Diversity and Adaptation in Populations of Clustering Ants

Erik D. Lumer[1,2,*] and Baldo Faieta[2]
[1] *Center for Nonlinear Phenomena and Complex Systems*
Université Libre de Bruxelles, C.P. 231
Blvd. du Triomphe, B-1050 Brussels, Belgium
and
[2] *Zetes Electronics*
Da Vinci Science Park, 3 rue de Strasbourg, B-4, 1130 Brussels, Belgium

Abstract

In this paper, we introduce a new method for structuring complex data sets into clusters. The method relies on local actions in a population of simple, ant-like processes, so that global data structures emerge in a self-organized fashion. The model provides a setting to explore the role of diversity and local adaptive behavior in shaping the collective *phenotype* of the population.

1 Background

Population thinking in biology dates back at least to Darwin's seminal work on the principles of natural selection [1]. More recently, the study of insect societies or the immune system, to name only two examples in the biological realms, have focused on the qualitatively distinct properties which can arise at the collective level as a result of local processes. The possible functional relevance of these global states and their constraining coupling to the local processes from which they emerge has prompted the development of new theories and techniques to elucidate the micro-macro links.

In particular, the advent of powerful computers over the past two decades has offered an experimental setup in which to explore some of the basic population effects exhibited in biological systems. At the same time, natural systems have begun to provide powerful insights toward the design of distributed forms of computation [2, 3]. The basic idea is to search for simple behavioral rules at the individual level which translate into an overall population "phenotype" that embodies the result of some computation. Whenever such rules are found, they open up the possibility to exploit the intrinsic parallelism in collections of locally interacting processes that perform in the absence of centralized controls. Several classical problem-solving and optimization tasks have recently been dealt with in such a fashion, from chryptarithmetic problems [4], to the sorting of objects in segregated piles [2], to the optimal partitioning of graphs [5], among others.

The work described in this paper explores several issues of general relevance to collective behavior and self-organization in populations, in particular the functional role and ecological expression of population diversity and individual behavioral adaptation. The study is carried on in the specific context of a novel approach to exploratory data analysis, akin to cluster analysis and multidimensional scaling, which provides a clear set of performance metrics against which different populations of clustering ant-like processes can be compared. In the next subsection, we review an earlier study of ant-like colonies by Deneubourg and colleagues [2] which inspired the present model. We then briefly introduce standard approaches to structuring multidimensional data sets, as well as some more recent development in this area, prompted by the need to support explorations in complex information spaces such as large textual databases.

This should set the stage for the present work. A more complete description of the task domain, along with the outgrowth of a practical clustering system, is beyond the scope of this paper, and is being reported elsewhere [6]. Instead, in the following sections, we focus on the self-organized properties in populations of simple autonomous processes. Section 2 describes the individual processes in our system, the basic rules governing their behavior, and the environment in which they evolve. In section 3, we introduce an element of diversity in the population, and endow the individual ant-like agents[1] with memory and context-dependent behavior. We then present the results from stochastic simulations of the model. Measures designed to track the dynamics of the self-organized clustering are applied to populations with varying de grees of diversity and individual complexity. Finally, we close with a discussion of our results and in-

*Present address: The Neurosciences Institute, 3377 North Torrey Pines Court, La Jolla, CA 92037

[1] We use the terms agent, process, and ant interchangeably. The latter is used for convenience only; it is not meant to imply any biological realism in our model agents.

dicate future developments.

1.1 Collective Sorting in Ant-Like Colonies

In section 2, we will introduce a distributed clustering method for multidimensional data sets inspired by a recent study of the manner in which ant colonies sort their brood [2]. In the model of Deneubourg and coworkers, lightweight ant-like agents move at random on a 2-d grid on which objects have been scattered. The agents do not communicate with each other and can only perceive their surrounding local environment. When they bump into an object, the probability that they pick it up decreases with the density of and similarity with other objects in the vicinity. Likewise, the probability of dropping an object carried by an agent is an increasing function of a simple similarity measure within a local region. This simple behavior, combined with a positive feedback from the environment which is reshaped by the the ongoing collective action, translates into the organization of initially dispersed objects into stacks of identical elements.

