Conflicts in a simple autonomy-based multi-agent system

F. Chantemargue, P. Lerena, M. Courant
University of Fribourg, Computer Science Department
Parallelism and Artificial Intelligence Group (PAI)
Chemin du musée 3, Pérolles
CH-1700 Fribourg, Switzerland
fabrice.chantemargue@unifr.ch

Abstract

This paper shows that, in some situations, conflict can deliberately be left in an autonomy-based multi-agent system. This study, supported by experimental results, has two major outcomes. First, it proves that conflict does not necessarily alter the global outcome of the system in a qualitative way. Second, it shows that it is possible to effect the way the global task is achieved by appropriately modifying the environment of the agents.

Introduction

Our work fits in the framework of *Bottom-Up Artificial Intelligence* (Brooks 1986), (Brooks 1991) and more particularly, in that of *Autonomous Agents* (Pfeifer 1995). We are concerned with collective phenomena and their issues and more precisely, the way to carry out solutions that allow an autonomy-based multi-agent system to achieve a global task by virtue of emergence and self-organization. Emergence offers indeed a bridge between the necessity of complex and adaptive behavior at a macro level (the one of the system) and situation-based interactions at a micro level (the one of every agent) (Forrest 1990). We are concerned with the study of systems made of autonomous agents, looked at from the point of view of their adaptive capabilities. For that, we take up a methodology which follows an engineering trend that can be characterized by:

1. a bio-inspired approach, drawing its models from evolutionary biology and the study of animal societies, and participating in the concept of embodiment of intelligence now surfacing in *artificial intelligence* (Robert, Chantemargue, & Courant 1998), (Lerena & Courant 1998);
2. the relinquishing of supervised control and stringent hierarchical data structures in favor of decentralized control strategies based on interactions through influences. These strategies, hence requiring autonomous components are expected to lead to solutions through emergent features (Chantemargue *et al.* 1998b);
3. an epistemological stance that fits in with the roots of cognitive science, in which can be found the theory of Varela (Varela, Thompson, & Rosch 1991).

Our work is supported by two types of experiments, namely multi-agent simulations applied to collective robotics, and collective robotics applications involving real robots. This paper will (i) focus exclusively on a multi-agent case study, which consists of studying through intensive simulations how a pool of operationally autonomous agents regroups objects that are distributed in their environment, and (ii) address the specific question of conflict which, according to a common-sense idea, is bound to happen in every system composed of multiple entities. In this paper, we take up an approach in which nothing is explicitly undertaken to *eliminate* conflict in the system. However, the results we obtained show that in our context and to some extent, the system is able to cope with conflict by itself. This paper is organized as follows: the first section starts by briefly reporting how conflict has been attempted to be avoided and solved till now in the framework of *Artificial Intelligence* and *Distributed Artificial Intelligence.* The second section describes our experimental testbed and highlights our approach on conflict. The next section gives the most significant experimental results for our purpose. The last section summarizes our point of view regarding conflict and concludes this paper.

Conflict in Artificial Intelligence and Distributed Artificial Intelligence

There are indeed numerous reasons for which conflict may arise, depending on the context of the application and on the type of entities that are considered. For instance, entities may have limited access to resources; entities may have knowledge conflicts, due to problems of incompleteness, uncertainty or non reliability in their own knowledge databases (see for instance (Kwong, Wong, & Low 1998), (Cholvy 1998), (Lamontagne & Benhamou 1998)). When the term *conflict* is evoked, it is very likely to relate to a problem of concurrent access to resources. As far as we are concerned, we consider conflict as a divergence at the level of goals between entities in the system: therefore, for us, conflict is linkened to antagonism (Ferber 1995). An access to a given resource can itself be classified as a goal (or sub-goal) and is consequently included in this interpretation of the term. Note that this notion of conflict could be extended so as to encompass *sources* of goal divergences, as in cases of *knowledge conflict*. Generally, it is argued that if conflict is not appropriately solved, the system may run into serious trouble (it may not fulfill the goals for which it was designed). If it appears to be true that conflict has to be planned ahead in certain contexts such as *operating systems*, it however seems paradoxical in the context of *autonomous agents*, wherein approaches to system development are typically bottom-up, starting from atomic entities, whose composition of behaviors is hard to be predicted by the designer.

Conflict is somehow addressed by coordination. Coordination is defined as the management of dependencies between activities (Malone & Crowston 1994). Work generally concentrates on peculiar aspects of coordination, among which the most commonly addressed topic is *cooperation*. In (Ferber 1995) an interaction taxonomy for goal-oriented agents is put forward: interactions are classified according to the goals of the agents, their resources and their skills. This taxonomy leads to three categories of situations: *indifference* and *cooperation* situations encompass situations where goals are compatible (with the different combinations for resources and skills, namely sufficient and insufficient), whereas *antagonistic* situations group together all possible situations in which goals are incompatible (indiscriminately of the status of resources and skills). In (Ferber 1995), numerous methods for handling cooperation situations are discussed (see (Chaib-Draa 1996) as well). These methods, according to Mataric's definition (Mataric 1992), imply information exchanges and actions performed by a couple of agents in order to help some others. We refer to them as *explicit cooperation*, that is, a set of interaction organization methods developed for foreseen situations. However, what is remarkable is that antagonistic situations are never addressed: it seems as if they were considered out of scope.

In fact, approaches to address the problem of conflict mostly consisted in replicating techniques for the prevention of conflict that were developed in the discipline of *operating systems*, and in fitting them to *Distributed Artificial Intelligence*. Our approach is quite the reverse: we are concerned with conceiving systems in which conflict is tolerated.

