An Effective Clustering Algorithm With Ant Colony

Xiaoyong Liu [a,b,c,*]
[a] Department of Computer Science, Guangdong Polytechnic Normal University,
Guangzhou, Guangdong, 510665, China
[b] National Science Library, Chinese Academy of Sciences, Beijing, 100190, China
[c] Graduate University of Chinese Academy of Sciences, Beijing 100049,China
liugucas@gmail.com, ryanliuxy@yahoo.com.cn

Hui Fu
Department of Computer Science, Guangdong Polytechnic Normal University,
Guangzhou, Guangdong, 510665, China
lindafh819@126.com

***Abstract*—This paper proposes a new clustering algorithm based on ant colony to solve the unsupervised clustering problem. Ant colony optimization (ACO) is a population-based meta-heuristic that can be used to find approximate solutions to difficult combinatorial optimization problems. Clustering Analysis, which is an important method in data mining, classifies a set of observations into two or more mutually exclusive unknown groups. This paper presents an effective clustering algorithm with ant colony which is based on stochastic best solution kept--ESacc. The algorithm is based on Sacc algorithm that was proposed by P.S.Shelokar. It's mainly virtue that best values iteratively are kept stochastically. Moreover, the new algorithm using Jaccard index to identify the optimal cluster number. The results of several times experiments in three datasets show that the new algorithm-ESacc is less in running time, is better in clustering effect and more stable than Sacc. Experimental results validate the novel algorithm's efficiency. In addition, Three indices of clustering validity analysis are selected and used to evaluate the clustering solutions of ESacc and Sacc.**

***Index Terms*—Ant colony optimization, Clustering Analysis, Clustering Algorithm, Clustering Validity Analysis**

I. INTRODUCTION

Deneubourg and his colleagues (Deneubourg et al., 1990; Goss et al., 1989) have shown in controlled experimental conditions that foraging ants can find the shortest path between their nest and a food source by marking the path they follow with a chemical called pheromone. The foraging behavior of ant colonies can be replicated in simulation and inspires a class of ant algorithms known as "ant colony optimization".[1] In the early 1990s, Marco Dorigo and his colleagues introduced ant colony optimization as a novel nature-inspired metaheuristic for the solution of hard combinatorial optimization (CO) problems. Ant colony optimization (ACO) belongs to the class of meta-heuristics, which are approximate algorithms used to obtain good enough solutions to hard CO problems in a reasonable amount of computation time. [2] Some experiments showed that the ACO algorithm is a fast suboptimal heuristic algorithm based on ant communication using pheromone and is often applied to combinatorial optimization problems that may be mapped onto a node-arc graph. The algorithm involves a node choice probability which is a function of a pheromone value and the distances between nodes, to construct a path through the graph. ACO is a meta-heuristic for solving NP-hard combinatorial optimization problems. [3] The algorithm has been applied to find approximate solutions to difficult optimization problems such as traveling salesman problem, quadratic assignment problem, vehicle routing problem, job schedule problem etc[4,5]. As shown in Figure 1, two ants start from their nest in search of food source at the same time to different directions. One of them chooses the path that turns out to be shorter while the other takes the longer sojourn. The ant moving in the shorter path returns to the nest earlier and the pheromone deposited in this path is obviously more than what is deposited in the longer path. Other ants in the nest thus have high probability of following the shorter route. These ants also deposit their own pheromone on this path. More and more ants are soon attracted to this path and hence the optimal route from the nest to the food source and back is very quickly established. Such a pheromone-meditated cooperative search process leads to the intelligent swarm behavior.[8]

*Corresponding author. Tel.:+86-13426198557

doi:10.4304/jcp.5.4.598-605

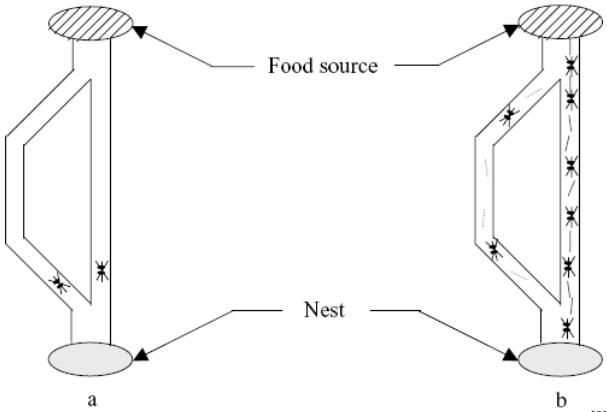

Figure1. Movement of ant algorithm from nest-food source and back[8]

Cluster analysis is a method for clustering a data set into groups of similar individuals.[6,7,9] It is a branch in multivariate analysis and an unsupervised learning in pattern recognition. Cluster analysis identifies and classifies objects individuals or variables on the basis of the similarity of the characteristics they possess. It seeks to minimize within-group variance and maximize between-group variance. Its aim is to establish a set of clusters such that cases within a cluster are more similar to each other than they are to cases in other clusters. Clustering analysis is used especially as preprocess to another data mining application. A variety of clustering algorithms exists. Mostly used clustering approach is fuzzy c-means and is one of the best for implementing the clustering process. But Fuzzy c-means isn't suitable for the data where are noise points and the imbalance of examples.

