Data Clustering and Visualization Using Cellular Automata Ants

Andrew Vande Moere, Justin J. Clayden, and Andy Dong

Key Centre of Design Computing and Cognition
The University of Sydney, Australia
{andrew, justin, adong}@arch.usyd.edu.au

Abstract. This paper presents two novel features of an emergent data visualization method coined "cellular ants": unsupervised data class labeling and shape negotiation. This method merges characteristics of ant-based data clustering and cellular automata to represent complex datasets in meaningful visual clusters. Cellular ants demonstrates how a decentralized multi-agent system can autonomously detect data similarity patterns in multi-dimensional datasets and then determine the according visual cues, such as position, color and shape size, of the visual objects accordingly. Data objects are represented as individual ants placed within a fixed grid, which decide their visual attributes through a continuous iterative process of pair-wise localized negotiations with neighboring ants. The characteristics of this method are demonstrated by evaluating its performance for various benchmarking datasets.

1 Introduction

This paper proposes a simple approach towards unsupervised data visualization. It uses principles of self-organization to determine the visual representation of complex, high-dimensional datasets. Self-organizing systems generally consist of a number of similar elements that perform numerous internal interactions, which can spontaneously generate an inherently complex pattern on a global level. The rules that govern this process are informed by local information only, without any reference to the global pattern. The proposed method, coined *cellular ants*, uses self-organization to determine the visual attributes of data items, including position, shape, color and size. By self-adapting the visual representation to data attributes, this approach goes beyond the traditional notion of using fixed and predefined data mapping rules.

The cellular ant method combines insights from ant-based clustering in the field of data mining and cellular automata in the field of artificial life with data mapping principles from the data visualization domain. It can be considered as a simple data clustering technique that is capable of creating visual representations similar to those of multidimensional scaling. As a non-optimized prototype, it demonstrates how simple behavior rules are capable of clustering complex, high-dimensional and large datasets. This work is built upon the methodology defined in [1], which introduced the data scaling in a toroidal grid. In this paper, two novel features are introduced: *color negotiation* (similar to data labeling or data clustering) and *shape negotiation*.

A. Sattar and B.H. Kang (Eds.): AI 2006, LNAI 4304, pp. 826 – 836, 2006.

2 Related Work

Ant-based sorting was introduced by Deneubourg et al. [2] to describe different types of emergent phenomena in nature. Ants are represented as simple agents that are capable of roaming around in a toroidal grid, on which objects, representing data items, are randomly scattered. Ant actions are biased by probabilistic functions, so that ants are more likely to pick up objects that are isolated, and more likely to drop them in the vicinity of similar ones. A predefined *object distance measure variable* α determines this degree of similarity between pairs of data objects. Ant-based clustering has been used for data mining purposes, and has been combined with fuzzy-set theory [3], topographic maps [4], or bio-inspired genetic algorithms [5]. The cellular ant method differs from the standard method by mapping data items directly onto the ants themselves. Recent examples of this approach exist: Labroche et al. [6] associates data objects to ants and simulates meetings between them to dynamically build partitions, according to the data labels that best fit their genome.

Multi-dimensional scaling (MDS) displays the structure of distance-like datasets as geometrical pictures [7]. MDS representations are arranged in 2D space, in which the distance between pairs of data items denotes the degree of data similarity. Several similar data visualization techniques exist, for instance in combination with animation [8] or recursive pattern arrangements [9]. Multi-dimensional scaling differs from *clustering* in that clustering partitions data into classes, while MDS computes positions, without providing an explicit decomposition into groups. Self Organizing Maps (SOM) is an unsupervised clustering technique capable of detecting and spatially grouping similar data objects in topologically distinct classes [10]. This visualization method orders an initially random distribution of high-dimensional data objects as the emergent outcome of an iterative training process. In this paper, we describe how the cellular ant method is capable of unsupervised clustering, as it is capable of coloring ants in classes depending on an emergent data scaling topography.

