

Research Article

An Effective Hybrid of Bees Algorithm and Differential Evolution Algorithm in Data Clustering

**Mohammad Babrdel Bonab,¹ Siti Zaiton Mohd Hashim,¹
Nor Erne Nazira Bazin,¹ and Ahmed Khalaf Zager Alsaedi²**

¹*Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia*

²*College of Science, Misan University, Ministry of Higher Education of Iraq, Iraq*

Correspondence should be addressed to Mohammad Babrdel Bonab; bbmohammad2@live.utm.my

Received 8 October 2014; Accepted 16 January 2015

Academic Editor: Yi-Chung Hu

Copyright © 2015 Mohammad Babrdel Bonab et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Clustering is one of the most commonly used approaches in data mining and data analysis. One clustering technique in clustering that gains big attention in clustering related research is k -means clustering such that the observation is grouped into k cluster. However, some obstacles such as the adherence of results to the initial cluster centers or the risk of getting trapped into local optimality hinder the overall clustering performance. The purpose of this research is to minimize the dissimilarity of all points of a cluster from gravity center of the cluster with respect to capacity constraints in each cluster, such that each element is allocated to only one cluster. This paper proposes an effective combination algorithm to find optimal cluster center for the analysis of data in data mining and a new combination algorithm is proposed to untangle the clustering problem. This paper presents a new hybrid algorithm, which is, based on cluster center initialization algorithm (CCIA), bees algorithm (BA), and differential evolution (DE), known as CCIA-BADE-K, aiming at finding the best cluster center. The proposed algorithm performance is evaluated with standard data set. The evaluation results of the proposed algorithm and its comparison with other alternative algorithms in the literature confirm its superior performance and higher efficiency.

1. Introduction

Data clustering is one of the most important knowledge discovery techniques to extract structures from dataset and is widely used in data mining, machine learning, statistical data analysis, vector quantization, and pattern recognition. The aim of clustering is to partition data into k cluster, so that each cluster contains data, which has the most similarity and maximum dissimilarity with the other clusters. Clustering algorithms can be comprehensively classified into hierarchical, partitioning, model-based, grid-based, and concentration-based clustering algorithms [1–3].

Hierarchical clustering algorithm divides a dataset into a number of levels of nested partitioning. In the partitioning algorithms observations of one dataset decompose into a set of k clusters with most similarity among intra-group members and least similarity among inter group members [4].

Dissimilarities are evaluated based on attribute values. Generally, distance criterion is used for data analysis [5].

The k -means algorithm is one of the partitioning clustering algorithm and one of the most popular algorithms, used in many domains. The k -means algorithm implementation is easy and often practical. However, results of k -means algorithm considerably depend on initial state. In other words, its efficiency highly depends on the first initial center [6].

The main purpose of k -means clustering algorithm is to minimize the diversity of all objects in a cluster from their cluster centers. The initialization problem of k -means algorithm is considered by heuristic algorithms, but it still risks being trapped in local optimality. Therefore, for achieving a better cluster algorithm we should find a solution for overcoming the problem of trap into local optimum [7].

There are many studies to overcome this problem. For instance, Niknam and Amiri have proposed a hybrid approach based on combining partial swarm optimization and ant colony optimization with k -means algorithm for data clustering [8], and Nguyen and Cios have proposed a combination technique based on the hybrid of k -means, genetic algorithm, and maximization of logarithmic regression expectation [9]. Kao et al. have presented a combination algorithm according to the hybrid of partial swarm optimization, Nelder-Mead simplex search and genetic algorithm [10]. Krishna and Murty proposed an algorithm for cluster analysis called genetic k -means algorithm [11]. Žalik proposed an approach for clustering without preassigning cluster numbers [12]. Maulik and Bandyopadhyay have introduced genetic based algorithm to solve this problem and evaluate the performance on real data. They define spatial distance-based mutation according to mutation operator for clustering [13]. Laszlo and Mukherjee have proposed another genetic based approach, that for k -means clustering exchanges neighboring cluster centers [14]. Fathian et al. have presented a technique to overcome clustering problem according to honey-bees mating optimization (HBMO) [15–17]. Shelokar et al. have presented to solve clustering problem based on the ant colony optimization [18]. Niknam et al., have combined to dominate this problem based on the simulated annealing and ant colony optimization [19]. Ng and Sung have introduced a technique based on the taboo search to find cluster center [20, 21]. Niknam et al. have introduced a hybrid approach based on combining partial swarm optimization and ant simulated annealing to solve clustering problem [22, 23].

The bees algorithms can be classified in two main categories including foraging-based honeybee algorithms and marriage-based honeybee algorithm. Each of these categories have many algorithm such as artificial bee algorithm (ABC) [3, 24, 25], corporate artificial bee algorithm (CABC) [26], parallel artificial bee algorithm (PABC) [27], bee colony optimization (BCO) [28, 29], bee algorithm (BA) [30], bee foraging algorithm (BFA) [31], bee swarm optimization (BSO) for first categories [32]. Marriage in honey-bees optimization (MBO) [32], fast marriage honey-bees optimization (FMBO) [33], and finally modified fast marriage in honey-bees optimization (MFMBO) are in the second category of bee algorithm [34].

One of the foraging-based algorithms is the bees algorithm that is a new population based search algorithm, developed by Pham et al. in 2006 [30]. The algorithm mimics the food foraging behavior of swarms of honeybees (Figure 3). In its basic version, the algorithm performs a kind of neighborhood search combined with random search and can be used for optimization problems [30].

Differential evolution is an evolutionary algorithm (EA), which has been widely used in to optimization problems, mainly in continuous search spaces [35]. Differential evolution was introduced by Storn and Price in 1995 [36]. Global optimization is necessary in fields such as engineering, statistics, and finance, but many practical problems have objective functions that are nonlinear, noisy, noncontinuous, and multidimensional or have many local minima and constraints. Such problems are difficult if not impossible to solve

analytically. Differential evolution can be used to find approximate solutions to such problems. Differential evolution also includes genetic algorithms, evolutionary strategies, and evolutionary programming. Differential evolution encodes solutions as vectors and new solution, compared to its parent. If the candidate is better than its parents, it replaces the parent in the population. Differential evolution can be applied in numerical optimization [37, 38].

In this paper, a hybrid evolutionary technique is used in order to solve the k -means problem. The proposed algorithm helps clustering technique to escape from being trapped in local optimum. Our algorithm takes the benefits of both algorithms. Also, in this survey, some standard datasets are used for testing the proposed algorithm. To obtain the best cluster centers, in proposed algorithm, the advantages of BA (bees algorithm) and DE (differential evolution) are used with a data preprocessing technique called CCIA (cluster center initialization algorithm) for data analysis. Through experiments, the proposed CCIA-BADE-K algorithm has shown that this algorithm efficiently selects the exact cluster centers.

The main contribution of this paper is the introduction of a novel combination of evolutionary algorithm according to bees algorithm and differential evolution to overcome data analysis problem and hybrid with CCIA (cluster center initialization algorithm) preprocessing technique.

The rest of this paper is arranged as follows: in Section 2, the data clustering issue is introduced. In Sections 3 and 4, the classic principles of the DE and BA evolutionary algorithm are discussed. In Section 5, the suggested approach is introduced. In Section 6, experimental results of proposed algorithm are shown and compared with PSO-ANT, SA, ACO, GA, ACO-SA, TS, HBMO, PSO, and k -means on benchmark data and finally Section 7 presents the concluding remarks.

2. Data Clustering

Clustering is defined as grouping similar objects either physically or in abstract. The groups inside one cluster have the most similarity with each other and the maximum diversity with other groups' objects [39].

Definition 1. Suppose the set of $X = \{x_1, x_2, \dots, x_n\}$ containing n objects. The purpose of clustering is to group objects in k clusters as $C = \{c_1, c_2, \dots, c_k\}$ while each cluster satisfies the following conditions [40]:

- (1) $C_1 \cup C_2 \cup \dots \cup C_k = X$;
- (2) $C_i \neq \emptyset, i = 1, \dots, k$;
- (3) $C_i \cap C_j = \emptyset$.

According to the mentioned definition, the possible modes for clustering n objects in k clusters are obtained as follows:

$$NW(n, k) = \frac{1}{k!} \sum_{i=1}^k (-1)^i \binom{k}{i} (k-i)^n. \quad (1)$$

(1) **Input:** Data SET ($X = \{x_1, x_2, \dots, x_n\}$), Attribute Set ($A = \{A_1, A_2, \dots, A_q\}$), Cluster Number (K),
(2) **Output:** Clusters Seed-Set ($SC = \{sc_1, sc_2, \dots, sc_H\}$)
(3) **Begin**
(4) While ($\forall A_j \in A$)
 (4.1) Compute Standard Deviation (σ_j) and Mean (μ_j)
 (4.2) Compute Cluster Center ($e = 1, 2, \dots, k$)

$$X_e = Z_e * \sigma_j + \mu_j \quad Z_e = \frac{2 * e - 1}{2 * k}$$

 (4.3) Execute k -means on this attribute
 (4.4) Allocate cluster labels obtained from Step (4.3) to every data pattern
(5) Find unique patterns ($H \geq k$) and clustering each data with obtained patterns.
(6) Return SC
(7) End

ALGORITHM 1: Pseudocode of CCIA algorithm.

In most approaches, the cluster number, that is, k , is specified by an expert. Relation (1) implies that even with a given k , finding the optimum solution for clustering is not so simple. Moreover, the number of possible solutions for clustering with n objects in k clusters increases by the order of $k^n/k!$. So, obtaining the best mode for clustering n objects in k clusters is an intricate NP-complete problem which needs to be settled by optimization approaches [5].

2.1. The K-Means Algorithm. There have been many algorithms suggested for addressing the clustering problem and among them the k -means algorithm which is one of the most famous and most practical algorithms [41]. In this method, besides the input datasets, k samples are introduced into the algorithm as the initial centers of k clusters. These representing k 's are usually the first k data samples [39]. The way these k representatives are chosen influences the performance of K -means algorithm [42]. The four stages of this algorithm are shown as follows.

Stage I. Choose k data items randomly from $X = \{x_1, x_2, \dots, x_n\}$ as cluster centers of (m_1, m_2, \dots, m_k) .

Stage II. Based on relation (2), add every data item to a relevant cluster. For example, if the following relation (2) holds, the object x_i from the set of $X = \{x_1, x_2, \dots, x_n\}$ is added to the cluster c_j

$$\|x_i - m_j\| < \|x_i - m_p\| \quad 1 \leq p \leq k, \quad j \neq p. \quad (2)$$

Stage III. Now, based on the clustering of Stage II, the new cluster centers $(m_1^*, m_2^*, \dots, m_k^*)$ are calculated by using relation (3) as follows (n_i is the number of objects in the cluster i):

$$m_i^* = \frac{1}{n_i} \sum_{x_j \in C_i} x_j \quad 1 \leq i \leq k. \quad (3)$$

Stage IV. If the cluster centers are changed, repeat the algorithm from Stage II, otherwise do the clustering based on the resulted centers.

