Clustering with Swarm Algorithms compared to Emergent SOM

Lutz Herrmann and Alfred Ultsch

Databionics Research Group
Department of Mathematics and Computer Science
Philipps University of Marburg
{lherrmann,ultsch}@informatik.uni-marburg.de

Summary. Swarm Based clustering (SBC) is a promising nature-inspired technique. A swarm of stochastic agents performs the task of clustering high-dimensional data on a low-dimensional output space. Most SBC methods are derivatives of the Ant Colony Clustering (ACC) approach proposed by Lumer and Faieta. Compared to clustering on Emergent Self-Organizing Maps (ESOM) these methods usually perform poorly in terms of topographic mapping and cluster formation. A unifying representation for ACC methods and Emergent Self-Organizing Maps is presented in this paper. ACC terms are related to corresponding mechanisms of the SOM. This leads to insights on both algorithms. ACC can be considered to be first-degree relatives of the ESOM. This explains benefits and shortcomings of ACC and ESOM. Furthermore, the proposed unification allows to judge whether modifications improve an algorithm's clustering abilities or not. This is demonstrated using a set of critical clustering problems.

Key words: Clustering, Emergent Self-Organizing Maps, Swarm Intelligence

1 Introduction

Flocking behaviour of social insects has inspired various algorithms in numerous research papers over the last decade due to the ability of simple interacting entities to exhibit sophisticated self-organization abilities. A particularly interesting field of application is cluster analysis, i.e. the retrieval of groups of similar objects in high-dimensional spaces. The idea behind Ant Colony Clustering (ACC) is that autonomous stochastic agents, called ants, move data objects on a low-dimensional regular grid such that similar objects are more likely to be placed on nearby grid nodes than dissimilar ones. This task is referred to as *topographic mapping*.

Most popular ACC methods are based on the algorithm proposed by Lumer and Faieta [7]. The most advanced derivative might be ATTA (Adaptive Time Dependent Transporter Ants, [4]). ACC methods are known for at least two flaws: results are highly dependent on parametrization [1] and even ATTA has found to be "not competitive to the established methods of Multi-dimensional Scaling or Self-Organizing Maps" [4] in terms of topographic mapping.

In the following sections, the basic ACC algorithm by Lumer/Faieta is introduced in a notation consistent with the well-known Batch-SOM. A unifying representation for both methods is therefore derived in Section 3. Sections 4 and 5 describe how to improve topographic mappings of ACC methods on basis of Batch-SOM. Finally, in Section 6 the effect of altered objective functions is empirically verified.

2 Ant Colony Clustering

The ACC method proposed by Lumer and Faieta [7] operates on a fixed regular low-dimensional grid $\mathbb{G} \subset \mathbb{N}^2$. A finite set of input samples X from a vector space with norm $\|.\|$ is projected onto the grid by $m : X \to \mathbb{G}$. The mapping m is altered by autonomous stochastic agents, called ants, that move input samples $x \in X$ from $m(x)$ to new location $m'(x)$. Ants move randomly on neighbouring grid nodes. Ants might pick input samples when facing occupied nodes and drop input samples when facing empty nodes. The probability for picking input sample $x \in X$ from node $i = m(x)$ and dropping picked x on node $j \in \mathbb{G}$ is $p_{pick,x}(i) = \left(\frac{k_1}{k_1+\phi_x(i)}\right)^2$ and $p_{drop,x}(j) = \left(\frac{\phi_x(j)}{k_2+\phi_x(j)}\right)^2$, respectively. Here, $k_1, k_2 \in \mathbb{R}^+$ are threshold constants. $\phi_x(i)$ denotes the average similarity between $x \in X$ and input samples located on the so-called perceptive neighbourhood. Usually, the perceptive neighbourhood consists of $\sigma^2 \in \{9, 25\}$ quadratically arranged nodes at which the ant is located in the center. The set of input samples mapped onto the perceptive neighbourhood around $i \in \mathbb{G}$ is denoted with $N_x(i) = \{y \in X : y \neq x,\ m(y) \text{ neighbouring } i\}$. In this context, ϕ is referred to as *objective function* since its minimization determines the ants' probabilistic modifications of mapping $m : X \to \mathbb{G}$.

$$\phi_x(i) = \frac{1}{\sigma^2} \sum_{y \in N_x(i)} \left(1 - \frac{\|x - y\|}{\alpha}\right) \tag{1}$$

ACC methods lead to a local sorting of input samples on the grid in terms of similarities. Ants gather scattered input samples into dense piles. In literature, it has been noticed that ACC derivatives are prone to produce too many and too small clusters [1] [4]. For illustration see Figure 1.

