World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:2, No:10, 2008

Application of a New Hybrid optimization Algorithm on Cluster Analysis

T. Niknam, M. Nayeripour and B.Bahmani Firouzi

***Abstract*—**Clustering techniques have received attention in many areas including engineering, medicine, biology and data mining. The purpose of clustering is to group together data points, which are close to one another. The K-means algorithm is one of the most widely used techniques for clustering. However, K-means has two shortcomings: dependency on the initial state and convergence to local optima and global solutions of large problems cannot found with reasonable amount of computation effort. In order to overcome local optima problem lots of studies done in clustering. This paper is presented an efficient hybrid evolutionary optimization algorithm based on combining Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO), called PSO-ACO, for optimally clustering *N* object into *K* clusters. The new PSO-ACO algorithm is tested on several data sets, and its performance is compared with those of ACO, PSO and K-means clustering. The simulation results show that the proposed evolutionary optimization algorithm is robust and suitable for handing data clustering.

***Keywords*—**Ant Colony Optimization (ACO), Data clustering, Hybrid evolutionary optimization algorithm, K-means clustering, Particle Swarm Optimization (PSO).

I. Introduction

CLUSTER analysis is a data analysis tool used to group data with similar characteristics. It has been used in data mining tasks such as unsupervised classification and data summation, as well as segmentation of large heterogeneous data sets into smaller homogeneous subsets that can be easily managed, separately modeled and analyzed. The basic objective in cluster analysis is to discover natural groupings of objects. Cluster analysis techniques have been used in many areas such as manufacturing, medicine, nuclear science, radar scanning and research and development planning [1]-[15]. Data clustering algorithms can be hierarchical or partitional. In hierarchical clustering the data are not partitioned into a particular cluster in a single step. Instead, a series of partitions takes place, which may run from a single cluster containing all objects to *n* clusters each containing a single object. Hierarchical clustering is subdivided into agglomerative methods, which proceed by series of fusions of *n* objects into groups, and divisive methods, which separate *n* objects successively into finer groupings. Partitional clustering, on the other hand, attempts to directly decompose the data set into a set of disjoint clusters. The criterion function that the clustering algorithm tries to minimize may emphasize the local structure of the data, as by assigning clusters to peaks in the probability density function, or the global structure. Typically the global criteria involve minimizing some measure of dissimilarity in the samples within each cluster, while maximizing the dissimilarity of different clusters [1]-[15].

In this paper we concentrate on partitional clustering, and in particular a popular partitional clustering method called K-means clustering. The K-mean clustering algorithm is one of the most efficient clustering algorithms [1]-[15]. The K-means algorithm starts by initializing the K cluster centers. The input vectors (data points) are then allocated (assigned) to one of the existing clusters according to the square of the Euclidean distance from the clusters, choosing the closest. The mean (centroid) of each cluster is then computed so as to update the cluster center. This update occurs as a result of the change in the membership of each cluster. The processes of reassigning the input vectors and the update of the cluster centers is repeated until no more change in the value of any of the cluster centers. However, the K-means algorithm suffers from several drawbacks. The objective function of the K-means is not convex and hence it may contain many local minima. Consequently, in the process of minimizing the objective function, there exists a possibility of getting stuck at local minima, as well as at local maxima and saddle points. The outcome of the K-means algorithm, therefore, heavily depends on the initial choice of the cluster centers. Recently, many clustering algorithms based on evolutionary algorithms such as GA, TS and SA have been introduced [11]-[15]. However, most of evolutionary methods such as genetic algorithms, tabu search, etc, are typically very slow to find optimal solution. Recently researchers have presented new evolutionary methods such as ant colony and particle swarm algorithms to solve hard optimization problems which not only have a better response but also converge very quickly in comparison with ordinary evolutionary methods [16]-[33]. All studies done by researchers confirm that the PSO should be taken into account as a powerful technique, which is efficient enough to handle various kinds of nonlinear optimization problems. Nevertheless, it may be trapped into local optima if the global best and local best positions are equal to the particle's position over a number of iterations. Recently, numerous ideas have been used to alleviate this drawback combining other global

This work was supported by Shiraz University of Technology.

