Journal of AI and Data Mining
Vol 5, No 2, 2017, 293-305

Improved COA with Chaotic Initialization and Intelligent Migration for Data Clustering

M. Lashkari[1] and M.- H. Moattar[2*]

1. Department of Computer Engineering, Ferdows Branch, Islamic Azad University, Ferdows, Iran.
2. Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran.

Received 23 November 2015; Revised 27 May 2016; Accepted 04 October 2016
**Corresponding author: moattar@mshdiau.ac.ir (M. H. Moattar).*

Abstract
K-means algorithm is a well-known clustering algorithm. In spite of its advantages such as high speed and ease of employment, this algorithm suffers from the problem of local optima. In order to overcome this problem, a lot of works have been carried out on clustering. This paper presents a hybrid extended cuckoo optimization algorithm (ECOA) and K-means (K) algorithm called ECOA-K. The COA algorithm has advantages such as fast convergence rate, intelligent operators, and a simultaneous local and global search work, which are the motivations behind choosing this algorithm. In ECOA, we have enhanced the operators in the classical version of the cuckoo algorithm. The proposed operator for production of the initial population is based upon a chaos sequence, whereas in the classical version, it is based upon a randomized series. Moreover, allocating the number of eggs to each cuckoo in the revised algorithm is done based on its fitness. Another improvement is in the cuckoos' migration, which is performed with different deviation degrees. The proposed method is evaluated on several standard datasets at the UCI database, and its performance is compared with those of black hole (BH), big bang big crunch (BBBC), cuckoo search algorithm (CSA), traditional cuckoo optimization algorithm (COA), and K-means algorithm. The results obtained are compared in terms of the purity degree, coefficient of variance, convergence rate, and time complexity. The simulation results show that the proposed algorithm is capable of yielding the optimized solution with a higher purity degree, faster convergence rate, and stability, in comparison with the other algorithms.

Keywords: *Clustering, K-means Algorithm, Cuckoo Optimization Algorithm (COA), Chaotic Function, Migration*

1. Introduction

Data clustering is one of the most important and popular data analysis techniques that refers to the process of grouping a set of data objects into clusters, in which within cluster similarity and between cluster divergence will be satisfied [1, 2, 3, 4, 5, 6]. Clustering is intrinsically a multi-dimensional high-complexity optimization problem with a deterministic objective that is to group related patterns to the same cluster. Since clustering is an unsupervised learning method, it has been used in many areas such as engineering, medical, and social sciences.

One of the widely-used clustering algorithms is the K-means algorithm [6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, and 18], which has been proposed by Macqueen in 1967 [3]. After four decades, this algorithm has remained a popular clustering technique. In spite of its advantages such as ease of implementation, high speed, and scalability for huge databases, it suffers from some weaknesses such as dependency on the initial centers. Improper selection of initial centroids may result in local optima. There have been different strategies suggested in the recent decades to improve the K-means algorithm. Most of them have been inspired by evolutionary algorithms that conduct a global and randomized search work around the problem space so that they achieve an optimal solution. For example, Nanda and Panda (2014) [19] have reviewed a number of major

nature-inspired metaheuristic algorithms in order to solve this problem.

2. Related works

The literature includes numerous works proposing metaheuristic algorithms for improving clustering outputs. For example, Alam et al. (2014) [20] have reviewed different combinations of the PSO algorithm for clustering improvement. Ultimately, in all of these strategies, attempts have been made to use evolutionary algorithms independently or in combination with K-means algorithm and benefits from the advantages of the two algorithms, and have, therefore, moved the results outside the locally optimal trap to a great extent [4, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, and 21]. Mualik and Bandyopadhyay (2000) [6] have proposed a genetic algorithm-based method. They have proposed a mutation operator specific to clustering called distance-based mutation. Sung and Jin (2000) have proposed an approach based on the tabu search (TS) for cluster analysis [7]. Shelokar et al. (2004) [8] have proposed an approach based on ant colony optimization (ACO). Fathian and Amiri (2007) [9] have proposed the honey bees mating optimization (HBMO) algorithm to solve the clustering problem. Laszlo and Mukherjee (2007) have proposed a genetic algorithm that exchanges neighboring centers for K-means clustering [10]. Niknam et al. (2008) [11] have presented a hybrid evolutionary optimization algorithm based on a combination of ACO and simulated annealing (SA) to solve the clustering problem.

Niknam et al. (2009) [12] have presented a hybrid evolutionary algorithm based on particle swarm optimization (PSO) and SA to find the optimal cluster centers. Rana and Jasola (2010) [13] have presented a hybrid evolutionary optimization algorithm based on a combination of PSO and K-means to solve the clustering problem. Firouzi et al. (2010) [14] have introduced a hybrid evolutionary algorithm based on combining PSO, SA, and K-means to find an optimal solution. Niknam and Amiri (2010) have proposed a hybrid algorithm based on a fuzzy adaptive PSO, ACO, and K-means for cluster analysis [15]. Niknam et al. (2011) [4] have proposed a hybrid algorithm based on imperialist competitive algorithm (ICA) and K-means for cluster analysis. Hatamlou et al. (2011) [16] have proposed a new optimization method that is based upon one of the theories of the evolution of the universe, namely the big bang and big crunch theory (BBBC) for cluster analysis. Hatamlou (2013) [17] has proposed a new heuristic optimization approach for data clustering that is inspired by the black hole (BH) phenomenon. Manikandan and Selvarajan (2014) [18] have presented a new algorithm based on the cuckoo search algorithm (CSA) to solve the clustering problem. Hatamlou et al. (2012) [21] have presented a hybrid data clustering algorithm based on the gravitational search algorithm (GSA) and *K*-means algorithm (GSA-KM), which uses the advantages of both algorithms and helps the *k*-means algorithm to escape from local optima and also increases the convergence speed of the GSA algorithm.

Some other applications of optimization algorithms include [29], which has proposed a new hybrid optimization algorithm based on the gravitational search algorithm and Nelder-Mead algorithm to improve crash performance of vehicles during frontal impact. Ref. [30] has proposed a new hybrid optimization approach based on the PSO algorithm and the receptor editing property of immune system. The aim of this work was to develop an approach in the design and manufacturing areas. Differential evolution algorithm is proposed to solve optimization problems in the manufacturing industry [32]. Ref. [34] has presented a comparison on the evolutionary optimization techniques for the structural design problems, and proposes a hybrid optimization technique based on the differential evolution algorithm to solve these problems. Also [35] has proposed a hybrid technique based on differential evolution for solving manufacturing optimization problems.

In [36], a particle swarm-based optimization approach has been presented for multi-objective optimization of vehicle crash worthiness, so the optimized structure can absorb the crash energy by controlled vehicle deformations, while maintaining enough space of the passenger compartment. The approach proposed in [37] is based upon an improved genetic algorithm, used to solve the multi-objective shape design optimization problems. The purpose of [38] has been to develop a novel hybrid optimization method (HRABC) based on the artificial bee colony algorithm and the Taguchi method. This approach is applied to a structural design optimization of a vehicle component and a multi-tool milling optimization problem. Also [39] has presented an optimization approach based on the artificial bee colony algorithm for optimal selection of cutting parameters in multi-pass turning operations.

