AN EFFICIENT HYBRID PARTICLE SWARM OPTIMIZATION FOR DATA CLUSTERING

Dr. M.Seetha[1], G. Malini Devi[2], Dr. K.V.N.Sunitha[3]

[1]Professor, Dept. of CSE, GNITS, Hyderabad-8, India
smaddala2000@yahoo.com
[2]Asst. Professor, Dept. of CSE, GNITS, Hyderabad-8, India
malini_g12@yahoo.co.in
[3]Principal, BVRIT for womens, Hyderabad., India
k.v.n.sunitha@gmail.com

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 its performance was improved by seeding the initial swarm with the result of the c-means algorithm. The fuzzy c-means, PSO and FPSO algorithms are compared using the performance factors of object function value (OFV) and CPU execution time. It was ascertained that the computational times for the FPSO method outperforms the FCM and PSO method and had higher solution quality in terms of the objective function value (OFV).

KEYWORDS

Fuzzy clustering, Fuzzy c-means, PSO, FPSO, objective function value (OFV).

1. INTRODUCTION

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 clusters [1]. 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. In fuzzy clustering, data elements can belong to more than one cluster, and associated with each element is a set of membership levels. These indicate the strength of the association between that data element and a particular cluster. Fuzzy clustering is a process of assigning these membership levels, and then using them to assign data elements to one or more clusters. 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.

DOI : 10.5121/ijdkp.2012.2602

The fuzzy clustering problem arrives 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 [15]. 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. Fuzzy clustering algorithms are partitioning methods that can be used to assign objects of the data set to their clusters [9]. 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 fuzzy c-means (FCM) algorithm is the most widely used in applications [15]. The problem is by nature a combinatorial optimization problem 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 [5].

Swarm intelligence (SI) describes the collective behavior of decentralized, self-organized systems, natural or artificial. The concept is employed in work on artificial intelligence. The expression was introduced by Gerardo Beni and Jing Wang in 1989, in the context of cellular robotic systems. Swarm Intelligence systems are typically made up of a population of simple agents or boids interacting locally with one another and with their environment. The agents follow very simple rules, and although there is no centralized control structure dictating how individual agents should behave local, and to a certain degree random, interactions between such agents lead to the emergence of "intelligent" global behavior, unknown to the individual agents [6].

Particle swarm optimization (PSO) is a global optimization algorithm for dealing with problems in which a best solution can be represented as a point or surface in an n-dimensional space [8]. Hypotheses are plotted in this space and seeded with an initial velocity, as well as a communication channel between the particles. Particles then move through the solution space, and are evaluated according to some fitness criterion after each time step. Over time, particles are accelerated towards those particles within their communication grouping which have better fitness values. The main advantage of such an approach over other global minimization strategies such as simulated annealing is that the large numbers of members that make up the particle swarm make the technique impressively resilient to the problem of local minima [10].

A fuzzy clustering problem is, in fact, a combinatorial optimization problem [1] 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 and optimized the hard c-means method with a genetic algorithm [2]. In addition, ant colony optimization (ACO) [4] has been successfully applied to clustering problems. PSO has been applied to image clustering [4], network clustering and clustering analysis, and data clustering [3].

Fuzzy clustering algorithm along with swarm intelligence has been implemented on the data sets. This paper emphasizes on the data clustering of large datasets within the given time limit with less fuzziness. To classify the datasets into clusters, three algorithms namely Fuzzy C-means (FCM), Particle Swarm Optimization (PSO) and Fuzzy PSO (FPSO), K-MEANS, KPSO, KPSOK have been used [14]. Along with these algorithms their dynamic versions are also been used. By taking the values of the datasets, the fuzzy c-means and fuzzy particle swarm optimization algorithms are implemented and analyzed with different ranges of clustering[5].By comparing the level of clustering and time taken to form cluster by the 6 algorithms conclusion can be made which is the best algorithm among them. Each algorithm calculates performance index, i.e. finds cluster centers Fitness, Overall Purity, Distance, Object function value (OFV).

The degree of clustering for each algorithm is calculated using OFV thus the lower the OFV the better the clustering. OFVs as well as CPU execution time is calculated over 10 simulation runs. The four data sets namely Iris plant, Zoo data set used in the study is taken from UCI machine repository [13]. The Iris plant data set has 150 data points containing 50 instances of each of three types of iris plant[17]. The Zoo dataset 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.