T. Niknam and M. Nayeripour are with Shiraz University of technology, Shiraz, Iran (Email: niknam@sutech.ac.ir).

B. Bahmani Firouzi is with Marvdasht Islamic Azad University, Marvdasht, Iran.

optimization algorithms such as GA, Evolutionary Programming (EP) or Simulated Annealing (SA) with the PSO [21], [25]-[29]. The basic idea is to increase the information exchange among particles using the crossover operator while mutation can help escaping local optima. In these approaches, new generation members are produced each iteration by using evolutionary algorithm and then PSO's movement rule is applied to these new members providing better opportunity of exploring new places. Another approach, which is the basic idea of this paper, deals with the improvement or reinforcement of PSO's movement rules. The ACO is one of the most powerful optimization techniques; which has been developed for solving discrete optimization problems [11] and [30]-[34]. This paper has been presented a novel hybrid evolutionary optimization method based on ACO and PSO, called PSO-ACO, for optimally clustering N object into K clusters, which not only has a better response but also converges more quickly than ordinary evolutionary algorithms.

II. Cluster analysis Problem

K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume K clusters) fixed a priori. The main idea is to define K centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to recalculate K new centroids as centers of the clusters resulting from the previous step. After we have these K new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the K centroids change their location step by step until no more changes are done. In other words, centroids do not move any more. Finally, this algorithm aims at minimizing an objective function, which in this case is a squared error function.

The objective function has been calculated as follows:

$$J(X) = \sum_{i=1}^{N} min\left\{\left\| Y_i - X_j \right\| \quad j = 1,2,3,...,K\right\} \qquad (1)$$

where $\left\| Y_i - X_j \right\|$ is a chosen distance measure between a data point Y_i and the cluster centre X_j. N and K are the number of input data and the number of cluster centers, respectively.

The algorithm is composed of the following steps:

- Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
- Assign each object to the group that has the closest centroid.
- When all objects have been assigned, recalculate the positions of the K centroids.
- Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

III. Original PSO Algorithm

PSO is a population-based stochastic search algorithm. It was first introduced by Kennedy and Eberhart [17]. Since then, it has been widely used to solve a broad range of optimization problems [16] - [18]. The algorithm was presented as simulating animals' social activities, e.g. insects, birds, etc. It attempts to mimic the natural process of group communication to share individual knowledge when such swarms flock, migrate, or hunt. If one member sees a desirable path to go, the rest of this swarm will follow quickly. In PSO, this behavior of animals is imitated by particles with certain positions and velocities in a searching space, wherein the population is called a swarm, and each member of the swarm is called a particle. Starting with a randomly initialized population, each particle in PSO flies through the searching space and remembers the best position it has seen. Members of a swarm communicate good positions to each other and dynamically adjust their own position and velocity based on these good positions. The velocity adjustment is based upon the historical behaviors of the particles themselves as well as their neighbors. In this way, the particles tend to fly towards better and better searching areas over the searching process. The searching procedure based on this concept can be described by (2).

$$\begin{aligned} V_i^{(t+1)} &= \omega.V_i^{(t)} + c_1.rand_1(\circ).(Pbest_i - X_i^{(t)}) \\ &+ c_2.rand_2(\circ).(Gbest - X_i^{(t)}) \\ X_i^{(t+1)} &= X_i^{(t)} + V_i^{(t+1)} \end{aligned} \qquad (2)$$

In these equations, i=1,2,…,NSwarm is the index of each particle, t is iteration number, $rand_1(\circ)$ and $rand_2(\circ)$ are random numbers between 0 and 1. Pbest is the best previous experience of the ith particle that is recorded. Gbest is the best particle among the entire population. N_{Swarm} is number of the swarms. Constants c1 and c2 are weighting factors of the stochastic acceleration terms, which pull each particle towards Pbest and Gbest positions. Low values allow particles to roam far from the target region before being togged back. On the other hand, high values result in abrupt movements toward, or backward the target region. Hence, the learning factors c1 and c2 are often set to 2.0 according to early experiences [16]. In general, the inertia weight ω is set according to the following equation [16]:

$$\omega^{(t+1)} = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{t_{max}} \times t \qquad (3)$$

In equation (3), t_{max} is the maximum number of iterations

and t is the current iteration number. ω_{max} and ω_{min} are maximum and minimum of the inertia weights, respectively.