Because the cellular ant methodology governs ants by principles of stigmergy and state density principles, it resembles that of *cellular automata*. Cellular automata is a computational method originally proposed by Ulam and Von Neumann [11]. It consists of a number of cells that each represents a discrete state (e.g. alive or dead). Cells are governed by behavior rules that are iteratively applied, and generally only consider the states of the neighboring cells. The cellular ant approach combines ant-based clustering and cellular automata, as the ants' reasoning takes into account grid cell states, rather than probabilistic functions. While ants can 'act' upon the environment and even change it to some degree, cellular automata 'make up' the environment itself. A recent clustering method [12] also maps data objects onto ants, that resemble cellular automata elements. It differs from our methodology as it does not order clusters, and is based on probability functions and internal ant states.

Agent-based visualizations have typically been used to display intrinsic relations (e.g. messages, shared interests) between agents for monitoring and engineering purposes [13], to represent complex fuzzy systems [14], or to support the choice of the most effective visualization method [15]. Other systems organize the visualization data flow, for instance by determining visualization pipeline parameters [16] or regulating rendering variables in distributed environments [17]. To our knowledge, agents have not yet been used to generate visualizations based on detected data

correlations. A few simple prototype applications of agent-based data visualization have been developed that are capable to represent complex data properties through an emergent, decentralized process: for instance, the *infoticle* (information-particle) metaphor is capable of representing time-varying data properties as recognizable motion typologies of dynamic particle or flocking patterns [18].

3 Approach

3.1 Cellular Ant Concept

Each single normalized data item (e.g. database tuple, row, object) corresponds to a single agent, coined *cellular ant*. Each single ant (and thus data item) is represented as a single colored square cell within a toroidal, rectangular grid. Each ant is governed by a set of simple behavior rules. These behavior rules are applied simultaneously to all ants, in a discrete and iterative way. Each ant can only communicate with ants in its immediate vicinity, limited to its eight neighboring cells. The dynamic behavior of an ant only depends only on the data values it represents, and the data values of its immediate neighbors. A cellular ant is capable of determining its visual cues autonomously, as it can move around or stay put, swap its position with a neighbor, and adapt a color or shape size, by a process of pair-wise negotiating. Each cellular ant is determined by four different negotiation processes: data scaling, position swapping, color determination and shape size adaptation. A detailed description of the data scaling and the ant swapping methodologies can be found in [1]. This paper will instead focus on the recent additions that determine the other visual cues of an agent.

At initialization, ants are randomly positioned within a grid. Similar to classical MDS (CMDS) method, each ant calculates the Euclidian distance between its own normalized data item and that of each of its eight neighbors. This data distance measure represents an approximation of the similarity between pairs of data items, even when they contain multidimensional data values. Next, an ant will only consider and summate those ants of which the pair-wise similarity distance is below a specific data *similarity tolerance threshold value t*. Value t is conceptually similar to the *object distance measure* α in common ant-based clustering approaches. However, t originates from a cellular automata approach in that it is a fixed and discrete value, which generates a Boolean result (either a pair of data objects is “similar enough” or not) instead of a continuous similarity value (e.g. representing a numerical degree of similarity between pairs of data objects). Depending on the amount of ants in its neighborhood it considers as ‘similar’, an ant will then decide either to stay put, or to move. For instance, an ant decides to stay put when it has more than four similar neighbors. The value four was chosen from the experience of cellular automata simulations, which tend to generate interesting cell constellations for this number.

As a result, ants with similar data items group together emergently. However, these clusters have little visualization value as they only convey the relative amounts of data objects. Therefore, a positional *swapping* rule was introduced that orders clusters internally as well as globally in respect to data similarity. As a result, diagrams are generated that look conceptually similar to those of basic CMDS approaches.

3.2 Color Negotiation

Conceptually, the color of an ant can be considered as the representation of its assumed *data class* or *data label*, so that the resulting diagrams resemble that of (ant-based) data clustering in the domain of data mining, but inherit the additional capability of being spatially and visually ordered. At initialization, all ants are assigned an unspecific color (white). At each iteration, ants execute the following behavior rules. Each ant that has not been swapped (and thus is probably well placed within its neighborhood) and is fully surrounded by eight similar neighbors, considers the degree of data similarity with all of its neighbors. If this degree is below a predefined, discrete *color seed similarity threshold c*, it will request the system to be assigned a unique color. As a result, such ants will act as initial 'color seeds'. All other ants will consider whether their neighborhood contains four or more data objects that are smaller than *t* but larger than *c*. If so, such ant is 'satisfied' with its current position and will adopt the color of the most similar ant in its neighborhood. In practice, once colors are introduced within the grid, they will spread gradually over the ant population. Once the collection of ants is sufficiently ordered, several color seeds become introduced. Because of the multitude of pair-wise interactions, any surplus of colors (in respect to data clusters) will disappear, while any shortfall of colors will reemerge once a potential seed is surrounded by eight neighbors.

