International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:5, 2007

An Ant-based Clustering System for Knowledge Discovery in DNA Chip Analysis Data

Minsoo Lee, Yun-mi Kim, Yearn Jeong Kim, Yoon-kyung Lee, and Hyejung Yoon

***Abstract*—**Biological data has several characteristics that strongly differentiate it from typical business data. It is much more complex, usually large in size, and continuously changes. Until recently business data has been the main target for discovering trends, patterns or future expectations. However, with the recent rise in biotechnology, the powerful technology that was used for analyzing business data is now being applied to biological data. With the advanced technology at hand, the main trend in biological research is rapidly changing from structural DNA analysis to understanding cellular functions of the DNA sequences. DNA chips are now being used to perform experiments and DNA analysis processes are being used by researchers. Clustering is one of the important processes used for grouping together similar entities. There are many clustering algorithms such as hierarchical clustering, self-organizing maps, K-means clustering and so on. In this paper, we propose a clustering algorithm that imitates the ecosystem taking into account the features of biological data. We implemented the system using an Ant-Colony clustering algorithm. The system decides the number of clusters automatically. The system processes the input biological data, runs the Ant-Colony algorithm, draws the Topic Map, assigns clusters to the genes and displays the output. We tested the algorithm with a test data of 100 to1000 genes and 24 samples and show promising results for applying this algorithm to clustering DNA chip data.

***Keywords*—**Ant Colony System, Biological data, Clustering, DNA chip.

I. Introduction

BIOLOGICAL science is being revolutionized by the availability of abundant information regarding complete genome sequences for many different organisms. Organisms are complex and genomes can be immense, and thus new and powerful technologies are being developed to analyze large numbers of genes and proteins as a complement to traditional methodologies that study a small number at a time. With the advanced technology at hand, the main trend in biological research is rapidly changing from structural DNA analysis to understanding cellular function of the DNA sequences. The recently developed DNA chips, in other words DNA microarrays, have emerged as a prime candidate for such high performance analysis methods [1][2][3].

In order to analyze DNA chips, a DNA analysis process is carried out. The steps that must be followed are: performing the experiments on DNA chips, scanning the results of the experiments, carrying out quality control and normalization in order to filter out the data, performing feature selection that selects specific parts, doing clustering and classification, and storing the result into a bio data warehouse to provide integrated analysis results to users [4].

One of such important analysis processes is clustering, which is the process of grouping together similar entities. There are many clustering algorithms like hierarchical clustering, self-organizing maps, K-means clustering and so on. But in this paper, we propose a clustering algorithm that imitates the ecosystem taking into account the features of biological data that are complex, large in amount, and are variable. Algorithms that imitate the ecosystem are generally used to solve very complex problems. The ecosystem is very complex and there exist many living things within it. In an ecosystem, each individual is assumed to be showing optimal behavior. The algorithms are designed according to such individuals' movement. They reflect biodiversity and living things' complexity. The ecosystem algorithms have the following benefits. First, they provide a solution based on a solid statistical model. In other words, they do not rely on a single solution but have more flexibility due to the fact that they are based on a statistical method that considers several solutions at one time. This allows the algorithm to find good solutions that may be missed by other algorithms. Second, algorithms that imitate the ecosystem make use of the interaction among the possible solutions. For example, the genetic algorithm allows solutions to pair together and create new solutions. Third, these algorithms allow exceptions. Therefore, solutions that are not typical but are actually better solutions can be found. Because of these reasons, the algorithms are suitable to solve the complex problem of analyzing mass amounts of complex biological data.

In this paper, we implemented a DNA chip data clustering system using the Ant-Colony clustering algorithm. The Ant Colony Optimization algorithm (ACO) uses a probabilistic

Manuscript received June 15, 2006. This work was supported in part by the Korean Ministry of Commerce, Industry and Energy, and also in part by the second stage of the BK21 program of the Ministry of Education and Human Resources Development.

Minsoo Lee is with the Dept of Computer Science and Engineering, Ewha Womans University, 11-1 Daehyun-Dong, Seodaemoon-Ku, Seoul, Korea 120-750 (e-mail: mlee@ ewha.ac.kr).

Yun-mi Kim is with the Dept of Computer Science and Engineering, Ewha Womans University, 11-1 Daehyun-Dong, Seodaemoon-Ku, Seoul, Korea 120-750 (e-mail: cherish11@ewhain.net).

