OPTIMIZATION OF FUZZY CLUSTERING CRITERIA BY A HYBRID PSO AND FUZZY C-MEANS CLUSTERING ALGORITHM

E. MEHDIZADEH, S. SADI-NEZHAD AND R. TAVAKKOLI-MOGHADDAM

ABSTRACT. This paper presents an efficient hybrid method, namely fuzzy particle swarm optimization (FPSO) and fuzzy c-means (FCM) algorithms, to solve the fuzzy clustering problem, especially for large sizes. When the problem becomes large, the FCM algorithm may result in uneven distribution of data, making it difficult to find an optimal solution in reasonable amount of time. The PSO algorithm does find a good or near-optimal solution in reasonable time, but we show that its performance may be improved by seeding the initial swarm with the result of the c-means algorithm. Various clustering simulations are experimentally compared with the FCM algorithm in order to illustrate the efficiency and ability of the proposed algorithms.

1. Introduction

From a general point of view, pattern recognition is defined as the process of searching for data structures and then classifying them into categories, so that while the association between the intra-categorical structures is high, the association between the inter-categorical structures is low. Clustering is the most fundamental and significant method in pattern recognition and is defined as a form of data compression, in which a large number of samples are converted into a small number of representative prototypes or clusters [18]. It plays a key role in searching for structures in data, and involves the task of dividing data points into homogeneous classes or clusters. Depending on the data and application, different types of similarity measures may be used to identify classes where the similarity measure controls how to form clusters. Some examples of such measures are distance, connectivity, and intensity.

In real-world cases, there may be no sharp boundaries between clusters, and in such cases, fuzzy clustering will be a better choice for the data. In non-fuzzy (crisp environment) or hard clustering, data are divided into crisp clusters, whose data point belongs to exactly one cluster. In fuzzy clustering, each data points may belong to more than one cluster and membership grades of each of

Received: July 2007; Revised: November 2007; Accepted: January 2008

Key words and phrases: Fuzzy Clustering, Particle Swarm Optimization (PSO), Fuzzy c-means (FCM).

the data points will represent the degree to which the point belong to a particular cluster. The fuzzy c-means (FCM) method is an efficient tool for solving fuzzy clustering problems. However, the problem is by nature a combinatorial optimization problem [31] and if the data sets are very high dimensional or contain severe noise points, the FCM often fails to find the global optimum. In these cases, the probability of finding the global optimum may be increased by the use of stochastic methods, such as evolutionary or swarm-based algorithms [23].

Eberhart and Kennedy [15] first introduced particle swarm optimization (PSO) to optimize various continuous nonlinear functions.This is a population-based metaheuristic method which optimizes a given objective function. In a PSO algorithm, each member of the population is called a "particle", and each particle "flies" around in a multidimensional search space with a velocity which is constantly updated according to the particle's own experience, the experience of its neighbors or the experience of the whole swarm. PSO has successfully been applied to a wide range of applications and a comprehensive survey of the method can be found in [16]. In this paper, we propose an efficient approach using PSO to improve the FCM algorithm. In fact, the new algorithm is developed by integrating and embedding PSO into the FCM algorithm. Our experimental results and extensive numerical analyses indicate that our proposed approach improves the performance of the FCM algorithm.

The remaining of this paper is organized as follows: Section 2 presents a literature review. An overview of the fuzzy c-means (FCM) algorithm is discussed in Section 3. Particle swarm optimization (PSO) is discussed in Section 4. The proposed fuzzy PSO algorithm is illustrated in Section 5. Experimental results are summarized in Section 6. Finally, conclusions are presented in Section 7.

2. Literature Review

The objective of a clustering problem is to group a set of objects into a number of clusters. Different clustering algorithms have been used for this purpose. These algorithms can be classified into three main categories: 1) heuristic; 2) hierarchical; and 3) partition clustering methods [29]. Fuzzy clustering algorithms are partitioning methods that can be used to assign objects of the data set to their clusters. These algorithms optimize a subjective function that evaluates a given fuzzy assignment of objects to clusters. Various fuzzy clustering algorithms have been developed, of which the FCM algorithm is the most widely used in applications.