An experimental multi-agent testbed

Our simulation tackles a quite common problem in collective robotics which is still given a lot of concern: agents seek for objects distributed in their environment in order to regroup all objects. However, the way we address this problem is not classic: the innovative aspect of our approach rests indeed on a system integrating operationally autonomous agents, that is, every agent in the system has the freedom to act. More precisely, every agent decides by itself which action to take on the basis of its own perception, which is strictly local and private. Therefore, there is not in the system any type of master responsible for supervising the agents, nor any type of cooperation protocol, thus allowing the system to be more flexible and fault tolerant. In that, this work relates to other work in the framework of collective robotics (see (J.C. Deneubourg et al. 1991), (Bonabeau & Theraulaz 1994), (Gaussier & Zrehen 1994), (R. Beckers and O.E. Holland and J.L. Deneubourg 1994), (Martinoli & Mondada 1995) and (Martinoli & Mondada 1998)) in which the focus is on the collective capabilities of a multi-robot system to achieve clustering tasks (the system typically creates several clusters of objects in the environment) and/or eventually regrouping tasks (the system typically creates a single cluster containing all the objects of the environment) on the basis of a stigmergic[1] coordination between robots.

We implemented our simulation in the Swarm Simulation System (developed at the Santa-Fe Institute, USA) (Langton, Minar, & Burkhart 1996). In our simulation, the environment is composed of a discrete two dimensional square grid, a set of objects and a set of transparent obstacles. A set of agents is present in this environment: agents roam (by avoiding obstacles), pick up and drop objects. At the outset of an experiment, objects are (randomly) distributed in the environment that may contain obstacles (randomly set or set at a chosen fixed location), with at maximum one object per cell. An experiment will be considered completed when all objects present in the environment will be regrouped by the agents onto a single cell (in this case, we will speak of a stack of all objects).

An agent possesses some sensors to perceive the world within which it moves, and some effectors to act in this world, so that it complies with the prescriptions of *simulated embodied agents* (Ziemke 1997). An agent consists of several modules, namely *perception*, *state*, *actions* and *control algorithm*. These (almost self-explanatory) modules depend on the application and are under the user's responsibility. The control algorithm module defines the type of autonomy of the agent: it is precisely inside this module that the designer decides whether to implement an operational autonomy or a behavioral autonomy (Ziemke 1997). Operational autonomy is defined as the capacity to operate without human intervention, without being remotely controlled. Behavioral autonomy supposes that the basis of self-steering originates in the agent's own capacity to form and adapt its principles of behavior: an agent, to be behaviorally autonomous, needs the freedom to have formed (learned or decided) its principles of behavior on its own (from its experience), at least in part.

For our purpose, and for sake of simplicity, we chose

[1] To our knowledge, Beckers et al. (R. Beckers and O.E. Holland and J.L. Deneubourg 1994) were the first to exploit a stigmergic coordination between robots. Stigmergic coordination means literally "incitement to work by the product of the work".

to implement operationally autonomous agents. Each agent decides by itself which action to take, according to local information. There is no master responsible for supervising the agents in the system, thus allowing the system to be more flexible and fault tolerant. Agents have neither explicit coordination features for detecting and managing antagonistic situations nor communication tools for negotiation. In fact they "communicate" in an indirect way, that is, via their influences in the environment. Under these constraints, several variants of control algorithm for our agents have been implemented and tried out. However, in this paper, we will focus on the following control algorithm: if an agent that does not carry an object comes to a cell containing `N` objects, it will pick one object up with a probability given by `N` to the power of `-Alpha`, where `Alpha` is a constant greater than or equal to zero; if an agent that carries an object comes to a cell containing some objects, it will systematically drop its object. If the cell is empty, nothing special happens and the agent will move to another cell. Note that an agent can not carry more than one object at a time. Such a simple control algorithm allows to explicitly modulate the probability of picking objects up as a function of the local density and it is a sufficient condition for the system to regroup objects. In (J.C. Deneubourg et al. 1991), the authors indeed showed in their model that a mechanism that involves the modulation of the probability of dropping objects as a function of the local density was sufficient to generate an observed sequence of clustering.

The State module encompasses the private information of the agent and of course depends on the control algorithm of the agent. In our case, it consists of the information relative to whether the agent carries or not an object plus other internal variables that include the states of their random number generators.

Agents can be endowed with several types of object perception and different moving strategies, thus leading to several families of agents. Yet the perception of other agents and/or obstacles in the environment is the same for all families in the sense that a cell containing an obstacle or an agent is implicitly perceived as a cell where the agent can not move to. Note that two agents can not come to the same cell; this induces spatial conflict. An agent endowed with what we refer to as *a basic object perception*, perceives only the quantity of objects that is present on the cell on which it stands. Such agents are endowed with random move capabilities to roam in the environment: at every step, a move direction is randomly drawn. Agents of this type will be referred to as *basic agents*. A variant is as follows: at every step, the agent, instead of randomly drawing a direction, is given a probability of `0.1` to randomly draw a direction and a probability of `0.9` to keep its direction. This family of agents is referred to as *inertial agents*. A last family of agents is that of *agents with camera*: every agent is given an artificial eye (pin-hole camera model (R. Horaud and O. Monga 1993) with the classical five intrinsic parameters and three out of the six extrinsic parameters, namely two translations out of three and one rotation out of three). So that, every agent, through its camera, is able to perceive objects (and objects only, not obstacle and not other agents) in (part of) the environment and goes towards the object which is the closest to its optical axis, thus avoiding wandering around in an area without interest.

Every agent can be tuned in numerous ways (`Alpha`, the lens of the camera) so that we have available a large number of individual behaviors. In a previous work, we conducted some studies on finding the appropriate parameters so as to optimize the global behavior of the system when varying the number of agents.