This paper uses an ACO algorithm for data clustering, in which a set of concurrent distributed agents collectively discover a sensible organization of objects for a given dataset. In the algorithm used in this study, each agent discovers a possible partition of objects in a given dataset and the level of partitioning is measured subject to some metric like Euclidean distance. Information associated with an agent about clustering of objects is accumulated in the global information hub (pheromone trail matrix) and is used by the other agents to construct possible clustering solutions and iteratively improve them. The algorithm works for a given maximum number of iterations and the best solution found with respect to a given metric represents an optimal or near-optimal partitioning of objects into subsets in given datasets[7-10].

II. RELEATED WORKS

Deneubourg and Goss et al.(1991) model and the subsequent algorithm-LF proposed by Lumer and Faieta(1994) shaped the basic form of the ant—based clustering method. Their algorithms first assumes that initially the items to be clustered are randomly laid down on a two—dimensional m * m grid, where m depends on the number of items. After completing such initialization process, a cyclic process is designed within which each ant sequentially conducts the following three activities at each step: picking up, moving and dropping. Repeating such activities, the ants may gradually divide different types of items into different clusters. The overall process ends when the clusters become stable or a given maximal iteration has been reached.[11]

Nicolas Labroche, Nicolas Monmarch´e and Gilles Venturini introduced a new method to solve the unsupervised clustering problem, based on a modeling of the chemical recognition system of ants.[12] This algorithm allow ants to discriminate between nest mates and intruders, and thus to create homogeneous groups of individuals sharing a similar odor by continuously exchanging chemical cues.

P.S.Shelokar, V.K.Jayaraman and B.D.Kulkarni proposed an ant colony optimization algorithm to solve clustering problems.[8] In the algorithm, the software ants use pheromone matrix as a kind of adaptive memory, which guide other ants towards the optimal clustering solution. The pheromone (weight) deposition at location (i, j) (i.e. allocation of sample i to the cluster j in a constructed solution) depends on its objective function value (smaller function value deposit higher pheromone) and the evaporation rate. The evaporation rate is a kind of forgetting factor that helps to look into other clustering locations of object i. Therefore, it will surely provide an optimal cluster representation for a clustering problem as iterations progress. Computational simulations revealed very encouraging results in terms of the quality of solution found, the average number of function evaluations and the processing time required. Figure 2 showed the flow chart of ACO clustering algorithm-Sacc in detail. Gülüzar Kekec, Nejat Yumusak and Numan Çelebi presented two new techniques based on Sacc. For the purpose of increasing the working performance of Sacc algorithm developed to cluster data with ant colony optimization technique, the first proposed technique brought the pheromone amount to initial values every 50 iteration to avoid from stagnation behavior. In other words, clustering was performed in a progressive technique. Moreover, aiming minimize the stagnation behavior of ants, the second proposed technique followed the pheromone amounts of ants and if there was no change on the pheromone concentration of every path after last 10 iterations, it brought the pheromone amount to initial values. In other words, to improve the solution, a feedback technique is applied on the algorithm. Numerical examples showed the increase on the performance with the addition of the above techniques.

III. CLUSTERING ALGORITHM WITH ANT COLONY BASED ON STOCHASTIC BEST SOLUTION KEPT-ESACC

A. Clustering With Ant Colony Optimization-Sacc

The aim of data-clustering is to obtain optimal assignment of N objects in one of the K clusters where N is the number of objects and K is the number of clusters.

