Proceedings of the International Multiconference on
Computer Science and Information Technology, pp. 123 – 130

ISSN 1896-7094
© 2006 PIPS

Documents clustering method based on Ants Algorithms

Łukasz Machnik[1]

[1]Departament of Computer Science, Warsaw University of Technology,
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland

Abstract. Ants Algorithms, particularly Ant Colony Optimization meta-heuristic, are universal, flexible and give possibility to scale because they are based on multi agent cooperation. The increase of demand for effective methods of big document collections management is sufficient stimulus to place the research on the new application of ant based systems in the area of text document processing. In this publication author presents the implementation of that technique in the documents clustering area – new documents clustering method. The aim of this document is to present the details of the ACO documents clustering method and detail results of experiments.

1 Ant Colony Optimization

One of the topics that was deeply explored in the past by ethnologist was the understanding of mechanism how almost blind animals were able to find the shortest way from nest to a food. Comprehension of the way to achieve this task by nature was the first step to implement that solution in algorithms area. Main inspiration to create ACO meta-heuristic were researches and experiments done by Goss and Deneubourg [1].

Ants (*Linepithaema humile*) are the insects that live in the community called colony. The primary goal of ants is the survival of the whole colony. A single specimen is not essential, only bigger community may efficiently cooperate.

Ants possess the ability of such efficient cooperation. It bases on work of many creatures, who evaluate one solution as a colony of cooperative agents. Individuals do not communicate directly. Each ant creates its own solution that contributes to the whole colony's solution [3]. The ability to find the shortest way between the source of the food and the ant-heel is a very important and interesting behavior of the ant colony. It has been observed that ants use the specific substance called pheromone to mark the route they have already gone through. When the first ant randomly chooses one route it leaves the specific amount of pheromone, which gradually evaporates. Next ants which are looking for the way, will, with greater probability, choose the route where they feel more pheromone and after that they leave their own pheromone there. This process is autocatalic – the more ants choose a specific way, the more attractive it stays for the others. Above, information bases mainly on Marco Dorigo publications. He is the one who most of all contributed to develop the research in the ant systems area. His publications are the biggest repository of ACO information [2] [3].

2 ACO-based clustering method

Noticed analogy between finding the shortest way by ants and finding documents most alike (the shortest way between documents), and in addition ability to use agents who construct their individual solutions as an element of the general solution, became the stimulus to begin research on using the ant based algorithms in the documents clustering process [4].

2.1 Adopting the ACO concepts to documents clustering task

For the needs of building an effective method of classifying the documents, it is necessary to make a choice of possible modification and adjusting of the concepts specific to real ants, so as could be effectively used to solve the problems connected with text mining.

- A colony of co-operating individual specimen:

Artificial ants build a solution by moving along the graph of a problem, from one document to the other. During each iteration *m* number of ants constructs a solution in *n* number of steps, using a probabilistic law of making a decision. In practice, when visiting a specific document *i* ant chooses the next document *j* to move to, a pair (*i*, *j*) is added to the solution constructed at the moment. This step is repeated until the ant builds a complete solution for the specific iteration. Considering the fact that this version of the algorithm is serial, after each ant finds a solution in a specific iteration process of leaving of certain amount of pheromone associated with a pair of documents follows. After that the ant dies. Yet new ants appear in her place, whose goal is to find a solution in the following iteration, leave a pheromone and die. The pattern repeats until gaining the best result, or until performing a specific amount of iterations.

- A pheromone trace and its force to influence:

From available variants of leaving pheromone on the path, the author chose a partial variant. The ants leave a pheromone in a specific amount which equals a quotient of a constant and a length of a found path. In addition the decay of the pheromone follows after constructing of all partial solutions – the sum of distances between all of visited documents. Communication pheromone path is being changed while finding a solution to a problem just to show the experience gained by ants while solving the problem.

- Finding the shortest path:

Co-ordinate describing the location of the specific document in space will be a vector representing the frequency of words occurring in the document. To describe the distance between the documents a simple measure in multidimensional space will be used – cosine distance. Finding of the shortest path will be represented by finding such sequence of passing from one document to the other, that the sum of the reverse of cosine distances between following elements of examined set would be smaller. The using of the reverse of cosine distance is necessary because the increase of cosine distance evidences on bigger similarity between documents.

- Accidental movement of individual ants in the beginning phase of finding the path:

Maintaining of this conditional is necessary because in the beginning phase of algorithm action the ants are not able to use the experience of their predecessors. The pheromone trace between individual documents is equal to selected constant value. Such situation forces fully accidental choice of the documents in the beginning phase of finding the path.