An agent is said to be *regrouping agent* if it is able to regroup all the objects into a single stack on a cell, in a finite time, when acting alone in the environment, for any initial configuration. Experimenting environments where some regrouping agents *concurrently* work is expected to yield a single stack containing all the objects. However this result does not appear as a goal embodied into the agents, but is generated by the recurrent interactions of the agents. It can be interpreted as a global goal, but such an interpretation is generally the description made by an external observer spying on the execution of the experiment.

In our experiment, conflict is naturally induced, due to the nature of the agents we consider. Even if there is only one agent in the system, the action of the agent at a given time can lead to an observed contradictory goal when compared to a previous goal suggested by a preceding action: the agent can indeed pick an object up from stack `A` and drop it on stack `B`, and then later, can do exactly the opposite actions, namely pick an object up from stack `B` and drop it on stack `A`. When several agents are present in the system, the different combined actions of all the agents will as well induce conflict.

Results

For a given size of the environment and a fixed number of objects, we have run an intensive number of experiments for two families of agents, namely *basic agents* and *agents with camera*. For *basic agents*, the size of the environment is 9 by 9 cells, there are 10 objects and we have run 400 experiments. For *agents with camera*, the size of the environment is 25 by 25 cells, there are 20 objects and we have run 1500 experiments. For every family, `Alpha` was set to `0.3` and experiments consisted in varying (i) the distribution of objects in the environment and (ii) the number of agents in the system, and in measuring the cost[2] of the system to achieve the task of object regrouping. Moreover, the same experiments have been repeated with a simple obstacle present in the south-west part of the environment (see figures 1 and 2).

[2]The cost of the system is defined as the number of iterations necessary to the system to achieve the task.

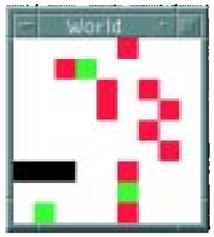

Figure 1: A 9 x 9 cell environment with 3 *basic agents*, 10 objects and a simple obstacle: red squares (grey if printed in B/W) represent objects, green squares (light grey if printed in B/W) represent agents, black squares (dark grey if printed in B/W) materialize the obstacle.

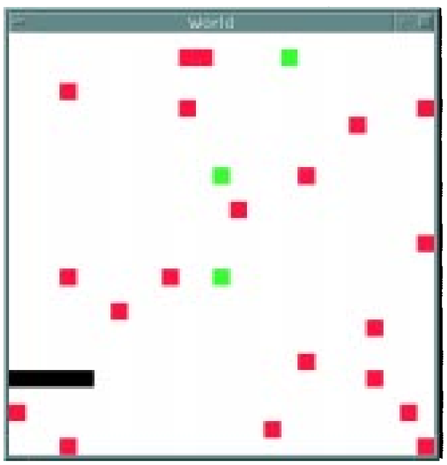

Figure 2: A 25 x 25 cell environment with 3 *agents with camera*, 20 objects and a simple obstacle.

Preliminary results were presented at the time of the ECAI'98 workshop on Conflicts (see (Chantemargue *et al.* 1998a)): these results were obtained with very simple environments (no obstacle), but they already introduced some features of the collective system, namely the fact that implicit cooperation takes place in the system, and the fact that conflict is naturally induced in the system. These results conform to the results presented in (R. Beckers and O.E. Holland and J.L. Deneubourg 1994). Our system has been deeply studied since. First, we quantified the different types of conflict that arise in the system. Second, we analyzed the influence of obstacle on the outcome of the system (i.e. the location of the final stack). We present hereafter the most significant results.

The first result yielded for all tried parameters, is that the collective system achieved the global task of object regrouping. Figure 3 (respectively figure 5) displays the cost for *agents with camera* (respectively *basic agents*) versus the number of agents in the system[3].

[3]Vertical bars on all charts indicate standard deviations.

Figure 4 (respectively figure 6) displays the speedup in the system for *agents with camera* (respectively *basic agents*) versus the number of agents in the system. The speed-up for n agents is defined as the ratio of the mean cost for one agent to the mean cost for n agents.

Speed-up charts show how the performance of the system scales with the number of agents. To some extent, the more the number of agents in the system, the better the performance of the system. This illustrates a feature in our system, namely that of cooperation: agents participate in achieving the global task without being aware of it and without any explicit cooperation protocol, but just in virtue of their design and the context in which they operate. A form of implicit cooperation takes place in the system through the recurrent actions of the agents. Apart from the fact that the obstacle increases the cost in the system (figures 3 and 5), it does not really alter the property of cooperation (figures 4 and 6).

In the case of *basic agents*, the speedup is approximately linear (even supra-linear) up to 4 agents (figure 6). The control algorithm of these agents is extremely sub-optimal with respect to the task; the context given by the presence of a few agents appears to improve the efficiency of the control algorithm. This is also true (but linearity only) for the case of *agents with camera* in the presence of an obstacle: a single agent is greatly perturbed by the obstacle (due to its design, it can be persistently obstructed by an obstacle when going towards an object); the "noise" generated by the presence of other agents attenuates this perturbation. Notice that in our experiments, the obstacle can hardly "trap" more than one agent at a time (due to its geometry).

The different types of conflict that may arise in our system are referred to as spatial conflicts and (sub-)goal conflicts. Spatial conflicts represent the number of times (per iteration per agent) an agent has been perturbed by other agents in its neighborhood when moving. Figure 8 (respectively figure 10) displays the spatial conflicts that were measured in the system versus the number of agents for *agents with camera* (respectively *basic agents*). (Sub-)goal conflicts (or useless operations) have been quantified by measuring the aggregate number of extra operations done by the agents in the system. `N-1` pick-up operations are theoretically enough to regroup `N` objects onto a cell containing already an object; the number of extra operations in the system will be determined by measuring the total number of pick-up operations in the system minus the theoretical number. Figure 7 (respectively figure 9) displays the (sub-)goal conflicts that were measured in the system versus the number of agents for *agents with camera* (respectively *basic agents*).