Many researchers have studied fuzzy clustering (Robens [21 and 22], Bezdek [2], Ruspini [25], Dunn [6], Hathaway and Bezdek [10], and Kaufman and Rousseeuw [14]). Yen and Bang [28] have defined four different types of clusters. Bezdek [3] developed the fuzzy c-means algorithm to solve clustering problems. Zimmermann [31] proposed methods for fuzzy data analysis. He maintains that three classes of methods can be distinguished in modern fuzzy data analysis. The first class consists of algorithmic approaches, which are fuzzified versions of classical methods, such as fuzzy clustering approaches. The second and third classes consist of knowledge-based and neural network approaches. Today, evolutionary algorithms are increasingly combined with clustering approaches. Klir and Yuan [18] have discussed fuzzy clustering methods which use pattern recognition. They have also introduced two basic methods for fuzzy clustering as follows: 1) the FCM clustering method based on fuzzy c-partitions; and 2) fuzzy equivalence relations based on a hierarchical clustering method.

A fuzzy clustering problem is, in fact, a combinatorial optimization problem [18] and obtaining optimal solutions to large problems can be quite difficult; hence approximate methods are required. Evolutionary methods are being increasingly used for obtaining good solutions to clustering problems. Bezdek and Hathaway [1] optimized the hard c-means method with a genetic algorithm. Klawonn and Keller [17] extended and applied this scheme to the FCM model. In addition, ant colony optimization (ACO) [5] has been successfully applied to clustering problems and Handl et al. [9] introduced a heuristic method based on the ACO algorithm. Similar heuristic algorithms, called ant clustering, were suggested by Kanade and Hall [13 and 12]. Runkler [24] introduced an ACO algorithm that explicitly solves the HCM and FCM cluster models, more details of which can be found in [23]. Recently, PSO has been applied to image clustering [20], network clustering [30] and [26], clustering analysis [4], and data clustering [27]. In particular, Van der Merwe and Engelbrecht [27] proposed new approaches for using PSO in clustering data; however, their clustering is not fuzzy. Runkler and Katz [23] applied PSO to cluster data using fuzzy clustering. They introduced two new methods to minimize the two reformulated versions of the FCM objective function by PSO.

3. Fuzzy c-means Clustering Algorithm

3.1. Fuzzy Clustering Model. Given the routing information of n data points and p clusters, the goal of fuzzy clustering is to cluster the data points into c clusters. The classification result can be expressed in terms of matrix $U=[\mu_{ik}]_{c\times n}$, where μ_{ik} is the membership degree of data point k with respect to cluster i and satisfies the following conditions:

$$0 \le \mu_{ik} \le 1 \qquad i=1,2,\dots,c \qquad k=1,2,\dots,n \tag{1}$$

$$\sum_{i=1}^{c} \mu_{ik} = 1 \qquad k=1,2,\dots,n \tag{2}$$

$$0 < \sum_{k=1}^{n} \mu_{ik} \le n \qquad i=1,2,\dots,c \tag{3}$$

The objective function value (OFV) of the clustering algorithm is:

$$J(p)=\sum_{k=1}^{n} \sum_{i=1}^{c} [\mu_{ik}]^m \|x_k - V_i\|^2 \tag{4}$$

where, $m > 1$ is a real number that governs the influence of membership grades. V_i is the cluster center of cluster i and x_k is the vector of data point and $\|x_k - V_i\|^2$ represents the Euclidean distance between x_k and V_i.

3.2. The FCM Procedure for Fuzzy Clustering Problem. When the requirement of a crisp partition of a finite set of data is replaced with the weaker requirement of a fuzzy partition or a fuzzy pseudo partition on the same set, we have a fuzzy clustering problem. The problem of fuzzy clustering is to find a fuzzy pseudo partition and the associated cluster centers by which the structure of the data is represented in the best possible way. To solve the problem of fuzzy clustering, we need to formulate a criterion for the performance index. Usually, the performance index is based upon cluster centers and to minimise $J(p)$ we use the following equations for updating solutions:

$$V_i = \frac{\sum_{k=1}^{n} [\mu_{ij}]^m x_k}{\sum_{k=1}^{n} [\mu_{ij}]^m} \quad , \quad i = 1,2,\ldots,c \tag{5}$$

$$\mu_{ik}^{(t+1)} = \left[\sum_{j=1}^{c} \left(\frac{\left\| x_k - V_i^{(t)} \right\|^2}{\left\| x_k - V_j^{(t)} \right\|^2} \right)^{\frac{1}{m-1}} \right]^{-1} \tag{6}$$

where, $\mu_{ik}^{(t+1)}$ is the membership degree of data point k in cluster i.

