This article can be cited as A. Kumar, D. Kumar and S. K. Jarial, A Hybrid Clustering Method Based on Improved Artificial Bee Colony and Fuzzy C-Means Algorithm, vol. 15, no. 2, pp. 40-60, 2017.

A Hybrid Clustering Method Based on Improved Artificial Bee Colony and Fuzzy C-Means Algorithm

Ajit Kumar[1], **Dharmender Kumar**[2] and **S.K.Jarial**[3]

[1]Department of Computer Science and Engineering,
Deenbandhu Chhotu Ram University of Science and Technology, Murthal (India);
Email: ajit.hisar@gmail.com

[2]Department of Computer Science and Engineering,
Guru Jambheshwar University of Science and Technology, Hisar (India);
Email: dharmindia24@gmail.com

[3]Department of Mechanical Engineering,
Deenbandhu Chhotu Ram University of Science and Technology, Murthal (India);
Email: s.jarial@rediffmail.com

ABSTRACT

Data clustering is an important data mining technique to create groups (clusters) of objects, in such a way that objects in one cluster are very similar and objects in different clusters are quite distinct. Fuzzy c-means (FCM) algorithm is a popular data clustering method that works according to the fuzzy membership between data points and cluster centers. However, it has possibilities of convergence to local minima. Artificial Bee Colony (ABC) algorithm is a swarm based algorithm inspired by intelligent foraging behavior of honey bees. In order to make use of merits of both algorithms, a hybrid algorithm (IABCFCM) based on improved ABC and FCM algorithms is proposed in this paper. The IABCFCM algorithm helps the FCM clustering escape from local optima and provides better experimental results on the well known data sets.

Keywords: Artificial bee colony, F-measure, fuzzy c-means, objective function value, tournament selection.

Mathematics Subject Classification: 62H30

Computing Classification System: I.5.3

1. INTRODUCTION

Clustering is a form of indirect data mining, as the goal is to find the relationships among all the variables in contrast to direct data mining, where some variables are pointed out as targets (Gan et al., 2007). It can be regarded as a form of unsupervised classification i.e. labeling of objects does not rely on predefined classes, rather it derives from the data itself. Objects similar to each other are identified in a cluster or group of a data set (different from those in other clusters/groups) using clustering techniques. Euclidean distance is the most used similarity metric; lesser the distance, more similarity is there between two objects or two clusters. Clustering is widely used in numerous applications, including market research, pattern recognition, data analysis, document retrieval, image segmentation, artificial intelligence, bioinformatics, financial investment, data compression, web mining, machine learning and image processing (Jain et al., 1999; Han and Kamber, 2006; Kumar and Sahoo, 2014).

Clustering is categorized as hard clustering and fuzzy clustering. Hard clustering is sub-categorized as hierarchical clustering and partitional clustering (Han and Kamber, 2006). In hierarchical clustering objects are gradually (dis)assembled into clusters whereas objects are iteratively relocated to form clusters of proper convex shapes in partitional clustering. Partitional algorithms divide the set of data in clusters by iteratively relocating objects without hierarchy. The clusters are gradually improved to ensure high quality of clustering. The popular k-means algorithm generates partitions of an N-dimensional population such that each partition is having small within-class variance. In k-means, each cluster has a center called mean and attempt is made to minimize its objective function (a square error function). The k-means algorithm also has some limitations: dependence on initialization of cluster centers, sensitivity to outliers, non-guaranteed optimal solutions, formation of unbalanced clusters.

Inspired by the Zadeh's idea of fuzzy theory, fuzzy c-means clustering algorithm was introduced by Bezdek (1981). This widely used algorithm represents the similarity of a sample point to more than one clusters using membership function (value between 0 and 1). However, it may trap to local optima due to randomly selected center points. A number of algorithms based on swarm, insects and natural phenomena have been proposed during recent years to solve clustering problems. These include simulated annealing (Selim and Alsultan, 1991), tabu search (Al-Sultan, 1995), genetic algorithm (Murthy and Chowdhury, 1996), ant colony optimization based on ant colonies (Shelokar et al., 2004), particle swarm optimization based on fish schooling/bird flocking (Chen and Ye, 2004), honey bee mating optimization algorithm (Fathian et al., 2007), cat swarm optimization based on behavior of cats (Santosa and Mirsa, 2009), artificial bee colony (Zhang et al., 2010), teacher learning based optimization (Satapathy and Naik, 2011), invasive weed optimization (Chowdhury et al., 2011), black hole optimization (Hatamlou, 2013), new artificial fish swarm algorithm (Yazdani et al., 2013), charge system search algorithm (Kumar and Sahoo, 2014) and many more.

Artificial Bee Colony is a population based algorithm introduced by Karaboga (2005), which is inspired by the intelligent foraging behavior of honey bees. ABC is good in exploration besides having simple and robust nature. ABC has successfully been used in wide range of applications by making use of special properties such as foraging, exploration and exploitation of food sources, information exchange, optimal nest site selection etc. However, it has shortcomings such as slow convergence and poor exploitation in solving complex problems. In order to solve these problems, a number of variants of ABC have been proposed in the history. A few of the important as well as efficient variants are (Zhu and Kwong, 2010; Barnharnsakun et al., 2011; Karaboga and Akay, 2011; Akay and Karaboga, 2012; Gao and Liu, 2012; Kiran and Gunduz, 2012; Luo et al., 2013; Kiran and Findik, 2015). In this paper, we make use of special properties such as foraging, exploration and exploitation of food sources as well as information exchange of ABC in order to overcome the limitations of FCM. A hybrid data clustering algorithm based on improved ABC and FCM algorithms, called IABCFCM, is proposed. The experimental results on several data sets prove the better performance of IABCFCM algorithm as compared to ABC and FCM algorithms.

The remainder of this paper comprises of the related work on hybrid clustering, fuzzy c-means algorithm, principles of basic ABC and improved ABC algorithm, proposed hybrid clustering algorithm, experimental results and finally conclusion.

2. RELATED WORK

Fuzzy C-Means is a popular and effective clustering algorithm, but may get stuck at local optima based on initial center points. In the past, several clustering algorithms as well as their hybrid approaches have been developed in order to minimize the limitations of FCM. The ABC algorithm to solve clustering problems was developed by Zhang et al. (2010) that adopted Deb's constraint handling method instead of greedy selection process to tackle infeasible solutions. The algorithm proves its performance in terms of quality of solutions and number of function evaluations. Karaboga and Ozturk (2010) applied ABC in fuzzy clustering. Various tests performed on medical data sets show the success of ABC in fuzzy clustering. Su et al. (2012) introduced some modifications in ABC such as variable length strings, mutation operations and scheme for candidate solutions to generate VABC. The proposed algorithm in combination with FCM was used to find fuzzy partitions with accuracy and proper convergence. Malaki et al. (2012) provided a Fuzzy C-means Bee method by making use of ABC algorithm to find the promising solutions of global optimum and then fuzzy C-means to sharply converge to the global solution. The hybrid method was successfully tested on three evaluation measures. Lin et al. (2013) developed a Fuzzy Artificial Bee Colony System by hybridizing ABC and FCM algorithms. The modified method was successfully tested for segmentation of medical images. Beloufa and Chikh (2013) modified ABC to enhance the exploration and exploitation phases by employing a blended crossover operator. The modified algorithm was used to create an effective fuzzy classifier and applied in diagnosis of diabetes disease. Krishnamoorthi and Natarajan (2013) proposed a modified ABC algorithm that incorporates the FCM operator in scout bee phase of ABC algorithm. The performance of the hybrid method was tested on three data sets and found significant results in terms of the quality of solutions and the execution time.

