Exploiting Physical Constraints: Heap Formation through Behavioral Error in a Group of Robots

Marinus Maris

Artificial Intelligence Laboratory
Department of Computer Science
University of Zurich
Winterthurerstrasse 190
8057, Zurich, Switzerland
email:maris@ifi.unizh.ch

René te Boekhorst

Artificial Intelligence Laboratory
Department of Computer Science
University of Zurich
Winterthurerstrasse 190
8057, Zurich, Switzerland
email:boekhors@ifi.unizh.ch

Abstract

In this paper we describe a collective heap building process by a group of robots. Instead of predefining "cognitive" capacities, we exploit the physical structure of the robots and the self-organizing properties of group processes. The robots' control program effectively contains just one behavioral rule to avoid detected obstacles. Due to the constrained sensory input, the robots collide with objects that are exactly in front of them. In this way, objects are pushed and clusters are formed. To study the dynamics of the cluster process we conducted experiments in which the number of objects and robots were varied. We found that a limited amount of mutual interference is crucial for the fusion of clusters; hence large, single heaps did never emerge in trials with just one robot. However, with more than 4 robots the heap building process slows down due to increased mutual avoidance movements.

1 Introduction

Within the field of autonomous mobile robots, an increasing tendency can be observed towards employing design methods that exploit constraints that are consequences of the morphology of the robot, and control cues that arise from interactions of the robot with its local environment. [1, 2, 3]. This differs from traditional approaches in which robot control is usually based on the manipulation of a set of predefined rules, often regulated by a central computing device, see for example [4]. For a robot that needs to perform in a continuously changing dynamic environment, the real time constraints will be difficult to handle if every action of the robot is to be controlled by a central planner without appropriate feedback from the environment.

Yet, at the same time we observe that even the most simple animal is able to perform remarkable feats. The fundamental issue here is that complex appearing patterns can be the result of a limited set of simple rules that steer the interactions between entities and their environment. The environmental structure, changed by the activities of the entities, affects the behavior of these agents which in turn influences the environment.

This principle has been most successfully applied to the study of social insects [5,6]. Model work and empirical evidence have made plausible that many of the seemingly intelligent "collective decisions" in the foraging behavior of ants result from the auto-catalyzed responses of a large number of, in itself inflexible, entities by using simple local cues (pheromone concentrations). Likewise, the intricated patterns of honey combs [7] and the impressive structures build by termites [8] can be explained in similar terms, without assuming complex computations, global knowledge or a central controller monitoring and regulating the activities of the workers.

These insights are an obvious source of inspiration for multi-robot oriented applications. For instance Beckers et al. [8] realized a straightforward implementation of the ideas of Deneubourg et al. [5], heap formation by ants, using a number of small mobile robots. These robots avoid each other and walls but not pucks, which they push around with a C-shaped shovel. When at least three pucks are collected in the shovel, a switch activates a turn-away reflex. After at least one and a half hour, always one large heap was formed. It is important to note that in this experiment a single robot could create one heap on its own. Adding more robots brought about only a proportional reduction of heap building executing time. In this sense, the work of Beckers et al. does not constitute a typical example of self-organized pattern formation through reinforcement by the presence of others.

In this paper, we report a similar task using a group of robots. Our robots lack specific actuators for handling objects and do not distinguish the objects to be collected from other obstacles. Clusters are formed because the robots make "mistakes" due to their sensor positions and mutual avoidance movements. The emergence of clusters in our experiments can therefore be seen as an example of

Proc. IROS 96
0-7803-3213-X/96/ $ 5.00 © 1996 IEEE

what Deneubourg et al. [6] have called a "strategy of errors".

Replicated trials under controlled conditions (systematic variation in the number of robots and objects) show that the robots collect the objects in about 20 minutes, often, but not always, in one cluster. The results also indicate that it is highly unlikely that a single robot is able to build just one, large sized cluster. Finally, the displacements of small clusters due to mutual avoidance movements among robots appear to be crucial.