A fuzzy pseudo-partition is often called a fuzzy c-partition, where c is the number of fuzzy classes in the partition. The fuzzy c-means (FCM) clustering method is based on fuzzy c-partitions developed by Bezdek [3] to solve the clustering problem and has proved to be quite successful. The algorithm is based on the assumption that the desired number of clusters c, real number m, stopping criterion ε and the distance function are given and proceeds as follows:

Step 1) Let $t=0$. Select an initial fuzzy pseudo-partition $p^{(0)}$.

Step 2) Calculate the c cluster centers $V_1^{(t)}$, ..., $V_c^{(t)}$ by (5) for $p^{(t)}$ and the chosen value of m.

Step 3) Compute $\mu_i^{(t+1)}$ by (6) and update $p^{(t+1)}$.

Step 4) Compare $p^{(t)}$ and $p^{(t+1)}$. If $\left| p^{(t+1)} - p^{(t)} \right| \le \varepsilon$, then stop; otherwise, increase t by one and return to Step 2.

In the above algorithm, the parameter $m > 1$ is selected to suit the problem under consideration. The partition becomes fuzzier with increasing m and there is currently no theoretical basis for an optimal choice for its value [18].

4. Particle Swarm Optimization

In PSO, the population dynamics resembles the movement of a “bird flock” searching for food, where a social sharing of information takes place and individuals can gain from the discoveries and previous experience of their companions. Thus, each companion (particle) in the population (swarm) is assumed to "fly" over the search space in order to find promising regions of the landscape. In the case of minimizing a function, such regions possess lower function values than others visited previously. In this context, each particle is treated as a point in a D-dimensional space, which adjusts its "flying" according to its own experience as well as the experience of other particles (companions). There are many variants of PSO proposed in the literature since Eberhart and Kennedy [15] first introduced this technique. The algorithm is described below.

First we define the notation adopted in this paper: the position of the i-th particle of a swarm of size n, is represented by the D-dimensional vector $x_i=(x_{i1}, x_{i2}, ..., x_{iD})$. The best previous position (i.e., the position giving the best function value) of the i-th particle is recorded and represented by $p_i=(\rho_{i1}, \rho_{i2}, ..., \rho_{iD})$, and the position change (velocity) of the i-th particle is $Vel_i=(Vel_{i1}, Vel_{i2}, ..., Vel_{iD})$. The position of the best particle of the swarm (i.e., the particle with the smallest function value) is denoted by index p_g. The particles are then manipulated according to the following equations.

$$Vel_{id}(t+1) = \chi\{wVel_{id}(t) + c_1\phi_1[\rho_{id}(t) - x_{id}(t)] + c_2\phi_2[(\rho_{gd}(t) - x_{id}(t)]\} \tag{7}$$

$$x_{id}(t+1) = x_{id}(t) + Vel_{id}(t+1) \tag{8}$$

where $d=1,2,...,D$ and $i=1,2,...,n$.

In (7), w is the inertia weight, c_1 and c_2 are two positive acceleration constants, ϕ_1 and ϕ_2 are two random values in the range [0, 1] and χ is the constriction factor used in constrained optimization problems to control the magnitude of the velocity (in unconstrained optimization problems x is usually set equal to 1.0).

5. Proposed Fuzzy PSO Algorithm (FPSO)

In fuzzy clustering, a single particle represents a cluster center vector.

In other words, each particle $part_l$ is constructed as follows:

$$part_l = (V_1, V_2, \ldots, V_i, \ldots, V_c) \tag{9}$$

where l represents the l-th particle and $l = 1,2,...,n\_particle$ and V_i is i-th cluster center.