Overall, the evolutionary algorithms introduced so far can be divided into two groups. The first is being those capable of global search such as GA,

ACO, and PSO versus those with a local search capability such as TS and SA. In the first group, the probability of the results getting trapped in the local optimum is lower than in the second group. However, due to no local search works, the final solutions in this group are less precise. In the second group, due to the lack of a global search around the problem, the probability of the results getting trapped in the local optimum is higher. However, due to the local search works, more attempts should be made to enhance the precision of the final solutions. Usually, in order to resolve the above-mentioned weaknesses, researchers hybridize these two approaches, which would enhance the precision of the final solutions but the complexity of the computational processes emerges.

Cuckoo optimization algorithm (COA) is a novel approach introduced, for the first time, in 2011 by Rajabioun (2011) [31] in order to solve a vast majority of optimization problems. In this algorithm, which is inspired by a cuckoo's life, there are certain operations that are capable of both local and global search works around the problem simultaneously. There are also certain operations contrived in case of emergence of the local optimum. Therefore, this algorithm is capable of achieving highly precise solutions with high rates of convergence. This algorithm, however, has its own weaknesses that we tried to overcome in an enhanced version, namely extended cuckoo optimization algorithm (ECOA). To do this, we optimized some of the traditional operators in a systematic way. The remainder of this paper is organized as what follows. In Section 3, the classical COA is introduced. In Section 4, the cluster analysis problem is discussed. Sections 5 and 6 introduce the proposed extended COA and hybrid ECOA-K algorithms, respectively. Sections 7 and 8 introduce the experimental setup and evaluations of the proposed approach, and comparisons are made with the BH, BB-BC, CSA, COA, and K-means approaches for different datasets. Finally, Section 9 includes the conclusion.

3. COA

COA is inspired by the life of a bird family, called cuckoo. The special lifestyle of these birds and their characteristics in egg-laying and breeding has been the basic motivation for development of this new optimization algorithm. Similar to the other evolutionary methods, COA starts with an initial population. The cuckoo population is of two types: mature cuckoos and eggs. The effort to survive among cuckoos constitutes the basis of COA. During the survival competition, some of cuckoos or their eggs demise. The survived cuckoo societies immigrate to a better environment and start reproducing and egg laying. Cuckoos' survival effort hopefully converges to a state that there is only one cuckoo society, all with the same fitness values. The COA algorithm is composed of the following steps [31]:

1- Initialize cuckoo habitats using some random points.
2- Dedicate some eggs to each cuckoo.
3- Define egg-laying radius (ELR) for each cuckoo based on the following formula:

$$ELR = \beta * \frac{No.\ of\ current\ eggs}{Total\ No.\ of\ eggs} * (\mathrm{var}_{hi} - \mathrm{var}_{low}) \quad (1)$$

In this equation, var_{hi} and var_{low} are the higher and lower limits of the search space, respectively, and β is an integer supposed to handle the maximum value for ELR.
4- Let cuckoos lay eggs inside their corresponding ELR.
5- Kill those eggs that are recognized by host birds. After that, all cuckoos' eggs are laid in host birds' nests, and some of them that are less similar to the host birds' eggs are detected by the host birds and thrown out of the nest. Thus after the egg-laying process, *P*% of all eggs (usually 10%) with less fitness values will be killed.
6- Let eggs hatch and chicks grow.
7- Evaluate the habitat of each newly-grown cuckoo.
8- Limit the cuckoos' maximum number in the environment and kill those in the worst habitats.
9- Cuckoos are clustered, and select a goal habitat.
10- New cuckoo population immigrates toward the goal habitat.
11- If the termination condition is not satisfied, go to 2.

In what follows, we explain the advantages of COA and the reasons behind selecting it as the fundamental clustering algorithm:
1- Fast convergence rate**:** The convergence rate of this algorithm is faster compared to the other optimization algorithms, and it is able to reach the optimum solution in less iterations [31, 32].
2- Simultaneous local and global search**:** In this algorithm, unlike the other optimization algorithms, both local and global searches are inherited in the algorithm nature. This improves the precision of the algorithm [33].
3- Intelligent operators**:** In this algorithm, there are intelligent operators as compared with the other

algorithms. For instance, the egg-laying procedure is a local search operator that helps exiting the local optimum [34].

4- Variable population size**:** This helps to destruct the population in poor areas and provide less fitness calculations [31].

There are numerous works in the literature that use COA and its variants for different optimization problems. For example, in [35], the cuckoo search (CS) algorithm has been introduced for solving the manufacturing optimization problems. This research work is the first application of the CS algorithm to the optimization of machining parameters. In [36], the CS algorithm has been proposed for solving structural design optimization problems. Also [37] shows the effectiveness of gravitational search algorithm (GSA) and charged system search algorithm (CSS) for the optimum design of a vehicle component.

4. Cluster analysis problem

K-means algorithm is one of the simplest unsupervised learning algorithms. The procedure follows a simple and easy way to classify a given dataset through a certain number of clusters (assume *K* clusters) fixed as a priori [4]. The resulting clusters will have high intra-similarity and inter-variability. Basically, to evaluate the similarity between the data objects, the distance measure is used. Particularly, the problem is specified as follows: given *N* objects, assign each object to one of the *K* clusters and minimize the sum of the squared Euclidean distances between each object and the center of the clusters:

$$F(O,Z) = \sum_{i=1}^{N}\sum_{j=1}^{K} W_{ij}\left\|(O_i - Z_j)\right\|^2 \qquad (2)$$

where, $||O_i\text{-}Z_j||$ is the Euclidean distance between a data object O_i and the cluster center Z_j. *N* and *K* are the number of data objects and the number of clusters, respectively. w_{ij} is the association weight of data object O_i with cluster *j*, which will be either 1 or 0 (if object *i* is assigned to cluster *j*; w_{ij} is 1, otherwise 0) [17].

ence, the fitness function for measuring the goodness of a clustering solution is based on the following formula:

$$Fitness(h) = \sum_{m=1}^{K}\sum_{z=1}^{N}\left\|x_z - h_m\right\| \qquad (3)$$

where, $hi = \{h_1, h_2 \ldots h_k\}$ denotes the cluster centers, and the i^{th} solution has *k* cluster centers. h_m indicates the center of the m^{th} cluster in the i^{th} solution. Furthermore, $x_z=\{x_1, x_2, \ldots, x_N\}$ and *N* are the data points and the total number of data points in the m^{th} cluster, respectively. The desired solution is reached when the above fitness function becomes minimal.

5. Extended COA

COA, in its primary version, suffers from certain deficiencies. In our proposed extended algorithm, we have improved and systematized a number of them. In the extended cuckoo algorithm, we intend to enhance the convergence rate, stability, and purity degree in comparison with the classical version. In what follows, we will discuss further the steps involved in the ECOA algorithm.