Deneubourg and coworkers restricted their studies to environments made of either identical objects or two distinct types of objects. Thus, the collective "phenotype" of their ant colonies was one of sorting, a fairly trivial computational task to carry on a conventional computer, although quite remarkably performed by social insects in the absence of central controls. Nevertheless, as we will show in section 3, their algorithm can be generalized to objects which differ along a continuous similarity measure, leading to a novel clustering method for multidimensional data sets with quite interesting properties. Before we turn to this algorithm, we briefly recall what cluster analysis and multidimensional scaling techniques are about.

1.2 Structuring Data and Explorations in Complex Information Space

Clustering and scaling techniques have been widely used in a variety of domains, as a way to probe underlying structures in complex data sets [7]. In particular, these techniques have seen a renewed interest in the context of information access. Indeed, in complex information spaces, such as text databases, it is important to provide means of representing the overall structure of the data sets, so as to support effective explorations of their content [8, 9]. Furthermore, some effort has been put in developing methods which reorganize data sets in real time, so as to allow ongoing interactions between an information system and its user.

The problem that cluster analysis addresses is the following one [7]: given a set of elements, and a similarity measure between pairs of elements, find an algorithm for grouping elements in clusters, so that similar elements end up in the same cluster. In general, each datum may be represented as a point in some high-dimensional space, and the number of clusters is not known a priori. Clearly, this problem does not have a best solution, and many statistical techniques have been proposed.

There are two major families of clustering methods: the hierarchical ones, in which clusters are formed by a process of agglomeration or division, and the partitioning methods, such as the k-means algorithm in which elements are allowed to move between clusters at different stages of the analysis, so as to join the cluster with the nearest centroid. In the partitioning methods, this process continues iteratively until convergence is reached with a predefined number of clusters. Hierarchical methods are computationally expensive, but analyze the data at different levels of granularity, while the faster partitioning methods establish clusters devoid of internal structure.

Multidimensional scaling represents a different class of numerical techniques which extract some global structure out of data sets. These methods are designed to construct a map displaying a faithful representation of the relationships between a collection of elements, given only a matrix of pairwise distances. Typically, the map will be in two or three dimensions, so as to allow its visualization. In addition to well established multidimensional scaling methods [10], recent stochastic techniques, such as simulated annealing [11], can be used in building global maps which minimize some "stress" function, although their computational cost is still very expensive.

As this brief survey indicates, effective exploratory data analysis imposes a trade-off between, one the one hand, responsiveness, and on the other, the complexity of representations constructed by structuring methods. In this context, the intrinsic parallelism in the self-organizing process of collective sorting motivates the search for similar paradigms to structure more complex data sets. In the next section, we develop such a process.

2 Self-Organized Clustering

We model the basic ants and their environment as follows. Ants perform a random walk on a 2d-grid, on which elements have been laid out at random, so that a site in the grid is occupied by at most one element. In fact, the dimensions of the grid are such that its number of sites exceeds the number of elements by roughly an order of magnitude. Furthermore, the number of elements exceeds the number of ants by at least another order of magnitude, so as to translate ants actions over a short period of time into small fluctuations of the environment.

The simulation evolves in discrete time steps. At each step, an ant is selected at random and can either pick or drop an element at its current location, given that there is either an element at that location or that the ant is carrying one, respectively. Assuming that an unloaded ant comes across an element, the probability of picking that element increases with low density and decreases with the similarity of the element vis-a-vis the other elements in a small surrounding area. (In what follows, we define this area as a square of $d \times d$ nodes) . Accordingly, the probability of picking an element, say i, is defined as

$$P_{pick}(i) = (\frac{k_p}{k_p + f(i)})^2 \tag{1}$$

where k_p is a constant and $f(i)$ a local estimation of the density of elements and their similarity to i.