In the case of *basic agents*, the number of spatial conflicts increases linearly with the number of agents: this is expected due to their moving strategies. In the case of *agents with camera*, the use of the camera strongly reduces the parts of the space used for moving, thus in-

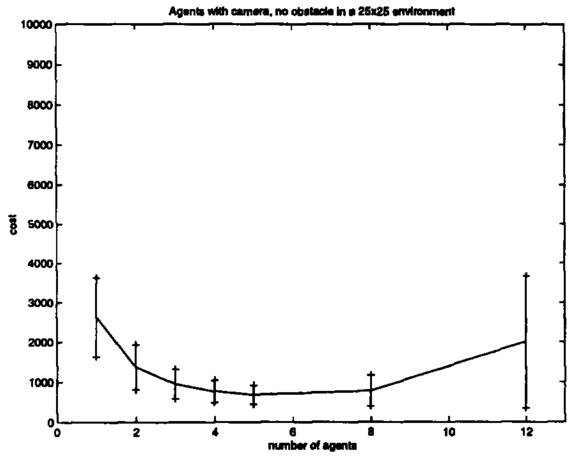

Figure 3: *Agents with camera*: cost to complete the task a) without obstacle, b) with obstacle.

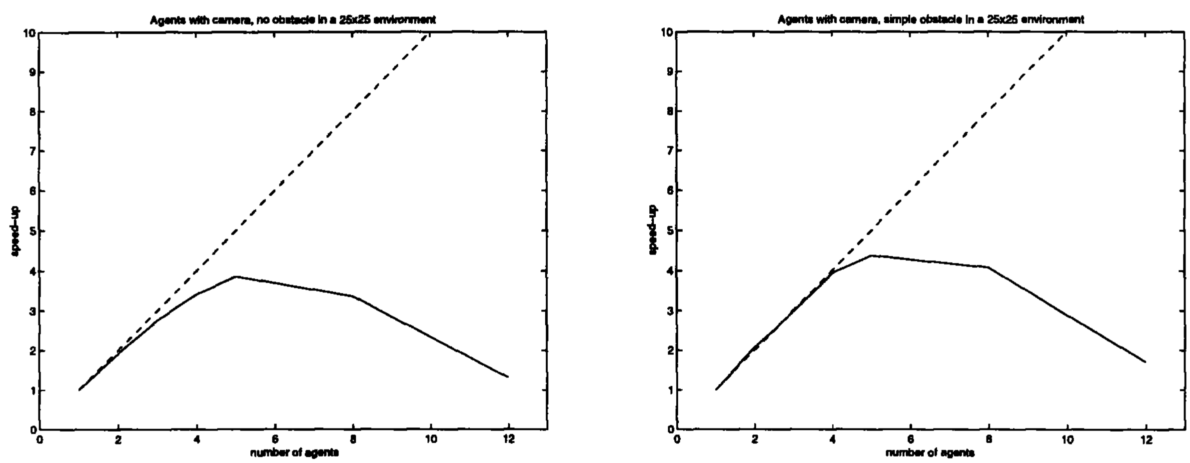

Figure 4: *Agents with camera*: speedup a) without obstacle, b) with obstacle. The dotted line indicates a linear speed-up.

creasing strongly the number of spatial conflicts as the number of agents grows.

In the case of ***basic agents***, the number of (sub-)goal conflicts decreases in the system with the number of agents ranging from 1 to 6. This is probably due to the fact that the control algorithm for these agents is very rudimentary and therefore not optimized for a single agent to act in the system. This relates to the supra-linearity observed in the speedup.

We have further run 400 experiments with a system involving ***inertial agents*** with `Alpha` set to `0.3`. The size of the environment is 9 by 9 cells and the number of objects is 10 (experimental conditions identical to those of ***basic agents***). Figures 11 and 12 display respectively the cost and the speed-up that were obtained. Figure 13 displays the (sub-)goal conflicts that were measured in the system versus the number of agents. Figure 14 displays the spatial conflicts that were measured in the system versus the number of agents. As for ***basic agents*** and for the same reasons, the number of spatial conflicts increases linearly.

We have also run 400 experiments (data not shown here) with a system involving ***agents with camera*** with `Alpha` set to `0.3`, for which the size of the environment is 9 by 9 cells and the number of objects is 10 (experimental conditions identical to those of ***basic agents***). Compared to ***basic agents***, ***agents with camera*** strongly reduce the cost. Moreover and especially for two or three agents in the system, they reduce the number of useless operations. On the other hand, the range of linearity in the speed-up is significantly reduced (especially in the absence of obstacles). There are at least two reasons for this optimization. The most obvious one is the use of the camera to guide movement. The other one, less obvious, is the fact that agents keep their direction for a while. This last feature is found in the

