K-Means-Based Nature-Inspired Metaheuristic Algorithms for Automatic Data Clustering Problems: Recent Advances and Future Directions

Abiodun M. Ikotun 1, Mubarak S. Almutari 2 and Absalom E. Ezugwu 1,*

1 School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal, King Edward Road, Pietermaritzburg 3201, South Africa; 220078470@stu.ukzn.ac.za
2 College of Computer Science, University of Hafr Al Batin, Hafar Al Batin 39524, Saudi Arabia; almutairi@uhb.edu.sa
* Correspondence: ezugwua@ukzn.ac.za

Abstract: K-means clustering algorithm is a partitional clustering algorithm that has been used widely in many applications for traditional clustering due to its simplicity and low computational complexity. This clustering technique depends on the user specification of the number of clusters generated from the dataset, which affects the clustering results. Moreover, random initialization of cluster centers results in its local minimal convergence. Automatic clustering is a recent approach to clustering where the specification of cluster number is not required. In automatic clustering, natural clusters existing in datasets are identified without any background information of the data objects. Nature-inspired metaheuristic optimization algorithms have been deployed in recent times to overcome the challenges of the traditional clustering algorithm in handling automatic data clustering. Some nature-inspired metaheuristics algorithms have been hybridized with the traditional K-means algorithm to boost its performance and capability to handle automatic data clustering problems. This study aims to identify, retrieve, summarize, and analyze recently proposed studies related to the improvements of the K-means clustering algorithm with nature-inspired optimization techniques. A quest approach for article selection was adopted, which led to the identification and selection of 147 related studies from different reputable academic avenues and databases. More so, the analysis revealed that although the K-means algorithm has been well researched in the literature, its superiority over several well-established state-of-the-art clustering algorithms in terms of speed, accessibility, simplicity of use, and applicability to solve clustering problems with unlabeled and nonlinearly separable datasets has been clearly observed in the study. The current study also evaluated and discussed some of the well-known weaknesses of the K-means clustering algorithm, for which the existing improvement methods were conceptualized. It is noteworthy to mention that the current systematic review and analysis of existing literature on K-means enhancement approaches presents possible perspectives in the clustering analysis research domain and serves as a comprehensive source of information regarding the K-means algorithm and its variants for the research community.

Keywords: K-means clustering; automatic clustering; nature-inspired metaheuristic algorithms; cluster analysis

1. Introduction

Data clustering is an aspect of data mining that aims at classifying or grouping data objects within a dataset based on their similarities and dissimilarities. A dataset is segmented into clusters so that the data objects within the same cluster are more similar than those in other clusters. In other words, data grouping is performed to reduce the intra-cluster distance among data objects while increasing the inter-cluster distance. Data clustering has been very useful for classifying data in many applications such as biological
data analysis, social network analysis, mathematical programming, customer segmentation, image segmentation, data summarization, and market research [1].

There are several methods used for clustering datasets. These methods are majorly classified into two categories: the hierarchical clustering methods and the partitional clustering methods. In the hierarchical clustering technique, data objects are iteratively grouped in a hierarchical format to generate a dendrogram that depicts the clustering sequence of the dataset. The partitional clustering technique generates a single dataset partition to recover the natural groupings within the dataset without any hierarchical structure using a specific objective function. Among the many partitional clustering methods is the well-known K-means clustering algorithm. The K-means clustering algorithm is a partitional non-deterministic method that MacQueen proposed in 1967 [2]. For the K-means algorithm, objects are grouped into a user-specified ‘k’ number of clusters based on the minimum distance between the data objects and cluster centers [3]. According to Ezugwu et al. [4], the K-means clustering algorithm is straightforward to implement, flexible, and efficient. It has been rated among the top ten algorithms most used in data mining, and it has enjoyed wide acceptability in many domains due to its low computation complexity and implementation simplicity. The dependability of the algorithm on the user’s specification of the number of clusters and the random initialization of the initial cluster center limits the performance and the accuracy of the cluster results. Different initial values of k produce different clustering results, and the random selection of the initial clusters makes the algorithm tends toward converging into local minimal.

Choosing appropriate cluster numbers for datasets containing high dimensional data objects with varying densities and sizes is difficult without prior domain knowledge [5]. The requirement for pre-defining the number of clusters makes the K-means algorithm inefficient for automatic clustering. Because, for the automatic clustering methods, the adequate number of clusters in a dataset are determined automatically without any background information of the data objects in the dataset. In view of this, nature-inspired metaheuristics have been adopted in finding solutions to automatic clustering problems [6,7]. A few nature-inspired metaheuristics algorithms have been combined with the traditional k-means algorithm to optimize its performance and increase its ability to handle automatic clustering problems. In this study, we review and analyze the different nature-inspired metaheuristic algorithms that have been integrated with K-means or any of its variants in recent times to solve the automatic data clustering problems.

There are many published articles on reviews regarding the use of nature-inspired clustering algorithms focusing on automatic clustering alone. An up-to-date study of all major nature-inspired metaheuristic algorithms for solving automatic clustering problems was presented by Jose-Garcia and Gomez-Flores [6]. Ezugwu et al. [8] presented a systematic taxonomical overview and bibliometric analysis of the trends and progress in nature-inspired metaheuristic clustering approaches, with emphasis on automatic clustering algorithms. There are domain-specific review works where different metaheuristics techniques were utilized [9–11]. A review of nature-inspired algorithms that have been employed to solve partitional clustering problems, including the major areas of application, was presented by Nanda and Panda [10]. Mane and Gaikwad [12] presented an overview of the nature-inspired techniques used for data clustering. Their study covers the hybridization of several nature-inspired techniques with some traditional clustering techniques to improve the performance of the existing clustering approaches. This study presents a systematic review on the different nature-inspired metaheuristic algorithms integrated with K-means or any of its variants for cluster analysis in the last two decades, emphasizing automatic clustering. A total of 147 articles were considered in the review.

Despite the various review papers published on the nature-inspired algorithm and clustering algorithms, including the traditional clustering and automatic clustering methods, to the best of our knowledge, at the point of writing this paper, no extensive review study on the hybridization of nature-inspired algorithms with the K-means clustering algorithm exist with a primary focus on automatic clustering. Because of this limitation
and identified gap, an up-to-date and in-depth review of hybridization of nature-inspired metaheuristic algorithm with K-means clustering algorithm and its variants over the last two decades is presented in this paper.

This study is significant in many ways, and more specifically due to its advantages of (i) identifying, categorizing, and analyzing the various improvement methods and hybridization techniques for the classical K-means algorithms in solving various automatic data clustering problems, (ii) identifying the variants of the K-means based nature-inspired metaheuristic algorithms, (iii) presentation of further comparative analysis of data in the form of charts and tables across a wide variety of hybridization techniques attributes, (iv) identifying the strengths and weaknesses of the existing implementation of hybrid K-means based nature-inspired metaheuristic algorithms, (v) identifying recent trends of hybridizing nature-inspired metaheuristic algorithms with the classical K-means algorithm for solving automatic data clustering problems and open challenges, and (vi) suggesting new possible future research directions for the domain enthusiasts. It is also noteworthy that researchers and practitioners interested in exploiting and harnessing the advantages of K-means clustering with those of nature-inspired algorithms for implementing a better-performed automatic clustering technique will find this work useful. It will also be helpful for researchers in the domain of constrained and unconstrained optimization techniques.

The remaining sections of the paper are structured as follows: Section 2 gives a brief description of the scientific background of K-means clustering algorithm, nature-inspired metaheuristic algorithms, and automatic clustering problems. The section similarly reiterates the research methodology approach to the systematic literature review and analysis of the study. The existing integration of the K-means clustering algorithm with nature-inspired metaheuristic algorithms in literature is presented in Section 3. Section 4 discusses the critical issues of integrating the K-means clustering algorithm with nature-inspired metaheuristic algorithms for automatic clustering. Subsequently, open challenges and future research directions are also covered in this section. Finally, Section 5 gives the study concluding remarks.