The performance of the k -means clustering algorithm relies on initial centers and this is a major challenge in this algorithm. Random selection of initial cluster centers makes this algorithm yield different results for different runs over the same datasets, which is considered as one of the potential disadvantages of this algorithm [43]. This mix is not sensitive to center initialization, but it still has tendency towards local optimality. In this algorithm, strong ties among data points and the nearest data centers cause cluster centers not to exit from their local dense ranges [44].

The algorithm of bees, first developed by Karaboga and Basturk [3] and Pham et al. in 2006 [30], is a new swarm-based algorithm to search solutions independently. The algorithm was inspired by the behavior of food foraging from swarms of honeybees. In classic edition, the algorithm used random search to find neighborhood to solve optimization problems and issues.

2.2. Algorithm for Finding Cluster Initial Centers. In this study, with regards to efficiency purposes, all data objects are first clustered using k -means algorithm to find the initial cluster centers to be used in the solutions based on all their attributes. Based on the generated clusters, the pattern for an object is produced from each attribute at any stage.

Objects with the same patterns are located in one cluster and hence all objects are clustered. The obtained clusters in this stage will be more than the original number of clusters. For more information, refer to paper [6]. In this paper, clustering is completed in two stages. The first stage is performed as discussed above and in the second stage similar clusters are integrated with each other until achieving a given number of clusters. Algorithm 1 shows the proposed approach for initial clustering of data objects and the achieved cluster centers are called seed points.

As can be observed in Algorithm 1, for every attribute of data objects, a cluster label is generated and this label is added to the data object pattern. Objects with identical patterns are placed in one cluster. To produce data object labels based on each attribute, first the mean and standard deviation of that attribute are computed for all data objects. Thereafter, based on the mean and standard deviation, the range of attribute

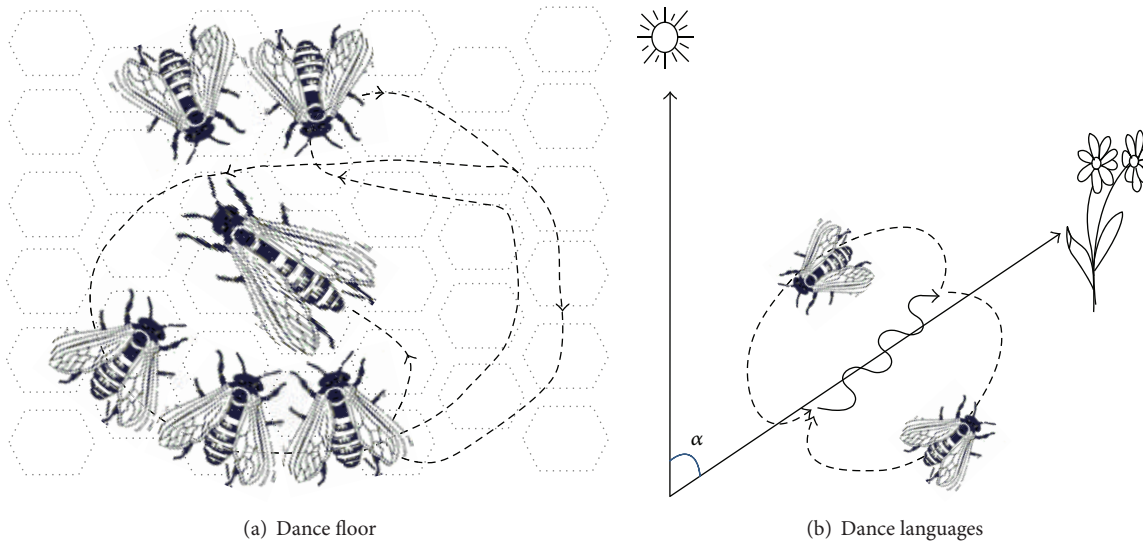


FIGURE 1: Waggle dance of the honeybee.

values are broken into k identical intervals so that the tail of each interval appears as an initial cluster center. Thus, using the initial centers, all data objects are clustered by the k -means method.

2.3. Fitness Function. To calculate the fitness of each solution, the distance between the centers of clusters and each data will be used. To do this, first a set of cluster centers will be generated randomly and then clustering of the numerator will be conducted based on (2). Now, according to centers obtained in the interaction step, the new centers of the clusters and fitness of solutions based on (3) will be calculated [40]

$$\text{Fitness}(C) = \sum_{i=1}^k \sum_{x_j \in C_i} \|x_j - m_i^*\|. \quad (4)$$

3. The Dance Language of Bees

For honeybees, finding nectar is essential to survival. Bees lead others to specific sources of food and then scout bees start to identify the visited resources by making movements as “dancing.” These dances are very careful and fast in different directions. Dancers try to give information about a food resource by specifying the direction, distance, and quality of the visited food source [45].

3.1. Describing the Dance Language. There are two kind of dance for Observed bees including “round dance” and “waggle dance” [46]. When a food resource is less than fifty meters away, they do round dance and when a food resource is greater than fifty meters away, they perform waggle dance (Figure 1).

There are some concepts in this dance, in which the angle between vertical and waggle run is equal to the angle between the sun and food resource. Dance “tempo” shows the distance of food resource (Figure 2). A slower dance tempo means that

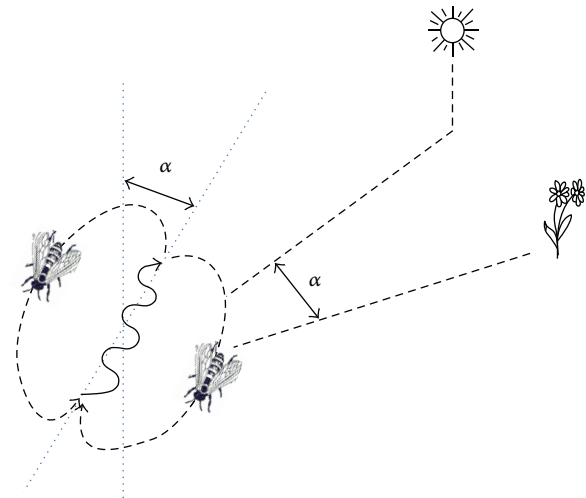


FIGURE 2: Communication and information sharing.

a food resource is farther and vice versa [47]. Another concept is the duration of dance and a longer dance duration means that a food resource is rich and better [45]. Audiences are other bees, which follow the dancer. In this algorithm, there are two kinds of bees, SCOUTS are bees that find new food sources and perform the dance. RECRUTTS are bees that follow the scout bees, dance, and then forage. One of the first people that translate the waggle dance mining was Austrian etologist Karl von Frisch.

Distance between flowers and hive is demonstrated by the duration of the waggle dance. The flowers that are farther from the hive have longer waggle dance duration. Each hundred meters distance between flowers from the hive is shown in the waggle dance phase close to 75 milliseconds.

3.2. Bee in Nature. A colony of honeybees can extend itself over long distances (more than 10 km) and in multiple

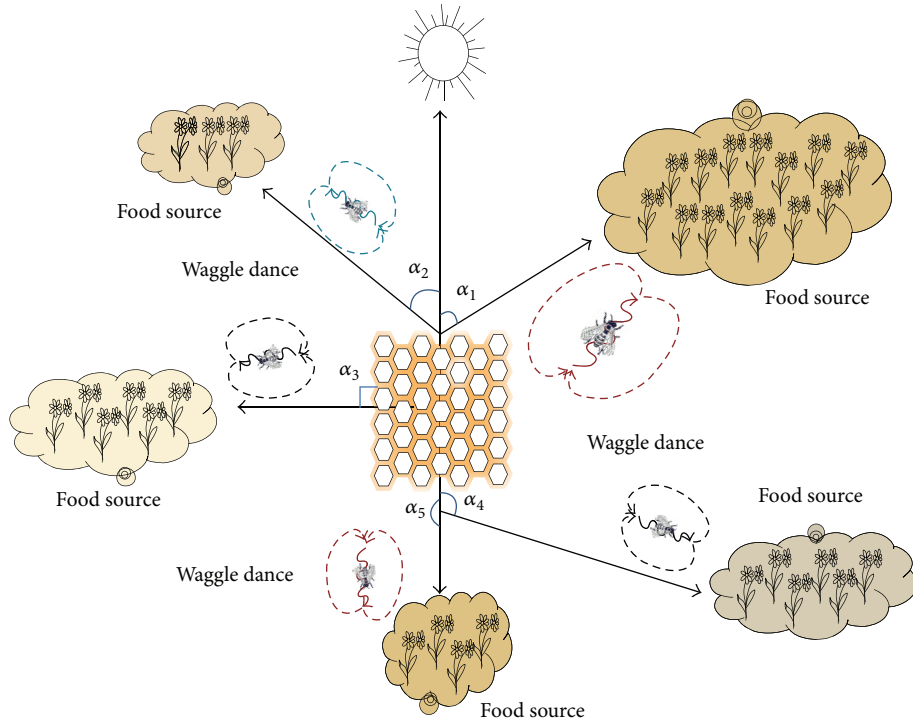


FIGURE 3: The intelligent foraging behavior of honeybee colony.

- (1) Initialize population with random solutions. (n scout bees are placed randomly in the search space.)
- (2) Evaluate fitness of the population.
- (3) While (Repeat optimization cycles for the specified number)
- (4) Select sites for neighborhood search. (Bee that have the highest fitness are chosen as “selected” and sites visited by them are chosen for neighborhood search.)
- (5) Recruit bees for selected sites (more bees for best e sites) and evaluate fitness.
- (6) Select the fittest bee from each patch. (For each patch, only the bee with the highest fitness will be selected to form the next bee population.)
- (7) Assign remaining bees to search randomly and evaluate their fitness.
- (8) End While

ALGORITHM 2: Pseudocode of Basic Bee Algorithm.

directions simultaneously to exploit a large number of food sources. In principle, flower patches with plentiful amount of nectar or pollen that can be collected with less effort should be visited by more bees, whereas, patches with less nectar or pollen should receive fewer bees [35, 47].

The foraging process begins in a colony with the scout bees being sent out to search for promising flower patches. Scout bees move randomly from one patch to another. During the harvest season, a colony continues its exploration, keeping a percentage of the population as scout bees. When the scout bees return to the hive, those that found a patch, which is rated above a certain quality threshold (measured as a combination of some constituents, such as sugar content), deposit their nectar or pollen and go to the “dance floor” to perform a dance known as “waggle dance” [46]. The waggle dance is essential for colony communication and contains three pieces of information regarding a flower patch:

the direction in which it will be found, its distance from the hive, and its quality rating (or fitness). This information helps the colony to send its bees to flower patches precisely, without using guides or maps [45]. After the waggle dance on the dancing floor, the dancers (i.e., scout bee) go back to the flower patch with follower bees that are waiting inside the hive. More follower bees are sent to patches that are most promising [48, 49]. The flowchart of bee algorithm is shown in Figure 4 [50].