However, data clusters that contain less than nine members in a dataset cannot be recognized. In some cases, ants continuously 'swap' from one color to another, within a single visual cluster. This dynamic phenomenon generally indicates that the positional rules could not accurately spatially group two different data types, which nonetheless were recognized by the color clustering rule. A future research direction could consist of inversely informing the positional clustering of the color label values.

3.3 Size Negotiation

Instead of mapping a data value to a specific shape size, each ant can map one of its data attributes onto its size by negotiating with its neighbors. Conceptually, the size of an agent does not necessarily correspond to the 'exact' value of that data attribute, but rather how a data value locally relates to its neighborhood, and therefore whether clusters are *homogeneous* in respect of a specific data attribute. Because no direct, predefined mapping rule between value and visual cue exists, the shape size scale can automatically adapt to any data scale, in an autonomous and self-organizing way.

For each iteration step, the visual shape size of an ant is determined by following inducements. First, an ant A chooses a random neighboring ant B with whom it compares its one-dimensional data value DA and circular radius size SA, measured in screen pixels. Step size P is a predefined amount of pixels. Ant A evaluates whether its radius versus data value ratio is similar to that of ant B, and adapts its own as well as its neighbor's shape size accordingly. If, in comparison to ant B, its size SA is too large in relation to its data value DA, it will decide to 'shrink' by decreasing its amount of available pixels with P pixels, and then provides these P pixels to ant B.

$$\begin{cases} S_A > \dfrac{S_B}{D_B}.D_A \Rightarrow \begin{cases} S_A = S_A - P \\ S_B = S_B + P \end{cases} \\ S_A < \dfrac{S_B}{D_B}.D_A \Rightarrow \begin{cases} S_A = S_A + P \\ S_B = S_B - P \end{cases} \end{cases} \qquad \begin{cases} S_A > S_{\max} \Rightarrow \begin{cases} S_A = S_A - P \\ S_B = S_B - P \end{cases} \\ S_A < S_{\min} \Rightarrow \begin{cases} S_A = S_A + P \\ S_B = S_B + P \end{cases} \end{cases} \tag{1}$$

These rules assure that no visual overlapping of ant shapes can occur. An additional rule checks whether ants do not grow too large or too small: when an ant becomes too large, it will 'punish' and shrink its neighbor, so that in the future, this 'action' will not longer be required. This constraint will emergently 'detect' the upper and lower shape size boundaries according to the data scale, and spreads throughout all ants. Because all ants are complying with these rules in random directions and over multiple iterations, a stable constellation of shape sizes appears in an emergent way.

3.4 Performance Measurements

A simple performance graph informs users of the actual visualization state. The number of similar ants in each ant neighborhood is squared for each ant, and summated over all ants. The visualization efficiency over time corresponds to the slope of the according graph: once a plateau value has been reached over a number of iterations, the visualization has reached a stable state and can be halted. Figure 1 captures the clustering performance of the 'Thyroid' dataset depending on varying variables similarity tolerance threshold value t (vertical) and color seed similarity treshold c, for the different amounts of iterations and different initial seeds. These 'solution space' diagrams enable users to pick the most appropriate variable values. The diagrams illustrate the 'hotspots' of effective clustering values, and the limited influence of the amount of iterations and the random initialization seeds on the quality of the results. Each initialization seed will result in different constellations and thus clustering error rates (see Table 1 for standard deviation).

