IN "Simulation of Animal Behaviour; From Animals to Animats"
J.A. Meyer and S. Wilson (eds.). MIT Press.

THE DYNAMICS OF COLLECTIVE SORTING ROBOT - LIKE ANTS AND ANT - LIKE ROBOTS

J.L. Deneubourg[1], S. Goss[1], , N. Franks[2], A. Sendova-Franks[2], C. Detrain[1] and L. Chrétien[1]

[1] Unit of Behavioural Ecology, CP 231
Université Libre de Bruxelles, 1050 Bruxelles

[2] School of Biological Sciences
University of Bath, Bath BA2 7AY

Abstract

A distributed sorting algorithm, inspired by how ant colonies sort their brood, is presented for use by robot teams. The robots move randomly, do not communicate, have no hierarchical organisation, have no global representation, can only perceive objects just in front of them, but can distinguish between objects of two or more types with a certain degree of error. The probability that they pick up or put down an object is modulated as a function of how many of the same objects they have met in the recent past. This generates a positive feed-back that is sufficient to coordinate the robots' activity, resulting in their sorting the objects into common clusters. While less efficient than a hierarchically controlled sorting, this decentralised organisation offers the advantages of simplicity, flexibility and robustness.

1. Introduction

What is the common point between a shopkeeper and an ant colony? Each of these organisms is able to sort similar but different objects. When one examines an ant nest it is clear that neither the workers, the brood nor the food are randomly distributed. For example the eggs are arranged in a pile next to a pile of larvae and a further pile of cocoons, or else the three categories are placed in entirely different parts of the nest. The same is true in a shop. There is, however, an essential difference. The shopkeeper decides where he is going to put his different goods, and if he has assistants he tells them where to place what. Ants work in parallel but do not, as far as we can tell, have the capacity to communicate like the shopkeeper, nor do they have a hierarchical organisation whereby one individual makes the necessary decisions and the others follow. Nevertheless, if you tip the contents of a nest out onto a surface, very rapidly the workers will gather the brood into a place of shelter and then sort it into different piles as before.

This article describes a simple behavioural algorithm, to be followed by each worker, that generates a sorting process. Sorting is achieved without requiring either external heterogeneities (e.g. temperature or humidity), hierarchical decision-making, communication between the individuals or any global representation of the environment. We also stress that the ants/robots have only very local information about the environment and a very short-term memory, and furthermore move randomly, no oriented movement being necessary. They can't see far off nor move directly towards objects or piles of objects.

Our aim in this article is not to prove that the model proposed is actually how the ants behave, but to show that such an algorithm both works and could be used by a team of robots. Inspired from our knowldege of the importance of functional self-organisation or distributed intelligence in ant colonies (Deneubourg, 1977; Deneubourg et al., 1984, 1986, 1987; Deneubourg and Goss, 1989; Goss et al., 1990; Aron et al., 1990), our idea presents a working illustration of how such a distributed system can have practical applications in robotics, in accordance with ideas developed by ourselves (e.g. Deneubourg et al., 1984, 1990; Deneubourg and Goss, 1989), and others (e g. Beni, 1988; Brooks and Flynn, 1989; Sandini and Dario, 1989; Fukuda and Kawauchi, 1989; Brooks et al., 1990; Steels, 1990). The environment for a real-time multi-robotic demonstration of this algorithm (and others related to ant-like foraging behaviour) is under preparation at MIT, directed by Maja Mataric and Rodney Brooks.

Firstly we describe a Monte-Carlo simulation of this collective behaviour, secondly we present the continuous mathematical model and its steady state analysis.

2. Monte Carlo Model

The model is based on the following principle. The ALRs or RLAs (ant-like robots or robot-like ants) move only randomly. When they come across an object the probability of picking it up is all the greater the more the object is isolated, i.e. the less the number of similar objects there are in the immediate neighbourhood. When carrying the object an ALR's probability of putting it down is all the greater as there are more of the same in the immediate neighbourhood. Either of these two rules is sufficient to form separate clusters of the two object types, but both rules together act much faster.

Isolated objects are picked up. When a small, albeit loose cluster appears by chance, it "encourages" passing carriers to add their load to the cluster thus increasing its "attractivity". With this positive feed-back mechanism the clusters grow, "absorbing" isolated objects and the smaller clusters through the action of the carriers. As clusters of each object type "attract" and fill up the nearby space with essentially the same type, this effectively isolates any object of another type in the immediate vicinity, thus making it more likely to be picked up by an ALR. Sorting is the consequence of this clustering and crowding out behaviour.