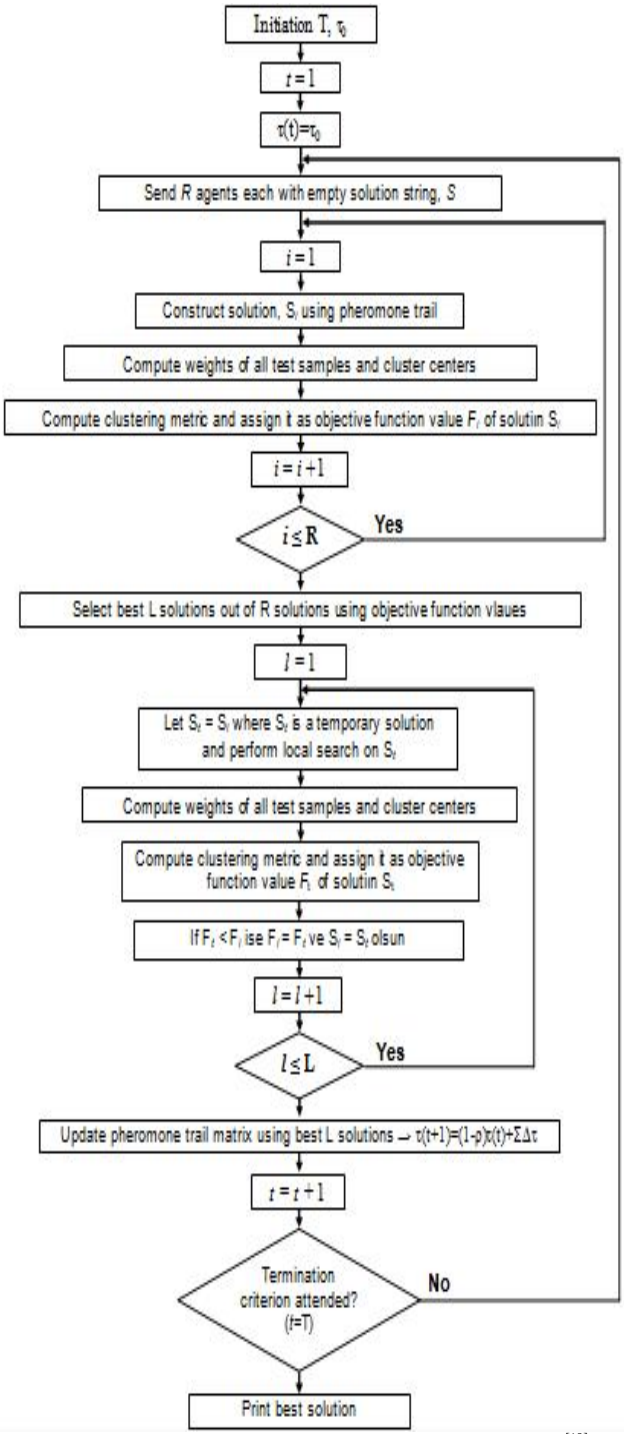

Figure 2. The flow chart of ACO clustering algorithm-Sacc[10]

In Clustering With Ant Colony Optimization-Sacc[8], ants start with empty solution strings and in the first iteration the elements of the pheromone matrix are initialized to the same values. With the progress of iterations, the pheromone matrix is updated depending upon the quality of solutions produced. Each ant selects a cluster number with a probability value for each element of S string to form its own solution string S. The quality of constructed solution string S is measured in terms of the value of objective function for a given data-clustering problem. This objective function is defined as the sum of squared Euclidian distances between each object and the center of belonging cluster. Then, the elements of the population, namely agents are sorted increasingly by the objective function values. Because, the lower objective function value, the higher fitness to the real solution, namely, lower objective function values are more approximated to real solution values. An optimal solution is that solution which minimizes the objective function value. If the value of best solution in memory is updated with the best solution value of the current iteration if it has a lower objective function value than that of the best solution in memory, otherwise the best solution in memory kept. This process explains that an iteration of the algorithm is finished. Algorithm iterates these steps repeatedly until a certain number of iterations and solution having lowest function value represents the optimal partitioning of objects of a given dataset into several groups. Obviously, the lower objective function value, the better the algorithm. Ref [8] and Ref [10] described the algorithm-Sacc in detail.

B. Cluster validity index

The main subject of cluster validity is evaluating the results of a clustering algorithm relative to others created by other clustering algorithms, or by the same algorithms using different parameter values. Cluster validity is important in clustering analysis because the result of clustering needs to be validated in most applications. There are several main different cluster validity indices.[13][14]

1) Dunn's index

Dunn's index is based on the idea of identifying the cluster sets that are compact and well separated. For any partition of clusters, where c_i represent the *i*-cluster of such partition, the Dunn's validation index, D, could be calculated with the following formula:

$$D = \min_{1\le i\le n}\left\{\min_{\substack{1\le i\le n\\ i\ne j}}\left\{\frac{d(c_i,c_j)}{\max_{1\le k\le n}\{d'(c_k)\}}\right\}\right\}$$

Where:

$d(c_i,c_j)$: distance between clusters c_i , and c_j (inter-cluster distance);

$d'(c_k)$:intra-cluster distance of cluster c_k ;

n : number of clusters.

The minimum is calculating for number of clusters defined by the mentioned partition. The main goal of the measure is to maximize the inter-cluster distances and minimize the intra-cluster distances.

2) Jaccard index

In Jaccard index, which has been commonly applied to assess the similarity between different partitions of the same dataset, the level of agreement between a set of class labels C and a clustering result K is determined by the number of pairs of points assigned to the same cluster in both partitions:

$$J(C,K)=\frac{a}{a+b+c}$$

where a denotes the number of pairs of points with the same label in C and assigned to the same cluster in K, b denotes the number of pairs with the same label, but in different clusters and c denotes the number of pairs in the same cluster, but with different class labels. The index produces a result in the range [0, 1], where a value of 1.0 indicates that C and K are identical.