5.1. Producing initial population based on chaotic sequence

The traditional version of the cuckoo algorithm uses random sequences to produce an initial population. The randomized parameters of COA might influence the algorithm efficiency. It might not be able to cover a global search, and therefore, the convergence rate may be reduced. In the suggested extended cuckoo algorithm, the chaotic numbers are used instead of the random sequences in order to improve the searching of the cuckoos. As a result, the population produced would be semi-randomized. A search supported by chaotic mappings has the possibility of access to most states in a certain zone and without any iteration. Through an extended population positioning via this process, most of the searching space would be explored. Then the results obtained would have the required distribution within the searching domain, which, in turn, would contribute to find a more efficient optimum. Then a number of members are found in the population that are either optimal by themselves or are distributed within a short distance of the optimal solution and would be selected as the best in the next round. It would also provide a possibility of escaping the locally-optimal points in which the algorithm might get entangled. This way, the convergence rate of the algorithm is raised. The chaotic mapping that was selected to produce chaos sequences in this work was a logistic map. This mapping is defined in (4).

$$Cr_{n+1} = \delta * Cr_n * (1 - Cr_n) \ \ for\ 0 < \delta \le 4 \qquad (4)$$

Here, δ is the initial value of the function. Equation 5 indicates the formula for producing the initial population based on a randomized sequence in classic COA algorithm:

$$(VAR_{hi} - VAR_{low}) * \text{rand} + \text{VAR}_{low} \qquad (5)$$

On the other hand, (6) indicates the production of the initial population based on the proposed chaotic approach in ECOA:

$$(VAR_{hi} - VAR_{low}) * \mathrm{Cr} + \mathrm{VAR}_{low} \quad (6)$$

In this equation, *Cr* represents a function based on the behavior of the logistic map varying between 0 and 1. Therefore, using the chaos sequence in the production of the initial population, we expect diversity of that population. An increase in the algorithm's convergence rate would lead to the final optimized solution. The primary formula for producing the initial population as in (5) causes the population to densely concentrate on some regions and have less distribution and diversity in the initial population. However, using the proposed approach as in (6), due to the ability of chaotic sequences to generate longer random sequences, the diversity of the solutions and their coverage is improved.

5.2. Systematic egg laying

In the primary version of the cuckoo algorithm, the number of eggs and the egg-laying radius for each bird are decided through randomization. Using a randomized sequence for estimating the number of eggs would decrease the convergence rate of the algorithm. That is due to the fact that some cuckoos in a better state in the problem space might be given fewer eggs, and vice versa. Therefore, an inappropriate state in the problem space will be analyzed more than an appropriate state in the space. This would, in turn, degrade the convergence rate of the primary cuckoo algorithm. In the suggested method, the number of eggs allocated to each bird depends on the bird's fitness. This variable can be estimated using (7).

$$Egg_i = \min_{egg} + round(fitness_i - \min_{fit}) + \frac{\max_{egg} - \min_{egg}}{\max_{fit} - \min_{fit}} \quad (7)$$

In which Egg_i is the number of allocated eggs to the i^{th} cuckoo, min_{egg} represents the minimum number of eggs, $fitness_i$ is the i^{th} cuckoo fitness, and min_{fit} is the minimal value of the cuckoo fitness function, whereas max_{egg} is the maximum number of eggs, and max_{fit} represents the maximal value for the fitness function. This would help to allocate an appropriate number of eggs systematically to each cuckoo. This formula assigns the number of eggs to each cuckoo based on a definite approach in spite of a random assignment. In this formula, when the fitness of the *i*th solution ($fitness_i$) is higher, more eggs are assigned to the cuckoo in that region. To control the number of eggs, the fitness is normalized, considering the minimum and maximum fitness of the solutions in the current population.

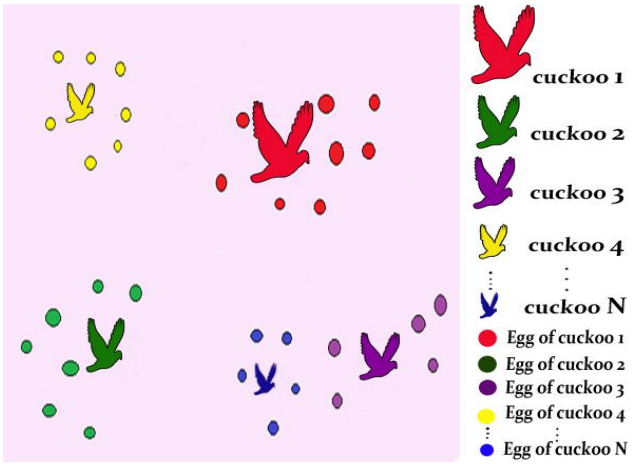


Figure 1. Results of egg-laying based on randomized policy in classical version of cuckoo algorithm.

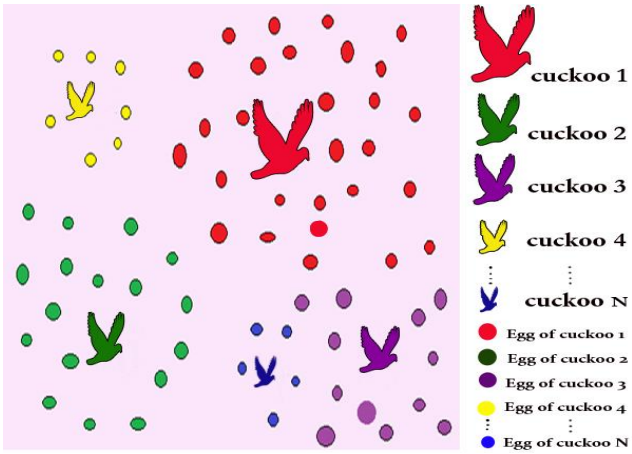


Figure 2. Results of estimating number of eggs for each cuckoo based on its fitness in extended cuckoo algorithm.

Figure 1 indicates the result of estimating the number of eggs for each cuckoo randomly as in the traditional version of the cuckoo algorithm, while figure 2 represents the result of allocating eggs to each cuckoo based on an intelligent policy in the proposed version of the cuckoo algorithm. In these figures, the cuckoos of bigger sizes are those in better positions in the problem space, while those of a smaller size belong to less proper positions.

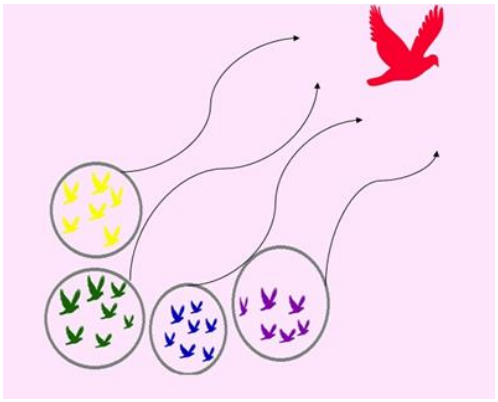
As it can be observed in figure 1, due to the randomized policy of egg allocation to cuckoos, fewer eggs might be allocated to birds at a better position, and vice versa. This would slow down the convergence rate and purity degree of the algorithm in achieving the final optimal solution. This problem has been met in the extended algorithm with the help of an intelligent egg allocation.