Therefore, a swarm represents a number of candidates clustering for the current data vector. Each point or data vector belongs to every cluster according to its membership function and thus a fuzzy membership is assigned to each point or data vector. Each cluster has a cluster center and at each iteration, presents a solution which gives a vector of cluster centers. We determine the position of vector $part_l$ for every particle, update it, and then change the position of cluster centers based on the particles. We shall use the following notation:

n: Number of data points

c: Number of cluster centers

$V_l^{(t)}$: Position of the l-th particle at stage t

$Vel_l^{(t)}$: Velocity of the l-th particle in stage t

x_k: Vector of data, where $k = 1,2,...,n$

$\rho_l^{(t)}$: Best position found by the l-th particle at stage t

$\rho_g^{(t)}$: Best position found by all particles at stage t

$P^{(t)}$: Fuzzy pseudo partition at stage t

$\mu_{ik}^{(t)}$: Membership function of the k-th data point with respect to into the i-th cluster at stage t

The fit is measured by Equation (4). The c-means algorithm tends to converge faster than the proposed FPSO algorithm, but with a less accurate clustering quality. In this section, we suggest an improvement of the performance of the PSO clustering algorithm by seeding the initial swarm with the result of the c-means algorithm.

At the initial stage, the FPSO algorithm executes the c-means algorithm once. This stage terminates according to one of two stopping criteria: (1) The maximum number of iterations; or (2) $\left|p^{(t+1)} - p^{(t)}\right| \le \varepsilon$. The result is then considered a particle in the swarm; the other particles are initialized randomly.

The following algorithm is now used to find the cluster for each data vector.

Step 1) Let $t=0$. Select the initial parameters, such as the number of cluster centers c, the initial velocity of particles, c_1, c_2, w, χ, a real number $m \in (1, \infty)$, and a small positive number ε for the stopping criterion. The initial position of the particles is that obtained by the FCM.

Step 2) Calculate $\mu_{ik}^{(t)}$ for all particles ($i=1,2,\ldots,c$ and $k=1,2,\ldots,n$) by Equation (6) and update $p^{(t+1)}$.

Step 3) For each particle, calculate the goodness of fitness using Equation (4).

Step 4) Update the global and the local best positions.

Step 5) Update $Vel_l^{(t)}$ and $V_l^{(t)}$ ($l=1,2,\ldots,n\_particle$) as given by Eq. (7) and (8).

Step 6) Go to Step 2. Compare $p^{(t)}$ and $p^{(t+1)}$ If $\left|p^{(t+1)} - p^{(t)}\right| \le \varepsilon$, then stop; otherwise, continue to Step 3.

6. Experimental Results

The main objective of this study is to assess the relative performance of the proposed FPSO with respect to the FCM modification. The performances are measured by the objective function value in Eq. (4) and the CPU time. A general rule of thumb is that a clustering result with lower $J(p)$ and lower CPU time is preferable. For a comparable assessment, we coded these methods by using the fuzzy tools available in MATLAB 7. For the FPSO algorithm we considered 10 particles, $w=0.72$, $c_1 = c_2 = 1.49$. For our experimental tests, we used a PC Pentium III (CPU 1133 MHz and 256 MB RAM) and the same parameters for all algorithms: $m=2$, $\varepsilon = 0.00001$ and at most the maximum 100 iterations.

The effectiveness of the two methods (FCM and FPSO) was tested on a number of data sets taken from the literature. These data sets are described in the following examples.

Example 1 [3]: The data set, X, consists of 15 points in $\Re^2$ as given in Table 1. Assume that we want to determine a fuzzy pseudo partition with two clusters (i.e., $c=2$).

K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x_{k1}	0	0	0	1	1	1	2	3	4	5	5	5	6	6	6
x_{k2}	0	2	4	1	2	3	2	2	2	1	2	3	0	2	4

Table 1. Data Set X of 15 Points.

Example 2 [18]: This small data set, X, consists of five points in $\Re^2$, as given in Table 2. We apply the algorithms proposed in this paper to this set of data with $c=2$ and $c=3$.

k	1	2	3	4	5
x_{k1}	0	1	2	3	4
x_{k2}	0	1	3	1	0

Table 2. Data Set X for Five Points.

Example 3: The data set, X is generated at random and consists of five points in $\Re^2$, as given in Table 3. We apply the proposed algorithms to this set of data with $c=3$.

k	1	2	3	4	5
x_{k1}	0	0	0	1	1
x_{k2}	0	2	4	1	2

Table 3. Data Set X for Five Points.

Example 4: A total of 100 data vectors are generated at random with U$\sim$ (0, 20), and we apply the algorithms to this set of data with $c=2$.

Example 5 [19]: The Zoo data set is from the UCI Machine Learning Repository [27]. This set has 101 data points, which contain information about an animal in terms of 18 categorical attributes. Each animal data point is classified into 7 classes.