2. FUZZY C-MEANS CLUSTERING

The FCM algorithm is one of the most widely used fuzzy clustering algorithms [1]. The FCM algorithm attempts to partition a finite collection of elements X= $\{x_1, x_2, ..., x_n\}$ into a collection of c fuzzy clusters with respect to some given criterion. FCM aims to minimize an objective function. Here, each point has a degree of belonging to clusters, as in fuzzy logic, rather than belonging completely to one cluster. Thus, points on the edge of a cluster may be in the cluster to a lesser degree than points in the center of cluster.

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 [7].

$U = [\mu_{ik}]_{c\times n}$: Matrix representing classification.

μ_{ik} : Membership degree of data point k with respect to cluster i.

J (p): Object function value

m : fuzzy parameter=2

V_i : Cluster center of cluster i.

X_k : vector of data point.

$\|x_k - v_i\|^2$: Euclidean distance

The classification result can be expressed in terms of matrix *U*= [*µik*] *c*×*n*, where *µik* is the membership degree of data point *k* with respect to cluster *i* and satisfies the following conditions:

$0 \le \mu_{ik} \le 1$

i=1, 2… c k=1, 2… n

$\sum_{k=1}^{c} \mu_{ik} = 1$ k=1, 2… n

$0 < \sum_{k=1}^{n} \mu_{ik} \le n$ i=1, 2… n

The Objective Function Value (OFV) of the clustering algorithm is:

$$J(P) = \sum_{k=1}^{n} \sum_{i=1}^{c} [\mu_{ik}]^m \|x_k - v_i\|^2 \quad (1)$$

Cluster center

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

Membership degree:

$$\mu_{ik}(t+1) = \left[\sum_{j=1}^{c} \left[\|x_k - V_i^{(t)}\|^2 / \|x_k - V_i^{(t)}\|^2\right]^{1/m-1}\right]^{-1} \qquad (3)$$

Where, μ_{ik} is Membership degree of data point k with respect to cluster i.

A fuzzy pseudo-partition is often called a fuzzy c-partition, where c is the number of fuzzy classes in the partition [20]. The fuzzy c-means (FCM) clustering method is based on fuzzy c-partitions developed by Bezdek 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 å and the distance function are given and proceeds as follows [21]:

1) Let t=0. Select an initial fuzzy pseudo-partition p(0).
2) Calculate the c cluster centers V1(t) , ..., Vc(t) using (2) for p(t) and the chosen value of m.
3) Compute µi(t+1) using (3) and update p(t+1).
4) Compare p(t) and p(t+1). If p (t +1) − p (t) ≤ Є , 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.

It has been ascertained that the FCM algorithm fails to find the global optimum if the data sets are very high dimensional or contain severe noise points. Because of its less accurate clustering quality, FCM tends to be slower [15]. Hence the hybrid form of FPSO gives better performance [19].

3. PROPOSED HYBRID PARTICLE SWARM OPTIMIZATION

In fuzzy clustering, a single particle represents a cluster center vector. In other words, each particle $part_l$ is constructed as follows:

$part_l = (v_1, v_2, \ldots\ldots v_i, \ldots\ldots\ v_c)$

Where l represents the l-th particle and l=1, 2... n_ particle and Vi 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, gives a vector of cluster centers. The position of vector $part_l$ is determined for every particle, updated and then the position of cluster centers is changed based on the particles.

The following notation has been used to implement particle swarm optimization.

1) The position of the i-th particle of a swarm of size n, is represented by the D-dimensional vector,

   xi=(xi1, xi2, ..., xiD).

2) The best previous position (i.e., the position giving the best function value) of the i-th particle is recorded and represented by

pi= (ρ*i*1, ρ*i*2, ..., ρ*iD*).

3) The position change (velocity) of the i-th particle is

   Veli=(*Veli*1, *Veli*2, ...,*VeliD*).

4) The position of the best particle of the swarm (i.e., the particle with the smallest function value) is denoted by index *pg*.

5) The particles are then manipulated according to the following equations.

$Vel_{id}(t+1)= \chi\{w\ Vel_{id}\ (t)+c_1\varphi_1[\rho_{id}(t)-x_{id}(t)]+ c_2\varphi_2\ [\rho_{gd}(t)-x_{id}(t)]\}$ (4)

$x_{id}(t+1)=x_{id}(t)+Vel_{id}(t+1)$ (5)

where d=1,2,…..D and i=1,2,…..n.

w: inertia weight=0.72

c1,c2 : positive acceleration constants=1.49

φ_1, φ_2 : random numbers

χ : constriction factor=1.0

The following algorithm is now used to find the best position .