Many researchers have successfully implemented fuzzy systems in a wide variety of real-life applications. Lam et al. (2000) performed the stability and robustness analysis of an uncertain multivariable fuzzy control based on single-grid-point approach. Precup and Preitl (2004) developed fuzzy control systems that can be very effective in real world plants. Precup et al. (2007) proposed new design method for Takagi-Sugeno proportional-integral-fuzzy controllers. The new method ensures the maximum imposed/desired sensitivities for the designed fuzzy control systems and is found to be effective for a large number of industrial applications. Palanisami and Selvan (2009) proposed an approach for clustering of high dimensional data set into a number of fuzzy partitions. Martin et al. (2009) have also used genetic algorithm during optimal tuning of a linear controller and successfully applied to control a high-performance drilling process. Moallem et al. (2015) also proposed a genetic algorithm inspired fuzzy system that could overcome the complexity of the face detection problems. A parallel genetic algorithm incorporating the use of deterministic and random moves has been used to solve the open-shop scheduling problem (Ghosn et al., 2016). Precup et al. (2014) proposed a novel adaptive charged system search algorithm for the optimal tuning of the fuzzy controllers for servo systems. The algorithm was successfully applied to the nonlinear control of a class of servo systems. A Fuzzy distance based approach method is also proposed for optimal ranking and selection of commercial off-the-shelf components of an e-payment system (Garg et al., 2016).

3. FUZZY C-MEANS CLUSTERING PROBLEM

Fuzzy clustering problem can be considered as a combinatorial optimization problem. On the basis of successful results obtained by swarm based methods in solving various engineering problems, it may be stated that these methods may also be applied to find optimal solutions in fuzzy clustering. In fuzzy clustering, an object may belong to one or more clusters with certain degree of membership ranging from 0 to 1; hence fuzzy clustering is, in fact, a class of partitioning clustering. Fuzzy c-means is a famous algorithm that has been successfully applied to clustering applications. The algorithm uses the well known objective function for clustering i.e. within-group sum of squared errors. A brief introduction on fuzzy c-means clustering algorithm is given as under (Chen et al., 2014):

$X = \{x_1, ..., x_N\}$: the data to be clustered.

$V = \{c_1, ..., c_k\}$: the set of cluster centers.

U : fuzzy membership function matrix with k rows and N columns.

u_{ij} : degree of belongingness of j^{th} data point to the i^{th} cluster, value between the interval of [0,1].

m : fuzzy index, any real number greater than 1.

Basic algorithm for FCM clustering:

1. Initialize the algorithm by choosing number of clusters $c\,(2 \le c \le N)$, fuzzy index $m\,(m > 1)$ and maximum number of iterations.
2. Randomly choose the initial centers.
3. Compute u_{ij} using equation (1) as given below

$$u_{ij} = \begin{cases} if\,(d(c_p - x_j) \neq 0) & \left[\sum_{p=1}^{k} \left(\dfrac{d(c_i - x_j)}{d(c_p - x_j)} \right)^{\frac{2}{m-1}} \right]^{-1} \\ if\left(d\left(c_p - x_j\right) = 0\right) & \begin{Bmatrix} if\,(i = p)1 \\ if\,(i \neq p)0 \end{Bmatrix} \end{cases} \tag{1}$$

4. Recalculate the centers using equation (2) as follows

$$c_i = \frac{\sum_{j=1}^{N} u_{ij}^m x_j}{\sum_{j=1}^{N} u_{ij}^m}, i = 1, ..., k \tag{2}$$

5. Repeat steps 3-4 predefined number of iterations until $\left| u_{ij}^{(t+1)} - u_{ij}^{(t)} \right| < \varepsilon$ for all i and j, value of ε between 0 and 1, t represents the iteration.
6. Calculate the objective function value using equation (3) as given by

$$J_m(U, V, X) = \sum_{i=1}^{k} \sum_{j=1}^{N} u_{ij}^m d^2(c_i, x_j) \tag{3}$$

where $d^2(c_i - x_j)$ shows the similarity between c_i and x_j and is given by equation (4).

$$d^2(c_i - x_j) = \left\| c_i - x_j \right\|^2 \tag{4}$$

4. INITIAL ABC ALGORITHM

ABC is a population based optimization algorithm which is iterative in nature. Basically, ABC consists of five phases: Initialization phase, Employed bee phase, Probabilistic selection phase, Onlooker bee phase and Scout bee phase. Bees going to a food source already visited by them are employed bees while the bees looking for a food source are unemployed. Scout bees carry out search for new food sources and onlooker bees wait for the information from employed bees for food sources. The information exchange among bees takes place through waggle dance. There is one employed bee for every food source. In this way, exploitation process is performed by employed and onlooker bees whereas scouts perform exploration of search space. The details of ABC algorithm are given as under:

i) Initialization phase

The locations of food sources are randomly initialized within the range of boundaries according to equation (5)

$$x_{ij} = x_j^{\min} + rand(0,1)\left(x_j^{\max} - x_j^{\min}\right) \tag{5}$$

where $i = 1,...,SN$ and $j = 1,...,D$. SN indicates the number of food sources and taken as half of the bee colony, D is dimension of the problem, x_{ij} represents the parameter for i^{th} employed bee on j^{th} dimension, $x_j^{\max}$ and $x_j^{\min}$ are upper and lower bounds for x_{ij}.

ii) Employed bee phase

Each employee bee is assigned to the food source for further exploitation. The resulting food source is generated according to equation (6) as given by:

$$v_{ij} = x_{ij} + \phi(x_{ij} - x_{kj}) \tag{6}$$

where k is a neighbor of i, $i \neq k$, ϕ is a random number in the range [-1,1] to control the production of neighbor solutions around x_{ij}, v_{ij} is the new solution for x_{ij}. The fitness of new food source is now calculated using equation (7) as below:

$$fit_i = \begin{cases} \dfrac{1}{1+f_i} & , f_i \geq 0 \\ 1+abs(f_i), & f_i < 0 \end{cases} \tag{7}$$

where f_i is the objection function associated with each food source and fit_i is the fitness value. A greedy selection is performed on x_{ij} and v_{ij} i.e. original and new food sources to choose better one according to its fitness value.

iii) Probabilistic selection phase

For each food source a probability value is calculated using equation (8) as given below, and an onlooker bee selects the food source according to this value.

$$p_i = \frac{fit_i}{\sum_{j=1}^{N} fit_j} \tag{8}$$

where fit_i is the fitness value of i^{th} solution and p_i is the selection probability of i^{th} solution.

iv) Onlooker bee phase

The employed bees share the information about food sources with the onlooker bees for further processing. Each onlooker bee selects a food source to exploit according to the probability associated with it (i.e. more fitness, higher the probability). The chosen food sources are exploited for better solutions using equation (6) and their fitness values are calculated using equation (7). A greedy selection is again applied on the original as well as new food sources, similar to employed bee phase.

v) Scout bee phase

If a food source does not produce better solutions even up to a predefined limit, the food source is abandoned and the corresponding bee becomes a scout bee. A new food source is randomly generated in the search space using equation (5).

The employed, onlooker, scout bee phases and probabilistic selection phase will execute until termination criteria is satisfied. The best food source solution is obtained as output.

5. IMPROVED ABC ALGORITHM

The following modifications are proposed in the original ABC algorithm so as to generate better solutions:

1. Replacing the roulette wheel selection mechanism by variable tournament selection and replacing the worst solution by a random better solution, to enhance the convergence speed and quality of solutions in onlooker bee phase.
2. Use of modified Hooke and Jeeves search method to generate better solutions in scout bee phase.

5.1. Onlooker Bees Phase

Two modifications have been proposed in this phase to improve the quality of solutions. First step is to replace the roulette wheel selection mechanism by varying tournament selection mechanism. The size of tournament is selected on the basis of population size and cycle number. In second stage, the worst solution is replaced by better solution generated randomly. The tournament selection scheme works by holding a tournament of TN individuals chosen from the population, where TN is taken as tournament size (Blickle and Thiele, 1995; Miller and Goldberg, 1995). A tournament size $TN = 2$ is chosen in early stages for better exploration and a variable tournament size based on the current cycle number is chosen in later stages for better exploitation.

If $SN \geq 20$, the tournament size is taken as:

$$TN = SN * \frac{i}{10}, \; if \; MCN * \frac{i-1}{10} \leq cycle \leq MCN * \frac{i}{10} \; and \; i = 1,...,10 \tag{9}$$

If $10 < SN < 20$, then tournament size is taken as:

$$TN = \begin{cases} 2 & , if \;\; cycle \leq \frac{MCN}{5} \\ TN + \frac{SN - \text{mod}(SN, 5)}{5} & , if \;\; \frac{MCN}{5} < cycle \leq \frac{MCN}{5} * 4 \; and \; TN < SN \\ SN & , if \;\; \frac{MCN}{5} * 4 < cycle \leq MCN \end{cases} \tag{10}$$

If $SN \leq 10$, then tournament size is taken as:

$$TN = \begin{cases} 2 & , if \; cycle \leq \frac{MCN}{5} \\ TN + 1 & , if \; \frac{MCN}{5} < cycle \leq \frac{MCN}{5} * 4 \; and TN < SN \\ SN & , if \; \frac{MCN}{5} * 4 < cycle \leq MCN \end{cases} \tag{11}$$

where SN represents the number of food sources, TN the tournament size and MCN the maximum cycle number.