The environment is a square network of points. At time zero a number of ALRs and objects of type A and B are placed at random in the network, only one object and/or one ALR being allowed at each point. At each time step, and in random order, the ALRs move randomly north, south, east or west, although they cannot move into the wall around the space or into a point already occupied by an ALR. When an ALR moves onto a point containing an object it decides whether or not to pick it up. The less objects of the same sort there are in the immediate enviroment, the greater the probability that it will pick it up, as given by the following function:

$$p\,(\text{pick up}) = (k^+ / (k^+ + f)\,)^2$$

where f is an estimation of the fraction of nearby points occupied by objects of the same type, and k^+ is a constant. The probability thus decreases with f, from 1 (when f=0), to 1/4 (when $f = k^+$), and less as f tends to 1.

An ant could (probably) estimate f by the strength of an odour associated with each brood type or else by tactile investigation, sight being less important as such sorting is usually done underground. A robot would need a rather sophisticated visual, chemical or other sensory system to do the same, and so we propose the following sampling-based estimation which, while less precise, has the advantage of being much simpler and more easily implemented.

Each ALR has a short-term memory of m steps, that records what it met in each of the last m time steps. Thus at t=10, a memory of length 10 could hold the string 00AB0AA0B0, indicating that during the previous ten time steps the robot met 3 objects of type A and 2 of type B, the other points having been empty. f_A would be equal to 3/10, and f_B to 2/10.

As the robot walks randomly, this sampling provides a rough estimation of the density of the two sorts of objects in the immediate neighbourhood. Note that similar sampling techniques are known or suspected to exist in ants (e.g. Lumsden and Hölldobler, 1983) and could possibly be at work in the way they decide to pick up or put down a larva.

As the memory is ten steps long, at the eleventh step the robot would forget what it met at the first step and add what it met at the eleventh step. In this example the string could become 0AB0AA0B0A if it next encountered a type A object.

Whatever its decision, the ALR then carries on its random walk. If it has picked up an object, then at each step that it finds itself in an unoccupied point it decides whether or not to put the object down. The more objects of the same sort there are in the immediate environment, the greater the probability that it will do so, as given by the following function:

$$p\,(\text{put down}) = (f / (k^- + f)\,)^2$$

where f is as before, and k^- is a constant. The probability thus increases with f, from 0 (when f=0), to 1/4 ($f = k^-$), and more as f tends to 1.