3 Analysis of Ant Colony Clustering by means of Self-Organizing Batch Maps

In order to compare Self-Organizing Maps (SOM) and Ant Colony Clustering (ACC), a unifying basis for both algorithms is derived. Input data X and

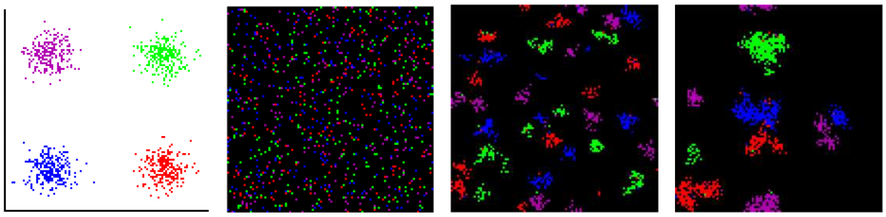


Fig. 1. Typical result of ACC methods. From left to right: gaussian data with 4 clusters, initial mapping of data objects, dense clusters appear, too many clusters with topological defects have finally emerged [1].

output grid $\mathbb{G} \subset \mathbb{N}^2$ are identical and mapping function $m : X \to \mathbb{G}$ is iteratively update in both cases as well.

Self-Organizing Batch Maps (Batch-SOM) are well-known artificial neural networks that consist of grid $\mathbb{G}$, codebook vectors $w_i \in \mathbb{R}^n, i \in \mathbb{G}$ and a mapping function $m : X \to \mathbb{G}$ with $m(x) = \arg\min_{i \in \mathbb{G}} \|x - w_i\|$. The codebook vectors are defined according to Equation 2 at which $h : \mathbb{G} \times \mathbb{G} \to [0, 1]$ denotes a time-dependent neighbourhood function. An update of $m : X \to \mathbb{G}$ leads to an update of codebook vectors $w_i, i \in \mathbb{G}$ and vice versa. This is how the Batch-SOM modifies mapping $m : X \to \mathbb{G}$. For details see [6].

In literature [10], two main types of Self-Organizing Maps (SOM) can be distinguished: first, SOM in which each codebook vector represents a single cluster of input samples. In contrast to that, SOM may be used as tools for visualization of structural features of the input space. A single codebook vector is meaningless. A characteristic of this paradigm is the large number of codebook vectors, usually several thousands ($\geq$ 4000). These SOM are referred to as Emergent Self-Organizing Maps (ESOM). For details see [10].

$$w_i = \frac{\sum_{x \in X} h(m(x), i) \cdot x}{\sum_{x \in X} h(m(x), i)} \tag{2}$$

A meaningful objective function for the Batch-SOM is derived from the quantization error $\|x - w_i\|$ because its minimization determines the update of $m : X \to \mathbb{G}$. Resolving the quantization error with Equation 2 leads to objective function Φ of the Batch-SOM (see Equation 3). Φ_x represents the norm of averaged differences $x - y$ over grid-neighbouring input samples $y \in X$.

$$\Phi_x(i) = \frac{\left\| \sum_{y \in X} h(m(y), i) \cdot (x - y) \right\|}{\sum_{y \in X} h(m(y), i)} \tag{3}$$

In the following, the mechanism of picking and dropping ants is no longer subject of consideration. In [8] it was shown that collective intelligence can be discarded in ACC systems, i.e. same results were achieved without ants but using objective function ϕ directly for probabilistic cluster assignments. This simplification is evident: over a sufficient period of time, randomly moving ants may select any arbitrary subset of input samples, but re-allocation through picking and dropping depends on ϕ only. Probability of selection is the same on all input samples such that ants might be omitted in favor of any other subset sampling technique.