IV. Ant Colony Optimization Method

Dorigo and his colleague's first proposed Ant Colony algorithms as a multi-agent approach to difficult combinatorial optimization problems like the Traveling Salesman Problem (TSP) and the Quadratic Assignment Problem (QAP) [11] and [19]-[22]. A number of studies based on ant colony algorithm have been conducted in order to solve some optimization problems such as TSP (Traveling Salesman Problem), unit commitment, economic dispatch, reactive power pricing in restructured networks, Vol / Var control in distribution networks, etc [19]-[22].

Ants are insects which live together. Since they are blind animals, they find the shortest path from nest to food with the aid of pheromone. The pheromone is the chemical material deposited by ants, which serves as critical communication media among ants, thereby guiding the determination of the next movement. Generally, the following factors are used to simulate ant systems:

- Intensity of pheromone
- Length of the path

To select the next path, the state transition probability is defined as follows:

$$P_{ij} = \frac{(\tau_{ij})^{\gamma_2}(1/L_{ij})^{\gamma_1}}{\sum_{\substack{j=1 \\ j\neq i}}^{NA}(\tau_{ij})^{\gamma_2}(1/L_{ij})^{\gamma_1}} \quad (4)$$

where τ_{ij} and L_{ij} are the intensity of pheromone and length of the path between nodes j and i, respectively. γ_1 and γ_2 are control parameters for determining the weight of trail intensity and length of the path, respectively. NA is the number of ants.

After selecting the next path, trail intensity of pheromone is updated as:

$$\tau_{ij}(k+1) = \rho\tau_{ij}(k) + \Delta\tau_{ij} \quad (5)$$

In the above equation, ρ is a coefficient such that (1-ρ) represents the evaporation of trail between time k and k+1 and $\Delta\tau_{ij}$ is the amount of pheromone trail added to τ_{ij} by ants.

V. Application of PSO-ACO to Clustering

In this section a new method is proposed to incorporate intelligent decision-making structure of ACO algorithm into the original PSO where the global best position is unique for every particle. However, it uses randomly selection procedure of ACO algorithm to assign different global best positions to every distinct agent. This new algorithm is called PSO-ACO and is applied to clustering. The control variables are the cluster centers. To apply the PSO-ACO algorithm to solve the clustering problem, the following steps should be taken and repeated.

Step1: Generate the initial population and initial velocity

The initial population and initial velocity for each particle are randomly generated as follows:

$$Population = \begin{bmatrix} X_1 \\ X_2 \\ \dots \\ X_{N_{Swarm}} \end{bmatrix}$$

$$X_i = [Center_1, Center_2, ...Center_K], \quad i = 1,2,3,..., N_{Swarm} \quad (6)$$

$$Center_j = [x_1, x_2, ..., x_d]$$

$$x_i^{min} < x_i < x_i^{max}$$

$$Velocity = \begin{bmatrix} V_1 \\ V_2 \\ \dots \\ V_{N_{Swarm}} \end{bmatrix}$$

$$V_i = [Center\_V_1, Center\_V_2, ...Center\_V_K], \quad i = 1,2,3,..., N_{Swarm} \quad (7)$$

$$Center\_V_j = [v_1, v_2, ..., v_d]$$

$$v_i^{min} < v_i < v_i^{max}$$

where $Center_j$ is the j^{th} cluster center for the i^{th} individual. $Center\_V_j$ is the velocity of the j^{th} cluster center for the i^{th} individual. V_i and X_i are velocity and position of the i^{th} individual, respectively. d is the dimension of each cluster center. v_i^{max} and v_i^{min} are the maximum and minimum value of velocity of each point belongs to the j^{th} cluster center, respectively. x_i^{max} and x_i^{min} (each feature of center) are the maximum and minimum value of each point belongs to the j^{th} cluster center, respectively.