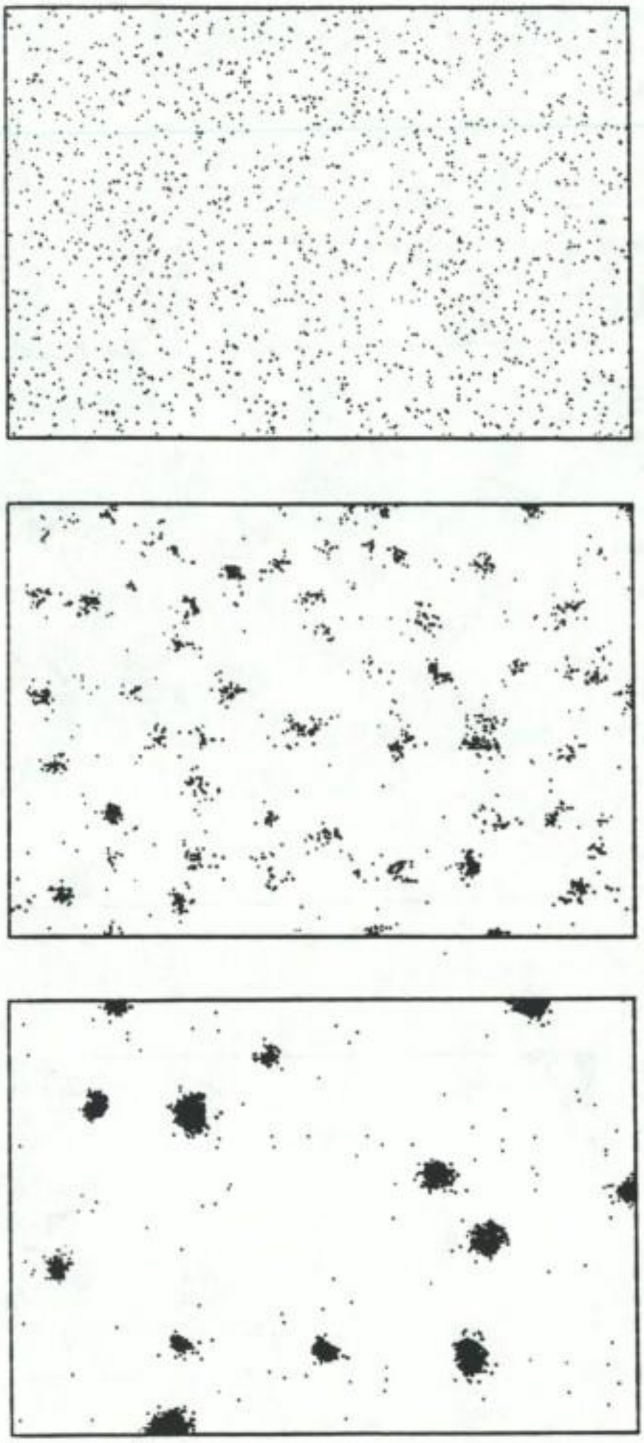

Fig. 1. Clustering after 1, 100000 and 2000000 steps. 100 ALRs, 1500 objects, $k^+=0.1$, $k^-=0.3$, m=50, e=0, space=290x200 points. Small evenly spaced clusters rapidly form, and later merge into fewer larger clusters.

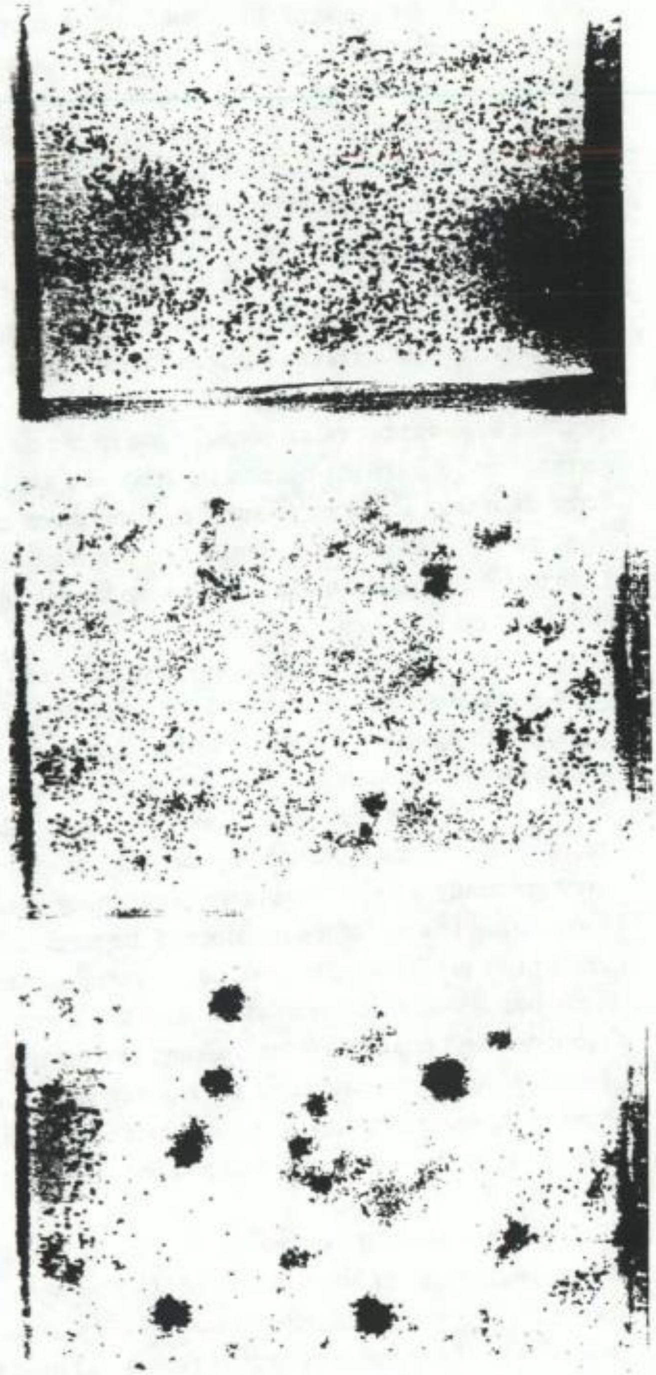

Fig. 2. Clustering in a colony of *Pheidole pallidula*. 4000 corpses were placed on a 50x50cm arena, and photos taken at time 0, 20 and 68 hrs. Small evenly spaced clusters rapidly form, and later merge into fewer larger clusters.