1) for each particle:
   Initialize particle: *xi*.
2) Do:
   For each particle:
      Calculate fitness value: *pi*
      If the fitness value is better than the best
      fitness value (*pg*) in History
         Set current value as the new fitness value.
   End.
3) For each particle:
   Find in the particle neighborhood, the particle with the best fitness
   Calculate particle velocity: *Veli,* using
   equation (4)
   Apply the velocity constriction.
   Update particle position: *xi* , using equation (5).
4) Apply the position constriction.

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 [11]. 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 [10]. At the initial stage, the FPSO algorithm executes the c-means algorithm once [12]. This stage terminates according to one of two stopping criteria:

(1) The maximum number of iterations; or

(2) $p(t+1) - p(t) \leq \varepsilon$.

The result is then considered a particle in the swarm; the other particles are initialized randomly.

The following notation has been used to implement hybrid particle swarm optimization [11].

n: Number of data points

c: Number of cluster centers

Vl(*t*): Position of the l-th particle at stage t

Vell(*t*): Velocity of the l-th particle in stage t

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

ρ*l*(*t*): Best position found by the l-th particle at stage t

ρ*g*(*t*): Best position found by all particles at stage t

P(*t*): Fuzzy pseudo partition at stage t

μ*ik*(*t*): Membership function of the k-th data point with respect to into the i-th cluster at stage t

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

1. Let t=0. Select the initial parameters, such as the number of cluster centers c, the initial velocity of particles, c1, c2, w, χ, a real number m(1, ∞), and a small positive number ε for the stopping criterion. 1.The initial position of the particles is that obtained by the FCM.
2. Calculate μik(t) for all particles (i=1,2,…,c and k=1,2,…,n) by Equation (3) and update p(t+1).
3. For each particle, calculate the goodness of fitness using Equation (1).
4. Update the global and the local best positions.
5. Update Vell(t) and Vl(t) (l=1,2,…,n_particle) as given by Eq. (2) and (3).
6. Go to Step 2. Compare p (t) and p (t+1) If $p(t+1) - p(t) \le \varepsilon$, then stop; otherwise, continue to Step 3.

4. RESULTS AND DISCUSSIONS

The main objective of this study is to assess the relative performance of the proposed FPSO with respect to the FCM modification. The performance is measured by the objective function value in Eq. (1) and the CPU execution time. The objective function value is analyzed for FCM and proposed FPSO by varying the number of clusters for both the datasets (i.e Iris and Zoo datasets). An analysis is done to find the better algorithm between FCM and FPSO using OFV with increased number of clusters. Lower the OFV for an algorithm, the better the clustering. If time taken by any algorithm to complete the execution is less then that algorithm can be considered to be best.

The table 1 shows the variation in OFV with increase in number of clusters for FCM and FPSO algorithm.

Table 1: Variation of OFV for FCM and FPSO algorithms with Number of clusters for Iris data set

No. of Clusters	OFV	
	FCM	FPSO
3	359.73	-43.18
5	166.2	-107.65
7	99.24	-151.51
9	74.58	-169.17
11	65.82	-186.41

The figure 1 shows the variation in OFV with increase in number of clusters for FCM and FPSO algorithm.

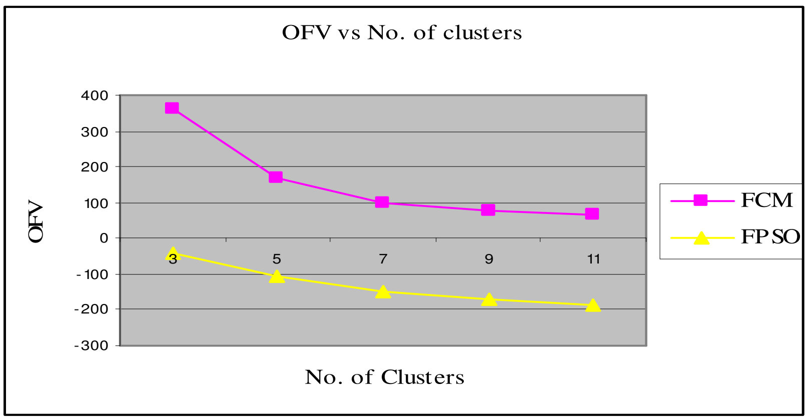

Figure 1: Variation of OFV for FCM and FPSO algorithms with Number of clusters for Iris data set

As the number of clusters is increased the OFV is decreased (Figure 1). It is observed that the OFV is optimized with proposed hybrid PSO when compared to FCM.