Yearn Jeong Kim is with the Dept of Computer Science and Engineering, Ewha Womans University, 11-1 Daehyun-Dong, Seodaemoon-Ku, Seoul, Korea 120-750 (e-mail: inverno@ewhain.net).

Yoon-kyung Lee is with the Dept of Computer Science and Engineering, Ewha Womans University, 11-1 Daehyun-Dong, Seodaemoon-Ku, Seoul, Korea 120-750 (e-mail: polyandry@hanmail.net).

Hyejung Yoon is with the Dept of Computer Science and Engineering, Ewha Womans University, 11-1 Daehyun-Dong, Seodaemoon-Ku, Seoul, Korea 120-750 (e-mail: auroree@ewhain.net).

technique for solving computational problems which can be reduced to finding close to optimal paths through graphs. They are inspired by the behavior of ants in finding paths from the colony to food [5].

The developed system works as follows. The first step loads the input data, and the second and third steps create the Topic Map while assigning clusters. In order to assign clusters, it is necessary to store the node's last position, compute the distance between the nodes, assign clusters and merge clusters. Our system can dynamically decide the number of clusters. The problem of deciding the number of clusters is an important issue in the field of data mining. The last step is displaying the results.

The organization of this paper is as follows. Section II provides a survey of related work about clustering techniques and algorithms that imitate the ecosystem. Section III gives an overview of the DNA chip analysis process. Section IV explains the Ant-based Clustering system for DNA chip data analysis. Section V describes the implementation and experimental results of the Ant-based Clustering system. Section VI gives the conclusion and future work.

II. Related Research

The objective of our work is to implement a clustering system for analyzing DNA chip data. Clustering is the process of grouping together similar entities. Clustering is appropriate when there is no a priori knowledge about the data. In such circumstances, the only possible approach is to study the similarity between different samples or experiments. There are many existing clustering algorithms: hierarchical clustering, self-organizing maps, K-means clustering and many others. Hierarchical clustering approaches calculate the distance between individual data points and then group together those that are close. The distances between the groups themselves can also be computed and used to create groups of groups. These can iteratively be organized into a 'tree' in which the closest groups constitute nearby 'branches' – far from groups that are less similar [6]. Hierarchical clustering strategies are easy to implement but suffer because the decision about where to create branches and in what order the branches should be presented can be arbitrary. K-means clustering requires a parameter, k, the number of expected clusters, and the initial cluster centers are selected randomly. In each iteration of the algorithm, all of the profiles are assigned to the cluster whose center they are nearest to (using the distance metric), and then the cluster center is recalculated based on the profiles within the cluster [7]. Instead of simply partitioning data into disjoint clusters, self-organizing maps organize the clusters into a 'map' where similar clusters are close to each other [8]. The number and topological configuration of the clusters are pre-specified. The computational details are similar to K-means clustering except that cluster centers are recalculated at each iteration using both the profiles within the cluster as well as the profiles in adjacent clusters. Over many iterations the clusters conform to the pre-specified topology [9].

Because of the characteristics of the DNA chip data such as the high complexity, large in amount, and variable properties, we propose a clustering algorithm which uses an algorithm that imitates the ecosystem. There currently are many algorithms that imitate the ecosystem. The Genetic algorithm, Neural Network algorithm, Particle Swarm algorithm and Ant Colony algorithm are the most popular algorithms. The genetic algorithm (GA) is a search technique used in computer science to find approximate solutions to optimization and search problems. Specifically it falls into the category of local search techniques and is therefore generally an incomplete search. Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination)[10]. Neural Networks is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. NNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons [11]. Particle Swarm Optimization (PSO) is a recently proposed algorithm by James Kennedy and R. C. Eberhart in 1995, motivated by social behavior of organisms such as bird flocking and fish schooling. PSO as an optimization tool, provides a population-based search procedure in which individuals called particles change their position (state) with time. In a PSO system, particles fly around in a multidimensional search space. During flight, each particle adjusts its position according to its own experience, and according to the experience of a neighboring particle, making use of the best position encountered by itself and its neighbor [12]. The Ant Colony Optimization algorithm (ACO) that our system uses is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs [13].

There are some work on clustering algorithms based on the Ant Colony algorithm. Yuqing et al. proposed the K-means clustering algorithm based on Ant Colony [14]. And Xiao at al. proposed gene clustering using self-organizing maps and particle swarm optimization [15]. Handl et al. proposed Ant-based clustering [16]. Our system extends this work to adapt ACO to DNA chip data analysis.