By the same token, the probability that an ant-like process will drop a carried element should increase with the density of similar elements in its surrounding area. A simple functional relation which satisfies this tendency, and the one being used here, is given by

$$P_{drop}(i) = \begin{cases} 2f(i) & \text{if } f(i) < k_d \\ 1 & \text{otherwise} \end{cases} \tag{2}$$

where k_d is a constant. In the probabilities for manipulating elements expressed by P_{pick} and P_{drop}, the density dependent function *f(i)* for an element i, at a particular grid location, is defined as

$$f(i) = \begin{cases} \frac{1}{d^2} \sum_j (1 - d(i,j)/\alpha) & \text{if } f > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In this expression, the sum extends over all the elements in the local area surrounding element i, and $d(i,j)$ measures the dissimilarity between the pair of elements (i,j). The constant α scales the dissimilarities. For the sake of concreteness, let us assume here that the elements can be represented as points in an n-dimensional space, so that $d(i,j)$ is simply the euclidian distance between i and j. The normalizing term d^2 equals the total number of sites in the local area of interest, and introduces a density dependency in $f(i)$. As a result, the maximum of f is reached if and only if all the sites in the neighborhood are occupied by identical elements (i.e. $d(i,j) = 0$), in which case $f = 1$. Whenever a loaded ant decides to drop its element, it looks for the first empty site in its vicinity in which to do so. A time step finishes with the selected ant moving to one of its four adjacent nodes, each direction of motion being equally likely.

In summary, the simulation consists in cycling through a loop involving: 1) the selection of an ant, 2) if appropriate, the manipulation of a local object by that ant, and 3) a random unitary displacement on the grid. We contend that the collection of ants so-defined performs a heuristic mapping of the data set onto the 2-d grid. This mapping amounts to a dynamic organization of data in a fashion halfway between a cluster analysis - in so far as elements belonging to different concentration areas in their n-dimensional space end up in different clusters - and a multidimensional scaling, in which an intracluster structure is constructed. Notice, however, that the relative positioning of clusters on the grid is arbitrary[2].

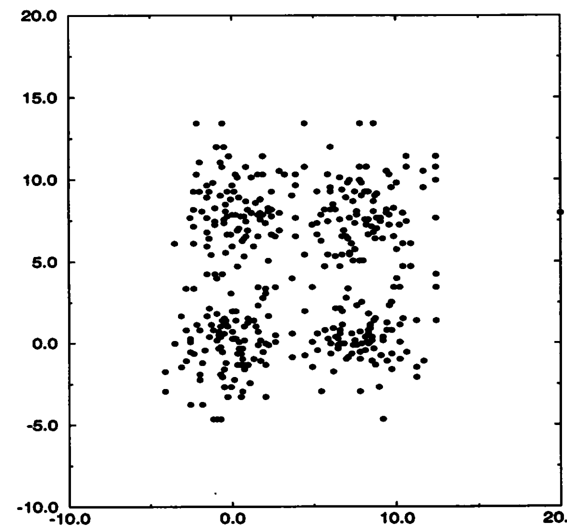

Figure 1: Data Set composed of 400 points sampled with equal probabilities from 4 gaussian distributions. The distributions are defined according to: $(N(0,2), N(0,2))$, $(N(0,2), N(8,2))$, $(N(8,2), N(0,2))$, and $(N(8,2), N(8,2))$.

This process is illustrated in the following example. We generated a 2d data set composed of 400 points sampled with equal probability from one out of four bivariate gaussian distributions. The distributions have identical variances but shifted means. The actual scatter plot of the sampled points in their 2d space is shown in figure 1 and the properties of the distributions are indicated in the accompanying caption. The data points were laid out at random locations on a 52x52 grid, populated by 40 ants. The other parameters of the simulation are indicated in the caption of figure 2a, which shows the initial distribution of data points on the grid. Notice that each element in this figure is labeled according to its originating distribution for illustrative purpose only. This information is not available to the ants. Figure 2b shows the layout of elements on the grid after 50 cycles of the simulation, each cycle being composed of 10,000 ant selections. As can be seen, the data points are clustered into patches , each one made of points issued mostly from a single statistical distribution. However, the number of patches exceeds and thus poorly reflects the number of distributions underlying the data set.