Table 2: Variation of Execution time for FCM and FPSO algorithms with Number of clusters for Iris data set

No. of Clusters	Execution time	
	FCM	FPSO
3	0.033	0.011
5	0.09	0.015
7	0.095	0.053
9	0.098	0.055
11	0.099	0.062

The figure 2 shows the variation in Execution time with increase in number of clusters for FCM and FPSO algorithm.

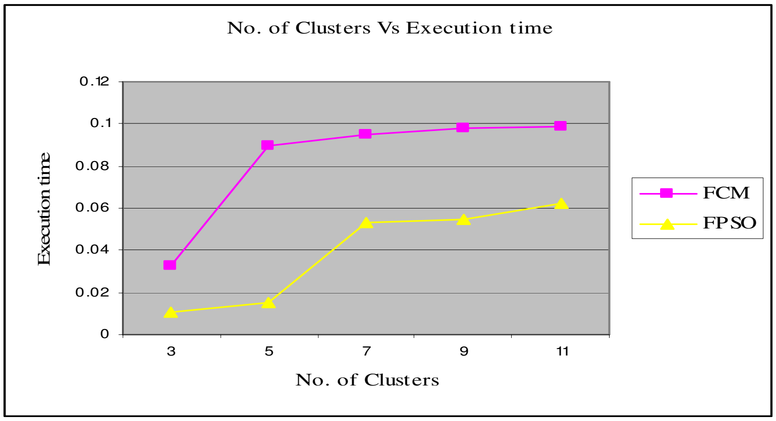

Figure 2: Variation of Execution time for FCM and FPSO algorithms with Number of clusters for Iris data set

The CPU execution time decreases with the increase in number of clusters (Table 2). The CPU execution time is less when compared to FCM. Hence, the Fuzzy Particle Swam Optimization is better than the Fuzzy C Means and Particle Swam Optimization.

Table 3: Variation of OFV for FCM & FPSO algorithms with Number of clusters for Zoo data set

No. of Clusters	OFV	
	FCM	FPSO
7	151.83	-25.5
9	117.92	-95.33
11	99.63	-117.88
13	69.68	-155.2
15	57.93	-163.02

The similar results are observed between the FCM and FPSO using OFV and CPU execution time with Zoo dataset (Table 3 and Figure 2).

The figure 3 shows the variation in OFV with increase in number of clusters for FCM and FPSO algorithm.

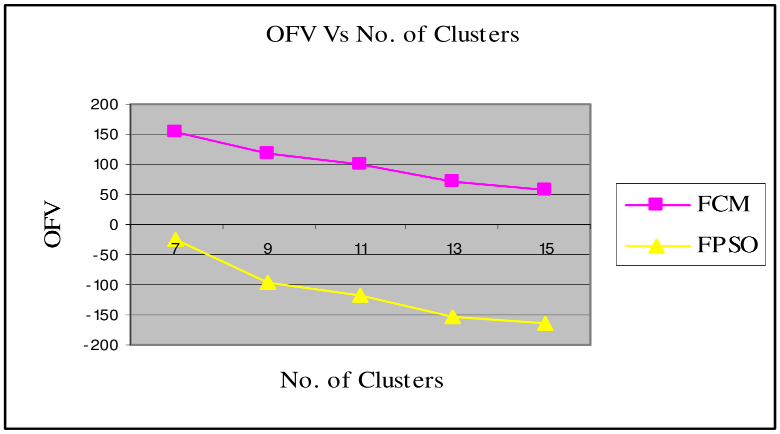

Figure 3: Variation of OFV for FCM and FPSO algorithms with Number of clusters for Zoo data set

Table 4: Variation of Execution time for FCM and FPSO algorithms with Number of clusters for Zoo data set

No. of Clusters	Execution time	
	FCM	FPSO
7	0.092	0.015
9	0.13	0.06
11	0.29	0.07
13	0.34	0.07
15	0.44	0.07

The figure 4 shows the variation in Execution time with increase in number of clusters for FCM and FPSO algorithm.