- Artificial ants live in the artificial, discreet world and can move only from one to the other specific position – states of the discreet world:

The set of states between which agents can move will be defined as a set of vectors representing the individual documents. As we assumed earlier, each document will be represented by a vector based on frequency of appearance of the specific worlds in examined text.

- The amount of the pheromone left by the artificial ant is connected with quality function of so far achieved solution:

The amount of the pheromone left by ants is proportional to the quality of the solution they find: the shorter is the distance between the documents the bigger is the amount of the pheromone left on the pairs of the documents – documents used to create the solution. The issue that still cannot be forgotten is requirement to evaporate the pheromone. It is also necessary to exclude the stagnation phenomenon, which means choosing the same route by all ants too early.

- Past states memory:

The artificial ants are equipped with the memory of passed states, which is supposed to prevent the multiple location of one ant in the same position (it is necessary because there is a possibility/danger that ants could fall into cycles, which could make finishing the building of the solution impossible)

2.2 Details of processing

The method of document clustering which is introduced here, is based on artificial ant system [5, 6]. Application of such solution will be used as a method of finding the shortest path between the documents, which is the goal of the first phase (trial phase) of considered method. The second phase (dividing phase) will have a task to actually separate a group of documents alike.

The aim of trial phase is to find the shortest path connecting every document in the set using ACO algorithm [2, 3]. That is equivalent to building a graph, whose nodes would make up a set of analyzed documents. The probability of choosing next document *j* by ant *k* occupying document *i* is calculated by following function (1).

$$p_{ij}^{k}(t) = \frac{[\tau_{ij}(t)]^{\alpha} * [s_{ij}]^{\beta}}{\sum_{k \in Z_k} [\tau_{ik}(t)]^{\alpha} * [s_{ik}]^{\beta}} \quad (1)$$

In the above formula, Z_k represents list of documents not visited by ant k, $\tau_{ij}(t)$ represents the amount of pheromone trail between documents i,j, α is intensity of pheromone trail parameter, β is visibility of documents parameter, however s_{ij} is cosine distance between documents i and j. After ants complete their trace the pheromone trail is evaporated and new amount of pheromone is left between every pair of documents. The amount of pheromone that is left by the ants is dependent on quality the constructed solution (length of the path). In practice, adding the new portion of pheromone to trail and its evaporating is implemented by formula presented below. This formula (2) is adapted to every pair of documents (i, j).

$$\tau_{ij}(t) \leftarrow (1 - \rho) * \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (2)$$

In the above formula, $\rho \in (0,1]$ stands for the pheromone trail decay coefficient, while $\Delta\tau_{ij}(t)$ is an increment of pheromone between documents (i,j). Below the dependence (3) that controls the amount of pheromone left by ant k between pair of documents (i,j) is presented.

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} n / L_k(t) & \text{, for } (i,j) \in T^k(t) \\ 0 & \text{, for } (i,j) \in T^k(t) \end{cases} \quad (3)$$

In the above formula, $T^k(t)$ means a set of document pairs that belong to path constructed by ant k, $L_k(t)$ is length of path constructed by ant k, while n is the amount of all documents. Finding the shortest path connecting every document in the set will be equivalent to building a graph, which nodes would make up a set of analyzed documents. Documents alike would be neighboring nodes in the graph, considering that the rank of the individual nodes will fulfill the condition of being smaller or equal to 2, which means that in the final solution one of the documents would be connected to only two others (similar) – each document in the designed solution would appear only ones. Gaining of such solution would mean the end of the first phase, known as *preparing*.

In the following stage of the process it is necessary to separate a group of documents alike in a sequence obtained in the first phase. The separation of groups is obtained by appropriate processing of the sequence of documents (the shortest path) received in the preparing phase. Following individual steps of that process are described. The vector that represent the first document in sequence is recognized as centroid μ of the first group that is separated. In the next step we calculate the sum of all elements (positions) of the centroid vector. After that we calculate the cosine distance between centroid vector μ and vector D that represent the next element of documents sequence. Next, we check the condition (4). If it is true, then the considered element permanently becomes the member of first group. We recalculate the value of centroid and try to extend this group by adding the next element from sequence.

$$\delta * \sum_{k=1}^{n} t_{\mu k} < \cos(\mu, D) \quad (4)$$

The δ parameter is called attachment coefficient and its range is (0,1]. However, if the condition is false, then the separation of first group is finished and the separation of the next (second) group begins. Vector of considered document that couldn't be added to the first group becomes initial centroid of the new group. The whole process is repeated from the beginning. Processing is finished when whole sequence of documents is done.