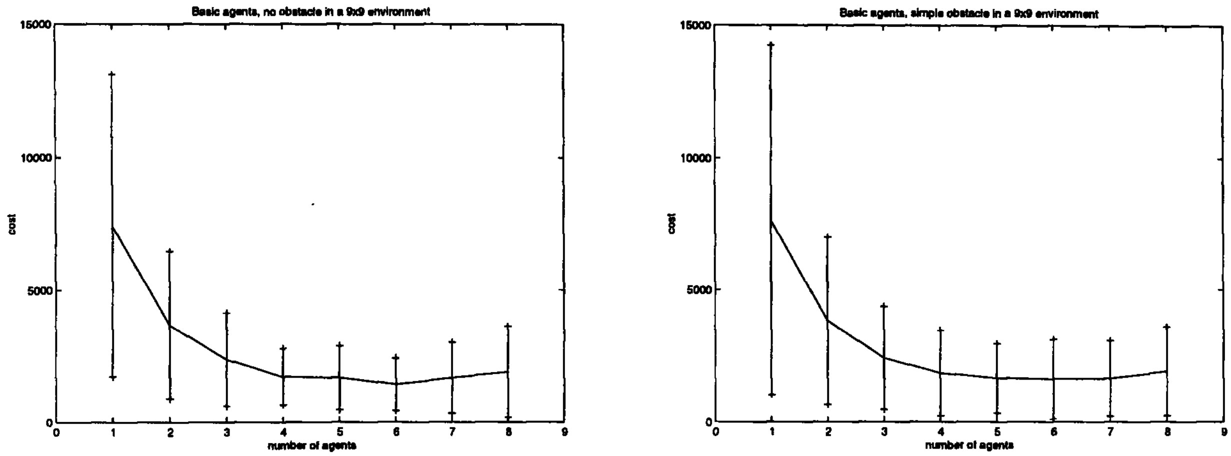

Figure 5: *Basic agents*: cost to complete the task a) without obstacle, b) with obstacle.

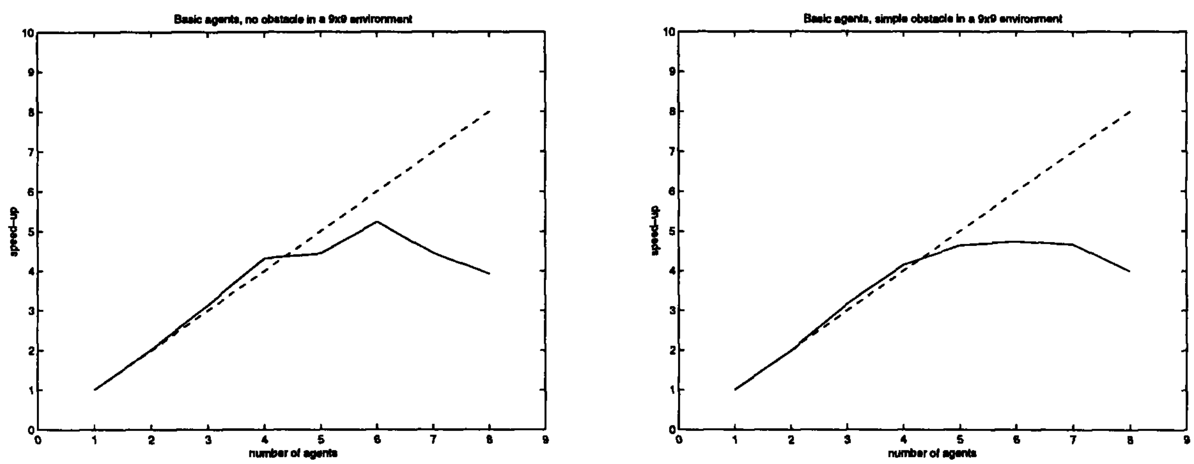

Figure 6: *Basic agents*: speedup a) without obstacle, b) with obstacle. The dotted line indicates a linear speed-up.

moving strategy of ***inertial agents***. Compared to ***basic agents***, ***inertial agents*** significantly reduce the cost and the number of useless operations. Furthermore, as they exhibit a larger range of (supra-)linearity in the speed-up, they improve the robustness of the system.

There is nowhere in the agent something encoded that specifies where to regroup objects in the environment: the location of the stack containing all the objects at the end of a regrouping experiment is indeed the result of the agents' interactions; it depends on the number of agents, the number of objects and their locations, the presence of obstacles and their locations. This illustrates another feature in our system, namely that of self-organization. The presence of obstacles in the environment influences the location of the stack, not only due to obvious spatial constraints: it may modify the spatial distribution of the final stacks.

In our experiments, even though we can bet on the global task to be achieved (from a thorough examination of the control algorithm of every agent), we can not really preempt the manner that will be used to achieve the task nor the exact location of the final stack containing all the objects, due to the non determinism in the system's actions. With the environment depicted in figure 15 (11 by 11 cells, 20 objects and a complex obstacle), we have run 600 experiments involving ***inertial agents*** with `Alpha` set to 1, so as to study the influence of such an obstacle on the spatial distribution of the final stack. Note that in this environment, the number of free cells (that may contain objects) in the north part is the same as the one in the south part. Results on the distribution of the locations of the stacks containing all objects at the end of experiments suggest that certain configurations of obstacle do influence the outcome of the system. Figure 16 displays the frequencies of the locations of the final stacks: the frequency of building the stack in the north part of the environment is significantly higher than that of building it in the south

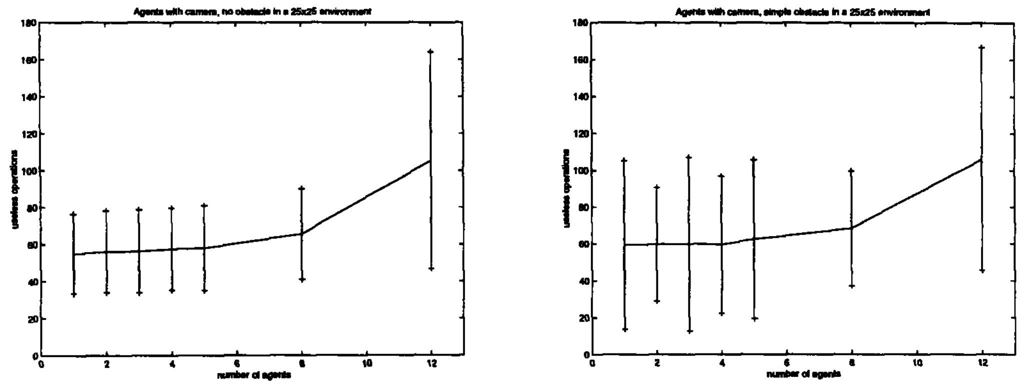

Figure 7: ***Agents with camera***: useless operations a) without obstacle, b) with obstacle.