A meaningful symmetrical neighbourhood function $h : \mathbb{G} \times \mathbb{G} \rightarrow [0, 1]$ for ACC methods is defined according to the perceptive neighbourhood of ants, i.e. $h(i, j)$ is 1 if $j \in \mathbb{G}$ is located in the perceptive neighbourhood of node $i \in \mathbb{G}$ and 0 elsewhere. This neighbourhood function allows to restate ϕ as Equation 4 by use of $|N_x(i)| = \sum_{y \in X} h(m(y), i)$.

$$\phi_x(i) = \frac{|N_x(i)|}{\sigma^2} \cdot \left(1 - \frac{\Phi'_x(i)}{\alpha}\right) \text{ with } \Phi'_x(i) = \frac{\sum_{y \in X} h(m(y), i) \cdot \|x - y\|}{\sum_{y \in X} h(m(y), i)} \tag{4}$$

The ACC error function $\phi = \frac{|N|}{\sigma^2}(1 - \frac{\Phi'}{\alpha})$ incorporates Φ' that is a weighted sum of local input space distances. Obviously, Φ' measures the local stress of topographic mapping $m : X \rightarrow \mathbb{G}$, comparable to Φ of the Batch-SOM. Φ' even acts as an upper limit to Φ since $\forall x \in X, i \in \mathbb{G} : \Phi_x(i) \leq \Phi'_x(i)$. Due to that $1 - \frac{\Phi'}{\alpha}$ is referred to as *topographic term* of ACC algorithms.

The term $\frac{|N_x(i)|}{\sigma^2}$ estimates the output space density around grid node $i \in \mathbb{G}$. Therefore, it is referred to as *output density term* of ACC algorithms.

	Batch-SOM	ACC
neighbourhood $h : \mathbb{G} \times \mathbb{G} \rightarrow [0, 1]$	large, shrinking	small, fixed
update of $m : X \rightarrow \mathbb{G}$	deterministic	probabilistic
searching for update of $m : X \rightarrow \mathbb{G}$	global $\mathbb{G}$	local $\subset \mathbb{G}$
objective function	Φ	$\frac{\|N\|}{\sigma^2}(1 - \frac{\Phi'}{\alpha})$
termination	cooling scheme	never

Table 1. differences of Batch-SOM and Ant Colony Clustering (ACC)

A unifying framework for analysis and assessment of Batch-SOM and ACC exists by means of objective functions Φ and ϕ. Both functions are denoted by means of three functions: norm $\|.\|$, neighbourhood $h : \mathbb{G} \times \mathbb{G} \rightarrow [0, 1]$ and mapping $m : X \rightarrow \mathbb{G}$.

This leads to the following insights: The ACC method uses a fixed neighbourhood function with small radius, whereas Batch-SOM uses shrinking neighbourhood functions with large radiuses. ACC has a probabilistic update of mapping $m : X \rightarrow \mathbb{G}$, whereas Batch-SOM is deterministic. The objective function of ACC algorithms decomposes into an output density term

$\frac{|N|}{\sigma^2}$ and a term $1 - \frac{\Phi'}{\alpha}$ related to topographic quality. Φ' is easily identified as a topographic distortion measure because of its relation to Φ of Batch-SOM. Therefore, the ACC algorithm is easily convertible into a special case of Batch-SOM, and vice versa. For a brief overview of differences see Table 1.

4 Improvement of Ant Colony Clustering

ACC methods are prone to produce bad topographic mappings, e.g. too many, too small and topographically distorted clusters. If one regards ACC as a derivative of the Batch-SOM, improvement of topographic mapping can easily be achieved.

Maximization of the *topographic term* $1 - \frac{\Phi'}{\alpha}$ corresponds to minimization of Φ' and Φ, too. This is known to produce sufficiently topography preserving mappings $m : X \rightarrow \mathbb{G}$, e.g. when using Batch-SOM [6].