3) Rand index

Rand index simply measures the number of pair wise agreements between a clustering K and a set of class labels C, normalized so that the value lies between 0 and 1:

$$J(C,K)=\frac{a+d}{a+b+c+d}$$

where a denotes the number of pairs of points with the same label in C and assigned to the same cluster in K, b denotes the number of pairs with the same label, but in different clusters, c denotes the number of pairs in the same cluster, but with different class labels and d denotes the number of pairs with a different label in c that were assigned to a different cluster in K. The index produces a result in the range [0,1], where a value of 1.0 indicates that C and K are identical. A high value for this measure generally indicates a high level of agreement between a clustering and the annotated natural classes.

4) Folks and Mallows index

Folks and Mallows introduced an index:

$$FM=\sqrt{\frac{a}{a+b}\cdot\frac{a}{a+c}}$$

Where, a denotes the number of pairs of points with the same label in C and assigned to the same cluster in K, b denotes the number of pairs with the same label, but in different clusters and c denotes the number of pairs in the same cluster, but with different class labels.

C. The novel algorithm—Esacc

In Sacc, The value of L is fixed according with the number of ants in local search. For example, the number of ants is 20 and the percent is 20%, so the value of L is 4(20*20%). For speeding up the convergence, the anterior of best value matrix that are chosen stochastically can be kept to become a part of best value matrix of next cycle in the new algorithm- ESacc.

In Sacc algorithm, the number of clusters of datasets is set as user parameter, while, in ESacc algorithm, Jaccard index is used to identify the optimal number of clusters. The number of clustering is optimal when the value of Jaccard index is the largest.

The novel algorithm-ESacc uses this algorithmic diagram demonstrated below:

Initiation of the parameters
Generate the optimal number of clustering of datasets using Jaccard-index
While (until result conditions supplied) do
Generate Initial Solutions
Clustering by the initial Clustering center
Apply Local Search
Update Pheromone Trail
Keep best solutions stochastically as a part of Initial Solution of next cycle
End

In ESacc algorithm, which is the same to Sacc algorithm[8], the data clustering problem can be described as the following mathematical formula:

$$\min F(U,W)=\sum_{j=1}^{K}\sum_{i=1}^{N}\sum_{v=1}^{n}W_{ij}\left\|x_{iv}-U_{jv}\right\|^2$$

Where, K is the number of cluster; N is the number of data points in dataset; n is the number of dimensions of data points; U is a cluster matrix of size $K\times n$; W is a weight matrix of size $N\times K$; x_{iv} is a value of v th dimension of i th data point.

$$W_{ij}=\begin{cases}1 & if\ i\ belongs\ to\ cluster\ j\\ 0 & else\end{cases}$$

$$U_{jv}=\frac{\sum_{i=1}^{N}W_{ij}x_{iv}}{\sum_{i=1}^{N}W_{ij}}$$

($i=1,2,3,\ldots N$; $j=1,2,3,\ldots K$; $v=1,2,3,\ldots n$)

The following formula is used to update pheromone trail matrix.

$$\tau_{ij}(t+1)=(1-\rho)\tau_{ij}(t)+\sum\Delta\tau_{ij}$$

Where, $(1-\rho)$ denotes the evaporation rate. The amount $\Delta\tau_{ij}$ is equal to $1/F_l$, if cluster j is assigned to i th data point of the solution constructed by ant l and zero otherwise. An optimal solution is that solution which minimizes the value of the objective function $F(U,W)$

The flow chart of ant colony optimization algorithm developed for solving data clustering problem and explained in detail above is shown in Figure 3.

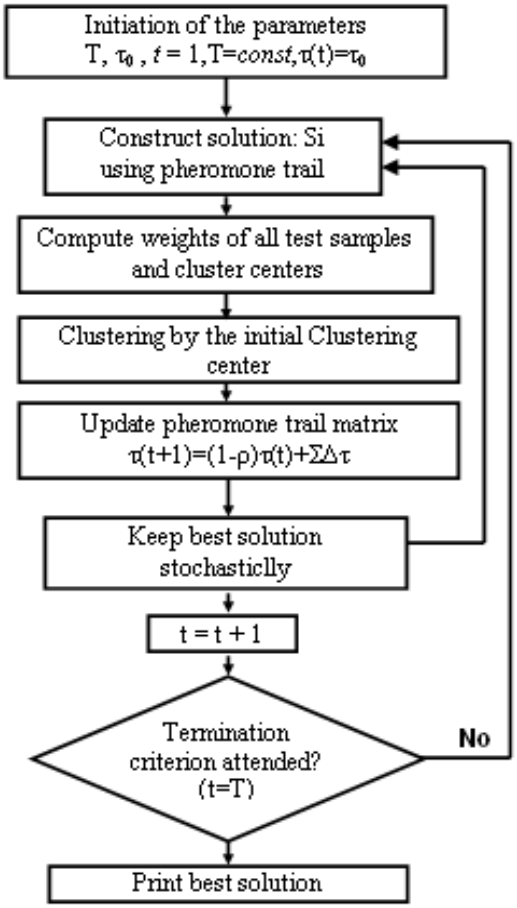

Figure 3. The flow chart of ESacc

Ⅳ. NUMERICAL EXAMPLES

For the compare of performance between Sacc and ESacc, three datasets, which are well known datasets in data mining, i.e. IRIS, WINE and Synthetic Control Chart Time Series, are chosen to test the novel algorithm. The program of the new algorithm is written by Matlab 7.0 (R14) and run on a computer with 1.8GHz CPU, 448MB DDR RAM.