III. DNA Chip Analysis Process

DNA chips, in other words microarrays, are a tool for gene expression analysis. The DNA chip consists of the probe which is a single strand DNA printed on the solid substrate. The types of the chip or the name of the chip depends on the method of the chip fabrication. The idea behind the bio-chip is that the DNA in the solution that contains sequences complementary to the sequences of the DNA deposited on the surface of the array will

be hybridized to those complementary sequences. Usually, this interrogation is done by washing the array with a solution containing ssDNA, called a target.

To analyze the DNA chip, an overview of the steps that must be gone through are as follows. First, the researchers perform experiments using DNA chips. Afterwards the DNA analysis process is carried out by first scanning the results of the experiments. Next, quality control and normalization processes are carried out to revise errors. During this step, a lot of data is filtered and the quantity of the test data decreases. And then feature selection that selects specific parts is performed. The next step is the data mining work such as classification and clustering. Finally, the results are stored in the biological data warehouse and the warehouse system provides analysis results of integrated biological information to users.

The key to interpreting DNA chip data is in the DNA material that is used to hybridize on the array. Since the target is labeled with a fluorescent dye, a radioactive element, the hybridization spot can be detected and quantified easily.

After the hybridization, the Scanning work of the results of the experiment is continued. The scanner scans the spots and converts quantity to numeric values namely expression values. This process is called image processing. The next process, Quality control and normalization, are performed in order to get rid of unnecessary data that can influence the whole expression values. In other words, through quality control and normalization, the data are filtered and the data that are meaningful and significant only remain as a result. And it also adjusts for any bias which arises from variations in the DNA chip technology rather than from biological differences between the RNA samples of the printed genes.

In order to discover meaningful information from the raw data obtained so far, data mining techniques are used. The most well-known techniques are clustering and classification techniques. First, clustering is performed. Clustering is similar to classification in that data is being grouped. However, unlike classification, the groups are not predefined. Instead, the grouping is accomplished by finding similarities between data according to characteristics found in the actual data. The groups are called clusters. Clustering is appropriate when there is no a priori knowledge about the data. In such circumstances, the only possible approach is to study the similarity between different samples or experiments. In fact, clustering is the process of grouping together similar entities. The algorithm will treat all inputs as a set of n numbers or an n-dimensional vector. Similarity is measured in many different ways, and the final result of the clustering depends on the formula used.

The next data mining task performed is classification. For classification, the resulting data from Q.C. and normalization is used as input to create the classification rules. The rules are applied to the set of test data and a prediction is made on which class the data point should belong to considering the accuracy of the rule.

Finally, the results are stored in the biological data warehouse. A data warehouse is a subject oriented, integrated, and nonvolatile, time variant collection of data that can support complex analysis and decision making processes. The warehouse system provides analysis of integrated biological information to users [4]. The whole process is shown in Fig. 1.

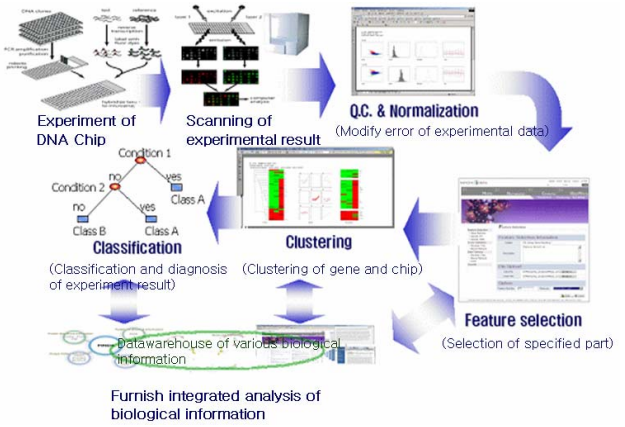


Fig. 1 DNA chip analysis process

IV. Architecture of Ant-Based Clustering System

The Ant-based Clustering System for DNA chip data analysis processes the data in four stages. The first stage loads the input data and the second stage creates a Topic Map while executing the Ant algorithm. The Topic Map is a map that displays the movement of the ants. The third stage assigns the clusters. And the last stage displays the results. In this section, we show the detail stages of the Ant-based Clustering System.