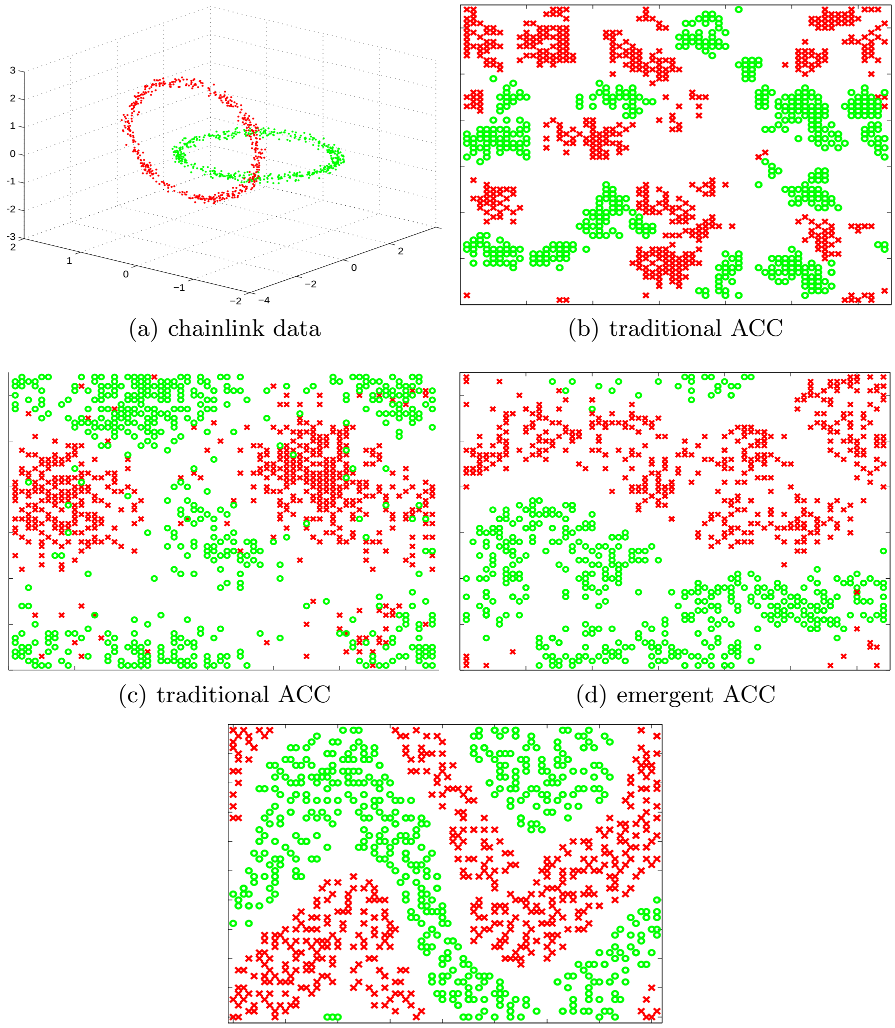
In contrast to that, the *output density term* $\frac{|N|}{\sigma^2}$ has some major flaws. First, the output density term leads to maximization of output space densities, instead of preservation. Obtained mappings are, therefore, not related to the configuration of available clusters in the input space. Traditional ACC algorithms are not allowed to assign two or more objects to a single grid node (see Section 2) in order to prevent the mapped clusters from collapsing into a single grid node. Due to that, densities of input data can hardly be preserved on grid $\mathbb{G}$. In comparison with the topographic term, the output density term is much easier to maximize and, therefore, will distort the objective function ϕ. Accounting of output densities is prone to distort the formation of correct topographic mappings because it is responsible for additional local optima of ϕ.