5.3. Systematic migration

In the original version of the cuckoo algorithm, after the birds' egg-laying in the space and the destruction of eggs with less fitness, the remaining cuckoos are clustered. Then the fitness of each cluster is estimated and the one with the highest fitness is selected as the superior region in which the most fitted cuckoo is identified and then the global optimal cuckoo is updated. The problem with this method is that once clustered, the number of cuckoos would differ across the clusters. Therefore, comparing the fitness in clusters with different numbers of cuckoos is a misleading attempt, and might lead to an improper optimal cuckoo in a cluster and a wrong updating of the global optimal cuckoo. This would be followed by a wrong migration and deviation of cuckoos in space as well as a decreased convergence rate of the algorithm.

For the same reason, in the extended cuckoo algorithm, to make up for this deficiency, before the clustering step, the best cuckoo in the present generation would be specified, and then the globally optimal cuckoo is updated. Once the clustering is done, diverse groups of cuckoos are produced that are ready for migration. In order to provide a wider global coverage in this problem, we would let different groups of cuckoos migrate towards the globally optimal cuckoo in the space with a different degree of deviation. In other words, only one group migrates towards the optimal point with a low deviation. They are to search for more optimal points of higher fitness in that region. The other groups follow different degrees of deviation in searching the space. In fact, the migration of all groups of cuckoos in the primary algorithm towards a certain point would create a high density of cuckoos in a particular region. This would provide a lower coverage of the problem space. In case the globally optimal cuckoo is better than the currently existing solutions, it is not identified. Therefore, it would have less chance of achieving the real globally optimal point through the classical cuckoo algorithm. Figure 3 illustrates migration in the primary version of the cuckoo algorithm; the manner of migration in the extended algorithm is indicated in figure 4.

As it can be seen in part (a) of figure 3, all the birds would follow the same degree of deviation towards the globally optimal cuckoo, which is indicated in part (b) of figure 3. If there exists a better globally optimal cuckoo than the current one in the problem at hand, the chance of finding it is reduced since all the cuckoos would be searching the same domain. Equation 8 indicates the migration function in the original COA algorithm.



a) Before migration.

b) After migration.

Figure 3. Migration in classical cuckoo algorithm.

$$X_{Nextij}(t+1) = X_{Currentij}(t) + F * (X_{Goal} - X_{Currentij(t)}) \; for \; \forall i, \mathrm{j} \tag{8}$$

In (8), F is the degree of deviation during migration that is constant for all clusters and different iterations, and $X_{Nextij}(t)$ and $X_{Nextij}(t+1)$ are the locations of the j^{th} cuckoos in the i^{th} cluster at iterations t and $t+1$, respectively. X_{Goal} is the location of the globally optimal cuckoo in the search space, and N represents the total number of cuckoo in the i^{th} cluster. As it can be seen in figure 4 part (a), in the extended algorithm, each group of cuckoos follows a different degree of deviation towards the globally optimal cuckoo. This would disperse them in the space. Even if there exists a globally optimal cuckoo better than the current optimal position, the chance for finding it increases. Equation 9 indicates the migration in the proposed extended COA algorithm.

$$X_{Nextij}(t+1) = X_{Currentij}(t) + F_h * (X_{Goal} - X_{Currentij(t)}) \; for \; \forall i, \mathrm{j}, \mathrm{h} \tag{9}$$

In (9), F_h serves as the degree of deviation during their migration, and is different for any clusters. Yet in another improvement, we considered β of egg-laying radius and F adaptive to the iteration. In other words, when the algorithm approaches its end, these variables start reducing. A step-by-step

reduction in the egg-laying radius and cuckoos' degree of deviation during migration improves the searching process. In other words, it increases the exploration rate in the initial iterations of our algorithm, and would decrease the exploitation rate. The closer we get to the final iterations, due to approaching the optimal solution, the exploration rate is reduced, and the exploitation rate is increased.

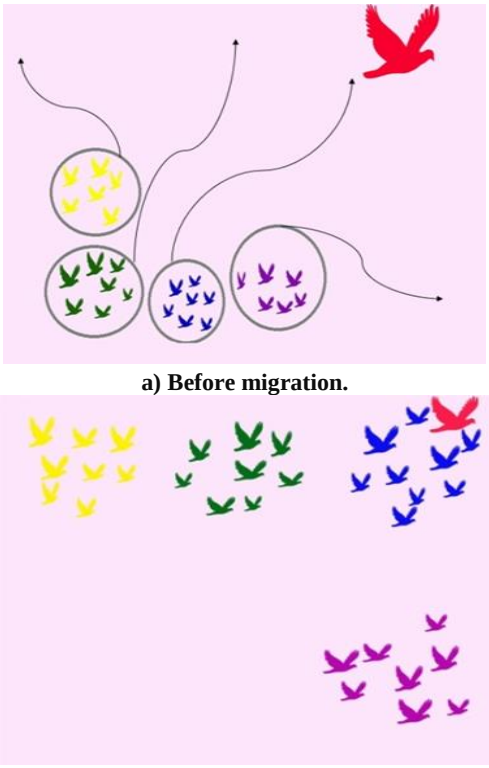
6. Hybrid ECOA-K algorithms

The randomized selection of the initial cluster centers in the K-means algorithm occasionally causes the clustering results to be located within the local optimum. In order to solve this deficiency, we use a hybrid of ECOA and K-means algorithms for clustering. In this hybrid algorithm, first, all the initial optimal centers are produced by ECOA, and the data points are clustered through the K-means algorithm. This algorithm is named ECOA-K in the paper, and is described in figure 5. Figure 6 shows the pseudo-code for the hybrid ECOA-K algorithm.

7. Experimental setup

To validate our method, three datasets, named iris, contraceptive method choice (CMC), and wine are used, which are available in the repository of the machine-learning databases (UCI) [38]. These datasets are used in evaluations to have the best correspondence with the previous works. On the other hand, these datasets have different dimensions (different number of records and different number of features). Thus we can study the generalizability and scalability of the approach for small-scale to large-scale problems. Table 1 summarizes the main characteristics of these datasets.

- **Iris:** This dataset has been collected by Anderson (1935). It contains three classes of 50 objects each, where each class refers to a type of iris flower. There are 150 random samples of iris flowers with four numeric attributes in this dataset. These attributes are sepal length and width in cm, and petal length and width in cm. There is no missing value for attributes.
- **CMC:** Contraceptive method choice is denoted as CMC. This dataset is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey. The samples are married women who either were not pregnant or did not know if they were at the time of interview. The problem is to predict the choice of the current contraceptive method (no use has 629 objects, long-term methods have 334 objects, and short-term methods have 510 objects) of a woman based on her demographic and socioeconomic characteristics [39].



a) Before migration.

b) After migration.

Figure 4. Migration in extended cuckoo algorithm.