A. System Flow of Ant-based Clustering System

In our system, the DNA chip data is stored in the database after going through the quality control and normalization. During the input stage, the system connects to the database and brings portions of the normalized biological data into the system memory. The next task is to run the Ant Colony algorithm and draw the Topic Map. The Topic Map shows the ants' movements that perform the clustering. While drawing the Topic Map, the system uses the Ant Colony algorithm underneath. The ant colony optimization algorithm (ACO) is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs. They are inspired by the behavior of ants in finding paths from the colony to food [13]. Each ant moves a biological data point and the ants' movements are displayed via the Topic Map. The next step is to assign cluster numbers to each gene. For this work, the system uses a distance matrix which stores the values that indicate the distance among the genes. The system decides the cluster's number for each gene with a threshold value which the user decides. This threshold represents the proximity limit for identifying genes as the same cluster. The last step is displaying the results. The system displays information such as the number of genes, the number of samples, iteration count, and the cluster number for each gene and so on. The overall system flow is shown in Fig. 2.

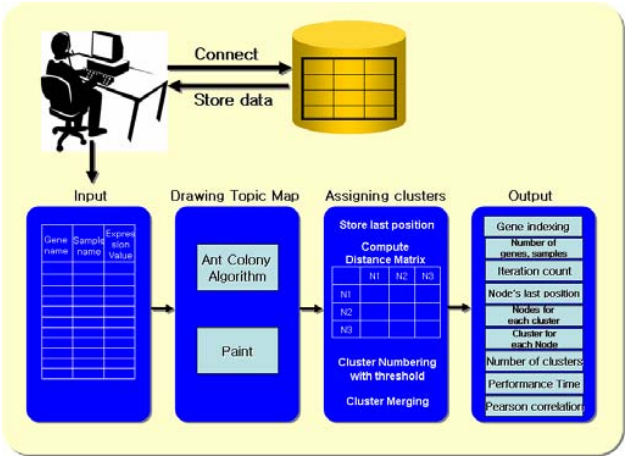


Fig. 2 System flow of Ant-based Clustering System

B. Data Input Process

The DNA chip data stored in the database is composed of many tables but our system only uses a table named TBDC_NORM_RESULT. The table consists of three attributes. The attributes are gene data, sample data, and expression values. Fig. 3 shows the TBDC_NORM_RESULT table.

Gene Name	Sample Name	Expression Value
293842	N00287	5.11359232185826
293842	N00288	-1.3389279170472
…	…	…
293843	N00287	1.82213569753198
293843	N00288	2.70243282124646
…	…	…

Fig. 3 TBDC_NORM_RESULT table in DB

The system connects to the database and loads portions of the values in memory. And then the system divides the values into regular intervals and assigns different colors to the genes. The reason for this is to provide a better view of the data in the Topic Map for the user.

C. Ant Colony Clustering and Drawing the Topic Map

While drawing the Topic Map, the system runs the Ant Colony Clustering algorithm. Ants create a large group consisting of the largest number of members in the group. Ants also have the longest history in creating colonies. Their behavior has been optimized for a long period of time and is suggested as a good way to imitate their behavior to solve a complex problem. Also, the algorithm is simple and can find the optimal solution using the characteristics of the ants. Ants scatter pheromone on the path that they follow. The larger amount of pheromone the path has, the more probability it has to be chosen. The amount of pheromone deposited on the path is incremented proportion to the quality of the candidate solution. Each path followed by an ant will be a candidate solution for a given problem. Each point of the Topic Map represents each ant. These ants randomly move in the map. Gene data items that are scattered within this map can be picked up, transported and dropped by the ants. The picking and dropping operations are biased by the similarity and density of gene data items within the ants' local neighborhood; ants are likely to pick up gene data items that are either isolated or surrounded by dissimilar ones; they tend to drop them in the vicinity of similar ones. In this way, a clustering of the elements on the map is obtained. One of the algorithm's special features is that it generates a clustering of a given set of data through the embedding of the high-dimensional data items on a two-dimensional grid; it has been said to perform both a vector quantization and a topographic mapping at the same time, much as do self-organizing feature maps (SOMs, [17]) [16]. The detail algorithm is described in the next paragraph.