In the next section, we discuss ways in which the colony can be modified to fix this discrepancy. We also investigate how the speed at which stable structures are constructed depends on the properties of individual members of the colony. To close this section, let us emphasize that all the results presented below have

[2]By relaxing the global positioning constraints, we are able to devise a faster algorithm. Also, in the context of textual databases, the relative positioning of coarse clusters is likely to be meaningless.

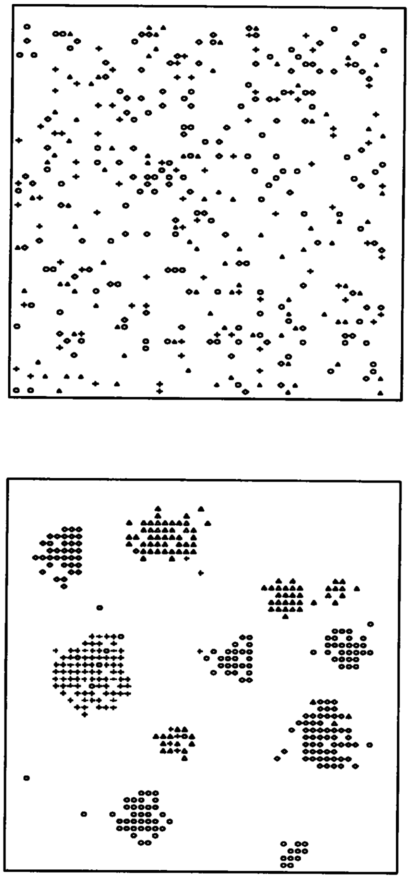

Figure 2: Layouts of elements on a 52x52 grid. Elements sampled from different distributions are represented by different symbols. a (top): Initial layout. b (bottom): Layout after 50 cycles using a population of basic clustering agents.

been verified to hold qualitatively for a variety of data sets, including those embedded in higher dimensions. For instance, we have applied successfully our algorithm to synthetic data sets embedded in an 8-dimensional space. Performance seems rather independent of the size of the data set provided that the following criterion is met: sampling size must be sufficiently large to produce locally dense sets of similar points, that is to say the average number of points differing from a given one by less than the constant α must be of the order of the sensing surface d^2 of an agent moving on its grid.

3 Population Diversity and Local Adaptive behavior

In this section, we shall introduce and compare three variants of the basic population of clustering ants considered above. First, we substitute to the homogeneous population one made of individuals possessing different displacement speeds and sensitivities to object similarities. Also, we endow the ants with an evanescent memory for locations where they recently succeeded in dropping an element. Finally, we consider the impact of individual behavioral changes at the collective phenotipic level.

Before we turn to this study, let us introduce a number of dynamic measures to track the unfolding of the clustering process. We compare various runs by means of three measures. First, we compute at each cycle the spatial entropy on the grid, and this, at different granularities. This measure informs us of the relative order on the grid, so that larger clusters will translate into lower entropies [13]. Such a measure, while correlating well with the visual feedback that an animation of the run provides, does not indicate whether similar points end up at neighboring locations on the grid in agreement with the underlying statistics of the data set. This possibility is measured by a criterion of global fit, which is simply an average over all the measures $f(i)$ for a given layout on the grid. Finally, the temporal profile of the total number of drops is recorded as a measure of the effectiveness of object manipulations performed in a colony.

3.1 Diverse vs. Homogeneous Populations

It is often argued that diversity plays a crucial role in shaping up the properties of populations. One obvious biological role of diversity is as a source of variability in selectionist systems. However, from a computational view, the issue has received attention only sporadically (e.g. [4, 12]).