- **Wine:** The wine dataset consists of 178 objects characterized by 13 features: alcohol, malic acid, ash content, alcalinity of ash, concentration of magnesium, total phenols, flavonoids, non-flavanoid phenols, proanthocyanins, color intensity, hue, and OD280/OD315 of diluted wines and pralines. The results were obtained by the chemical analysis of wines produced in the same region of Italy but derived from three different cultivars [40].

Table 1. Main characteristics of validation datasets.

Dataset	Number of clusters	Number of features	Number of data objects
Iris	3	4	150
Wine	3	13	178
CMC	3	9	14783

The performance of the ECOA-K algorithm was compared against the well-known and most recent algorithms reported in the literature including K-means [3], big bang-big crunch (BB-BC) [16], Black hole (BH) [17], cuckoo search algorithm (CSA) [18], and cuckoo optimization algorithm (COA) [31].

As mentioned in the literature, the BB-BC algorithm has advantages such as a simple

structure and an easy implementation [16], although this algorithm has disadvantages such as a relatively low accuracy rate, low stability, and low convergence rate in some cases. The BH algorithm has similar Cons and Pros [17]. The CSA algorithm has a simple structure but a low accuracy rate, a low stability, and a low convergence rate [41]. Finally, the COA algorithm has a fast convergence rate and a high accuracy but a high computation cost and a low stability. The performance of the algorithms is evaluated and compared using the following criteria.

▪ **Purity index:** This index examines the purity degree of the clustering algorithm, and can be estimated using (10).

$$Purity = \sum_{r=1}^{k} \frac{n_r}{n} p(S_r) \tag{10}$$

In this equation, k indicates the number of clusters, and $p(S_r)$ represents the purity degree of cluster r, which can be estimated through (11). This equation would take into account the highest distribution of the samples for a given cluster.

$$p(S_r) = \frac{1}{n_r} \max_i (n_r^i) \tag{11}$$

In this equation, n_r refers to the number of samples in cluster r; r is the number of clusters, and n represents the total number of samples. The output would vary between 0 and 1. The closer it is to 1, the higher the purity index of data clustering [42].

▪ **Coefficient of variance (CV):** In the probability theory and statistics, CV is a normal criterion used to measure the distribution of statistical data. It is obtained by driving the standard deviation by the mean as in (12).

$$C_v = \frac{\lambda}{\mu} \tag{12}$$

An algorithm whose CV measure is lower after several iterations would yield more stable and reliable results. In other words, it can help to provide the stability of responses to a great extent.

▪ **Convergence rate:** An algorithm that manages to gain the optimal solution with a higher purity degree in comparison with the other algorithms and with less iteration, and the estimation is said to be the most efficient one.

▪ **Time complexity:** It is defined as the time it takes for an algorithm to be conducted to gain an optimal solution. Since the actual time (in seconds) highly depends on the encoding and programming language, we considered the number of fitness function evaluations as the time complexity measure.

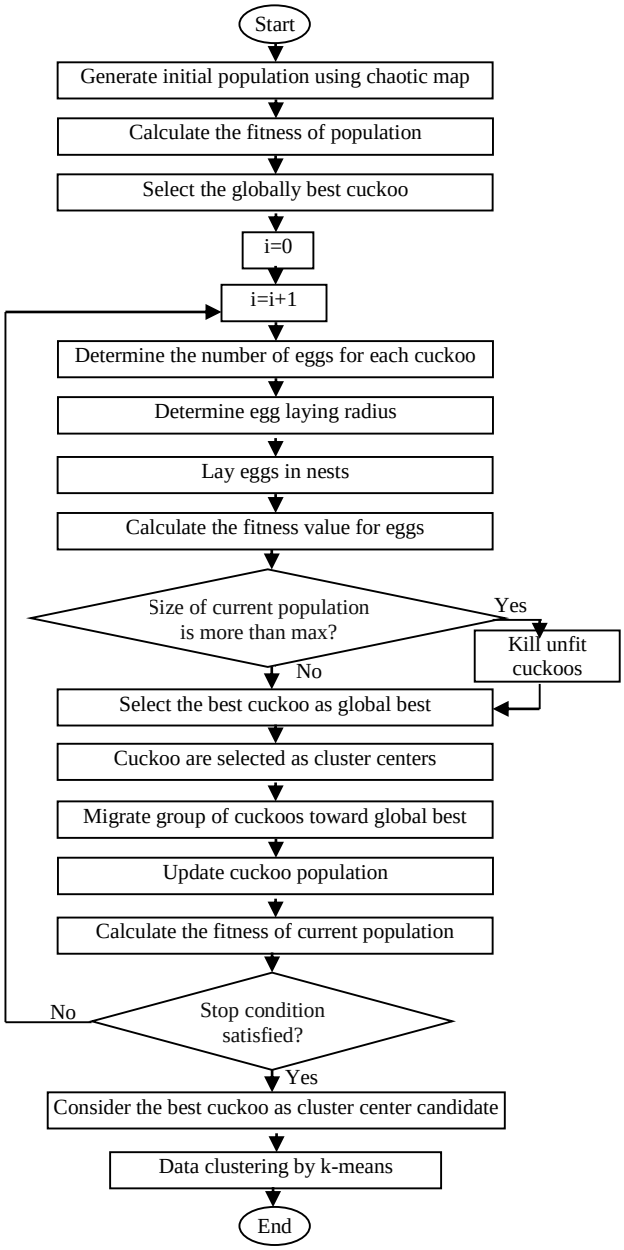


Figure 5. Flowchart of ECOA-K algorithm.

8. Experimental results

In this section, the proposed algorithm is evaluated for 3 different scenarios with purity index, CV index, convergence rate, and time complexity. Almost all control parameters are adaptive with the convergence of the algorithm, and they do not need specific tuning prior to running the algorithm. These performances are the results of 20 independent runs, and show the stability of the approach.

The values for the variables such as Var_{hi} and Var_{low} are selected based on the dimension scales of the datasets. Similar values for parameters for different datasets are not appropriate because

using smaller or bigger values for the parameters may lead to slower or faster movements in the search space that leads to a slower convergence and a more search time. These parameter values are determined experimentally, and the only weakness of the proposed approach is selecting the appropriate values form these parameters. However, this weakness is general for all the optimization algorithms. Other algorithms such as those evaluated in this work are also highly parameter-dependent.

```
Initialize:
Numcuckoos: The initialize of population cuckoos.
Maxcuckoos: The maximum number of cuckoo in the environment.
Minegg; Maxegg: The minimum and maximum number of eggs.
Maxiteration: The maximum number of iteration;
Cr (1, 1) =0. 80; a=4; the initial values for the chaos sequence.
K; the number of cluster;
Varlow; Varhigh; Fi;
Generate an initial population based on logistic chaotic map using (6).
Calculate the fitness function for the initial population using (3).
Select cuckoo with the best fitness as global best cuckoo.
Begin
While (number of Maxiteration, or the stop criterion is not met)
  For j=1: Numcuckoos
    Dedicate some egg to j^th cuckoo based on its fitness using (7).
  End
  For h=1: Numcuckoos
    Define ELR for h^th cuckoo using (1).
  End
  Let cuckoos to lay eggs inside their corresponding ELR
  Check eggs and delete duplicate eggs in a nest.
  For z=1: sumeggs
    Calculate the fitness function for z^th egg.
  End
  Sort the population based on their fitness (cuckoos and eggs).
  Check the size of population.
  If size of current population is more than Maxcuckoos
    limit cuckoos number and kill those who live in worst habitats
    Numcuckoos=Maxcuckoos;
  End if
  Find best cuckoo in population and update global best cuckoo.
  Cuckoos are clustered to k clusters.
  Let new population immigrate toward global best using (9).
  Update population of cuckoos.
  Calculate the fitness function for the current population.
End while
Select global best cuckoo as initial cluster center for k-means.
Assign objects to the group that has the closest centroid (run k-means).
```

Figure 6. Pseudo-code for hybrid ECOA-K algorithm.