2 Robot Control Architecture

Our robots have been developed for teaching and research purposes, and hence are called DidaBots (from: Didactical Robots) (Figure1). Due to their straightforward construction, Didabots have proven to be robust, very flexible and easy-to-use. The robots (23 x 13 cm) are described in detail in [9]. The standard configuration consists of six infrared (IR) sensors for obstacle detection; six sensors for ambient light measurements; differential steering; wheel encoders for speed control and two external signalling devices (a beeper and a light bulb) to inform about the robot's internal state. Power supply is from two 6V batteries. The robot is autonomous in the sense that it carries a micro processor board (196KD, Intel) running its control program. The speed of the robots ranges continuously from 0 to 10m/s. For the experiments described here, the maximum speed of the robots is set to 1m/s.

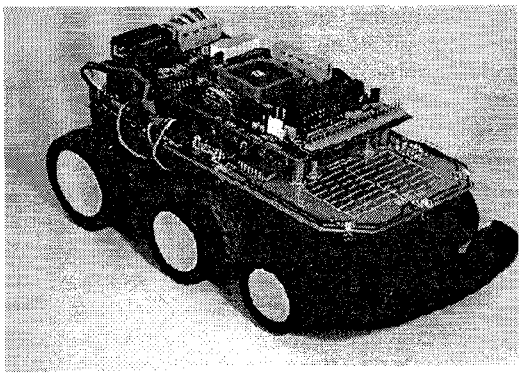

Figure 1. The mobile robot Didabot.

The control program of the robots is based on a Braitenberg type of network [10]. The activation of a sensor is multiplied by a weight value which connects each sensor to both of the two motors (Figure 2). The products are summed to obtain an output value that determines the activation of each of the motors. In turn, the motors control the rotation speed of the two middle wheels and by changing the weights, the behavior of the robot can be manipulated.

If the robot detects an obstacle to its right, IR4 is activated. The excitatory weight (w+) forces forward movement on the right wheel and the inhibitory weight (w-) induces backwards movement on the left wheel. As a consequence, the robot will turn away from the obstacle.

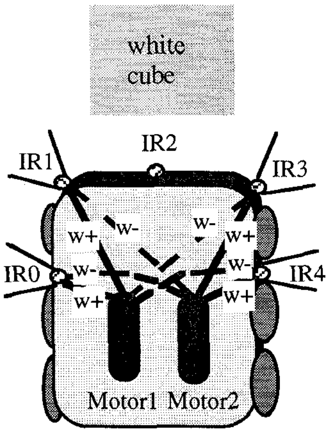

Figure 2. Control scheme of the robot. In the experiments the weights between IR2 and the motors are set to zero (not drawn). This causes the white cube to be pushed along when it appears exactly in front of the robot. The rear sensor is omitted for clarity.

For the experiments, the weights of the frontal IR-sensor are set to zero. This has the effect that the Didabot cannot "see" anything that is exactly in front of it and too small to be detected by any of the other IR-sensors. It implies that the Didabot collides with such an object and pushes it until the robot detects another obstacle (a wall, another object or another Didabot). Then an avoidance movement occurs and the shifted object is left behind. If the shifted object is deposited close enough to another, they form a constellation that is easily detected and avoided by the robots. In this manner, clusters are created. Objects that are pushed against the wall will not be removed anymore and are "lost" for cluster building.

3 Experimental Setup

The experiments were performed in an arena of 230 x 260 cm. As objects we chose polystyrene white cubes of 8 cm on a side. The experiments were observed by a "bird's eye" color CCD camera (hung straight above the arena) that sent its images to a computer (PowerMacintosh 8100). The starting positions of the cubes were generated by drawing random coordinate values from a uniform distribution and were displayed as dark squares on the same window as the one for the camera image. We then shoved the cubes in the arena so that they covered the corresponding squares on the screen (Figure 3a). This enabled a quick and objective initialization of the experiments.