For small population the tournament size is incremented by 1, however with the growth in population the tournament size becomes dependent on current cycle. The high fitness food sources within this tournament size only are chosen by the onlooker bees thus speeding up the algorithm. Moreover, the replacement of worst fitness solution by a randomly generated solution provides the scope for better quality of solutions.

5.2 Scout bee phase

If the position of a food source cannot be upgraded further using limit, then it is intended to be abandoned and the corresponding bee becomes a scout bee. A new position for the abandoned food source is generated randomly in the search space. In order to generate a better position instead of random position, the modified Hooke and Jeeves method is used (Kumar and Sahoo, 2015). The new position may be generated using equation (12) as follows:

$$x_{new} = x_{best} + rand(0,1)\left(x_{best} - x_{curr}\right) \tag{12}$$

where x_{curr} indicates the current position and x_{best} is the best position achieved by a candidate solution.

6. PROPOSED IABCFCM ALGORITHM

We propose a hybrid algorithm, called IABCFCM, based on improved ABC and FCM algorithms for clustering problems. The hybrid algorithm incorporates the merits of IABC as well as FCM algorithms. The proposed algorithm is able to prevent the local optima trap of FCM algorithm and also improve the convergence speed of ABC algorithm. In short, the result of FCM algorithm is taken as a food source, the other food sources are initialized randomly within the given data set. We use equation (3) to evaluate the objective value of swarm for solving the clustering problem. The solutions with low objective value are considered as high fitness solutions. The brief steps of IABCFCM for clustering problem are given as follows and corresponding flowchart is mentioned in Figure 1.

1. (Initialization phase)

Initialize the parameters including number of food sources SN, limit, maximum cycle number MCN, and the current cycle number CN =0;

Initialize one food source as output of FCM, other food sources randomly within the given data set;

Evaluate the fitness of food sources using equation (3);

Send the employed bees to the current food source;

2. While (CN **<=** MCN ***) do***

3. (Employed bee phase)

  for (each employed bee)

    Find a new food source in the neighborhood of old food source using equation (6);

    Evaluate the fitness value of new food source using equation (3);

    Apply greedy selection on the original food source and the new one;

  end for

4. (Probabilistic selection phase)

  Calculate the probability values p_i for each food source using equation (8);

5. (Onlooker bee phase)

  t =1;

  while (current onlooker bee t <= SN)

    Calculate the tournament size based on population using equation (9) or (10) or (11);

    Out of the chosen tournament, find the food source having maximum probability value;

    Generate new solution for the selected food source using equation (6);

Evaluate the fitness value of new food source using equation (3);

Apply greedy selection on the original food source and the new one;

$t = t + 1$;

end while

Replace the worst fitness food source with a randomly produced food source, generate new solution, calculate the fitness value, apply greedy selection on the original food source and the new one;

6. ***(Scout bee phase)***

If (any employed bee becomes scout bee)

Send a scout bee to the solution of food source produced using equation (12);

end if

7. ***Memorize the best solution obtained so far***

$CN = CN + 1$;

8. ***end while***
9. ***Output the final cluster centers***

7. EXPERIMENTAL RESULTS AND ANALYSIS

Control parameters: The algorithms are implemented using MATLAB R2012a on an Intel (R) Core (TM) i3 CPU 3.06 GHZ with 4 GB RAM computer. Each food source (SN) is taken as a vector of real numbers of dimension k * d, where k is the number of clusters and d is the dimension of data set. The limit, used to control occurrence of scout bee, is calculated as: limit= SN * k * d. We choose m =2, ε = 0.00001, SN=n/k and maximum cycle number (MCN) as 100 iterations.

Data sets: Six data sets are employed to test our proposed algorithm. The six data sets taken from UCI Machine Repository are iris, glass, lung cancer, soyabean (small), wine and vowel data sets. The details of clusters, features and data objects in each data set are given as:

1. Iris data set (n=150, d=4, k=3): which consists of three different species of Iris flowers: Iris Setosa, Iris Versicolour and Iris Verginica. For each species, 50 samples with four features (sepal length, sepal width, petal length, and petal width) were collected.

2. Glass data set (n=214, d=9, k=6): This data set consists of six different types of glass: building windows float processed (70 objects), building windows non-float processed (76 objects), vehicle windows float processed (17 objects), containers (13 objects), tableware (9 objects), and headlamps (29 objects). Each data type has nine features, which are refractive index, sodium, magnesium, aluminium, silicon, potassium, calcium, barium, and iron.

3. Lung cancer data set (n=32, d=56, k=3): This data set consists of 32 samples of 56 feature parameters extracted from the clinical data and X-ray data. It describes 3 types of pathological lung cancers with 9, 13 and 10 samples.

4. Soyabean (small) data set (n=47, d=35, k=4): which consists of 47 instances, each being described by 35 attributes. Each instance is labelled as one of the four diseases: Diaporthe Stem Canker, Charcoal Rot, Rhizoctonia Root Rot, Phytophthora Rot. Except for Phytpphthora Rot which has 17 instances, all other diseases have 10 instances each.

5. Wine data set (n=178, d=13, k=3) These data, consisting of 178 objects characterized by 13 features namely alcohol, malic acid, ash, alkalinity of ash, magnesium, total phenols, flavanoids, nonflavanoid phenols, proanthocyanins, color intensity, hue, OD280/OD315 of diluted wines, and proline, are the results of a chemical analysis of wines brewed in the same region in Italy but derived

from three different cultivators. The three categories of data are: class 1 (59 instances), class 2 (71 instances), and class 3 (48 instances).

6. Vowel data set (n=871, d=3, k=6) This data set consists of 871 Indian Telugu vowels sounds, having six overlapping vowel classes namely d (72 instances), a (89 instances), i (172 instances), u (151 instances), e (207 instances) and o (180 instances). Each class has three input features corresponding to the first, second, and third vowel frequencies.