2. Scientific Background

The K-means clustering algorithm is a partitional clustering technique that splits a dataset into $k$ number of clusters using a specific fitness measure. That is, given a dataset $X$, $X$ is divided into $K$ non-overlapping groups $C = \{c_1, c_2, \ldots, c_k\}$, $c_i \neq \emptyset$, $i = 1, \ldots$, and such that $\bigcup_{i=1}^{k} c_i = X; c_i \cap c_j = \emptyset$, $i, j = 1 \ldots k$ and $i \neq j$. The partitioning process is handled as an optimization problem with the fitness measure taking as the objective function such as minimizing the distances between data objects or maximizing the correlation between data objects [10]. Mathematically, the optimization problem for cluster analysis is defined as follows:

Given a dataset $X = \{x_i\}$, where $i = 1, 2, \ldots, n$ of d-dimension data points of size $n$, $X$ is partitioned into ‘$k$’ clusters such that

$$J(c_k) = \sum_{x_i \in c_k} ||x_i - \mu_k||^2$$

with the objective function: minimize the sum of the square error over all the $k$ clusters. That is, minimize

$$J(C) = \sum_{k=1}^{K} \sum_{x_i \in c_k} ||x_i - \mu_k||^2$$

In automatic clustering, the main concerns are determining the best estimate for cluster number $k$ and correct identification of all the partitions [6]. In other words, an automatic
clustering algorithm seeks to optimize the number of combinations in the assignments of \( N \) objects into \( k \) clusters. This is mathematically represented as:

\[
S(N, K) = \frac{1}{K!} \sum_{i=0}^{K} (-1)^{K-i} \binom{K}{i} i^N
\]  

In finding the optimal cluster number, the search space is mathematically represented as:

\[
B(N) = \sum_{K=1}^{N} S(N, K)
\]  

The task of finding an optimal solution for this problem when \( k > 3 \) is NP-hard [6,13] and this makes the search computationally expensive for moderately sized problems [6,14]. In recent times, there has been an immeasurable increase in the magnitude of data being generated. The current real-world datasets are characterized as being high dimensional and massive in size. Automatic clustering of such datasets with no background knowledge of the features of the data objects can be termed a difficult task. Without prior domain knowledge, it is difficult to determine the appropriate number of clusters for a massive, high-dimensional dataset. Moreover, due to the enormous size of data objects in real-world datasets, the distribution of data objects into appropriate clusters to produce an optimal cluster result is computationally intensive and time-consuming.

2.1. Nature-Inspired Metaheuristics for Automatic Clustering Problems

Metaheuristics are global optimization techniques used to solve complex real-life problems [8,15]. A higher-level procedure applies simpler procedures in solving optimization problems [16]. In optimization, inputs to an objective function are adjusted to find the optimum solution. According to Engelbrecht [17], it is possible to formulate clustering problems as an optimization problem that can be comfortably solved using single objective and multi-objective metaheuristics. Metaheuristics can find the optimum solution to global optimization problems with less computational effort. They find an approximate solution and are non-deterministic as well as non-problem dependent. Agbaje et al. [18] stated that most metaheuristic algorithms can partition datasets automatically into an optimal number of clusters when a good validity measure is applied.

The nature-inspired metaheuristic algorithms are modeled after the behavioral pattern of natural phenomena exhibiting the learning ability and adaption to emerging situations in finding appropriate solutions to problems in changing and complex environments [17]. According to Ezugwu et al. [8], nature-inspired algorithms are designed practically to find a solution to high-dimensional and complex real-world problems. They have satisfactorily proffer suboptimal solutions to automatic clustering problems within an acceptable time limit [7]. As a result of their capability for higher heuristic search, they seek the most appropriate solution in the search space and at the same time try to maintain the balance between intensification (local optimal search) and diversification (global optimal search) [19]. The nature-inspired metaheuristic uses the population to explore the search space, ensuring a greater probability of achieving optimal cluster partitions [10].

Alongside the successes recorded with solving automatic clustering problems using nature-inspired metaheuristic algorithms, it has been observed that hybridizing two or more metaheuristics for the same purpose produces better clustering performance. According to Nanda and Panda [10], the performance of the hybrid algorithms is superior to that of the individual algorithms in terms of robustness, efficiency, and accuracy. Nature-inspired metaheuristics have also been hybridized with some of the traditional clustering algorithms to improve their performance [5]. K-means clustering algorithm is one of the most fundamental and popular traditional partitional clustering algorithms that has been used in many applications. In order to improve its performance for the general clustering problem, several variants of K-means have been proposed in the literature. The traditional K-means algorithm, with its numerous variants though credited with computational simplicity, are
however limited in their performance due to the possibility of getting trapped in the local optimum because of its hill-climbing approach. As a result, some of the metaheuristic algorithms have been hybridized with it to improve its performance.

2.2. Review Methodology

A comprehensive literature review includes the basic introduction to a specific problem and the critical assessment, evaluation, and interpretation of existing related literature and materials. When considering the authors, countries, publishers, studies, journals, and universities, there should be no bias. In this comprehensive review, three major phases are considered: the review planning, the conducting of the review, and the review reporting. This methodology process is illustrated in Figure 1. The primary aim of the planning phase is the identification of the need and worth of this review. It includes designing the research questions that guide selecting relevant related manuscripts for the review and analysis processes. It also addressed the strategy adopted for literature search from the relevant academic databases to ensure unbiased and extensive primary studies.

![Figure 1. Adopted review of research methodology.](image)

### 2.2.1. Research Questions

In this study, answers are provided to the following research questions:

- **RQ1**: What are the various nature-inspired meta-heuristics techniques that have been hybridized with the K-means clustering algorithm?
- **RQ2**: Which of the reported hybridization of nature-inspired meta-heuristics techniques with K-means clustering algorithm handled automatic clustering problems?
- **RQ3**: What various automatic clustering approaches were adopted in the reported hybridization?
- **RQ4**: What contributions were made to improve the performance of the K-means clustering algorithm in handling automatic clustering problems?
- **RQ5**: What is the rate of publication of hybridization of K-means with nature-inspired meta-heuristic algorithms for automatic clustering?

It is equally important to note that providing answers to these research questions establishes this study’s primary goal and specific objectives. In other words, the responses help define the study’s motivation and focus relative to the reader’s interest.

2.3. Adopted Strategy for Article Selection

In determining and shortlisting relevant articles that cover and answer all the designed research questions stated in Section 2.2.1 successfully, the quest approach was adopted. To extract relevant articles from the database with better coverage for the study, different keywords that basically relate to the study along with their synonyms were used in the search. Relevant articles on “nature-inspired metaheuristic”, “K-means clustering algorithm”, and “automatic clustering” published in the last two decades were obtained. The search for articles was performed on seven different academic databases: ACM Digital Library, Elsevier Journal, Wiley Online Library, IEEE Explore, Springer Link, DLBP, and CiteSeer. The search for relevant articles was streamlined to the last two decades to reduce
the number of articles. A total number of 3423 articles were extracted, and 1826 duplicate copies were removed from the lot. The selected articles were restricted to those published only in the English language. On careful investigation of the articles’ title, abstract, and contents, the remaining 1597 articles were further reduced to 147, which made up the final most relevant articles selected for the study.

3. Data Synthesis and Analysis

This section covers the answers to the designed research questions stated in Section 3.1, with each subsection distinctly handling answers to each research question.

3.1. RQ1. What Are the Various Nature-Inspired Meta-Heuristics Techniques That Have Been Hybridized with the K-Means Clustering Algorithm?

Meta-heuristics techniques are developed for providing optimal solutions to optimization problems through iterative exploration and exploitation of the entire search space [20]. A number of these algorithms have been integrated with the traditional K-means algorithm to improve the process of data clustering. The following section presents the various nature-inspired meta-heuristics techniques that have been hybridized with the K-means clustering algorithm.

3.1.1. Genetic Algorithm

The genetic algorithm (GA) was introduced by Holland in 1975 [21] based on the evolutionary principle of Charles Darwin [22]. The evolutionary principle states that “only the species that are fit to survive can reproduce their kind”. The computer simulation of this evolutionary process produced the Genetic Algorithm [21]. The earliest work on hybridizing the K-means clustering algorithm with GA for data clustering was reported by Krishna and Murty [23] in their paper titled ‘Genetic K-Means Algorithm’. The main purpose of the hybridization was to find a global optimal partition of a given dataset based on a given number of clusters. It also addressed the problem of expensive crossover operators and costly fitness functions common with the traditional GA. Even though GKA was able to converge to the best-known optimum, the number of clusters needs to be specified. Bandyopadhyay and Maulik [24] introduced KGA-clustering, which exploits the searching capability of K-means while avoiding the problem of local optimal convergence. Cheng et al. [25] presented prototypes-embedded genetic K-means (PGKA) where prototypes of clusters were encoded as chromosomes. Laszlo and Mukherjee [26] evolve centers for K-means clustering algorithm using GA by constructing hyper-quad tree on the datasets to represent cluster centers set. In their paper, Laszlo and Mukherjee [27] also proposed a novel crossover operator for neighboring centers exchange for superior partitions of large simulated datasets. Dai, Jiao, and He [28] proposed parallel genetic algorithm-based K-means clustering adopting the variable-length chromosome encoding strategy. Chang, Zhang, and Zheng [29] integrated the K-means algorithm with GA with gene rearrangement (GAGR) to improve clustering performance. Sheng, Tucker, and Liu [30] proposed niching genetic K-means algorithm (NGKA) for the partitional clustering algorithm. KMQGA was proposed by Xiao et al. [31] as a quantum-inspired genetic algorithm for K-means clustering with the Q-bit-based representation for exploration and exploitation purposes. It was able to obtain the optimal number of clusters and also provide the optimal cluster centroid. Rahman and Islam [32] proposed a novel GA-based clustering technique that automatically finds the correct numbers of clusters and produces high-quality cluster centers that serve the initial seeds for the K-Means algorithm to produce a high-quality clustering solution. Kapil, Chawla, and Ansari [3] optimized the K-means algorithm using GA.