8.1. Evaluations on iris dataset

The first experiment concerns the evaluations of the proposed algorithm on the iris dataset, which are depicted in table 2.

The simulation results given in table 2 show that our proposed algorithm is capable of achieving solutions of a higher purity degree in comparison with those of the other methods. The results obtained on the iris dataset show that the ECOA-K algorithm converges the global optimum by a purity degree equal to 0.8933, while purity in the K-means, CS, BH, BBBC, and COA algorithms are 0.8299, 0.8486, 0.8896, 0.8810, and 0.8733, respectively.

Table 2. Experimental results on iris data.

Algorithm	Purity degree	CV Index	Convergence	
			Number of iterations	Fitness function calculations
K-means	0.8299	0.4924	9.3	9.3
CSA	0.8486	0.1681	8.9	340
BH	0.8896	0.055	6.2	255
BBBC	0.8810	0.1446	14.65	832
COA	0.8733	0.2720	7.45	2134
ECOA-K	0.8933	0	4.6	1685

Therefore, due to the application of intelligent operators such as chaotic sequences, systematic egg-laying and intelligent migration, the proposed approach is more precise than the other compared approaches, which makes it appropriate for sensitive problems such as medical applications. Also CV for the proposed algorithm is zero, which is significantly less than the other methods. Therefore, this algorithm is capable of providing more stable results, as compared to the other algorithms. Its solutions are similar across iterations, and its fluctuation is minimal. The results obtained on the iris dataset show that the ECOA-K algorithm converges the global optimum by a CV index equal to 0, while this index in the K-means, CS, BH, BBBC and COA algorithms are 0.4924, 0.1681, 0.055, 0.1446, and 0.2720, respectively. Due to a better distribution of cuckoos in the problem space and its better convergence, the proposed approach results are more robust and trustable solutions, which means that the results are less different in different runs of the proposed approach.

The proposed algorithm is capable of converging the global optimum in 4.6 iterations, which is significantly less than other methods but the number of fitness function calculations is more than K-means, BH, BBBC, CS, and lower than COA. The results obtained on the iris dataset show that ECOA-K converges the global optimum by 1685 fitness function calculation, while the average number of calculations in K-means, CS, BH, BBBC, and COA are 9.3, 340, 255, 832, and 2134, respectively. As seen from the results obtained, the ECOA-K algorithm is far superior to the other algorithms and leads to a faster convergence than the other approaches including the original versions of COA.

In order to find the degree of significance of the results obtained by the clustering algorithms, the statistical analysis was carried. We employed the non-parametric Wilcoxon test to determine whether there were significant differences in the

results of the clustering algorithms. The purpose behind each statistical test was to see whether the research results had been induced as a result of the independent hypothesis or the mere effect of random factors. The test has two hypothesis called H0 and H1, which are defined as follow:

$$\begin{aligned} H0 &= method\ X_1\ is\ better\ than\ X_2 \\ &\quad E(X_1) \geq E(X_2) \\ H0 &= method\ X_2\ is\ better\ than\ X_1 \\ &\quad E(X_2) \geq E(X_1) \end{aligned} \tag{13}$$

If case H0 is rejected, we can conclude that the result obtained has not been due to random factors but due to the independent variable. In this method, we do always consider our claim or method as H1 and other methods as H0. Then with the help of a significance test, the hypotheses will be either accepted or rejected. In the following tests, we used α = 0.05 as the confidence level. A wider description of these tests has been presented in [43,44].

Table 3. Results obtained by statistical analysis of algorithms based on purity criteria on iris dataset.

	K-means *vs.* ECOA-K	**CS *vs.* ECOA-K**	**BH *vs.* ECOA-K**	**BBBC *vs.* ECOA-K**	**COA *vs.* ECOA-K**
z	–2.850	–2.090	–2.000	–2.096	–2.431
. Sig.	0.004	0.037	0.04	0.036	0.01

Tables 3 and 4 show the results obtained by statistical analysis of the proposed algorithm and the other compared algorithms based on purity criteria and CV index on the iris dataset.

According to table 3 and the significance level (below 0.05), H0 was rejected. Therefore, at the confidence level of 95%, the suggested method was superior to the other algorithms in terms of the purity index.

Table 4. Statistical analysis of algorithms based on CV criteria on iris dataset.

	K-means *vs.* ECOA-K	**CS *vs.* ECOA-K**	**BH *vs.* ECOA-K**	**BBBC *vs.* ECOA-K**	**COA vs. ECOA-K**
z	–2.848	–2.803	–2.823	–2.8112	–2.805
. Sig.	0.004	0.005	0.005	0.005	0.005

According to table 4 and the significance level (below 0.05), H0 was rejected. Therefore, at the confidence level of 95%, the suggested method was superior to the other algorithms in terms of the CV index. The statistical tests demonstrated that the experimental results were stable and trustable, and we could expect that the proposed approach performed the same in different executions.

8.2. Evaluations on CMC dataset

The second experiment concerns the evaluations of the proposed algorithm on the CMC dataset, which are depicted in table 5.

As seen in the results tabulated in table 5, the ECOA-K algorithm achieved the best results among all the algorithms. The results obtained on the CMC dataset showed that the proposed algorithm was capable of achieving solutions of higher purity degree as compared to the other algorithms. The purity index in ECOA-K equaled 0.4427, while this index in K-means, CS, BH, BBBC, and COA were 0.4320, 0.4349, 0.4388, 0.4337, and 0.4354, respectively.

Table 5. Experimental results on CMC dataset.

Algorithm	**Purity degree**	**CV index**	**Convergence**	
			Number of iterations	**Fitness function calculations**
K-means	0.4320	0.041	14.5	14.5
CSA	0.4349	0.050	3.1	260
BH	0.4388	0.055	-	-
BBBC	0.4337	0.011	-	-
COA	0.4354	0.07	12.2	4838.7
ECOA-K	0.4427	0.004	1.2	1697

Also the proposed algorithm was able to converge in 1.2 iterations, in average, while BH and BBBC were unable to reach the convergence condition in some executions. Other algorithms suffered from a low convergence rate in low iterations (a dash is used to imply no convergence rate) for huge datasets, the suggested algorithm can achieve the optimal result with a higher purity degree in a less iteration. Achieving the optimum solution is always guaranteed in the proposed approach, which is due to better initialization and migration of the cuckoos that leads to better coverage of the algorithm throughout the iterations.

