Distributed Coordination of a Set of Autonomous Mobile Robots

P. Flocchini
School of Inf. and Eng.
University of Ottawa
Ottawa, Canada
flocchin@site.uottawa.ca

G. Prencipe
Dip. di Informatica
Università di Pisa
Pisa, Italy
prencipe@di.unipi.it

N. Santoro
Dept. of Comp. Sc.
Carleton University
Ottawa, Canada
santoro@scs.carleton.ca

P. Widmayer
Theoretische Informatik
ETH Zürich
Zürich, Switzerland
pw@inf.ethz.ch

Abstract

The distributed coordination and control of a set of autonomous mobile robots is a problem widely studied in a variety of fields, such as engineering, artificial intelligence, artificial life, robotics. Generally, in these areas the problem is studied mostly from an empirical point of view. In contrast, we aim to understand the fundamental limitations on what a set of autonomous mobile robots can achieve. In this paper we describe the current investigations on what autonomous mobile robots can and cannot do with respect to some coordination problems.

1 Introduction

In this paper we describe the current investigations on the algorithmic limitations of what autonomous mobile robots can do with respect to pattern formation.

The interest in distributed autonomous robot systems has increased considerably in recent years. The problem of controlling a set of autonomous, mobile robots in a distributed fashion has been studied extensively, but almost exclusively from an engineering and from an artificial intelligence point of view. Leading research activities in the engineering area include the Cellular Robotic System (CEBOT) of Kawaguchi et al [20], the Swarm Intelligence of Beni et al. [4], the Self-Assembly Machine (”fructum”) of Murata et al. [22], etc. In the AI community there has been a number of remarkable studies, eg., on social interaction leading to group behavior by Matarić [21], on selfish behavior of cooperative robots in animal societies by Parker [24], on primitive animal behavior in pattern formation by Balch and Arkin [2], to cite just a few.

In all these investigations, algorithmic aspects were somehow implicitly an issue, but clearly not a major concern, let alone the focus, of the study. An investigation with an algorithmic flavor has been undertaken within the AI community by Durfee [13], who argues in favor of limiting the knowledge that an intelligent robot must possess in order to be able to coordinate its behavior with others. The work of Suzuki and Yamashita [1, 26, 27] is the closest to our study (and, with this focus, a rarity in the mobile robots literature); it gives a nice and systematic account on the algorithmics of pattern formation for robots, under several assumptions on the power of the individual robot. Our model, however, differs with respect to the assumptions on the robots capabilities (the robots we consider are “weaker”); our results are practically more relevant and theoretically more powerful.

In Section 2 the formal definition of the model under study is presented. In Section 3, we present an overview of the main research investigations which motivates the “weak” assumptions of our model. In Section 4 we review some results related to the analysis of the problem of pattern formation by autonomous mobile robots. Finally, in Section 5 we draw some conclusions and present suggestions for further study.

2 The Model

The robots we consider are: *homogeneous* (they all follow the same set of rules), *autonomous* (there is no a priori central authority, and each robot's computing capabilities are independent from the others), *asynchronous* (there is no central clock, no a priori synchronization, no a priori bounds on processing or motorial speed), *mobile* (robots are allowed to move on a plane), *anonymous* (they are a priori indistinguishable), *oblivious* (they do not explicitly remember the past). Moreover, there are no explicit direct means of communication: the communication occurs in a totally implicit manner, through the environment (in biology, this communication is called *stigmergy* [3, 18]).

These assumptions make our robots simple and rather "weak" in light of current engineering technology. But, as already noted, we are interested in approaching the problem from a *computational* point of view; it is precisely by assuming the "weakest" robots that it is possible to analyze the strengths and weaknesses of the distributed control.

Each robot has its own *local view* of the world. This view includes a local Cartesian coordinate system with origin, unit of length, and the *directions* of two coordinate axes, identified as x axis and y axis, together with their *orientations*, identified as the positive and negative sides of the axes. Notice, however, that the local views could be totally different making impossible for the robots to agree on directions or on distances.