Step2: Generate the initial trail intensity

At initialization phase, it is assumed that the trail intensity between each pair of swarms is the same and is generated as follows:

$$Trail\_Intensity = [\tau_{ij}]_{N_{Swarm} \times N_{Swarm}}$$

$$\tau_{ij} = \tau_0 \quad (8)$$

where τ_{ij} and τ_0 are trial intensity between the i^{th} and j^{th} swarms and initial trial intensity, respectively.

Step3: Calculate objective function value

The objective function is evaluated for each individual.

Step4: *Sort the initial population based on the objective function values*

The initial population is ascending based on the value of the objective function.

Step5: Select the best global position

The individual that has the minimum objective function is selected as the best global position (*Gbest*).

Step6: *Select the best local position*

The best local position (*Pbest*) is selected for each individual.

Step7: Select *the i^{th} individual*

The i^{th} individual is selected and neighbors of this particle should be defined dynamically as below:

$$S_i = \{ X_j \mid \|X_i - X_j\| \le 2D_0 (\frac{1}{1-exp(\frac{-at}{t_{max}})}), i \ne j \} \quad (9)$$

where D_0 is the initial neighborhood radius, a is a parameter used to tune the neighborhood radius over the iteration, t, and $\|.....\|$ is the Euclidean distance operator.

Step8: Calculate the next position for the i^{th} individual:

There are two approaches to calculate the next position as follows:

- *Case A)* if $S_i \ne \{\ \}$, where $\{\ \}$ stands for null set

In this case, at first, the transition probabilities between the X_i and each individual in S_i are calculated as indicated in (10):

$$[Probability\ ]_i = [\ P_{i1}, P_{i2}, ..., P_{i,M}\ ]_{1\times M}$$

$$P_{ij} = \frac{(\tau_{ij})^{\gamma_2}(1/L_{ij})^{\gamma_1}}{\sum_{j=1}^{M}(\tau_{ij})^{\gamma_2}(1/L_{ij})^{\gamma_1}} \quad (10)$$

$$L_{ij} = \frac{1}{|J(X_i) - J(X_j)|}$$

where P_{ij} is the state transition probability between X_i and the j^{th} individual in S_i. M is the number of members in S_i. The roulette wheel is used for stochastic selection of the best global position.

The i^{th} particle is then moved according to following rules, if X_j is selected as the best:

$$V_i^{(t+1)} = \omega.V_i^{(t)} + c_1.rand_1(\circ).(Pbest_i - X_i^{(t)})$$
$$+ c_2.rand_2(\circ).(X_j - X_i^{(t)}) \quad (11)$$
$$X_i^{(t+1)} = X_i^{(t)} + V_i^{(t+1)}$$

The presumed pheromone level between X_i and X_j is updated at the next stage:

$$\tau_{ij}(t+1) = \rho.\tau_{ij}(t) + P_{ij} \quad (12)$$

- *Case B)* if $S_i = \{\ \}$, which means there is not any individual in particle's neighborhood,

In this case, the i^{th} particle is moved according to the following rules:

$$V_i^{(t+1)} = \omega.V_i^{(t)} + c_1.rand_1(\circ).(Pbest_i - X_i^{(t)})$$
$$+ c_2.rand_2(\circ).(Gbest - X_i^{(t)}) \quad (13)$$
$$X_i^{(t+1)} = X_i^{(t)} + V_i^{(t+1)}$$

Then, the trail intensity is updated as following, where index j represents the best particle index in the group.

$$\tau_{ij}(t+1) = \rho.\tau_{ij}(t) + r;\ \ 0.1 \le r \le 0.5 \quad (14)$$

The modified position for the i^{th} individual is checked with its limit.

Step9: If all individuals are selected, go to the next step, otherwise *i=i+1* and go back to *step7*.

Step10: *Check the termination criteria*

If the current iteration number reaches the predetermined maximum iteration number, the search procedures is stopped, otherwise the initial population is replaced with the new population of swarms and then goes back to step 3.