In more recent works, Sinha and Jana [33] combined GA with Mahalanobis distance and K-means for clustering distributed datasets using the MapReduce framework. Islam et al. [34] extended GENCLUS by combining genetic operators’ capacity to combine the different search space solutions with the K-means’ hill climber exploitation. Zhang and Zhou [35] proposed NClust, which combined novel niching GA (NNGA) with K-
means to determine clusters number automatically. Mustafi and Sahoo [36] explored the GA framework and differential evolution (DE) heuristic to improve the cluster center selection and obtain the required number of clusters respectively for the traditional K-means algorithm. El-Shorbagy et al. [37] proposed an enhanced GA with a new mutation where the K-means algorithm initializes the GA population for finding the best cluster centers. Genetic K-Means clustering (GKMC) was proposed by Ghezelbash, Maghsoudi, and Carranza [38] for optimally delineating multi-elemental patterns in stream sediment geochemical data. Kuo et al. [39] integrated self-organizing feature maps neural network with genetic k-means for Market segmentation. Sheng, Tucker, and Liu [30] employed NGKA in clustering gene expression data. Li et al. [40] combined GA with an improved K-means clustering algorithm for video image indexing. Karegowda et al. [41] used GA and entropy-based fuzzy clustering (EFC) to assign initial cluster centers for the K-means algorithm for PIMA Indian diabetic dataset clustering. Genetic K-Means clustering (GKMC) was proposed by Ghezelbash, Maghsoudi, and Carranza [38] for optimally delineating multi-elemental patterns in stream sediment geochemical data.

Kuo et al. [39] integrated self-organizing feature maps neural network with genetic k-means for Market segmentation. Sheng, Tucker, and Liu [30] employed NGKA in clustering gene expression data. Li et al. [40] combined GA with an improved K-means clustering algorithm for video image indexing. Karegowda et al. [41] used GA and entropy-based fuzzy clustering (EFC) to assign initial cluster centers for the K-means algorithm for PIMA Indian diabetic dataset clustering. Eshlaghy and Razi [42] used an integrated framework that combines a grey-based K-means algorithm with GA for project selection and project management. Lu et al. [43] combined GA and K-means to solve the multiple travelling salesman problem (MTSP). K-means was combined with improved GA by Barekatain, Dehghani, and Pourzaferani [44] for energy consumption reduction and network lifetime extension in wireless sensor networks. Zhou et al. [45] proposed NoiseClust, which combines GA and K-means++ with an improved noise method for mining better origins and destinations in global position system (GPS) data. Mohammadrezapour, Kisi, and Pourahmad [46] used K-means clustering with GA to identify homogeneous regions of groundwater quality.

3.1.2. Particle Swarm Optimization

The particle swarm optimization (PSO) is a population-based metaheuristic search algorithm that is based on the principle of social behavior of swarms [47]. It is a powerful optimization tool credited with implementation simplicity, fewer parameter configuration, and global exploration ability [48]. According to Niu et al. [48], diverse versions of PSO have been reported in the literature with a number implemented for clustering purposes [49–57]. Several pieces of literature report the hybridization of PSO with the K-means clustering algorithm. Van der Merwe and Engelbrecht [49] proposed two different approaches of integrating PSO with K-means clustering algorithm for data clustering. In one of the approaches, PSO was used to find centroid for a specified number of clusters, while in the other approach, K-means was used to find the initial swarm for PSO. Omran, Salman, and Engelbrecht [58] presented a dynamic clustering approach (DCPSO) based on the integration of PSO with the K-means clustering algorithm. The PSO is used to select the best number of clusters with the K-means clustering algorithm used to refine the chosen clusters’ centers.

Chen and Zhang [59] combined K-means and PSO to propose RVPK for clustering Web Usage patterns achieving better stability. Kao, Zahara, and Kao [60] proposed K-NM-PSO, which hybridized PSO and Nelder–Mead simplex search with K-means clustering algorithm. Kao and Lee [61] presented KCPSO—K-means and combinatorial particle swarm optimization, which do not require the specification of cluster number a priori. K-harmonic means (KHM) was hybridized with PSO by Yang, Sun, and Zhang [62] to fully use the advantages of the two algorithms for better cluster analysis. Niknam and Amiri [53] introduced FAPSO-AC-K, which combines fuzzy adaptive particle swarm optimization with ant colony optimization and K-means clustering algorithm for better cluster partition. Tsai and Kao [63] presented a selective regeneration PSO (SRPSO), which was hybridized with a K-means clustering algorithm to develop an efficient, accurate and robust K-means selective regeneration PSO (KSRPSO) for data clustering. Prabha and Visalakshi [64] proposed an improved PSO-based K-means clustering algorithm that integrates PSO and the traditional K-means clustering algorithm with normalization as a preprocessing step for transforming the dataset attributes values.
Emami and Derakhshan [65] proposed PSOFKM, which combined PSO with fuzzy K-means to explore the merits of the two algorithms solving the problem of initial states sensitivity of the traditional K-means clustering algorithm. Hybridization of K-means with improved PSO and GA for improved convergence speed and global convergence was proposed by Nayak et al. [66]. The IPSO handled the global search for optimal cluster center while GA was used to improve the particles quality and diversification of solution space. Niu et al. [48] proposed a population-based clustering technique that integrates PSO with the traditional K-means algorithm. Six different variants of PSO were integrated with the Lloyd’s K-means [67] separately, varying the PSO’s neighbor social communications. Ratanavilisagul [68] proposed an improvement on the regular hybridization of PSO and K-means clustering algorithm by applying mutation operation with PSO particles. Paul, De, and Dey [69] presented a modified PSO (MiPSO) based K-means algorithm where the MiPSO is employed to generate initial cluster centers for the K-means clustering algorithm. IPSO handled the global search for optimal cluster center while GA was used to improve the particles quality and diversification of solution space. Niu et al. [48] proposed a population-based clustering technique that integrates PSO with the traditional K-means algorithm. Six different variants of PSO were integrated with the Lloyd’s K-means [67] separately, varying the PSO’s neighbor social communications. Ratanavilisagul [68] proposed an improvement on the regular hybridization of PSO and K-means clustering algorithm by applying mutation operation with PSO particles. Paul, De, and Dey [69] presented a modified PSO (MiPSO) based K-means algorithm where the MiPSO is employed to generate initial cluster centers for the K-means clustering algorithm. Jie and Yibo [70] proposed a technique for outlier detection by combining PSO with K-means for fault data sorting of feeder in distribution network information system. The PSO was used to optimize the cluster centroid while the K-means algorithm determined the optimal number of clusters. Chen, Miao, and Bu [71] presented an aggregation hybrid of K-means clustering algorithm with PSO for image segmentation.

3.1.3. Firefly Algorithm

The firefly algorithm (FA) is a swarm intelligence metaheuristic optimization technique that was first introduced by Yang in 2009 [72]. According to Xie et al. [73], FA has a unique capability of automatic subdivision compared with other metaheuristic search algorithms. Hassanzadeh and Meybodi [74] presented a hybrid algorithm K-FA that combined the K-means algorithm and firefly algorithm. The firefly algorithm was used to find centroid for specified k number of clusters with K-means algorithm used for refining the centroid. Mathew and Vijayakumar [75] proposed using a firefly-based clustering method to parallel K-means for handling a large number of clusters. Similar to Hassan-zadeh and Meybodi [74], the FA finds the initial optimal centroid, which is then refined using K-means for improved clustering accuracy. Nayak et al. [76] presented an integrated clustering framework combining optimized K-means with firefly algorithm and Canopies for better clustering accuracy.