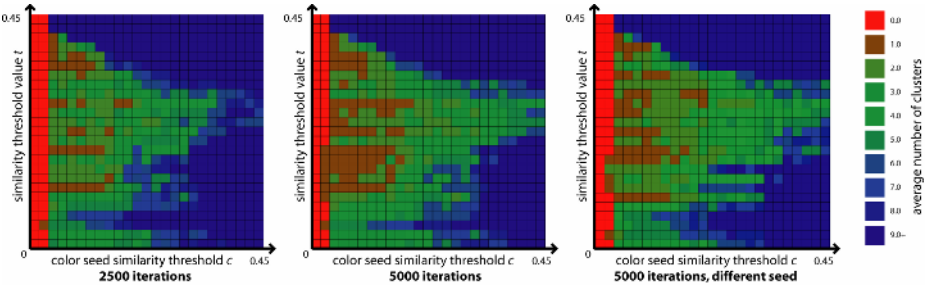

Fig. 1. A dot plot diagram of the Cellular Ant's method performance by varying the similarity tolerance threshold value t (vertical) and the color seed similarity threshold c (horizontal) for the Thyroid dataset (see Table 1). Color denotes the amount of data labels/clusters detected. These diagrams.

4 Application

4.1 Case Studies

The synthetic dataset visualized in Figure 2 consists of 500 data items with two data dimensions. Data objects and classes are distributed using a Gaussian distribution function to demonstrate the data scaling, color negotiation and size negotiation capabilities. The color negotiation successfully resulted in four distinct clusters or data class labels. As shown by the highlighted ants, the clusters are *internally* ordered: data items that are similar in data space, are positioned nearby each other in visualization space. Also, the clusters are *globally* ordered: clusters that are dissimilar in data space, have no 'common' borders in visualization space. For instance, the purple and yellow clusters (or blue and green) have no common orthogonally directed borders and have empty cells between their borders in visualization space, as they are diagonally positioned in data space, and thus have a larger 'global' data distance.

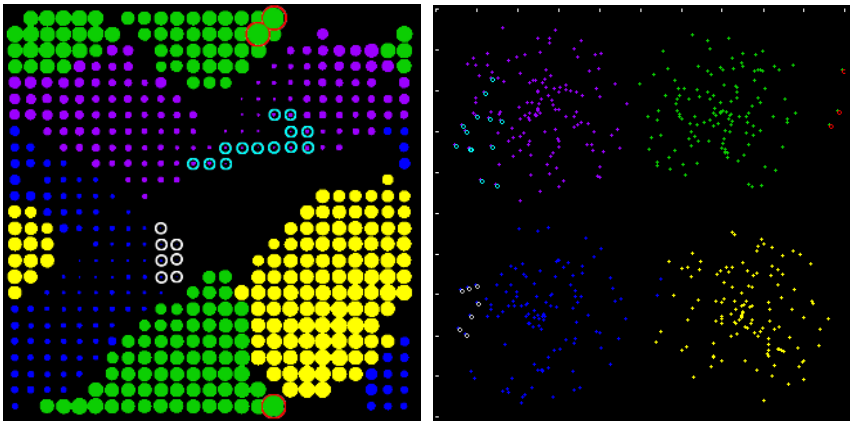

Fig. 2. Visualization with color and shape negotiation, size representing the 1st data attribute (left), and data scatterplot (right), on which the 1st data attribute is mapped on the X-axis. Corresponding ants (left) and data items (right) are highlighted in red, cyan and white.

The display of different circular shape sizes enables the user to understand how a single data attribute is distributed over the clustering representation. For instance, the three largest ants (highlighted in red) are positioned within an outlying green sub-cluster (see Fig. 2). The ants highlighted in cyan and white show that the smallest ants in shape size correspond to those ants with the smallest data value for that attribute.

Figure 3 shows two different clustering techniques of the car dataset, containing 38 items and 7 data dimensions, as taken from [19]. On the left, the multidimensional scaling technique positioned the cars in three apparent clusters (the color coding was artificially added by the authors for visual clarification). The cellular ant method, on the right, positioned the cars in a single visual cluster, but recognized 3 separate class labels that roughly correspond to those apparent clusters.