3. Monte Carlo Simulations and Comparable Experiments With Ants

Fig. 1 shows how randomly distributed objects of one type are rapidly gathered into small and regularly spaced clusters, which over a longer period of time gradually merge into a smaller number of larger clusters. The clusters are constantly having elements removed and added, and therefore tend to drift about slowly. When two clusters meet they fuse.

Fig. 2 shows a similar process in a colony of *Pheidole pallidula*. When ants die, workers carry the corpse out of the nest, and in laboratory conditions place them in a pile, a behaviour common to many ant species. In this experiment, a large number of ant corpses were spread out on an arena. Very quickly the workers (or robot-like ants) gathered them into a number of small clusters, which after a long period of time merge into one or two large clusters (the experiment shown in fig. 2 did not run long enough for this last stage to be seen).

Fig. 3 shows how randomly distributed objects of type A and B are rapidly sorted into small clusters of each type, which again over a longer period of time gradually merge into a smaller number of larger clusters of each type.

The parameter values used have been selected more or less arbitrarily, helped by the fact that the model sorts efficiently within a wide range of parameter values. A more formal analysis of the influence of the parameter values will be performed elsewhere on the continuous model described below. It is nevertheless clear that a very long memory length prevents effective sorting or clustering as it gives the ALRs the equivalent of a perceptive radius as large as the space they operate in, and so they could not distinguish between an isolated object and one in a local cluster.

An interesting variant of the model introduces some overlapping, or imperfection of discrimination, between the two sorts of objects, in the following sense. An unloaded ALR has met an object of type A and must thus calculate f for A, or f_A, being the number of objects A met in the last m steps divided by m. However one can introduce some confusion between A and B by calculating f_A as the number of objects of type A plus a fraction, e (the error rate), of objects of type B, the sum being divided by m. In the example above, 00AB0AA0B0, f_A would become (3 + 2e)/10. This is equivalent to the ALRs making a certain