The last *Gbest* is the solution of the problem.

VI. Experimental Results

The experimental results comparing the PSO-ACO clustering algorithm with several typical stochastic algorithms including the ACO and PSO algorithms and K-means are provided for four real-life data sets (*Iris*, *Wine*, *Vowel* and *Contraceptive Method Choice(CMC)*), which are described as follows:

Iris data (N=150, d=4, K=3): This is the irish data set. These data set with 150 random samples of flowers from the iris species setosa, versicolor, and virginica collected by Anderson (1935). From each species there are 50 observations for sepal length, sepal width, petal length, and petal width in cm. This dataset was used by Fisher (1936) in his initiation of the linear-discriminant-function technique.

Wine data (N=178, d=13, K=3)*:* This is the wine data set, which also taken from MCI laboratory. These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. There are 178 instances with 13 numeric attributes in wine data set. All attributes are continuous. There is no missing attribute value.

Contraceptive Method Choice (N = 1473, d = 10, K = 3): This dataset is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey. The samples are married women who were either not pregnant or do not know if they were at the time of interview. The problem is to predict the current contraceptive method choice (no use, long-term methods, or short-term methods) of a woman based on her demographic and socio-economic characteristics.

The parameters required for implementation of the PSO-ACO algorithm are γ_1, γ_2, ρ, a, r, *D0, c_1, c_2, ω_{min}* and ω_{max}. In this paper, the best values for the aforementioned parameters are $\gamma_1 = \gamma_2 = 1.0$, $\rho = .99$, $a = 15$, $r = 0.5$, *D_0=10, c1=c_2=2*, N_{Swarm}=10 to 15, ω_{min}=0.4 and ω_{max}=0.9 determined by 10 runs of the algorithm.

The algorithms are implemented by using Matlab 7.1 on a Pentium IV, 2.8 GHz, 512 GB RAM computer.

Tables 1 to 3 present a comparison among the results of PSO-ACO, ACO, PSO and K-means for 100 random tails on the mentioned data sets.

TABLE 1 RESULT OBTAINED BY THE FIVE ALGORITHMS FOR 100 DIFFERENT RUNS ON IRIS DATA

Method	Function Value			Standard deviation	Number of function evaluations
	F_{best}	$F_{average}$	F_{worst}		
PSO-ACO	96.6542	96.6548	96.67412	0.009764	2523
PSO	96.8942	97.2328	97.8973	0.347168	4953
ACO	96.753	97.453	98.023	0.567	4931
K-means	97.333	106.05	120.45	14.6311	120

TABLE II RESULT OBTAINED BY THE FIVE ALGORITHMS FOR 100 DIFFERENT RUNS ON WINE DATA

Method	Function Value			Standard deviation	Number of function evaluations
	F_{best}	$F_{average}$	F_{worst}		
PSO-ACO	16295.34	16295.9	16297.9	0.86966	6432
PSO	16345.96	16417.4	16562.3	85.4974	16532
ACO	16346.78	16417.1	16502.9	80.3731	15473
K-means	16555.68	18061	18563.12	793.213	390

TABLE III RESULT OBTAINED BY THE FIVE ALGORITHMS FOR 100 DIFFERENT RUNS ON CMC DATA

Method	Function Value			Standard deviation	Number of function evaluations
	F_{best}	$F_{average}$	F_{worst}		
PSO-ACO	5694.5179	5694.9214	5697.4254	0.868771	6923
PSO	5700.985	5820.964	5923.249	46.9597	21456
ACO	5701.923	5819.1347	5912.430	45.6347	20436
K-means	5842.20	5893.60	5934.43	473.16	270

The simulation results given in Tables 1 to 3 show that PSO-ACO is very precise. In the other word, it provides the optimum value and small standard deviation in compare to those of other methods. For instance, the results obtained on the iris dataset shows that PSO-ACO converges to the global optimum of 96.6542 at almost times while the best solutions of PSO, ACO and K-means are 96.8942, 96.853 and 97.333, respectively. The standard deviation of the fitness function for this algorithm is less than of 0.009764, which it significantly is smaller than other methods.