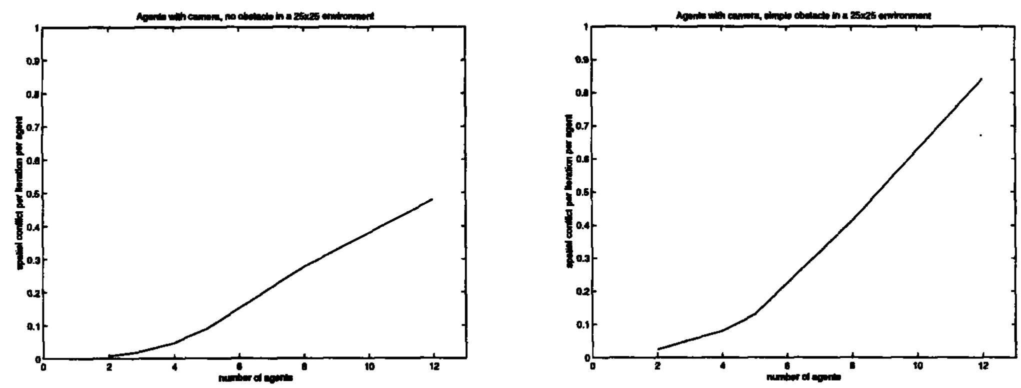

Figure 8: ***Agents with camera***: spatial congestion a) without obstacle, b) with obstacle.

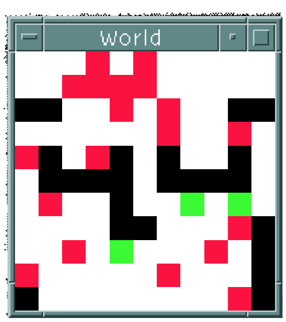

Figure 15: A 11 x 11 cell environment with 3 ***inertial agents***, 20 objects and a complex obstacle.

Figure 16: Location frequencies of the final stacks

part.

Conclusion

In this paper, simulations of a simple system of object-regrouping agents have been presented. In our system, agents decide by themselves over time the action to take in relation to their internal state and their perception. The collective behavior (at the system level) is implicitly driven by the individual behaviors (at the agent level): we can speak of a certain behavioral autonomy at the level of the system, though the autonomy is operational at the level of each agent. There is no supervisor in the system and the global task to be achieved, viz regrouping objects, is not encoded explicitly within the agents; the environment is not represented within the agent and there is no explicit cooperation protocol between agents. Therefore in our experiments, the

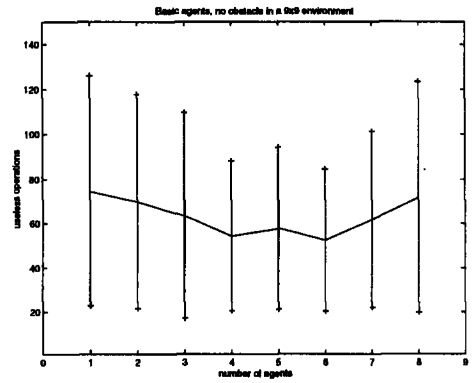

Figure 9: *Basic agents*: useless operations a) without obstacle, b) with obstacle.

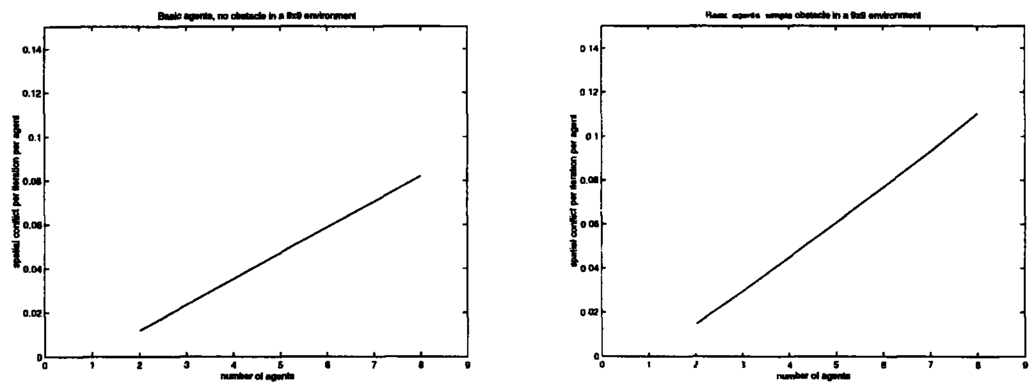

Figure 10: *Basic agents*: spatial congestion a) without obstacle, b) with obstacle.

global task is achieved by virtue of implicit cooperation, where self-organization plays a role. Advantages of such an approach are flexibility, adaptivity and fault tolerance, since agents and obstacles can be added to or removed from the system to some extent without running severely into trouble.

We pointed out different kinds of conflict. The system was observed under the effects of these forms of conflict, instead of explicitly handling to avoid them. The results we obtained suggest that, despite the increase of spatial conflict, the cost can quickly be decreased; in particular the less individual behaviors are optimized, the higher are the chances that a strong form of cooperation (supra-linearity) takes place, and this appears to be true in spite of several forms of conflict that are natural to the system. Furthermore, different configurations of obstacle can be used to influence the distribution of the location of the final stacks. Finding some means to control an autonomy-based multi-agent system so as to compel it to fulfill some specific tasks would indeed be very profitable.