A robot is initially in a waiting state (*Wait*); at any point in time, asynchronously and independently from the other robots, it observes the environment in its area of visibility (*Observe*), it calculates its destination point based only on the current locations of the observed robots (*Compute*), it then moves towards that point (*Move*) and goes back to a waiting state.

1. Wait The robot is idle. A robot cannot stay infinitely idle.

2. Observe The robot observes the positions of all other robots with respect to its local coordinate system. Each robot is viewed as a point, and therefore the *observation* returns a set of points to the observing robot. Two different models can arise depending on whether we assume that a robot can see all the other robots in the system (called *Unlimited Visibility* model) or that a robot can see only the robots that are at most at some fixed distance from it (*Limited Visibility* model).

3. Compute The robot performs a *local computation* according to its algorithm, based only on its local view of the world and the observed set of points.

4. Move As a result of the computation, the robot either stands still, or it moves (along any curve it likes). The robot moves towards the computed destination of an unpredictable amount of space, which we assume neither infinite, nor infinitesimally small.

A computational cycle is defined as the sequence of the *Wait-Observe-Compute-Move* actions; the "life" of a robot is then a sequence of computational cycles.

Notice that, because of the obliviousness assumption, both the result of the computation and that of the observation phase will not be available to a robot at its next computational cycle.

The system is totally *asynchronous*, in the sense that there is no common notion of time, and a robot observes the environment at unpredictable time instants. Moreover, we do not make any assumption on the cycle time of each robot, neither on the time a robot stays in waiting state. We only assume that each cycle is completed in finite time, and that the distance travelled in a cycle is finite; we also require the distance not to be infinitesimally small (unless the robot reaches its destination).

Notice that, since during the computing phase of a robot other robots could obviously move, the movement of a robot could be based on a past situation which is not valid anymore at the moment of the actual move.

3 Emergent Behaviour and Self-Stabilization

Our model finds in the behavior-based approach its main motivations. This fields of study on collaborative behaviors analyzes *emergent behaviors* arising from the coordination and cooperation of the set of robots in the system, and it is characterized by goals that are not "explicitly programmed in but result from local interactions between a system's component" [21]. In other words, the functionality of a machine usually arises from its component and it is part of the project of its designer, whereas the emergent functionality (behavior), on the other hand, arises from the interaction of its components which are not directly programmed with a particular function in mind. The only things, therefore, that are programmed in the robots are the *behaviors*: a set of laws that guides the robot to react to environmental stimuli, with the property that there is no explicit goal programmed in. Hence, although the agents are working together from an observer's viewpoint, they are not from the agents' viepoint. The cooperation and the goals simply *emerge* (externally) as the computation goes on [12]. This feature renders this approach environment-independent, allowing quick reaction to changes in the environment. These kind of systems are also called *self-organizing*, because of their capacity to exploit behaviour that are not directly designed.

One of the first studies conducted in this direction is by R. A. Brooks. In [5] he describes the *subsumption architecture*, composed by a set of layers, each describing a behavior of the robot associated to increasing level of competence.

Another noteworthy study in this field, the one which has most motivated the use of the "weak" assumptions of our model, is that of Matarić [21]. The main idea in Matarić's work is that "interactions between individual agents need not to be complex to produce complex global consequences". She tries to understand which kind of simple interactions are necessary to produce complex group behaviors in a fully distributed multi-agent system. The set of agents under study has no *a priori* leader, hence all the agents are homogeneous. There is no explicit one-to-one communication, and all communication is based on changes in the environment that the agent sense (implicit communication). There is no central coordination.

At first a set of simple interactions between agents is described: *Collision Avoidance*, *Following*, *Dispersion*, *Aggregation*, *Homing*. Then, ways for compounding these basic interactions in order to obtain more complex group behaviors are illustrated. Examples of experimented compound behaviors include *flocking*, that is the ability of the agents of moving in a flock towards a goal, and *foraging*, consisting of finding pucks in the environment, picking them up and delivering them to the home region.

