Engineering with Computers (2022) 38 (Suppl 3):S1973–S1999
https://doi.org/10.1007/s00366-021-01345-3

ORIGINAL ARTICLE

Variants of bat algorithm for solving partitional clustering problems

Yugal Kumar[1] · Arvinder Kaur[1]

Received: 30 July 2020 / Accepted: 9 February 2021 / Published online: 11 March 2021

Abstract

Clustering is an exploratory data analysis technique that organize the data objects into clusters with optimal distance efficacy. In this work, a bat algorithm is considered to obtain optimal set of clusters. The bat algorithm is based on the echolocation feature of micro bats. Moreover, some improvements are proposed to overcome the shortcoming associated with bat algorithm like local optima, slow convergence, initial seed points and trade-off between local and global search mechanisms etc. An enhanced cooperative co-evolution method is proposed for addressing the initial seed points selection issue. The local optima issue is handled through neighbourhood search-based mechanism. The trade-off issue among local and global searches of bat algorithm is addressed through a modified elitist strategy. On the basis of aforementioned improvements, three variants (BA-C, BA-CN and BA-CNE) of bat algorithm is developed and efficacy of these variants is tested over twelve benchmark clustering datasets suing intra-cluster distance, accuracy and rand index parameters. Simulation results showed that BA-CNE variant achieves more effective clustering results as compared to BA-C, BA-CN and BA. The simulation results of BA-CNE are also compared with several existing clustering algorithms and two statistical tests are also applied to investigate the statistical difference among BA-CNE and other clustering algorithms. The simulation and statistical results confirmed that BA-CNE is an effective and robust algorithm for handling partitional clustering problems.

Keywords Echolocation · Neighbourhood · Cooperative co-evolution · Elitist strategy · Clustering · Bat algorithm

1 Introduction

Clustering is an unsupervised machine learning approach that divides set of objects into groups or clusters [1, 2]. The objects inside same cluster are similar to each other and are dissimilar to the one present in other clusters [3]. Further, clustering is classified as partitional clustering, hierarchical clustering, grid-based clustering, density-based clustering and model-based clustering [4]. The partitional clustering decomposes the data set into several disjoint groups that are optimal in terms of some predefined criteria. The prime objective of partitional clustering is to maximize the intra-cluster compactness and minimizing inter-cluster likeness. In hierarchical clustering, the formation of the cluster in as tree structure and each node represents a cluster. Furthermore, two approaches, agglomerative (bottom up) and divisive (top down) are followed in hierarchal clustering. In agglomerative approach, each data point is taken in a separate cluster and is merged if it is similar to the other data point of the same cluster. While in divisive approach, all data points belong to one cluster and repeatedly divided into smaller clusters if they are dissimilar. While in grid-based clustering the data space is quantized into finite number of cells that form a grid like structure. However, the clusters are formed according to data item's compactness in density-based clustering. The density associated with a point is obtained by counting the number of points in a region of specified radius around the point. The points with a density above a specified threshold are constructed as clusters. While, the model-based clustering involves the generation of data clusters through the probability distributions and each component represents a cluster. There are number of fields where the clustering techniques has been adopted such as bio-engineering, stock market, pattern recognition, image processing and medical data analysis [5–7].

✉ Yugal Kumar
yugalkumar.14@gmail.com

Arvinder Kaur
er.arvinderdhillon@gmail.com

[1] Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat, Solan, Himachal Pradesh, India

Last few decades, momentous research has been carried in clustering field. Large numbers of meta-heuristic algorithms inspired from natural phenomenon have been reported on partitional clustering [8].These natural phenomena can include swarm intelligence, insect's behaviour, well defined laws of physics, and some other natural process of living beings [9–11].These optimization techniques inspired from natural phenomenon's have been reported in clustering field to obtain augmented solutions like charged system search approach [11], particle swarm optimization [12], magnetic optimization algorithm [13], black hole [14], artificial bee colony algorithm [15], ant colony optimization [16] and big bang big crunch algorithm [17].These algorithms consist of approximation function to determine the optimal solution. Further, it is noticed that the local and global search mechanisms of these algorithms should be balanced in to obtain good candidate solution. In clustering techniques number of consecutive steps are followed to partition data objects into clusters such as initialization, evaluation and update etc. In recent time, a Bat algorithm is developed on the basis of echolocation characteristics of bat. Initially, Yang developed the bat algorithm based on echolocation feature of micro-bats [18, 19]. These micro-bats use echolocation feature to detect prey and avoid obstacles. In search of pray, microbats emit short pulse and wait for echo. According to the strength of echo, bats can determine the shape and size of objects. But there are several issues associated with bat algorithm such as ack of trade-off between search mechanism, convergence, trap in local optima etc., [19–23]. These issues can affect the performance of bat algorithm to solve optimization problems. The issues associated with bat algorithm are discussed in details in subsection 1.1.

1.1 Issues with Bat Algorithm (BA)

This subsection discusses the various issues that can affect the performance of BA algorithm. Tang et al. [24] addressed the diversity and local optima issues of bat algorithm through adaptive inertial weight strategy and Doppler effect-based mechanism. Gan et al. [25] stated that BA can easily trapped in local optima and also produced unstable results due to weak global exploration ability. So, an iterative search and stochastic inertia weight methods are designed to address the local optima and weak global exploration ability issues. BA obtained the good optimal results for nonlinear optimization problems, but converged on less optimal solution (premature solution) with complex optimization problems [26]. Hence, premature solution issue can be rectified using a double mutation operator and this operator is incorporated into position updated equation of BA. The local search ability and local minima issues are addressed through external optimization-based method and chaos strategy [27]. Similarly, an iterative hill climbing based method and clustering based similarity control framework are developed for handling local search and optima [28]. The balancing factor and local minima issues are considered and resolved through weight co-efficient strategy and DE inspired local search ability [29]. Ghanem and Jantan [30] addressed the local optima issue using mutation operator and this operator generates diverse population for escaping the local optima. Cui et al. [31] considered the population diversity and low global search ability issues and rectified using levy mechanism and principal component analysis. It is seen that optimization performance and convergence speed of BA algorithm highly depend on global search ability [32], and bat velocity component having major contribution in global search. Sometimes, the updation in velocity can affect the performance of algorithm. So, a triangle flipping strategy-based mechanism is developed for updating the velocity of bat and in turn, achieving the better convergence rate. BA consists of different user defined parameters such as pulse rate, pulse loudness, velocity etc. Zhu and Wang [33] investigated the impact of these parameters on BA performance and exploited different optimal values for pulse rate and pulse loudness parameters. The local minima and slow convergence rate issues resolved through inertia weight concept and improved dimensional size [34]. An island model-based strategy is presented for handling the population diversity issue [35]. A neighbourhood operator is designed for addressing the premature convergence [36]. This operator generates. Further, initial population selection also affects the performance of BA. The population selection issue of BA discussed in [23, 37, 38]. From the above discussion, the major issues associated with BA are summarized as local optima/minima, premature convergence, population diversity, convergence rate, local search (exploration), global search (exploitation) and its balancing, parameter tuning etc. To make BA more robust and effective, this work considers the following issues listed as.

- *Local optima* It is noticed that several times, bat algorithm traps in local optima [24, 25, 27]. This problem occurs due to lack of appropriate mechanism for updating the population of micro-bats as well balancing of searching processes.
- *Convergence speed* The convergence rate of an algorithm depends on optimization process and its search space [32, 34]. The convergence rate can also be affected due to less coordination among exploration and exploitation processes.
- *Initial population selection* Initial population selection having significant impact on the performance of algorithm for attaining the global optimum solution especially in terms of clustering algorithms [23, 37, 38]. If, the population is not selected in effective manner, then premature convergence problem can occur.

1.2 Contribution of the work

The aim of this work is to address the aforementioned issues of bat algorithm and the main contributions of work are highlighted as

1. To develop an enhanced cooperative co-evolution method for selecting the initial population.
2. To design a neighbourhood strategy for handling local optima issue.
3. To develop an elitist strategy for improving the convergence speed.
4. The aforementioned improvements (1–3) are incorporated into BA and checked the efficiency of algorithm for solving partitional clustering problems.

Rest of paper is organized as follows. Section 2 presents related works on clustering problems. Section 3 describes bat algorithm. The improvements in bat algorithm are listed in Sect. 4. Section 5 discusses the experimental and statistical results of the proposed work. The entire work is concluded in Sect. 5.

2 Related work

This section describes the related works in the field of partitional clustering problems. Since past few decades, large numbers of clustering algorithms have been developed to solve partitional clustering problems.

To determine best initial population and automatic cluster numbers, Rahman and Islam [39] designed a novel hybrid clustering algorithm based on K-means and genetic algorithm. In this work, genetic algorithm is applied to determine optimized initial cluster centres for K-means algorithm. Further, it is noticed that instead of K-means algorithm, Fuzzy C-means algorithm is adopted to obtain optimum clustering results. The performance of proposed algorithm is tested on twenty benchmark clustering datasets. The simulation results of proposed algorithm are compared with five well-known clustering techniques. It is seen that the proposed algorithm gives state of art clustering results.

Liu et al. [40] presented a clone selection algorithm for automatic clustering. In this work, a new genetic operator, called antibody gene transposon is introduced to detect number of clusters automatically. The performance of proposed algorithm is tested on twenty-three data sets and compared with ILS, ACDE, VGA and DCPSO algorithms. It is observed that the proposed algorithm gives better results without the prior knowledge of number of clusters.

To handle the initialization issue of K-mean algorithm, Cao et al. [41] developed a new initialization method based on neighbourhood rough set model. The proposed initialization method is integrated with K-mean algorithm. In this work, intra-cluster similarity and inter-cluster similarity of an object are represented in terms of cohesion and coupling. The performance of proposed algorithm is tested on three well-known datasets and compared with other two initialization algorithms. Authors claimed that proposed initialization method provides superior results than traditional methods. To optimize clustering problems, a two-step artificial bee colony algorithm is reported in [42]. In this work, authors have proposed three improvements in ABC algorithm. These improvements are initial cluster centre positions, modified search equations and abandoned food source locations. The initial cluster centre positions are determined through one step K-means algorithm. A PSO based search equation is developed to discover more promising search space. The abandoned food source location is determined using Hooke and Jeeves based search method. The performance of algorithm is tested on both of artificial and real-life data sets and compared with well-known clustering algorithms. It is seen that the proposed algorithm significantly improves the performance of conventional artificial bee colony algorithm.

Erisoglu et al. [43] developed a new initialization method for K-means algorithm to overcome the randomized cluster centre initialization problem. In this work, two variables (attributes) of a dataset are selected to map the data in bi dimensional feature space. The first variable is an attribute having maximum value of the coefficient of variation, called main axis. The second variable is determined through the value of the correlation between main axis (first variable) and rest of attributes. The attribute having minimum value of correlation is chosen as second variable. The several benchmark datasets are used to evaluate the performance of the proposed algorithm. From simulation results, it is noticed that proposed method significantly improves the performance of K-means algorithm. Hatamlou [44] developed a novel binary search algorithm to obtain high quality clusters with better convergence speed. In this work, initial seed points are generated from different location and data objects are assigned to nearest one. Further, objective function is evaluated, if value of current objective function is lesser then previous objective function value, the search will proceed in same direction otherwise it proceeds in opposite direction. The performance of proposed algorithm is tested on six benchmark datasets and compared with K-means, GA, SA, TS, ACO, HBMO and PSO. It is seen that proposed algorithm effectively solves clustering problems.

Zalik [45] presented a K-Means algorithm for clustering problems to work without pre-assignment of cluster numbers. In this work, the cost-function of k-means is extended with new cost-function. The proposed algorithm works in two phases. In first phase, k numbers of cluster centres are allocated in such a manner that each cluster consists of one or more cluster centres. While, in second phase, cluster centres are adjusted to minimize the cost-function. The performance of this algorithm is tested on several benchmark datasets. It is stated that proposed algorithm efficiently determines the actual number of cluster centres.

A meta-heuristic algorithm inspired from class room teaching is developed for clustering problem [46]. This algorithm works in two phases- teacher phase and learner phase. In teacher phase, student learns from teacher to improve his knowledge, while in learner phase they interact with other learners. Moreover, some chaotic maps are introduced to generate diverse solution. The properties of K-means algorithm are also investigated for effective clustering results [47]. Six benchmark datasets are used to evaluate the performance of aforementioned algorithm. The performance is measured in terms of overlapping, number of clusters, dimensionality and cluster size.

Taherdangkoo et al. [48] presented a new clustering algorithm based on blind-naked mole-rats algorithm for data clustering. This algorithm is inspired through blind-naked mole-rat colonies that can effectively search food source and protect colony from attacks. The algorithm starts with the population of blind-naked mole rats and searches the whole problem space for optimal solution in random fashion. In next iterations, employed mole rats start movement to target food source and their neighbours. The performance of proposed algorithm is tested on six real-life datasets and compared with other well-known clustering algorithms. Authors claimed that blind-naked mole-rats algorithm provides higher accuracy with faster convergence speed.

Senthilnath et al. [49] adopted fire fly algorithm in clustering field. This algorithm simulates the flash pattern and social insect behaviours of fire flies. The proposed algorithm works in two phases, in first phase, the variation of light intensity is measured. In second phase, the movement of fireflies is measured. The performance of fire fly algorithm is tested on thirteen standard datasets and compared with ABC and PSO. The simulations results show that the FA effectively solves clustering problems.