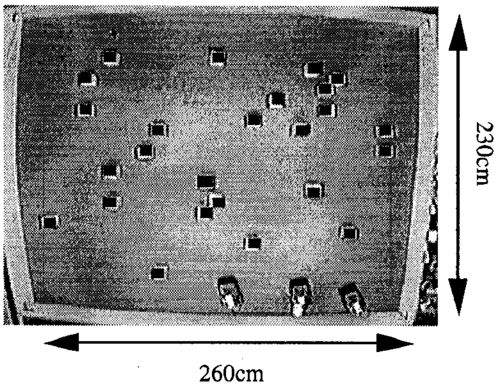

Figure 3a . The experimental setup.The dark squares indicate the initial locations of the cubes.

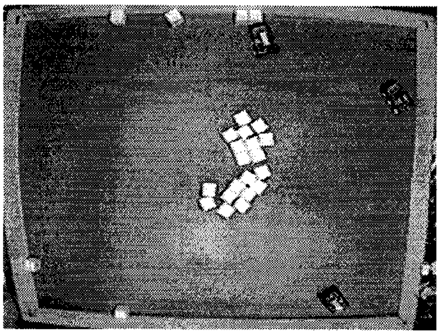

Figure 3b. Typical end configuration.

At the beginning and end of each experiment, one frame was recorded (Figures 3a, 3b) and stored together with the corresponding data file. In this way, we documented the initial and final spatial configuration of the cubes. To analyze the movements of the robots over time, trajectories were recorded during several tests. For this, the light bulb of one of the robots was switched on so that the route of the Didabot could be traced by the camera system (Figure 4). Moreover, a number of trials were video-taped for further ethological analysis.

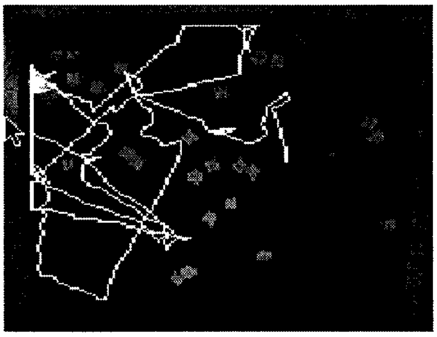

Figure 4. Recorded trajectory of the robot (70secs).

The effects of two main factors were studied, namely the influence of the number of cubes (12 or 25) and of the number of robots (ranging from 1 to 5). Each combination of factors was replicated three times, giving 2*5*3 = 30 trials. Each trial lasted 30 minutes. The particular experiment to be run was drawn randomly from a normal distribution.

All contacts between robots and cubes were recorded continuously. Every 15 seconds we counted the number of cubes along the wall as well as the frequency of clusters of all sizes. Cubes were considered to belong to a cluster whenever the distance to their nearest neighbour was smaller than the size of a cube. The data were recorded as keystroke codes and automatically transferred to a data file. The contacts were summed per 15 second time interval (= pushing rate) and stored together with time stamp, the distribution of cluster sizes and data about the number of cubes pushed against the wall.

4 Data Analysis

The development of clusters could be characterized by a variety of measurements such as mean cluster size, the number of clusters formed and the number of cubes pushed against the wall per unit time.

We used two factor analysis of variance to tease out the effects of the number of cubes, robots and their statistical interaction on the measurements mentioned above. Regression analysis (after scale transformation) was applied to describe how pushing rate changed in time. When the assumptions of parametric statistics were not met, nonparametric tests were used instead. The level of significance is put at a = 0.05.

Within half an hour the robots shifted about 60% of 25 randomly distributed cubes onto heaps and the remainder against the wall. With 12 cubes, only an average of 40 % ended up in clusters (Figure 5b). Most often, one or two clusters were formed (Figure 5a). The size of the clusters was related to their frequency; when more heaps emerged, they were smaller (Figure 5c) and their sizes were unequally distributed (data not shown). The final number of clusters and mean cluster size depended significantly on both the number of cubes and robots (Figure 6, Table I). When presented with only 12 cubes, on average 2.3 small clusters (with a mean size of 4 cubes per cluster) emerged. Using 25 cubes, a mean of 3.4 clusters was formed which contained on average 7.92 cubes. More clusters also resulted when either one or five robots were set in, but this effect was only visible in the

25 cube experiments. This explains the almost significant interaction term in the ANOVA table (Table I).