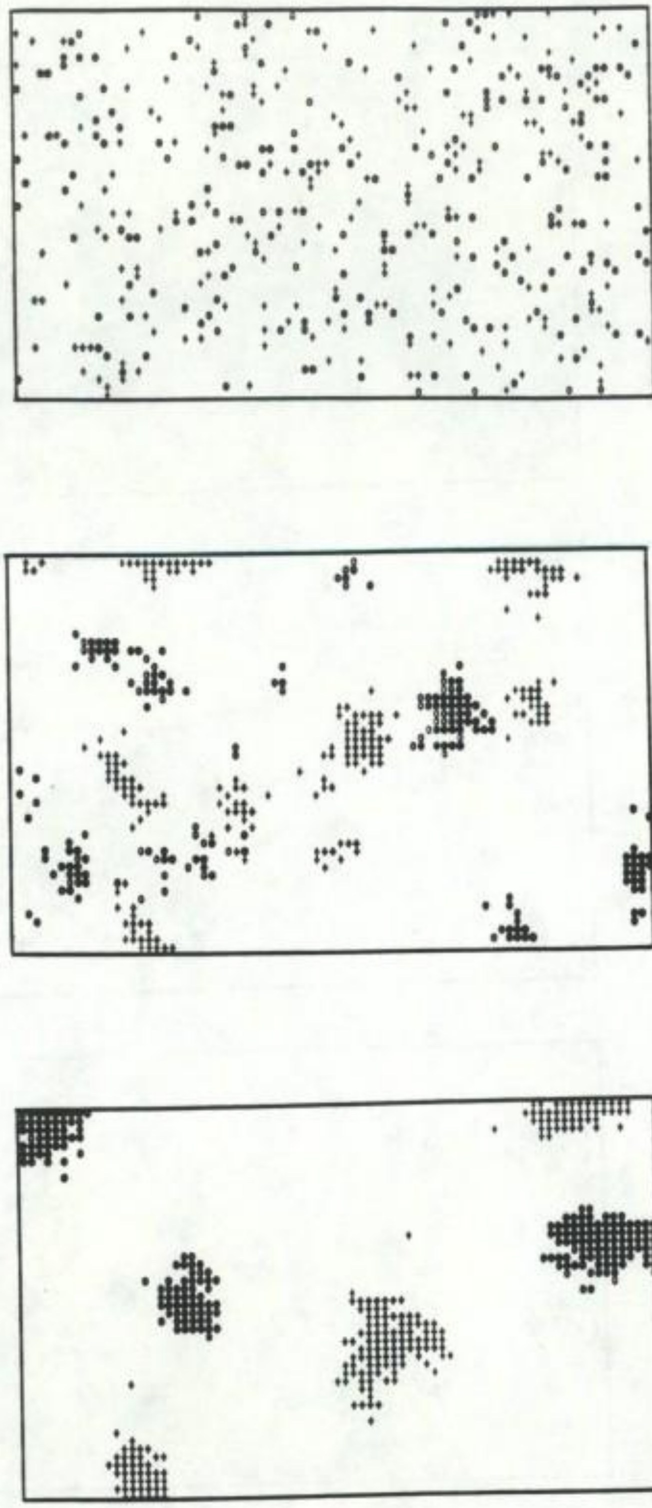

Fig. 3. Sorting after 1, 60000 and 570000 steps. 20 ALRs, 200 o and 200 + objects, k^+=0.1, k^-=0.3, m=15,e=0, space=80x49 points. Small evenly spaced clusters of each object type rapidly form, and later merge into fewer larger clusters with a high degree of sorting.

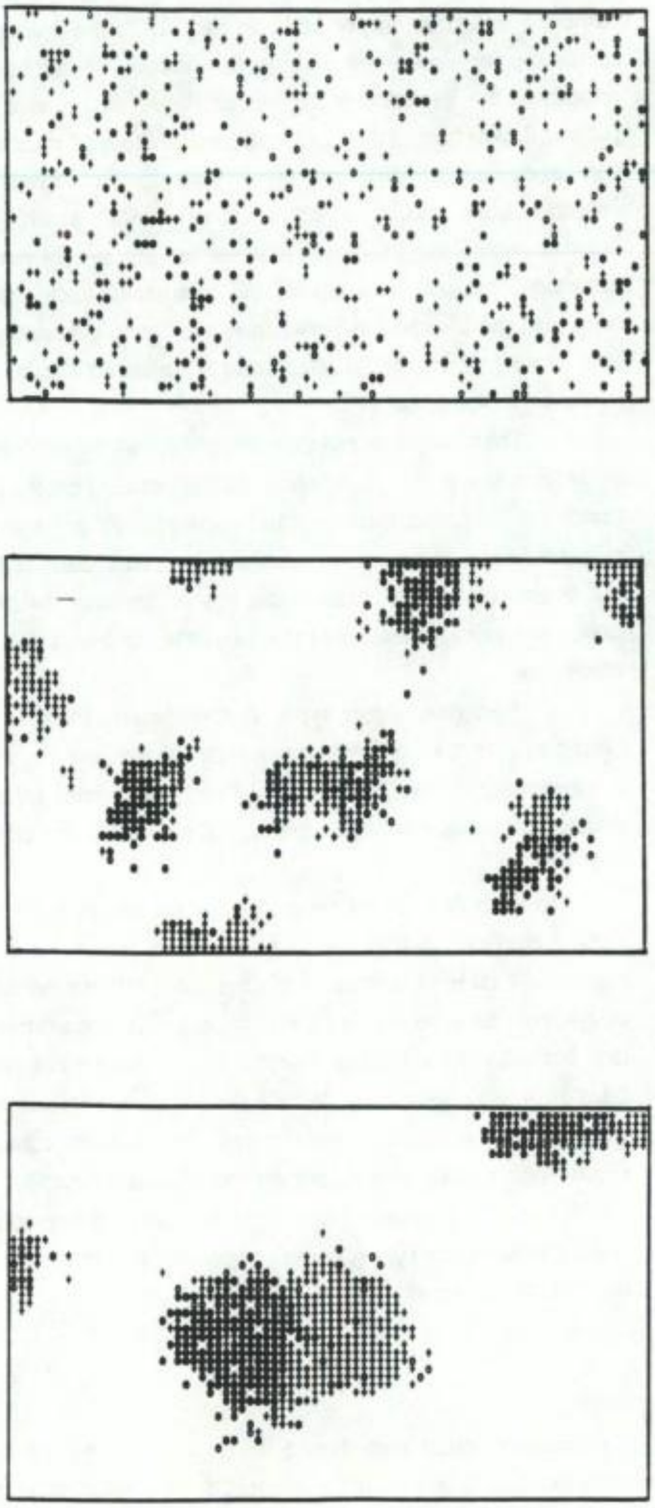

Fig. 4. Sorting after 1, 225000 and 1660000 steps. 20 ALRs, 300 o and 300 + objects, k^{+}=0.1, k^{-}=0.3, m=15, e=0.2, space=80x49 points. Small evenly spaced clusters, containing both types of object but placed adjacently, rapidly form, and later merge into fewer larger clusters, with a high degree of sorting.

