorem 2.9 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 early stages 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 4.1. *In the full (sudden-stop) $\mathcal{ASYNC}$ model, consider a swarm of N robots that execute Algorithm `Go_to_COG`. If $1 \le M \le N-2$ robots crash during the execution, then the remaining $N-M$ robots will converge to the center of gravity of the crashed robots. Moreover, the size of the robots' convex hull is halved every $O\left(N\left\lceil \frac{h}{S}\right\rceil\right)$ time units.*

Proof. Let us first consider an execution of the gravitational algorithm by a swarm of N robots in one dimension. Without loss of generality, assume that the crashed robots were $1,\ldots,M$ and their crashing times were $t_1 \le \cdots \le t_M$, 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 their 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 $t_0 \ge t_M$. Let $H[t_0] = [a,b]$ for some $a \le 0 \le b$. By Corollary 2.2, the robots will remain in the segment $[a,b]$ at all times $t \ge t_0$. The center of gravity calculated by any nonfaulty robot $M+1 \le j \le N$ at time $t \ge t_0$

will then be

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

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 and no sudden stops occur, we have that $H[\hat{t}] \subseteq [a', b']$, and 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 the point 0, namely, the center of gravity of the crashed robots.

If sudden stops do occur, then the robots are not guaranteed to reach their destination. However, they are guaranteed to travel a distance of at least S. Therefore, using arguments similar to the proof of Lemma 2.8, if sudden stops occur, then the size of the hull in the largest dimension is decreased by at least $\frac{S}{4N\sqrt{d}}$, leading to the theorem. ∎

5. Conclusions. We have considered the properties of the gravitational algorithm for convergence of mobile robot swarms. We have shown that the algorithm guarantees convergence to a point for any number of robots N and in any dimension d. We have shown that an appropriate quantity is halved every $O(N^2)$ steps and have shown a case where $O(N)$ steps are needed for halving the invariant. Thus, a gap still exists between our bounds on the convergence rate, which is left as an open question. We have shown analogous (and somewhat stronger) results for the simpler synchronous and semisynchronous models. We have also shown that the algorithm is resilient to crash failures of any number of robots.

Acknowledgment. We are grateful to the anonymous reviewers, whose comments helped clarify and improve the presentation of our results.

REFERENCES

[1] N. Agmon and D. Peleg, *Fault-tolerant gathering algorithms for autonomous mobile robots*, in Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms, 2004, pp. 1063–1071.

[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 (1999), pp. 818–828.

[3] H. Ando, I. Suzuki, and M. Yamashita, *Formation and agreement problems for synchronous mobile robots with limited visibility*, in Proceedings of the IEEE Symposium of Intelligent Control, 1995, pp. 453–460.

[4] T. Balch and R. Arkin, *Behavior-based formation control for multi-robot teams*, IEEE Trans. Robotics and Automation, 14 (1998), pp. 926–939.

[5] G. Beni and S. Hackwood, *Coherent swarm motion under distributed control*, in Proceedings of the International Symposium on Distributed Autonomous Robotic Systems, 1992, pp. 39–52.

[6] Y. U. Cao, A. S. Fukunaga, and A. B. Kahng, *Cooperative mobile robotics: Antecedents and directions*, Autonomous Robots, 4 (1997), pp. 7–23.

[7] Y. U. Cao, A. S. Fukunaga, A. B. Kahng, and F. Meng, *Cooperative mobile robots: Antecedents and directions*, in Proceedings of the IEEE/RSJ International Conference of Intelligent Robots and Systems, 1995, pp. 226–234.

[8] M. Cieliebak, P. Flocchini, G. Prencipe, and N. Santoro, *Solving the robots gathering problem*, in Proceedings of the 30th Annual International Colloquium on Automata, Languages, and Programming, Springer, New York, 2003, pp. 1181–1196.

[9] M. Cieliebak and G. Prencipe, *Gathering autonomous mobile robots*, in Proceedings of the 9th Annual International Colloquium on Structural Information and Communication Complexity, Carleton Scientific, Waterloo, ON, Canada, 2002, pp. 57–72.

[10] S. Dolev, *Self-Stabilization*, MIT Press, Cambridge, MA, 2000.

[11] 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 Proceedings of the 10th Annual International Symposium on Algorithms and Computation, Springer, New York, 1999, pp. 93–102.

[12] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer, *Gathering of autonomous mobile robots with limited visibility*, in Proceedings of the 18th Annual Symposium on Theoretical Aspects of Computer Science, Springer, New York, 2001, pp. 247–258.

[13] D. Jung, G. Cheng, and A. Zelinsky, *Experiments in realising cooperation between autonomous mobile robots*, in Proceedings of the International Symposium on Experimental Robotics, Springer, New York, 1997.

[14] Y. Kawauchi, M. Inaba, and T. Fukuda, *A principle of decision making of cellular robotic system (CEBOT)*, in Proceedings of the IEEE Conference on Robotics and Automation, 1993, pp. 833–838.

[15] M. J. Mataric, *Interaction and Intelligent Behavior*, Ph.D. thesis, MIT, Cambridge, MA, 1994.

[16] S. Murata, H. Kurokawa, and S. Kokaji, *Self-assembling machine*, in Proceedings of the IEEE Conference on Robotics and Automation, 1994, pp. 441–448.

[17] L. E. Parker, *Designing control laws for cooperative agent teams*, in Proceedings of the IEEE Conference on Robotics and Automation, 1993, pp. 582–587.

[18] L. E. Parker, *On the design of behavior-based multi-robot teams*, J. Advanced Robotics, 10 (1996), pp. 547–578.

[19] L. E. Parker and C. Touzet, *Multi-robot learning in a cooperative observation task*, Distributed Autonomous Robotic Systems, 4 (2000), pp. 391–401.

[20] L. E. Parker, C. Touzet, and F. Fernandez, *Techniques for learning in multi-robot teams*, in Robot Teams: From Diversity to Polymorphism, T. Balch and L. E. Parker, eds., A. K. Peters, Wellesley, MA, 2001, pp. 191–236.

[21] G. Prencipe, *CORDA: Distributed coordination of a set of atonomous mobile robots*, in Proceedings of the 4th Annual European Research Seminar on Advances in Distributed Systems, 2001, pp. 185–190.

[22] G. Prencipe, *Distributed Coordination of a Set of Atonomous Mobile Robots*, Ph.D. thesis, Universita Degli Studi Di Pisa, Pisa, Italy, 2002.

[23] K. Sugihara and I. Suzuki, *Distributed algorithms for formation of geometric patterns with many mobile robots*, J. Robotic Systems, 13 (1996), pp. 127–139.

[24] I. Suzuki and M. Yamashita, *Agreement on a common x-y coordinate system by a group of mobile robots*, in Intelligent Robots: Sensing, Modelling and Planning, World Scientific, Singapore, 1997, pp. 305–321.

[25] I. Suzuki and M. Yamashita, *Distributed anonymous mobile robots - formation and agreement problems*, in Proceedings of the 3rd Annual Colloquium on Structural Information and Communication Complexity, Carleton Scientific, Waterloo, ON, Canada, 1996, pp. 313–330.

[26] I. Suzuki and M. Yamashita, *Distributed anonymous mobile robots: Formation of geometric patterns*, SIAM J. Comput., 28 (1999), pp. 1347–1363.

[27] I. A. Wagner and A. M. Bruckstein, *From ants to a(ge)nts*, Ann. Math. Artif. Intell., 31 (1996), pp. 1–5.