Our system offers a nice domain in which to quantify the impact of diversity at a macroscopic level. Indeed, we introduce variability by virtue of a range of paces, comprised between 1 and V_{max}, the latter being defined as the quantal displacement of an ant in a single time step along a given grid axis. At the same time, the pace is coupled in an inverse manner to the pickiness of an ant: fast moving ants have a more liberal measure of similarity, while slow moving ants are very selective in their similarity criterion. This trend is expressed quantitatively in the following manner. Let v be the pace of an ant. The local density function f computed by that ant now reads as

$$f(i) \quad \begin{cases} \frac{1}{d^2}\sum_j\left(1 - \frac{d(i,j)}{\alpha+\alpha(v-1)/V_{max}}\right) & \text{if } f>0 \\ 0 & \text{otherwis} \end{cases} \qquad (4)$$

As an example, a population of ants is built by selecting for each individual a pace chosen uniformly in the range (1,6). Figure 3a shows the result of their clustering after 50 cycles. The data and initial conditions are identical to those associated with figures 2. Notice now that most of the elements are distributed across only four clusters, in a way which reflects their underlying statistical distributions. Figure 3b shows the equivalent

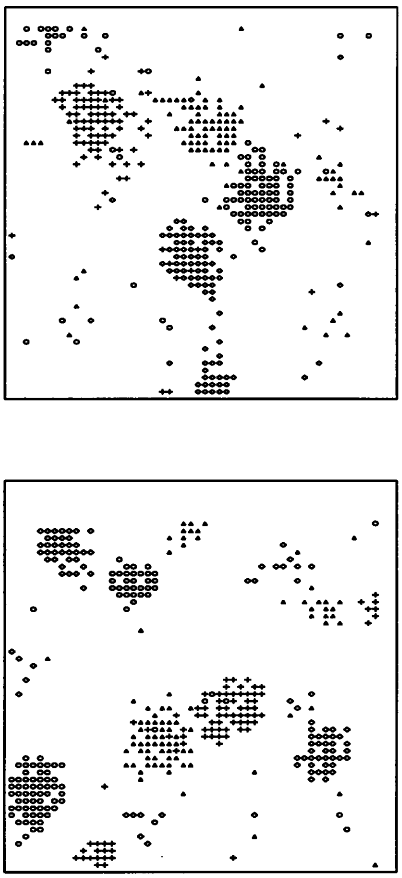

Figure 3: Layouts after 50 cycles. a (top): Diverse population of agents with paces in the range (0,6) and individual memory buffers for up to 8 items. b (bottom): Homogeneous population of ants with same memory setting as in fig. 3a.

layout using a homogeneous population of ants moving at a median pace of 3 sites/step. We observe in this case a larger number of clusters. The differences between the two populations is further characterized in figures 4a-c. In figure 4a, we compare the time course of the mean fit for the two populations. While the plot indicates a more rapid stabilization of the overall fit in the homogeneous population, the saturation takes place at a value which is inferior to the one in the diverse population. The faster convergence of elements in the homogeneous population is co nfirmed in figure 4b, where the spatial entropies are shown. Notice that the slightly lower entropy of the grid organized by homogeneous ants is an artifact of the proximity between two pairs of large, although distinct clusters. Finally, figure 4c indicates that the better clustering achieved by the diverse population involves a smaller number of manipulations.

These results are observed in a number of runs with different initial conditions and for a variety of data sets. The diverse population seems to outperform a homogeneous one for the following reason: the fast, sloppy ants rapidly seggregate elements on a coarse scale, as they tend to bring elements somewhat in the right "ballpark". On the other hand, the slow and picky ants act more locally, by sorting and placing elements with greater care.

Finaly, let us point out that the clustering process is density dependent in the n-dimensional space of the data sets. Indeed, the largest concentrations of element to appear on the grid, in fact the nucleuses for the surviving clusters, correspond to data points near the peaks of their original distribution, as such points are sampled more often.