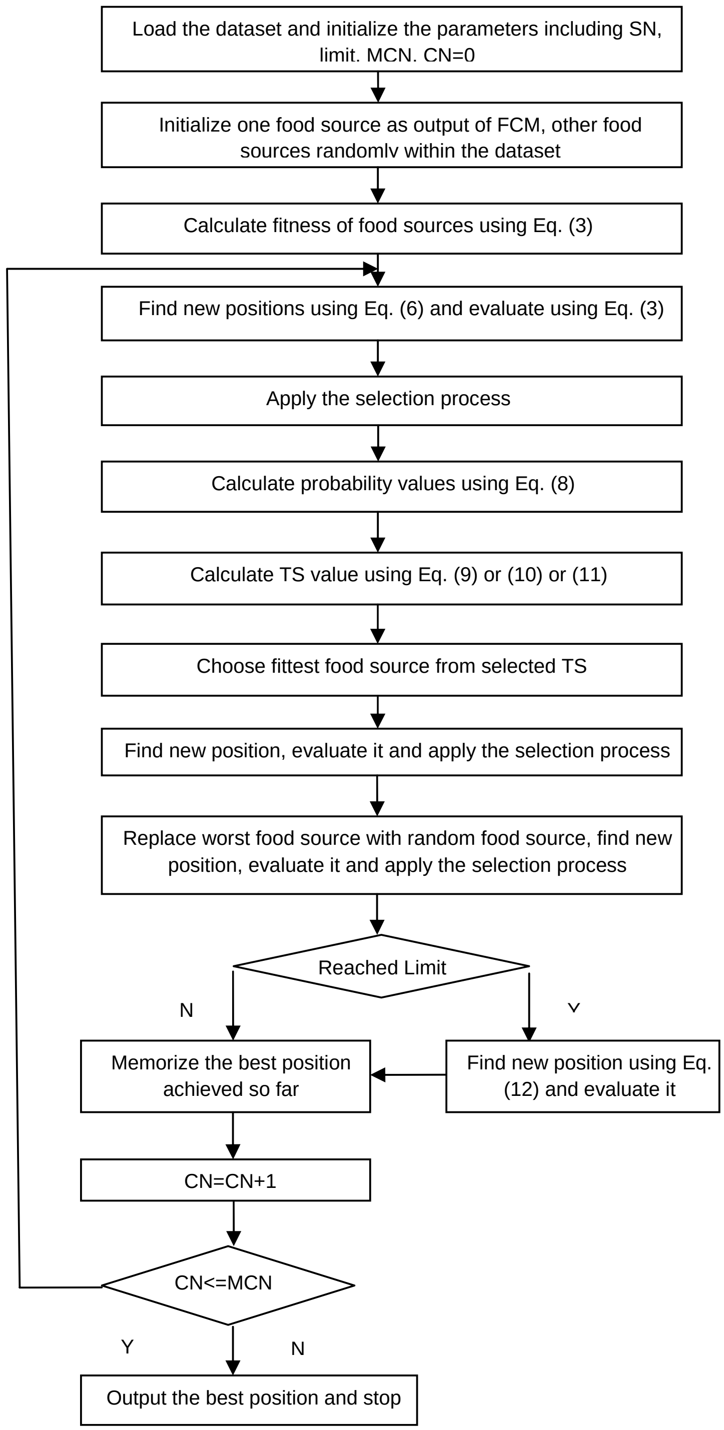

Figure 1. Flowchart of proposed IABCFCM

Evaluation measures: We evaluate and compare the performances of FCM, ABC and IABCFCM algorithms using three criteria:

- The objective function value (OFV) as defined in equation (3). Clearly, the smaller the value of OFV is, the higher the quality of clustering.
- The F-measure using the ideas of precision and recall from information retrieval (Dalli, 2003; Handl et al., 2003). Each class i (as given by the class labels of the used benchmark data set) is regarded as the set of n_i items desired for a query; each cluster j (generated by the algorithm) is regarded as the set of n_j items retrieved for a query; n_{ij} gives the number of elements of class i within cluster j. For each class i and cluster j, precision and recall are then defined as $p(i, j)$ and $r(i, j)$ using equation (13) and the corresponding value under the F-measure is as equation (14), where we choose b =1 to obtain equal weighting for precision and recall. The overall F-measure for the data set of size N is given by equation (15). Obviously, the bigger F-measure suggests a good quality of clustering.

$$p(i, j) = \frac{n_{ij}}{n_j}, \quad r(i, j) = \frac{n_{ij}}{n_i} \tag{13}$$

$$F(i, j) = \frac{(b^2 + 1).p(i, j).r(i, j)}{b^2.p(i, j) + r(i, j)} \tag{14}$$

$$F = \sum_{i=1}^{k} \frac{n_i}{N}.\max_j \{F(i, j)\} \tag{15}$$

- The error rate (ER): It is the number of misplaced points divided by the total number of points, as shown in equation (16), where N denotes the total number of points, and A_j and B_j denote the data sets of which the j^{th} point is a member before and after clustering, respectively. A low value of ER indicates a better clustering result.

$$ER = \left(\sum_{j=1}^{N} \left(if \left(A_j = B_j \right) then\ 0\ else\ 1 \right) \div N \right) \times 100 \tag{16}$$

All three algorithms were also tested and evaluated using the following measures:

- Rand Index (RI): Rand Index is a measure of degree of similarity in terms of correctly classified pairs of elements, between the known partition P and the solution C produced by a clustering algorithm. In a data set X having N data objects with partition P and solution C, following four different cases may arise:

i) pairs of data objects belonging to same class in C and same cluster in P (say a).
ii) pairs of data objects belonging to same class in C but different clusters in P (say b).
iii) pairs of data objects belonging to different classes in C but same clusters in P (say c).
iv) pairs of data objects belonging to different classes in C and different clusters in P (say d).

The value of Rand Index lies between 0 and 1 and is defined using equation (17). For a good clustering partition, RI is to be maximized.

$$RI = \frac{a + d}{a + b + c + d} \tag{17}$$

- Adjusted Rand Index (ARI): The adjusted Rand Index is another measure of agreement to compare clustering results. It assumes that the model of randomness takes the form of the generalized hypergeometric distribution. It is taken as the (normalized) difference of the RI and its expected value under the null hypothesis (Hubert and Arabie, 1985).

$$ARI = \frac{\binom{N}{2}(a+d)-(a+b)(a+c)+(c+d)(b+d)}{\binom{N}{2}^2-(a+b)(a+c)+(c+d)(b+d)} \tag{18}$$

The adjusted Rand Index can have a zero, one or negative value. Good clustering results correspond to high value of ARI .

- Hubert Index (HI): It is calculated as difference between agreement and disagreement between the known and generated partitions (Hubert, 1977). It can be stated as:

$$HI = \frac{a+d}{a+b+c+d} - \frac{b+c}{a+b+c+d} \tag{19}$$

Obviously, a high HI value indicates the occurrence of good clustering partitions.

- Class Entropy (CE): The entropy for a class i is defined as:

$$E_i = \sum_j p(j\,|\,i)\log_2 p(j\,|\,i) \tag{20}$$

where $p(j\,|\,i)$ indicates the probability that a data point is assigned to cluster j given that it belongs to class i in such a way that $\sum_j p(j\,|\,i)=1$. The overall class entropy is calculated as a sum of the entropies weighted by the class probabilities and is described as follows (Bakus et al., 2002):

$$CE = \sum_i \frac{N_i}{N} E_i \tag{21}$$

where N_i is the number of data points in class i, and N is the total number of data points. A low value of CE suggests the presence of good clustering results.

Results and analysis: The experimental results are averages of 20 runs of simulation. Table 1 describes the data sets used in experiments. Table 2 to 7 present the summary of results as well as centers obtained by various algorithms on different data sets.

Table 1. Description of data sets

Name of data set	Classes	Features	Size of data set
Iris	3	4	150 (50,50,50)
Glass	6	9	214 (70,17,76,13,9,29)
Lung cancer	3	56	32 (9,13,10)
Soyabean(small)	4	35	47 (10,10,10,17)
Wine	3	13	178 (59,71,48)
Vowel	6	3	871 (72,89,172,151,207,180)

Table 2. Results obtained by various algorithms on different data sets. Bold face indicates the best and italic face the second best result

Data sets	Best OFV	Average OFV	Worst OFV	Standard deviation	Error Rate (ER)	F-measure
Iris (k=3)						
FCM	*67.5915*	*67.8142*	*72.5743*	*0.5093*	12.1212	0.8917
ABC	67.6281	69.1361	79.6990	2.6379	**10.0157**	**0.9060**
IABCFCM	**67.4980**	**67.5162**	**67.5824**	**0.0170**	*10.8457*	*0.8991*
Glass (k=6)						
FCM	*184.2600*	*185.3420*	240.0190	6.0053	47.2798	0.7192
ABC	200.6300	209.3510	*219.7400*	*5.3516*	*44.4788*	*0.7599*
IABCFCM	**183.2460**	**183.4880**	**192.5720**	**1.2257**	**41.5623**	**0.7672**
Lung cancer (k=3)						
FCM	*225.8820*	*240.2140*	*242.5670*	*3.8637*	*42.7746*	**0.5922**
ABC	335.1440	351.8070	372.4240	10.5360	46.8571	0.5650
IABCFCM	**213.8850**	**213.9590**	**214.8850**	**0.1155**	**41.4508**	*0.5683*
Soyabean (k=4)						
FCM	**151.5720**	*158.4540*	*158.8790*	*1.3853*	*23.8938*	0.7854
ABC	207.9480	217.6660	232.5510	8.4737	25.9965	*0.8133*
IABCFCM	*153.1020*	**153.1980**	**156.4990**	**0.5497**	**16.6378**	**0.9129**
Wine (k=3)						
FCM	*2.0336e6*	*2.0366e6*	*2.2064e6*	*18450.00*	**32.2607**	0.7099
ABC	2.0396e6	2.1499e6	2.3461e6	92425.30	*33.1954*	*0.7196*
IABCFCM	**2.0235e6**	**2.0235e6**	**2.0247e6**	**118.9290**	34.3574	**0.7325**
Vowel (k=6)						
FCM	*2.0962e7*	2.2116e7	2.8782e7	1.9229e6	**44.1665**	**0.6386**
ABC	2.1234e7	*2.1452e7*	*2.1965e7*	*201354*	48.2342	0.5626
IABCFCM	**2.0934e7**	**2.0936e7**	**2.0950e7**	**4491.47**	*44.5988*	*0.5754*