Han et al. [50] introduced a new diversity mechanism in gravitational search algorithm to handle clustering problems in effective manner. The proposed diversity mechanism is inspired from the collective response of birds. This mechanism works in three steps i.e., initialization, identification (nearest neighbours) and orientation alteration. In initialization step, the candidate population is generated and forwarded to second step i.e., nearest neighbour. In second step, nearest neighbours are identified through a neighbourhood strategy. In third step, the current position of a candidate solution is changed based on nearest neighbour. The performance of the proposed algorithm is tested on thirteen data sets and compared with well-known clustering algorithms. It is seen that proposed algorithm is an effective and efficient algorithm for solving clustering problems.

Zhang et al. [51] presented an artificial bee colony algorithm for data clustering. In this work, bees are categorised into three categories onlooker bees, employed bees and scout bees. The onlooker bees and employed bees are responsible for global search, while scout bees are responsible for local search. Further, Deb's rule is incorporated to redirect search in solution space. The performance is tested on three real-life datasets and compared with other clustering algorithms. It is revealed that the proposed algorithm provides good quality results.

Further, a hybrid version of artificial bee colony algorithm with genetic algorithm presented in [52]. The primary objective of this work is to enhance the information exchange mechanism between bees. To achieve the same, the crossover operator of genetic algorithm is incorporated in artificial bee colony algorithm. The performance of proposed algorithm is evaluated on six benchmark datasets and compared with other ABC, CABC, PSO, CPSO and GA clustering algorithms. The simulation results of proposed algorithm are better than other algorithms being compared.

To solve clustering problems effectively, Kumar and Sahoo [53] developed a new clustering algorithm based on magnetic charge system search algorithm and particle swarm optimization algorithm. In this work, the personal best mechanism of particle swarm optimization algorithm is added into magnetic charge system search algorithm for better exploration. Further, neighbourhood strategy is developed to avoid local optima situation. The performance of MCSS–PSO algorithm is tested on ten datasets and compared with K-means GA, PSO, ACO, CSS, CCSSA and MCSS. From experimental results, it is noted that proposed algorithm provides state of art clustering results.

Zhou et al. [54] introduced simplex method-based social spider optimization algorithm for solving clustering task. In this work, simplex method is incorporated in social spider algorithm, to enhance local search ability and improved convergence speed. The performance of proposed algorithm is tested on eleven datasets and compared with well-known clustering algorithms. It is seen that proposed algorithm perform well in terms of accuracy, robustness, and convergence speed.

Boushaki et al. [55] designed a new quantum chaotic cuckoo search algorithm for data clustering. To extend the global search ability of quantum chaotic cuckoo search algorithm, a nonhomogeneous update mechanism is employed in proposed algorithm. Further, to improve convergence speed, chaotic maps are incorporated in this algorithm. The performance of algorithm is compared with genetic quantum cuckoo search, hybrid cuckoo search and differential evolution, hybrid K-means and improved cuckoo search, standard cuckoo search, quantum particle swarm optimization, differential evolution, hybrid K-means chaotic particle swarm optimization and genetic algorithm. The experimental results show that proposed algorithm performs well in comparison to other algorithms.

A new clustering algorithm based on genetic algorithm and message-based similarity measure is presented in [56].

The message-based similarity measure contains two types of messages-responsibility and availability. These messages are exchanged between data points and cluster centres. The responsibility of data point reflects the evidence regarding cluster centres, while the availability reflects the evidence that about the appropriateness of data point with respect to cluster centres. Further, variable-length real-value chromosome representation and a set of problem-specific evolutionary operators are incorporated in GAMS. The performance of proposed algorithm is tested on both of artificial and real-life data set. The simulation results show that the algorithm obtains significant results for clustering problems.

To overcome the shortcomings of K-means algorithm, Xiao et al. [57] developed a quantum-inspired genetic algorithm for partitional clustering. In this work, Q-bit based representation and rotation operation of quantum gates are applied for better trade-off between exploration and exploitation. The performance of proposed algorithm is tested on both real-life and simulated datasets. Authors claimed that proposed algorithm is able to find optimal cluster centres without prior knowledge of number of cluster centres.

Bijari et al. [58] presented a memory-enriched big bang–big crunch algorithm for clustering. The BB-BC algorithm works in two phases- big bang and big crunch phase. In big bang phase, random points are generated near to initial point. In big crunch phase, these points are optimized in single one. Further, a memory concept with limited size is integrated to make better trade-off between exploration and exploitation. The performance of algorithm is tested on six data sets and compared with well-known algorithms like GA, PSO, GWO and original BB–BC. The simulation results show that BB-BC algorithm provides superior clustering results than other algorithms.

To solve clustering search space problems, Abualigah et al. [59] combined the krill herd optimization algorithm with harmony search to overcome local optima problem. Moreover, a global exploration operator and reproduction procedure is integrated in krill herd algorithm. The performance of proposed algorithm is tested on seven standard datasets and compared with well-known GA, PSO, HS, KHA, H-GA, and H-PSO clustering algorithms. It is seen that improved krill herd algorithm performs well in clustering field.

To determine optimal cluster centres, Pakrashi et al. [60] hybridized Kalman filtering algorithm with K-means algorithm. In this work, the convergence rate of K-Means algorithm is improved through global search of Kalman filtering algorithm. Further, a conditional restart mechanism is also incorporated in K-Means algorithm to handle local optima situation. The performance of proposed algorithm is tested on seven benchmark data sets and compared with HKA, KGA, GAC, ABCC and PSO algorithms. Authors claimed that superior results are provided by the proposed algorithm as compared to other algorithms.

A hybrid clustering algorithm based on K-Means and mussels wandering optimization algorithm is presented for effective clustering in [61]. The proposed hybrid algorithm contains the merits of both the algorithms i.e., local search abilities of K-means and global optimization ability of mussels wandering optimization algorithm. The performance of this algorithm is tested on nine datasets and compared with K-means K-PSO K-MWO algorithms. It is stated that proposed algorithm obtains quality results than other algorithms.

To solve clustering search space problems Wang et al. [62] presented a hybrid version of flower pollination algorithm with bee pollinator. In this work, discard pollen operator of artificial bee colony is used to enhance population diversity and global search ability of flower palliation algorithm. Further, elite based mutation and crossover operator are applied to enhance local search mechanism. Several artificial and real datasets are considered to evaluate the performance of proposed algorithm. The simulation results are compared with k-means, PSO, DE, CS, ABC, FPA algorithms. Authors stated that proposed algorithm is one efficient and effective algorithm for solving clustering problems.

Hatamlou and Hatamlou [63] designed a two-stage clustering approach to overcome the draw backs of particle swarm optimization like local optima and slow convergence speed. The proposed approach works in two-stages, in first stage initial candidate solution is generated through PSO algorithm. In second stage, quality of solution is improved with help of heuristic search algorithm. The performance of proposed algorithm is tested on seven benchmarks and compared with K-means, PSO GSA, BB-BC methods. It is seen that the proposed algorithm determines good quality clusters.

Jiang and Wang [64] developed a PSO based clustering algorithm. In this work, a cooperative co-evolution method is integrated with PSO to enhance the convergence rate and population diversity. The cooperative co-evolution method works as decomposer and PSO algorithm acts as optimizer. The performance of proposed algorithm is tested on some of real-life datasets and compared with PSO, SRPSO, ACO, ABC and DE, and K-means algorithms. The Simulations results showed that proposed algorithm performs well with most of datasets.

To solve clustering problems effectively, an improved version of cat swarm optimization algorithm is reported in [65]. In this work, new search equations are proposed to explore search space in efficient manner. Further, a local search method is also incorporated in CSO algorithm for handling local optima problem. Simulations result showed that proposed algorithm provides effective clustering results.

To perform efficient clustering, Kwedlo [66] combined the differential evolution algorithm with the K-means algorithm. In this work, K-means algorithm is used to tune candidate solutions generated through mutation and crossover operators of differential evolution algorithm. Additionally, a recording procedure is also introduced to handle redundant solutions. The performance of proposed algorithm is compared with five other well-known clustering algorithms. It is noticed that the proposed algorithm gives state of art clustering results.

Yin et al. [67] presented a hybridized version of an improved gravitational search algorithm with k-harmonic means for clustering problems. The performance of proposed algorithm is tested on seven benchmark datasets and compared with other well-known clustering algorithms. Authors claimed that the proposed algorithm effectively overcomes the local optima problem of k-harmonic algorithm and also enhances the convergence speed of gravitational search algorithm.

A hybrid version of ant clustering algorithm with k-harmonic means is presented for handling clustering problems [68]. The proposed algorithm contains the merits of both algorithms i.e., initialization ability of K-harmonic means and local optima ability of ant algorithm. The performance of proposed algorithm is tested on five benchmark datasets and compared with KHM and ACA. It is stated that proposed algorithm achieves better clustering results than compared algorithms, but runtime is slightly longer.

Kaur and Kumar [69] presented a water wave optimization algorithm for handling data clustering problems. It is claimed that WWO provides superior results for most of constrained and unconstrained optimization problems, but sometimes obtains less optimal results with complex optimization problems. In their work, two improvements are incorporated for addressing the absence global best information and premature convergence in terms of updated search mechanism and decay operator. The performance is assessed over thirteen clustering datasets using accuracy and F-score parameters and it is reported that WWO achieves good clustering results as compared other algorithms.

For determining the structure of data in large datasets, an appropriate clustering algorithm is required. A probabilistic fuzzy k-modes (PFKM) algorithm developed for computing the structure of data [70]. This algorithm integrates the possibility concept and fuzzy k-modes for effective analysis of categorical datasets. Moreover, PFKM also integrated with PSO, SCA and GA algorithms. Results confirmed that PFKM improves the simulation results of PSO-PFKM, SCA-PFKM and PFKM-GA significantly using sum of squared error and accuracy. The PSO-PFKM achieves better quality of results with most of datasets among all algorithms.

In case of clustering problems, the optimal solution is highly influenced due to initial cluster sensitiveness and local optima [71]. These problems also responsible for premature convergence. To remove these pitfalls, a multi-search and multi-start framework is developed for addressing the clustering problems in effective manner. This framework directs the search in different direction for computing the optimum solution. Several benchmark datasets are selected to investigating the performance of proposed frameworks and results are compared with several single solution based meta-heuristic algorithms. Results claimed that proposed frame work is effective for solving clustering problems.

Grey wolf optimizer (GWO) is an efficient algorithm for solving variety of optimization problems, but this algorithm traps in local optima and converges on premature solution on large variables problems such as clustering [72]. Tabu search method is adopted for overcoming the aforementioned problems of GWO, called GWOTS. The performance of GWOTS is investigated over thirteen well defined clustering datasets and it is observed that GWOTS achieves better results in terms of SSE, entropy and purity.

In clustering filed, FCM is popular clustering algorithm, but it lacks in terms of determining the intersection of clusters. Further, this algorithm is also sensitive to initial centroids. Zhou et al. [73] considered these problems of FCM and developed a new kernel intuitionistic FCM (KIFCM) algorithm. The features of KIFCM are described as 1) adopted new similarity measure based on kernel function for determining the intersection of clusters. 2) state transition algorithm is applied for handling initial centroids sensitiveness, called STA-KIFCM. Simulation results are taken over five clustering datasets and it is found that KIFCM reveals more effective clustering results.

3 Bat algorithm

Bat algorithm is based on the echolocation behaviour of microbats. Microbats use echolocation to detect prey and avoid obstacles. In search of pray, microbats emit short pulse and listen echo from surrounding objects to determine its shape and size. The loudness, emission rate and random variable of bat algorithm are initialized using Eqs. 1–2.

$$A_i^{t+1} = \alpha\left(A_i^t\right) \tag{1}$$

$$r_i^{t+1} = \left[1 - \exp\left(-\alpha\right)\right] \tag{2}$$

where A_i^t is loudness, r_i^t is pulse emission rates and (α) is a user specified variable having values between [0–1]. The frequency and velocity of bats are computed using Eqs. 3,4.

$$f_i^t = f_{\min}^t + \left(f_{\max}^t - f_{\min}^t\right)\text{rand}() \tag{3}$$

$$v_i^t = v_i^{t-1} + \left(x_i^t - x_*\right)f_i^t \tag{4}$$

wheref_{min}^t and f_{max}^t denotes lowest and highest frequencies at time stamp (t) and rand () is a random function having values between [0–1], v_i^{t-1} is initial velocity, x_i^t is current position and x_* is current best position. The positions of bats are updated using Eqs. 5,6

$$x_i^{t+1} = X_{\text{new}} + v_i^t \tag{5}$$

$$X_{\text{new}} = x_i^t + \text{rand}i[-1, 1]A_i^t \tag{6}$$

where X_{new} is new position and x_i^{t+1} is final updated position.

4 Data clustering

Data clustering is a process to divide the dataset into a several disjoint clusters which are optimal in terms of some predefined criteria and it is an unsupervised technique in nature. Generally, Euclidean distance is used as similarity measure to determine the group of similar data objects. It can be defined as the distance between data objects and cluster centres. In clustering, Euclidean distance is computed for each cluster centre to each data objects. On the basis of minimum Euclidean distance, data objects are associated with clusters. The Euclidean distance can be computed using Eq. 7.

$$d(X, C) = \text{minimun} \sum_{i=1}^{n} \sqrt{\left(X_i\right)^2 - \left(C_k\right)^2} \tag{7}$$

where X_i denotes the ith data object of a dataset, C_j represents the kth cluster centre, n denotes the total number of data objects in a dataset. The basic steps of BA clustering algorithm are listed in Algorithm 1.