2.3 Variants of method

The amount of separated groups depends precisely on attachment coefficient. When we use a big value (close to 1) of δ parameter as a result of processing we received a big amount of groups with high degree of cohesion. The decrease of δ value causes receiving smaller amount of groups with less cohesion. In connection with above conclusion there is a possibility to propose two variants of considered method [7, 8].

The first variant called by author – single pass, is based on very precise execution of the trial phase—a lot of ants. The duration of the first phase increases, however this activity permits to accept smaller value of attachment coefficient during dividing phase and finishing processing after single pass of algorithm—single trial phase and single dividing phase.

The clustering method that uses the single pass variant is the example of non-hierarchical clustering method. The main advantage of that method is that operator does not have to set the expected number of clusters at the beginning of processing. Results received in this variant are less precise than results from second variant, however the time of processing is much shorter than time of a second proposed variant. This type of considered method can also act as trial phase for other clustering algorithms. The example can be separations of centroids for K-means method.

Second variant called by author—periodic, differs a little bit from variant proposed earlier. It assumes periodic processing of both phases: trial and dividing. In, every iteration of dividing phase the small numbers of neighbors are connected into small groups. The value of attachment coefficient is very high in initial phases and is gradually decreased to allow group creation in next iterations. Each group during processing is represented by centroid. After group creation and centroids calculations the next iteration can be started—finding the shortest path between centroids and documents. The whole process is finished when all documents are connected as a single cluster or when the stop criterion is reached.

This variant is an example of agglomerative hierarchical clustering method that begins from a set of individual elements which are then connected to the most similar elements forming bigger and bigger clusters. The result of hierarchical technique processing is creating nested sequence of partitions. The main partition is placed at the top of hierarchy. It includes all elements from considered collections. The base of hierarchy creates individual elements. Every middle level can be represented as combination of clusters that are at the lower level in hierarchy. User can choose any level that satisfied him as solution.

3 Results of the experiments

3.1 Experimental system

The experiments that are presented in this publication were executed using the KLASTERYZATOR_ACO document clustering system. That system was implemented by ANSI C++. During the researches two collections of documents were used. The first collection was *McCallum newsgroups* that contained documents from twenty forums from the USENET network. Documents were chosen random. The second set was created by documents from *Reuters-21578* repository. The documents from that collection were representatives of the biggest thematic groups.

3.2 Clustering algorithms

In [4] the most popular clustering methods were presented. In the experimental system three of them were implemented: K-means (non-hierarchical), single link method (hierarchical) and average link method (hierarchical). These methods were chosen because they are popular and commonly implemented in practice and that is the reason why they were good candidates to comparison.

3.3 Results evaluation

The results of experiments were evaluated using internal quality measure – intra-cluster variance. This method was chosen for two reasons. First, the application of ant-based clustering in real clustering task required the evaluation of the obtained results without knowledge of the correct solution. Second, these functions provided additional information about structure of the obtained solutions and can therefore help to understand and analyze results. Additionally it is important to remember, that the amount of groups that we received from processing was also cluster evaluation measure. Method that is presented by author has unique properties to control the trend of cluster creation number.

3.4 Number of groups

ACO clustering method is characterized by the ability to identify the number of clusters in the collection that is processed. The majority of popular methods (K-means, single link method, average link method) require the input parameter that constitutes the number of outcome groups. This kind of behavior requires the ‘a priori’

knowledge of collection that will be processed or interaction with another algorithm that has the preparatory function. Such interaction is very often the source oh many problems. Also, clustering algorithms that are able to identify the number of cluster automatically have many limits. Incorrect choice of number and value of the centroids can have dramatic impact on final solution. This kind of situation we can observe on Fig. 4 and Fig. 5.

On the other hand, impossibility to directly define the number of resultant clusters can be recognized as a disadvantage. There are many application in which user requires the ability to define that value by himself. Clustering method presented in this publication beside identifying the number of resultant cluster, also delivers tool to manipulate the trend of cluster identification number. The role of this tool attends a attachment coefficient δ. Fig. 1 describes flexibility in manipulating the number of clusters using δ parameter.