Example 6 [19]: The Iris plants data set is from the UCI Machine Learning Repository. This is perhaps the best known database to be found in pattern recognition literature. The data set has 150 points containing 50 instances of each of three types of Iris plants.

Example 7 [19]: The Wine data set is from the UCI Machine Learning Repository. This data set has 178 data points and 13 attributes. Each data point is classified to three classes.

Example 8 [19]: The training Image Segmentation data set is from the UCI Machine Learning Repository. This data set has 210 data points, which contain information of an image in terms of 19 categorical attributes. Each data point is classified to seven classes (i.e., brick face, sky, foliage, cement, window, path, grass).

6.1. Effects of Parameters. We present the impact of parameters on the solution quality by using FPSO. The values given in the Table 4 report the CPU time for Example 5 taken from [19]. As mentioned before, the combination $c1 = 1.49$, $c2 = 1.49$ and $w = .72$ is the best in terms of the solution quality.

Problem	Method	Average OFV	Best OFV	Worst OFV	Average CPU time	Best CPU time	Worst CPU time
Example 1 with $c = 2$	FCM FPSO	26.328158 26.328158	26.328156 26.328156	26.328156 26.328156	.31727 .12454	.2650 .0889	.3891 .1945
Example 2 with $c = 2$	FCM FPSO	5.566497 5.566492	5.566496 5.566492	5.566499 5.566492	.62192 .12654	.3753 .0844	.7936 .1892
Example 2 with $c = 3$	FCM FPSO	1.758945 1.758944	1.758944 1.758944	1.758946 1.758944	.44425 .12451	.3374 .0900	.5997 .1941
Example 3 with $c = 2$	FCM FPSO	3.256326 3.256323	3.256325 3.256323	3.256327 3.256323	.79469 .50545	.7519 .4385	.8678 .5288
Example 4 with $c = 2$	FCM FPSO	2633.734765 2633.734765	2633.734763 2633.734757	2633.734767 2633.734757	1.13997 .55227	.7683 .5417	1.4024 .5768
Example 5 with c = 7	FCM FPSO	89.15341 88.268703	87.322890 84.002407	90.068412 90.06800	1.345331 .82019	1.1078 .8140	1.6259 .8371
Example 6-IRIS with c = 3	FCM FPSO	60.575960 60.575956	60.575958 60.575956	60.575962 60.575956	.80155 .65138	.5684 .6315	.9724 .6692
Example 7-WINE with c = 3	FCM FPSO	1796125.937087 1796125.937076	1796125.937085 1796125.937076	1796125.93709 1796125.937076	2.28148 .78182	1.7611 .7514	3.6325 .9043
Example 8 – Image with c = 7	FCM FPSO	678309.4305 678163.185	677260.451174 677260.451137	680757.049759 680757.048272	3.94473 1.20124	2.8881 1.1937	4.2313 1.2059

Table 4. Comparison of FCM and FPSO clustering algorithms.

6.2. Comparison of Methods. Table 4 shows the result of two clustering algorithms (i.e., the FCM and FPSO algorithms) for the eight examples given above, when there are 764 data points for each experiment. The values reported are: the average, the best, and the worst OFVs (objective function values) as well as their CPU time over 10 simulation runs. Our tests show that the FPSO computation times (i.e., CPU time) for all the examples are significantly lower than that for the FCM method and the solution quality as measired by OFV is also very good. When the problem becomes large, the difference between two algorithms becomes even larger. It is worth noting that the pure PSO algorithm works at high speed, but the quality of the solution is fair. On the other hand, the FCM has low speed but high solution quality, especially for large-sized problems. The FPSO algorithm has both the advantages of high speed (i.e., low CPU time) and high solution quality (i.e., OFV). The results indicate that:

(i) The FPSO algorithm has better performance than the FCM algorithm in terms of CPU time and the solution quality for large sizes.
(ii) There are a few differences between two algorithms, when the size of problems is small.

Figure 1 summarizes the effect of varying the number of clusters for the different algorithms for Example 4. It is expected that the OFV should decrease when the number of clusters increases. This figure also shows that the FPSO algorithm consistently performs better than the FCM.