3.2 Agents with Short-Term Memory

In runs presented in the previous subsections, the ants were endowed with a memory for the m most recent elements dropped, along with their new location. (In these runs, the memory was set to $m = 8$). When a new element is picked by an ant-like process, a comparison is made with the elements in memory, and the ant automatically goes toward the location of the memorized element most similar to the one just collected. This behavior leads to the formation of a smaller number of statistically equivalent clusters, a desirable property indeed. Figures 5a-c illustrate the impact of varying the memory size in individuals forming a diverse population. As we can see, the quality of the clustering depends strongly on the size of the memory buffers, with performance peeking for a memory size of around six items.

3.3 Behavioral Switches

Although the clustering process is stochastic, and allows for ongoing fluctuations at a micro-scale, it is basically irreversible in that the spatial entropy gradually decreases

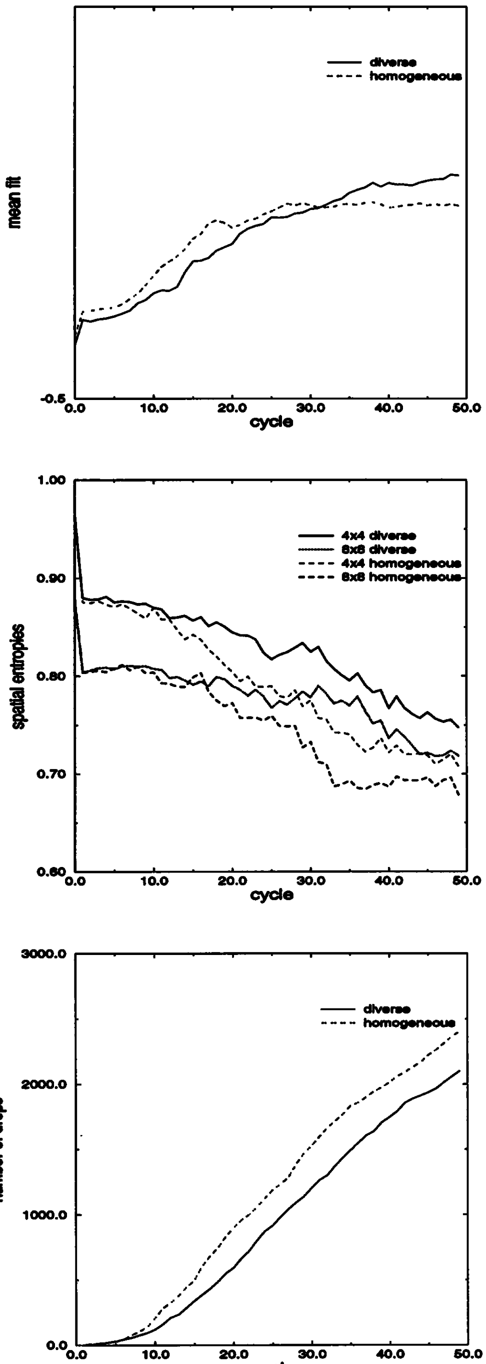

Figure 4: Analysis of performance for runs shown in figures 3. a (top): Average fit as a function of time. b (middle): Time course of spatial entropies computed over ensembles of squared areas of dimensions 4x4 and 8x8, respectively. c (bottom): Aggregate number of drops in the population as a function of time.