Table 3. Centers obtained for the best OFV on iris data set

Center 1	5.0054	3.4210	1.4725	0.2506
Center 2	5.8657	2.7794	4.3066	1.3762
Center 3	6.7849	3.0382	5.6744	2.0537

Table 4. Centers obtained for the best OFV on glass data set

Center1	1.5165	14.5371	0.0464	2.2535	73.2195	0.1172	8.6873	1.0658	0.0193
Center2	1.5281	11.8267	0.0125	1.0928	71.9409	0.2158	14.4829	0.2077	0.1154
Center3	1.5218	13.8208	3.5568	0.8891	71.7804	0.1667	9.6349	0.0272	0.0589
Center4	1.5172	12.9372	3.3904	1.3387	73.0171	0.5844	8.4995	0.0216	0.0689
Center5	1.5174	13.3400	3.5665	1.4394	72.5757	0.5923	8.2426	0.0245	0.0537
Center6	1.5204	13.5120	0.2943	1.4370	72.9630	0.3053	11.2660	0.0164	0.0424

Table 5. Centers obtained for the best OFV on wine data set

Center1	12.5	2.5	2.3	20.8	92.6	2.1	1.7	0.4	1.4	4.2	0.9	2.5	464.2
Center2	13.0	2.5	2.4	19.6	105.1	2.2	1.7	0.4	1.6	5.6	0.9	2.5	761.8
Center3	13.8	1.9	2.5	16.9	105.2	2.9	3.0	0.3	1.9	5.9	1.1	3.1	1232.5

Table 6. Centers obtained for the best OFV on vowel data set

Center 1	653.9	1297.9	2281.5
Center 2	417.4	1021.8	2332.9
Center 3	360.7	2305.5	2985.2
Center 4	409.5	2101.1	2651.2
Center 5	516.9	1757.3	2512.3
Center 6	442.4	996.0	2677.1

Table 7. Results obtained by various algorithms for different data sets. Bold face indicates the best and italic face the second best result

Data set	Rand Index (RI)	Adjusted Rand Index (ARI)	Hubert Index (HI)	Class Entropy (CE)
Iris (k=3)				
FCM	0.8797	0.7302	0.7594	**0.6739**
ABC	**0.8922**	**0.7570**	**0.7845**	*0.6763*
IABCFCM	*0.8859*	*0.7429*	*0.7718*	0.6979
Glass (k=6)				
FCM	*0.7046*	*0.2048*	*0.4092*	*2.7848*
ABC	0.7037	0.1992	0.4074	2.8925
IABCFCM	**0.7114**	**0.2157**	**0.4228**	**2.7167**
Lung cancer (k=3)				
FCM	**0.6411**	*0.1734*	**0.2822**	*2.1722*
ABC	0.6250	0.1391	0.2500	2.1907
IABCFCM	*0.6371*	**0.1911**	*0.2742*	**1.9881**
Soyabean (k=4)				
FCM	0.8316	0.5451	0.6632	0.9507
ABC	*0.8427*	*0.5783*	*0.6854*	*0.8420*
IABCFCM	**0.8834**	**0.6874**	**0.7668**	**0.4772**
Wine (k=3)				
FCM	**0.7135**	**0.3602**	**0.4271**	*1.5919*
ABC	*0.7105*	*0.3539*	*0.4210*	1.6167
IABCFCM	0.7056	0.3447	0.4111	**1.5284**
Vowel (k=6)				
FCM	**0.8015**	**0.3185**	**0.6031**	**2.9189**
ABC	0.7124	0.2872	0.4248	3.0165
IABCFCM	*0.8005*	*0.3150*	*0.6012*	*2.9499*

Table 2 presents the OFV, error rate and F-measure values obtained from the three clustering algorithms for the above data sets. The OFV values reported are best, average and worst values with standard deviations to indicate the range of values that the algorithms span from the 20 simulations. The results show that IABCFCM is very precise i.e. it produces the optimum value and very small standard deviation in comparison to other algorithms. For iris, the proposed hybrid algorithm is able to find the global optimum value 67.4980. The worst OFV 67.5824 generated by proposed algorithm is even better than best OFV 67.5915 and 67.6281 of FCM and ABC algorithms respectively. The proposed algorithm also provides significant lower standard deviation 0.0170 as compared to other algorithms. Moreover, the hybrid algorithm produces comparable error rate and F-measure values, which is an indication of better quality of solutions. In glass data set, the average OFV 183.4880 of hybrid algorithm is better than best OFV with much lower standard deviation 1.2257 as compared to other two algorithms. Also, the modified algorithm exhibits better error rate 41.5623 as well as F-measure value 0.7672 in comparison to other two methods. In case of lung cancer, there is significant difference between worst OFV 214.8850 of hybrid algorithm and best OFV of other algorithms. Also, the standard deviation 0.1155 is much lower as that of other algorithms. The error rate 41.4508 for the modified algorithm is also better than FCM and ABC algorithms. However, FCM algorithm provides better F-measure 0.5922 in comparison to other algorithms. In case of soyabean data set also, the modified algorithm outperforms other two algorithms on all evaluation measures except best OFV 153.1020, which is little different from best OFV 151.5720 produced by FCM algorithm. However, this OFV value is considerably better than best OFV 207.9480 of ABC algorithm. The ABC algorithm provides a much higher standard deviation 8.4737, meaning that it is less likely to reach the optimal values than IABCFCM if they execute just once. The hybrid algorithm provides much smaller error rate 16.6378 as compared to other algorithms. In addition, the modified algorithm produces significant F-measure value 0.9129. For wine data set, the modified algorithm outperforms the other methods in terms of best, average and worst OFV as 2.0235e+06, 2.0235e+06 and 2.0247e+06 respectively with much less standard deviation 118.9290 in comparison to other algorithms. However, FCM exhibits better error rate 32.2607 and modified algorithm provides F-measure 0.7325, better than 0.7196 of ABC algorithm. The results for vowel data set also prove the superiority of modified algorithm in terms of best, average and worst OFV as 2.0934e+07, 2.0936e+06 and 2.0950e+07, with much smaller standard deviation 4491.47. The FCM algorithm provides slightly better error rate 44.1665 in comparison to 44.5988 of hybrid algorithm. Also, the FCM produces better F-measure 0.6386. It follows that IABCFCM is efficient in finding the global optimum solution with much lower standard deviation.

Tables 3, 4, 5 and 6 provide the best centroids found by the proposed algorithm in iris, glass, wine and vowel data sets respectively. Table 7 provides the results obtained by various algorithms on various parameters. For iris data set, the hybrid algorithm provides the values of RI 0.8859, ARI 0.7429 and HI 0.7718, that are close to the best values by ABC i.e. 0.8922, 0.7570 and 0.7845 respectively. In case of glass data set also, the modified algorithm provides good clustering results as given by RI 0.7114, ARI 0.2157, HI 0.4228 and CE 2.7167 values. The results also show that the increase in number of dimensions does not affect the behavior of proposed algorithm in terms of RI, ARI, HI and CE indices as reported in case of lung cancer and soyabean data sets. The algorithm is able to provide best values of ARI and CE for lung cancer data set, while the values of RI and HI are better than those by ABC algorithm. The values of all four measures are sufficiently better than other algorithms in case of soyabean data set. In case of wine data set, the modified algorithm does not provide the desired values of RI, ARI and HI, but that of CE is found best. With the increase in number of samples, the proposed algorithm does not produce best partitioning results as given in vowel data set, rather the results are closer to those of FCM algorithm.

From Table 2 and 7, it may be concluded that the proposed hybrid algorithm exhibits best performance on various evaluation measures for glass and soyabean(small) data sets, whereas it provides nearly best results for iris, lung cancer, wine and vowel data sets. The proposed method generates good clustering partitions on low-dimensional as well as high-dimensional data sets.