The topographic term $1-\frac{\Phi'}{\alpha}$ of the ACC objective function depends on the shape of the neighbourhood function $h : \mathbb{G} \times \mathbb{G} \rightarrow \{0, 1\}$. Usually, the neighbourhoods' sizes are chosen as $\sigma^2 \in \{9, 25\}$, i.e. the immediate neighbours. From the Batch-SOM it is known that the cooling scheme of the neighborhood radius influences the goodness for topographic mapping very strongly (see [5] for details). A bigger radius enables a more continuous mapping in the sense that proximities existing in the original data are visible on the grid. This is evident because smaller neighbourhoods are more likely to exclude parts of a cluster.

In order to cope with the shortcomings mentioned above, we introduce the *Emergent Ant Colony Clustering* method. An ACC method is said be be emergent if it fulfills the following conditions:

- Ants' modifications of mapping $m : X \rightarrow \mathbb{G}$ is directed by maximization of $1 - \frac{\Phi'}{\alpha}$ and minimization of Φ', respectively.
- Ants do not account for output densities.

- The perceptive neighbourhood of ants is not limited to immediate neighbours on grid $\mathbb{G}$. Instead, bigger neighbourhood radiuses are to be chosen in order to obtain ESOM-like mappings.



(a) chainlink data

(b) traditional ACC

(c) traditional ACC

(d) emergent ACC

(e) emergent SOM

Fig. 2. ACC projects looped cluster structures on a *toroid* grid. (a) Chainlink data from FCPS [9]. (b) Traditional ACC with small σ produces too many small clusters. (c) Traditional ACC with big σ produces fewer clusters, but no loops. (d) Emergent ACC enables the formation of looped clusters. (e) Emergent SOM enables the formation of looped clusters.

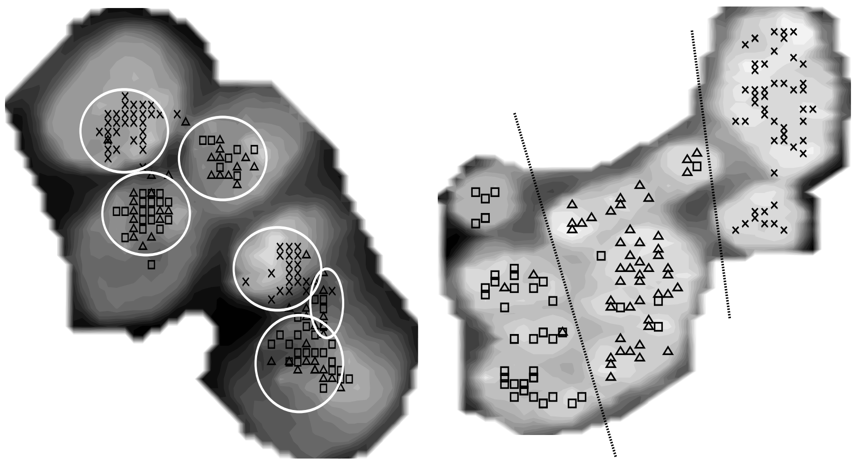
Figure 2 illustrates the ability of emergent ACC method to preserve even looped input space clusters, which is hardly possible for traditional ACC.

5 Data Analysis with Emergent Ant Colony Clustering

Emergent ACC usually will provide an ESOM-like projection, i.e. input samples are uniformly mapped onto the grid. See Figure 2 for illustration. In this case, cluster retrieval cannot be achieved according to sparse regions dividing dense clusters on the grid.

A promising technique for cluster retrieval is based on so-called U-Maps [10]. Arbitrary projections from normed vector spaces onto grid $\mathbb{G} \subset \mathbb{N}^2$ are transformed into landscapes, so-called U-Maps. The U-Map technique assigns each grid node a height value that represents the averaged input space distance to its' neighbouring nodes and codebook vectors, respectively. Clusters lead to valleys on U-Maps whereas empty input space regions lead to mountains dividing the cluster valleys. This is illustrated in Figure 3 using Fisher's well-known iris data [2]. Traditional ACC produces too many valleys, whereas Emergent ACC preserves cluster structures.

The U*C cluster algorithm uses the so-called watershed transformation to retrieve cluster valleys on U-Maps. See [11] for details.