The Basic Bee Algorithm is shown as in Algorithm 2 [51].

4. Differential Evolution

Differential evolution is a type of standard genetic algorithm. Differential evolution algorithm evaluates the initial population by using probability motion and observation models and population evolution is performed by using evolution

- (1) **Begin**
 (2) First j_0 is selected randomly between 1 and D
 (3) j_0 is added to set J
 (4) For all values of j the following operations are repeated:
 (a) One random number is generated such as rand_j that has uniform distribution between zero and one
 (b) If r_j is less than or equal to P_{cr} then number of j is added to J set
 (5) **End.**

ALGORITHM 3: Pseudocode of the binomial crossover.

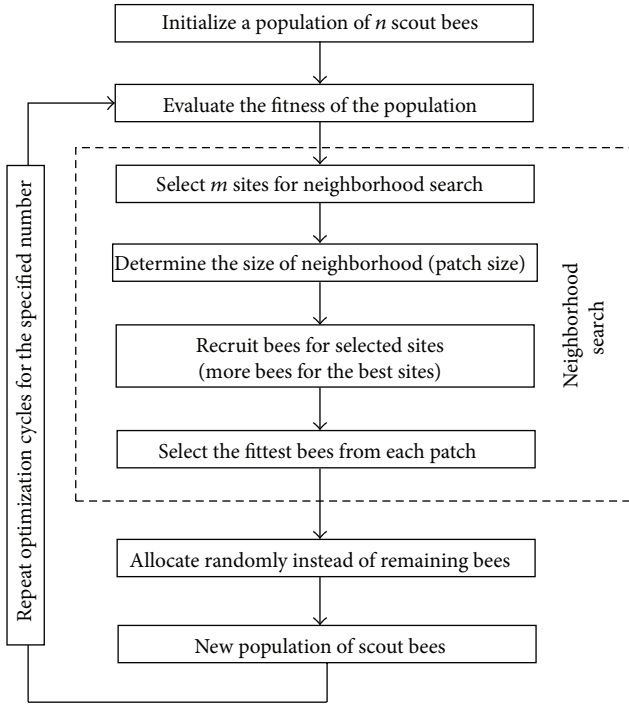


FIGURE 4: The flowchart of bee algorithm.

operators [52]. The main idea in the differential evolution algorithm is to generate a new solution for each solution by using one constant member and two random members. In each generation, the best member of population is selected and then the difference between each member of population and the best member is calculated. Two random members are then selected and the difference between them is calculated. Coefficient of this difference is added to i th member and thus a new member is created. The cost of each new member is calculated and if the cost value of the new member is less, the i th member is replaced instead of i th member; otherwise, the previous value can be kept in the next generation [35].

Differential evolution is one of the population based popular algorithms that uses point floating (real coded) for presentation as follows [53]:

$$Z_i(t) = [Z_{i,1}(t), Z_{i,2}(t), \dots, Z_{i,D}(t)], \quad (5)$$

where t is the number of generation (iteration), i refers to members (population), and d is the number of optimization parameters. Now, in each generation (or each iteration of

algorithm) to perform changes on members of population $Z_i(t)$, one donor vector $Y_i(t)$ is formed. The various methods of DE are used to determine how to make the donor vector. The first kind of DE named 1/rand/DE generates i th member $Y_i(t)$, in which three members of current generation (r_1, r_2, r_3) are chosen randomly as

$$r_1 \neq r_2 \neq r_3 \in \{1, 2, \dots, D\}. \quad (6)$$

Then, the difference between two vectors from three selected vectors are calculated and multiplied by F coefficient and with the third vector added [53]. Therefore, donor vector $Y_i(t)$ is obtained. Calculation process of donor vector for j th element from i th vector can be demonstrated as follows [54]:

$$Y_{i,j}(t) = Z_{r_1,j}(t) + F(Z_{r_2,j}(t) - Z_{r_3,j}(t)). \quad (7)$$

To increase the exploration of algorithm a crossover operation is then performed. Differential algorithm has generally two kinds of crossover exponential and binomial [55]. In this paper to save time, the binomial mode has been used. To apply the binomial crossover, it requires that set of J is constituted as in Algorithm 3.

Therefore, for each target vector $Z_i(t)$, there is a trial vector as follows [56]:

$$R_{i,j}(t) = \begin{cases} Y_{i,j}(t) & \text{If } \text{rand}_{j,i} \leq P_{cr} \text{ or } j = j_0 \quad i = 1, 2, \dots, N \\ Z_{i,j}(t) & \text{If } \text{rand}_{j,i} > P_{cr} \text{ and } j \neq j_0 \quad j = 1, 2, \dots, D, \end{cases} \quad (8)$$

where j is equal to $j = 1, 2, \dots, D$ and rand_j is uniform distribution number between $[0, 1]$. Set of J is guaranteed where there is at least one difference between $R_i(t)$ and $Z_i(t)$. In the next step, the selection process is performed between target vector and trial vector as follows:

$$Z_i(t+1) = \begin{cases} R_i(t) & \text{If } f(R_i(t)) \leq f(Z_i(t)) \quad i = 1, 2, \dots, N \\ Z_i(t) & \text{otherwise,} \end{cases} \quad (9)$$

where $f(\cdot)$ is a function that should be the minimum. In this paper, to escape from premature convergence, two new strategies of merging have been studied. In the basic DE are used difference vector of $Z_{r_2,j}(t) - Z_{r_3,j}(t)$ multiplied F

- (1) Define algorithm parameter
- (2) Generate and evaluate initial population or solutions
- (3) For all members of population per form the following steps
 - (a) With mutation operator create a new trial solution
 - (b) By using the crossover generate new solution and evaluate them
 - (c) Replace new solution with current solution if new solution is better than current solution otherwise, the current solution is retained.
- (4) Return to step three if termination condition is not achieved.

ALGORITHM 4: Pseudocode of differential evolution.

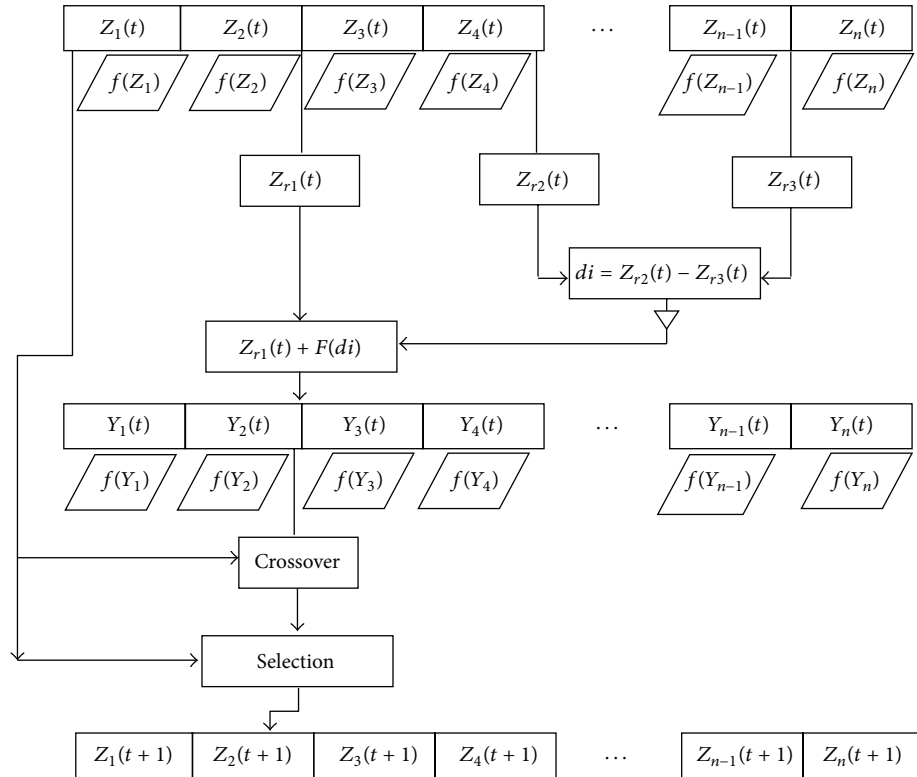


FIGURE 5: The process of differential evolution.

where F is control parameter between 0.4 and one [55, 57]. To improve the convergence feature in the DE, this paper makes the following proposal:

$$F = 0.5 (\text{Rnd} + 1), \tag{10}$$

where Rnd is uniform distribution number between zero and one. Generally, the DE algorithm steps are as in Algorithm 4.

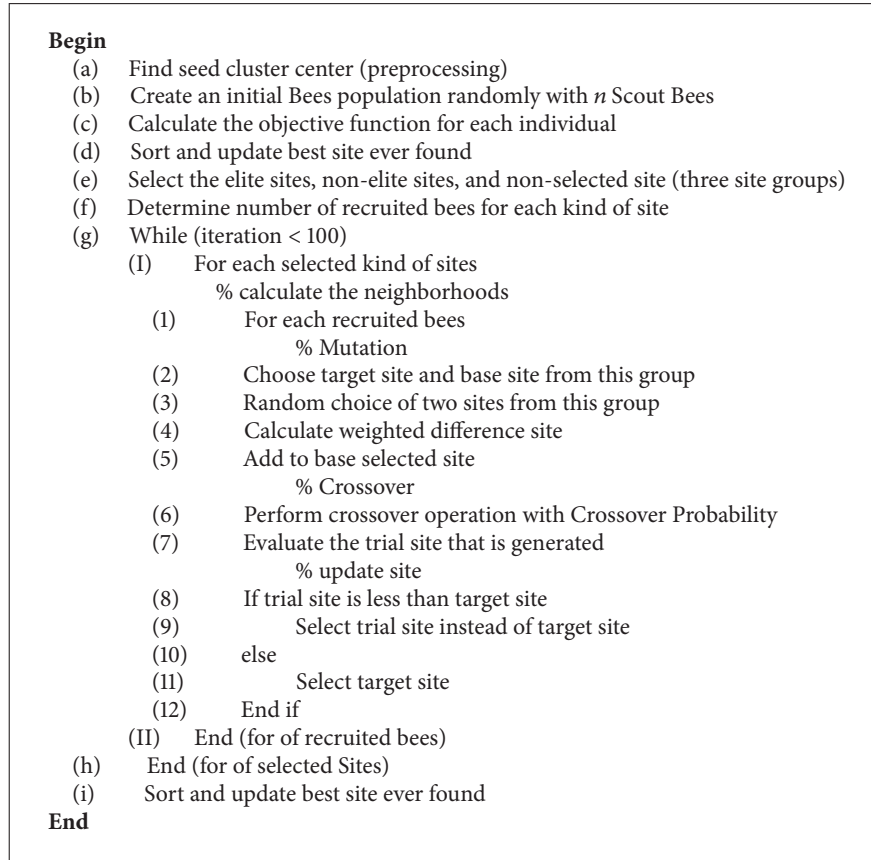
In Figure 5, the process of differential evolution is illustrated.