In terms of the number of function evaluations, K-means needs the least number of function evaluations, but the results are less than satisfactory. For the CMC data set, the number of function evaluations of PSO-ACO, PSO, ACO and K-means are 6923, 21456, 20436 and 270, respectively. These results show that the number of function evaluations of PSO-ACO is less than those of other evolutionary algorithms. The number of function evaluations of the proposed algorithm for iris, wine, vowel and CMC data is 2532, 6432, 3524 and 6923, respectively. Based on the obtained simulation results, we can conclude that the changes of the number of fitness function evaluations of the proposed algorithm are less than other evolutionary algorithms for all cases. In the other words, the number of swarms in the PSO-ACO algorithm does not depend on the number of variables greatly. In the proposed algorithm, N_{Swarm} for iris, wine, CMC and vowel is 10, 15 and 15, respectively; while for the original PSO is 20, 40 and 45, respectively.

The simulation results in the tables and figures demonstrate that the proposed hybrid evolutionary algorithm converges to global optimum with a smaller standard deviation, lower convergence time and less function evaluations and leads naturally to the conclusion that the PSO-ACO algorithm is a viable and robust technique for data clustering.

VII. CONCLUSION

The particle swarm optimization algorithm is a new method, which has great abilities to cope with different types of optimization problems. However, it is still in its infancy and intensive studies are needed to improve its performance. In this paper, a novel hybrid methodology called PSO-ACO was introduced and debated in detail. PSO-ACO is a combination of two powerful optimization algorithms; Ant Colony Optimization and Particle Swarm Optimization. In this new algorithm, the decision making process of each particle for selecting the best guide just before its movement is reinforced with the ACO method. It has been shown that this combination can provide a good opportunity for all individuals and especially the most-fit particle to search the surrounding area better. PSO-ACO algorithm has been developed in this paper to solve clustering problems. The result illustrate that the proposed PSO-ACO optimization algorithm can be considered as a viable and an efficient heuristic to find optimal or near optimal solutions for clustering problems of allocating N objects to k clusters.

REFERENCES

[1]. Yi-Tung Kao, Erwie Zahara and I-Wei Kao, "A hybridized approach to data clustering" Expert Systems with Applications, Volume 34, Issue 3, pp. 1754-1762, April 2008.
[2]. J.F. Lu, J.B. Tang, Z.M. Tang and J.Y. Yang, "Hierarchical initialization approach for K-Means clustering", Pattern Recognition Letters, January 2008.
[3]. Hong Zhou and Yonghuai Liu, "Accurate integration of multi-view range images using k-means clustering", Pattern Recognition, Volume 41, Issue 1, pp. 152-175, January 2008.
[4]. Michael Laszlo and Sumitra Mukherjee , "A genetic algorithm that exchanges neighboring centers for k-means clustering", Pattern Recognition Letters, Volume 28, Issue 16, pp. 2359-2366, 1 December 2007.
[5]. Amir Ahmad and Lipika Dey , "A k-mean clustering algorithm for mixed numeric and categorical data", Data & Knowledge Engineering, Volume 63, Issue 2, pp. 503-527, November 2007.
[6]. Chung-Chian Hsu, Chin-Long Chen and Yu-Wei Su , "Hierarchical clustering of mixed data based on distance hierarchy", Information Sciences, Volume 177, Issue 20, 15, pp. 4474-4492, October 2007.
[7]. Mohammad Fathian, Babak Amiri and Ali Maroosi , "Application of honey-bee mating optimization algorithm on clustering", Applied Mathematics and Computation, Volume 190, Issue 2, pp. 1502-1513, July 2007.
[8]. Stephen J. Redmond and Conor Heneghan , "A method for initialising the K-means clustering algorithm using kd-trees", Pattern Recognition Letters, Volume 28, Issue 8, pp. 965-973, June 2007.
[9]. Georgios P. Papamichail and Dimitrios P. Papamichail , "The k-means range algorithm for personalized data clustering in e-commerce", European Journal of Operational Research, Volume 177, Issue 3, pp. 1400-1408, March 2007
[10]. R.J. Kuo, H.S. Wang, Tung-Lai Hu and S.H. Chou , "Application of ant K-means on clustering analysis", Computers & Mathematics with