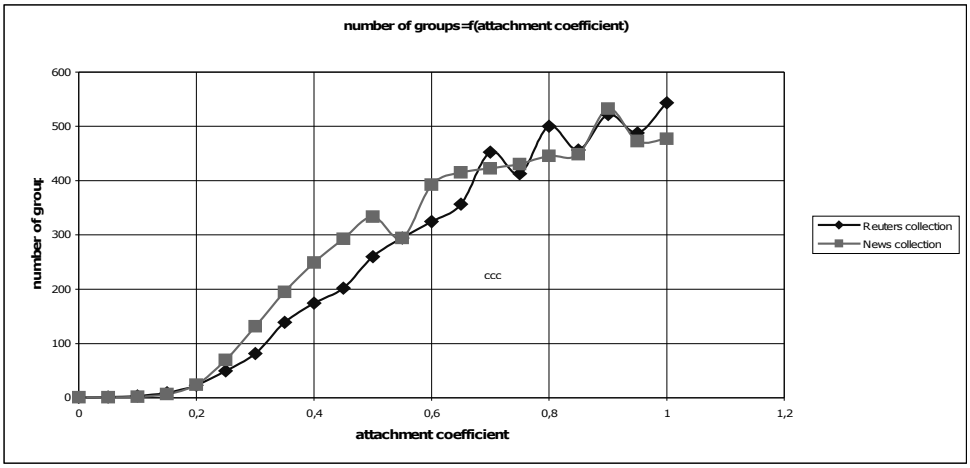

Fig. 1 . The influence of attachment coefficient on the number of groups

3.5 Sizes of the groups

Fig. 2 and Fig. 3 present the way of forming the sizes of the groups for methods considered during experiments. The analysis of the results shows that the method proposed by author is characterized by the proportional distribution of elements among clusters. Also, the trend of creating one superior group can be noticed. The results of ACO processing are quite similar to results for K-means processing. It is quite important to observe that the ACO clustering method has a tendency to limit the effect of creating one superior group instead of creating more balance clusters with high degree of cohesion (Fig. 4 and Fig. 5). The single link method and the average link method give much worse results then the first two methods. They have a tendency to create one predominant group.

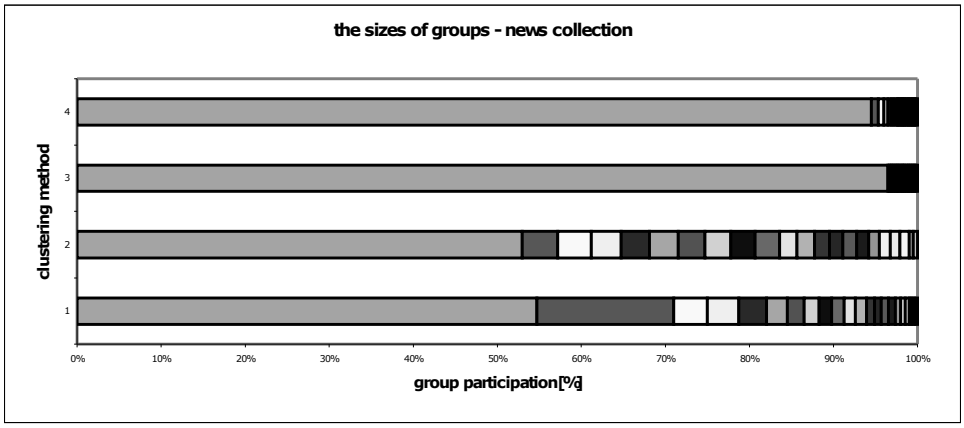

Fig. 2 . The sizes of groups – news collection (1) ACO method, (2) K-means method, (3) single link method, (4) average link method

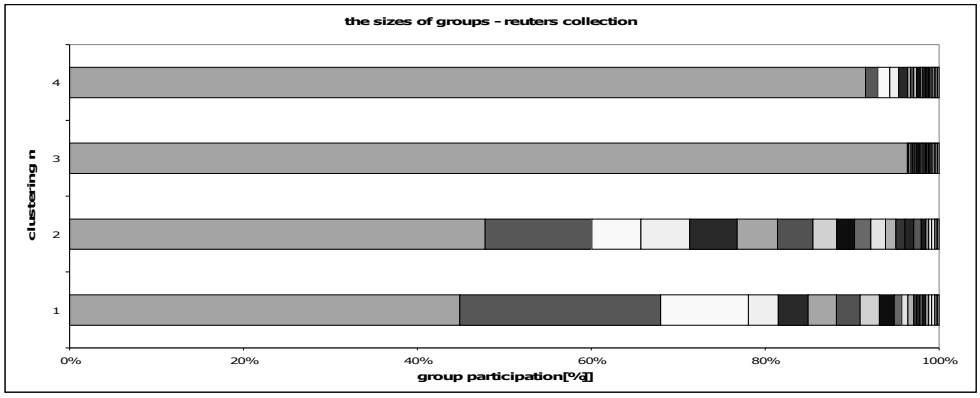

Fig. 3 . The sizes of groups – reuters collection (1) ACO method, (2) K-means method, (3) single link method, (4) average link method