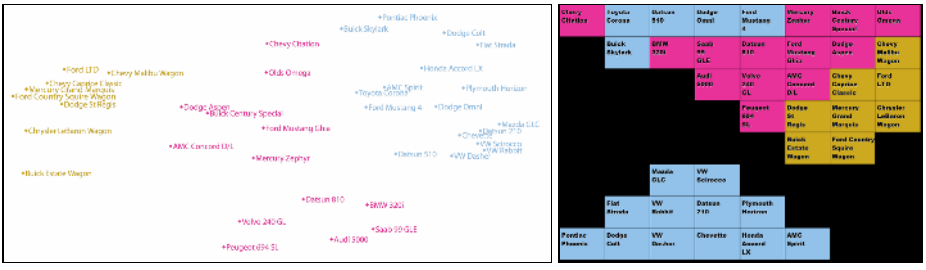

Fig. 3. The car dataset represented by MDS (left, based on [19], color coding artificially added by the authors) and the Cellular Ants representation with color negotiation (right)

Figure 4 illustrates how shape size negotiation is used to clarify data dependencies for high-dimensional datasets, without prior knowledge of the data scale and without using any predefined data mapping rules. Figure 4 uses the same representation as Figure 3, and maps a single data attribute to the decentralized shape size negotiation. As a result, one can investigate how the clusters are internally ordered for different attributes. Here, it shows the relative dominance of the cylinder count and MPG within specific clusters, and some cars visually stand out within the formed clusters.

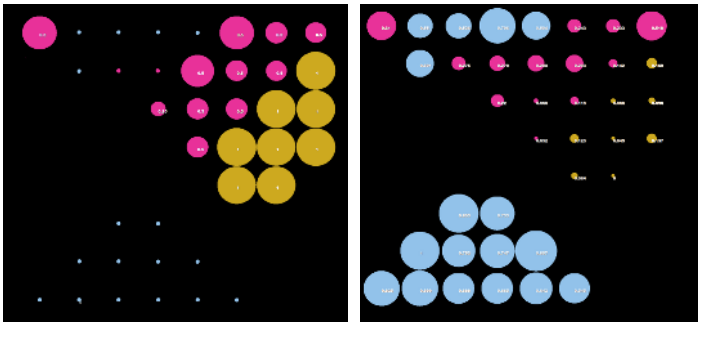

Fig. 4. A Cellular Ant representation in a toroidal grid using color and shape negotiation. Data attribute represented by the shape size: cylinders (left) and Miles per Gallon (MPG) (right).

The cellular ant method has been evaluated with typical benchmarking data, such as the IRIS dataset. The iteration timeline in Figure 5 (left) shows how several colors were introduced, but only three remained. In effect, the IRIS dataset is clustered in two distinct visual clusters, but the color negotiation recognizes that three different data classes exist, of which two are very similar. This interplay between visual and spatial clustering contains a high visualization value. The figure shows a momentary snapshot only: during the simulation the orange and yellow colors take over ants from each other. Using shape negotiation, one can investigate how a data attribute is relatively distributed over a cluster. As shown in Figure 5 (right), subclusters of high or low data values are made apparent, demonstrating the ordering power of the swapping rule. For instance, one can perceive that for attribute 4, the yellow type

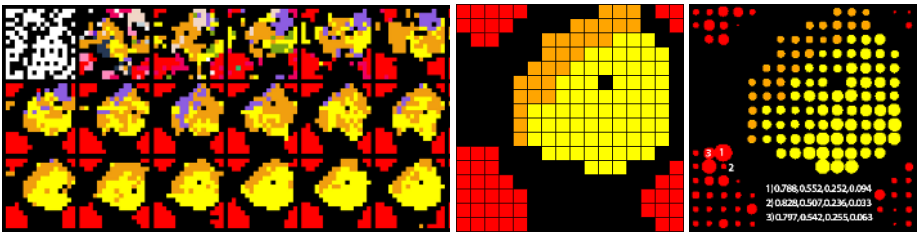

Fig. 5. Clustered IRIS dataset (150 data items, 4 attributes, 3 clusters, 1821 iterations) in a toroidal grid. Left: iteration timeline. Middle: resulting spatial clustering with color negotiation. Right: same result, with shape size negotiation for attribute 4.

(Virginia) has larger data values than the orange one (Versacolor), and that this attribute is very volatile for the red type (Setosa) (varying between values 0.1 and 0.3) when considering their relative numerical proportion to one another within the cluster.