(a) Traditional ACC (b) Emergent ACC

Fig. 3. Well known iris data [2]: setosa (×), versicolor (△), virginica (□). U-Maps shown as islands generated from toroid grids. Dark shades of gray indicate high inter-cluster distances. (a) Too many small clusters emerge from traditional ACC. (b) Emergent ACC preserves three clusters after the same learning epochs.

6 Experimental Settings and Results

In order to measure the distortion of a topographic mapping method in question, a collection of fundamental clustering problems (FCPS) is used [9]. Each data set represents a certain problem that arbitrary algorithms shall be able to handle when facing unknown real-world data. Here, traditional and emergent ACC are tested on which one delivers the best topographic mapping.

A comprehensive overview on topographic distortion measurements can be found in [3]. Here, the so-called *minimal path length* (MPL) measurement is used. It is an easy-to-compute measurement that sums up input space distances of grid-neighbouring data objects and codebook vectors, respectively.

$$mpl = \sum_{x \in X} \frac{1}{|N_x|} \sum_{y \in N_x} \|x - y\| \tag{5}$$

Lower MPL values indicate less topographic distortion when moving on the grid and, therefore, a more trustworthy topographic mapping. Each algorithm is run several times with the same parametrization. MLP values indicate if accounting for output densities assists the formation of good topographic mappings, or not. All data sets from the FCPS collection were processed with the same parameters established in literature, i.e. $\alpha = 0.5$, $\sigma^2 = 25$, $k_1 = 0.3$ and $k_2 = 0.1$ on a 64×64 grid with 100 ants during 100000 iterations. The results are illustrated in Figure 4. Accounting for output densities leads to increasing MPL values on an average, i.e. worsenings of topographic mappings. Significance has been confirmed using a Kolmogorov-Smirnov test on a $\alpha = 5\%$ level. All obtained p-values are below 10^{-5}.

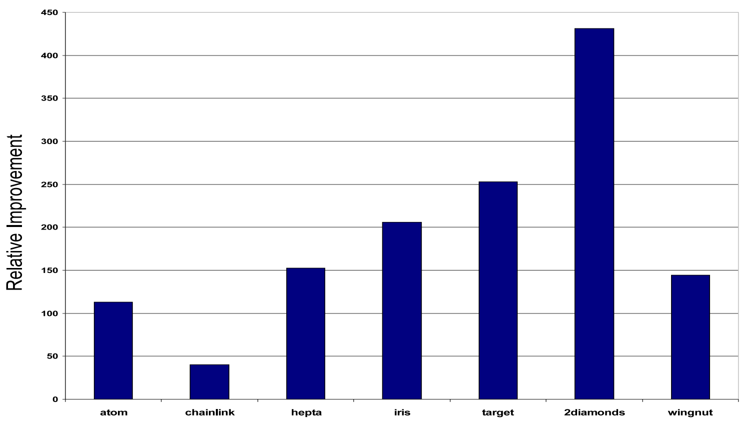


Fig. 4. Improvement of topographic quality measured by *minimal path length* method: percental z-scores of traditional over emergent ACC. Emergent ACC leads to improvements between 50% to 400% when compared to traditional ACC on different FCPS data sets.

7 Discussion

This work shows a previously unknown relation of two topographic mapping techniques, namely Self-Organizing Batch-Maps and Ant Colony Clustering (ACC). It is based on the assumption [8] that stochastic agents, e.g. ants, are nothing more than an arbitrary sampling technique that is to be omitted for further analysis of formulae. This simplification is evident but may be invalid for stochastic agents guided by more than just randomness and topographic distortion, e.g. ants following pheromone trails. Our analysis of formulae does not cover popular algorithms that are not ACC derivatives following the Lumer/Faieta scheme.

Minimal path lengths (MPL), as proposed in Section 6, are well-known topographic distortion measures. The length of *paths* is normalized by the cardinality $|N_x|$ of the corresponding grid neighbourhood, i.e. the number of objects mapped onto the grid neighbourhood. This is supposed to decrease error values of locally dense mappings, as produced by traditional ACC, because small radial neighbourhoods usually do not cover objects of another cluster, since locally dense mappings imply sparse dividing grid regions around clusters. Nevertheless, traditional ACC produces bigger MPL errors than emergent ACC that is not accounting for densities. We conclude that the topographic mapping quality is improved beyond our empirical evaluation.