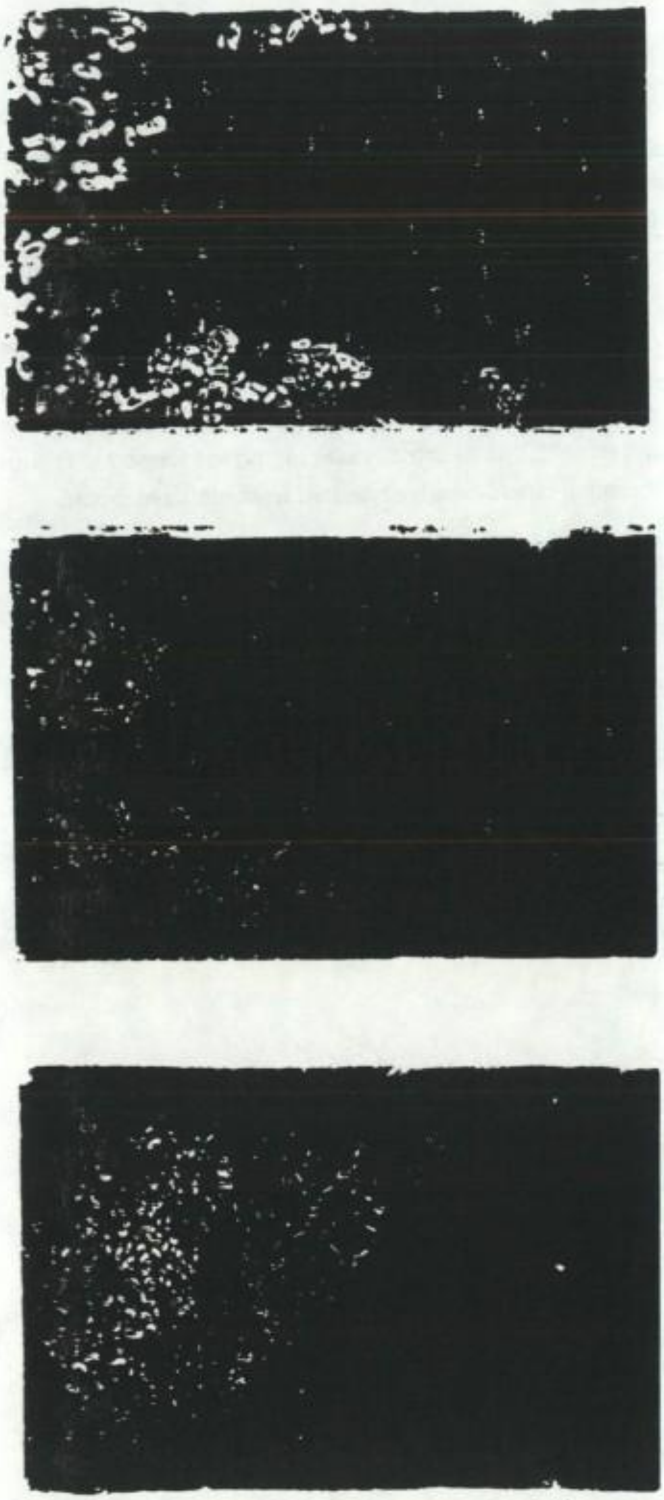

Fig. 5. Sorting 1h, 1h40m and 72h after a nest of *Leptothorax unifasciatus* has been mixed up. The larvae are rapidly gathered into a number of small clusters, and later merge into one large cluster with the small larvae (centre left) clearly separated from the large larvae (centre and top and bottom left).

number of mistakes in their identification of the two types of object.

As on might expect, with a large error rate ($e>0.3$), the objects are clustered indiscriminately and not sorted. With a small error rate ($e<0.1$), the sorting is more or less as efficient as with no error (fig.3). With an intermediate error rate, the objects are efficiently sorted, but into overlapping clusters whose dispersion is thus reduced (see fig. 4).

Figure 5 shows a similar process in an ant colony. A small microscope-slide nest of *Leptothorax unifasciatus* is tipped out onto an area. The workers rapidly bring the larvae back into the nest (5a), placing them in small piles. At this point the sorting is not very strong. Rapidly the piles merge (5b), and eventually one big pile is formed (5c) with the small larvae clearly separated from the large larvae.