3.6 Quality

The quality of results were evaluated using internal quality measure – intra-cluster variance. The results of experiments that are presented on below figures show that the quality of ACO clustering is very high for both texts collections. The results obtained for different amounts of groups demonstrate the dominance of ACO method over others tested methods. The quality stability of results for ACO clustering should be noticed.

The results generated by single link method and average link method are quite similar but the difference between them and other results is significant. For K-means method we received good results for small amount of groups but the quality of processing is getting worse at higher number of groups. For K-means method we can also observe the dramatical deterioration of results quality at very high number of groups. This effect is caused by random selection of centroids and can be limited by using special algorithms for centroids generation.

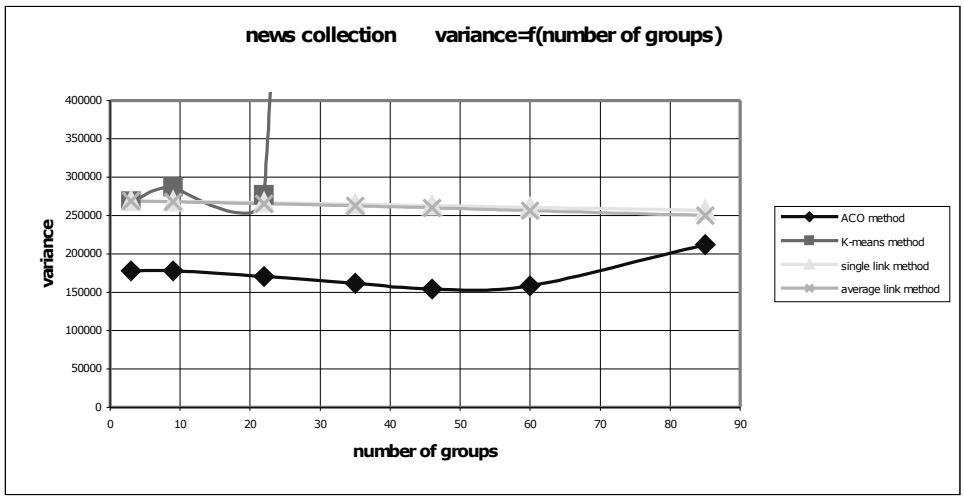

Fig. 4 . The value of variance – news collection

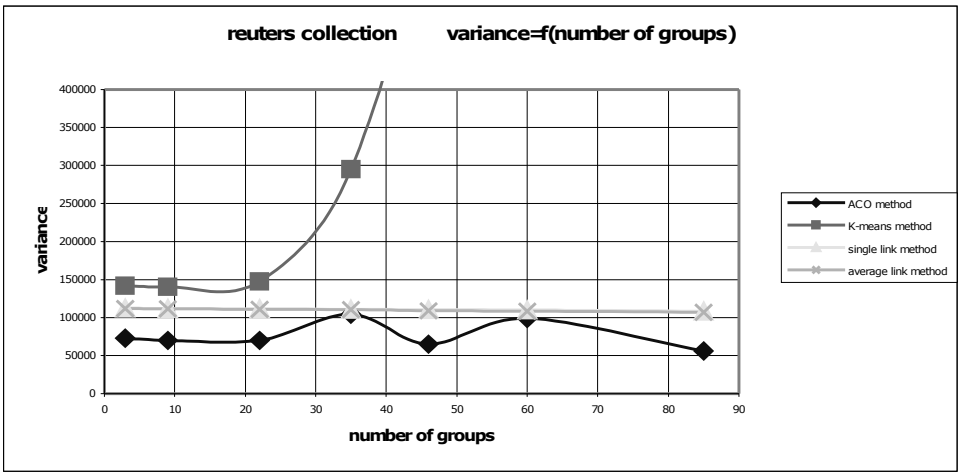

Fig. 5 . The value of variance – reuters collection

3.7 Time

The experiments show that for small collections of documents the ACO method is much slower than other tested methods. However, it should be noticed that the bigger size of collection presented method tends to be ahead of the competitors. Only the single link method is able to return results faster than the ACO method but at the same time the quality and group distribution is much worse.