Applications, Volume 50, Issues 10-12, pp. 1709-1724, November-December 2005.
[11]. P. S. Shelokar, V. K. Jayaraman and B. D. Kulkarni ",An ant colony approach for clustering ",Analytica Chimica Acta, Volume 509, Issue 2, pp. 187-195, May 2004.
[12]. Michael K. Ng and Joyce C. Wong , "Clustering categorical data sets using tabu search techniques", Pattern Recognition, Volume 35, Issue 12, December 2002, pp. 2783-2790
[13]. C. S. Sung and H. W. Jin, "A tabu-search-based heuristic for clustering", Pattern Recognition, Volume 33, Issue 5, pp. 849-858, May 2000.
[14]. Khaled S. Al-Sultan, "A Tabu search approach to the clustering problem", Pattern Recognition, Volume 28, Issue 9, pp. 1443-1451, September 1995.
[15]. Michael Laszlo and Sumitra Mukherjee, "A genetic algorithm that exchanges neighboring centers for k-means clustering" Pattern Recognition Letters, Volume 28, Issue 16, pp. 2359-2366, December 2007
[16]. J. Olamaei , T. Niknam and G. Gharehpetian, "Application of Particle Swarm Optimization for Distribution Feeder Reconfiguration Considering Distributed Generators", accepted for future publication in Applied Mathematics and Computation journal, 2008.
[17]. J. Kennedy and R. Eberhart, "Particle Swarm Optimization," IEEE International Conf. on Neural Networks, Piscataway, NJ, vol. 4, pp. 1942-1948, 1995.
[18]. T. Niknam, "An Approach Based on Particle Swarm Optimization for Optimal Operation of Distribution Network Considering Distributed Generators,", 32nd Annual Conference onIEEE Industrial Electronics, IECON 2006 , pp. :633 – 637
[19]. T. Niknam , A.M. Ranjbar and A.R. Shirani, "A new approach for distribution state estimation based on ant colony algorithm with regard to distributed generation" Journal of Intelligent and Fuzzy Systems, vol. 16, no. 2, pp.119-131, June 2005.
[20]. T.Niknam, H. Arabian and M. Mirjafari, "Reactive power pricing in deregulated environments using novel search methods" Proceedings of 2004 International Conference on Machine Learning and Cybernetics, 2004, vol. 7, p.p. 4234 – 4239, Aug. 2004
[21]. T. Niknam, A.M. Ranjbar, and A.R. Shirani, "A new approach based on ant algorithm for Volt/Var control in distribution network considering distributed generation," Iranian Journal of Science & Technology, Transaction B, vol. 29, no. B4, pp. 1-15, 2005.
[22]. M. Dorigo, G.D. Caro and L.M. Gambardella, "Ants algorithms for discrete optimization", Artificial Life, vol. 5, no. 3, pp. 137-172, 1999.

Taher Niknam was born in Shiraz, Iran. He received his B.S, M.S and PhD degrees from Shiraz University and Sharif University of Technology respectively. He is a member of faculty at the electrical engineering department of Shiraz University of Technology. His interests include power system restructuring, impact of DGs on power system, power electronics, optimization methods, and evolutionary algorithms.

Majid Nayeripour was born in 1971. He received his B. S. degree in electronic Eng. from Guilan University and M.S degree in Electrical Eng. from Esfahan University of Technology and PhD degree in Electrical Eng. form Tarbiat Modares University, Tehran, Iran. Currently, he is an Assistant Professor with the Shiraz University of Technology. His research interests include FACTS devices, Power Quality and impact of DGs on power system.

Bahman Bahmani was born in Shiraz, Iran. He received his B.S and M.S degrees from Shiraz University and Sharif University of Technology respectively. He is a member of faculty at the electrical engineering department of Marvdaht Islamic Azad University. His interests include power system modeling, power electronics, optimization methods, and evolutionary algorithms.