Acknowledgements

This work is financially supported by the Swiss National Foundation for Scientific Research, grants 20-05026.97 and 21-47262.96.

References

Bonabeau, E., and Theraulaz, G. 1994. *Intelligence collective*. Paris: Hermes. in french.

Brooks, R. 1986. A robust layered control system for a mobile robot. *IEEE Robotics and Automation* RA(2):pp. 14–23.

Brooks, R. 1991. Intelligence without Reason. In *Proceedings of IJCAI-91.*

Chaib-Draa, B. 1996. Interaction between Agents in routine, familiar and unfamiliar situations. *International Journal of Intelligent and Cooperative Information Systems* 5.

Chantemargue, F.; Courant, M.; Dagaeff, T.; and Robert, A. 1998a. Conflict: a pragmatic approach. In *Proceedings of ECAI'98: Workshop on Conflicts among agents: avoid or use them?*

Chantemargue, F.; Krone, O.; Schumacher, M.; Dagaeff, T.; and Hirsbrunner, B. 1998b. Autonomous

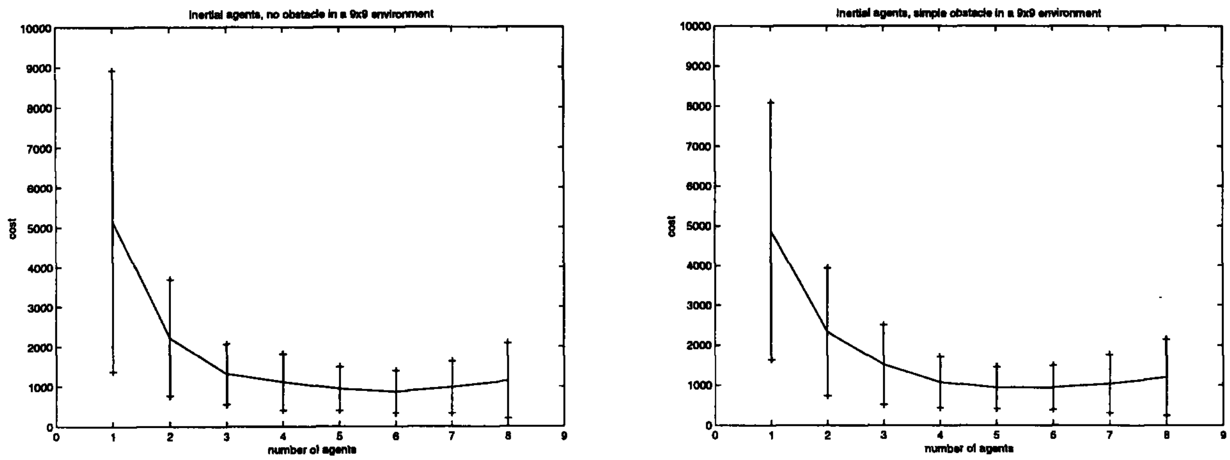

Figure 11: *Inertial agents*: cost to complete the task a) without obstacle, b) with obstacle.

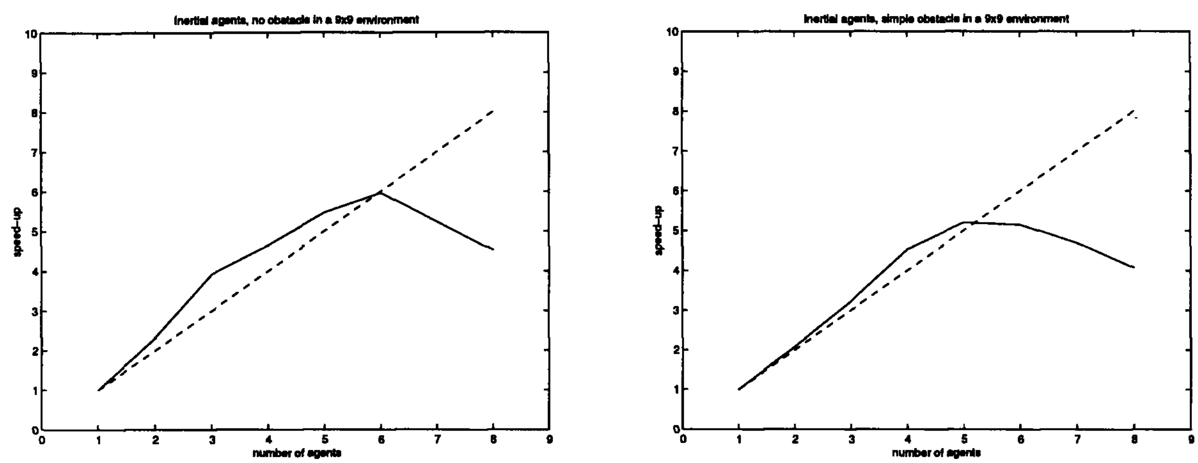

Figure 12: *Inertial agents*: speedup a) without obstacle, b) with obstacle. The dotted line indicates a linear speed-up.

Agents: from Concepts to Implementation. In *Proceedings of the Fourteenth European Meeting on Cybernetics and Systems Research (EMCSR'98)*, volume 2, 731–736.

Cholvy, L. 1998. A general framework for reasoning about information provided by contradictory agents and some of its applications. In *Proceedings of ECAI'98: Workshop on Conflicts among agents: avoid or use them?*

Ferber, J. 1995. *Les systèmes multi-agents, vers une intelligence collective.* Paris: InterEditions. in french.

Forrest, S. 1990. Emergent Computation: Self-organizing, Collective and Cooperative Phenomena in Natural and Artificial Computing Networks. *Physica D* 42.

Gaussier, P., and Zrehen, S. 1994. A constructivist approach for autonomous agents. In Thalmann, I. D., and N., eds., *Artificial Life in Virtual Reality*, 97–113. London: John Wiley and Sons.