5 Proposed improvements in Bat algorithm

To improve the performance and efficiency of bat algorithm, three improvements are proposed. These improvements are enhanced co-operative co-evaluation for initial seed points selection, elitist strategy for better tradeoff between local and global solutions, and neighbourhood strategy to avoid local optima. Hence in this work, three variants of bat algorithm are proposed on the basis of improvements. The first variant of BA, called BA-C considers the initial seed points selection issue and it can be resolved through enhanced co-operative co-evolution method. Second variant of BA, called BA-CN addresses the initial seed points selection and local optima issues using enhanced co-operative co-evaluation method and neighbourhood search strategy respectively. Third BA variant (BA-CNE) considers the initial seed points selection, local optima and trade-off issue among local and global searches of BA and addresses these using enhanced cooperative co-evolution method, neighbourhood search strategy and modified elitist strategy and considers. Further, this work also investigates the impact of each issue related to BA on its performance as well as outcome of each improvements. The detailed descriptions of these improvements can be described as below.

5.1 First improvement: enhanced co-operative co-evolution method

From extensive literature review, it is observed that efficiency of clustering algorithms also depends on initial cluster points. In literature, several initialization methods are reported to address initial cluster points problem of clustering algorithms [39–42]. Hence, for improving the performance of bat algorithm, an enhanced co-operative co-evolution framework is introduced to select initial cluster centres. The co-operative co-evolution method works in divide and conquers manner. In this method, problem is divided into

Algorithm 1: Steps of Bat Algorithm (BA) for clustering

```
Step 1:  Load the dataset and initialize the basic parameters of clustering algorithm such as
         number of clusters (K), total number of data objects (n), number of attributes (d),
         maximum number of iterations etc.
Step 2:  Select initial cluster centres from the data set.
Step 3:  Evaluate the objective function using equation 7.
Step 4:  Assign data objects to clusters using minimum value of objective function.
Step 5:  Update the cluster centres.
Step 6:  Check the stopping criteria, if met, then stopped. Otherwise repeat the steps 3-5.
Step 7:  Obtain optimal cluster centres.
```

sub problems and each sub problem is solved individually and final solution is obtained by combing each sub problem solution. The goal of partitional clustering is to partition the datasets into number of clusters that are optimal in-terms of some predefined criteria. To achieve same, a co-operative co-evolution method with centroid selection mechanism is proposed. This method includes several points which are discussed as below.

- Number of partitions
- Size of partition
- Selection criteria for centroid

5.1.1 Population partitions and size description

To implement the co-operative co-evolution method, first task is to divide population (data instances) into several predefined partitions. The number of partitions is equal to number of clusters (K) for a given dataset as defined using Eq. 8.

$$p_n \propto K \tag{8}$$

where p_n denotes number of partitions and K denotes the number of clusters. The size of subpopulation is obtained through Eq. 9.

$$p_s = T/K \tag{9}$$

where T is total population size i.e., total number of data instances in a dataset, K is number of clusters and p_s denotes size of subpopulations. The number of subpopulations is computed through Eq. 10.

$$\begin{cases} p_{s1} = 1 \ \text{ to } \lceil p_s \rceil \\ p_{s2} = \lceil \mathrm{UL}(p_{s1}) + 1 \rceil \ \text{ to } \lceil p_{s1} + p_s \rceil \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots \\ p_{s(n-1)} = \left\lceil \mathrm{UL}(p_{s(n-2)}) + 1 \right\rceil \ \text{ to } \ \dots + \lceil p_{s2} \rceil + \lceil p_{s1} \rceil + \lceil p_s \rceil n = \ \{1,\ 2,\ 3 \dots\dots K\} \\ p_{sn} = \left\lceil \mathrm{UL}(p_{s(n-1)}) + 1 \right\rceil + \sum_{n=0}^{K-1} \lceil p_s \rceil \end{cases} \tag{10}$$

In Eq. 10, $p_{s1}, p_{s2}, \dots p_{sn}$ denotes subpopulations. The number of subpopulations is equal to number of clusters. UL represents the upper limit a sub population. Further, from each subpopulation an appropriate centroid is selected using Eq. 11.

$$C_{kn} = \min(p_{sn}) + (\max(p_{sn}) - \min(p_{sn})) * \mathrm{rand}(0,1); \quad \text{where } n = 1 \text{ to } K, \tag{11}$$

where C_k denotes kth cluster centre, $\min(p_n)$ and $\max(p_n)$ are minimum and maximum values of each n$^{\text{th}}$ sub population (p_{sn}) and rand () is a random number in the range of 0 and 1.

The steps of BA-C clustering algorithm are mentioned in Algorithm 2. And the working of co-operative co-evolution method is illustrated using iris dataset. This dataset contains one hundred fifty data instances and four attributes. These data instances are divided into three classes. Hence number of clusters (K) considered for iris dataset is three. The co-operative co-evolution method contains the Eqs. (8–11) to determine the initial population for clustering algorithm in terms of cluster centres.

(a) First step is to divide the population into subpopulation. Equation 8 is used to compute the subpopulation.
$p_n \propto K$, For iris dataset, $K = 3$;

$$p_n = 3$$

(b) In second step, size of subpopulations is computed. The size of subpopulation is computed through Eq. 9.
$p_s = T/K$, where $T = 150$ and K = 3;
$p_s = 150/3 = 50$.
The size of subpopulation $\left(p_s\right)$ is 50. Further, the number of subpopulations is determined using Eq. 10.
$p_{s1} = 1$ to p_s;$p_s = 50$;

$$p_{s1} = 1 \text{ to } 50,\ \mathrm{LB}\ (p_{s1}) = 1 \text{ and } \mathrm{UB}(p_{s1}) = 50;$$

$$p_{s2} = \mathrm{UB}\left(p_{s1}\right) + 1 \text{ to } p_{s1} + p_s;\ \mathrm{UB}\left(p_{s1}\right) = 50,\ p_s = 50;$$

$$p_{s2} = 51 \ \text{ to } 100;\ \mathrm{LB}\left(p_{s2}\right) = 51,\ \ \mathrm{UB}\left(p_{s2}\right) = 100;$$

$$p_{sn} = \left\lceil \mathrm{UB}(p_{s(n-1+1)}) \right\rceil + \sum_{i=0}^{n-1} \lceil p_s \rceil; \quad n = 3$$

$$p_{s3} = \lceil \mathrm{UB}(p_{s2}) + 1 \rceil + \sum_{i=0}^{2} \lceil p_s \rceil; \quad p_{s0} = p_s$$

$$p_{s3} = 101 \ \text{ to } \ 150;$$

{5.2810 2.4032 4.8048 1.5397}.

(c) In third step, Eq. 11 is used for computing initial cluster centres for clustering algorithm.

- When $n = 1$ $c_{k1} = \min(1 : 50) + (\max(1 : 50) - \min(1 : 50))$*rand()
  {5.5221 4.0109 1.7333 0.5074}.
- When $n = 2$ $c_{k2} = \min(51 : 100) + (\max(51 : 100) - \min(51 : 100))$*rand()
  {6.8022 3.2681 4.9022 1.7246}.
- When$n=3$$c_{k3} = \min(101 : 150) + (\max(101 : 150) - \min(101 : 150))$*rand()
  {5.2810 2.4032 4.8048 1.5397}.

Algorithm 2: Pseudo code of Bat algorithm with Enhanced cooperative co-evolution (BA-C) method

```
Step 1:  Load the data sets and specify the number of clusters $K_i \in (i = 1, 2, \dots, n)$
Step 2:  Select initial clusters positions using enhanced cooperative co-evolution strategy equations 8-11.
Step 3:  Initialize pulse rates ($r_i$) loudness ($A_i$)and initial velocity($v_i$) using eequations 1-2.
Step 4:  Evaluate the objective function using equation 7 and assign items to the clusters with
         minimum objective function values
Step 5:  While (t < max number of iterations /* t = iteration number), do
Step 6:  Calculate pulse frequency $f_i^t$ and velocity $v_i^t$ using equations 3-4.
Step 7:  if (rand >$r_i$)
         Accept $X_n$ position as solution
         Else
Step 8:  Update the position equations 5-6.
         Increase $r_i$ and reduce $A_i$
         Accept the new solution
         end if
Step 9:  Recalculate the objective function and assign items with minimum objective function values
Step 10: Memorize the result and store in memory pool
         t=t+1;
Step 11: End while
Step 12: Global optimal solution with highly accurate value
```

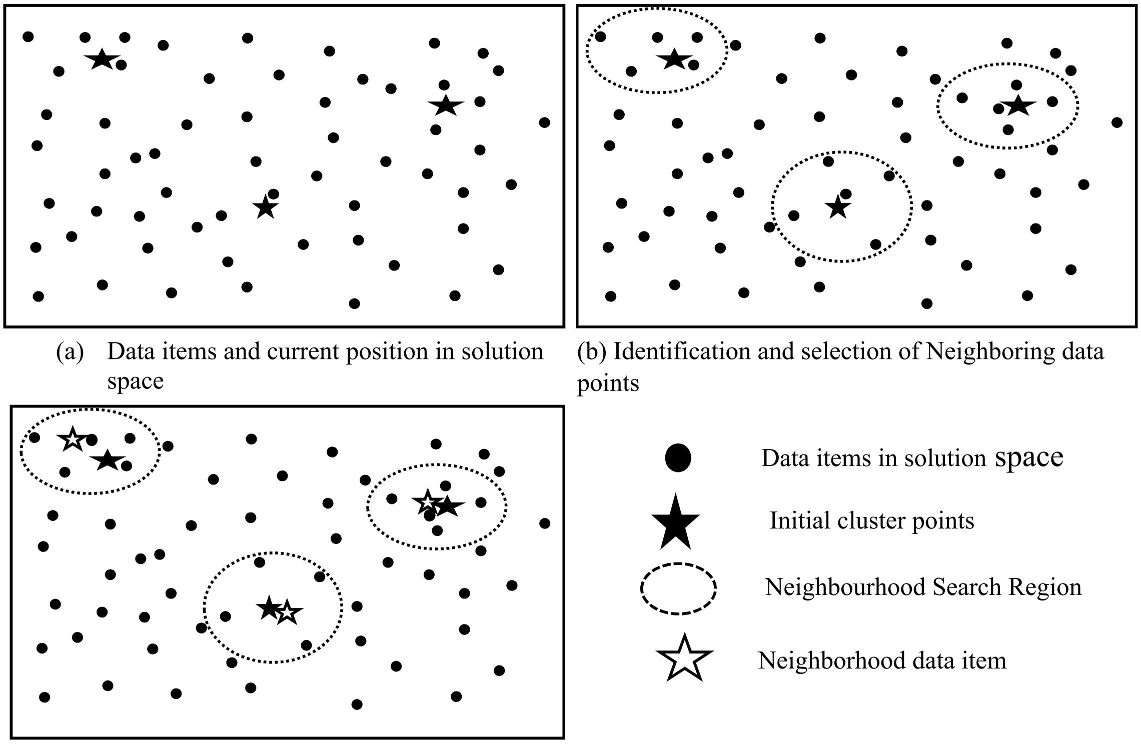
6 Second improvement: neighbourhood search strategy with harmonic mean

In clustering, local optima is a prominent problem in which solution converges on local best solution. In other words, it is interpreted as there is no update in solutions in subsequent iterations. There are numerous strategies that can protect algorithms to be avoiding local optima. In-order to overcome the local optima problem, a neighbourhood search strategy based on harmonic mean is incorporated into bat algorithm. The proposed neighbourhood strategy works in two phases- Identification Phase and Evaluation Phase. In first phase neighbours are identified, while in second phase they are evaluated to produce single outcome. The process of neighbourhood strategy is illustrated in Fig. 1. The phases of neighbourhood strategy are described below.

1. *Identification* In neighbourhood strategy, first step is to identify neighbouring data points of cluster centres. To determine the position of neighbouring data points, Euclidean distance is adopted as evaluation measure. In this work, we have chosen five neighbouring data points of a cluster centre to evaluate the position of corresponding neighbourhood data point. So, five data points with minimum Euclidean distance are selected as neighbouring data points of a cluster centre. Let X_i represents *i*th cluster centre and $X_{i,\mathrm{neigh}}$ represents set of neighbouring data points of *i*th cluster centre. $X_{i,\mathrm{\ neigh}}$ can be described as $X_{i,\mathrm{neigh}} = \{X_{i,1}, X_{i,2}, \dots X_{i,5}\}$ where neigh = 1 to 5.
2. *Evaluation Phase* In evaluation phase, the neighbouring data points are used to determine the position of neighbourhood data item i.e., to produce single output using multiple inputs. In this phase, harmonic mean is used instead of arithmetic mean to compute the position of neighbourhood data items. The harmonic mean of neighbouring data points is calculated using Eq. 12.

$$X_{\mathrm{neigh}} = \text{Harmonic mean of } \left(X_{i,\mathrm{\ neigh}}\right) \tag{12}$$

The steps of BA-CN clustering algorithm that includes enhanced co-operative co-evaluation method and neighbourhood search strategy for addressing initial centroid selection and local optima issues are mentioned in Algorithm 3.