Moreover, the CVindex in the suggested algorithm was lower than all the other algorithms. Therefore, this algorithm was capable of providing more stable results as compared to the other algorithms. The CV index in ECOA-K equaled0.004, while this index in K-means, CS, BH, BBBC, and COA was0.041, 0.050, 0.055, 0.011, and 0.07, respectively.

As seen in the results obtained for the CMC dataset, the ECOA-K algorithm was far superior to the other algorithms. Again, statistical analysis was carried for these experiments. Tables 6 and 7

show the results obtained by the statistical analysis of the proposed algorithm and the other algorithms based on the purity criteria and the CV index in the CMC dataset.

Table 6. Results obtained by statistical analysis of algorithms based on purity criteria on CMC dataset.

	K-means *vs.* ECOA-K	CS *vs.* ECOA-K	BH *vs.* ECOA-K	BBBC *vs.* ECOA-K	COA *vs.* ECOA-K
z	–2.913	–2.608	–2.429	–2.987	–2.193
Sig.	0.004	0.009	0.015	0.003	0.028

Table 7. Results obtained by statistical analysis of algorithms based on CV criteria on CMC dataset.

	K-means *vs.* ECOA-K	CS *vs.* ECOA-K	BH *vs.* ECOA-K	BBBC *vs.* ECOA-K	COA *vs.* ECOA-K
Z	–2.848	–2.608	–2.805	–2.807	–2.805
Sig.	0.004	0.009	0.005	0.005	0.005

According to table 7, at the confidence level of 95%, the suggested method was superior to the other algorithms in terms of the CV index.

8.3. Evaluations on wine dataset

The third experiment concerns the evaluation of the proposed algorithm on the wine dataset (Table 8). The simulation results given in table 8 show that again our proposed algorithm was capable of achieving solutions of a higher purity degree in comparison with the other methods. For the wine dataset, the purity index for k-means, CS, BH, BBBC, and COA were 0.6980, 0.7089, 0.7132, 0.7106, and 0.7185, respectively, while it was 0.7196 for ECOA-K. As it can be seen in table 8, the CV index values for the suggested algorithm were significantly less compared with the other algorithms.

Table 8. Experimental results on wine dataset.

Algorithm	Purity degree	CV Index	Convergence	
			Number of iterations	Fitness function calculations
K-means	0.6980	0.046	8.4	8.4
CSA	0.7089	0.040	3.4	4.6
BH	0.7132	0.033	16.5	1142
BBBC	0.7106	0.08	13.3	1067.5
COA	0.7185	0.01	4.7	468.1
ECOA-K	0.7196	0.01	4.4	511.3

Tables 9 and 10 show the results obtained by the statistical test of the proposed algorithm and the other algorithms based on the purity criteria and CV index in the wine dataset. According to table 9, at the confidence level of 95%, the suggested method was superior to the other algorithms in terms of the purity index.

Table 9. Results obtained by statistical analysis of algorithms based on purity criteria on wine dataset.

	K-means *vs.* ECOA-K	CS *vs.* ECOA-K	BH *vs.* ECOA-K	BBBC *vs.* ECOA-K	COA *vs.* ECOA-K
z	–2.859	–2.328	–2.070	–2.312	–1.34
. Sig.	0.004	0.02	0.038	0.021	0.18

Table 10. Statistical analysis of algorithms based on CV criterion on wine dataset.

	K-means *vs.* ECOA-K	CS *vs.* ECOA-K	BH *vs.* ECOA-K	BBBC *vs.* ECOA-K	COA *vs.* ECOA-K
z	–2.772	–2.814	–2.744	–2.603	–2.224
Sig.	0.006	0.005	0.006	0.009	0.026

Again, according to table 10 and the significance level (below 0.05), H0 was rejected. Therefore, at the confidence level of 95%, the suggested method was superior to the other algorithms in terms of the CV index.

9. Conclusion

In this paper, a novel hybrid methodology called ECOA-K was introduced and debated in detail. The hybrid ECOA-K algorithm is a combination of a modified COA and the K-means algorithm. In the hybrid new algorithm, we used the ECOA algorithm to select the initial centers for the K-means algorithm. The proposed algorithm is an extension of the classic COA algorithm with more intelligent and enhanced operations. These modifications include chaotic initial population generation, a systematic egg-laying procedure, and a modified migration function, all with the purpose of increasing the global search and convergence rate of the algorithm.

The experimental results using three benchmark datasets showed that the proposed optimization algorithm was capable of achieving solutions of higher purity degree in comparison with some recent methods, and the algorithm was capable of providing more stable results. The proposed ECOA-K algorithm is more efficient in finding the global optimum solution than the other compared algorithms. It can find high-quality

solutions and provides a small coefficient of variance. Also the convergence rate of the proposed algorithm was faster than the other algorithms.

Regardless of the robustness and efficiency of the hybrid ECOA-K algorithm, it is applicable when the number of clusters is known a priori. In the future research works, the proposed algorithm can also be utilized for many different application areas, for example, clustering of unbalanced data. In addition, its performance can be improved via combining with some other evolutionary algorithms properly. Developing a method for selecting the algorithm parameters can be another good direction for future works.

References

[1] Chuang, L., Hsiao, C. & Yang, C. (2011). Chaotic particle swarm optimization for data clustering. Expert Systems with Applications, vol. 38, no. 12, pp. 14555–14563.

[2] Duda, R., Hart, P.E. & Stork, D.G. (1973). Pattern classification and scene analysis. Wiley-Interscience Publication, New York.

[3] Mac Queen, J. (1967). Some methods for classification and analysis of multivariate observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297.

[4] Niknam, T., TaherianFard, E., Pourjafarian, N. & Rousta, A. (2011). An efficient hybrid algorithm based on modified imperialist competitive algorithm and K-means for data clustering. Engineering Applications of Artificial Intelligence, vol. 24, no. 2, pp. 306-317.

[5] Singh, S. & Chauhan, N. (2011). K-means v/s K-medoids: A comparative study. National Conference on Recent Trends in Engineering & Technology.

[6] Khazaei, A. & Ghasemzadeh, M. (2015) Comparing k-means clusters on parallel Persian-English corpus. Journal of AI and Data Mining, vol. 3, no. 2, pp. 203-208.

[7] Sung, C. & Jin, H. (2000). A tabu-search-based heuristic for clustering. Pattern Recognition, vol. 3, no. 33, pp. 849–858.

[8] Shelokar, P., Jayaraman, V. & Kulkarni, B. (2004). An ant colony approach for clustering. AnalyticaChimicaActa, vol. 509, no. 2, pp. 187–195.

[9] Fathian, M. & Amiri, B. (2007). A honey-bee mating approach on clustering. International Journal of Advanced Manufacturing Technology, vol. 38, no. 7-8, pp. 809–821.

[10] Laszlo, M. & Mukherjee, S. (2007). A genetic algorithm that exchanges neighboring centers for k-means clustering. Pattern Recognition Letters, vol. 28, no. 16, pp. 2359–2366.