Ant Colony Clustering algorithm
begin
INITIALIZATION PHASE
Randomly scatter gene data items on the grid
for each *j* in 1 to *#ants* **do**
i := random_select(*remaining_items*)
pick_up(ant(*j*), *i*)
g:= random_select(*remaining_empty_grid_locations*)
place_agent(ant(*j*), *g*)
end for
MAIN LOOP
for each *it_ctr* in 1 to *#iterations* **do**
j := random_select(*all_ants*)
step(ant(*j*), *stepsize*)
i := carried_item(ant(*j*))
drop := drop_item?(*f*(i)*)
if *drop* = TRUE **then**
while *pick* = FALSE **do**
i := random_select(*free_gene_data_items*)
pick := pick_item?(*f*(i)*)
end while
end if
end for
end

Fig. 4 Ant Colony Clustering Algorithm

The Ant Colony Clustering algorithm starts with an initialization phase, in which (i) all gene data items are randomly scattered on the Topic map; (ii) each ant randomly picks up one gene data item; and (iii) each ant is placed at a random position on the grid. Subsequently, the sorting phase starts: this is a simple loop, in which (i) one ant is randomly selected; (ii) the ant performs a step of a given step size (in a randomly determined direction) on the grid; and (iii) the ant (probabilistically) decides whether to drop its gene data item. In the case of a 'drop'-decision, the ant drops the gene at its current grid position (if this grid cell is not occupied by another gene data item), or in the immediate neighborhood of it (it locates a nearby free grid cell by means of a random search). It then immediately searches for a new gene data item to pick up. This is done using an index that stores the positions of all 'free' data items on the grid: the ant randomly selects one data item out of the index, proceeds to its position on the grid, and evaluates the neighborhood function f*(i), and (probabilistically) decides whether to pick up the gene data item. It continues this search until a successful picking operation occurs. Only then the loop is repeated with another

ant. For the picking and dropping decisions the following threshold formulae are used:

$$p^*_{pick}(i) = \begin{cases} 1.0 & \text{if } f^*(i) \leq 1.0 \\ \frac{1}{f^*(i)^2} & \text{else} \end{cases}$$

$$p^*_{drop}(i) = \begin{cases} 1.0 & \text{if } f^*(i) \geq 1.0 \\ f^*(i)^4 & \text{else,} \end{cases}$$

The algorithm is shown in Fig. 4. The movements of the ants are shown by the colored moving points.

D. Assigning Clusters

After drawing the Topic Map, the last position of nodes can be obtained. To assign a cluster number to each node, the system computes the distance between any two nodes using their last position and stores the information in the distance matrix. When the system computes the distance, it uses the concept of Euclidean distance. Euclidean distance is the straight line distance between two points. The system assigns a cluster number based on the distance calculation. The distance matrix is shown in Fig. 5.

	Node 1	Node 2	…	Node n
Node 1	0	0.3425673	…	38.231435
Node 2	0.3425673	0	…	38.563426
…	…	…	…	…
Node n	38.231435	38.563426	…	0

Fig. 5 The distance matrix

The process of assigning cluster numbers is divided into two phases. First, there is an assigning phase and then a merging phase for merging nearby clusters. During the assigning phase, the system initializes each node's cluster number to each node index. It iteratively chooses one point, and then compares the distances to other nodes. If the distances between the selected node and other nodes are smaller than a threshold, the Ant-based Clustering System assigns the same cluster number to those nodes. The threshold is decided by the user. This process is repeated until all nodes are compared with their neighboring nodes and are assigned a cluster number. For the merging phase, the system computes the distance between two clusters. The system chooses one node from each cluster. The selected nodes should have the shortest distance between the two clusters. If the distance between the selected nodes is smaller than a threshold, the system merges the two clusters. The threshold used in this phase is not the same threshold with the threshold of the assigning phase. This phase is repeated until all clusters are evaluated. Through this process, the system decides the number of clusters automatically. Our system can solve the problem of deciding the number of clusters which is an important issue in the data mining field.

E. Displaying Clustering Rresults

The final displayed results for the Ant-based Clustering System are the Topic Map and the cluster number of each node. The Topic Map is a user interface which shows the ants' movement. The node's cluster information includes nine kinds of detail information: node indices of the genes, the number of genes and the number of samples, the number of iterations, coordinates of the last positions of the nodes, the nodes which each cluster has, the cluster number to which each node belongs, the number of clusters, the time taken, and the Pearson correlation. The Topic Map is shown in Figure 6 and the results are shown in Fig. 7.