J.C. Deneubourg et al. 1991. The dynamics of collective sorting: Robot-like ant and ant-like robot. In J.A. Mayer and S.W. Wilson., ed., *Simulation of Adaptive Behavior: From Animals to Animats*, 356–365. MIT Press.

Kwong, B.; Wong, K.; and Low, B. 1998. A logical framework for reasoning over attacking conflicts in multi-agent systems. In *Proceedings of ECAI'98: Workshop on Conflicts among agents: avoid or use them?*

Lamontagne, L., and Benhamou, P. 1998. Conflicts in search and rescue coordination. In *Proceedings of ECAI'98: Workshop on Conflicts among agents: avoid or use them?*

Langton, C.; Minar, N.; and Burkhart, R. 1996. The Swarm simulation System: a toolkit for build-

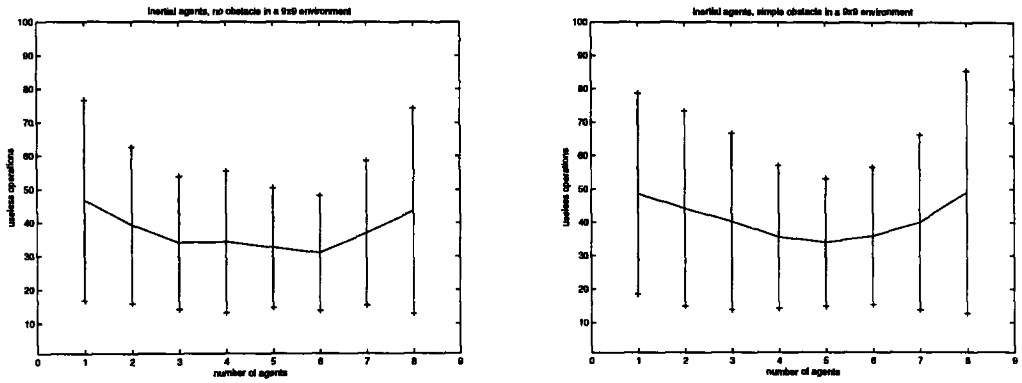

Figure 13: *Inertial agents*: useless operations a) without obstacle, b) with obstacle.

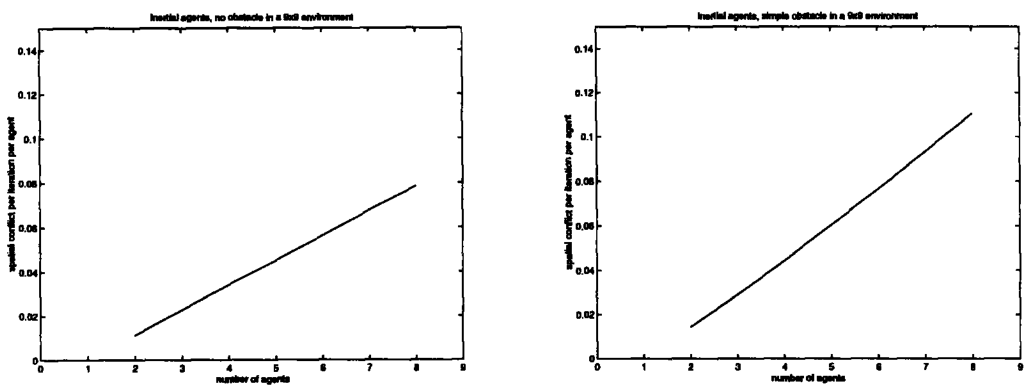

Figure 14: *Inertial agents*: spatial congestion a) without obstacle, b) with obstacle.

ing Multi-agent simulations. Technical report, Santa Fe Institute.

Lerena, P., and Courant, M. 1998. Mate Choice and NK-Landscapes. In *Proceedings of the Fifth International Conference of the Society for Adaptive Behavior (SAB98).*

Malone, T., and Crowston, K. 1994. The Interdisciplinary Study of Coordination. *ACM Computing Surveys* 26(1):87–119.

Martinoli, A., and Mondada, F. 1995. Collective and Cooperative Group Behaviours: Biologically Inspired Experiments in Robotics. In *Proceedings of the Fourth Symposium on Experimental Robotics ISER-95.*

Martinoli, A., and Mondada, F. 1998. Probabilistic Modelling of a Bio-Inspired Collective Experiment with Real Robots. In *Proceeding of the Third International Symposium on Distributed Autonomous Robotic Systems DARS-98.*

Mataric, M. 1992. Minimizing complexity in controlling a mobile robot population. In *Proceedings of the International Conference on Robotics and Automation.* Los Alamitos: IEEE Computer Society Press.

Pfeifer, R. 1995. Cognition - Perspective from autonomous agents. *Robotics and Autonomous Systems* 15:47–70.

R. Beckers and O.E. Holland and J.L. Deneubourg. 1994. From local actions to global tasks: stigmergy and collective robotics. In Brooks, R., and Maes, P., eds., *Fourth Workshop on Artificial Life.* Boston, MA, USA: MIT Press.

R. Horaud and O. Monga. 1993. *Vision par ordinateur: outils fondamentaux.* Hermes.

Robert, A.; Chantemargue, F.; and Courant, M. 1998. Grounding Agents in EMud Artificial Worlds. In *Proceedings of the First International Conference on Virtual Worlds.* Paris, France: Springer-Verlag.

Varela, F.; Thompson, E.; and Rosch, E. 1991. *The embodied mind: Cognitive science and human experience.* Cambridge, MA: MIT Press.

Ziemke, T. 1997. Adaptive Behavior in Autonomous Agents. *Autonomous Agents, Adaptive Behaviors and Distributed Simulations' journal.*