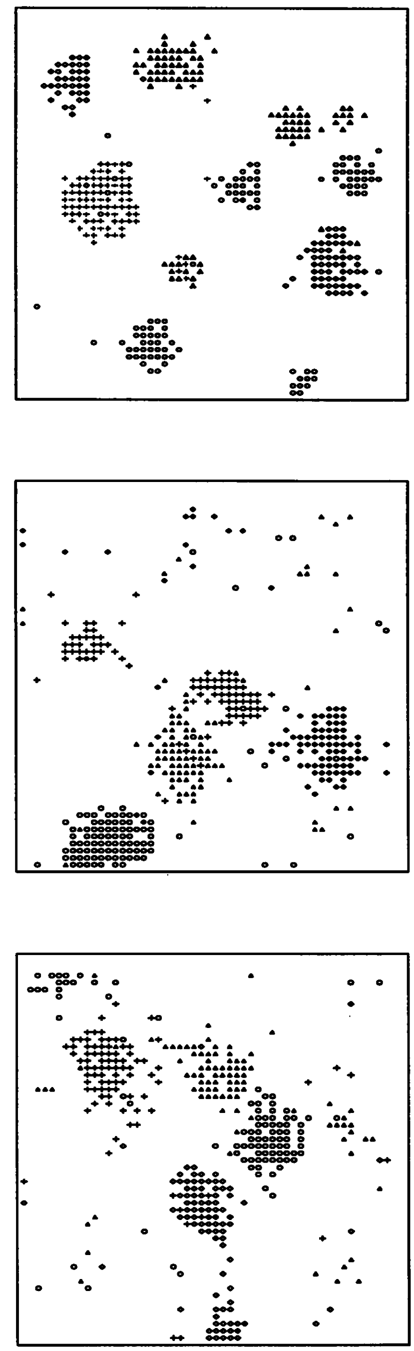

Figure 5: Global effect of individual memory sizes. a (top): m=0. b (middle): m=4. c (bottom): m=8.

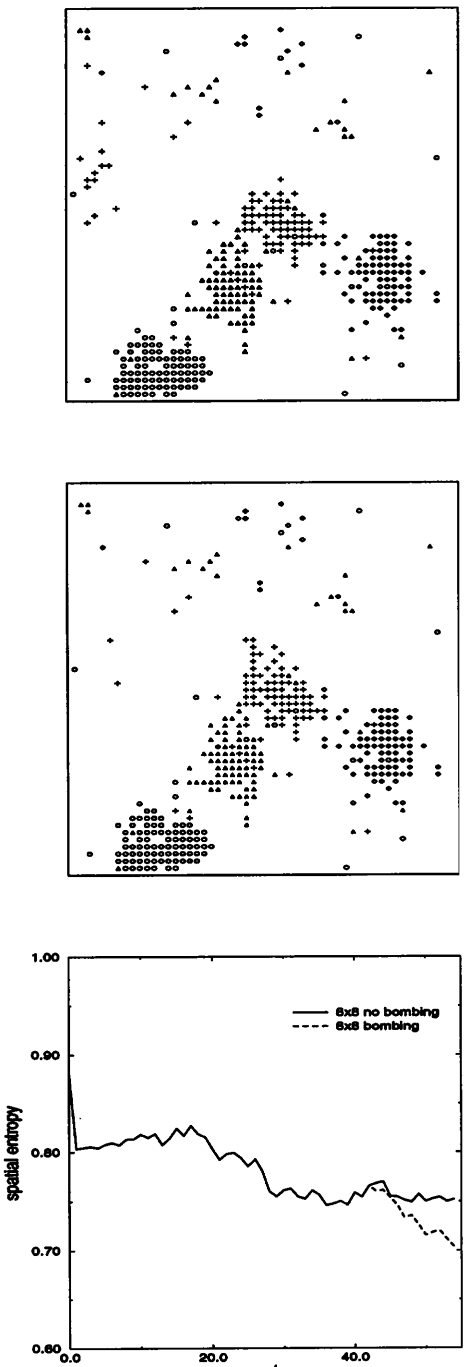

Figure 6: Self-regulated behavioral switching from gathering elements to destroying clusters. a (top): Grid shortly after an ant has begun destroying the upper left cluster. b (middle): 5 cyles later. Ants have reverted to gathering the disseminated elements. c (bottom): As a result, elements issued from the same statistical distributionend up in the same cluster.

with time. As elements get positioned in an acceptable neighborhood, they become less likely to be moved again towards another location. Accordingly, the ants will pick and drop fewer elements as time goes on. This latter property can be exploited in an adaptive behavior which introduces a sort of annealing in the system. Thus, we modify the individual agents so that each one can switch from gathering elements to destroying clusters if it hasn't manipulated an element for more than a preset number of cycles.