A. Iris

1) Data Description[15]

The iris dataset consists of 150 data points with four attributes, and it is stored in a text file. It is one of the best known datasets to be found in the pattern recognition literature. Fisher's paper is a classic in the field and is referenced frequently to this day. The data set contains 3 classes of 50 instances each class, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are not linearly separable from each other. Figure 4 and Figure 5 show the distribution of Iris data points in first two dimensions and first three dimensions respectively. To find the best number of cluster of iris dataset, Jaccard index is used as an evaluation index. From figure 6, the value of Jaccard index is largest when the value of K is equal three. So, the optimal number of cluster is three.

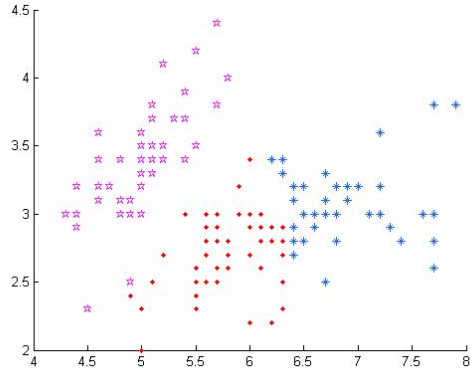

Figure 4. The distribution of Iris data(two dimensions)

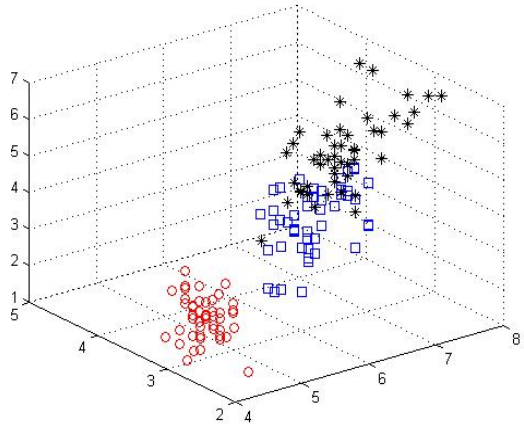

Figure 5. The distribution of Iris data(three dimensions)

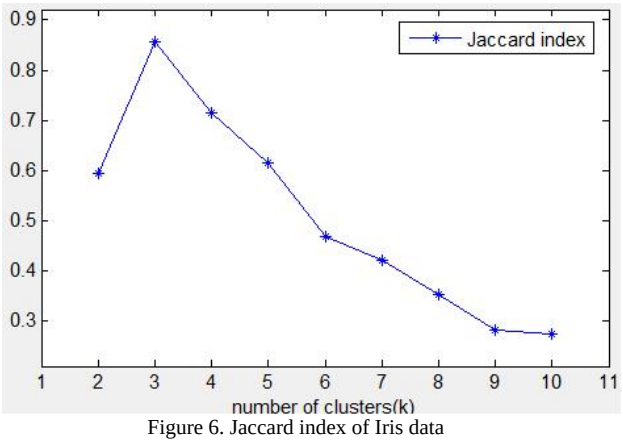

Figure 6. Jaccard index of Iris data

2) Experiment Results

Table Ⅰ lists the appropriate values of some main parameters in two algorithms in detail. L denotes the number of ants in ant colony. The value of E is the number of the best values kept stochastically in each iteration. In Sacc algorithm, the value of E is fixed, i.e. E is equal zero, while the value of E is chosen randomly between 2 and 6 in ESacc algorithm. The value of evaporation rate is equal 0.1 in two algorithm. The number of Iterations is 1000.

The two algorithms- Sacc and ESacc are run ten times respectively. The numeric results are listed in Table Ⅱ.

The running time of two algorithms during ten times experiments is showed in Figure 7. It is obviously that Sacc and ESacc aren't difference in running times, but ESacc is more stable and robust than Sacc. The compare of average of best value in ten experiments is showed in figure 7.

The Best Value that is gotten by ESacc is less than by Sacc from figure 8. In addition, the speed of the ESacc's convergence is manifestly faster than one of Sacc.