(a) Data items and current position in solution space

(b) Identification and selection of Neighboring data points

(c) Evaluation of new current position

Fig. 1 **a–c** Neighbourhood strategy procedures

Algorithm 3: Pseudo code of Bat algorithm with Enhanced cooperative co-evolution method and neighborhood strategy (BA-CN) method

```
Step 1:  Load the data sets and specify the number of clusters $K_i \in (i = 1, 2, \ldots, n)$
Step 2:  Select initial clusters positions using enhanced cooperative co-evolution strategy equations 8-11.
Step 3:  Initialize pulse rates $(r_i)$ loudness $(A_i)$and initial velocity $(v_i)$ equations 1-2.
Step 4:  Evaluate the objective function using equation 7 and assign items to the clusters with
         minimum objective function values
Step 5:  Apply neighbourhood strategy to find $X_{neigh}$ position using equation 12.
Step 6:  While (t < max number of iterations /* t = iteration number), do
Step 7:  Calculate pulse frequency $f_i^t$ and velocity $v_i^t$ using equations 3-4.
Step 8:  if (rand >$r_i$)
         Accept $X_n$ position as solution
         Else
Step 9:  Update the position using equations 5-6.
         Increase $r_i$ and reduce $A_i$
         Accept the new solution
         end if
Step 10: Recalculate the objective function and assign items with minimum objective function values
Step 11: Apply neighbourhood strategy
Step 12: Memorize the result and store in memory pool
         t=t+1;
Step 13: End while
Step 14: Global optimal solution with highly accurate value
```

7 Third improvement: elitist strategy

In clustering, another important aspect is convergence speed. The convergence rate depends on the searching pattern of the algorithm. To improve convergence speed, elitist strategy with certain enhancement is incorporated in bat algorithm. According to elitist strategy, best positions move from previous iteration to next iteration. In the proposed algorithm, elitist strategy is implemented in two phases- Evaluation Phase and Updating Phase.

Evaluation phase: In this phase, personal best and global best positions are computed using Eqs. 13,14. The comparison operator is used to calculate global best position X_{Gbest} and personal best position X_{Pbest}.

$$X_{\mathrm{Pbest}} = \min(\text{fitness value}) \tag{13}$$

$$X_{\mathrm{Gbest}} = \min\ (\text{distance value}) \tag{14}$$

The personal best (X_{Pbest}) is obtained using the fitness function as described in Eq. 15.

$$F\left(C_K\right) = \sum_{K\in 1}^{K} \frac{\mathrm{SSE}\left(C_K\right)}{\sum_{K=1}^{K} \mathrm{SSE}\left(C_K\right)} \tag{15}$$

where SSE denotes the sum of square error and C_K represents the Kth centroid object. After evaluation of fitness function minimum value is selected as personal best. In next step the global best (X_{Gbest}) is evaluated using Eq. 14, which is minimum value of distance function or objective function.

2. Updating phase: In this phase, the personal best and global best positions are compared with previous iteration values. If, current values are better than previous values, than positions are updated using Eqs. 16, 17. Otherwise, previous values are considered.

$$X_{\mathrm{Pbest}} = \begin{cases} X_{Pbest}^{t-1} = X_{pbest}^{t} & \text{fitness}(t) \leq \ \text{fitness}(t-1) \\ X_{Pbest}^{t} = X_{pbest}^{t-1} & \text{fitness(t)} \geq \text{fitness}(t-1) \end{cases} \tag{16}$$

$$X_{\mathrm{Gbest}} = \begin{cases} X_{\mathrm{Gbest}}^{t-1} = X_{\mathrm{Gbest}}^{t} & \text{sum}(t) \leq \ \text{sum}(t-1) \\ X_{\mathrm{Gbest}}^{t} = X_{\mathrm{Gbest}}^{t-1} & \text{sum}(t) \geq \text{sum}(t-1) \end{cases} \tag{17}$$

Moreover, to make better trade-off between exploration and exploitation mechanisms, basic frequency, velocity and search equations of bat algorithm are modified using Eqs. 18–21.

$$f_i^t = \frac{\min\left(X_{\mathrm{Gbest}}^t\right) + \left(\max\left(X_{\mathrm{Gbest}}^t\right) - \min\left(X_{\mathrm{Gbest}}^t\right)\right)\beta}{\max\left(X_{\mathrm{Pbest}}^t\right)} \tag{18}$$

$$v_i^t = v_i^{t-1} + \left(X_{\mathrm{Gbest}}^t - X_{\mathrm{Pbest}}^t\right)f_i^t \tag{19}$$

$$X_{\mathrm{new}} = X_{\mathrm{Gbest}}^t + \mathrm{randi}[-1, 1] \tag{20}$$

$$X_i^t = \begin{cases} X_{\mathrm{Gbest}} & \text{if rand} > r_i \\ X_{\mathrm{new}} + v_i^t & \text{otherwise} \end{cases} \tag{21}$$

where f_i^t represent frequency ith bat, v_i^t denotes the velocity ith bat, and X_i^t represents the position of ith bat. Where, $\min\left(X_{\mathrm{Gbest}}^t\right)$ and $\max\left(X_{\mathrm{Gbest}}^t\right)$ denote minimum and maximum values of sum function associated with X_{Gbest}^t position. $\max\left(X_{\mathrm{Pbest}}^t\right)$ represents the maximum value of fitness function associated with personal best position and β denotes a random value between [0, 1].

7.1 Proposed complete Bat algorithm

This subsection describes the complete bat algorithm (BA-CNE) that includes enhanced cooperative co-evolution concept, neighbourhood search strategy and modified elitism strategy. The working of proposed algorithm is divided into three phases- Initialization Phase, Evaluation and Assignment Phase and Update and Decision-making Phase.

1. *Initialization phase* In this phase, user defined and other algorithmic parameters are initialized. The dataset is loaded in memory. The number of centroids is defined and initial cluster centres are selected. In this work, a cooperative co-evolution method is developed to select initial centroid. The other parameters like loudness, emission rate and random variable are specified.
2. *Evaluation and assignment phase* The main work of this phase is to allocate data objects to different clusters. This allocation is done through objective function. In this work, Euclidean distance is taken as objective function. The data objects allocate to clusters using minimum Euclidean distance. Moreover, a neighbourhood strategy is adopted to overcome local optima situation. A limit operator is applied to determine the local optima situation. If, the candidate solution is not improved in predefined limit operator range, then it is assumed that algorithm traps in local optima. According to neighbourhood strategy, a new candidate solution is generated to overcome local optima problem.
3. *Update and decision-making phase* In this phase, positions of bats are updated using search mechanism of bat algorithm. The emission rate of bat algorithm is compared with random function. If random function provides less value than emission rate, then neighbourhood value is accepted. Otherwise, a new value is calculated using parameters variations i.e., loudness and emission rate. If, a termination criterion is met, then algorithm

stops it working and final solution is obtained. Otherwise, repeats phase 2–3.

The steps of BA-CNE clustering algorithm are mentioned in Algorithm 4.

Step 2 The Evaluation and Assignment Phase comprises of objective function, allocate the data respective clusters and fitness function.

Algorithm 4: Pseudo code of Complete Bat algorithm with Enhanced cooperative co-evolution, neighborhood and modified elitism strategy (BA-CNE) method

```
Step 1:  Load the data sets and specify the number of clusters $K_i \in (i = 1, 2, \ldots, n)$
Step 2:  Select initial cluster positions using enhanced cooperative co-evolution strategy equations 8-11
Step 3:  Initialize pulse rates ($r_i$) loudness ($A_i$)and initial velocity ($v_i$) equations 1-2
Step 4:  Evaluate the objective function using equation 7 and assign items to the clusters with
         minimum objective function values
Step 5:  Apply neighbourhood strategy to find $X_{neigh}$ position using equation 12.
Step 6:  Apply elitist strategy, calculate $X_{Pbest}$ and $X_{Gbest}$ equations 13-14.
Step 7:  While (t < max number of iterations /* t = iteration number), do
Step 8:  Calculate pulse frequency $f_i^t$ and velocity $v_i^t$ using equations 18-19.
Step 9:  if (rand >$r_i$)
         Accept $X_n$ position as solution
         Else
Step 10: Update the position using equations 20-21.
         Increase $r_i$and reduce $A_i$
         Accept the new solution
         end if
Step 11: Recalculate the objective function and assign items with minimum objective function values
Step 12: Update $X_{Pbest}$ and $X_{Gbest}$ using equations 16-17.
Step 13: Apply neighbourhood strategy
Step 14: Memorize the result and store in memory pool
         t=t+1;
Step 15: End while
Step 16: Global optimal solution with highly accurate value
```

7.2 Time complexity

The working procedure of proposed algorithm is summarized into three phases, initialization, evaluation and assignment and update phase. The complexity of each phase is calculated and combined to get the final output.

Step 1 The initialization phase starts with the selection of initial centroid positions. These positions are evaluated using Eqs. 8–11. According to these, a dataset is divided into M number of distinct parts. M is equal to number of clusters (K) i.e., M = K. Now, the population of bat algorithm is described in terms of number of bats. Hence, the time required by initialization phase of BA-CNE variant is given as O(number_ of_bats × number_of_attributes). Here, number_ of_bats indicates number of clusters (K) and number_of_attributes indicates the dimension (d) i.e., the length of cluster centre O(K × d).

(a) To compute the objective function and allocate data to respective cluster require $O\left(n^2 \times K \times d\right)$ time.
(b) In worst case scenario, the neighbourhood strategy can require $O(n \times K \times d)$ time.
(c) The fitness function evaluation requires $O(n \times K \times d)$ time.

Hence, the total complexity of Step 2 can be given as $O\left(\left(n^2 \times K \times d\right) + (n \times K \times d) + (n \times K \times d)\right) = O\left(n^2 \times K \times d\right)$.

Step 3 In evaluation and update phase, the position of individual bat i.e., cluster centre is updated. This updating requiresO(K × d) time. Further, the above process is repeated up to termination condition i.e., max_iter. Hence, the total time required in this phase is given as $O(K \times d \times \text{max\_iter})$

The overall complexity of the BA-CNE variant is the combination of all above steps and can be given as $O\left(n^2 \times K \times d \times \text{max\_iter}\right)$. Whereas,

Table 1 Characteristics of datasets used for experiment

Data sets	*K*	*D*	*N*	Description
Iris	3	4	150	Fisher's iris data
CMC	3	9	1473	Contraceptive method choice
Glass	6	9	214	Glass identification data
Wine	3	13	178	Wine data
Ionosphere	2	34	351	Radar observations
Control	6	60	600	Synthetic control chart
Vowel	6	3	871	Indian Telugu vowel
LD	2	6	345	Liver Disorder Data
WBC	2	9	683	Wisconsin Breast Cancer
Balance	3	4	625	Balance
Crude Oil	3	5	56	Crude Oil data
Thyroid	3	5	215	Thyroid

the space complexity of the proposed algorithm is $O(\text{number\_of\_attributes } \times \text{number\_of\_bats })$ i.e.$(d \times K)$.

8 Experimental results and parameter settings

This section describes the simulation results of proposed bat variants. These variants are BA, BA-C, BA-CN and BA-CNE. The performance of these bat algorithm variants is tested on several standard clustering datasets. These datasets are Iris, CMC, Glass, Wine, Ionosphere, Control, Vowel, Liver Disorder (LD), Wisconsin Breast Cancer (WBC), Balance, Crude Oil and Thyroid. These data sets are downloaded from UCI repository. Table 1 summarizes the description of these datasets. MATLAB 2016 environment is used to implement the bat variants. The performances of bat variants are evaluated using intra-cluster distance, standard deviation and accuracy parameters. The simulation results are compared with PSO [10], ABC [51], K-Means [41], MEBBC [58], H-KHA [59], CBPSO [64], ACO [68], IKH [74], PSO-BB-BC [75], ICMPKHM [76] and DE [77] clustering algorithms. The parameters setting of aforementioned algorithms are taken same as reported in corresponding literature. For every dataset, the algorithms run thirty times individually and each run consists of 200 iterations. The simulation results are evaluated using intra-cluster distance, accuracy and rand index parameters. The intra-cluster distance parameter can be measured in terms of minimum, average and maximum distances.

8.1 Simulation results and discussion

This subsection presents the simulation results of proposed bat variants (BA, BA-C, BA-CN, and BA-CNE) and other well-known clustering algorithms. The parameters setting of bat variants are presented in Table 2.