Other studies that deal with the issue of investigating this area can be found in [3], where the authors study in particular stigmergy communication and use a set of simple robots that operate completely autonomously and independently for the task of collecting pucks spread over the environment (a square arena) in a single cluster, showing that, despite the simplicity of the robots and the kind of implicit communication adopted, potentially useful tasks can be performed. Furthermore, we can cite the ALLIANCE architecture of [24], the formation and navigation problems in multi-robot teams of [2], the experiments in cooperative cleaning behavior of [19].

Our work arises from emergent behavior studies, in particular that of Matarić, as they motivate the simplicity of the agents under analysis. As mentioned earlier, this simplicity allows us to formally highlight by an algorithmic and computational viewpoint the minimal capabilities the agents must have in order to accomplish basic tasks and produce interesting interactions. Furthermore, it allows us to understand better the limitations of the distributed control in an environment inhabited by mobile agents, hence to formally prove what can not be achieved under the "weakness" assumptions of our model, that will be detailed later.

Alternative approaches to the problem of studying multi-robot systems, can be found in the CEBOT system of [17], in the planner-based architecture of [23], or in the information requirements theory of [11] (see [6] for a survey).

The common feature of all these approaches is that they do not deal with formal correctness and they are only analyzed empirically. Algorithmic aspects were somehow implicitly an issue, but clearly not a major concern. We aim to identify the algorithmic limitations of what autonomous, mobile robots can do.

The work of Suzuki and Yamashita [27] is the closest to our study. The model that we use differ from those of [27] in that our agents are as weak as possible in every single aspect of their behavior. The reason is that we want to identify the role of the robots' common knowledge of the world for performing a task.

Other significant work which has informed and motivated us in our study comes from the area of self-stabilization. The term *self-stabilization* was first introduced by Dijkstra, who called a distributed system self-stabilizing "if and only if, regardless of the initial state [...], the system is guaranteed to find itself in a legitimate state after a finite number of moves" [10]. From his definition, we have that such a system does not need to be initialized and can recover from transient failures caused by moves that brought the system in an illegitimate state, where *transient failures* are events that may change the state of the system by corrupting the local state of a processor (for an example its local memory or program counter). Due to these features, recently this notion has been intensely studied as an approach to fault tolerance, since a self-stabilizing system can recover from inconsistent system states that could occur without any outside intervention [25].

One of the features of our agents is that they are oblivious, in that they do not have memory of the past. This clearly gives the self-stability property to the algorithms designed for our model, since the decision an agent takes at some time in the computation does not depend on what happened in the system previously, therefore from what could be stored in its local memory. But there is one main difference between our self-stabilizing algorithms and the known ones: our system is not modeled on a graph. In fact, the main work done on self-stabilization models distributed systems as graphs whose nodes are the processors in the system and whose edge represent the connection between the processors, hence the topology of the system. In contrast, our model allows the agents (the processors in the distributed system) to move on the plane, without any restriction in their moves.

A study by Debest [9] points out the importance of better understending self-stabilization in systems built on several components moving independently from each other and cooperating in order to reach a common goal. He studies the problem of the circle formation by a set of autonomous mobile robots, describing informally an algorithm and analyzing the reaction of the system to "unexpected events", such as the changing of position of some robots, or failure, program modification, and erroneous behavior by one robot. In his paper he concludes that "in an environment where software applications are growing more and more distributed [...], a good understanding of how the self-stabilizing state of a complex system is influenced by the intrinsic behavior of its individual components is a prerequisite to designs working well [...]".

The idea of studying self-stabilizing concepts in systems composed by cooperating mobile agents is new and the only work to our knowledge dealing with this problem is the one by Debest. Therefore, we find that our work can be useful in contributing to a better understanding of self-stabilization in "self-organizing" systems.