Traditional and emergent ACC methods do not converge due to the architecture of stochastic agents. Instead, they enable perpetual machine learning. ACC methods are, therefore, to be favored over traditional methods, like Self-Organizing Maps and hierarchical clustering, when dealing with incremental learning tasks. In contrast to Self-Organizing Maps, ACC methods enable the creation of topographic maps despite the absence of vector-space axioms, i.e. when pairwise (dis)similiarity data is available only.

8 Summary

To the best of our knowlege, this is the first work that shows how the Ant Colony Clustering (ACC) method by Lumer and Faieta [7] is related to Self-Organizing Maps [6]. The mechanism of picking and dropping ants was omitted in favor of a formal analysis of the underlying formulae and comparison with Kohonen's Batch-SOM. It could be shown that a unifying framwork for both methods does exist in terms of closely related topographic error functions. The ACC method is to be considered a probabilistic, first-class relative of the Batch-SOM. The behaviour of ACC methods becomes explainable on that unifying basis.

ACC methods exhibit poor clustering abilities because of distorted topographic mappings. Improvements of topographic mapping were derived by means of SOM architecture. Perceptive areas are to be increased, and accounting for density of mapped data is futile. The obtained method *Emergent ACC* does not produce dense clusters any more but uniformly distributed, SOM-like projections. Due to that, clusters are to be retrieved using U-Map technology. As predicted by our theory, an empirical evaluation showed on critical clustering problems that disregarding the density of mapped data improves the quality of topographic mapping despite of unfavorable settings.

References

1. C. Aranha, H. Iba, The effect of using evolutionary algorithms on ant clustering techniques. In: The Long Pham and Hai Khoi Le and Xuan Hoai Nguyen (editors). *Proceedings of the Third Asian-Pacific workshop on Genetic Programming.* pages 24–34, Military Technical Academy, Hanoi, VietNam, 2006.
2. R. A. Fisher, The use of multiple measurements in taxonomic problems, Annals of Eugenics, 7, Part II, pages 179–188, Cambridge University Press, 1936.
3. G. J. Goodhill, T. J. Sejnowski, Quantifying neighbourhood preservation in topographic mappings, In: *Proc. 3rd Joint Symposium on Neural Computation*, California Institute of Technology, 1996.
4. J. Handl, J. Knowles, M. Dorigo, Ant-Based Clustering and Topographic Mapping, *Artificial Life 12(1)*, MIT Press, Cambridge, MA, USA, 2006.
5. K. Nybo, J. Venna, S. Kaski, The self-organizing map as a visual neighbor retrieval method, In: Proc. of the *Sixth Int. Workshop on Self-Organizing Maps (WSOM 2007)*, Bielefeld, 2007.
6. T. Kohonen, Self-Organizing Maps, Springer Series in Information Sciences, Vol. 30, Springer, Berlin, Heidelberg, New York, 1995, 1997, 2001,
7. E. Lumer, B. Faieta, Diversity and adaption in populations of clustering ants, In Proceedings of the *Third International Conference on Simulation of Adaptive Behaviour: From Animals to Animats 3*, pages 501-508, MIT Press, Cambridge, MA, 1994.
8. S. C. Tan, K. M. Ting, S. W. Teng, Reproducing the Results of Ant-Based Clustering Without Using Ants, *IEEE Congress on Evolutionary Computation*, 2006.
9. Fundamental Clustering Problem Suite, `http://www.uni-marburg.de/fb12/datenbionik/data`.
10. A. Ultsch, F. Mörchen, U-maps: topograpic visualization techniques for projections of high dimensional data, In: *Proc. 29th Annual Conference of the German Classification Society (GfKl 2006)*, Berlin, 2006.
11. A. Ultsch, L. Herrmann *Automatic Clustering with U*C*, Technical Report, Dept. of Mathematics and Computer Science, Philipps-University of Marburg, 2006.