[11] Niknam, T., Olamaie, J. & Amiri, B. (2008). A hybrid evolutionary algorithm based on ACO and SA for cluster analysis. Journal of Applied Science, vol. 8, no. 15, pp. 2695–2702.

[12] Niknam, T., Amiri, B., Olamaie, J. & Arefi, A. (2009). An efficient hybrid evolutionary Optimization algorithm based on PSO and SA for clustering. Journal of Zhejiang University Science, vol. 10, no. 4, pp. 512–519.

[13] Rana, S. & Jasola, S. (2010). A hybrid sequential approach for data clustering using K-Means and particle swarm optimization algorithm. International Journal of Engineering, Science and Technology, pp. 167-176.

[14] Bahmani Firouzi, B., Shasadeghi, M. & Niknam, T. (2010). A new hybrid algorithm based on PSO, SA and K-means for cluster analysis. International Journal of Innovative Computing Information and Control, vol. 6, no. 4, pp. 1–10.

[15] Niknam, T. & Amiri, B. (2010). An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. Applied Soft Computing, vol. 10, no. 1, pp. 183–197.

[16] Hatamlou, A., Abdullah, S. & Hatamlou, M. (2011). Data Clustering Using Big Bang–Big Crunch Algorithm. Innovative Computing Technology, Communications in Computer and Information Science, pp. 383-388.

[17] Hatamlou, A. (2013). Black hole: A new heuristic optimization approach for data clustering. Information Sciences, pp. 175-184.

[18] Manikandan, P. & Selvarajan, S. (2014). Data clustering using cuckoo search algorithm (CSA). Advances in Intelligent Systems and Computing, vol. 236, pp. 1275-1283.

[19] Nanda, S. J. & Panda, G. (2014). A survey on nature inspired metaheuristic algorithms for partitional clustering. Swarm and Evolutionary Computation, vol. 16, pp. 1–18.

[20] Alam, S., Dobbie, G., Koh, Y., Riddle, P. & Rehman, S. (2014). Research on particle swarm optimization based clustering: A systematic review of literature and techniques. Swarm and Evolutionary Computation, vol. 17, pp. 1–13.

[21] Hatamlou, A., Abdullah, S. & Nezamabadi-pour, H. (2012). A combined approach for clustering based on K-means and gravitational search algorithms. Swarm and Evolutionary Computation, vol. 17, pp. 47–52.

[22] Yildiz, A. R., Kurtuluş, E., Demirci, E., Sultan Yildiz, B. & Karagöz, S. (2016). Optimization of thin-wall structures using hybrid gravitational search and Nelder-Mead algorithm. Materials Testing, vol. 58, no. 1, pp. 75-78.

[23] Yildiz, A. R. (2009). A novel particle swarm optimization approach for product design and manufacturing. International Journal of Advanced Manufacturing Technology, vol. 40, no. 5-6, pp. 617-628.

[24] Yildiz, A. R. (2013). A new hybrid differential evolution algorithm for the selection of optimal machining parameters in milling operations. Applied Soft Computing, vol. 13, no. 3, pp. 1561–1566.

[25] Yildiz, A. R. (2013). Comparison of evolutionary based optimization algorithms for structural design optimization. Engineering Applications of Artificial Intelligence, vol. 26, no. 1, pp. 327–333.

[26] Yildiz, A. R. (2012). A comparative study of population-based optimization algorithms for turning operations. Information Sciences, vol. 210, pp. 81-88.

[27] Yildiz, A. R. & Solanki, K. N. (2012). Multi-objective optimization of vehicle crashworthiness using a new particle swarm based approach. International Journal of Advanced Manufacturing Technology, vol. 59, no. 1-4, pp. 367-376.

[28] Yildiz, A. R., Öztürk, N., Kaya, N. & Öztürk, F. (2006). Hybrid multi-objective shape design optimization using Taguchi's method and genetic algorithm. Structural and Multidisciplinary Optimization, vol. 34, no. 4, pp. 317-332.

[29] Yildiz, A. R. (2013). A new hybrid bee colony optimization approach for robust optimal design and manufacturing. Applied Soft Computing, vol. 13, no. 5, pp. 2906-2912.

[30] Yildiz, A. R. (2013). Optimization of cutting parameters in multi-pass turning using artificial bee colony-based approach. Information Sciences, vol. 220, pp. 399–407.

[31] Rajabioun, R. (2011). Cuckoo Optimization Algorithm. Applied Soft Computing, pp. 5508-5518.

[32] Shadkam, E. & Bijari, M. (2014). Evaluating the Efficiency of Cuckoo Optimization Algorithm. International Journal on Computational Sciences & Applications (IJCSA), vol. 4, no. 2, pp.39-47.

[33] Yang, X. & Deb, S. (2014). Cuckoo search: recent advances and applications. Neural Computing and Applications, vol. 24, no. 1, pp. 169-174.

[34] Ameryan, M., Akbarzadeh Totonchi, M. R. & Seyyed Mahdavi, S.J. (2014). Clustering Based on Cuckoo Optimization Algorithm, 2014 Iranian Conference on Intelligent Systems (ICIS), pp. 1-6.

[35] Merz, C. & Blake, C. L. UCI Repository of Machine Learning Databases. <http://www.ics. uci. edu/-mlearn/MLRepository. html>.

[36] Loh, W. & Shih, Y. (2000). A comparison of prediction accuracy, complexity and training time of thirty-three old and new classification algorithms. Kluwer Academic Publishers, Manufactured in the Netherlands Machine Learning, pp. 203-228.

[37] Cortez, P., Cerdeira, A., Almeida, F., Matos, T. & Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. Decision Support Systems, vol. 47, no. 4, pp. 533-547.

[38] Yildiz, A. R. (2013). Cuckoo search algorithm for the selection of optimal machining parameters in milling operations. International Journal of Advanced Manufacturing Technology, vol. 64, no. 1-4, pp. 55-61.

[39] Durgun, I. & Yildiz, A. R. (2012). Structural design optimization of vehicle components using Cuckoo search algorithm. Materials Testing, vol. 54, no. 3, pp. 185-188.

[40] Yildiz, B. S., Lekesiz, H. & Yildiz., A. R. (2016). Structural design of vehicle components using gravitational search and charged system search algorithms, Materials Testing, vol. 58, no. 1, pp. 79-81.

[41] Zhao, J., Lei, X., Wu, Z. & Tan, Y. (2014). Clustering using improved cuckoo search algorithm. Springer International Publishing Switzerland, pp. 479–488.

[42] Zhao, Y. & Karypis, G. Evaluation of hierarchical clustering algorithms for document datasets. Proceedings of the International Conference on Information and Knowledge Management, pp. 515-524.

[43] Derrac, J., Garcia, S., Molina, D. & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation, vol. 1, no. 1, pp. 3-18.

[44] Mendenhall, W., Beaver, R. J. & Beaver, B. M. (2010). Introduction to Probability and Statistics, Thomson publications, 12th edition.