By increasing iteration from 1 to 100, Figures 2 to 7 demonstrate the change in objective function value (OFV) by using different methods. From Figure 2, it is clear that worst objective function value of modified algorithm is even better than best value of FCM and ABC. The FCM algorithm exhibits a fast but premature convergence to the local optimum using less than 10 iterations. The ABC shows the slower convergence using 61 iterations, but converges near to the global optimum with the increase in iterations. The improved method provides much better and stable objective function values with the increase of iterations.

Figure 3 shows the worst and unstable performance of ABC method, the modified algorithm has a fast convergence with the increase in iterations, using less than 10 iterations. The overall level of the hybrid algorithm is significantly lower than FCM and stable enough. The performance of FCM is significantly better than ABC, but converges slowly to the local optimum using 22 iterations.

Figure 4 illustrates the performance of ABC is worse, the hybrid method is much better than FCM in terms of convergence speed and stable objective function value. The FCM exhibits unpredicted behavior, but fast convergence before trapping to the local optimum. The modified algorithm converges to the global optimum using less than 10 iterations and generates best objective function values on a data set having dimensions more than the number of samples.

Figure 5 displays the performance of ABC is the worst one, the FCM is much better than ABC, but worse than the improved algorithm obviously. The objective function value provided by ABC is much higher and shows slow convergence, using more than 70 iterations. Here also, the FCM displays unpredicted curve before converging to the local optimum. FCM converges to a much better optimal solution but does not reach the global optimum. The hybrid algorithm provides excellent results with much faster convergence, using less than 5 iterations and stable convergence to the global optimum.

Figure 6 depicts the convergence trends of the algorithms for wine data set. The ABC performs slowly and provides function value slightly higher than FCM. The FCM shows fast convergence using 11 iterations, in addition to stable and better performance than ABC. However, the modified algorithm exhibits best convergence rate and significant objective function value through all the iterations.

Figure 7 shows the objective function value in FCM and modified method is at the same level, and the FCM has slow convergence with the increase of iterations, using 40 iterations. The performance of ABC is the worst one and is never able to reach the global optimum. The improved algorithm performs well with fast convergence using less than 10 iterations and produces stable as well as minimum objective function value leading to global optimum.

In summary, for iris data set FCM and improved method display consistent performance but improved method is able to reach the global optimum. The ABC performs slow and converges close to the global optimum in later iterations. In glass data set, ABC shows the worst performance and hybrid method is the best one. The FCM is again trapped to the local optima. The experiment using lung cancer data set displays the significant results of hybrid method in comparison to other algorithms. The results in soyabean(small) data set prove that FCM is not able to generate sufficiently good results on high dimensional data sets. For wine data set, ABC is initially slow, but performs well in later iterations. FCM is again better than ABC and modified method provides the best objective function value. In case of vowel data set, ABC is the worst one and FCM performs very close to the improved method with the increase of iterations.