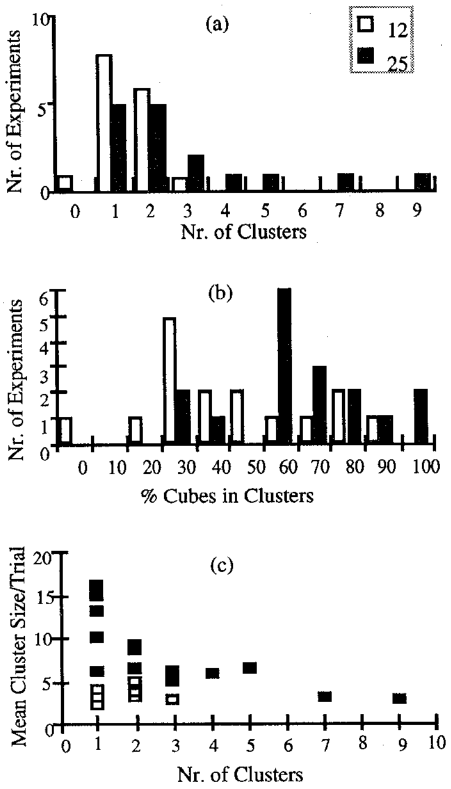

Figure 5. Frequency distributions of the number of clusters (a), the percentage of cubes in heaps (b) and cluster size (c).

Source	SS	df	MS	F	p
Main Effects	1.980	5	0.396	4.117	0.010
Nr. of Robots (R)	1.216	4	0.304	3.161	0.036
Nr. of Cubes (C)	0.764	1	0.764	7.942	0.011
Interaction: (RxC)	1.024	4	0.256	2.660	0.063
Rest	1.924	20	0.096		
Total (Corr.)	4.928	29			

Table I. Analysis of variance for the effects of numbers of robots and objects on the number of heaps formed.

The number of contacts that the Didabots made with the objects decreased quickly in time and in all cases negative power functions gave a good fit to the data. Estimates of the intercepts, but not of the slopes, differed significantly between trials with different number of robots (Kruskal-Wallis Test, H =11.708, p = 0.02). Hitting objects occurred at a higher rate in experiments with more Didabots, however, for five robots the trend reversed.

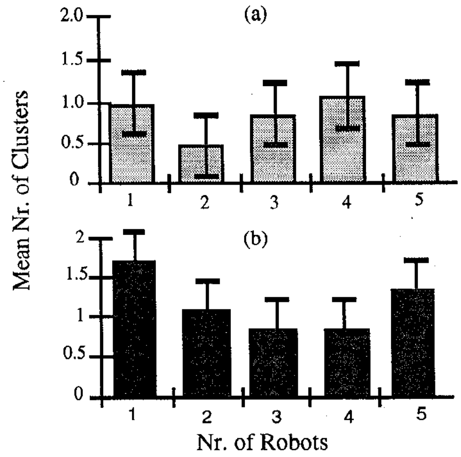

Figure 6. Dependency of final number of clusters on the number of robots for experiments with 12 (a) and 25 (b) obstacles. Bars show the 95 % confidence intervals around the mean (based on pooled estimate of within variance).

To demonstrate the dependency of robot behavior on the structure of the environment and vice versa, we recorded the trajectories of one of the robots with the camera system. Initially, the robot meanders between the obstacles and no systematic pattern is apparent in its trajectory (Figure 4). When cluster formation progresses, the Didabot more often moves along the sides of the arena. Finally, when one cluster dominates, it circles around the heap without hardly touching it (Figure 7). Correspondingly, the pushing rate decreases with further patterning of the environment. In Figure 8 we visualized this by plotting robot behavior (pushing rate) and environmental structure (mean cluster size, final number of clusters) against each other.