4 Pattern Formation

4.1 Unlimited Visibility.

We first consider the coordination problem of forming a specific geometric pattern in the unlimited visibility setting. The *pattern formation* problem has been extensively investigated in the literature [8, 26, 27, 28], where usually the first step is to gather the robots together and then let them proceed in the desired formation (just like a flock of birds or a troupe of soldiers). The problem is practically important, because, if the robots can form a given pattern, they can agree on their respective roles in a subsequent, coordinated action.

The geometric pattern is a set of points (given by their Cartesian coordinates) in the plane, and it is initially known by all the robots in the system.

The robots are said to *form the pattern*, if, at the end of the computation, the positions of the robots coincide, in everybody's local view, with the points of the pattern, where the formed pattern may be *translated*, *rotated*, *scaled*, and *flipped* into its mirror position with respect to the input one. Initially the robots are in arbitrary positions, with the only requirement that no two robots are in the same position, and that, of course, the number of points prescribed in the pattern and the number of robots are the same.

The pattern formation problem is quite a general member in the class of problems that are of interest for autonomous, mobile robots. It includes as special cases many coordination problems, such as leader election, where the pattern is defined in such a way that the leader is uniquely represented by one point in the pattern. This reflects the general direction of our investigations: what coordination problems can be solved, and under what conditions? The only means for the robots to coordinate is the observation of the others' positions; therefore, the only means for a robot to send information to some other robot is to move and let the others observe (reminiscent of bees in a bee dance). For oblivious robots, even this sending of information is impossible, since the others will not remember previous positions.

Also I. Suzuki and M. Yamashita solve the same problem in their model [27], characterizing what kind of patterns can be formed. But all their algorithms are non-oblivious; in fact, they require the capability of the robots to remember the past, while ours are totally oblivious.

In an attempt to understand the power of common knowledge for the coordination of robots, we have studied the pattern formation problem under several assumptions, obtaining a complete characterization of what can and what cannot be achieved. The following theorem summarizes the results holding for a set of n autonomous, anonymous, oblivious, mobile robots:

Theorem 4.1. *([16])*

1. *With common knowledge of two axes directions and orientations, the robots can form an arbitrary given pattern.*
2. *With common knowledge on only one axis direction and orientation, the pattern formation problem is unsolvable when n is even; it can be solved if n is odd.*
3. *With common knowledge on only one axis direction, the robots can form an arbitrary pattern if n is odd.*
4. *With no common knowledge, the robots cannot form an arbitrary given pattern.*

We have then studied what patterns can or cannot be formed when the arbitrary pattern formation is unsolvable.

Theorem 4.2 ([14]). *With common knowledge on only one axis direction and orientation, there exists no deterministic algorithm that allows an even number of robots to form an asymmetric pattern. Moreover, in this case they can only form symmetric patterns that have at least one symmetric axis not passing through a vertex of the input pattern.*

Finally, we have concentrated on the situation when no common knowledge neither on directions nor on orientations is available (case 4. of Theorem 4.1). As seen before, in this case an arbitrary pattern cannot be formed; we have then investigated the simpler problem of *point formation*. In the point formation problem, the robots are required to gather in a point of the plane (not fixed in advance), assuming they start from distinct positions and that they have unlimited visibility.

Our algorithm requires that the robots have the ability to recognize positions with multiplicity greater than one (in this case, we say that the robots detect multiplicity). This assumption is not too restrictive; we know that it is actually necessary, in fact the following result holds for a set of anonymous, oblivious, mobile robots:

Theorem 4.3 ([7]). *If the robots cannot detect multiplicity, the point formation problem is unsolvable.*

Even assuming the ability to detect multiplicity, the solution of the problem is not trivial. The difficulties arise from the fact that the robots, during the computation, could form some particular patterns from which the point formation becomes hard. For instance, when the robots are placed in a regular n-gon, no algorithm can be designed so to allow a single robot to move; in fact, due to the anonymity of the robots, any moving strategy could make all of them move. Other problematic situations arise when there exists a point p in the plane such that the robots are in an *equiangular* situation with respect to p.