5. Proposed Algorithm

As noted in the former sections, studies conducted on the BA method have shown that this algorithm can be a powerful approach with enough performance to handle different types of nonlinear problems in various fields. However, it can be possibly trapped into local optimum. Lately, several ideas

have been used to reduce this problem by hybrid different evolutionary techniques such as partial swarm optimization, genetic algorithm, and simulating annealing. In most population based evolutionary algorithms, in each iteration, new members are generated and then the movement operations are applied to explore new positions based on providing better opportunities. To increase the diversity of algorithm, in the differential evolution algorithm, all members have a possibility to win the global optimum and move to that side. The ability of the best particle to local search also depends on the other particles by selecting the two other particles and calculating the difference between them. This situation may lead to local convergence.

In this proposed algorithm, to escape from random selecting of the global best particle, we used competency selection for choosing the global best particle. If particle is better than the other solutions, then the probability of being selected is greater.



ALGORITHM 5: Pseudocode of proposed CCIA-BADE-K algorithm.

The basic idea behind the proposed algorithm is that our solutions are grouped based on the bees' algorithm.

On the other hand, in this algorithm new approach is proposed to the movement and selects the recruiting bees for selecting sites. This algorithm classified the bees into three groups and named them elite sites, nonelite sites, and nonselected site. To increase diversity, the two modes for movement based on the differential evolution algorithm operator as parallel mode and serial mode were used. The suggested algorithm tries to use the advantage of these algorithms to find the best cluster center and to improve simulation results. In other words, in this algorithm, first, a preprocessing technique is performed on the data and then the proposed hybrid algorithm is used to find the best cluster center for k -means problem.

The flowchart and pseudocode of the combined algorithm, called CCIA-BA-DE, are illustrated in Algorithm 5 and Figure 6.

6. Application of CCIA-BA-BE on Clustering

The application of CCIA-BADE-K algorithm on the clustering problem in this section is presented. To perform the CCIA-BADE-K algorithm to find best cluster centers, the following steps should be repeated and taken.

Step 1 (generate the seed cluster center). This step is a preprocessing step to find the seed cluster center to choose the best interval for each cluster.

Step 2 (generate the initial bees' population randomly). In other words, generate initial solutions to find the best cluster centers statistically

$$\text{Population} = \begin{bmatrix} \text{center}_1^T \\ \text{center}_2^T \\ \vdots \\ \text{center}_{n\text{Scout}}^T \end{bmatrix}, \quad (11)$$

where center is a vector with k cluster and each vector has p dimension:

$$\text{center}_i = [C_1, C_2, \dots, C_k], \quad i = 1, 2, \dots, n\text{Scout},$$

$$C_j = [c_1, c_2, \dots, c_p], \quad j = 1, 2, \dots, k, \quad (12)$$

$$C_j^{\min} < C_j < C_j^{\max},$$

where C_j is cluster center of j for i th scout bee and p is the number of dimension for each cluster center. In fact, each

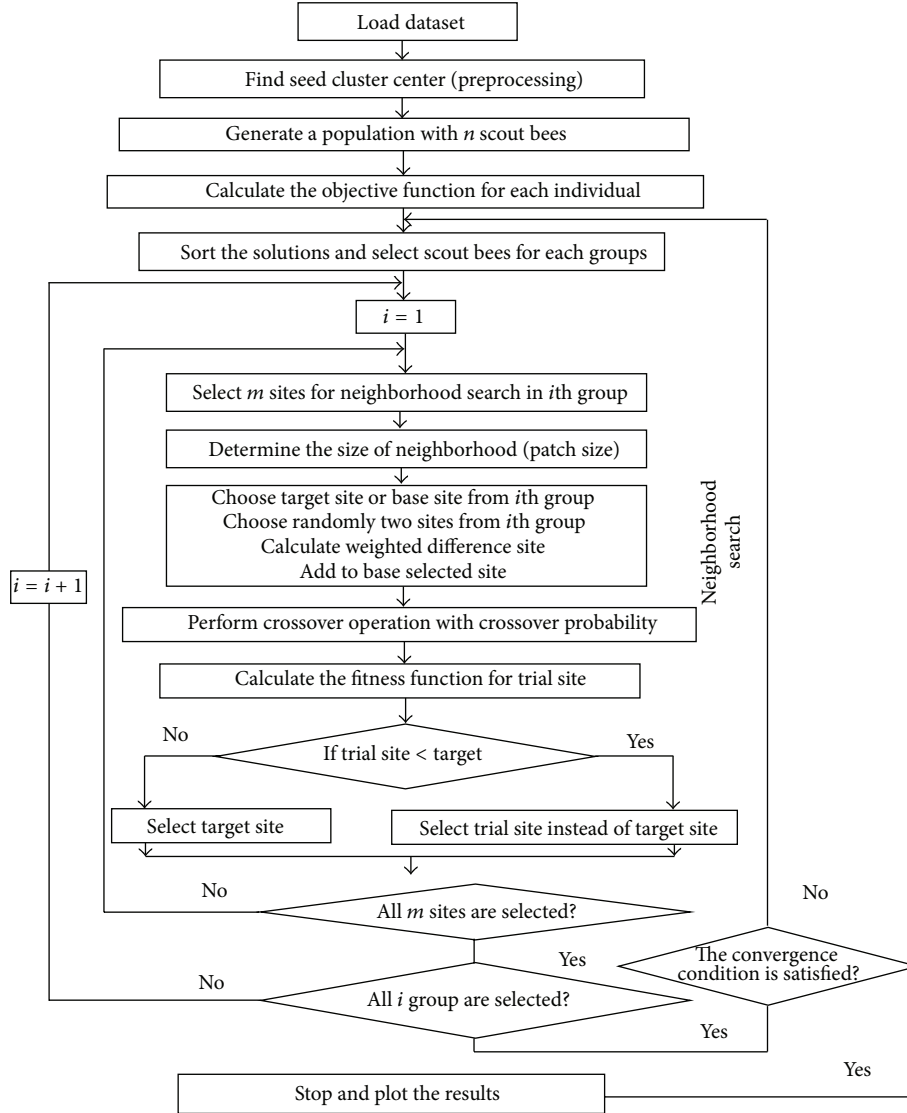


FIGURE 6: Flowchart of proposed algorithm.

solution in the algorithm is a matrix with $k \times p$. c_i^{\min} and c_i^{\max} are values of the minimum and maximum for each dimension (each feature of center).

Step 3 (calculate the objective function for each individual). Calculate the cost function for each solution (each site) in this algorithm.

Step 4 (sort the solutions and select scout bees for each groups). The sorting of the site is carried out based on the objective function value.

Step 5 (select the first group of sites). Finding the new solutions is performed by selecting the group of sites. There are three groups of sites in which the first group or elite sites are evaluated to find the neighbors of the selected site followed by nonelite site and finally nonselected sites. To find the neighbors of each group of sites, either the serial mode or parallel mode may be used. This algorithm used parallel model.

Step 6 (select the number of bees for each site). Numbers of bees for each site depend on their group and are considered as competence, more bees for better site. If the site is rich then more bees are allocated to this site. In other words, if the solution is better, it is rated as more important than the other sites.

Step 7 (performing the differential evolution operator (mutation)). In this step, the target site is chosen from the group sites and then two other sites from this group are selected randomly to calculate the weighted difference between them. After calculating this difference, it is added to base trial site as shown in the following equation:

$$v_{i,G+1} = x_{r1,G} + W(x_{r2,G} - x_{r3,G}), \quad (13)$$

$$r_1 \neq r_2 \neq r_3 \in \{1, 2, \dots, N\},$$

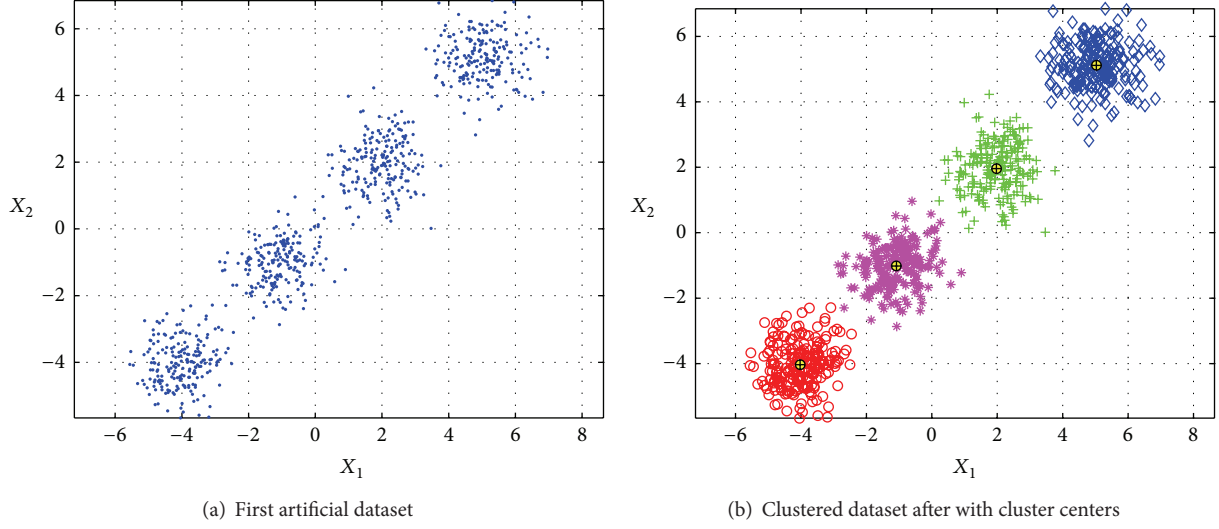


FIGURE 7: Used artificial dataset one.

where $x_{r1,G}$ is the target site, W is the weight, and the $x_{r2,G}$, $x_{r3,G}$ are the selected sites from target's group. $v_{i,G+1}$ is the trial solution for comparison purposes.

Step 8 (perform crossover operation with crossover probability). The recombination step incorporates successful solutions from the previous generation. The trial vectors $u_{i,G+1}$ is developed from the elements of the target vector, $x_{i,G}$ and the elements of donor vector $v_{i,G+1}$. Elements of the donor vector enter the trial vector with probability CR

$$u_{j,i,G+1} = \begin{cases} v_{j,i,G+1} & \text{If } \text{rand}_{j,i} \leq \text{CR} \text{ or } j = I_{\text{rand}} \quad i = 1, 2, \dots, N \\ x_{j,i,G} & \text{If } \text{rand}_{j,i} > \text{CR} \text{ and } j \neq I_{\text{rand}} \quad j = 1, 2, \dots, L, \end{cases} \quad (14)$$

where $\text{rand}_{j,i} \sim U[0, 1]$ and I_{rand} is a random integer from $[1, 2, \dots, L]$ and I_{rand} ensures that $u_{i,G+1} \neq x_{i,G}$.