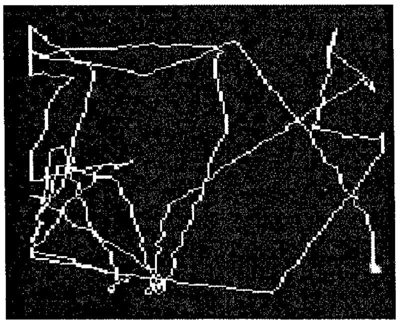

Figure 7. Trajectory at the end of the experiment

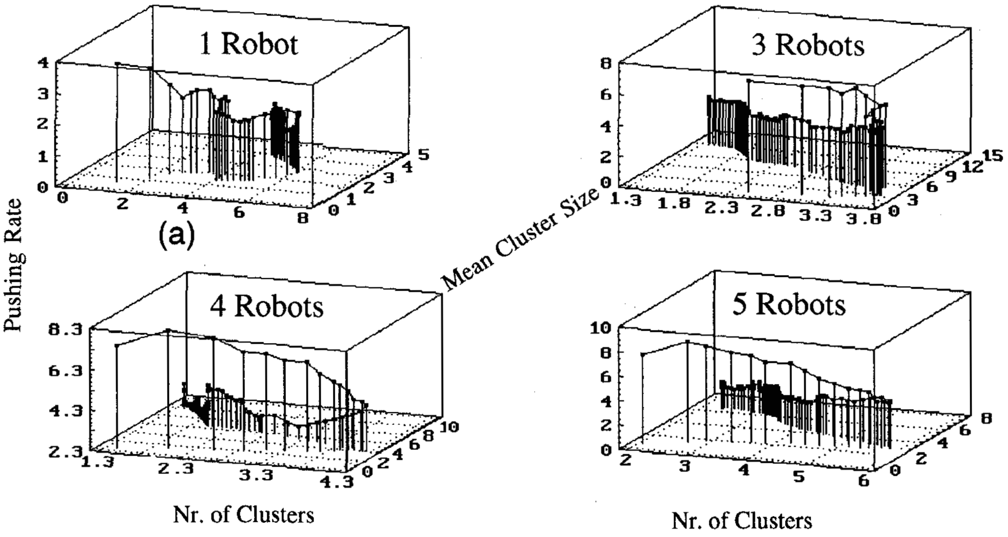

Figure 8. Changes in the robot-environment system displayed as trajectories in a 3D state space. Dimensions of the state space are *pushing rate (number of collisions by the robots), number of clusters formed and mean cluster size* (structure of the environment). Each curve was obtained by averaging the curves of the three separate trials for experiments involving various numbers of robots and by smoothing over five neighbouring values.

5 Discussion

In this heap building process, three stages can be distinguished that roughly correspond with those described by Beckers et al. [8]. Initially, the robots collide almost at random with the objects. During this stage many obstacles are brought together in pairs or are pushed against the wall (and therefore effectively removed from the arena). Because most of this "removal" takes place before clusters arise, the absolute number of objects pushed against the wall is almost independent of the initial number of obstacles. This explains why the percentage of clustered obstacles in experiments with 25 cubes is about twice that of experiments with 12 cubes. In Figure 8 we see that mean cluster size and the number of clusters increase over time, but at a decreasing rate; this is reflected in the smaller distances between points on the trajectories after the initially rapid formation of pairs. The slow formation of somewhat larger heaps characterizes the second stage of the process. The increased patchiness improves the robots ability to avoid collisions, which slackens the clustering of objects even more until the process almost comes to a halt. When only one robot is employed, this effect occurs when there are still a relatively large number of small heaps; the environment is sufficiently structured for the Didabot to manoeuvre, almost without hitting obstacles. This appeared to be a stable situation. In three additional experiments with one Didabot we prolonged the test duration to one and a half hour and found no change in the configuration. The contact rate at the end of these long trials dropped to an average of six pushes in a period of five minutes. Adding subsequently a second and a third Didabot (every 5 minutes), the contact rate increased to a mean of respectively 15.3 and 25 hittings per five minutes. The supra-linear increase was due to the robots avoiding each other in the vicinity of clusters which results in "erroneous" pushes. In one of the additional experiments, a constellation of three heaps that had been stable for more than one hour fused into a single cluster within five minutes after the introduction of a third Didabot. In contrast, none of the nine single robot trials resulted in the formation of a large single cluster.