To address K-means’ initialization sensitivity and local optimal convergence, Behera et al. [77] proposed FCM-FA, hybridizing fuzzy C-means with a firefly algorithm for faster convergence. Nayak, Naik, and Behera [78] proposed a novel firefly-based K-means algorithm—FA-K-means, in which the global search capacity of the FA was used to resolve the problem of local convergence of the K-means for efficient cluster analysis. Xie et al. [73] proposed two variants of the FA (IIEFA—inward intensified exploration FA and CIEFA—compound intensified exploration FA) which are incorporated into the K-means clustering algorithm for improved clustering performance. Jitpakdee, Aimmanee, and Uyyanonvara [79] proposed a hybrid firefly algorithm and K-means algorithm for color image quantization. Kuo and Li [80] integrate a firefly-algorithm-based support vector regression with wavelet transform in developing an export trade value prediction system. Kaur, Pal, and Singh [81] introduced a K-means and firefly algorithm hybridization for the intrusion detection system.

Langari et al. [82] proposed KFCFA—K-member fuzzy clustering and firefly algorithm, which is a combined anonymizing algorithm for protecting anonymized databases against identity disclosure in social networks. HimaBindu et al. [83] proposed a firefly-based K-means algorithm with global search capability for clustering big data. Wu et al. [84] proposed a novel kernel extreme learning machine model coupled with K-means clustering and firefly algorithm (Kmeans-FFA-KELM) for the monthly reference evapotranspiration estimation in parallel computation.
3.1.4. Bat Algorithm

The bat algorithm (BA), introduced by Xin-She Yang in 2010 [85], is one of the nature-inspired optimization algorithms based on the echolocation behavioral pattern of bats. K-Medoids was combined with the bat algorithm by Sood and Bansal [86] for partitioning clustering using the echolocation behavior of bats to determine the initial cluster number. Tripathi, Sharma, and Bala [87] hybridized the K-means algorithm with a novel dynamic frequency-based bat algorithm variant (DFBPKBA) as a new approach for clustering in a distributed environment with a better exploration and exploitation capability. The MapReduce model in the Hadoop framework was used to parallelize the hybrid algorithm to ensure satisfactory results within a reasonable time limit. Pavez, Altimiras, and Villavicencio [88] introduced the K-means binary bat algorithm (BKBA) using a generalized K-means-based binarization mechanism applied to the bat algorithm to solve multidimensional backpack problems. Gan and Lai [89] introduce a bat algorithm clustering based on K-means (KMBA) for automated grading of edible birds nest, which produce nearly 86% dataset clustering accuracy compared with the standard bat algorithm. Chaudhary and Banati [90] hybridized K-means and K-medoids with an enhanced shuffled bat algorithm (EShBAT). K-means and K-medoids were used in generating a rich starting population for EShBAT to produce an efficient clustering algorithm.

3.1.5. Flower Pollination Algorithm

The flower pollination algorithm (FPA) is a metaheuristic optimization algorithm motivated by the process of pollinating flowering plants. Xin-She Yang developed the first FPA in 2012 [91] as a global optimization technique. Jenssi and Jiji [92] proposed a novel hybrid FPAKM clustering method that combines the flower pollination algorithm with the K-means clustering algorithm. Kumari, Rao, and Rao [93] introduce a flower pollination-based K-means clustering algorithm using vector quantization for better medical image compression.

3.1.6. Artificial Bee Colony

The artificial bee colony (ABC) is a swarm intelligence algorithm inspired by bees’ search mode and division of labor in finding the maximum amount of nectar [94]. Armano and Farmani [95] proposed kABC, which combined K-means and ABC to improve K-means capability in finding global optimum clusters. Tran et al. [96] presented EABCK, an enhanced artificial bee colony algorithm, and K-means to improve the performance of the K-means clustering algorithm. The ABC was guided by the global best solution with mutation operation to produce an enhanced version of EABC. Bonab et al. [97] combined an artificial bee colony algorithm and differential evolution with a modified K-means clustering algorithm to address the problem of local optimum convergence of K-means in color image segmentation.

The CAABC-K, which is a hybrid of chaotic adaptive artificial bee colony algorithm (CAABC) with K-means algorithm, was proposed by Jin, Lin, and Zhang [98]. The CAABC-K had better convergence speed and accuracy compared with some conventional clustering algorithms. Dasu, Reddy, and Reddy [99] integrate the K-means clustering algorithm and ABC optimization algorithm for remote sensing images classification. K-means algorithm was used for image segmentation, while ABC was used for classification. Huang [100] combined ABC with an accelerated k-means algorithm for color image quantization. Wang et al. [101] proposed the ABC-KM algorithm for the improvement of wind farm clustering. Modified artificial bee colony combined with K-means clustering algorithm—MABC-K, was proposed by Cao and Xue [102] to establish a hybrid algorithm framework for clustering problems.

3.1.7. Grey Wolf Optimizer

Mirjalili, Mirjalili, and Lewis [103] proposed the grey wolf optimizer (GWO) as a metaheuristic optimization algorithm mimicking grey wolves’ hunting mechanism and leadership hierarchy. Katarya and Verma [104] combined fuzzy c-mean (FCM) with grey wolf
optimizer as a collaborative recommender system proposed to enhance system accuracy and precision. Korayem, Khorsid, and Kassem [105] proposed ‘K-GWO’—a combination of GWO and traditional K-means clustering algorithm into which a capacity constraint was incorporated for solving capacitated vehicle routing problems. Pambudi, Badharudin, and Wicaksono [106] enhanced the K-means clustering algorithm using GWO. The GWO rule was used in minimizing the SSE of the population and searching for a new cluster center. Mohammed et al. [107] introduced KMGWO, in which the K-means clustering algorithm was used to enhance GWO’s performance.

3.1.8. Sine–Cosine Algorithm

The sine–cosine algorithm (SCA) is a population-based optimization algorithm that uses a mathematical model based on sine and cosine function in finding the optimal solution to optimization problems [108]. The SCAK-means is a hybridization of the sine-cosine algorithm and K-means clustering algorithm proposed by Moorthy and Pabitha [109]. They integrated with a resource discovery system adopted in cloud computing resource sharing management.

3.1.9. Cuckoo Search Algorithm

The cuckoo search (CS) algorithm is a nature-inspired metaheuristic algorithm developed by Xin-She Yang in 2009 [110]. It imitates the obligate parasitism of special female cuckoo species, which mimic the color and pattern of their chosen host birds. Step size affects the precision of the cuckoo search metaheuristic algorithm [111]. Saida, Kamel, and Omar [112] combined the K-means algorithm with CS for document clustering to avoid the problem of a drastic increase in the iterations in the standard CS. Girsang, Yunanto, and Aslamiah [113] proposed a combination of cuckoo search algorithm and K-means called FCSA to accelerate the computational time of the clustering algorithm. The FCSA uses CS in building robust initialization while K-means was used to accelerate the building of the solutions. Ye et al. [111] presented an improved cuckoo search K-means algorithm (ICS-Kmeans) to address the step size problem common with the cuckoo search algorithm. Lanying and Xiaolan [114] used the CS algorithm in optimizing the K-means algorithm for collaborative filtering recommendations. Tarkhaneh, Isazadeh, and Khamnei [115] introduced a hybrid algorithm combining the K-means algorithm with CS and PSO that yields more optimized results than each of the individual standard algorithms.

Singh and Solanki [116] integrate K-means with a modified cuckoo search algorithm (K-means modified cuckoo search) to achieve a global optimum solution in a recommender system. Arjmand et al. [117] proposed a hybrid clustering algorithm that combined the K-means clustering algorithm used for segmentation with cuckoo search optimization for generating the initial centroids for the K-means algorithm in breast tumor segmentation. García, Yepes, and Martí [118] proposed a K-means cuckoo search hybrid algorithm with the cuckoo search metaheuristics serving as the continuous space optimization mechanism and using the learning technique of the unsupervised K-means algorithm in the discretization of the obtained solution. Multiple kernel-based fuzzy c-means algorithm was hybridized with cuckoo search to produce MKF-cuckoo by Binu, Selvi, and George [119] with more effective objective functions designed by the researchers instead of using the K-means objective function. Manju and Fred [120] solved the problem of segmentation and compression of compound images using a hybrid of K-means clustering algorithm and multi-balanced cuckoo search algorithm. Deepa and Sumitra [121] combined cuckoo search optimization with a K-means clustering algorithm to achieve an optimal global solution in an intrusion detection system.

