World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:10, 2008

A New Evolutionary Algorithm for Cluster Analysis

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

***Abstract*—**Clustering is a very well known technique in data mining. One of the most widely used clustering techniques is the k-means algorithm. Solutions obtained from this technique depend on the initialization of cluster centers and the final solution converges to local minima. In order to overcome K-means algorithm shortcomings, this paper proposes a hybrid evolutionary algorithm based on the combination of PSO, SA and K-means algorithms, called PSO-SA-K, which can find better cluster partition. The performance is evaluated through several benchmark data sets. The simulation results show that the proposed algorithm outperforms previous approaches, such as PSO, SA and K-means for partitional clustering problem.

***Keywords*—**Data clustering, Hybrid evolutionary optimization algorithm, K-means algorithm, Simulated Annealing (SA), Particle Swarm Optimization (PSO).

I. Introduction

PARTITIONING a set of data points into some non-overlapping clusters is an important topic in data analysis and pattern classification. It has many applications, such as codebook design, data mining, image segmentation, data compression, etc. Many efficient clustering algorithms have been developed for data sets of different distributions in the past several decades.

Actually clustering algorithms can be divided into two main categories: hierarchical and 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]. Among the clustering algorithms that have been developed, the k-means algorithm is the most popular one. However, in the K-means algorithm the clustering results are often dependent on the selection of the initial points. That is, the clustering solution is very much sensitive to the initial-point choices. An inappropriate setting up of initial points would lead to some unacceptable clustering results. The K-means algorithm gave better results only when the initial partitions were close to the final solution. To overcome this problem, evolutionary algorithms can be used. Recently, many clustering algorithms based on evolutionary algorithms such as GA, TS and SA have been introduced [11]-[15]. Nevertheless, 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.

Particle Swarm Optimization (PSO) is a new evolutionary computation technique first introduced by Kennedy and Eberhart in 1995 [16]. Development of its idea was based on simulation of social behavior of animals such as a flock of birds, a school of fish or a group of people who pursue a common goal in their lives. In this algorithm, each individual is referred to as a particle and presents a candidate solution to the optimization problem. Unlike other population-based methodologies, every agent moves along its velocity vector, which is updated using two different best experiences; one is the best experience which a particle has gained itself during the search procedure and the other is the best experience gained by the whole group. The combination of these can provide useful information for each particle to explore new positions in the domain of the problem. From the time when the PSO algorithm was first developed to train neural network weights until now, a lot of widespread studies have been conducted to employ this methodology in solving optimization problems [16]-[30]. All these studies 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.

To overcome this shortcoming, this paper has been presented a novel hybrid evolutionary optimization method based on SA and PSO, called PSO-SA. Also in order to the K-means shortcomings a hybrid algorithm is developed based on

This work was supported by Shiraz University of Technology.

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

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

combination of proposed PSO-SA and K-means algorithms for optimally clustering *N* object into *K* clusters. In the proposed algorithm, the output of hybrid PSO-SA algorithm is considered as the initial state of K-means.

Through experiments, we show that the PSO-SA-K algorithm efficiently finds accurate clusters on several datasets. Our simulation results indicate that the proposed hybrid evolutionary clustering algorithm out perform the PSO, SA and K-means algorithms.

The rest of this paper is organized as follows. In section 2, the cluster analysis problem is discussed. In sections 3 and 4, the basic principles of the PSO and SA algorithms are introduced, respectively. In section 5, the application of the PSO-SA-K algorithm in clustering is shown. In section 6, the feasibility of the PSO-SA-K is demonstrated and compared with the original PSO, SA and K-means for different data sets. Finally, section 7 includes the conclusion.

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:

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

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] - [17]. 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

Simulated annealing is a generalization of a Monte Carlo method for examining the equations of state and frozen states of n-body systems. The concept is based on the manner in which liquids freeze or metals recrystalize in the process of annealing. In an annealing process a melt, initially at high temperature and disordered, is slowly cooled so that the system at any time is approximately in thermodynamic equilibrium. As cooling proceeds, the system becomes more ordered and approaches a "frozen" ground state at *T=0*. Hence the process can be thought of as an adiabatic approach to the lowest energy state. If the initial temperature of the system is too low or cooling is done insufficiently slowly the system may become quenched forming defects or freezing out in meta-stable states (ie. trapped in a local minimum energy state) [18]-[19]. The original Metropolis scheme was that an initial state of a thermodynamic system was chosen at energy E and temperature T, holding T constant the initial configuration is perturbed and the change in energy ΔE is computed. If the change in energy is negative the new configuration is accepted. If the change in energy is positive it is accepted with a probability factor $\exp(-\Delta E/T)$. This processes is then repeated sufficient times to give good sampling statistics for the current temperature, and then the temperature is decremented and the entire process repeated until a frozen state is achieved at *T=0*.

V. Application of PSO-ACO to Clustering

A hybrid algorithm is developed in this paper, which is intended to improve the performance of data clustering techniques currently used in practice. SA algorithm has the advantage of being a very efficient local search procedure but its convergence is extremely sensitive to the chosen staring point. PSO belongs to the class of global search procedures but requires much computational effort. K-means algorithm trends to converge faster than the PSO and SA as it requires fewer function evaluations, but it usually results in less accurate clustering. In this paper, we try to use the advantages of SA, PSO and K-means algorithms for clustering simultaneously. To apply the hybrid algorithm named PSO-SA-K on clustering the following steps should be repeated:

Step1: Generate the initial population and initial velocity

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

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

$$X_i = [Center_1, Center_2, ...Center_K],$$
$$i = 1,2,3,..., N_{Swarm} \quad (4)$$
$$Center_j = [x_1, x_2, ..., x_d]$$
$$x_i^{min} < x_i < x_i^{max}$$

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

$$V_i = [Center\_V_1, Center\_V_2, ...Center\_V_K],$$
$$i = 1,2,3,..., N_{Swarm} \quad (5)$$
$$Center\_V_j = [v_1, v_2, ..., v_d]$$
$$v_i^{min} < v_i < v_i^{max}$$

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: Calculate objective function value

The objective function is evaluated for each individual.