4. The Continuous Model

(1) $\delta_t O_i = b_i(1-\Sigma O_i/e)R_i - aO_iR_0/b_i$

(2) $\delta_t R_i = D\nabla^2 R_i - \delta_t O_i$

(3) $\delta_t R_0 = D\nabla^2 R_0 + \Sigma\delta_t O_i$

(4) $b_i = (b + b'\Sigma\int_{-\infty}^{\infty} g_{ij}(r,z)O_j(z)dz)^n$

with:

O_i the number of objects of type i
R_i the number of ALRs carrying an object i
R_0 the number of ALRs not carrying an object
b_i the probability of putting down an object i
a/b_i the probability of picking up an object i
e no of empty places per unit surface
g_{ij} the distance-interaction function between objects i and j
D, a, b and b' being constants, and $n>=1$.

The continuous model functions more or less exactly as the Monte Carlo model. Equation (1) states that the number of objects i added, at a given point and time, is given by the product between the number of ALRs carrying an object i, the probability of putting down an object i and the fraction of room left at that point and time. The number of objects removed is a product of the number of empty ALRs and the probability of picking up an object i.

Equations (2) and (3) state that the robots diffuse randomly, and that empty and carrying ALRs are "created" by the putting down and picking up process. Equation (4) expresses the probability of picking up (the inverse of putting down) as a distance-dependant function of the number of objects of different types both in the immediate neighbourhood and, to a lesser extent, further away. The distance-interaction function expresses an instantaneous distance-dependant measure of the number of each object type, and the interaction between the different types of object. Note that in the Monte Carlo simulations the corresponding n was equal to 2.

The complete stability analysis will be published elsewhere, but it is already clear that, depending on the parameter values, the model exhibits either only a homogeneous solution or else this homogeneous distribution is unstable and the model reaches an inhomogeneous distribution, i.e. sorting, these inhomogeneties being like a Turing instability.

With one object type, at low density there is no clustering. As the density increases, clustering appears, with a wavelength dependent on the different parameter values, notably the distance-interaction function and the size of the nest.

With two object types the system has roughly the same behaviour, with respect to the size and number of clusters. What is of interest is of course the relative position of each object type. With no error, i.e. no distance-interaction between the different object types, each cluster contains only one object class. With a small distance-interaction between the different object types, the clusters contain mostly one sort of object, but are positioned adjacently. If the density is, however, very low, the clusters appearing contain both object types, sorting appearing progressively as the density increases.

5. Conclusions

Whereas we stated that the ALRs (and the RLAs) do not communicate, one might be tempted to say that they communicate in a very indirect manner, via their effect on the environment. In other words each ALR is influenced by its own and the other ALRs' past actions in moving objects. By this means they give the appearance of coordinating their activity to form clusters, in the sense that because many different ALRs add objects of type A to a particular cluster of type A, they appear to have agreed to form a cluster of type A at this point, whereas we know that no such consensus exists.

By the same logic it is evident that one solitary ALR can also form clusters and sort by interacting with its own past actions, although it would act for example ten times as slowly as a group of ten ALRs.

Another example of how such indirect "communication" can organise the activity of a group of non-communicating agents can be seen in the way the foragers of some species of ants set up individual and non-overlapping foraging territories (Deneubourg et al., 1987). Individual foragers learn progressively to return to the area where they find food. When an individual becomes specialised to an area then by its activity it reduces the amount of food there. Other individuals passing through that zone will have less chance of finding food and so will be less encouraged to return to that zone by their own learning process. The same principle can also be used to allocate different tasks dynamically among the members of a group (see also Theraulaz et al., this volume).

Returning to the sorting context, it is clear that a hierarchical system, wherein either a human supervisor or an alpha-robot decides exactly where to put which type of object, would sort more efficiently. However such a robot would no longer be ant-like, and would require a capacity for the analysis of how many types of object there are in the environment and a means of communicating its decision to other robots. All the robots would require at least a rudimentary map of the environment to transport the objects they find to the pre-arranged locations, or else would need to home towards a beacon placed by some means at each location. Furthermore, any fluctuation in the environment could make the original decision inappropriate.

The ALR system, while less efficient, requires no supervision and is capable of operating in a wide range of environments without specific programming, and with a large number of object types. The ALRs are only capable of perceiving an object at the point they occupy, and have no long range perception either for objects, piles of objects or homing beacons.