Our algorithm is *oblivious* and solves the problem under the condition that the starting position of the robots is not an equiangular one. If, instead of total obliviousness, we allow the robots to have one bit of memory [7], the problem is then solved under all starting conditions.

An oblivious algorithm for point formation under all starting conditions is still an open problem.

4.2 Limited Visibility.

Other interesting problems arise in the limited visibility model. Currently, we have an algorithm that solves the point formation problem under the assumption that there is common knowledge on the orientation and direction of both axes [15]. A necessary condition to solve this problem, is that no robot is completely ”isolated” from the others at the beginning of the computation. More formally, let $\mathcal{C}_i$ be the visibility area of a robot r_i. We define the *visibility graph* as follows:

Definition 4.1 (Visibility Graph). The *visibility graph* $G = (N, E)$ is a graph whose node set N is the set of the input robots and $(r_i, r_j) \in E$ iff $r_j \in \mathcal{C}_i$ and $r_i \in \mathcal{C}_j$, where r_i and r_j are two robots in their initial positions.

We can state the following:

Lemma 4.1. *If the visibility graph is disconnected, the problem is unsolvable.*

Let us call Universe the smallest isothetic rectangle containing the initial configuration of the robots and let us call *Right* and *Bottom* respectively, the rightmost and the bottom most side of the Universe. When the visibility graph is connected, the idea of the algorithm is to make the robots move either towards the Bottom or towards the Right of the Universe (a robot will never move up or to its left), in such a way that, after a finite number of steps, they will gather at the bottom most rightmost corner of the Universe.

Theorem 4.4 ([15]). *There exists a deterministic algorithm that let the robots gather in one point in a finite number of movements, in the limited visibility setting and assuming common knowledge on direction and orientation of both axes.*

This same problem has been investigated also in [1], where the authors have presented an algorithm that *converges*, but *does not reaches*, the point. On the other hand, our results imply that the robots gather in a point in finite time. Moreover, their algorithm would not work in our model since it is based on stronger assumptions.

5 Conclusion and Discussion

The purpose of our study is to gain a better understanding of the power of the distributed control from an algorithmic point of view. We presented a model consisting of a set of autonomous, anonymous, memoryless, mobile robots - features that render our robots “weak” - and we have outlined the current status of the investigation.

There are many issues which merit further research. The operating capabilities of our robots are quite limited; it would be interesting to look at models where robots have more complex capabilities. For instance, we could use a totally *non-oblivious* model, that is, one with an unlimited amount of memory that each robot could use. Alternatively, we could equip the robots with just a bounded amount of memory (*semi-obliviousness*), and see if this added “power” can be useful in solving problems otherwise unsolvable, if it could be used to design faster algorithm, and how it could affect the self-stabilizing feature of the oblivious algorithms.

Other alterations we could make to our system which would inspire further study include adding the possibility that at any given time a robot may never see the world as it actually is, since the robots' cameras rotate slowly as they are taking a picture; we could also explore robots that have some kind of direct communication, and we could assume different kind of robots that move in the environment.

Slightly faulty snapshots, a limited range of visibility, obstacles that limit the visibility and that moving robots must avoid or push aside, as well as robots that appear and disappear from the scene clearly suggest that the algorithmic nature of distributed coordination of autonomous, mobile robots merits further investigation.

References

[1] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita. A Distributed Memoryless Point Convergence Algorithm for Mobile Robots with Limited Visibility. *IEEE Trans.*, 15(5):818–828, 1999.

[2] T. Balch and R. C. Arkin. Behavior-based Formation Control for Multi-robot Teams. *IEEE Trans. on Robotics and Automation*, 14(6), December 1998.

[3] R. Beckers, O. E. Holland, and J. L. Deneubourg. From Local Actions To Global Tasks: Stigmergy And Collective Robotics. In *Art. Life IV, 4^{th} Int. Worksh. on the Synth. and Simul. of Living Sys.*, 1994.

[4] G. Beni and S. Hackwood. Coherent swarm motion under distributed control. In *Proc. DARS'92*, pages 39–52, 1992.