Step 9 (calculate the cost function for trial site). In the selection step, target vector $x_{i,G}$ is compared with trial vector $u_{i,G+1}$. There are two modes to calculate the new site as follows:

$$x_{i,G+1} = \begin{cases} u_{i,G+1} & \text{If } f(u_{i,G+1}) \leq f(x_{i,G}) \quad i = 1, 2, \dots, N \\ x_{i,G} & \text{otherwise.} \end{cases} \quad (15)$$

Trial vector $u_i(G)$ is compared to target vector $x_i(G)$. To use greedy criterion, if $u_i(G)$ is better than the $x_i(G)$, then replace $x_i(G)$ with $u_i(G)$; otherwise, $x_i(G)$ "survive" and $u_i(G)$ are discarded.

Step 10. If not all sites from this group are selected, go to Step 6 and select another site from this group; otherwise, go to the next step.

TABLE 1: Table type styles.

Dataset name	Dataset attribute		
	Dataset size	Cluster number	Attribute number
Iris	150 (50, 50, 50)	3	4
Wine	178 (59, 71, 48)	3	13
CMC	1473 (629, 334, 510)	3	9
Glass	214 (70, 17, 76, 13, 9, 29)	6	9

Step 11. If not all groups are selected, go to Step 5 and select the next group; otherwise, go to the next step.

Step 12 (check the termination criteria). If the current number of iteration does not reach the maximum number of iterations, go to Step 4 and start next generation; otherwise, go to the next step.

7. Evaluation

To evaluate the accuracy and efficiency of the proposed algorithm, experiments have been performed on two artificial datasets, four real-life datasets and four standard datasets to determine the correctness of clustering algorithms. This collection includes Iris, Glass, Wine, and Contraceptive Method Choice (CMC) datasets that have been chosen from standard UCI dataset.

The suggested algorithm is coded by an appropriate programming language and is run on an i5 computer with 2.60 GHz microprocessor speed and 4 GB main memory. For measuring the performance of the proposed algorithm, the benchmarks data items of Table 1 are used.

The execution results of the proposed algorithm over the selected datasets as well as the comparison figures relative to K -means, PSO, and K -NM-PSO results in [10] are tabulated in Table 2. As easily seen in Table 2, the suggested

TABLE 2: The obtained results from implementing the suggested algorithm over selected datasets.

Dataset	K-means [10]	PSO [10]	K-NM-PSO [10]	Proposed Alg.	
				Result	CPU Time (S)
Iris	97.33	96.66	96.66	96.5403	~15
Wine	16555.68	16294.00	16292.00	16,292.25	~30
CMC	5542.20	5538.50	5532.40	5532.22	~57
Glass	215.68	271.29	199.68	210.4318	~34

TABLE 3: The results of implementing the algorithms over Iris test data for 100 runs.

Method	Result			CPU time (S)
	Best	Average	Worst	
PSO-ACO-K	96.650	96.650	96.650	~16
PSO-ACO	96.654	96.654	96.674	~17
PSO	96.8942	97.232	97.897	~30
SA	97.457	99.957	102.01	~32
TS	97.365	97.868	98.569	~135
GA	113.986	125.197	139.778	~140
ACO	97.100	97.171	97.808	~75
HBMO	96.752	96.953	97.757	~82
PSO-SA	96.66	96.67	96.678	~17
ACO-SA	96.660	96.731	96.863	~25
<i>k</i> -Means	97.333	106.05	120.45	0.4
MY proposed ALG.	96.5403	96.5412	96.5438	~15

TABLE 4: The results of implementing the algorithms over Wine test data for 100 runs.

Method	Result			CPU time (S)
	Best	Average	Worst	
PSO-ACO-K	16,295.31	16,295.31	16,295.31	~30
PSO-ACO	16,295.34	16,295.92	16,297.93	~33
PSO	16,345.96	16,417.47	16,562.31	~123
SA	16,473.48	17,521.09	18,083.25	~129
TS	16,666.22	16,785.45	16,837.53	~140
GA	16,530.53	16,530.53	16,530.53	~170
ACO	16,530.53	16,530.53	16,530.53	~121
HBMO	16,357.28	16,357.28	16,357.28	~40
PSO-SA	16,295.86	16,296.00	16,296.10	~38
ACO-SA	16,298.62	16,310.28	16,322.43	~84
<i>k</i> -Means	16,555.68	18,061.01	18,563.12	0.7
MY proposed ALG.	16,292.25	16,293.76	16,294.98	~30

algorithm provides superior results relative to *K*-means and PSO algorithms. The real-life datasets compared with several optimization algorithms are included.

For better study and analysis of the proposed approach, the execution results of the proposed approach along with HBMO, PSO, ACO-SA, PSO-ACO, ACO, PSO-SA, TS, GA, SA, and *k*-means clustering algorithm results as reported

TABLE 5: The results of implementing the algorithms over CMC test data for 100 runs.

Method	Result			CPU time (S)
	Best	Average	Worst	
PSO-ACO-K	5,694.28	5,694.28	5,694.28	~31
PSO-ACO	5,694.51	5,694.92	5,697.42	~135
PSO	5,700.98	5,820.96	5,923.24	~131
SA	5,849.03	5,893.48	5,966.94	~150
TS	5,885.06	5,993.59	5,999.80	~155
GA	5,705.63	5,756.59	5,812.64	~160
ACO	5,701.92	5,819.13	5,912.43	~127
HBMO	5,699.26	5,713.98	5,725.35	~123
PSO-SA	5,696.05	5,698.69	5,701.81	~73
ACO-SA	5,696.60	5,698.26	5,700.26	~89
<i>k</i> -Means	5,842.20	5,893.60	5,934.43	0.5
MY proposed ALG.	5,532.22	5,532.45	5,532.85	~57

TABLE 6: The results of implementing the algorithms over Glass test data for 100 runs.

Method	Result			CPU time (S)
	Best	Average	Worst	
PSO-ACO-K	199.53	199.53	199.53	~31
PSO-ACO	199.57	199.61	200.01	~35
PSO	270.57	275.71	283.52	~400
SA	275.16	282.19	287.18	~410
TS	279.87	283.79	286.47	~410
GA	278.37	282.32	286.77	~410
ACO	269.72	273.46	280.08	~395
HBMO	245.73	247.71	249.54	~390
PSO-SA	200.14	201.45	202.45	~38
ACO-SA	200.71	201.89	202.76	~49
<i>k</i> -Means	215.74	235.5	255.38	~1
MY proposed ALG.	210.431	215.54	216.93	~34

in [8] are tabulated in Tables 3–6. It is worth mentioning that the investigated algorithms of [8] are implemented with MATLAB 7.1, using a Pentium IV system of 2.8 GHz CPU speed and 512 MB main memory.

Frist artificial dataset includes ($n = 800, k = 4, d = 2$) where n is the number of instance, k is the number of clusters, and d is the number of dimensions. The instances were drawn

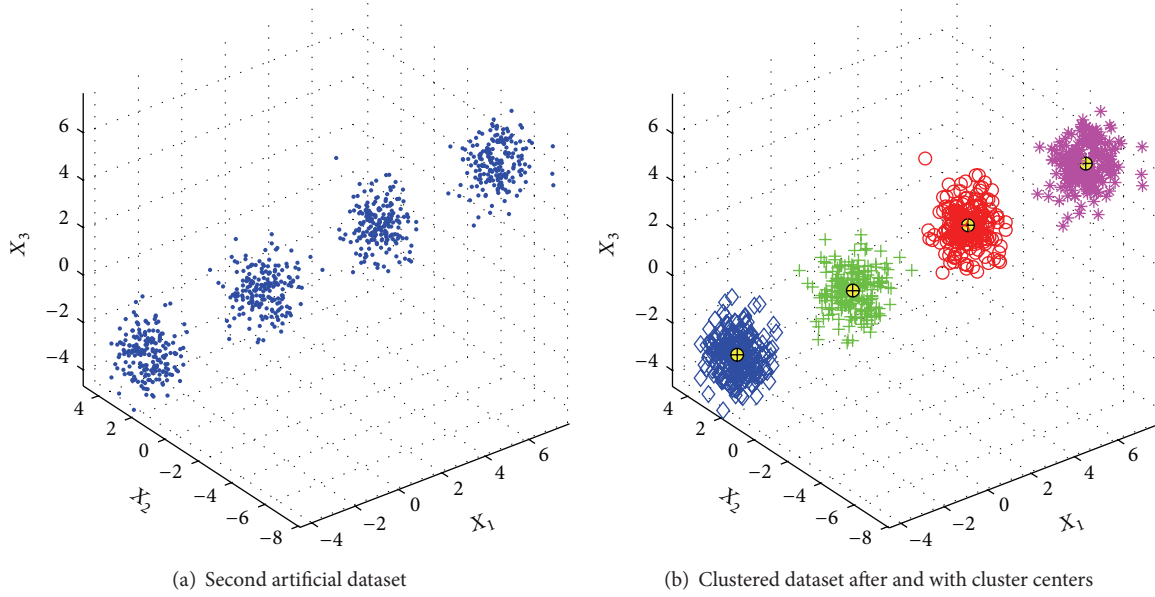


FIGURE 8: Used artificial dataset two.