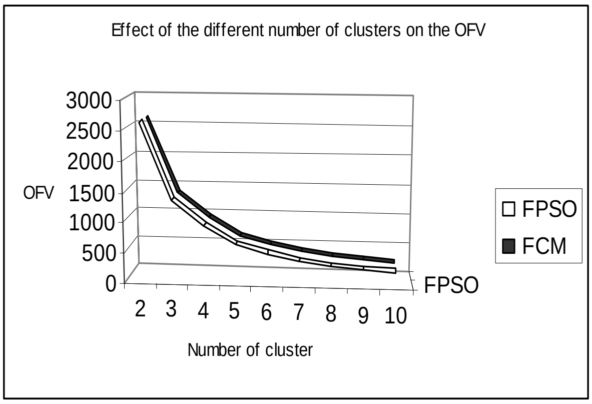

FIGURE 1. Effect of the different number of clusters on the OFV.

7. Conclusion

The fuzzy clustering problem is combinatorial by nature and hard to solve optimally in a reasonable time. In this paper, we have investigated the application of PSO to cluster data vectors by fuzzy considerations. It is shown how PSO can be used to find the cluster centers of a number of clusters and how a data point is related to a cluster. We have also presented an efficient hybrid method, called fuzzy particle swarm optimization (FPSO), which is based on the particle swarm optimization (PSO) and fuzzy c-means (FCM) algorithms, to solve the fuzzy clustering problem, especially for large problem sizes. Our proposed algorithm was compared with the fuzzy *c*-means (FCM) clustering algorithm using eight examples from the literature. It was shown that the performance of the PSO clustering algorithm can be improved further by seeding the initial swarm with the result of the *c*-means algorithm. Our experimental tests showed that the computational times for the FPSO method for the all examples were significantly lower than those for the FCM method and had higher solution quality in terms of the objective function value (OFV).

Acknowledgments. The authors are very grateful to the referees for their valuable comments and suggestions.

REFERENCES

[1] J. C. Bezdek and R. J. Hathaway, *Optimization of fuzzy clustering criteria using genetic algorithms,* Proceedings of the IEEE Conf. on Evolutionary Computation, Orlando, **2** (1994), 589-594.

[2] J. C. Bezdek, *Cluster validity with fuzzy sets,* Journal of Cybernetics, **3** (1974), 58-72.

[3] J. C. Bezdek, *Pattern recognition with fuzzy objective function algorithms,* New York, 1981.

[4] C. Y. Chen and F. Ye, *Particle swarm optimization and its application to clustering analysis,* Proceedings of the Int. Conf. on Networking, Sensing and Control, Taipei: Taiwan, 2004, 789-794.

[5] M. Dorigo and V. Maniezzo, *Ant system: Optimization by a colony of cooperating agents,* IEEE Transactions on Systems, Man, and Cybernetics B, **26** (**1**) (1996), 29-41.

[6] J. C. Dunn, *Fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters,* Journal of Cybernetics, **3** (1974), 32-57.

[7] R.C. Eberhart and Y. H. Shi, *Evolving artificial neural networks,* Proceedings of the Int. Conf. on Neural Networks and Brain, Beijing: P. R. China, Publishing House of Electronics Industry, PL5-PL13, 1998.

[8] T. Gu and B. Ddubuissonb, *Similarity of classes and fuzzy clustering,* Fuzzy Sets and Systems, **34** (1990), 213-221.

[9] J. Handl, J. Knowles and M. Dorigo, *Strategies for the increased robustness of ant–based clustering,* in: Engineering self-organizing systems, Heidelberg, Germany: Springer-Verlag, LNCS, **2977** (2003), 90-104.

[10] R. J. Hathaway and J. C. Bezdek, *Recent convergence results for fuzzy c-means clustering algorithms,* Journal of Classification, **5** (1988), 237-247.

[11] M. A. Ismail, *Soft clustering: Algorithms and validity of solutions,* Fuzzy Computing Theory Hardware and Applications, North Holland, 1988, 445-471.

[12] P. M. Kanade and L. O. Hall, *Fuzzy ant clustering by centroids,* Proceeding of the IEEE Conference on Fuzzy Systems, Budapest: Hungary, 2004, 371-376.

[13] P. M. Kanade and L. O. Hall, *Fuzzy ants as a clustering concept,* The 22nd Int. Conf. of the North American Fuzzy Information Processing Society (NAFIPS), Chicago, 2003, 227-232.