The well-known twelve benchmark clustering datasets are considered to compute the simulation results of BA

Table 2 Parameters setting of bat variants

Bat algorithm (BA)		BA-C variant	
Population	$K \times d$	Population	$K \times d$
Pulse rates (r_i)	0.5	Pulse rates (r_i)	0.5
Loudness (A_i)	$\in (0.1, 0.9)$	Loudness (A_i)	$\in (0.1, 0.9)$
Initial velocity (v_i)	0.1	Initial Velocity (v_i)	0.1
		Number of partitions	*K*
		Population size (*T*)	No. of data Instance
BA-CN variant		**BA-CNE variant**	
Population	$K \times d$	Population	$K \times d$
Pulse rates (r_i)	0.5	Pulse rates (r_i)	0.5
Loudness (A_i)	$\in (0.1, 0.9)$	Loudness (A_i)	$\in (0.1, 0.9)$
Initial Velocity (v_i)	0.1	Initial Velocity (v_i)	0.1
Number of partitions	K	Number of partitions	*K*
Neighbourhood structure	5	Randi	[− 1, 1]
Population size (*T*)	No. of data instance	β	∈ (0, 1)
		Neighbourhood structure	5
		Population size (*T*)	No. of data Instance

Table 3 Comparison of intra-cluster distance parameter using different Bat variants

Dataset	Intra cluster distance	BA variants			
		BA	BA-C	BA-CN	BA-CNE
Iris	Max	2.80E+02	2.30E+02	2.27E+02	2.21E+02
	Avg	1.84E+02	1.53E+02	1.36E+02	9.16E+01
	Min	1.50E+02	1.26E+02	1.06E+02	7.01E+01
	Std	3.95E+01	3.82E+01	3.76E+01	3.12E+01
CMC	Max	1.05E+04	8.69E+03	7.27E+03	7.27E+03
	Avg	6.77E+03	4.76E+03	2.79E+03	2.79E+03
	Min	5.90E+03	4.43E+03	2.38E+03	2.38E+03
	Std	9.37E+02	5.83E+01	3.67E+00	2.39E+00
Glass	Max	2.01E+02	1.38E+02	1.38E+02	1.38E+02
	Avg	1.54E+02	1.31E+02	1.17E+02	1.01E+02
	Min	1.46E+02	1.29E+02	1.11E+02	9.91E+01
	Std	1.13E+01	8.41E+00	4.29+00	1.98E+00
Wine	Max	2.31E+04	2.02E+04	2.02E+04	2.02E+04
	Avg	1.38E+04	1.20E+04	9.41E+03	8.05E+03
	Min	1.29E+04	1.08E+04	8.04E+03	6.51E+03
	Std	1.94E+01	1.75E+01	1.66E+01	1.54E+01
Ionosphere	Max	3.91E+03	3.76E+03	3.83E+03	8.60E+02
	Avg	3.03E+03	2.64E+03	2.33E+03	7.50E+02
	Min	2.78E+03	2.33E+03	2.06E+03	1.70E+02
	Std	4.16E+02	4.02E+02	3.64E+02	3.53E+02
Control	Max	4.09E+04	3.78E+04	3.78E+04	3.78E+04
	Avg	3.30E+04	2.98E+04	2.68E+04	2.39E+04
	Min	3.16E+04	2.90E+04	2.62E+04	2.26E+04
	Std	2.10E+03	1.87E+03	1.51E+02	1.24E+02
Vowel	Max	7.50E+04	6.42E+04	5.89E+04	6.42E+04
	Avg	5.12E+04	4.72E+04	3.46E+04	2.56E+04
	Min	4.66E+04	4.21E+04	2.87E+04	2.27E+04
	Std	5.83E+03	4.83E+03	3.98E+03	4.08E+02
LD	Max	2.77E+02	2.63E+02	2.63E+02	2.67E+02
	Avg	2.49E+02	2.45E+02	2.36E+02	2.31E+02
	Min	2.45E+02	2.40E+02	2.34E+02	2.29E+02
	Std	2.07E+01	1.97E+01	1.52E+01	1.67E+01
WBC	Max	3.55E+03	3.26E+03	3.23E+03	3.21E+03
	Avg	3.21E+03	3.05E+03	2.96E+03	2.89E+03
	Min	3.16E+03	2.98E+03	2.92E+03	2.80E+03
	Std	2.91E+02	2.51E+02	1.98E+02	1.30E+02
Balance	Max	5.24E+04	5.19E+04	5.18E+04	5.09E+04
	Avg	5.20E+04	5.16E+04	5.12E+04	5.01E+04
	Min	5.20E+04	5.16E+04	5.11E+04	5.00E+04
	Std	4.96E+03	3.67E+02	3.26E+02	3.59E+02
Crude oil	Max	3.17E+02	3.09E+02	3.09E+02	3.00E+02
	Avg	2.89E+02	2.86E+02	2.75E+02	2.73E+02
	Min	2.85E+02	2.83E+02	2.74E+02	2.72E+02
	Std	2.06E+01	1.93E+01	1.76E+01	1.61E+01
Thyroid	Max	2.97E+02	2.88E+02	2.92E+02	2.69E+02
	Avg	2.72E+02	2.68E+02	2.54E+02	2.51E+02
	Min	2.66E+02	2.61E+02	2.48E+02	2.47E+02
	Std	5.42E+01	3.32E+01	2.28E+01	1.32E+01

Table 4 Comparison of accuracy parameter of different BA variants

Dataset	Accuracy	BA variants			
		BA	BA-C	BA-CN	BA-CNE
Iris	Max	94.6667	90.4217	93.2364	95.0003
	Avg	87.9467	88.8067	90.5033	93.0067
	Min	61.3333	64.6667	60.6667	85.3333
CMC	Max	38.0381	41.093	45.4175	58.146
	Avg	34.0832	35.5889	37.6212	51.2979
	Min	27.649	33.6728	40.9369	46.7882
Glass	Max	24.8976	40.2645	39.0562	64.9962
	Avg	24.9626	37.5636	38.7623	59.1706
	Min	22.8411	38.3916	37.9721	53.3831
Wine	Max	69.7639	73.0337	71.7628	80.9101
	Avg	64.3989	66.0562	65.486	76.1348
	Min	61.944	63.9441	58.8764	65.0562
Ionosphere	Max	52.3647	54.9288	63.5328	78.5026
	Avg	51.4829	51.5239	60.0883	73.941
	Min	37.3504	41.6524	55.641	68.1026
Control	Max	59.8333	83.3333	83.8333	86.1667
	Avg	52.505	61.2983	66.8408	78.3008
	Min	39.6754	65.1667	64.3343	66.8333
Vowel	Max	51.7796	53.6395	58.3467	69.8393
	Avg	34.3117	51.1452	57.2101	67.1119
	Min	31.2285	49.0471	55.0287	64.3915
LD	Max	49.5261	50.8363	55.5128	57.4325
	Avg	48.2543	49.1021	53.0715	54.0211
	Min	46.1589	49.0009	51.8024	52.0299
WBC	Max	91.7096	91.3965	93.0821	96.6952
	Avg	84.3577	85.1452	89.2109	96.6104
	Min	81.2158	81.0471	88.0271	93.5736
Balance	Max	81.7796	83.6390	89.0045	89.9745
	Avg	74.3782	75.1025	86.7581	88.9278
	Min	71.2518	72.0488	82.1032	82.8536
Crude oil	Max	54.8473	61.6725	67.3720	75.3211
	Avg	51.3117	53.6952	63.1721	70.0457
	Min	41.2529	49.9710	59.8751	62.1074
Thyroid	Max	64.5470	63.4219	72.3720	75.0918
	Avg	61.1009	61.6280	63.8294	71.9825
	Min	58.6548	60.3605	61.5581	68.9939

variants. The simulation results are described in terms of intra-cluster distance, standard deviation (Std), accuracy and rand index parameters. Further, the intra-cluster distance parameter is defined using maximum (Max), minimum (Min) and average (Avg.) distances. Table 3 presents the simulation results of all BA variants (BA, BA-C, BA-CN and BA-CNE) using intra-cluster distance and standard deviation parameters. Simulation results claimed that BA-CNE variant obtains minimum average intra-cluster distance among all BA variants. It is also found that BA-CNE variant achieves minimum standard deviation (Std) values among all other BA variants. for BA-CNE variants. These parameters reflect the effectiveness of BA-CNE variant for solving clustering problems. The minimum value of standard deviation indicates that data objects are strictly bound with cluster centres and clusters are compact in nature.

The comparison of accuracy parameter for different BA variants is illustrated in Table 4. The results are measured in terms of maximum, average and minimum accuracy. It is seen that BA-CNE variant provides the better accuracy results than BA, BA-C and BA-CN variants using all datasets. Further, BA variant provides less accurate performance than other BA variants. It is noticed that improvements enhance the performances of BA variants (BA-C, BA-CN and BA-CNE) considerably.

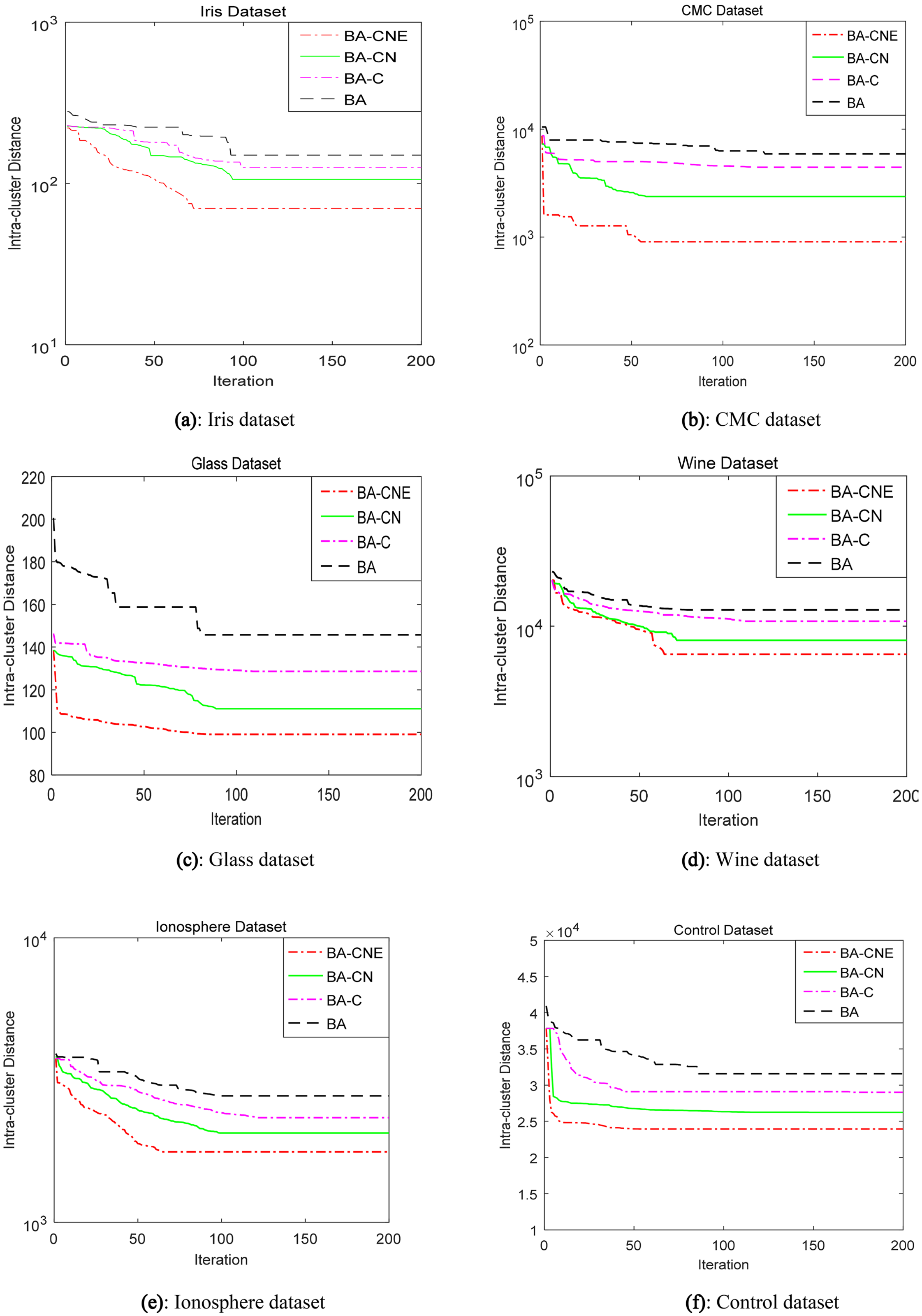
Table 5 illustrates the simulation results of all bat varinats using rand index parameter. It is also an important parameter for investigating the perofromance of clustering algorithms apart from inta cluster distance and accuracy. It can be interpreated as number of times data objects belong to the same cluster. It is seen that BA-CNE variant achieves better rand index rate as comapred to BA, BA-C and BA-CN variants using all datasets. On the analysis of intra-cluster distance, accuracy and rand index parameter, it is said that BA-CNE variant achieves effective clustering results for all benchmark clustering datasets as compared to BA, BA-C, BA-CN. It also noticed that improvements in BA considerably enhance the quality of clustering results such as BA-C, BA-CN and BA-CNE performs better than BA. It's also revealed that BA variants successfully addressed the shortcomings associated with BA. Furthermore, BA-CNE variant is best performing among BA, BA-C and BA-CN variants. Hence, the simulation results of BA-CNE variant are also compared with state of art existing clustering algorithms.

Table 5 Comprasion of different BA variants using rand index parameter

Dataset/algorithm	BA	BA-C	BA-CN	BA-CNE
Iris	0.684	0.703	0.724	0.736
CMC	0.030	0.032	0.035	0.038
Glass	0.381	0.394	0.419	0.428
Wine	0.378	0.384	0.391	0.395
Ionosphere	0.316	0.319	0.331	0.339
Control	0.765	0.781	0.789	0.810
Vowel	0.848	0.852	0.853	0.858
LD	0.451	0.456	0.467	0.473
WBC	0.239	0.243	0.255	0.258
Balance	0.521	0.524	0.531	0.535
Crude oil	0.047	0.061	0.069	0.074
Thyroid	0.354	0.358	0.372	0.383

Convergence behaviour of BA variants is shown in Fig. 2(a–l). The intra-cluster distance parameter is used to illustrate the convergence of the BA-CNE, BA-CN, BA-C and BA variants. It is seen that improvements in BA improves the convergence rate. Further, it is observed that BA-CNE variant converges on minimum values in comparison to other BA variants. Hence, it is stated that BA-CNE variant outperforms than other BA variants. Further, the performance of BA-CNE variant is also compared with well-known clustering algorithms. Table 6 demonstrates the simulation results of best performing BA variant (BA-CNE) and other well-known clustering algorithm using average intra-cluster distance parameter. It is observed that BA-CNE variant obtains minimum intra-cluster distance than H-KHA, MEBBC, IKH, ICMPKHM, PSO-BB-BC, CBPSO, ACO, ABC, DE and K-means clustering algorithms with most of datasets except balance and crude oil. ACO algorithm achieves lowest intra-cluster distance with crude oil dataset, while, K-mean is best performing algorithm with balance dataset. Moreover, standard deviation parameter also reflects the significance of BA-CNE variant as compered other clustering algorithms. This parameter illustrates the dispersion of data objects within a cluster. The minimum value of standard deviation parameter claimed that data objects are tightly bound with clusters.