TABLE I.
PARAMETERS' VALUES

	No. of ants	Iterations	Evaporation rate	L	E
Sacc	20	1000	0.1	10	0
ESacc	20	1000	0.1	10	[2,6]

TABLE II.
THE COMPARE OF NUMERIC RESULTS

	Best Value	Avg. of Best Value	Min CPU Time (s)	Avg. of CPU Time	Std. of CPU Time
Sacc	256.91	260.41	105.6	107	2.27
ESacc	250.63	258.61	106.7	107	0.40

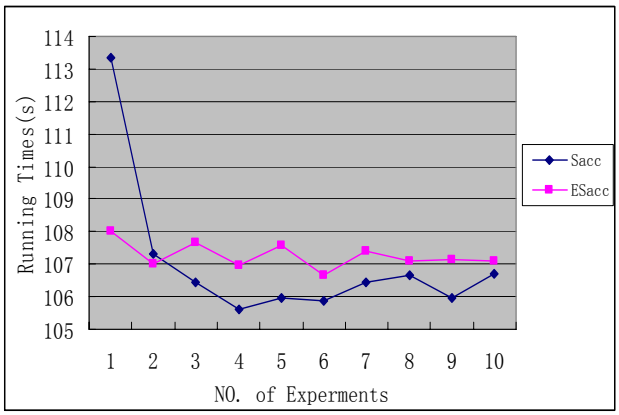

Figure 7. Running time during ten experiments (Iris data)

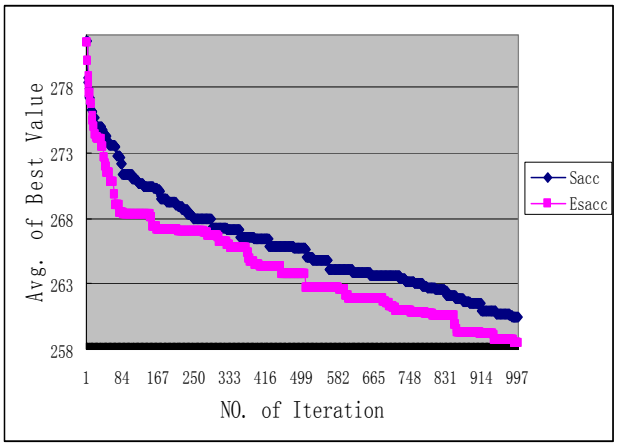

Figure 8. The compare of Avg. of best value in ten experiments (Iris data)

B. Wine

1) Data Description[15]

The dataset is from the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. It consists of 178 instances with 13 dimensions. To find the best number of cluster of iris dataset, Jaccard index is used as indicator. As showed in figure 9, the value of Jaccard index is largest when the value of K is equal three. So, the optimal number of cluster of wine dataset is three.

2) Experiment Results

Table1 Ⅲ lists the appropriate values of mainly parameters in two algorithms in detail. In Sacc, E=0, while the value of E is chosen randomly between 5 and 10 in ESacc.

The two algorithms- Sacc and ESacc are run ten times respectively. The numeric results are listed in Table . The running time of two algorithms during ten times experiments is showed in figure 10. In figure 11, the compare of Avg. of best value in ten experiments is showed. It is obviously that ESacc is better than Sacc in running times including minimal running time and average of running time, so ESacc is more stable and robust than Sacc. In additional, the Best Value that is gotten by ESacc is less than Sacc. So ESacc has better performance than Sacc in wine dataset.

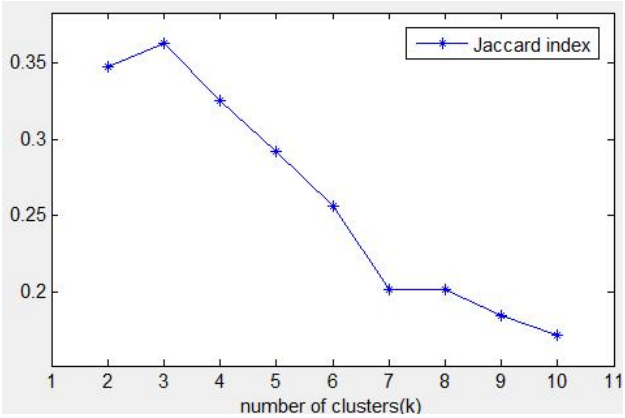

Figure 9. Jaccard index of Wine data

TABLE III.
PARAMETERS' VALUES

	No. of ants	Iterations	Evaporation rate	L	E
Sacc	20	1000	0.1	16	0
ESacc	20	1000	0.1	16	[5,10]

TABLE IV.
THE COMPARE OF NUMERIC RESULTS

	Best Value (10^3)	Avg. of Best Value	Min Time (s)	Avg. of Time	Std. of Time
Sacc	42.8	43.2	269.3	288.8	15.9
ESacc	41.9	42.9	267.8	270.6	1.85

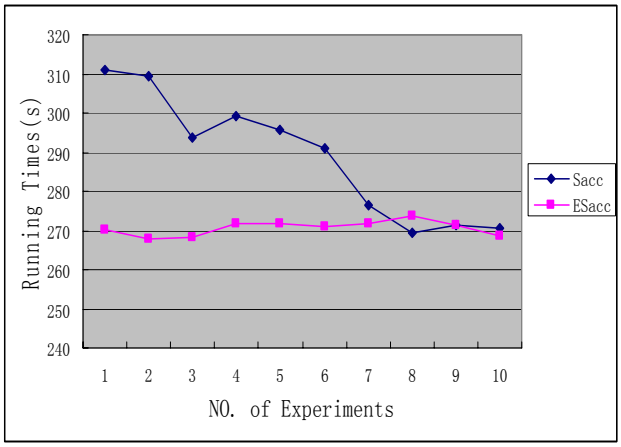

Figure 10. Running time during ten experiments (Wine data)