Step3: *Sort the initial population based on the objective function values*

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

Step4: Select the best global position

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

Step5: Apply SA to search around the global solution

In this step, SA searches around *Gbest.* If the obtained solution by SA is better than *Gbest*, swap it.

Step6: Consider *Gbest* as the initial solution for the K-means algorithm

In this step to use the K-means clustering algorithm advantageous apply it on the *Gbest* as an initial solution.

Step7: Select the best local position

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

Step8: Select the i^{th} individual

The i^{th} individual is selected and is moved according to the following rules:

$$V_i^{(t+1)} = \omega V_i^{(t)} + c_1.rand_1(\circ).(Pbest_i - X_i^{(t)}) \quad (6)$$
$$+ c_2.rand_2(\circ).(Gbest - X_i^{(t)})$$
$$X_i^{(t+1)} = X_i^{(t)} + V_i^{(t+1)}$$

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 *step8*.

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-SA 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-SA-K, PSO-SA, PSO, SA 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-SA-K	96.6500	96.6500	96.6500	0	2472
PSO-SA	96.6501	96.6548	96.67412	0.010826	2553
PSO	96.8942	97.2328	97.8973	0.347168	4953
SA	97.4573	99.957	102.01	2.018	5314
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-SA-K	16,295.32	16,295.32	16,295.32	0	6,321
PSO-SA	16,295.36	16,295.98	16,298.29	0.88534	6,453
PSO	16,345.9	16,417.47	16,562.31	85.4974	16,532
SA	16,473.4	17,521.09	18,083.251	753.084	17,264
K-means	16,555.6	18,061	18,563.12	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-SA-K	5,694.2901	5,694.29	5,694.29	0	6,881
PSO-SA	5,694.60	5,696.84	5,698.08	1.7062	6,945
PSO	5,700.98	5,820.96	5,923.24	46.959	21,456
SA	5,849.0	5,893.48	5,966.9	501.86	26,829
K-means	5,842.2	5,893.60	5,934.43	473.16	270

The simulation results given in Tables 1 to 3 show that PSO-SA-K 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-SA-K converges to the global optimum of 96.6500 in all of runs and PSO-SA reaches to 96.6501at almost times while the best solutions of PSO, SA and K-means are 96.8942, 97.4573 and 97.333, respectively. The standard deviation of the fitness function for this algorithm and PSO-SA are 0 and 0.010826, which they significantly are smaller than other methods. Table 2 shows the result of algorithms on the wine dataset. The optimum value is 16295.32, which is obtained in all the runs of PSO-SA-K algorithm. Noticeably other algorithms fail to attain this value even once within 100 runs. Table 3 provides the results of algorithms on the CMC dataset. As seen from the results of the PSO-SA-K algorithm are far superior that of others.

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 iris dataset, the number of function evaluations of PSO-SA-K, PSO-SA, PSO, SA and K-means are 2472, 2553, 4953, 5314 and 120, respectively. The number of function evaluations of PSO-SA-K for Wine, and CMC are 6,321 and 6,881 respectively. These results show that the number of function evaluations of PSO-SA-K and PSO-SA are less than those of other evolutionary algorithms. 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-SA-K algorithm does not depend on the number of variables greatly. In the proposed algorithm, N_{Swarm} for iris, wine, and CMC is 10, 12 and 12, respectively.

VII. Conclusion

Clustering has long been the basis of many knowledge discovery tasks such as machine learning, statistics, data mining, and pattern recognition. The well-known K-means algorithm, which has been successfully applied to many practical clustering problems, suffers from several drawbacks due to its choice of initializations. In order to overcome K-means shortcomings, evolutionary algorithms can be considered as a choice. The particle swarm optimization algorithm is one of popular evolutionary algorithms that converge rapidly during the initial stages of a global search, but around global optimum, the search process will become very slow. To improve the performance of PSO on clustering problem, we have proposed a hybrid PSO algorithm, which is combining of the PSO, SA and K-means algorithms. In the proposed algorithm, at first SA algorithm searches around the best solution obtained by PSO and then the output results feed to the K-means algorithm. The result illustrate that the proposed PSO-SA-K 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. The experimental results indicate that the proposed optimization algorithm is at least comparable to the other algorithms in terms of function evaluations and standard deviations.

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 and Babak Amiri, "A honey-bee mating approach on clustering", The International Journal of Advanced Manufacturing Technology, DOI 10.1007/s00170-007-1132-7.

[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. Kennedy and R. Eberhart, "Particle Swarm Optimization," IEEE International Conf. on Neural Networks, Piscataway, NJ, vol. 4, pp. 1942-1948, 1995.

[17]. 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.

[18]. Tai-Hsi Wu, Chin-Chih Chang and Shu-Hsing Chung, "A simulated annealing algorithm for manufacturing cell formation problems", Expert Systems with Applications, Volume 34, Issue 3, April 2008, Pages 1609-1617.

[19]. Klaus Meer, "Simulated Annealing versus Metropolis for a TSP instance" Information Processing Letters, Volume 104, Issue 6, 16 December 2007, Pages 216-219

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.

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.