We tried to influence cluster formation by adding a seed of four blocks in the middle of the arena and ran 5 trials with three robots and 25 cubes to test the results. In every trial, after at most 15 minutes, one heap resulted consisting of a large number of cubes (on average 17 blocks). This shows that the cluster forming process can easily be bootstrapped and that, with simple modifications, this process can be engineered such that it may be successfully employed in a typical robotics task like cleaning an area.

6 Conclusion

We have shown that simple robots, exploiting their physical structure and the nonlinear effects of group processes, can build patterns by structuring their environment. The heap building is a typical example of a self-organized (robot) group process. To attain a single cluster, more than one robot is needed to get the mutual interactions that generate the crucial "mistakes" in the obstacle avoidance behavior of the robots.

7 Future Work

Figures 8 a-d are actually phase plots that describe the state of the robot-environment system as a point in a 3D space. Changes in the state of the system correspond with movements of the point along trajectories in this state space. In accordance with this representation and in line with a view that becomes increasingly popular in the autonomous agent community, we can interpret our results using the concepts of dynamical system theory (see also Beer [11]). The next step in the analysis of the process described here is to formalize the system in terms of a mathematical model.

References

[1] J. H. Connell, *Minimalist mobile Robotics: a colony style architecture for an artificial creature.* Academic Press, SanDiego, CA, 1990.

[2] D. Lambrinos, C. Scheier and R. Pfeifer, "Unsupervised classification of sensory-motor states in a real world artefact using a temporal Kohonen map". In *Proceedings of the International Conference in Artificial Neural Networks* ICANN'95, 467-472, Paris, October 1995.

[3] R. Pfeifer, "Cheap Designs: Exploiting the Dynamics of the System-Environment Interaction", In *Conference on Prerational Intelligence - Phenomenology of Complexity Emerging in Systems of Agent Interacting Using Simple Rules,* Center for Interdisciplinary Research, University of Bielefeld, 81-91.

[4] J. Najera, F. de la Rosa and C. Laugier, "Planning Robot Motion Strategies under Geometric Uncertainty Constraints", in "*Proceedings of the International Conference on Intelligent Robots and Systems*", vol 1, 462-469, 1994.

[5] J.L. Deneubourgh, S. Goss and J. M. Pasteels, "The Dynamics of collective sorting: robot-like ants and ant-like robots, In *Simulation of Adaptive Behavior: from Animals to Animats.* J.A.Meyer and S. Wilson (eds), MIT Press, 1990.

[6] J.L. Deneubourgh, S. Aron, S. Goss and J. M. Pasteels, "Probabilistic behaviour in ants: a strategy of errors ?," *Journal of Theoretical Biology,* Vol. 105, 259-271, 1983.

[7] S. Camazine, "Self-organizing pattern formation on the combs of honey bee colonies," *Behavioural Ecology and Sociobiology,* Vol.28, pp. 61-76.

[8] R. Beckers, O. E. Holland, and J. L. Deneuborugh, "From Local actions to Global Tasks: Stigmergy and Collective Robots", in *Proceedings of the Fourth International Workshop on the Syntheses and Simulation of Living Systems,* MIT Press, 1994.

[9] M. Maris and R. Schaad, "The didactic robots". Techreport No. IFI-AI-95.09, AI Lab, Department of Computer Science , University of Zurich, 1995.

[10] V. Braitenberg, *Vehicles: experiments in synthetic psychology.* Cambridge, Mass. MIT Press, 1984.

[11] R. Beer, A Dynamical Systems Perspective on Autonomous Agents, Techreport CES-92-11, Case Western Reserve University, Cleveland OH, 1992.