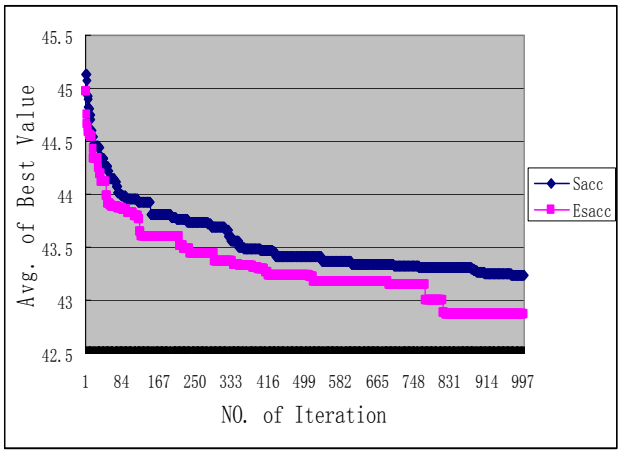

Figure 11. The compare of Avg. of best value in five experiments (Wine data)

C. Synthetic Control Chart Time Series

1) Data Description

This dataset- Synthetic Control Chart Time Series (Sccts) contains 600 examples of control charts synthetically generated by the process in Alcock and Manolopoulos (1999) with six different classes. The data is stored in an ASCII file, 600 rows, 60 columns. (From: http://kdd.ics.uci.edu/) To find the best number of cluster of Sccts dataset, Jaccard index is used as evaluation index. From figure 12, the value of Jaccard index is largest when the value of K is equal three. So, the optimal number of cluster is six.

2) Results

Table Ⅴ lists the appropriate values of mainly parameters in two algorithms in detail. The value of E is the number of the best values kept. In Sacc algorithm, the value of E is equal zero, while the value of E is chosen randomly between 10 and 20 in ESacc algorithm.

In Sccts dataset, the two algorithm is run several times because of resource limitations. The number of data points is more than the other of Iris and Wine. But it's showed that ESacc algorithm has better performance than Sacc algorithm.

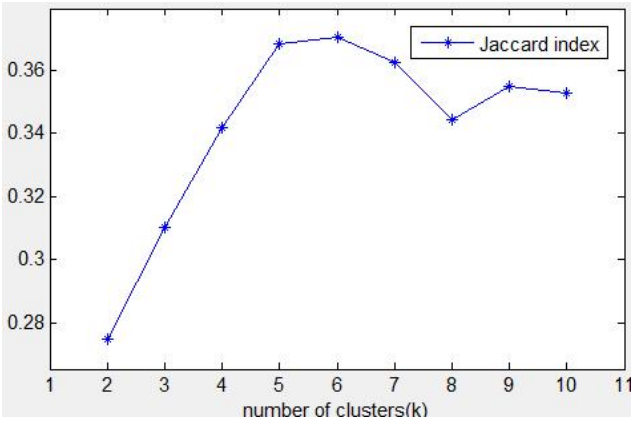

Figure 12. Jaccard index of Sccts data

TABLE V.
PARAMETERS' VALUES

	No. of ants	Iterations	Evaporation rate	L	E
Sacc	60	1000	0.1	30	0
ESacc	60	1000	0.1	30	[10,20]

TABLE VI.
THE COMPARE OF NUMERIC RESULTS

	Best Value(10^3)	Running Time(min)
Sacc	44.3	57
ESacc	44.1	50

The two algorithms- Sacc and ESacc are run several times respectively. The numeric best results of two algorithms in several times are listed in Table Ⅵ. It is obviously that ESacc is better than Sacc in running times and best value, so ESacc has better performance than Sacc in Synthetic dataset that is complex.

D. Clustering Validity analysis

To further assess two algorithms, Jaccard index, Rand index and Folks and Mallows index (FM index) are chosen to compare the performance of Sacc and ESacc. Those indices can be used to evaluate the results of a clustering algorithm relative to others created by other clustering algorithms. The results are listed in Table Ⅶ, Ⅷ and Ⅸ. The results of compare demonstrates the value of Jaccard index, Rand index and FM index of ESacc algorithm is more than one of Sacc except for the FM index in the iris dataset.