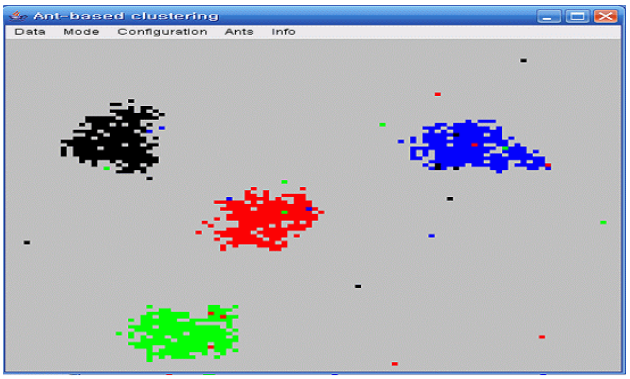


Fig. 6 Output : Topic Map

```
*** match gene name - node index ***
297784 : node 0
297912 : node 1
297990 : node 2
298000 : node 3
298143 : node 4
298165 : node 5
298174 : node 6
298200 : node 7
298276 : node 8
298316 : node 9
298331 : node 10
298367 : node 11
298428 : node 12
298459 : node 13
298479 : node 14
298518 : node 15
   ...

The number of Genes: 100
The number of experiments: 24
45.0
0
1
2
3
4
5
6
7
8
9
10
   ...

*** node의 last position ***
node 0 - x: 2  y: 29
node 1 - x: 93  y: 10
node 2 - x: 19  y: 81
node 3 - x: 13  y: 92
node 4 - x: 32  y: 74
node 5 - x: 15  y: 90
node 6 - x: 91  y: 11
node 7 - x: 86  y: 56
node 8 - x: 85  y: 55
node 9 - x: 96  y: 10
node 10 - x: 2  y: 93
node 11 - x: 47  y: 56
node 12 - x: 84  y: 53
node 13 - x: 89  y: 34
node 14 - x: 12  y: 92
node 15 - x: 63  y: 70
   ...

*** Cluster Matrix ***
cluster0 : node_num(12)  0 63 70 79 81 85 4 0 15 0 18 0
cluster1 : node_num(6)  1 6 9 22 28 71
cluster2 : node_num(21)  2 3 5 10 14 30 31 37 38 40 43 44 45 48 50 52 54 55 82 86 88
cluster7 : node_num(8)  7 8 12 19 25 29 33 49
cluster11 : node_num(4)  11 21 24 36
cluster13 : node_num(17)  13 35 42 46 51 56 58 59 61 64 72 73 76 78 94 95 97
cluster16 : node_num(5)  16 39 57 87 93
cluster17 : node_num(6)  17 20 32 41 53 80
cluster23 : node_num(4)  23 26 27 89
cluster34 : node_num(13)  34 60 65 66 68 74 75 83 84 91 92 96 99
cluster47 : node_num(7)  47 62 67 69 77 90 98

Number of Clusters : 11

Evaluation took 15391 milliseconds
Pearson correlation: 0.056111004024475185

*** Cluster ***
node 0 : cluster 0
node 1 : cluster 1
node 2 : cluster 2
node 3 : cluster 2
node 4 : cluster 0
node 5 : cluster 2
node 6 : cluster 1
node 7 : cluster 7
node 8 : cluster 7
node 9 : cluster 1
node 10 : cluster 2
   ...
```

Fig. 7 Output: The node information

V. Experimental Results

A. Setup

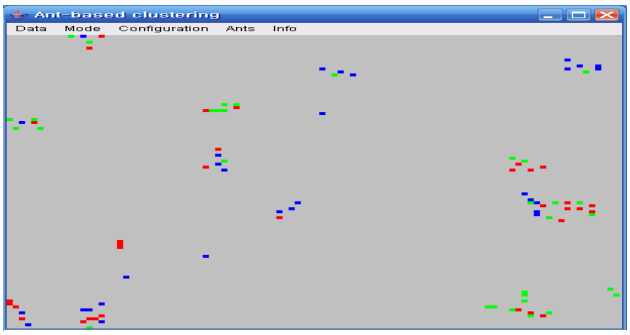
The Ant-based Clustering System has been implemented in Java. We used the Java 2 standard edition development kit version 1.5.0_07 as the development language. And we used the Eclipse SDK 3.2 as the IDE for development. The database server for this system is Oracle 9i. To easily manage the database, we also used Orange 3.1.

B. *Experimental Results*

First, we experimented with a test data set of 100 genes and 24 samples and then 1000 genes and 24 samples. We performed 100 iterations for the first data set and performed 1000 iterations for the second data set. The performance results are shown in Fig. 8 and the clustering results are shown in Fig. 9.