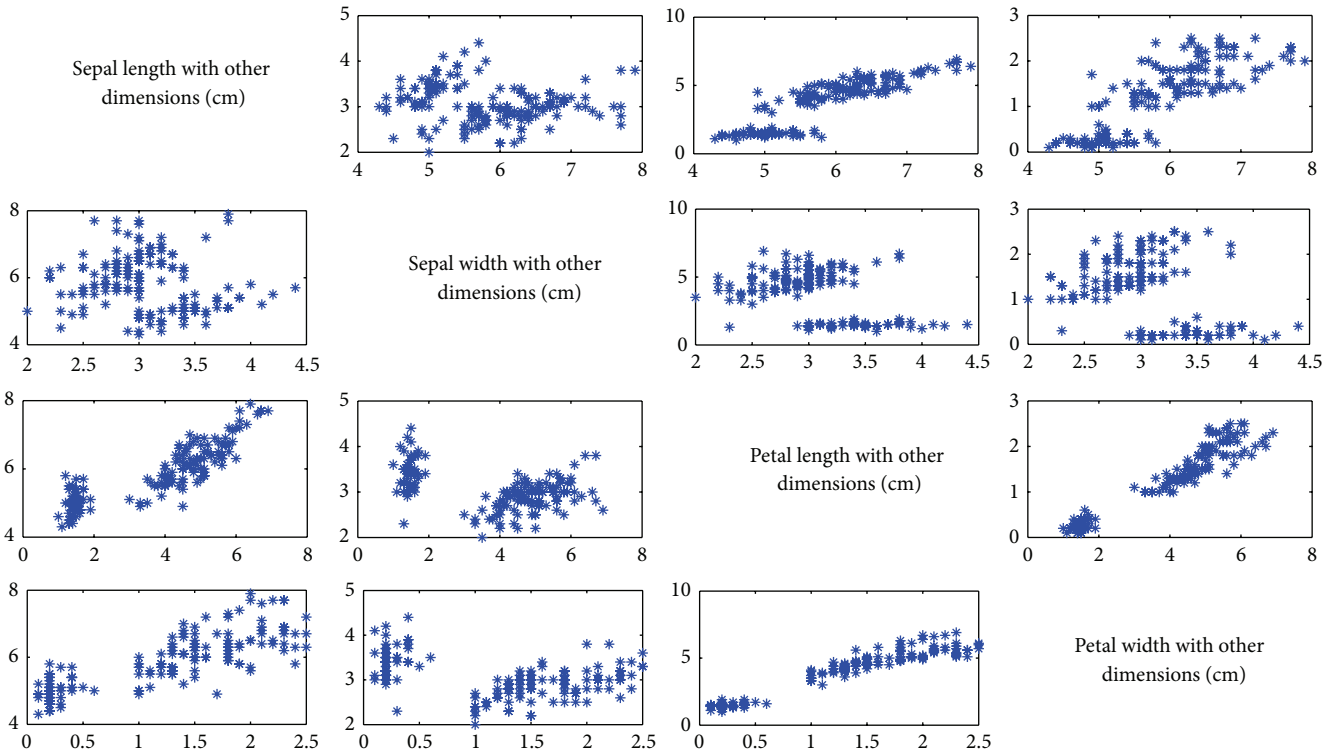


FIGURE 9: The scatter plot to show nonlinear relationship between variables for Iris dataset.

for four absolute classes where each of these groups was distributed as

$$\text{Art}_1 \left(\mu = \begin{pmatrix} m_i \\ m_i \end{pmatrix}, \Sigma = \begin{bmatrix} 0.5 & 0.05 \\ 0.05 & 0.5 \end{bmatrix} \right) \quad i = 1, 2, 3, 4 \quad (16)$$

$$m_1 = -4, \quad m_2 = -1, \quad m_3 = 2, \quad m_4 = 5,$$

where Σ and μ are covariance matrix and vector, respectively [10]. The first artificial dataset is demonstrated in Figure 7(a). Figure 7(b) illustrated the clustered data after applying CCIA-BADE-K algorithm on data.

Second artificial dataset includes ($n = 800, k = 4, d = 3$) where n is the number of instance, k is the number of clusters, and d is the number of dimensions. The instances were drawn

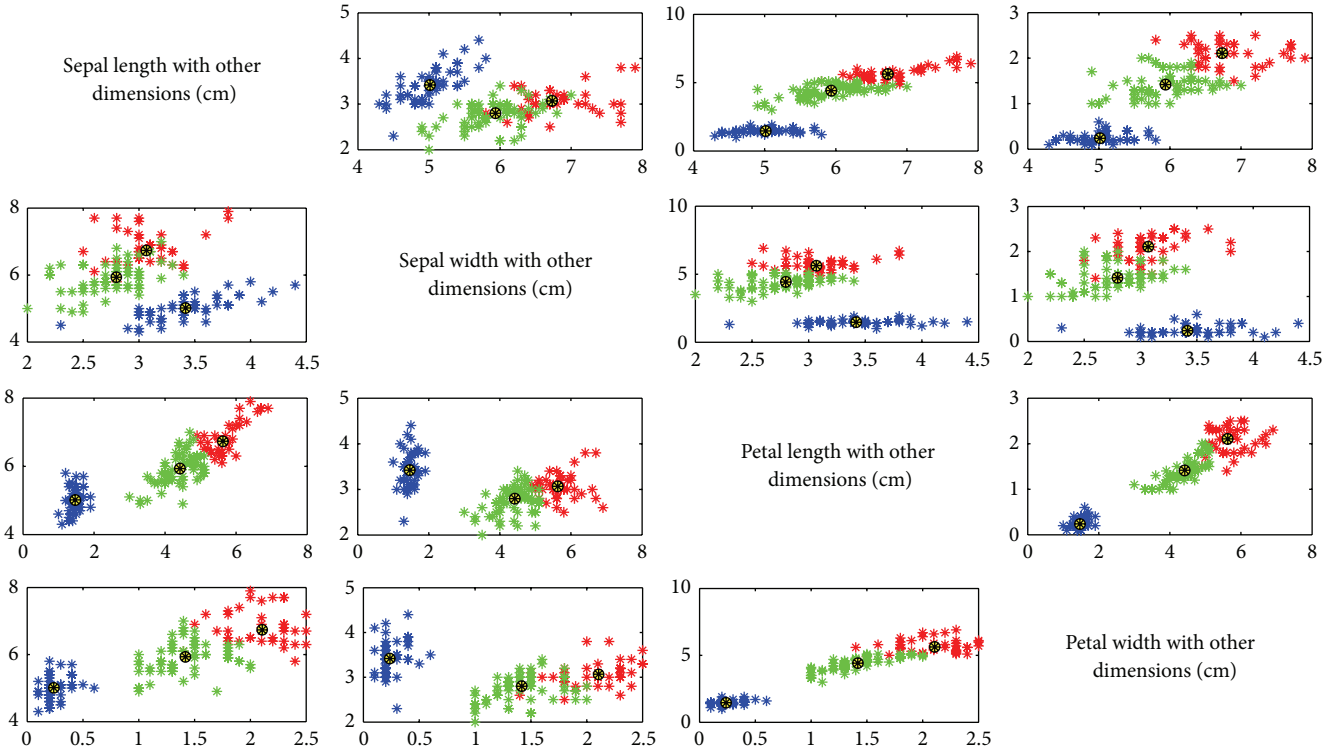


FIGURE 10: Clustered scatter plot to show nonlinear relationship between variables for Iris dataset.

for four absolute classes where each of these groups was distributed as

$$\text{Art}_2 \left(\mu = \begin{pmatrix} m_i \\ -m_i \\ m_i \end{pmatrix}, \Sigma = \begin{bmatrix} 0.5 & 0.05 & 0.05 \\ 0.05 & 0.5 & 0.05 \\ 0.05 & 0.05 & 0.5 \end{bmatrix} \right)$$

$$i = 1, 2, 3, 4 \quad m_1 = -3, \quad m_2 = 0, \quad m_3 = 3, \quad m_4 = 6, \quad (17)$$

where Σ and μ are covariance matrix and vector, respectively [10]. The second artificial dataset is demonstrated in Figure 8. Figure 8 shows clusters after applying proposed algorithm on the artificial dataset.

In Tables 3–6, best, worst, and average results are reported for 100 runs, respectively. The resulting figures represent the distance of every data from the cluster center to which it belongs and is computed by using relation (4). As observed in the table, regarding the execution time, the proposed algorithm generates acceptable solutions.

To clarify the issue, in Figure 10, the scatterplot (scatter-graph) is illustrated. The scatter-graph is one kind of mathematic diagram, which shows the values for a dataset for two variables using Cartesian coordinates. In this diagram, data is demonstrated as a set of spots. This type of diagram is known as a scatter diagram or scatter-gram. This kind of diagram is also used to display relation between response variables with control variables when a variable is below the control of the experimenter. One of the strongest aspects of the scatter-diagram is the ability to show nonlinear relationship between

variables. In Figure 9, the scatter-diagram of Iris dataset is displayed and in Figure 10 the clustered Iris data on the scatter-diagram is shown.

In Table 4, best, worst, and average results of Wine dataset are reported for 100 runs. The resulting figures represent the distance of every data from the cluster center.

In Figure 11 best cost and average best costs of results for all datasets are reported for 100 runs. The resulting figures represent the distance of every data from the cluster center by using relation (4). Figure 11(a) is related to the best cost and mean of best cost for Iris dataset, and Figure 11(b) illustrated the best cost and mean of best cost for Wine dataset. Figure 11(c) reported best cost and mean of best cost for CMC datasets, and finally Figure 11(d) demonstrated mean value of best cost and best cost of Glass dataset.

According to the reported results in Tables 3 to 6, the proposed method over Iris, CMC, and Wine Datasets provides the best results in comparison with other mentioned algorithms. According to Table 6, the suggested algorithm over Glass dataset provides more acceptable results than the alternative algorithms. The reason for this behavior is justified by the fact that as data objects increase in number the efficiency of the alternative algorithms decreases while the deficiency of the suggested algorithm highlights more.

8. Image Segmentation

In Section 7, it was shown that the proposed CCIA-BADE-K algorithm is one of the best methods for data clustering. For further investigation of the performance of algorithm,