The results of accuracy parameter of BA-CNE variant and other cluster algorithms are reported in Table 7. Its revealed that BA-CNE variant provides more accurate results than other clustering algorithms except glass and crude oil datasets. For glass dataset, PSO-BB-BC algorithm achieves higher accuracy among all clustering algorithms. It is also observed that ICMPKHM and IKH algorithms having more accuracy rate than BA-CNE variant. Table 8 presents the simulation results of BA-CNE variant and other clustering algorithms using rand index parameter. Rand index is also an important parameter to establish the performance of new algorithm. It is seen that BA-CNE variant obtains better results as compared to other clustering algorithms. Hence, it is stated that BA-CNE is one of efficient and effective algorithm for solving clustering problems.



(a): Iris dataset

(b): CMC dataset

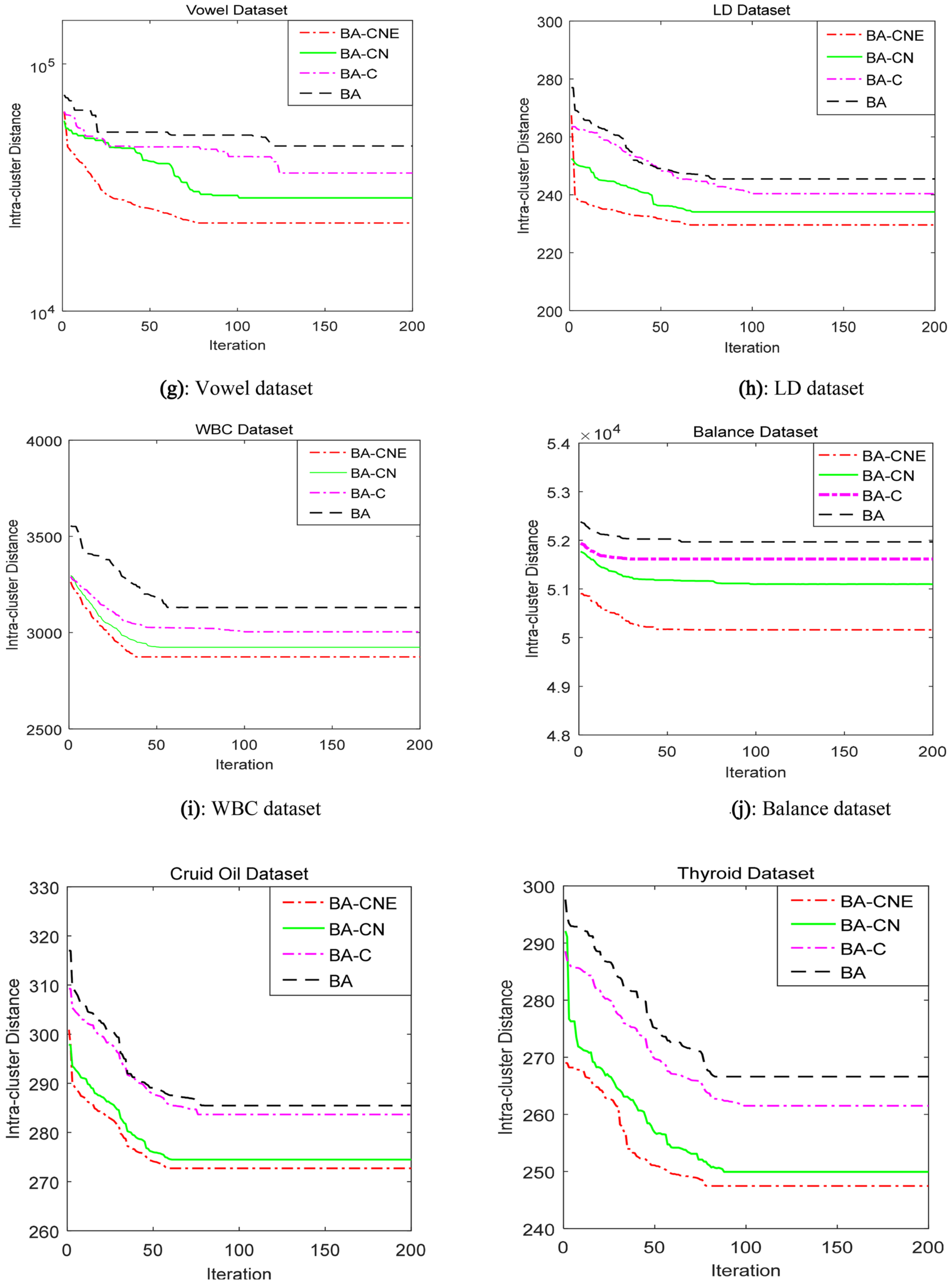
(c): Glass dataset

(d): Wine dataset

(e): Ionosphere dataset

(f): Control dataset

Fig. 2 **a–l** Convergence behaviour of BA variants using intra-cluster distance parameter



(g): Vowel dataset

(h): LD dataset

(i): WBC dataset

.**(j)**: Balance dataset

(k): Crude oil dataset

(l): Thyroid dataset

Fig. 2 (continued)

Table 6 Comparison of the simulation results of BA-CNE variant and other well-known clustering algorithms average intra-cluster distance and standard deviation (in brackets)

Dataset	BA-CNE	H-KHA	MEBBC	IKH	ICMPKHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K-means
Iris	9.16E+01 (3.12E+01)	9.65E+01 (4.45+01)	9.68E+01 (2.10E-01)	9.67E+01 (9.8E–06)	9.66E+01 (1.06E-02)	9.67E+01 (2.16E-03)	9.69E+01 (3.18E-01)	1.00E+02 (1.42E+00)	1.05E+02 (3.59E+00)	1.16E+02 (5.00E+00)	9.20E+01 (2.67E+01)
CMC	2.79E+03 (2.39E+00)	5.60E+03 (3.06+02)	5.53E+03 (2.60E-01)	5.69E+03 (7.98E-03)	5.70E+03 (8.12E-01)	5.53E+03 (4.98E-01)	5.53E+03 (3.35E+00)	6.47E+03 (1.23E+02)	5.94E+03 (1.31E+02)	5.95E+03 (8.69E+01)	2.37E+04 (6.76E+00)
Glass	1.01E+02 (1.98E+00)	2.16E+02 (1.07+01)	2.27E+02 (1.01E+01)	2.23E+02 (2.74E+00)	1.99E+02 (1.27E-01)	2.16E+02 (1.99E+00)	2.13E+02 (2.09E+00)	2.26E+02 (3.42E+00)	3.31E+02 (1.01E+01)	3.56E+02 (1.15E+01)	3.79E+02 (7.05E+01)
Wine	8.05E+03 (1.54E+01)	1.64E+04 (2.3+00)	1.63E+04 (6.90E-01)	1.63E+04 (7.07E-01)	1.63E+04 (4.80E-01)	1.63E+04 (1.56E+00)	1.63E+04 (9.41E-01)	1.63E+04 (3.78E+01)	1.71E+04 (4.63E+02)	1.65E+04 (5.60E+01)	2.42E+06 (1.06E+05)
Ionosphere	7.5E+02 (3.53E+02)	8.95E+02 (4.03E+02)	1.95E+03 (3.67E+02)	9.34E+02 (3.93E+02)	2.05E+03 (3.69E+02)	2.45E+03 (3.62+02)	8.08E+02 (1.35E+01)	8.13E+02 (4.54E+00)	1.03E+03 (2.65E+01)	1.04E+03 (1.68E+01)	2.42E+03 (4.55E-13)
Control	2.39E+04 (1.24E+02)	2.51E+04 (1.28E+02)	3.01E+04 (1.16E+03)	2.42E+04 (1.68E+02)	2.92E+04 (1.99E+02)	5.04E+04 (7.08E+02)	2.54E+04 (2.88E+02)	2.42E+04 (1.68E+02)	5.01E+04 (1.26E+03)	5.14E+04 (9.08E+02)	1.01E+06 (5.05E+04)
Vowel	2.56E+04 (4.08E+02)	3.52E+04 (1.25E+03)	1.49E+05 (4.52E+02)	1.59E+05 (1.74E+03)	1.49E+05 (1.23E+03)	1.50E+05 (7.69E+02)	1.50E+05 (7.76E+02)	1.94E+05 (2.63E+03)	1.69E+05 (4.73E+03)	1.75E+05 (2.97E+03)	3.18E+07 (1.50E+06)
LD	2.31E+02 (1.67E+01)	3.12E+03 (2.00E+03)	1.32E+04 (6.52E+02)	3.11E+03 (2.01E+03)	3.09E+03 (1.95E+02)	9.98E+03 (4.39E+02)	1.00E+04 (4.40E+02)	2.35E+02 (1.75E+01)	9.85E+03 (8.20E+02)	1.15E+04 (2.07E+03)	1.17E+04 (6.68E+02)
WBC	2.89E+03 (1.30E+02)	2.99E+03 (0.00E+00)	3.01E+03 (0.80E+01)	2.96E+03 (1.26E-03)	3.02E+03 (3.81E-01)	2.96E+03 (2.30E-02)	2.97E+03 (1.82E+00)	3.37E+03 (4.17E+01)	3.50E+03 (2.12E+02)	3.73E+03 (1.84E+02)	1.93E+04 (5.14E-12)
Balance	5.01E+04 (3.59E+02)	6.78E+04 (9.32E+02)	5.83E+04 (3.74E+03)	6.01E+04 (8.12E+03)	6.26E+04 (3.32E+03)	6.19E+04 (7.32E+02)	6.20E+04 (2.80E+02)	5.99E+04 (7.62E+03)	6.63E+04 (4.80E+03)	6.87E+04 (2.10E+04)	2.01E+04 (9.28E+03)
Crude oil	2.73E+02 (1.61E+01)	2.82E+02 (1.92E+01)	2.76E+02 (1.32E+01)	2.80E+02 (2.00E+01)	2.81E+02 (1.81E+01)	2.82E+02 (7.78E+00)	2.78E+02 (1.07E-01)	2.54E+02 (7.09E+00)	2.77E+02 (1.1E+01)	3.72E+02 (2.1E+01)	2.87E+02 (2.54E+01)
Thyroid	2.51E+02 (1.32E+01)	1.80E+03 (1.31E+01)	2.56E+02 (1.76E+01)	1.77E+03 (1.39E+01)	1.39E+03 (1.16E+00)	2.82E+02 (7.78E+00)	1.86E+03 (9.97E+00)	1.99E+03 (3.09E+01)	1.96E+03 (2.1E+02)	2.96E+02 (2.06E+01)	2.39E+03 (2.46E+02)

Table 7 Comparison of average accuracy parameter of BA-CNE variant and other well-known clustering algorithms

Dataset/ Algorithm	BA-CNE	H-KHA	MEBBC	IKH	ICMPKHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K-means
Iris	93.00	89.24	90.02	89.87	92.44	90.52	90.79	72.87	89.03	88.37	82.33
CMC	51.29	47.45	46.58	46.63	46.69	47.61	39.58	36.89	40.06	39.58	39.69
Glass	59.17	58.89	58.73	68.39	69.06	69.52	51.92	37.36	48.43	48.48	51.87
Wine	76.13	75.00	72.63	72.37	72.88	73.58	71.31	59.21	70.34	71.10	67.53
Ionosphere	73.94	67.92	65.87	58.93	60.23	68.19	69.11	60.70	64.37	63.02	71.23
Control	78.30	74.58	72.32	69.56	71.87	72.31	67.06	39.53	35.56	39.26	59.72
Vowel	67.11	66.98	59.21	61.76	59.24	60.18	51.72	31.69	56.31	53.41	51.16
LD	54.02	51.91	49.86	52.96	52.15	52.17	53.65	52.89	49.89	52.01	52.16
WBC	96.61	61.01	88.05	72.07	46.69	95.50	96.48	77.92	95.01	94.68	96.5
Balance	88.92	75.42	68.42	76.42	77.42	88.76	76.62	74.28	76.67	74.96	84.99
Crude oil	70.04	65.77	68.21	69.32	66.87	55.02	76.57	59.12	56.78	66.52	65.45
Thyroid	71.98	65.40	65.22	66.91	66.82	56.8312	71.21	64.87	64.39	65.76	63.76

Table 8 Comparison of Random Index parameter of BA-CNE variant and other well-known clustering algorithms

Dataset/algorithm	BA-CNE	H-KHA	MEBBC	IKH	ICMP KHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K-means
Iris	0.736	0.730	0.720	0.721	0.730	0.726	0.729	0.610	0.630	0.562	0.616
CMC	0.038	0.030	0.036	0.038	0.032	0.038	0.036	0.029	0.031	0.025	0.026
Glass	0.428	0.366	0.352	0.306	0.262	0.420	0.256	0.404	0.258	0.417	0.406
Wine	0.395	0.380	0.368	0.371	0.378	0.370	0.371	0.363	0.372	0.321	0.343
Ionosphere	0.339	0.320	0.314	0.319	0.322	0.316	0.310	0.311	0.320	0.323	0.318
Control	0.810	0.750	0.783	0.729	0.746	0.740	0.741	0.758	0.766	0.723	0.728
Vowel	0.858	0.850	0.883	0.851	0.846	0.852	0.863	0.853	0.856	0.848	0.837
LD	0.473	0.464	0.482	0.469	0.467	0.465	0.468	0.460	0.462	0.441	0.456
WBC	0.258	0.243	0.246	0.232	0.249	0.239	0.241	0.226	0.250	0.222	0.217
Balance	0.535	0.531	0.532	0.536	0.536	0.534	0.534	0.537	0.540	0.541	0.546
Crude oil	0.074	0.022	0.072	0.037	0.032	0.026	0.044	0.053	0.047	0.025	0.018
Thyroid	0.383	0.332	0.350	0.370	0.330	0.361	0.383	0.360	0.359	0.321	0.310

8.2 Statistical results and discussion

This subsection illustrates the statistical results of BA-CNE variant and other clustering algorithms. A statistical test is carried out to confirm the existence of the newly proposed algorithm. So, in this study, two statistical tests are selected to confirm the existence of BA-CNE algorithm in the field of clustering. These tests are Friedman and Quade. The aim of these tests is to prove that the proposed algorithm is statistically better than rest of algorithms. So, a significant difference is occurred between the performance of BA-CNE algorithm and rest of clustering algorithms. The statistical tests are applied on both of performance measures i.e., intra-cluster distance and accuracy.