[14] L. Kaufman and P. Rousseeuw, *Finding groups in Data: Introduction to cluster analysis,* John Wily & Sons Inc., New York, 1990.

[15] J. Kennedy and R. C. Eberhart, *Particle swarm optimization,* Proceedings of the IEEE International Joint Conference on Neural Networks, **4** (1995), 1942-1948.

[16] J. Kennedy, R. C. Eberhart and Y. Shi, *Swarm intelligence,* San Mateo: Morgan Kaufmann, CA, 2001.

[17] F. Klawonn and A. Keller, *Fuzzy clustering with evolutionary algorithms,* Int. Journal of Intelligent Systems, **13 (10-11)** (1998), 975-991.

[18] J. G. Klir and B. Yuan, *Fuzzy sets and fuzzy logic, theory and applications,* Prentice-Hall Co., 2003.

[19] D. J. Newman, S. Hettich, C. L. Blake and C. J. Merz, *UCI Repository of machine learning databases,* http://www.ics.uci.edu/~mlearn/MLRepository.html, Department of Information and Computer Science, University of California, Irvine, CA, 1998.

[20] M. Omran, A. Salman and A. P. Engelbrecht, *Image classification using particle swarm optimization,* Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning, Singapore, 2002, 370-374.

[21] M. Roubens, *Pattern classification problems and fuzzy sets,* Fuzzy Sets and Systems, **1** (1978), 239-253.

[22] M. Roubens, *Fuzzy clustering algorithms and their cluster validity,* European Journal of Operation Research, **10** (1982), 294-301.

[23] T. A. Runkler and C. Katz, *Fuzzy clustering by particle swarm optimization,* IEEE Int. Conf. on Fuzzy Systems, Vancouver: Canada, July 16-21, 2006, 601-608.

[24] T. A. Runkler, *Ant colony optimization of clustering models,* Int. Journal of Intelligent Systems, **20 (12)** (2005), 1233-1261.

[25] E. H. Ruspini, *Numerical methods for fuzzy clustering,* Information Sciences, **2** (1970), 319-50.

[26] J. Tillett, R. Rao, F. Sahin and T. M. Rao, *Particle swarms optimization for the clustering of wireless sensors,* Proceedings of SPIE: Digital Wireless Communications V, **5100** (2003), 73-83.

[27] D. W. Van der Merwe and A. P. Engelbrecht, *Data clustering using particle swarm optimization,* Proceedings of the IEEE Congress on Evolutionary Computation, Canberra: Australia, 2003, 215-220.

[28] R. T. Yen and S. Y. Bang, *Fuzzy relations, fuzzy graphs and their applications to clustering analysis,* in: L. Zadeh et al. (Eds.), Fuzzy Sets and their Applications to Cognitive and Decision Processes, New York: Academic Press, Inc., 1975, 125-150.

[29] N. Zahid, M. Limoun and A. Essaid, *A new cluster-validity for fuzzy clustering,* Pattern Recognition, **32** (1999), 1089-1097.

[30] C. J. Zhang, Y. Gao, S. P. Yuan and Z. Li, *Particle swarm optimization for mobile and hoc networks clustering,* Proceeding of the Int. Conf. on Networking, Sensing and Control, Taipei: Taiwan, 2004, 372-375.

[31] H. J. Zimmermann, *Fuzzy set theory and its applications,* Lower Academic Publishers, 1996.

E. MEHDIZADEH*, DEPARTMENT OF INDUSTRIAL ENGINEERING, SCIENCE & RESEARCH BRANCH, ISLAMIC AZAD UNIVERSITY, TEHRAN, IRAN

E-mail address: `mehdizadeh@qazviniau.ac.ir`

S. SADI-NEZHAD, DEPARTMENT OF INDUSTRIAL ENGINEERING, SCIENCE & RESEARCH BRANCH, ISLAMIC AZAD UNIVERSITY, TEHRAN, IRAN

E-mail address: **sadinejad@hotmail.com**

R. TAVAKKOLI-MOGHADDAM, DEPARTMENT OF INDUSTRIAL ENGINEERING, COLLEGE OF ENGINEERING, UNIVERSITY OF TEHRAN, TEHRAN, IRAN

E-mail address: **tavakoli@ut.ac.ir**

* CORRESPONDING AUTHOR