3.1.10. Differential Evolution

The differential evolutionary (DE) algorithm is a powerful and efficient population-based optimization algorithm based on evolutionary theory. It is presented as a floating-point encoding evolutionary algorithm for minimizing possibly nonlinear and non-differentiable continuous space functions [122,123]. Kwedlo [124] introduced DE-KM, a combination of differen-
tial evolution algorithm and K-means clustering algorithm. The mutation and crossover operation of DE generates each candidate solution, which is then fine-tuned using the K-means algorithm. Cai et al. [125] proposed a hybrid of DE and one-step K-means algorithm termed CDE (clustering-based DE) for solving unconstrained global optimization problems. The one-step K-means was introduced to enhance DE performance by acting as several multi-parent crossover operators to utilize the population information efficiently. Kuo, Suryani, and Yasid [126] proposed ACDE-K-means integrating automatic clustering based differential evolution algorithm with K-means algorithm seeking to improve ACDE algorithm’s performance by the use of the K-means algorithm for tuning the cluster centroids.

Sierra, Cobos, and Corrales [127] hybridized the K-means clustering algorithm and DE for continuous optimization using the DE operators to work on the groups generated by the K-means algorithm for better diversification and escaping from local convergence. Hu et al. [128] proposed an improved K-means clustering algorithm using a hybrid of DE and FOA (fruit fly optimization algorithm) embedded into K-means. Wang [129] proposed a weighted K-means algorithm based on DE with an initial clustering center and strong global search capability. Silva et al. [130] used a u-control chart (UCC) to automatically determine the k activation threshold for ACDE with the cluster number generated serving as the specified k value for the K-means algorithm, thus improving the performance of the clustering algorithm. Sheng et al. [131] presented a combination of differential evolution algorithms with adaptive niching and K-means termed DE-NS-AKO for partitional clustering. The K-means-based adaptive niching adjusts each niche size to avoid premature convergence. As reported earlier, Bonab et al. [97] presented a combination of DE with a modified K-means algorithm with ABC for color image segmentation. Mustafi and Sahoo [132] explored the combination of GA and DE to find the original seed point and determine the required cluster numbers for the traditional K-means algorithm to reduce the possibility of its convergence into local optimal.

3.1.11. Invasive Weed Optimization

The invasive weed optimization (IWO) proposed by Mehrabian and Lucas in 2006 [133] is a stochastic optimization algorithm that was inspired by a common agricultural phenomenon of invasive weeds colonization. IWO has a powerful exploitative and explorative capability [134]. Fan et al. [134] proposed a clustering algorithm framework for hybridizing IWO with a K-means algorithm to improve the performance of the traditional K-means algorithm. Pan et al. [135] presented a clustering algorithm combining IWO and K-means based on the cloud model—CMIWOKM. The cloud model-based IWO directs the K-means algorithm iterative search operation to ensure a definite evolution direction to improve the proposed algorithm’s performance. Boobord, Othman, and Abubakar [136] proposed a WK-means hybrid clustering algorithm combining IWO and K-means clustering. In WK-means, the initial solutions for the K-means algorithm are generated by the IWO algorithm. They further proposed hybridized clustering algorithm PCAWK adopting principal component analysis method to reduce redundant dimensionality of a real-world dataset and employed their WK-means algorithm to generate optimal clusters from the dataset [136]. Razi [137] presented a hybridization of IWO and DEA-based K-means algorithm for facility location problems where K-means was used for maintenance stations clustering while a zero-one programming model based on IWO was used to conduct the Pareto analysis of rank and distance.

3.1.12. Imperialist Competitive Algorithm

The imperialist competitive algorithm (ICA) is an evolutionary optimization algorithm inspired by imperialistic competition [138]. Niknam et al. [139] proposed a robust and efficient hybrid evolutionary clustering algorithm called hybrid K-MICA. K-MICA is a combination of K-means clustering algorithm and modified imperialist competitive algorithm where MICA is used to generate the population and form the initial empires;
the K-means algorithm is then used to improve the empire’s colonies and imperialists’
positions, which are then fed back into MICA. Abdeyazdan [140] presented ICAKHM, a
hybridization of modifier imperialist competitive algorithm and K-harmonic means to
solve the problem of local optimum convergence of the K-harmonic means. Emami and
Derakhshan [65] proposed ICAFKM combining imperialist competitive algorithm with
fuzzy K-means to assist the regular FKM escape from converging into local optimum and
increase convergence speed.

3.1.13. Harmony Search

The harmony search (HS) is a metaheuristic optimization algorithm that imitates
musicians’ music improvisation process of searching for a perfect state of harmony [141].
Forsati et al. [141] presented a pure HS clustering algorithm for a globalized search in the
solution space. The proposed HSCLUST was then hybridized with a K-means clustering
algorithm in three different modes to avoid the problem of initial parameter dependence of
the K-means algorithm. Each proposed hybridization depended on the stage at which the
K-means algorithm is performed in the clustering process. Mahdavi and Abolhassani [142]
proposed harmony K-means (HKA) based on an HS optimization algorithm for document
clustering for faster global optimum convergence. Cobos et al. [143] hybridized the K-
means algorithm with global best HS, frequent term sets, and Bayesian information criterion
termed IGBHSK for automatic Web document clustering. The Global-Best HS performs
the global search in the solution space while the K-means algorithm seeks the optimum
value in the local search space. Chandran and Nazeer [144] proposed an enhanced K-
means clustering algorithm based on hybridization of the K-means with improved HS
optimization technique for finding global optimum solutions. Nazeer, Sebastian, and
Kumar [145] presented HSKH—harmony search K-means hybrid for gene expression
clustering, which produced a more accurate gene clustering solution. Raval, Raval, and
Valiveti [146] proposed a combination of HS and K-means for optimizing wireless sensor
network clustering. The HS was used to generate the initial solution, which is then fed into
the K-means algorithm for a more precise solution. Kim et al. [147] proposed a scheme for
load balancing with switch migration for the distributed software-defined network (SDN)
employing a combination of HS and K-means for clustering the switches.

3.1.14. Blackhole Algorithm

The phenomenon of the black hole in astrophysics inspired the design of the blackhole
(BH) algorithm. During optimization, the best candidate acts as the black hole in each
iteration and pulls other candidates to itself [148]. It does not require manual parameter
setting [149], and it lacks the capability for exploring the search space [150]. Eskandarzadeh-
halamdary et al. [151] proposed BH-BK comprising blackhole and bisecting K-means
algorithms for precise clustering and global optimal convergence with local refinement.
Pal and Pal [152] hybridized the K-means clustering algorithm with the BH optimization
approach for data clustering. Some better results from the K-means algorithm are used in
initializing a portion of the population while the rest are randomly initialized. The BH al-
gorithm was used by Feng, Wang, and Chen [153] in determining the K-means algorithm’s
initial centroids for their proposed new clustering method for Image classification based
on the improved spatial pyramid matching model.

3.1.15. Membrane Computing

Membrane computing (MC) is a P system classified under a distributed parallel com-
puting model [154]. A K-means clustering method based on the P system and DNA genetic
was proposed by Jiang, Zang, and Liu [155]. The initial cluster center was analyzed using
DNA encoding, and the clustering was realized using the P system. Zhao and Liu [156]
proposed a GKM-genetic K-means membrane-clustering algorithm combining genetic
K-means algorithm and membrane computing for clustering multi-relational dataset har-
nessing the benefit of the P system parallelism with the K-means algorithm local search.
capability and the good convergence of the GA. Weisun and Liu [157] proposed a new P system hybridized with a modified differential evolution K-means algorithm to improve the K-means algorithm’s initial centroids.

Zhao, Liu, and Zhang [158] constructed a P system for solving the K-medoids algorithm providing a new idea for great parallelism and lower computational time complexity for cluster analysis. Wang, Xiang, and Liu [159] designed a tissue-like P system for their proposed hybrid algorithm of K-medoids and K-means algorithms. The K-means algorithm is used to obtain the elementary clustering result, and the K-medoids is then used to optimize the results. The tissue-like P system creates a parallel platform for the execution, thus efficiently improving the computational time. Wang, Liu, and Xiang [160] proposed an effective method for initial centroid selection for the K-means algorithm, which incorporates a tissue-like P system to avoid the boundedness of the K-means initialization method.