Table 1 lists the performance of the color negotiation (or data classification) for various standard benchmarking datasets, after executing the cellular ant algorithm over 50 runs, each with a different, random initialization seed. The clustering error rate is calculated by counting the ants with correct colors over the whole population, and dividing this summation by the total amount of ants. In general, these results are worse but relatively similar to comparable clustering methods, such as reported in [6].

Table 1. Performance measurements of the color negotiation method for different benchmarking datasets. Averages are taken over 50 runs, each with a different random seed.

				#Clusters		Clustering Error	
Datasets	**#Objects**	**#Attributes**	**#Clusters**	**Average**	**[std]**	**Average**	**[std]**
Iris	150	4	3	2.68	[0.65]	*0.37*	*[0.11]*
Pima	768	8	2	1.14	[0.35]	*0.36*	*[0.04]*
Thyroid	215	5	3	3.45	[0.80]	*0.41*	*[0.14]*

5 Discussion

The performance of the current implementation depends on two variables: the data similarity tolerance threshold t and color seed similarity threshold c. The ant density (or the grid size determined by dividing the available cells in the grid by the dataset size) has been kept constant at about 75%. Similarly to the object distance measure α in common ant-based clustering approaches, the optimal value of t and c cannot be determined without prior knowledge of the dataset, unless the value is adaptable [20].

We consider the current implementation as a simple proof-of-concept prototype, and kept its implementation as simple as possible. Therefore, the scaling and clustering performance of the cellular ant method is not that effective as existing MDS methods. Its first aim is not to compete with alternative approaches, but rather to be considered as an early prototype towards more powerful cellular automata

clustering algorithms, or towards data visualizations that are emergent and self-adaptive. As shown in the diagrams, the combination of spatial clustering with data class clustering can result in visual representations that are meaningful and useful. Following aspects can also be considered.

- **Performance.** In its current simple form of implementation, the amount of required iterations seems to be similar with comparable approaches in the field of ant-based data mining. However, the 'data-to-ant' model always requires less iteration steps because all data objects are able to move to increasingly ideal positions simultaneously. Similarly to existing ant-based data mining optimizations, the clustering performance could be addressed with increasing the data similarity cap value over time, so that clusters grow more rapidly and steadily. The method requires a considerable amount of calculations, as each ant is required to calculate many pair-wise dependencies for each iteration step.
- **Clustering Quality.** Grid density influences the clustering quality in two ways. Small grid densities do not assure that ants with equal colors (data labels) will be represented in a single spatial cluster, because two or more clusters might emerge without ever 'touching' and 'merging'. Too dense grids generate single, large groups with diverse labels and thus little visualization value.
- **Simplicity**. The current behavior rules have been kept as simple as possible, to demonstrate the potential value of the cellular automata-like decentralized negotiation for data mining and data visualization purposes. Further calculation optimization or solutions towards data size scalability can be accomplished by considering a combination of following three approaches: 1) real-time data optimization, including data approximation and gradual data streaming, 2) agent adaptation, which includes the distribution or balancing of loads between multiple agents, and 3) agent cooperation, by generating adaptive coalition formations of 'super agents' that have similar objectives, experience or goals.

6 Conclusion

This paper presented two new features of the cellular ant method: color (data clustering) and shape size negotiation. It combines ant-based data mining algorithms with cellular automata insights, or data scaling with data clustering to derive an approach that is capable of representing multidimensional datasets. The resulting diagrams are visually similar to those of ant-based data mining clustering approaches. However, the clusters are also similar to multi-dimensional scaling images, as they are ordered internally as well as globally over multiple data dimensions. As a simple prototype towards self-organizing visualization, inter-agent negotiations determine typical visual cues, such as position, color and size, depending on multidimensional data properties. Color negotiation can recognize data clusters of similar type. Shape size negotiation displays the relative distribution of a single data attribute and the internal structure of clusters. Conceptually, the self-adaptive, unsupervised data mapping process of the cellular ants proposes a conceptual alternative to the common fixed data mapping rules that are based on preconceived dataset assumptions.