- *Using intra-cluster distance parameter* This sub section presents the statistical test results using intra-cluster distance parameter. Table 9 illustrates the results of the Friedman statistical test. For, Friedman test, two hypotheses are designed; hypothesis (H_0) denotes the population's equivalency, while, hypothesis (H_1) stands for inequivalent population at the significance level of 0.05.

  Quade test is an advanced nonparametric test that gives the better result than Friedman test. This test gives weightage to the size of datasets. Tables 10, 11 summarize the results of Quade test using intra-cluster distance parameter. It is observed that the proposed BA-CNE variant obtains first rank in most cases accept balance and crude oil datasets. Further, it is also seen that K- means algorithms exhibits worst performance using intra-clus-

Table 9 Results of Friedman test based on avg. intra-cluster distance for BA-CNE and other clustering algorithms

Datasets	Clustering algorithms										
	BA-CNE	H-KHA	MEBBC	IKH	ICM PKHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K-means
Iris	**1**	3	7	5.5	4	5.5	8	9	10	11	2
CMC	**1**	5	3	6	7	3	3	10	8	9	11
Glass	**1**	4.5	8	6	2	4.5	3	7	9	10	11
Wine	**1**	8	4.5	4.5	4.5	4.5	4.5	4.5	10	9	11
Ionosphere	**1**	4	8	5	9	11	2	3	6	7	10
Control	**1**	4	7	2.5	6	9	5	2.5	8	10	11
Vowel	**1**	2	3.5	7	3.5	5.5	5.5	10	8	9	11
LD	**1**	5	11	4	3	7	8	2	6	9	10
WBC	**1**	5	6	2.5	7	2.5	4	8	9	10	11
Balance	2	10	3	5	8	6	7	4	9	11	**1**
Crude oil	2	8.5	3	6	7	8.5	5	**1**	4	11	10
Thyroid	**1**	7	2	6	5	3	8	10	9	4	11
Sum	**14**	66	66	60	66	70	63	71	96	110	110
Average ranking	1.17	5.5	5.5	5	5.5	5.83	5.25	5.92	8	9.17	9.17
Number of observations: 132					Number of problems: 12			Number of algorithms: 11			
Sum of squares of ranks: 22,409.5					Correction factor: 4752			Friedman test statistic: 55.250578			
Degree of freedom: 10					*p*-value:2.835e-8			Critical value: 18.307038			

The bold values repersent the best value obtained by an algorithm among all other algorithms for a dataset

Table 10 Relative size of observation using Quade test for BA-CNE and other clustering algorithms (using avg. intra-cluster distance)

Datasets	Clustering algorithms										
	BA-CNE	H-KHA	MEBBC	IKH	ICMP KHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K− means
Iris	− 5	− 3	1	− 0.5	− 2	− 0.5	2	3	4	5	− 4
CMC	− 40	− 8	− 24	0	8	− 24	− 24	32	16	24	40
Glass	− 15	− 4.5	6	0	− 12	− 4.5	− 9	3	9	12	15
Wine	− 55	22	− 16.5	− 16.5	− 16.5	− 16.5	− 16.5	− 16.5	44	33	55
Ionosphere	− 20	− 8	8	− 4	12	20	− 16	− 12	0	4	16
Control	− 50	− 20	10	− 35	0	30	− 10	− 35	20	40	50
Vowel	− 60	− 48	− 30	12	− 30	− 6	− 6	48	24	36	60
LD	− 30	− 6	30	− 12	− 18	6	12	− 24	0	18	24
WBC	− 35	− 7	0	− 24.5	7	− 24.5	− 14	14	21	28	35
Balance	− 36	36	− 27	− 9	18	0	9	− 18	27	45	− 45
Crude oil	− 8	5	− 6	0	2	5	− 2	− 10	− 4	10	8
Thyroid	− 25	5	− 20	0	− 5	− 15	10	20	15	− 10	25
Relative size of observation	− 379	− 36.5	− 68.5	− 89.5	− 36.5	− 30	− 64.5	4.5	176	245	279

ter distance parameter. The p value for Quade test is 6.978e-9. Hence, the hypothesis is strongly rejected at the significance level 0.05. So, substantial difference is occurred between the performance of proposed BA-CNE variant and other clustering algorithms being compared.

- *Using accuracy parameter* Table 12 demonstrates the results of the Friedman statistical test using BA-CNE, H-KHA, MEBBC, IKH, ICMPKHM, PSO-BB-BC, CBPSO, ACO, ABC, DE and K-means using average accuracy parameter. It is stated that BA-CNE variant obtains first rank in comparison accept glass and crude oil datasets. Moreover, ACO algorithm exhibits poor performance using average accuracy parameter. The significance level is set to 0.05 for Friedman test. The p-value obtained using Friedman test is 1.548e-7 which is minimum than significance level. Hence, the hypothesis is rejected at the level of 0.05 which shows the significant difference between the performances of proposed BA-CNE and other clustering algorithms. The computational value of Friedman test is 51.303524.

The results of Quade statistical test using average accuracy parameter are reported in Tables 13, 14. These results indicate that the proposed BA-CNE algorithm is an efficient algorithm for solving clustering problems. The p-value for Quade test is 1.29e-6which much smaller than significant level 0.05. The critical value for Quade test is 1.9178. Overall, these statistical tests conclude that the substantial difference is exhibited between the performance of proposed BA-CNE and another clustering algorithm. These statistical tests validate the performance of the proposed BA-CNE algorithm statistically. Hence, the proposed BA-CNE algorithm is statistically as well as experimentally better than other clustering algorithms being compared.

Table 11 Results of Quade test based on avg. intra-cluster distance for BA-CNE and other clustering algorithms

Datasets	Clustering Algorithms											Relative Size of Datasets
	BA-CNE	H-KHA	MEBBC	IKH	ICMP KHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K-means	
Iris	**1**	3	7	5.5	4	5.5	8	9	10	11	2	1
CMC	**1**	5	3	6	7	3	3	10	8	9	11	8
Glass	**1**	4.5	8	6	2	4.5	3	7	9	10	11	3
Wine	**1**	8	4.5	4.5	4.5	4.5	4.5	4.5	10	9	11	11
Ionosphere	**1**	4	8	5	9	11	2	3	6	7	10	4
Control	**1**	4	7	2.5	6	9	5	2.5	8	10	11	10
Vowel	**1**	2	3.5	7	3.5	5.5	5.5	10	8	9	11	12
LD	**1**	5	11	4	3	7	8	2	6	9	10	6
WBC	**1**	5	6	2.5	7	2.5	4	8	9	10	11	7
Balance	2	10	3	5	8	6	7	4	9	11	**1**	9
Crude oil	2	8.5	3	6	7	8.5	5	**1**	4	11	10	2
Thyroid	**1**	7	2	6	5	3	8	10	9	4	11	5
Sum	**14**	66	66	60	66	70	63	71	96	110	110	78
Average ranking	**1.17**	5.5	5.5	5	5.5	5.83	5.25	5.92	8	9.17	9.17	6.5
	Quade Statistics = 7.392221					*p*-value = 6.978e-9				Critical value = 1.917827		

The bold values repersent the best value obtained by an algorithm among all other algorithms for a dataset

Table 12 Results of Friedman test based on avg. accuracy for BA-CNE and other clustering algorithms

Datasets	Clustering algorithms										
	BA-CNE	H-KHA	MEBBC	IKH	ICM PKHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K-means
Iris	**1**	7	5	6	2	4	3	11	8	9	10
CMC	**1**	3	6	5	4	2	9.5	11	7	9.5	8
Glass	4	5	6	3	2	**1**	7	11	10	9	8
Wine	**1**	2	5	6	4	3	7	11	9	8	10
Ionosphere	**1**	5	6	11	10	4	3	9	7	8	2
Control	**1**	2	3	6	5	4	7	9	11	10	8
Vowel	**1**	2	6	3	5	4	9	11	7	8	10
LD	**1**	9	11	3	7	5	2	4	10	8	6
WBC	**1**	10	7	9	11	4	3	8	5	6	2
Balance	**1**	8	11	7	4	2	6	10	5	9	3
Crude oil	2	7	4	3	5	11	**1**	9	10	6	8
Thyroid	**1**	6	7	3	4	11	2	8	9	5	10
Sum	**16**	66	77	65	63	55	59.5	112	98	95.5	85
Average ranking	**1.33**	5.5	4.62	5.42	5.25	4.58	4.96	9.33	8.17	7.96	7.08
Number of observations: 132				Number of problems: 12				Number of algorithms: 11			
Sum of squares of ranks: 63,793.5				Correction factor: 4752				Friedman test statistic: 51.303524			
Degree of freedom: 10				*p*-value: 1.548e−7				Critical value: 18.307038			

The bold values repersent the best value obtained by an algorithm among all other algorithms for a dataset

Table 13 Relative size of observation using Quade test for BA-CNE and other clustering algorithms (using avg. accuracy distance)

Datasets	Clustering algorithms										
	BA-CNE	H-KHA	MEBBC	IKH	ICMPKHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K-means
Iris	– 30	6	– 6	0	– 24	– 12	– 18	30	12	18	24
CMC	– 10	– 6	0	– 2	– 4	– 8	7	10	2	7	4
Glass	– 18	– 9	0	– 27	– 36	– 45	9	45	36	27	18
Wine	– 25	– 20	– 5	0	– 10	– 15	5	25	15	10	20
Ionosphere	– 15	– 3	0	15	12	– 6	– 9	9	3	6	– 12
Control	– 55	– 44	– 33	0	– 11	– 22	11	33	55	44	22
Vowel	– 50	– 40	0	– 30	– 10	– 20	30	50	10	20	40
LD	– 5	3	5	– 3	1	– 1	– 4	– 2	4	2	0
WBC	– 60	48	12	36	60	– 24	– 36	24	– 12	0	– 48
Balance	– 35	14	35	7	– 14	– 28	0	28	– 7	21	– 21
Crude oil	– 32	8	– 16	– 24	– 8	40	– 40	24	32	0	16
Thyroid	– 20	0	4	– 12	– 8	20	– 16	8	12	– 4	16
Relative size of observation	– 355	– 43	– 4	– 40	– 52	– 121	– 61	284	162	151	79

9 Conclusion

In this paper, three variants of bat algorithm are developed for effectively addressing the clustering problems. These variants of bat algorithm are BA-C, BA-CN and BA-CNE. The BA-C variant deals with the initial population selection problem of bat algorithm. To resolve this issue, a cooperative co-evolution collaborator selection method is introduced in bat algorithm, called BA-C variant. The BA-CN variant deals with the initial population selection and local optima problems of bat algorithm. In BA-CNE variant, an elitist strategy is integrated in bat algorithm to make better balance between local and global searches including aforementioned improvements. Moreover, the solution search equations of bat algorithm are also modified to determine optimal solution in last iterations. The performance of these BA variants is evaluated using twelve benchmark clustering datasets. It is observed that BA-CNE variant gives better clustering results than BA, BA-C and BA-CN variants. Further, the simulation results of BA-CNE variant are compared with H-KHA, MEBBC, IKH, ICMPKHM, PSO-BB-BC, CBPSO, ACO, ABC, DE and K-means clustering algorithms. It is stated that BA-CNE variant provides better results than aforementioned clustering algorithms. Statistical tests are also performed in this study. These tests also validate the performance of BA-CNE variant. Finally, it is concluded that BA-CNE variant is one of effective and efficient algorithm to solve partitional clustering problems.