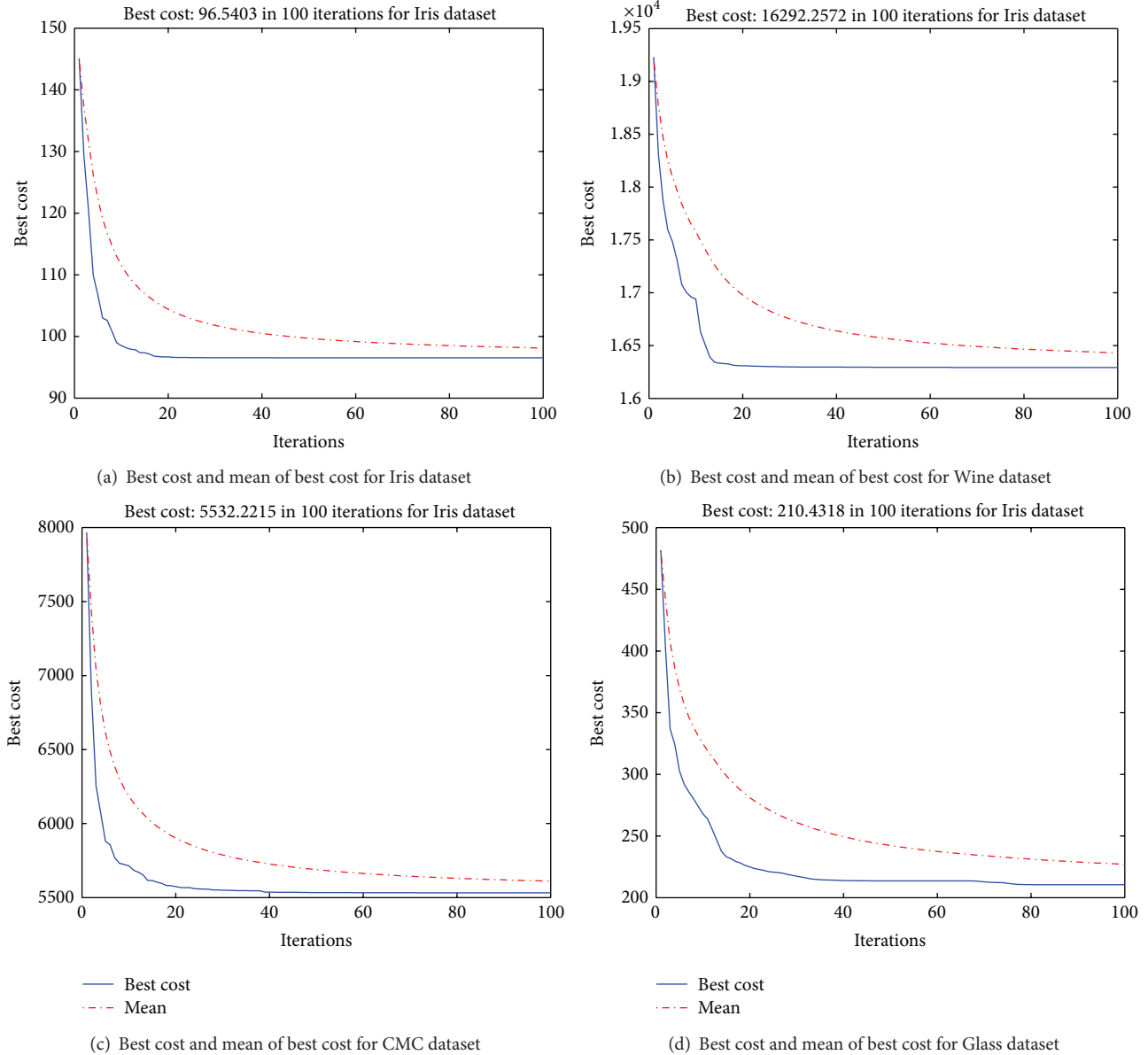


FIGURE 11: Best cost and mean of best cost in 100 iterations.

the algorithm was tested on one standard image and one industrial image. Each digital image in RGB space is formed by three-color components consisting of red, green, and blue. Each of these three alone is a grayscale image and the numerical value of each pixel is between 1 and 255. Image histogram is a chart that is made by the number of pixels on an image that is determined based on the brightness level [58]. To obtain a histogram of image it is enough to scroll the whole pixel of image and to calculate the number of pixels for each brightness level. The normalized histogram is obtained by dividing the total number of histogram value to each value of pixels. Normalizing the histogram causes the histogram value to be in $[0, 1]$ interval. Figures 12 and 13 show that image samples in this paper are shown for image segmentation. In Figure 12, the color, grayscale, and clustered modes of these images are shown and, in Figure 13, histogram

diagrams for these four images are shown. Furthermore, these segmentation charts will be used to detect segmentation an image.

9. Concluding Remarks

In this paper, a new technique based on a combination of bees algorithm and differential evolution algorithm with k -means was presented. In the proposed algorithm, bee algorithm was assigned to perform globally and differential evolution algorithm was assigned to implement local searching on k -means problem, which is responsible for the task of finding the best cluster centers. The new proposed algorithm CCIA-BADE-K applies abilities of both algorithms and, by removing shortcomings of each algorithm, it tries to use its own strengths to cover other algorithm defects as well

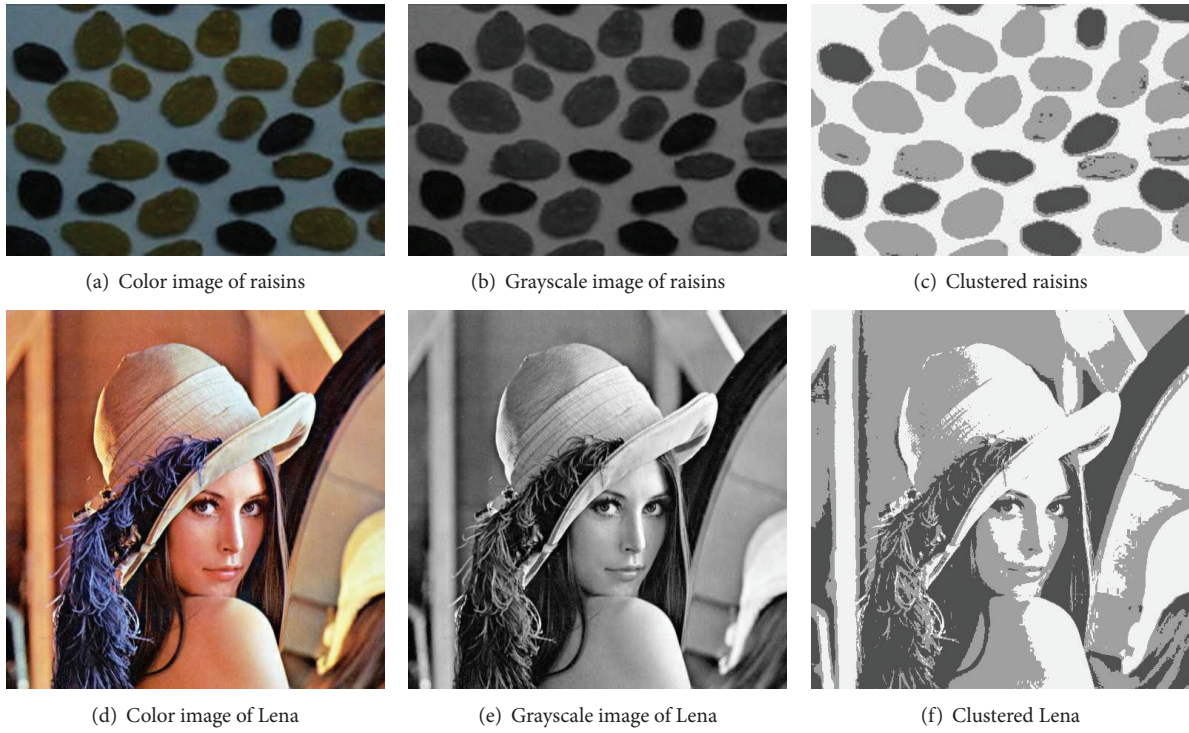


FIGURE 12: Images used for image segmentation with proposed algorithm.

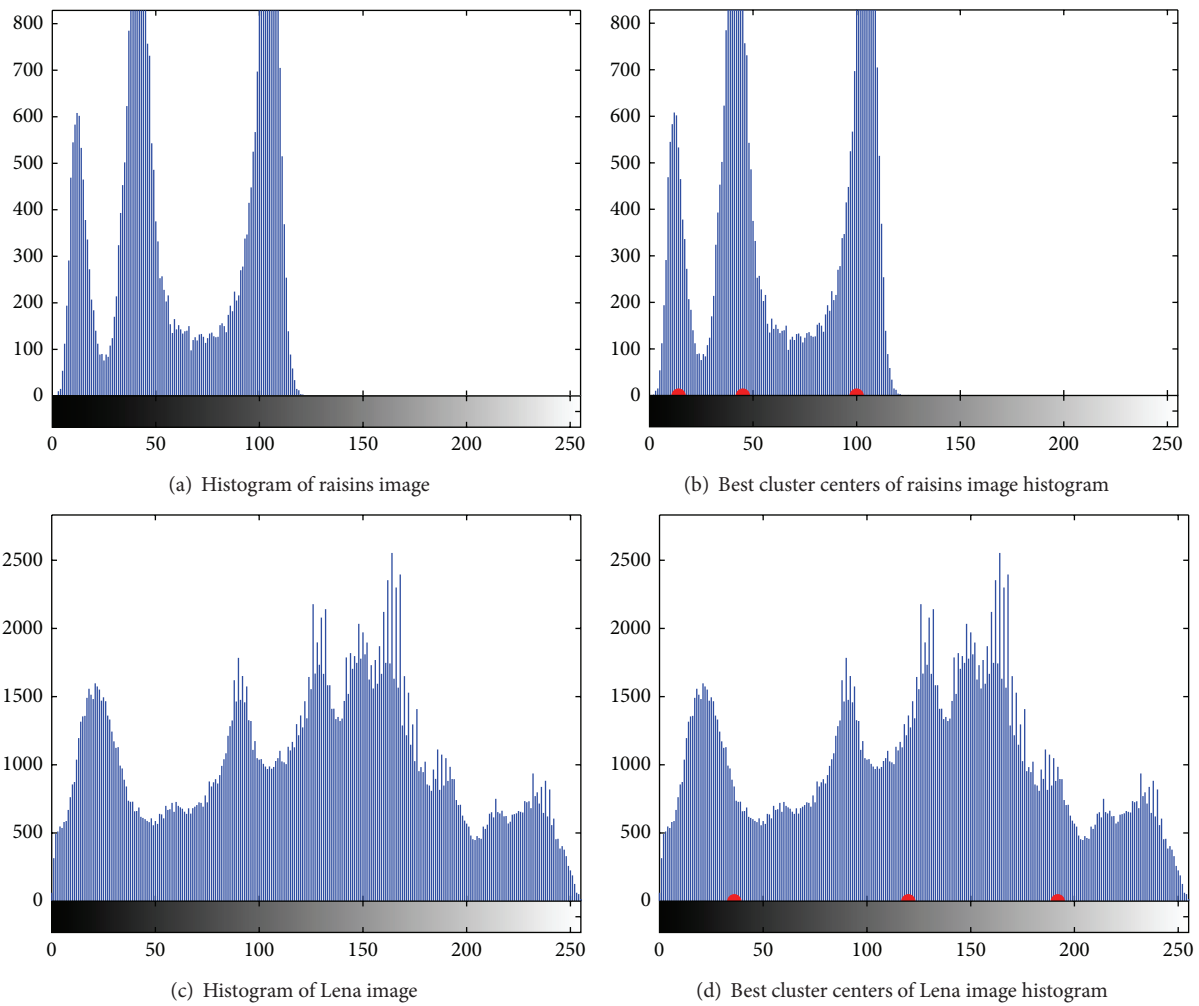


FIGURE 13: Sample used images for image segmentation in the grayscale mode.

as to find best cluster centers that is the proposed seed cluster center algorithm. Experimental results showed that the CCIA-BADE-K algorithm enjoys acceptable results.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to express their cordial thanks to the Ministry of Education (MoE), University Technology Malaysia (UTM), for the Research University Grant no. Q.J130000.2528.06H90. The authors are also grateful to Soft Computing Research Group (SCRG) for their support and incisive comments in making this study a success.