[5] R. A. Brooks. A Robust Layered Control System for a Mobile Robot. Technical Report AI-TR-864, MIT, September 1985.

[6] Y. U. Cao, A. S. Fukunaga, A. B. Kahng, and F. Meng. Cooperative Mobile Robotics: Antecedents and Directions. In *Int. Conf. on Intel. Robots and Sys.*, pages 226–234, 1995.

[7] M. Celiebak, P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer. Point Formation for Oblivious Robots. submitted.

[8] Q. Chen and J. Y. S. Luh. Coordination and control of a group of small mobile robots. In *Proc. IEEE Int. Conf. on Rob. and Aut.*, pages 2315–2320, 1994.

[9] X. A. Debest. Remark About Self-stabilizing Systems. *Comm. of the ACM*, 238(2):115–117, 1995.

[10] E. W. Dijkstra. Self-stabilizing Systems in Spite of Distributed Control. *Comm. of the ACM*, 17(11):643–644, 1974.

[11] B. R. Donald, J. Jennings, and D. Rus. Information Invariants for Distributed Manipulation. *The Int. Journal of Robotics Research*, 16(5), October 1997.

[12] J. E. Doran, S. Franklin, N. R. Jennings, and T. J. Norman. On Cooperation in Multi-Agent Systems. *The Knowledge Eng. Review*, 12(3):309–314, 1997.

[13] E. H. Durfee. Blissful Ignorance: Knowing Just Enough to Coordinate Well. In *ICMAS*, pages 406–413, 1995.

[14] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer. Achievable Patterns by an Even Number of Mobile Robots. manuscript.

[15] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer. Gathering of Autonomous Mobile Robots With Limited Visibility. submitted.

[16] 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 *ISAAC '99*, pages 93–102, 1999.

[17] T. Fukuda, Y. Kawauchi, and H. Asama M. Buss. Structure Decision Method for Self Organizing Robots Based on Cell Structures-CEBOT. In *Proc. IEEE Int. Conf. on Robotics and Autom.*, volume 2, pages 695–700, 1989.

[18] P. P. Grassè. La theorie de la stigmergie: Essai D'Interpretation des Termites Constructeurs. *Ins. Soc.*, 6:41–83, 1959.

[19] D. Jung, G. Cheng, and A. Zelinsky. Experiments in Realising Cooperation between Autonomous Mobile Robots. In *ISER*, 1997.

[20] Y. Kawauchi and M. Inaba and. T. Fukuda. A principle of decision making of cellular robotic system (cebot). In *Proc. IEEE Conf. on Robotics and Autom.*, pages 833–838, 1993.

[21] M. J Matarić. *Interaction and Intelligent Behavior*. PhD thesis, MIT, May 1994.

[22] S. Murata, H. Kurokawa, and S. Kokaji. Self-assembling machine. In *Proc. IEEE Conf. on Robotics and Autom.*, pages 441–448, 1994.

[23] F. R. Noreils. Toward a Robot Architecture Integrating Cooperation between Mobile Robots: Application to Indoor Environment. *The Int. J. of Robot. Res.*, pages 79–98, 1993.

[24] L. E. Parker. On the Design of Behavior-Based Multi-Robot Teams. *Journal of Advanced Robotics*, 10(6), 1996.

[25] M. Schneider. Self-stabilization. *ACM Computing Surveys*, 25(1), 1993.

[26] K. Sugihara and I. Suzuki. Distributed Algorithms for Formation of Geometric Patterns with Many Mobile Robots. *Journal of Robotics Systems*, 13:127–139, 1996.

[27] I. Suzuki and M. Yamashita. Distributed Anonymous Mobile Robots: Formation of Geometric Patterns. *Siam J. Comput.*, 28(4):1347–1363, 1999.

[28] P. K. C. Wang. Navigation Strategies for Multiple Autonomous Mobile Robots Moving in Formation. *Journal of Robotic Systems*, 8(2):177–195, 1991.