3.1.16. Dragonfly Algorithm

The dragonfly algorithm (DA) is inspired by the natural static and dynamic swarming behaviors of dragonflies. In DA, the exploration and the exploitation phases are modeled using the dragon flies social interaction in their navigation, food searching, and enemy avoidance while swarming statically or dynamically [161]. Angelin [162] proposed a dragonfly-based K-means clustering combined with a multi-layer feed-forward neural network for outlier detection using an optimization-based approach. Kumar, Reddy, and Rao [163] combined the fuzzy c-means algorithm with the wolf hunting-based dragonfly to detect change in synthetic aperture radar (SAR) images.

3.1.17. Ant Lion Optimizer

The ant lion optimizer (ALO) is inspired by the hunting mechanisms of antlions in nature. It involves five main steps: ants’ random walks, traps building, entrapments in traps, prey catching, and traps rebuilding. Majhi and Biswal [164] proposed a K-means clustering algorithm with ALO for optimal cluster analysis, which performed better in terms of F-measure and sum of intra-cluster distances. Chen et al. [165] combined quantum-inspired ant lion optimizer with K-means algorithm to propose QALO-K, an efficient hybrid clustering algorithm. Murugan and Baburaj [166] integrated improved K-medoids with ant lion optimizer and PSO to proposed ALPSOC, which can obtain optimized cluster centroid with improved clustering performance while preserving the computational complexity. Naem and Ghali [167] proposed a hybridized clustering algorithm termed K-median modularity ALO that combined K-median with ant lion optimizer to handle the problem of community detection in the social network. Dhand and Sheoran [168] proposed a secure multi-tier energy-efficient routing protocol (SMEER) that combined an ant lion optimizer (as cluster head selector) with a K-means algorithm (for clustering).

3.1.18. Social Spider Algorithm

The social spider optimization (SSO) algorithm was proposed by Cuevas in 2013, simulating the cooperative behavior of social spiders based on the biological laws of a cooperative colony [169]. Chandran, Reddy, and Janet [170] proposed a hybrid of social spider optimization and K-means termed SSOKC to speed up the clustering process of SSO. Thiruvenkatasuresh and Venkatachalam [171] employed the fuzzy c-means clustering process, which adopted the social spider optimization technique with GA for finding optimized cluster centroid in their proposed brain tumor images segmentation process.

3.1.19. Fruit Fly Optimization

The fruit fly (FFO) is inspired by the fruit fly’s foraging behavior in nature [172]. A hybrid of K-means and fruit fly optimization termed Kmeans-FFO was proposed by Sharma and Patel [173] for optimal clustering quality. Jiang et al. [174] used a fruit fly algorithm and K-means clustering algorithm to optimize earthquake rescue center site selection and layout. Gowdham, Thangavel, and Kumar [175] proposed using the fruit fly
algorithm to select the initial cluster centroid for the k-means clustering algorithm in finding
the optimal number of clusters in a dataset. Hu et al. [128] proposed DEFOA-K-means,
an improved K-means clustering algorithm that uses a hybrid of fruit fly optimization
algorithm and differential evolution (DEFOA) for optimal cluster solutions that are not
zero. Wang et al. [176] proposed FOAKFCM, a kernel-based fuzzy c-means clustering
based on fruit fly algorithm where the initial cluster center is determined using the fruit fly
algorithm first, and then the kernel-based fuzzy c-means is applied in classifying the data.

3.1.20. Bees Swarm Optimization

The bees swarm optimization (BSO) is a swarm-intelligence-based optimization al-
algorithm inspired by the foraging behavior of bees such that a swarm of bees cooperates
together in finding a solution to a problem [177]. Djenouri, Belhadi, and Belkebir [178]
used the combination of the K-means algorithm and bee swarm optimization in document
information retrieval. The K-means algorithm generates similar clusters from the collec-
don document, while the BSO was used to deep explore the document clusters. Aboubi,
Drias, and Kamel [179] proposed BSO-CLARA for clustering large datasets combining
K-medoids clustering and bees swarm optimization behavior. Djenouri, Habbas, and
Aggoune-Mtalaa [180] used the K-means clustering algorithm as a decomposition tool
in their proposed improved version of the BSO metaheuristic, termed BSOGD1, which
incorporates the decomposition method for solving the MAX-SAT problem.

3.1.21. Bacterial Colony Optimization

The bacterial colony optimization (BCO) algorithm is inspired by the basic growth
law of bacterial colonies [181]. It requires a high computational cost for completing a given
solution. Revathi, Eswaramurthy, and Padmavathi [182] hybridized the K-means clustering
algorithm with BCO to produce a BCOKM clustering algorithm for better cluster partition
with reduced computational cost compared with BCO clustering. The BCO searches for the
global optimum solution in the search space and then hands the clustering process to the
K-means algorithm. Vijayakumari and Deepa [183] combined the fuzzy c-means algorithm
with the fuzzy BCO (FBCO) to propose a hybrid fuzzy clustering algorithm (HFCA) for
higher cluster analysis performance.

3.1.22. Stochastic Diffusion Search

The stochastic diffusion search (SDS) is a multi-agent global search and swarm intel-
ligence optimization algorithm based on simple iterated agents’ interactions [184]. The
strong mathematical framework of the SDS algorithm describes its behavior in relation to
resource allocation, global optimum convergence, and linear time complexity with robust-
ness and criteria for minimal convergence. Karthik, Tamizhazhagan, and Narayana [185]
proposed a stochastic diffusion search K-means clustering technique named ‘scattering
search K-means’ (SS-K means) for locating optimal clustering points for the identification
of points of data leakage in social networks.

3.1.23. Honey Bee Mating Optimization

The honey bee mating optimization (HBMO) is a swarm-based optimization algorithm
inspired by the natural process of real honey bees mating [186]. Teimoury et al. [187] hy-
bridized K-means with the honey bee mating algorithm to resolve the problems associated
with the K-means clustering algorithm to improve the performance of the clustering algorithm.
Aghaeabrahimi, Golkhandan, and Ahmadnia [188] combined the K-means algorithm with
HBMO to solve the problem of localization and sizing of flexible AC transmission systems
(FACTS) in a power system to reduce the generation, transmission, and power costs.

3.1.24. Cockroach Swarm Optimization

The cockroach swarm optimization (CSO) is a swarm intelligence algorithm inspired
by the social behavior of cockroaches mimicking their ruthless social behavior, chase
swarming, and dispersion [189]. Senthilkumar and Chitra [190] combined the K-means algorithm and cockroach swarm optimization (MCSO) in their proposed novel hybrid heuristic–metaheuristic load balancing algorithm for IaaS-cloud computing resource allocation. K-means clustering was used to cluster the files into small chunks to reduce the time required for file download, while the MCSO was employed in measuring the load ratio.

3.1.25. Glowworm Swarm Optimization

The glowworm swarm optimization (GSO) is a nature-inspired optimization algorithm based on lighting worms’ natural behavior, which controls their light emission using it for different purposes [191]. K-means algorithm was combined with basic glowworm swarm optimization by Zhou et al. [192] for their proposed novel K-means image clustering algorithm based on GSO termed ICGSO to effectively override the problems inherent in the K-means algorithm and produce better clustering qualities. Onan and Korukoglu [193] presented a cluster analysis approach based on GSO and K-means clustering algorithms. Tang et al. [194] hybridized the k-means algorithm with an improved GSO self-organizing clustering algorithm for automatic cluster analysis with better cluster quality.

3.1.26. Bee Colony Optimization

The bee colony optimization (BCO) is a swarm-intelligence-based algorithm that simulates the bees’ autonomy and self-organizing with distributed functioning behavior [195]. The intelligence of collective bees’ is explored in BCO for possible applications in finding the solution to combinatorial problems which are characterized by uncertainty. Das, Das, and Dey [196] integrate the K-means algorithm and modified bee colony optimization algorithms producing MKCLUST and KMCLUST to improve the performance of MBCO in terms of global optimum convergence and diverse clustering solutions. In MKCLUST, the K-means algorithm was used to fine-tune MBCO explorative power further, while in the KMCLUST, the local optimal problem of K-means was dealt with improving the exploration capability and solution’s diversity. Four different K-means algorithms with BCO algorithm hybrids were proposed by Forsati, Keikha, and Shamsfard [197] which solved the problem of local optimum convergence for large and high dimensional datasets.

3.1.27. Symbiotic Organism Search

The symbiotic organism search (SOS) is a nature-inspired metaheuristic algorithm based on the three symbiosis relationships mechanism often employed by the individual prescribed for survival in the ecosystem. These relationship behaviors include mutualism, commensalism, and parasitism, denoting the biological interactions between organisms. SOS has only one control parameter, which makes its implementation easier compared with other metaheuristic optimization approaches. In Yang and Sutrisno [198], automatic K-means was applied to symbiotic organisms search algorithm initial solution for the creation of subpopulation which enhances the quality and efficiency of searching. The sub-ecosystem created through the automatic K-means enables the CSOS algorithm to combine the local and global searches on the dataset.