TABLE VII.
THE COMPARE OF JACCARD INDEX

	Iris	Wine	Sccts
Sacc	0.43	0.30	0.090
ESacc	0.49	0.33	0.092

TABLE VIII.
THE COMPARE OF RAND INDEX

	Iris	Wine	Sccts
Sacc	0.48	0.56	0.723
ESacc	0.58	0.59	0.724

TABLE IX.
THE COMPARE OF FM INDEX

	Iris	Wine	Sccts
Sacc	0.48	0.34	0.166
ESacc	0.40	0.38	0.169

E. Discussion

From the result of three datasets and a cluster validity indictor-Jaccard index, the performance of ESacc algorithm is better than Sacc. But the two algorithms need spend more time to run. Furthermore, in Sccts dataset, ESacc and Sacc have worse result due to the number of dataset is more. So, the ESacc algorithm can be improved in further work.

Ⅳ. SUMMARY

This paper presents a novel clustering algorithm with ant colony based on stochastic best solution kept--ESacc. The results of numeric experiments in three datasets show that the new clustering algorithm-ESacc is more robust and have stable performance than Sacc. When the dataset consists of abundant and several dimensions data points such as wine dataset and synthetic dataset, the performance of ESacc is better than Sacc.

ACKNOWLEDGMENT

The authors would like to thank anonymous reviewers for their constructive and enlightening comments, which improved the manuscript. This work has been supported by grants from Program for Excellent and Creative Young Talents in Universities of Guangdong Province (LYM08074) and Guangdong Polytechnic Normal University (No.08kjy02).

REFERENCES

[1] Marco Dorigo and Thomas Stutzle. “Ant Colony Optimization”. The MIT Press, 2004, pp.21-22.
[2] Marco Dorigo, Christian Blum. “Ant colony optimization theory: A survey”. *Theoretical Computer Science*, 2005 (344) pp.243 – 278.
[3] Chi-Bin Cheng and Chun-Pin Mao, “A modified ant colony system for solving the travelling salesman problem with time windows”. *Mathematical and Computer Modelling* .2007(46), pp.1225–1235.
[4] C. Blum. “Ant colony optimization: Introduction and recent trends”. *Physics of Life Reviews*, 2005 (2) , pp.353-373.
[5] Marco Dorigo and L.M. Gambardella. “Ant colony system: a cooperative learning approach to the tracelling salesman problem”, *IEEE Transaction on Evolutionary Computation*, 1997(1), pp.53-66.
[6] L. Kaufman, P.J. Rousseeuw, “Finding Groups in Data: An Introduction to Cluster Analysis”, Wiley, New York,1990.
[7]Wu Youshou, Ding Xiaoqing. “A new clustering method for Chinese character recognition system using artificial neural networks”, *Chinese J. of Electronics*, 1993,2(3),pp.1-8.
[8] P.S.Shelokar, V.K.Jayaraman, B.D.Kulkarni, “Ant colony approach for clustering”. *Analytica Chimica Acta*, 2004(509), pp.187 -195.
[9] Sara Saatchi, Chih Cheng Hung, “Hybridization of the Ant Colony Optimization with the K-Means Algorithm for Clustering”, *Lecture notes in Computer Science, Image Analysis*, 2005 (3540), pp.511- 520.
[10] Gülüzar KEKEÇ, Nejat YUMUŞAK and Numan ÇELEBİ. “Data Mining and Clustering With Ant Colony”. *Proceedings of 5th International Symposium on Intelligent Manufacturing Systems, Sakarya,* TURKEY 2006, pp. 1178-1190.
[11] Haoxiang Xia, Shuguang Wang and Taketoshi YOSHIDA. “A modified ant-based text clustering algorithm with semantic similarity measure”. *Journal of systems science and systems engineering*, 2006, 15(4), pp.474-492.
[12] Nicolas Labroche, Nicolas Monmarch´e and Gilles Venturini. “A new clustering algorithm based on the chemical recognition system of ants”. *Proceedings of the European Conference on Artificial Intelligence,* Lyon, France, pp.345-349,2002.
[13] R.J.G.B. Campello. “A fuzzy extension of the Rand index and other related indexes for clustering and classification assessment”. *Pattern Recognition Letters.* 2007 (28), pp.833 − 841
[14] Francois Boutin and Mountaz Hascoet. “Cluster Validity Indices for Graph Partitioning”. *Proceedings of the Conference on Information Visualization, Eighth International Conference,* Lyon, France, pp.376-381, London, England, July, 2004.
[15] UCI Repository for Machine Learning Databases retrieved from the World Wide Web: http://archive.ics.uci.edu/ml/.

Xiaoyong Liu, Currently a Lecturer who joined Department of Computer Science, Guangdong Polytechnic Normal University in 2007. He obtained his Master degree from South China University of Technology, Guangzhou City, China, in 2007, specializing in Applied Mathematics. And now, he is studying in Graduate University of Chinese Academy of Sciences. Current research interests include text mining, data mining, ant colony optimization and genetic algorithm.

Hui Fu, A Lecturer of Department of Computer Science, Guangdong Polytechnic Normal University. She obtained her Master degree from South China University of Technology, China, in 2006. Her major is Applied Mathematics. Her current research interests include data mining, operations research and genetic algorithm.