The ensuing collective dynamics is illustrated in figures 6a-c. The leftmost figure should be compared with figure 5b, which showed the clusters built after 50 cycles by diverse agents with a memory buffer sized to 4 items. In the present case, the grid is displayed after 55 cycles. One can see that the upper left clusters has been destroyed, and that this results in a regrouping of its elements in a single cluster, as illustrated in figure 6b after 60 cycles of the simulation. Indeed, following this action by a single agent, most of the other processes will engage in gathering the disseminated elements, thus preventing for a while any further destruction of clusters.

4 Conclusions

In this paper, we presented a simple model of clustering, which is performed in a fully decentralized fashion by a population of simple processes. This model offers a controlled setting in which to evaluate how changes in local properties translate into different performance at the system level. In particular, we observed that diverse populations of clustering agents consistently outperform homogeneous groups. Furthermore, we found that local behavioral changes, based on prior experience, can be manipulated to produce desired macroscopic effects. We have considered two kinds of adaptations, on the one hand via a memory for relevant locations, and on the other hand, in the form of behavioral switches regulated by the global feedback from the environment.

Interestingly enough, clusters which uncover global structures in a data set are constructed by local agents with no mutual interactions other than indirectly via the environment which they modify. This collective process amounts to a heuristic mapping of a possibly high-dimensional and sparse data set onto a plane, in a way which preserves neighborhood relationships as much as possible. In this plane, the emergent structure of a data set can be readily seen. Clustering is nonparametric, insofar as the number of constructed clusters is not specified a priori. This property stands in contrast with the parametric nature of most clustering algorithms. Notice also that the same collective process can be generalized to more complex environments, such as landscapes and 3d lattices [6].

These observations suggest that our mechanism could be integrated in a continuously evolving system used to visualize, organize and analyze interactively unstructured data sets. As a matter of fact, the experiments presented here have allowed us to derive a novel clustering algorithm and system, which capture this functionality. Preliminary studies applied to textual databases have been encouraging, and further results will be reported in the near future [6].

Acknowledgement

The authors would like to thank J.L. Deneubourg for many fruitful discussions about the work presented in this paper. They are also grateful to G. Nicolis at the University of Brussels and A. Wirtz at Zetes Electronics for providing a forum in which this work could be developed. This research was supported by a grant from Zetes Electronics.

References

[1] Darwin, C. (1859) "On the origin of species by means of natural selection of favoured races in the struggle for life". London: Murray

[2] J.L. Deneubourg, S. Goss, N. Franks, A. Sendova-Franks, C. Detrain, and L. Chretien (1991) "The dynamics of collective sorting robot-like ants and ant-like robots", in Proc. of the 1st Conf. on Sim. of Adaptive Behavior

[3] B.A. Huberman ed. (1988) "The Ecology of Computation", North-Holland, Amsterdam

[4] S. Clearwater, T. Hogg, and B.A. Huberman (1992) "Cooperative Problem Solving", in Computation: The Micro-Macro View, B.A. Huberman ed., World Scientific Press

[5] P. Kuntz and D. Snyers (1993), "Ant Colonies and Self-Organizing Graph", submitted paper

[6] B. Faieta and E.D. Lumer, "Exploratory data analysis via self-organization", in preparation

[7] A.K. Jain and R.C. Dubes (1988) "Algorithms for clustering data", Prentice Hall

[8] D.R. Cutting, D.R. Karger, J.O. Pederson, and J.W. Tukey (1992) "Scatter/Gather: a cluster-based approach to browsing large document collections", in Proc. SIGIR92

[9] M. Chalmers and P. Chitson (1992) "Explorations in information visualizations", in Proc. SIGIR92

[10] J.B. Kruskal (1964) "Multidimensional scaling by optimizing goodness of fit to a nommetric hypothesis", Psychonometrika 29

[11] S. Kirpatrick, C.D. Gelatt, and M.P. Vecchi (1983) "Optimization by simulated annealing" vol. 220 (4698)

[12] N.A. Nowak and K. Sigmund (1993) "Tit for tat in heterogeneous populations", Nature, vol. 355, pp. 250-252

[13] H. Gutowitz (1993) "Complexity-seeking ants", in Proc. ECAL 1993