3.2. RQ2. Which of the Reported Hybridization of Nature-Inspired Meta-Heuristics Techniques with K-Means Clustering Algorithm Handled Automatic Clustering Problems?

Table 1 presents the summary of the reviewed literature on hybridized algorithms. It includes a hundred and forty-seven (147) hybridized K-means with 28 different MOA clustering algorithms. The fifth column indicates the characteristic of each hybridized clustering algorithm as either automatic or non-automatic. The role of the corresponding MOA and K-means algorithms in the hybridized algorithms was stated in columns eight and nine, respectively. In contrast, columns ten and eleven, respectively, report the dataset used for the algorithm testing and the criteria for their performance measure. From the 147 reviewed articles, only 23 K-means/MOA hybrid algorithms addressed the issue of automatic data clustering.
Table 1. List of hybridized algorithms combining K-means algorithm with various MOA.

| Metheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|------------------------|-----------|-------------|---------------------------------|----------|--------------|------------------------|---------------|---------------------|
| Genetic Algorithm (GA) |           |             |                                 |          |              |                        |               |                     |
| 1 Zhou et al., [45]-NoiseClust | Niche Genetic Algorithm (NGA) and K-means++ | Automatic | Global Positioning System | Density-Based Method | Adaptive probabilities of crossover and mutation |                       |               |                     |
| 2 Dai, Jiao & He, [28]-PGAClust | Parallel Genetic Algorithm (PGA) and K-means | Automatic | Dynamic mining of cluster number | | | | |               |                     |
| 3 Li et al. [40] | GA and K-means | Automatic | Adopting a k-value learning algorithm using GA | | | | |               |                     |
| 4 Kuo et al. [39]-SOM+GKA | SOM and Modified GKA | Automatic | Market Segmentation in Electronic Commerce | Self-Organizing Feature Maps (SOM) neural networks | | | | |                     |
| 5 Eshlaghy & Razi [42] | Grey-based K-means and GA | Non-Automatic | Research and Project selection and management | Project allocation selection clustering of different projects | | | | | SSE |
| 6 Sheng, Tucker & Liu [31]-NGKA | Niche Genetic Algorithm (NGA) + one step of K-means | Non-Automatic | Improving GA optimization procedure for general clustering | Gene Expression | | | | | Sum of Square Error (SSE) |
| 7 Karegowda et al. [41] | K-means + GA | Non-Automatic | Avoidance of random selection of cluster centers | Medical Data Mining | initial cluster center assignment | Clustering of dataset | | Classification error and execution time |
| 8 Bandyopadhyay & Maulik [24]-KGA | GA + K-means | Non-Automatic | Escape from local optimum convergence | Satellite image classification | GA perturb the system to avoid local convergence | | | | |
| 9 Cheng et al. [25]-PGKA | Prototype Embedded GA + K-Means | Non-Automatic | Encoding Cluster prototypes for general GA clustering | | | | | | Ga based criteria |
| 10 Zhang & Zhou [35]-Nclust | novel niching genetic algorithm (NNGA) + K-means | Automatic | Finding better cluster number automatically | Improved canopy and K-means ++ | | | | | UCI dataset, GAK and GenClust | SSE, DBIPBM, COSEC, ARI and SC |
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|----------|--------------|-------------------------|---------------|---------------------|
| Ghezelbash, Maghsoudi & Carranza [38] | Genetic K-means clustering + Traditional K-means algorithm | General Improvement of K-Means | Non-Automatic | Geochemical Anomaly Detection from stream sediments | Determining cluster center locations | General Clustering | GSI Analytical Data | Traditional K-Means Clustering (TKMC) | Prediction rate curve based on a defined fitness function |
| Mohammadrezaeppou, Kisi & Pourahmadm [46] | K-means + Genetic Algorithm and Fuzzy C-Means + Genetic Algorithm | Avoidance of random selection of cluster centers | Automatic | Homogeneous regions of groundwater quality identification | GA method | Determining the optimum number of clusters | General Clustering | | Av. Silhouette width Index, Levene’s homogeneity test, Schuler & Wilcox classification; Piper’s diagram |
| El-Shorbagy et al. [37] | K-means + GA | Combining the advantages of K-means and GA in general clustering | Non-Automatic | Electrical Distribution system | GA Clustering with a new mutation | GA population Initialization for best cluster centers | UCI dataset | K-means Clustering and GA-Clustering |
| Barekatain, Dehghani & Pourazaferani [44] | K-means + Improved GA | New cluster-based routing protocols | Automatic | Energy Reduction and Extension of network Lifetime | Determine the optimum number of clusters | Dynamic clustering of the network | Nc2Network (Fedora10) | Other network routing protocols, i.e., LEACH, GABEEC and GAEEP |
| Lu et al. [43] | K-means + GA | Combining the advantages of the two algorithms | Non-Automatic | Multiple Travelling Salesman Problem | | | | |
| Sinha & Jana [33] | GA with Mahalanobis distance + K-Means with K-means++ Initialization clustering algorithm for distributed dataset | Automatic | | GA method | Initial clustering using GA method | Fine-tuning the result obtained from GA clustering | Breast Cancer (Iris Glass Yeast) | Map Reduced-based Algorithms, MK-means, parallel K-means and scalingGA | Davies-Bouldin index, Fisher’s discriminant ratio, Sum of the squared differences |
| Laszlo & Mukherjee [26] | GA + K-means | improving GA-based clustering method | Non-Automatic | Evolving initial cluster centers using hyper-quad tree | To return the fitness value of a chromosome | CTD, RTD, BPZ, TSP-LIB-1060, TSP-LIB-R38; Simulated dataset | GA-based clustering, J-Means |
| Zhang, Leung & Ye [198] | GA + K-means | Improving accuracy | Non-Automatic | Credit Scoring | Reduce data attribute’s redundancy | Removal of noise data | German credit dataset, Australian credit dataset | C4.5, BPN, GP, SVM+GA and RSC |
| Kapil, Chawla & Ansari [3] | K-means + GA | Optimizing the K-means | Automatic | General Cluster Analysis | Generating initial cluster centers | Basic K-means clustering | Online User dataset | K-means SSE |
| Rahman & Islam [52]-GENCLUST | GA + K-means | Finds cluster number with high-quality centers | Automatic | General Cluster Analysis | Deterministic selection of Initial Genes | Generating initial cluster centers | Basic K-means clustering | UCI dataset | RUDAW, AGCUK, GAGR, GFCM and SABC | Xie-Beni Index, SSE, COSEC, F-Measure, Entropy, and Purity |
Table 1. Cont.

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|----------|--------------|--------------------------|---------------|---------------------|
| 21 | Islam et al. [34] GENCLUST++ | GA + K-means | Computational complexity reduction | Automatic | General Cluster Analysis | GA method | Generating initial cluster centers | Basic K-means clustering | UCI dataset | GENCLUST-H, GENCLUST-E, AGCUC, GAGR, K-Means | GA Performance Measure |
| 22 | Mustafi & Sahoo [36] | GA + DE + K-means | Choice of initial centroid | Automatic | Clustering Text Document | Using DE | Generating initial cluster centers | Basic K-means clustering | Corpus | K-Means | Standard validity parameters |
| 23 | Laszlo & Mukherjee [27] | GA + K-means | Superior Partitioning | Non-Automatic | | | | | German credit dataset | Australian credit dataset |
| 24 | Patel, Raghuwanshi & Jaiswal [20] GAS3KM | GAS3 + K-means | Improving the performance of GAS3 | Automatic | GA method | Generating initial cluster centers | Basic K-means clustering | | Unconstrained unimodal and multi-modal functions with or without epistasis among n-variables. | GAS3 | GA Performance Measure |
| 25 | Xiao, Yan, Zhang, & Tang [31] KMVGA | K-means + QGA | Quantum inspired GA for K-means | Automatic | GA method | Generating initial cluster centers | Basic K-means clustering | | Simulated datasets | Glass, Wine, SPECTF-Heart, Iris | Davies–Bouldin rule index |
| 26 | Jie and Yibo [70] | PSO + K-means | Outlier detection | Non-Automatic | Distribution Network Sorting | Optimizing the clustering center | Determining the optimal number of Clusters | | Simulated datasets | K-Means | SSE |
| 27 | Tsai & Kao [63] KSRPSO | Selective Regeneration PSO + K-means | SRPSO Performance improvement | Non-Automatic | Global optimal convergence | Basic K-means clustering | Artificial datasets, Iris, Crude oil, Cancer, Vowel, CMC, Wine, and Glass | SRPSO, PSO and K-Means | Sum of intra-cluster distances and Error Rate (ER) |
| 28 | Paul, De & Dey, [60] MIPSO based K-Means | MIPSO + K-Means | Improved multi-dimensional data clustering | Non-Automatic | Cluster center generation | Basic K-Means clustering | Iris, Wine, Seeds, and Abalone | K-Means and Chaotic Inertia weight PSO | DBI, SI, Means, SD and computational time, ANOVA test and a two-tailed t-test conducted at 5% significance |
| 29 | Prabha & Visalakshi [64] PSO-K-Means | PSO + Normalisation + K-Means | Improving performance using normalization | Non-Automatic | Global optimal convergence | Basic K-means clustering | Australian, Wine, Bupa, Mammography, Satellite Image, and Pima Indian Diabetes | PSO-KM, K-Means | Rand Index, FMeasure, Entropy, and Jacquard Index |

Particle Swarm Optimisation (PSO)
Table 1. Cont.