Genes	Samples	iteration	Time
100	24	100	1656ms
1000	24	1000	23656ms

Fig. 8 Execution time for clustering on the test data




```
*** Cluster Matrix ***
cluster0 : node_num(12)  0 63 70 79 81 85 4 0 15 0 18 0
cluster1 : node_num(6)  1 6 9 22 28 71
cluster2 : node_num(21)  2 3 5 10 14 30 31 37 38 40 43 44 45 48 50 52 54 55 82 86 88
cluster7 : node_num(8)  7 8 12 19 25 29 33 49
cluster11 : node_num(4)  11 21 24 36
cluster13 : node_num(17)  13 35 42 46 51 56 58 59 61 64 72 73 76 78 94 95 97
cluster16 : node_num(5)  16 39 57 87 93
cluster17 : node_num(6)  17 20 32 41 53 80
cluster23 : node_num(4)  23 26 27 89
cluster34 : node_num(13)  34 60 65 66 68 74 75 83 84 91 92 96 99
cluster47 : node_num(7)  47 62 67 69 77 90 98
```

Fig. 9 Clustering result of the first test data set

VI. Conclusion

In this paper, we proposed a clustering method that makes use of an algorithm that imitates the ecosystem. We implemented the system using the Ant Colony clustering algorithm. The system works in several steps such as loading the input data, running the Ant Colony algorithm while drawing the Topic Map, assigning clusters and displaying the results. We have tested the algorithm with data sets consisting of 100 genes and 24 samples and then 1000 genes and 24 samples. The results of the clustering are very promising and can also be visually verified. Future work includes performing more extensive experiments on different types of DNA chip data, and also work on areas of more optimization.

References

[1] DJ Lockhart, HL Dong, MC Byrne, MT Follettie, MV Gallo, MS Chee, M Mittmann, CW Wang, M Kobayashi, H Horton,EL Brown, Expression monitoring by hybridization to high-density oligonucleotide arrays, Nature Biotechnology, 14(13):1675-1680, 1996.

[2] JL DeRisi, VR Iver, PO Brown, Exploring the metabolic and genetic control of gene expression on a genomic scale, Science, 278(5338):680-686, 1997.

[3] C Debouck, PN Goodfellow, DNA microarrays in drug discovery and development", Nature Genetics, 21(1 suppl):48-50, 1999.

[4] David Bowtell, Joseph Sambrook, DNA Microarrays, CSHL Press, 2002

[5] WIKIPEDIA, http://en.wikipedia.org/wiki/Ant_colony_optimization

[6] Michael B. Eisen, Paul T. Spellman, Patrick O. Browndagger, and David Botstein, Cluster analysis and display of genome-wide expression patterns, Proceedings of the National Academy of Sciences of the United States of America (PNAS), 95:25, 1998.

[7] G. Sherlock, Analysis of large-scale gene expression data, Brief Bioinform. vol. 2, pp.350-362, 2001.

[8] P Toronen, M Kolehmainen, G Wong, E Castren , Analysis of gene expression data using self-organizing maps, FEBS Letters, 451(2):142-146, 1999.

[9] DNA chip, http://mbel.kaist.ac.kr/research/DNAchip_en.html.

[10] WIKIPEDIA, http://en.wikipedia.org/wiki/Genetic_algorithm.

[11] Aleksander I. and Morton H., An introduction to neural computing, 2nd edition.

[12] Particle Swarm Optimization Homepage, http://www.cis.syr.edu/ ~mohan/pso/.

[13] WIKIPEDIA, http://en.wikipedia.org/wiki/Ant_colony_optimization.

[14] Peng Yuqing, Hou Xiangdan, Liu Shang, The K-means Clustering Algorithm based on Density and Ant colony, IEEE Int. Conf. Neural Networks & Signal Processing Nanjing, China, December 14-17, 2003.

[15] Xiang Xiao, Ernst R. Dow, Russell Eberhart, Zina Ben Miled, Robert J. Oppelt, Gene Clustering Using Self-Organizing Maps and Particle Swarm Optimization, IEEE International Workshop On High Performance Computational Biology, 2003.

[16] Julia Handl, Joshua Knowles, Marco Dorigo, Ant-Based Clustering: A Comparative Study of its relative performance with respect to k-means, average link and 1D-SOM, IRIDIA-Technical Report Series, 2003.

[17] T.Kohonen, Self-Organizing Maps, Springer-Verlag, Berlin, Germany, 1995.