This simplicity makes the ALRs cheaper and more robust, and the lack of any hierarchy prevents fatal breakdowns, no one unit being essential. Even if a number of ALRs broke down this would only slow and not prevent the sorting. Again even if the ALRs frequently mistake one object for another, sorting is unimpaired. Indeed, rather than simply tolerating a certain degree of error, it can even be desirable to deliberately add error where none or little exist. We have seen how a small error in discriminating objects can lead to the ALRs placing the pile or piles of the two sorts of objects adjacently. Another example in the social insect world shows how a certain degree of error in following nestmates' trails to food sources allows the colony to exploit spatial heterogeneities in the food distribution more efficiently (Deneubourg et al., 1984). Put simply, if they followed the trails too exactly they would never find new sources nearby the one they were guided to. Error can thus be more creative than inefficient, and room should be allowed for it (see also the different works of Charles Darwin).

Overall, the system's simplicity, flexibility, error tolerance and reliability largely compensate for their lower efficiency. This is a general characteristic of systems in which the collective behaviour of a group of autonomous agents is emergent rather than explicitly programmed, and is surely one of the reasons for the 100 million year long evolutionary triumph of social insects. There are many circumstances in which a robot team could be profitably organised in a similar fashion.

Acknowledgements

This work was supported in part by the Belgian Program for Inter-University Attraction Poles, the Belgian Fonds National de Recherche Scientifique, NATO Scientific Affairs Division Grant 034/88, the Venture Research Unit of British Petroleum and the Ernst och Victor Hasselblads Foundation. We would like to thank Stefano Focardi for his helpful scepticism.

References

Aron S., Deneubourg J.L., Goss S. and Pasteels J.M. (1990) Functional self-organisation illustrated by inter-nest traffic in ants: the case of the Argentine ant. In: W. Alt and G. Hoffman (eds.) Biological Motion, Lecture Notes in Biomathematics, Springer Verlag, Heidelberg.

Beni G. (1988) The concept of cellular robotic systems. IEEE International Symposium on Intelligent Control, Arlington, VA, August 24-26.

Brooks R. and Flynn A. (1989) Robot beings. IEEE International Workshop on Intelligent Robots and Systems (IROS '90), Tokyo, Japan, Sept 4-6.

Brooks R.A., Maes P., Mataric M.J. and More G. (1990) Lunar base construction robots. Proc. Japan-USA symposium on Flexible Automation (in press).

Deneubourg J.L. (1977) Application de l'ordre par fluctuations à la description de certaines étapes de construction du nid chez les termites. Insectes Sociaux 24: 117-130.

Deneubourg J.L., Pasteels J.M and Verhaeghe J.-C. (1984) Quand l'erreur alimente l'imagination d'une société: le cas des fourmis. Nouvelles de la Science et des Technologies 2: 47-52.

Deneubourg J.L., Aron S., Goss S., Pasteels J.M and Duerinck G. (1986) Random behaviour, amplification processes and number of participants: how they contribute to the foraging properties of ants. In: Los Alamos Conference on Evolution, Games and Learning, Physica 22D: 176-186.

Deneubourg J.L., Fresneau D., Goss S., Lachaud J.-P. and Pasteels J.M (1987) Self-organisation mechanisms in ant societies (II): learning during foraging and division of labor. In: J.M. Pasteels and J.L. Deneubourg (eds.) From Individual Characteristics to Collective Organisation in Social Insects. Experientia Supplementum 54, 177-196, Birkhaüser, Bâle.

Deneubourg J.L. and Goss S. (1989) Collective patterns and decision making. Ecol. Ethol. Evol. 1: 295-311.

Deneubourg J.L., Goss S., Sandini G., Ferrari F. and Dario P. (1990) Self-organizing collection and transport of objects in unpredictable environments. Proc. Japan-USA symposium on Flexible Automation (in press).

Fukuda T. and Kawauchi Y. (1989) Cellular Robotics, Construction of Complicated Systems from Simple Functions. In: P. Dario, G. Sandini and P. Aebisher (eds.) Robots and Biological Systems. Proc. NATO ARW, Il Ciocco, Italy.

Goss S., Deneubourg J.L., Aron S., Beckers R. and Pasteels J.M. (1990) How trail laying and trail following can solve foraging problems for ant colonies. In : R.N. Hughes (ed.) Behavioural mechanisms of food selection: 661-678. NATO ASI Series, Vol G 20, Springer Verlag, Heidelberg.

Lumsden C.J. and Hölldobler B. Ritualized combat and intercolony communication in ants. J. theor. Biol. 100: 81-98.

Sandini G. and Dario P. (1989) Sensing strategies in cellular robotic systems. In: T. Kanade, F. Groen and L. Hertzberger (eds.) Proc. IAS 2, 937-942.

Steels L. Cooperation between distributed agents through self-organisation. In: Y. Demazeau and J.-P. Müller (eds.) Decentralized AI, 175-196, Elsevier, Holland.