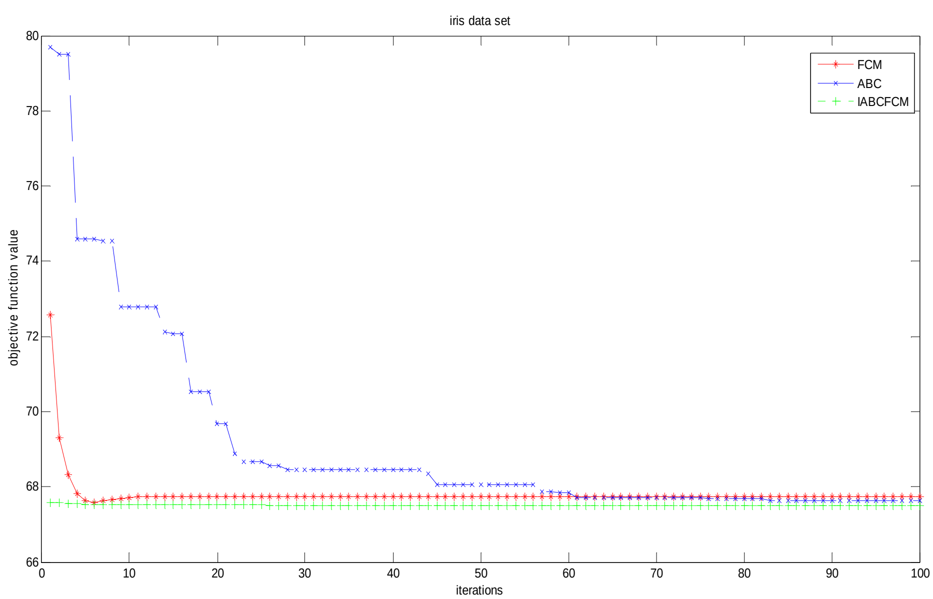

Figure 2. Comparison of OFV on iris data set

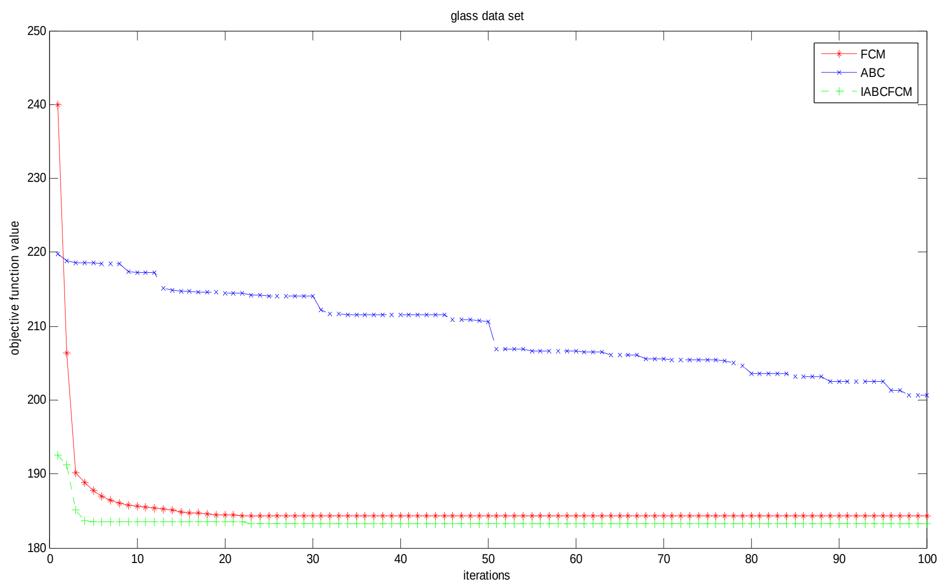

Figure 3. Comparison of OFV on glass data set

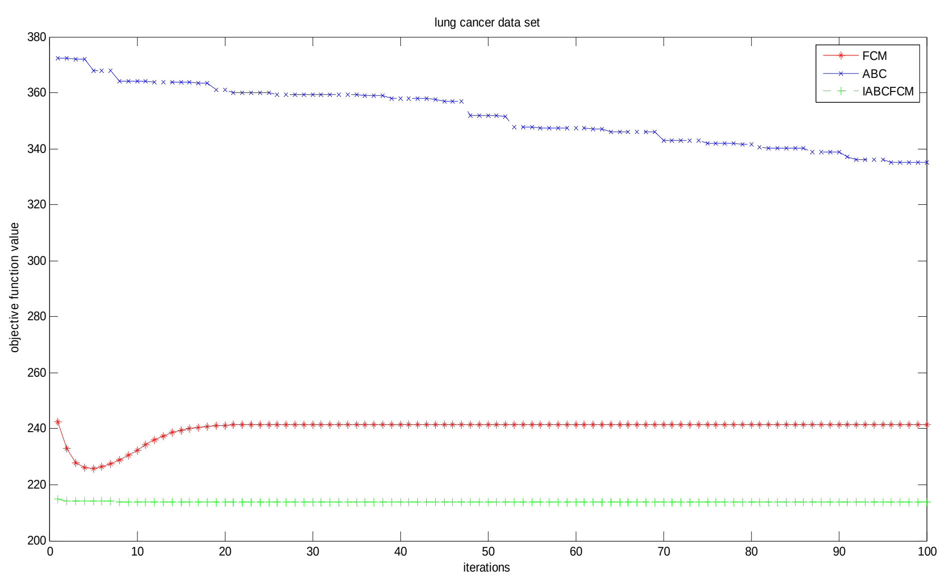

Figure 4. Comparison of OFV on lung cancer data set

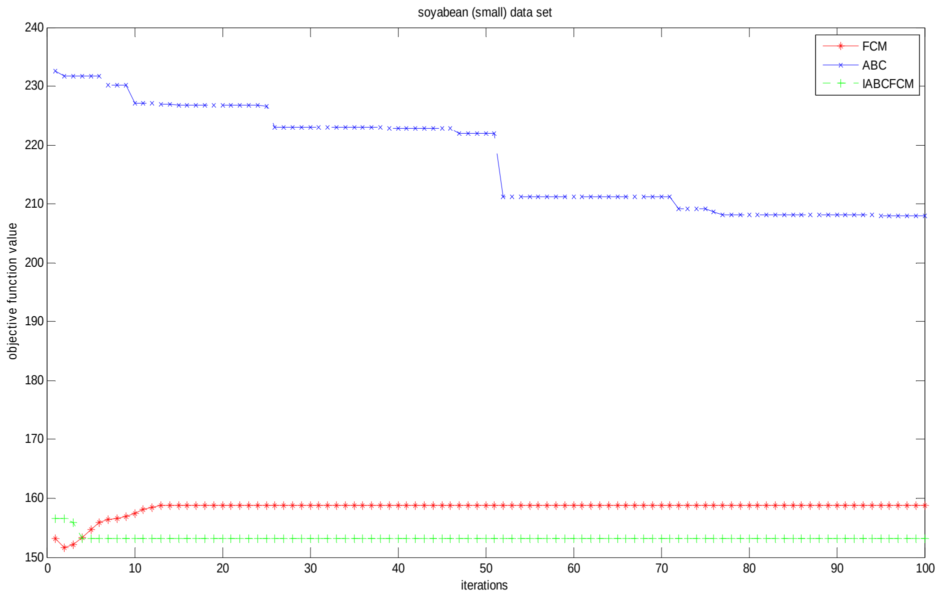

Figure 5. Comparison of OFV on soyabean(small) data set

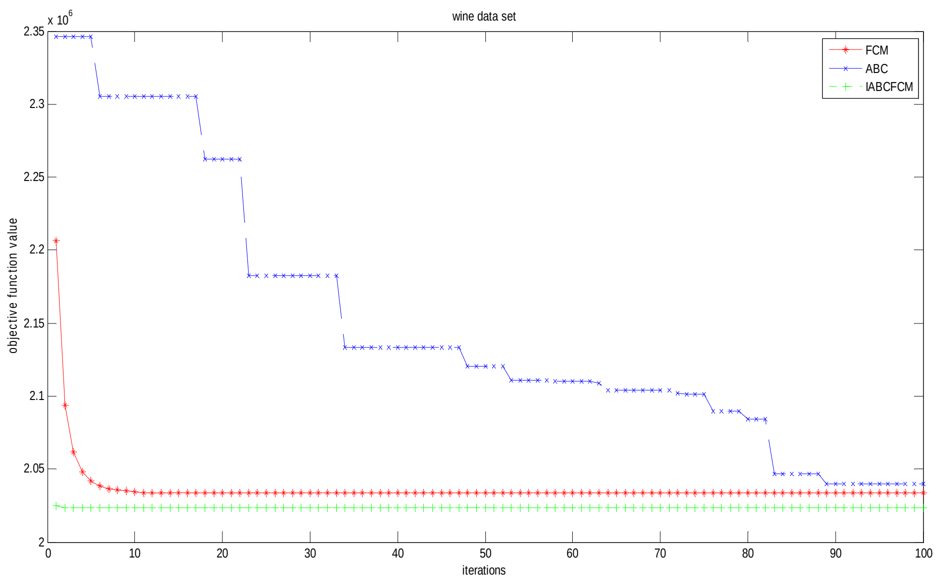

Figure 6. Comparison of OFV on wine data set

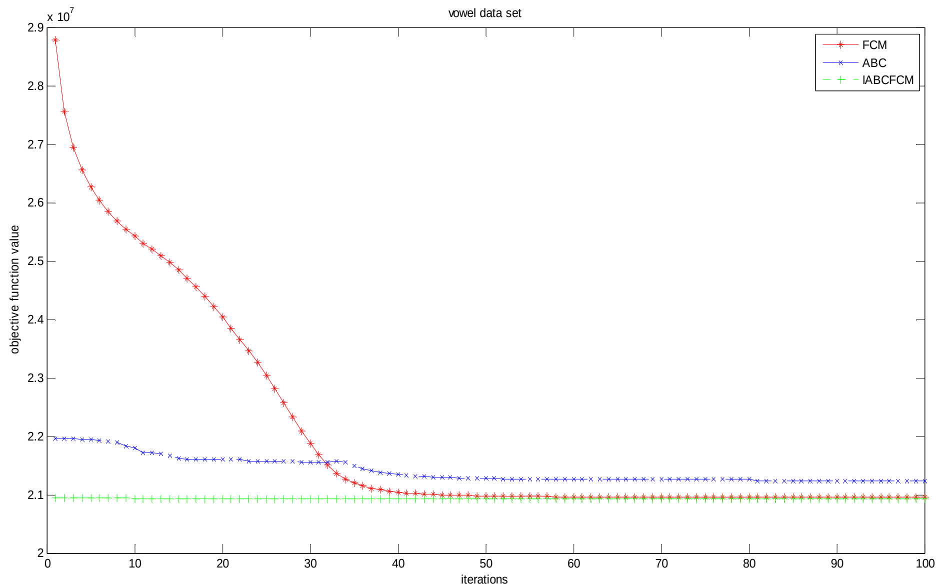

Figure 7. Comparison of OFV on vowel data set

8. CONCLUSION

This paper investigates a hybrid clustering algorithm (IABCFCM) based on improved ABC and FCM algorithms. The hybrid algorithm incorporates the merits of FCM and ABC algorithms to achieve the desired results. The experiments are performed on six standard data sets from UCI Machine Learning Repository. The hybrid algorithm searches robustly the data cluster centers in an N-dimensional search space, using the minimum objective function value as a metric. Using the same metric, FCM algorithm is easily trapped in local optima, while the ABC algorithm is not able to provide satisfactory results in mostly cases. The experimental results show that the proposed algorithm is able to escape local optima and find global optimum as compared to other two algorithms. The proposed hybrid algorithm also outperforms the other methods in terms of the various evaluation measures and achieves best ranking among three methods.

The results prove that the proposed algorithm can better handle data sets irrespective of the dimension less than or more than the number of samples. Moreover, the hybrid algorithm converges to global optimum with a smaller standard deviation and better clustering partitions and leads naturally to the conclusion that IABCFCM is a viable and robust technique for data clustering. The proposed method needs improvement to perform automatic clustering without any prior knowledge of number of clusters.

9. REFERENCES

Akay, B., Karaboga, D., 2012, A modified artificial bee colony algorithm for real-parameter optimization, Inf. Sci., 192, 120-142.

Al-Sultan, K.S., 1995, A tabu search approach to the clustering problem, Pattern Recognition, 28(9), pp. 1443-1451.

Bakus, J., Hussin, M.F., Kamel, M., 2002, A SOM-based document clustering using phrases, in ICONIP'02: Proceedings of the 9th international conference on neural information processing, Vol. 5, pp. 2212-2216.

Barnharnsakun, A., Achalakul, T., Sirinaovakul, B., 2011, The best-so-far selection in artificial bee colony algorithm, Appl. Soft Comput., 11, 2888-2901.

Beloufa, F., Chikh, M.A., 2013, Design of fuzzy classifier for diabetes disease using modified artificial bee colony algorithm, Computer Methods and Programs in Biomedicine, Vol. 112, No. 1, pp. 92-103.

Bezdek, J., 1981, Pattern recognition with fuzzy objective function algorithms, Advanced Applications in Pattern Recognition, Plenum Press, New York.

Blickle, T., Thiele, L., 1995, A mathematical analysis of tournament selection, in Eshelman, L.(Ed.), Proceedings of Sixth International Conf. Genetic Algorithms (ICGA95), Morgan Kaufmann, San Francisco, CA, pp. 9-16.

Chen, C.Y., Ye, F., 2004, Particle swarm optimization algorithm and its application to clustering analysis, in IEEE International Conference on Networking, Sensing and Control, Taiwan, pp. 789-794.