References

- [1] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*, vol. 20, SIAM, 2007.
- [2] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2006.
- [3] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [4] E. Alpaydin, *Introduction to Machine Learning*, MIT Press, 2004.
- [5] S. Bandyopadhyay and U. Maulik, "An evolutionary technique based on K-means algorithm for optimal clustering in \mathbb{R}^N ," *Information Sciences*, vol. 146, no. 1–4, pp. 221–237, 2002.
- [6] S. S. Khan and A. Ahmad, "Cluster center initialization algorithm for K-means clustering," *Pattern Recognition Letters*, vol. 25, no. 11, pp. 1293–1302, 2004.
- [7] G. Hamerly and C. Elkan, "Alternatives to the k-means algorithm that find better clusterings," in *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM '02)*, pp. 600–607, McLean, Va, USA, November 2002.
- [8] T. Niknam and B. Amiri, "An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis," *Applied Soft Computing Journal*, vol. 10, no. 1, pp. 183–197, 2010.
- [9] C. D. Nguyen and K. J. Cios, "GAKREM: a novel hybrid clustering algorithm," *Information Sciences*, vol. 178, no. 22, pp. 4205–4227, 2008.
- [10] Y.-T. Kao, E. Zahara, and I.-W. Kao, "A hybridized approach to data clustering," *Expert Systems with Applications*, vol. 34, no. 3, pp. 1754–1762, 2008.
- [11] K. Krishna and M. N. Murty, "Genetic K-means algorithm," *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, vol. 29, no. 3, pp. 433–439, 1999.
- [12] K. R. Žalik, "An efficient k/-means clustering algorithm," *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1385–1391, 2008.
- [13] U. Maulik and S. Bandyopadhyay, "Genetic algorithm-based clustering technique," *Pattern Recognition*, vol. 33, no. 9, pp. 1455–1465, 2000.
- [14] M. Laszlo and S. Mukherjee, "A genetic algorithm that exchanges neighboring centers for k-means clustering," *Pattern Recognition Letters*, vol. 28, no. 16, pp. 2359–2366, 2007.
- [15] M. Fathian and B. Amiri, "A honeybee-mating approach for cluster analysis," *The International Journal of Advanced Manufacturing Technology*, vol. 38, no. 7-8, pp. 809–821, 2008.
- [16] A. Afshar, O. Bozorg Haddad, M. A. Mariño, and B. J. Adams, "Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation," *Journal of the Franklin Institute*, vol. 344, no. 5, pp. 452–462, 2007.
- [17] M. Fathian, B. Amiri, and A. Maroosi, "Application of honey-bee mating optimization algorithm on clustering," *Applied Mathematics and Computation*, vol. 190, no. 2, pp. 1502–1513, 2007.
- [18] P. S. Shelokar, V. K. Jayaraman, and B. D. Kulkarni, "An ant colony approach for clustering," *Analytica Chimica Acta*, vol. 509, no. 2, pp. 187–195, 2004.
- [19] T. Niknam, B. B. Firouzi, and M. Nayeripour, "An efficient hybrid evolutionary algorithm for cluster analysis," *World Applied Sciences Journal*, vol. 4, no. 2, pp. 300–307, 2008.
- [20] M. K. Ng and J. C. Wong, "Clustering categorical data sets using tabu search techniques," *Pattern Recognition*, vol. 35, no. 12, pp. 2783–2790, 2002.
- [21] C. S. Sung and H. W. Jin, "A tabu-search-based heuristic for clustering," *Pattern Recognition*, vol. 33, no. 5, pp. 849–858, 2000.
- [22] T. Niknam, B. Amiri, J. Olamaei, and A. Arefi, "An efficient hybrid evolutionary optimization algorithm based on PSO and SA for clustering," *Journal of Zhejiang University SCIENCE A*, vol. 10, no. 4, pp. 512–519, 2009.
- [23] T. Niknam, "An efficient hybrid evolutionary algorithm based on PSO and HBMO algorithms for multi-objective Distribution Feeder Reconfiguration," *Energy Conversion and Management*, vol. 50, no. 8, pp. 2074–2082, 2009.
- [24] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 687–697, 2008.
- [25] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Tech. Rep. tr06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
- [26] W. Zou, Y. Zhu, H. Chen, and X. Sui, "A clustering approach using cooperative artificial bee colony algorithm," *Discrete Dynamics in Nature and Society*, vol. 2010, Article ID 459796, 16 pages, 2010.
- [27] H. Narasimhan, "Parallel artificial bee colony (PABC) algorithm," in *Proceedings of the World Congress on Nature & Biologically Inspired Computing (NABIC '09)*, pp. 306–311, IEEE, Coimbatore, India, December 2009.
- [28] D. Teodorović, "Bee colony optimization (BCO)," in *Innovations in Swarm Intelligence*, C. Lim, L. Jain, and S. Dehuri, Eds., vol. 248, pp. 39–60, Springer, Berlin, Germany, 2009.
- [29] D. Teodorovic, P. Lucic, G. Markovic, and M. Dell' Orco, "Bee colony optimization: principles and applications," in *Proceedings of the 8th Seminar on Neural Network Applications in Electrical Engineering (NEUREL '06)*, pp. 151–156, 2006.
- [30] D. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, "The bees algorithm—a novel tool for complex optimisation problems," in *Proceedings of the 2nd Virtual International Conference on Intelligent Production Machines and Systems (IPROMS '06)*, pp. 454–459, 2006.

- [31] R. Akbari, A. Mohammadi, and K. Ziarati, "A novel bee swarm optimization algorithm for numerical function optimization," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 10, pp. 3142–3155, 2010.
- [32] H. Drias, S. Sadeg, and S. Yahi, "Cooperative bees swarm for solving the maximum weighted satisfiability problem," in *Computational Intelligence and Bioinspired Systems*, J. Cabestany, A. Prieto, and F. Sandoval, Eds., vol. 3512 of *Lecture Notes in Computer Science*, pp. 318–325, Springer, Berlin, Germany, 2005.
- [33] C. Yang, J. Chen, and X. Tu, "Algorithm of fast marriage in honey bees optimization and convergence analysis," in *Proceedings of the IEEE International Conference on Automation and Logistics (ICAL '07)*, pp. 1794–1799, Jinan, China, August 2007.
- [34] M. T. Vakil-Baghmisheh and M. Salim, "A modified fast marriage in honey bee optimization algorithm," in *Proceedings of the 5th International Symposium on Telecommunications (IST '10)*, pp. 950–955, December 2010.
- [35] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [36] R. Storn and K. Price, *Differential Evolution—A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces*, ICSI, Berkeley, Calif, USA, 1995.
- [37] V. Feoktistov, "Differential evolution," in *Differential Evolution*, vol. 5, pp. 1–24, Springer, New York, NY, USA, 2006.
- [38] K. V. Price, R. M. Storn, and J. A. Lampinen, "The differential evolution algorithm," in *Differential Evolution*, pp. 37–134, Springer, Berlin, Germany, 2005.
- [39] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.
- [40] R. J. Kuo, E. Suryani, and A. Yasid, "Automatic clustering combining differential evolution algorithm and k-means algorithm," in *Proceedings of the Institute of Industrial Engineers Asian Conference 2013*, Y.-K. Lin, Y.-C. Tsao, and S.-W. Lin, Eds., pp. 1207–1215, Springer, Singapore, 2013.
- [41] W. Kwedlo, "A clustering method combining differential evolution with the K-means algorithm," *Pattern Recognition Letters*, vol. 32, no. 12, pp. 1613–1621, 2011.
- [42] Y.-J. Wang, J.-S. Zhang, and G.-Y. Zhang, "A dynamic clustering based differential evolution algorithm for global optimization," *European Journal of Operational Research*, vol. 183, no. 1, pp. 56–73, 2007.
- [43] M. Babrdelbonab, S. Z. M. H. M. Hashim, and N. E. N. Bazin, "Data analysis by combining the modified k-means and imperialist competitive algorithm," *Jurnal Teknologi*, vol. 70, no. 5, 2014.
- [44] P. Berkhin, "A survey of clustering data mining techniques," in *Grouping Multidimensional Data*, J. Kogan, C. Nicholas, and M. Teboulle, Eds., pp. 25–71, Springer, Berlin, Germany, 2006.
- [45] J. R. Riley, U. Greggers, A. D. Smith, D. R. Reynolds, and R. Menzel, "The flight paths of honeybees recruited by the waggle dance," *Nature*, vol. 435, no. 7039, pp. 205–207, 2005.
- [46] C. Grüter, M. S. Balbuena, and W. M. Farina, "Informational conflicts created by the waggle dance," *Proceedings of the Royal Society B: Biological Sciences*, vol. 275, no. 1640, pp. 1321–1327, 2008.
- [47] A. Dornhaus and L. Chittka, "Why do honey bees dance?" *Behavioral Ecology and Sociobiology*, vol. 55, no. 4, pp. 395–401, 2004.
- [48] K. O. Jones and A. Bouffet, "Comparison of bees algorithm, ant colony optimisation and particle swarm optimisation for PID controller tuning," in *Proceedings of the 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing (CompSysTech '08)*, Gabrovo, Bulgaria, June 2008.
- [49] D. T. Pham and M. Kalyoncu, "Optimisation of a fuzzy logic controller for a flexible single-link robot arm using the Bees Algorithm," in *Proceedings of the 7th IEEE International Conference on Industrial Informatics (INDIN '09)*, pp. 475–480, Cardiff, Wales, June 2009.
- [50] D. T. Pham, S. Otri, A. Ghanbarzadeh, and E. Koc, "Application of the bees algorithm to the training of learning vector quantisation networks for control chart pattern recognition," in *Proceedings of the 2nd Information and Communication Technologies (ICTTA '06)*, vol. 1, pp. 1624–1629, Damascus, Syria, 2006.
- [51] L. Özbakir, A. Baykasoğlu, and P. Tapkan, "Bees algorithm for generalized assignment problem," *Applied Mathematics and Computation*, vol. 215, no. 11, pp. 3782–3795, 2010.
- [52] P. Rocca, G. Oliveri, and A. Massa, "Differential evolution as applied to electromagnetics," *IEEE Antennas and Propagation Magazine*, vol. 53, no. 1, pp. 38–49, 2011.
- [53] R. Mallipeddi, P. N. Suganthan, Q. K. Pan, and M. F. Tasgetiren, "Differential evolution algorithm with ensemble of parameters and mutation strategies," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 1679–1696, 2011.
- [54] R. Storn, "On the usage of differential evolution for function optimization," in *Proceedings of the Biennial Conference of the North American Fuzzy Information Processing Society (NAFIPS '96)*, pp. 519–523, June 1996.
- [55] U. K. Chakraborty, *Advances in Differential Evolution*, Springer, Berlin, Germany, 2008.
- [56] G. Liu, Y. Li, X. Nie, and H. Zheng, "A novel clustering-based differential evolution with 2 multi-parent crossovers for global optimization," *Applied Soft Computing Journal*, vol. 12, no. 2, pp. 663–681, 2012.
- [57] Z. Cai, W. Gong, C. X. Ling, and H. Zhang, "A clustering-based differential evolution for global optimization," *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 1363–1379, 2011.
- [58] M. Abbasgholipour, M. Omid, A. Keyhani, and S. S. Mohtasebi, "Color image segmentation with genetic algorithm in a raisin sorting system based on machine vision in variable conditions," *Expert Systems with Applications*, vol. 38, no. 4, pp. 3671–3678, 2011.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