Some of the limitations of method are caused by the simplicity of the rule-based approach, and its dependency on fixed, discrete cellular automata characteristics,

instead of more continuous probability functions. Several optimizations can be accomplished, for instance by altering the data similarity tolerance threshold over time, or by informing agents of the global effectiveness. As a simple prototype, it demonstrates a potential future in which data visualization agents are capable of autonomously detecting complex data patterns and proactively acting upon them to make underlying data phenomena more visually apparent and the perceptual and cognitive understanding by humans more effective.

References

1. Vande Moere, A., Clayden, J.J.: Cellular Ants: Combining Ant-Based Clustering with Cellular Automata. International Conference on Tools with Artificial Intelligence (ICTAI'05). IEEE (2005) 177-184
2. Deneubourg, J., Goss, S., Franks, N., A., S.F., Detrain, C., Chretian, L.: The Dynamics of Collective Sorting: Robot-Like Ants and Ant-Like Robots. From Animals to Animats: 1st International Conference on Simulation of Adaptative Behaviour (1990) 356-363
3. Schockaert, S., De Cock, M., Cornelis, C., Kerre, E.E.: Fuzzy Ant Based Clustering. Lecture Notes in Computer Science 3172 (2004) 342-349
4. Handl, J., Knowles, J., Dorigo, M.: Ant-Based Clustering and Topographic Mapping. Artificial Life 12 (2005) 35-61
5. Ramos, V., Abraham, A.: Evolving a Stigmergic Self-Organized Data-Mining. International Conference on Intelligent Systems, Design and Applications, Budapest, (2004) 725-730
6. Labroche, N., Monmarché, N., Venturini, G.: A New Clustering Algorithm Based on the Chemical Recognition System of Ants. European Conference on Artificial Intelligence, Lyon, France (2003) 345-349
7. Torgerson, W.S.: Multidimensional Scaling. Psychometrika 17 (1952) 401-419
8. Bentley, C.L., Ward, M.O.: Animating Multidimensional Scaling to Visualize N-Dimensional Data Sets. Symposium on Information Visualization. IEEE (1996) 72 -73
9. Ankerst, M., Berchtold, S., Keim, D.A.: Similarity Clustering of Dimensions for an Enhanced Visualization of Multidimensional Data. Symposium on Information Visualization (Infovis'98). IEEE (1998) 52-60
10. Kohonen, T.: The Self-Organizing Map. Proceedings of the IEEE 78 (1990) 1464-1480
11. Von Neumann, J.: Theory of Self-Reproducing Automata. University of Illinois Press, Illinois (1966)
12. Chen, L., Xu, X., Chen, Y., He, P.: A Novel Ant Clustering Algorithm Based on Cellular Automata. Conference of the Intelligent Agent Technology (IAT'04). IEEE (2004) 148-154
13. Schroeder, M., Noy, P.: Multi-Agent Visualisation based on Multivariate Data. International Conference on Autonomous Agents. ACM Press, Montreal, Quebec, Canada (2001) 85-91
14. Pham, B., Brown, R.: Multi-Agent Approach for Visualisation of Fuzzy Systems. Lecture Notes in Computer Science 2659 (2003) 995-1004
15. Healey, C.G., Amant, R.S., Chang, J.: Assisted Visualization of E-Commerce Auction Agents. Graphics Interface 2001. Canadian Information Processing, Ottawa, (2001) 201-208
16. Ebert, A., Bender, M., Barthel, H., Divivier, A.: Tuning a Component-based Visualization System Architecture by Agents. International Symposium on Smart Graphics. Hawthorne, IBM T.J. Watson Research Center (2001)

17. Roard, N., Jones, M.W.: Agent Based Visualization and Strategies. Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG), Pilzen (2006)
18. Vande Moere, A.: Time-Varying Data Visualization using Information Flocking Boids. Symposium on Information Visualization (Infovis'04). IEEE, Austin, USA (2004) 97-104
19. Wojciech, B.: Multivariate Visualization Techniques. Vol. 2006 (2001)
20. Handl, J., Meyer, B.: Improved Ant-based Clustering and Sorting in a Document Retrieval Interface. International Conference on Parallel Problem Solving from Nature (PPSN VII) LNCS 2439 (2002) 913-923