Chen, S., Xu, Z., Tang, Y., 2014, A hybrid clustering algorithm based on fuzzy c-means and improved particle swarm optimization, Arab J Sci Eng, Vol. 39, No. 12, pp. 8875-8887.

Chowdhury, A., Bose, S., Dos, S., 2011, Automatic Clustering based on Invasive Weed Optimization Algorithm, in Proceedings of the 2nd International Conference, Swarm, Evolutionary and Memetic Computing, Andhra Pradesh, pp. 105-112.

Dalli, A., 2003, Adaptation of the F-measure to cluster-based Lexicon quality evaluation, In Proceedings of EACL 2003 Workshop, Budapest, pp. 51-56.

Fathian, M., Amiri, B., Maroosi, A., 2007, Application of Honey-Bee Mating Optimization Algorithm on Clustering, Applied Mathematics and Computation, Vol. 190, No. 2, pp. 1502-1513.

Gan, G., Ma, C., Wu, J., 2007, Data clustering: Theory, algorithms, and applications, ASA-SIAM Series on Statistics and Applied Probability, SIAM, Philadelphia, VA.

Gao, W.-F., Liu, S.-Y., 2012, A modified artificial bee colony algorithm, Comput. Oper. Res., 39, 687-697.

Gar, R., Sharma, R.K., Sharma, K., 2016, Ranking and selection of commercial off-the-shelf using fuzzy distance based approach, Decision Selection Letters, 5, 201-210.

Ghosn, S.B., Drouby, F., Harmanani, H.M., 2016, A parallel genetic algorithm for the open-shop scheduling problem using deterministic and random moves, International Journal of Artificial Intelligence, Vol. 14, No. 1, pp. 130-144.

Han, J., Kamber, M., 2006, Data Mining: Concepts and Techniques, 2nd edition, Morgan Kaufmann Publishers, California, USA.

Handl, J., Knowles, J. Dorigo., M., 2003, On the performance of ant-based clustering, Design and Application of Hybrid Intelligent Systems, Frontiers in Artificial Intelligence and Applications, Vol. 104, pp. 204-213.

Hatamlou, A., 2013, Black hole: A new heuristic optimization approach for data clustering, Information Sciences, Vol. 222, pp. 175-184.

Hubert, L., 1977, Nominal scale response agreement as a generalized correlation, Br. J. math. statist. Psychol., Vol. 30, pp. 98-103.

Hubert, L., Arabie, P., 1985, Comparing partitions, Journal of classification, Vol. 2, pp. 193-218.

Jain, A.K., Murty, M.N., Flynn, P.J., 1999, Data clustering: a review, ACM computing surveys, Vol. 31, No. 3, pp. 264-323.

Karaboga, D., 2005, An idea based on honey bee swarm for numerical optimization, Technical Report – TR06, Erciyes University.

Karaboga, D., Ozturk, C., 2010, Fuzzy clustering with artificial bee colony algorithm, Scientific Research and Essays, Vol. 5, No. 14, pp. 1899-1902.

Karaboga, D., Akay, B., 2011, A modified artificial bee colony (ABC) algorithm for constrained optimization problems, Appl. Soft Comput., 11, 3021-3031.

Kiran, M.S., Gunduz, M. 2012, A novel artificial bee colony-based algorithm for solving the numerical optimization problems, Int. J. Innov. Comput. Inf. Control, 9, 6107-6121.

Kiran, M.S., Findik, O., 2015, A directed artificial bee colony algorithm, Applied Soft Computing, Vol. 26, pp. 454-462.

Krishnamoorthi, M., Natarajan, A.M., 2013, Artificial Bee Colony Algorithm integrated with Fuzzy C-Mean Operator for Data Clustering, Journal of Computer Science, 9(4): 404-412.

Kumar, Y., Sahoo, G., 2014, A charged system search approach for data clustering, Prog. Artif. Intell., doi: 10.1007/s13748-014-0049-2.

Kumar, Y., Sahoo, G., 2015, A two-step artificial bee colony algorithm for clustering, Neural Comput & Applic, DOI 10.1007/s00521-015-2095-5.

Lam, H.K., Leung, H.H.F., Tam, P.K.S., 2000, Stable and Robust Fuzzy Control for Uncertain Nonlinear Systems, IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, Vol. 30, No. 6, pp. 825-840.

Lin, J.-S., Wu, S.-H., 2013, Fuzzy Artificial Bee Colony System with Cooling Schedule for the Segmentation of Medical Images by using of Spatial Information, Research Journal of Applied Sciences, Engineering and Technology 5(5): 1483-1490.

Luo, J., Wang, Q., Xiao, X., 2013, A modified artificial bee colony algorithm based on converge-onlookers approach for global optimization, 219, 10253-10262.

Malaki, M., Ahangar, F.S., Zakerolhosseini, A., 2012, A Novel Hybrid Fuzzy Clustering Algorithm Based on Artificial Bee Colony and C-means, in Congress on Electrical, Computer and Information Technology (I.T.) Engineering.

Martin, D., Toro, R.D., Haber, R., Dorronsoro, J., 2009, Optimal tuning of a networked linear controller using a multi-objective genetic algorithm and its application to one complex electromechanical process, International Journal of Innovative Computing, Information and Control, Vol. 5, No. 10(B), pp. 3405-3414.

Miller, B.L., Goldberg, D.E., 1995, Genetic algorithms, tournament selection, and the effects of noise, Complex Systems, Vol. 9, pp. 193-212.

Moallem, P., Mousavi, B.S., Naghibzadeh, S.Sh., 2015, Fuzzy inference system optimized by genetic algorithm for robust face and pose detection, International Journal of Artificial Intelligence, Vol. 13, No. 2, pp. 73-88.

Murthy, C.A., Chowdhury, N., 1996, In search of optimal clusters using genetic algorithms, Pattern Recognition Letters, 17(8), 825-832.

Palanisamy, C., Selvan, S., 2009, Wavelet based fuzzy clustering of higher dimensional data, International Journal of Artificial Intelligence, Vol. 2, No. S09, pp. 27-36.

Precup, R.-E., Preitl, S., 2004, Optimisation criteria in development of fuzzy controllers with dynamics, Engineering Applications of Artificial Intelligence, Vol. 17, pp. 661-674.

Precup, R.-E., Preitl, S., Korondi, P., 2007, Fuzzy controllers with maximum sensitivity for servosystsems, IEEE Transactions on Industrial Electronics, Vol. 54, No. 3, pp. 1298-1310.

Precup, R.-E., David, R.-C., Petriu, E.M., Preitl, S., Radac, M.-B., 2014, Novel adaptive charged system search algorithm for optimal tuning of fuzzy controllers, Expert Systems with Applications, Vol. 41, No. 4, pp. 1168-1175.

Santosa, B., Mirsa, K.N., 2009, Cat swarm optimization for clustering, in International Conference on Soft Computing and Pattern Recognition (SOCPAR'09), pp. 54-59.

Satapathy, S.C., Naik, A., 2011, Data clustering based on teaching-learning-based optimization, in Swarm, Evolutionary, and Memetic Computing, Springer Berlin Heidelberg, pp. 148-156.

Selim, S.Z., Alsultan, K., 1991, A simulated annealing algorithm for the clustering problem, Pattern Recognition, 24(10), pp. 1003-1008.

Shelokar, P.S., Jayaraman, V.K., Kulkarni, B.D., 2004, An ant colony approach for clustering, Analytica Chimica Acta, 509(2), 187-195.

Su, Z.G., Wang, P.H., Shen, J., Li, Y.G., Zhang, Y.F., Hu, E.J., 2012, Automatic fuzzy partitioning approach using variable string length artificial bee colony (VABC) algorithm, Applied Soft Computing, Vol. 12, No. 11, pp. 3421-3441.

Yazdani, D., Saman, B., Sepas-Moghaddam, A., Mohammad-Kazemi, F., Meybodi, M.R., 2013, A new algorithm based on improved artificial fish swarm algorithm for data clustering, International Journal of Artificial Intelligence, Vol. 11, No. A13, pp. 193-221.

Zhang, C., Ouyang, D., Ning, J., 2010, An artificial bee colony approach for clustering, Expert Systems with Applications, Vol. 37, No. 7, pp. 4761-4767.

Zhu, G., Kwong, S., 2010, Gbest-guided artificial bee colony algorithm for numerical function optimization, Appl. Math. Comput., 217, 3166-3173.