Figure 4: Variation of Execution time for FCM and FPSO algorithms with Number of clusters for Zoo data set

A general thumb rule is that a clustering with lower OFV and lower CPU time is preferable. The effectiveness of these algorithms was reiterated by the following observations tested on the 2 data sets, Iris and Zoo and analyzed as below (Tables 5 and 6).

Table 5: Variation of OFV for FCM, PSO & FPSO algorithms for Iris & Zoo data set

	FCM	PSO	FPSO
OFV FOR IRIS	359.7308	-111.625	-169.172
OFV FOR ZOO	151.836	-118.902	-163.02

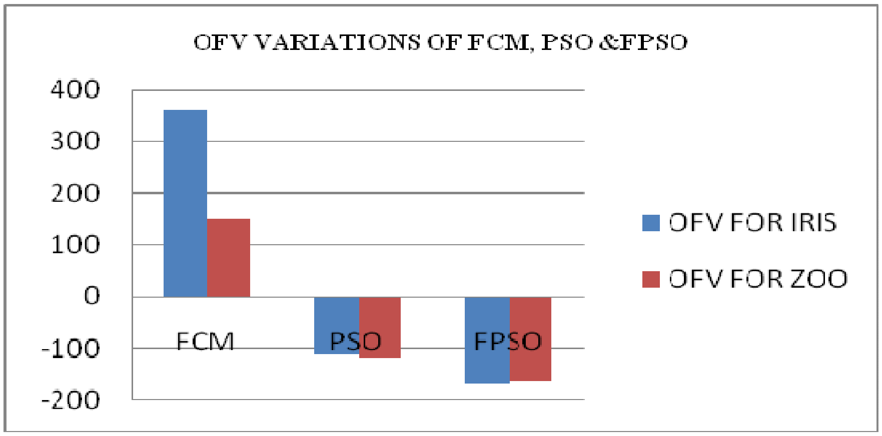

Figure 5: Variation of OFV for FCM, PSO & FPSO algorithms for Iris & Zoo data set

Table 5: Variation of Execution time for FCM, PSO & FPSO algorithms for Iris & Zoo data set

	FCM	PSO	FPSO
EXEC TIME FOR IRIS	0.011	0.034	0.033
EXEC TIME FOR ZOO	0.0925	0.0247	0.0247

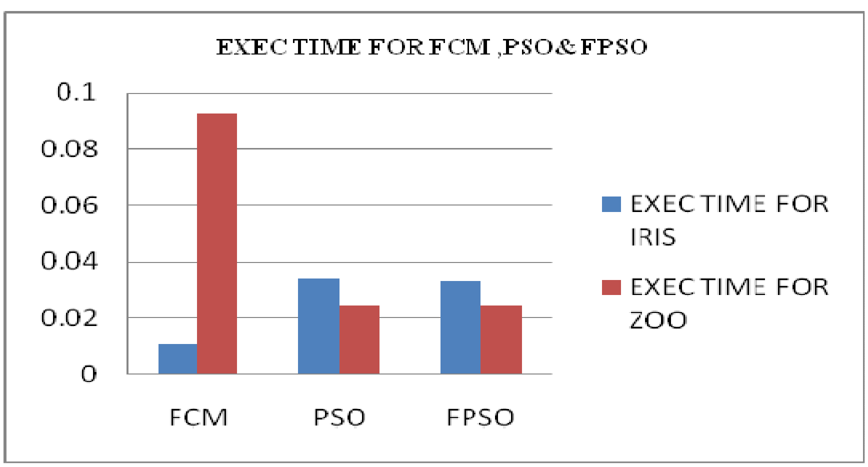

Figure 6: Variation of Execution time for FCM, PSO & FPSO algorithms for Iris & Zoo data set

It has been ascertained that the proposed FPSO algorithm has both the advantage of high speed (i.e. low CPU time) and high solution quality (i.e. OFV).The results indicates that the FPSO algorithm has better performance than FCM algorithm in terms of CPU time and solution quality for both the Iris and Zoo datasets of large sizes.

5. CONCLUSIONS

The fuzzy clustering problem is combinatorial by nature and hard to solve optimally in a reasonable time. This paper investigated the application of PSO to cluster data vectors by fuzzy considerations. 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 was proposed to solve the fuzzy clustering problem, especially for large problem sizes. The proposed algorithm was compared with the fuzzy *c*-means (FCM) clustering for performance using objective function value and CPU execution time. 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. It was ascertained that the computational times for the FPSO method for the Iris and Zoo datasets were significantly lower than those for the FCM method and had higher solution quality in terms of the objective function value (OFV). The study can be extended further by combining other fuzzy clustering algorithms like K-means and then the best hybrid algorithm can be derived for large sized data set and using other measures like intensity and connectivity.