Table 14 Results of Quade test based on avg. Accuracy parameter for BA-CNE and other clustering algorithms

Datasets	Clustering algorithms											Relative size of observations
	BA-CNE	H-KHA	ME BBC	IKH	ICM PKHM	PSO-BB-BC	CBPSO	ACO	ABC	DE	K-means	
Iris	1	7	5	6	2	4	3	11	8	9	10	6
CMC	1	3	6	5	4	2	9.5	11	7	9.5	8	2
Glass	4	5	6	3	2	1	7	11	10	9	8	9
Wine	1	2	5	6	4	3	7	11	9	8	10	5
Ionosphere	1	5	6	11	10	4	3	9	7	8	2	3
Control	1	2	3	6	5	4	7	9	11	10	8	11
Vowel	1	2	6	3	5	4	9	11	7	8	10	10
LD	1	9	11	3	7	5	2	4	10	8	6	1
WBC	1	10	7	9	11	4	3	8	5	6	2	12
Balance	1	8	11	7	4	2	6	10	5	9	3	7
Crude oil	2	7	4	3	5	11	1	9	10	6	8	8
Thyroid	1	6	7	3	4	11	2	8	9	5	10	4
Sum	16	66	77	65	63	55	59.5	112	98	96	85	78
Average Ranking	1.33	5.5	6.42	5.42	5.25	4.58	4.96	9.33	8.17	8	7.08	6.5
Quade test statistic = 5.514609					*p*-value = 1.29e-6			Critical value = 1.917827				

References

1. Aggarwal CC, Chandan KR (eds) (2013) Data clustering: algorithms and applications. CRC Press, Hoboken
2. Xu R, Wunsch D (2005) Survey of clustering algorithms. IEEE Trans Neural Netw 16(3):645–678
3. Kant S, Ansari IA (2016) An improved K means clustering with Atkinson index to classify liver patient dataset. Int J Syst Assur Eng Manag 7(1):222–228
4. Chang DX, Zhang XD, Zheng CW (2009) A genetic algorithm with gene rearrangement for K-means clustering. Pattern Recogn 42(7):1210–1222
5. Scheunders P (1997) A genetic c-means clustering algorithm applied to color image quantization. Pattern Recogn 30(6):859–866
6. Mitra S, Banka H (2006) Multi-objective evolutionary biclustering of gene expression data. Pattern Recogn 39(12):2464–2477
7. Gomez-Muñoz VM, Porta-Gándara MA (2002) Local wind patterns for modeling renewable energy systems by means of cluster analysis techniques. Renew Energy 25(2):171–182
8. Nanda SJ, Panda G (2014) A survey on nature inspired metaheuristic algorithms for partitional clustering. Swarm Evol Comput 16:1–18
9. Karaboga D, Basturk B (2008) On the performance of artificial bee colony (ABC) algorithm. Appl Soft Comput 8(1):687–697
10. Cura T (2012) A particle swarm optimization approach to clustering. Expert Syst Appl 39(1):1582–1588
11. Kumar Y, Sahoo G (2014) A charged system search approach for data clustering. Progress Artif Intell 2(2–3):153–166
12. Jordehi AR (2015) Enhanced leader PSO (ELPSO): a new PSO variant for solving global optimisation problems. Appl Soft Comput 26:401–417
13. Kushwaha N, Pant M, Kant S, Jain VK (2018) Magnetic optimization algorithm for data clustering. Pattern Recogn Lett 115:59–65
14. Hatamlou A (2013) Black hole: a new heuristic optimization approach for data clustering. Inf Sci 222:175–184
15. Karaboga D, Akay B (2009) A comparative study of artificial bee colony algorithm. Appl Math Comput 214(1):108–132
16. Dorigo M, Maniezzo V, Colorni A (1996) Ant system: optimization by a colony of cooperating agents. IEEE Trans Syst Man Cybern Part B Cybern 26(1):29–41
17. Jordehi AR (2014) A chaotic-based big bang–big crunch algorithm for solving global optimisation problems. Neural Comput Appl 25(6):1329–1335
18. Yang XS (2010) A new metaheuristic bat-inspired algorithm. In: Nature inspired cooperative strategies for optimization (NICSO 2010) (pp 65–74). Springer, Berlin, Heidelberg
19. Ashish T, Kapil S, Manju B (2018) Parallel bat algorithm-based clustering using MAPreduce. Networking communication and data knowledge engineering. Springer, Singapore, pp 73–82
20. Yilmaz S, Kucuksille EU (2013) Improved bat algorithm (IBA) on continuous optimization problems. Lecture Notes Softw Eng 1(3):279
21. Fister Jr I, Fister D, Yang XS (2013) A hybrid bat algorithm. ArXiv: 1303.6310
22. Meng XB, Gao XZ, Liu Y, Zhang H (2015) A novel bat algorithm with habitat selection and Doppler effect in echoes for optimization. Expert Syst Appl 42(17–18):6350–6364
23. Senthilnath J, Kulkarni S, Benediktsson JA, Yang XS (2016) A novel approach for multispectral satellite image classification based on the bat algorithm. IEEE Geosci Remote Sens Lett 13(4):599–603
24. Tang H, Sun W, Yu H, Lin A, Xue M (2020) A multirobot target searching method based on bat algorithm in unknown environments. Expert Syst Appl 141:112945
25. Gan C, Cao W, Wu M, Chen X (2018) A new bat algorithm based on iterative local search and stochastic inertia weight. Expert Syst Appl 104:202–212
26. Liu Q, Li J, Wu L, Wang F, Xiao W (2020) A novel bat algorithm with double mutation operators and its application to low-velocity impact localization problem. Eng Appl Artif Intell 90:103505
27. Liu Q, Wu L, Xiao W, Wang F, Zhang L (2018) A novel hybrid bat algorithm for solving continuous optimization problems. Appl Soft Comput 73:67–82
28. Eskandari S, Javidi MM (2020) A novel hybrid bat algorithm with a fast clustering-based hybridization. Evol Intel 13(3):427–442
29. Yildizdan G, Baykan ÖK (2020) A novel modified bat algorithm hybridizing by differential evolution algorithm. Expert Syst Appl 141:112949
30. Ghanem WA, Jantan A (2019) An enhanced Bat algorithm with mutation operator for numerical optimization problems. Neural Comput Appl 31(1):617–651
31. Cui Z, Li F, Zhang W (2019) Bat algorithm with principal component analysis. Int J Mach Learn Cybern 10(3):603–622
32. Cai X, Wang H, Cui Z, Cai J, Xue Y, Wang L (2018) Bat algorithm with triangle-flipping strategy for numerical optimization. Int J Mach Learn Cybern 9(2):199–215
33. Zhu LF, Wang JS (2019) Data clustering method based on bat algorithm and parameters optimization. Eng Lett 27(1):241–250
34. Shehab M, Khader AT, Laouchedi M, Alomari OA (2019) Hybridizing cuckoo search algorithm with bat algorithm for global numerical optimization. J Supercomput 75(5):2395–2422
35. Al-Betar MA, Awadallah MA (2018) Island bat algorithm for optimization. Expert Syst Appl 107:126–145
36. Liu L, Luo S, Guo F, Tan S (2020) Multi-point shortest path planning based on an Improved Discrete Bat Algorithm. Appl Soft Comput 95:106498
37. Aboubi Y, Drias H, Kamel N (2016) BAT-CLARA: BAT-inspired algorithm for Clustering LARge Applications. IFAC-Papers OnLine 49(12):243–248
38. Neelima S, Satyanarayana N, Murthy PK (2018) Minimizing frequent itemsets using hybrid ABCBAT algorithm. Data engineering and intelligent computing. Springer, Singapore, pp 91–97
39. Rahman MA, Islam MZ (2014) A hybrid clustering technique combining a novel genetic algorithm with K-Means. Knowl-Based Syst 71:345–365
40. Liu R, Jiao L, Zhang X, Li Y (2012) Gene transposon based clone selection algorithm for automatic clustering. Inf Sci 204:1–22
41. Cao F, Liang J, Jiang G (2009) An initialization method for the K-Means algorithm using neighborhood model. Comput Math Appl 58(3):474–483
42. Sahoo G (2017) A two-step artificial bee colony algorithm for clustering. Neural Comput Appl 28(3):537–551
43. Erisoglu M, Calis N, Sakallioglu S (2011) A new algorithm for initial cluster centers in k-means algorithm. Pattern Recogn Lett 32(14):1701–1705
44. Hatamlou A (2012) In search of optimal centroids on data clustering using a binary search algorithm. Pattern Recogn Lett 33(13):1756–1760
45. Žalik KR (2008) An efficient k′-means clustering algorithm. Pattern Recogn Lett 29(9):1385–1391
46. Kumar Y, Singh PK (2019) A chaotic teaching learning based optimization algorithm for clustering problems. Appl Intell 49(3):1036–1062
47. Fränti P, Sieranoja S (2018) K-means properties on six clustering benchmark datasets. Appl Intell 48(12):4743–4759
48. Taherdangkoo M, Shirzadi MH, Yazdi M, Bagheri MH (2013) A robust clustering method based on blind, naked mole-rats (BNMR) algorithm. Swarm Evol Comput 10:1–11
49. Senthilnath J, Omkar SN, Mani V (2011) Clustering using firefly algorithm: performance study. Swarm Evol Comput 1(3):164–171

50. Han X, Quan L, Xiong X, Almeter M, Xiang J, Lan Y (2017) A novel data clustering algorithm based on modified gravitational search algorithm. Eng Appl Artif Intell 61:1–7
51. Zhang C, Ouyang D, Ning J (2010) An artificial bee colony approach for clustering. Expert Syst Appl 37(7):4761–4767
52. Yan X, Zhu Y, Zou W, Wang L (2012) A new approach for data clustering using hybrid artificial bee colony algorithm. Neurocomputing 97:241–250
53. Kumar Y, Sahoo G (2015) Hybridization of magnetic charge system search and particle swarm optimization for efficient data clustering using neighborhood search strategy. Soft Comput 19(12):3621–3645
54. Zhou Y, Zhou Y, Luo Q, Abdel-Basset M (2017) A simplex method-based social spider optimization algorithm for clustering analysis. Eng Appl Artif Intell 64:67–82
55. Boushaki SI, Kamel N, Bendjeghaba O (2018) A new quantum chaotic cuckoo search algorithm for data clustering. Expert Syst Appl 96:358–372
56. Chang D, Zhao Y, Zheng C, Zhang X (2012) A genetic clustering algorithm using a message-based similarity measure. Expert Syst Appl 39(2):2194–2202
57. Xiao J, Yan Y, Zhang J, Tang Y (2010) A quantum-inspired genetic algorithm for k-means clustering. Expert Syst Appl 37(7):4966–4973
58. Bijari K, Zare H, Veisi H, Bobarshad H (2018) Memory-enriched big bang–big crunch optimization algorithm for data clustering. Neural Comput Appl 29(6):111–121
59. Abualigah LM, Khader AT, Hanandeh ES, Gandomi AH (2017) A novel hybridization strategy for krill herd algorithm applied to clustering techniques. Appl Soft Comput 60:423–435
60. Pakrashi A, Chaudhuri BB (2016) A Kalman filtering induced heuristic optimization based partitional data clustering. Inf Sci 369:704–717
61. Kang Q, Liu S, Zhou M, Li S (2016) A weight-incorporated similarity-based clustering ensemble method based on swarm intelligence. Knowl-Based Syst 104:156–164
62. Wang R, Zhou Y, Qiao S, Huang K (2016) Flower pollination algorithm with bee pollinator for cluster analysis. Inf Process Lett 116(1):1–14
63. Hatamlou A, Hatamlou M (2013) PSOHS: an efficient two-stage approach for data clustering. Memetic Comput 5(2):155–161
64. Jiang B, Wang N (2014) Cooperative bare-bone particle swarm optimization for data clustering. Soft Comput 18(6):1079–1091
65. Kumar Y, Singh PK (2018) Improved cat swarm optimization algorithm for solving global optimization problems and its application to clustering. Appl Intell 48(9):2681–2697
66. Kwedlo W (2011) A clustering method combining differential evolution with the K-means algorithm. Pattern Recogn Lett 32(12):1613–1621
67. Yin M, Hu Y, Yang F, Li X, Gu W (2011) A novel hybrid K-harmonic means and gravitational search algorithm approach for clustering. Expert Syst Appl 38(8):9319–9324
68. Jiang H, Yi S, Li J, Yang F, Hu X (2010) Ant clustering algorithm with K-harmonic means clustering. Expert Syst Appl 37(12):8679–8684
69. Kaur A, Kumar Y (2021) A new metaheuristic algorithm based on water wave optimization for data clustering. Evol Intell 2021:1–25
70. Kuo RJ, Zheng YR, Nguyen TPQ (2021) Metaheuristic-based possibilistic fuzzy k-modes algorithms for categorical data clustering. Inf Sci 557:1–15
71. Hu KC, Tsai CW, Chiang MC (2020) A multiple-search multi-start framework for metaheuristics for clustering problems. IEEE Access 8:96173–96183
72. Aljarah I, Mafarja M, Heidari AA, Faris H, Mirjalili S (2020) Clustering analysis using a novel locality-informed grey wolf-inspired clustering approach. Knowl Inf Syst 62(2):507–539
73. Zhou X, Zhang R, Wang X, Huang T, Yang C (2020) Kernel intuitionistic fuzzy c-means and state transition algorithm for clustering problem. Soft Comput 24(20):15507–15518
74. Jensi R, Jiji GW (2016) An improved krill herd algorithm with global exploration capability for solving numerical function optimization problems and its application to data clustering. Appl Soft Comput 46:230–245
75. Hatamlou A (2017) A hybrid bio-inspired algorithm and its application. Appl Intell 47(4):1059–1067
76. Bouyer A, Hatamlou A (2018) An efficient hybrid clustering method based on improved cuckoo optimization and modified particle swarm optimization algorithms. Appl Soft Comput 67:172–182
77. Xiang WL, Zhu N, Ma SF, Meng XL, An MQ (2015) A dynamic shuffled differential evolution algorithm for data clustering. Neurocomputing 158:144–154

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.