Fig. 6 presents time of processing for documents collections with different sizes. The time of processing depends on the number of resultant groups. The results with the best quality and good speed are obtained only by method proposed by author. It is also important to mention that the fastest results are generated using quite small group of ants. It is connected with loss of quality but even so the results are still better than results obtained by other methods.

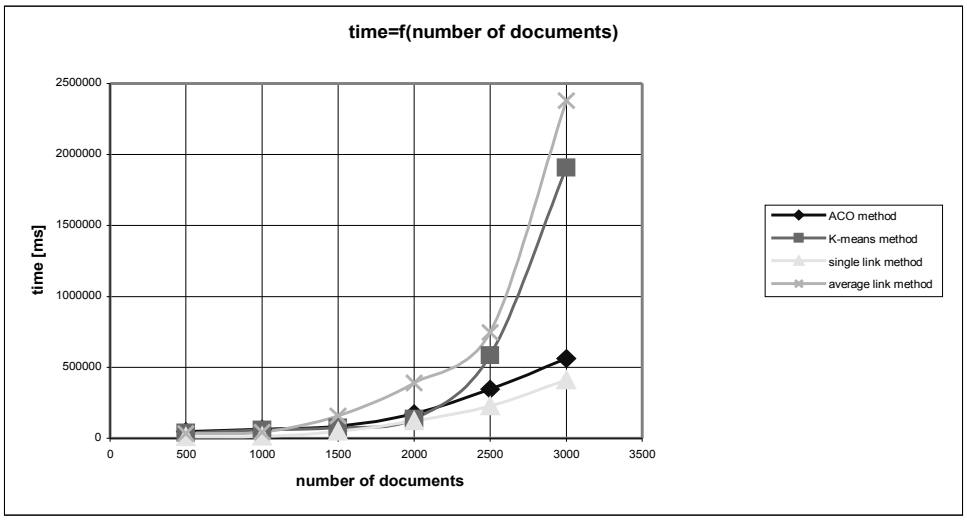

Fig. 6 . Relation between time and number of processed documents

4 Summary

The experiments confirm an argument that the ant algorithms can be successfully implemented in text documents processing. The attempt of creating valuable clustering method based on ACO meta-heuristic was success. This proves the universal nature and flexibility of ACO meta-heuristic. Tests performed in test environment proved the utility and advantages of method created by author of this publication. The results obtained during experiments are characterized by good quality, speed for big collections of documents and flexibility in determining the number of resultants groups. It seems that there is a possibility to increase the performance of calculations by implementing a parallelization in processing. This topic will be considered in following research of author.

References

1. Deneubourg J.-L., Pasteels J. M., Verhaeghe J. C.: *Probabilistic behaviour in Ants: a strategy of errors*, Journal of Theoretical Biology, 259-271, 1983.
2. Dorigo M.: *Optimization, Learning and Natura Algorithms,* (In Italia), PhD thesis Dipartimento di Elettronica e Informazione, Politecnico di Milano, IT, 1992.
3. Dorigo M., Maniezzo V., Colorni A.: *The ant systems: optimization by colony of cooperating agents*, IEEE Transactions on Systems, Man, and Cybernetics-PartB, 1996.
4. Machnik Ł.: *Documents Clustering Techniques,* IBIZA 2004, Annales UMCS Informatica 2004, Poland, 2004.
5. Deneubourg J.-L., Goss S., Franks N., Sendova-Franks A., Detrain C., Chretien L.: *The dynamics of collective sorting: Robot-like ants and ant-like robots*, First International Conference on Simulation of Adaptive Behaviour: From Animals to Animats 1, 356-365, MIT Press, MA, 1991.
6. Lumer E., Faieta B.: *Diversity and adaptation in populations of clustering ants*, Third International Conference on Simulation of Adaptive Behaviour: From Animals to Animats 3, 501-508, MIT Press, 1994.
7. Machnik Ł.: *Ants in text documents clustering*, Proceedings of the International Conference on Systems, Computing Sciences and Software Engineering (SCSS 2005), 2005.
8. Machnik Ł.: *ACO-based document clustering method*, Konferencja Informatyka – Badania i Zastosowania, Kazimierz Dolny 2005, Annales UMCS Informatica, Poland, 2005.