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, pp589-594.

[2] J. C. Bezdek, Cluster validity with fuzzy sets, Journal of Cybernetics, 3, 1974, pp 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, pp 789-794.

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

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

[7] F. Klawonn and A. Keller, Fuzzy clustering with evolutionary algorithms, Int. Journal of Intelligent Systems, 13 2010, pp 975-991.

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

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

[10] 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.

[11] E. mehdizeded, S. Sadinezhad, R. Tavakkolimoghaddam, Optimization of Fuzzy Criteria by a Hybrid PSO and Fuzzy C- Means Clustering Algorithm, Iranian Journal of Fuzzy Systems, pp. 1-14, (2008).

[12] Y. Shi and R. Eberhart, (1998). "A Modified Particle Swarm Optimizer", Proceedings of IEEE International Conference on Evolutionary Computation (ICEC'98), pp.69-73, Anchorage.

[13] 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.

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

[15] J. Demsar, “Statistical comparisons of classifiers over multiple data sets,”J. Mach. Learn. Res.,vol. 7, pp. 1–30, 2006.

[16] S. Miyamoto, Introduction to Cluster Analysis: Theory and Application of Fuzzy Clustering, Morikita-Shuppan, Tokyo, 1999 (Japanese).

[17] UCI Repository for Machine Learning Databases retrieved from the World Wide Web: http://www.ics.uci.edu/~mlearn/MLRepository.htm.

[18] Jain, A. K. And Dubes, R. C. (1988). Algorithms for Clustering Data. Prentice-Hall advanced reference series. Prentice-Hall, Inc., Upper Saddle River, NJ.

[19] Anderberg, M. R., Cluster Analysis for Application, Academic Press, New York, NY, U.S.A. (1973).

[20] Selim, S. Z. and Ismail, M. A., “K-means Type Algorithms: A Generalized Convergence Theorem and Characterization of Local Optimality,” IEEE Trans. Pattern Anal. Mach. Intell. Vol. 6, pp. 81-87 (1984).

[21] Wu, K.-L. and Yang, M.-S., “Alternative C-means Clustering Algorithms,” Pattern Recognition, Vol. 35, pp. 2267-2278 (2002).

AUTHORS

Dr. M.Seetha She had completed Ph.D in Computer Science and Engineering in the area of image processing in December 2007 from Jawaharlal Nehru Technological University, Hyderabad, India. She has teaching experience of 18 years in various engineering colleges. She is guiding 10 Ph.D scholars and her research interest includes image processing, neural networks, computer networks and data mining. She had published more than 50 papers in refereed journals and in the proceedings of National/International Conferences and Symposiums. She is the recipient of the AICTE Career Award for Young Teachers (CAYT) in FEB, 2009, and received the grant upto 10.5 lakhs over a period of three years by AICTE, INDIA. She is the reviewer for the International Journals and International conferences. She holds the Life Membership of Indian Society for Technical Education (ISTE) and The Institution of Electronics and Telecommunication Engineers (IETE).

G.Malini Devi did her B.Tech CSE from Jawaharlal Nehru Technological University, Hyderabad. M.Tech Computer Science from Jawaharlal Nehru Technological University, Ananthapur in 2006,Pursuing Ph.D from JNTUH. She had completed 6 years of Teaching experience in G.N.I.T.S. She had published 3 papers

Dr K.V.N.Sunitha did her B.Tech ECE from Nagarjuna University, M.Tech Computer Science from REC Warangal and Ph.D from JNTUH in 2006. She has 21 years of Teaching Experience, working as HOD, CSE Dept in G.N.I.T.S from the inception of the CSE Dept since 2001. She is a recipient of Academic Excellence award by GNITS in 2005. She has received "Best computer Science Engineering Teacher award" by ISTE (Indian society for Technical education) in Feb 2008. She is guiding 12 PhDs & published more than 65 papers in International & National Journals and conferences. She has authored three text books. She is Board of Studies member for many other Engg. colleges. She has received funding for her research proposal from CSI Computer society of India. Her auto biography is listed in Marquis who is who in the world ,28th edition 2011.She is fellow of Institution of Engineers, Sr.Member for IEEE & Life member for CSI.