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|----------|--------------|-------------------------|--------------|---------------------|
| 30 Ratanavilisagul [69]-PSOM | PSO + K-Means + mutation operations applied with particles | Avoidance of getting entrapped in local optima | Non-Automatic | Global optimal convergence | Basic K-means clustering | Iris, Wine, Glass, Heart, Cancer, E.coli, Credit, Yeast | Standard PSO, PSOFKM, PSOLF-KHM | F-Measure (FM), Average correct Number (ACN), and Standard Deviation (SD)of FM. |
| 31 Nayak et al. [66] | Improved PSO + K-Means | optimal cluster centers for non-globular clusters | Non-Automatic | Global optimal convergence | Basic K-means clustering | K-Means, GA-K-Means, and PSO-K-Means | | |
| 32 Emami, & Derakhshan [65]-SOFKM | PSO + FKM | Escape from local optimum with increased convergence speed | Non-Automatic | Global optimal convergence | Fuzzy K-means clustering | FKM, ICA, PSO, PSOKHM, and HABC algorithms | Sample, Iris, Glass, Wine, and Contraceptive Method Choice (indicated as CMC) | F-Measure (FM) and Runtime Metrics |
| 33 Chen, Miao & Bu [71] | PSO + K-Means | Solving initial center selection problem and escape from local optimal | Non-Automatic | Global optimal convergence | Dynamic clustering using k-means algorithm | Lena, Tree and Flower images from the Matlab Environment | K-Means PSOK | Sphere function and Griewank function |
| 34 Niu et al. [48] | Six different PSOs with different social communications + K-means | Escape from local optimum with accelerated convergence speed | Non-Automatic | Global optimal convergence | Refining partitioning results for accelerating convergence | Iris, Wine, Coil2, Breast Cancer, German Credit, Optdigits, Musk, Magic-04, and Road Network with synthetic datasets | PSC-RCE, MacQuens K-Means, ACA-5L, ACA-CL, ACA-AI, and Lloyd’s K-means | Mean squared error (MSE—sum of intra-cluster distances) |
| 35 Yang, Sun & Zhang [62]-PSOKHM | PSO + KHM | Combining the merits of PSO and KHM | Non-Automatic | Global optimal convergence | Refining cluster center and KHM clustering | Artificial datasets, Wine, Glass, Iris, breast-cancer, Wisconsin, and Contraceptive Method Choice. | KHM, PSO | Objective function of KHM and F-Measure |
| 36 Chen & Zhang [59]-RVPSO-K | PSO + K-means | Improved stability, precision, and convergence speed | Non-Automatic | A parallel search for optimal clustering | Refining cluster center and K-means clustering | Two-day Web log of a university website | PSO-K | Fitness Measure and Run-time |
| 37 Niknam, & Amiri, [53]-APSO-ACO-K | Fuzzy Adaptive PSO + ACO + K-means | Solving non-linear partitioning clustering problem | Non-Automatic | Provide Initial state for K-means algorithm | Basic K-means clustering | Artificial datasets, Iris, Wine, Vowel, Contraceptive Method Choice (CMC), Wisconsin breast cancer, and Ripley’s glass | PSO, ACO, SA, PSO-SA, ACO-ACO, GA, TS, HBMO, and K-means | Total mean-square quantization error and F-Measure |
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing Compared with Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|-----------|--------------|----------------------------------------------------------|
| 38 Kao, Zahara & Kao [60]-K-NM-PSO | Effective global convergence | Non-Automatic | Providing more accurate clustering | Provide initial seeding | Artificial datasets, Vowel, Iris, Crude-oil, CMC, Cancer, Glass, and Wine | PSO, NM-PSO, K-PSO, and K-means; Sum of the intra-cluster distances and the Error Rate |
| 39 Omran, Salman & Engelbrecht [58]-DCPSO | An automatic clustering with reduced effect of initial conditions | Automatic | Image Segmentation | Binary PSO optimization | Basic PSO clustering | Refine cluster center | Lenna, mandrill, jet, peppers, one MRI, and one satellite image of Lake Tahoe; GA and Random Search Dynamic Clustering | Dunn’s index, Validity index proposed by Turi, S_Dbw validity index |
| 40 Van der Merwe & Engelbrecht [49] | Improving the performance of PSO | Non-Automatic | PSO clustering | Initial Seeding for PSO | Artificial datasets Iris, Wine, Breast cancer, and Automobiles | K-Means and PSO | The Quantization error, the intra-cluster distances and the inter-cluster distances |
| 41 Kao & Lee [61]-KCPSO | Discretize PSO to optimize cluster numbers | Automatic | Optimizing the number of clusters | Basic K-means clustering | Artificial datasets, Iris, and Breast Cancer | DCPSO and GC-LR | DB index |
| 42 Mathew & Vijayakumar [75]-MPKM | Parallelization of K-Means | Non-Automatic | Initial optimal cluster centroid | Refine optimized centroid | Wisconsin diagnostic breast cancer, Wine, Glass, and Credit data | Parallel K-means | Accuracy, SSW, SSB, DBI, DIJ and SC. |
| 43 Jitpakdee, Aimmanee & Uyyanonvara [73]-FA-K | Hybrid-clustering-based color quantization | Non-Automatic | Initial cluster centroids and Global optimal convergence | Refine initial centroids | Three images from USC-SIPI Image Database (Lena, Peppers, and Mandrill) | FA and K-Means | Mean Square Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) |
| 44 Kuo & Li [80] | Wavelet Transform + FA based K-Means + FA based SVR | Non-Automatic | Noise detection and Normalisation | Basic clustering | GA-SVR, PSO-SVR, FA-SVR and DE-SVR | Mean Square Error (MSE) |
| 45 HimaBindu et al. [83] | Improved Big Data Clustering | Non-Automatic | Generate initial cluster centroids | Basic K-means clustering | Iris Plants database, Glass, Wine, two microarray information indexes and Artificial datasets. | K-Means, K-Means++ Total computation time, centroid selection time and accuracy |
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|------------|-------------------------------|----------|--------------|------------------------|--------------|---------------------|
| 46 Langari et al. [82]-KFCFA | K-member Fuzzy clustering + FA | A combined anonymizing algorithm | Non-Automatic | Social Network Privacy Preservation | Optimizing the primary clusters | Social network databases from Facebook, Twitter, Google + and YouTube | K-Means + Bat K-Means + Cuckoo, K-Means++, K-Means, Farthest First and Canopy | CCI, TP, FP Precision, Recall, F-Measure, ROC and Time to build training model. |
| 47 Kaur, Pal & Singh [81] | K-means + Firefly | IDS training model for data classification | Non-Automatic | Intrusion Detection | Initialize method for the K-Means | NSL-KDD dataset | | |
| 48 Nayak et al. [77] | Optimized K-means with firefly and Canopies | A hybrid algorithm for classification | Non-Automatic | | | | Haberman’s survival dataset | K-means algorithm | Classification accuracy |
| 49 Xie et al. [74]-IIEFA and CIEFA | K-means + Improved Firefly | resolve initialization sensitivity and local optimal traps | Non-Automatic | FA Clustering | Generate Initial Cluster Models as seed solution | ALL-IDB2 database Sonar, Ozone, Wbc1, Wbc2, Wine, Iris, Balance, Thyroid, and E. coli | GA, ACO, K-means, FA, DA, SCA, CIA, CFA, NFA, VSSFA, and MFA | Sum of intra-cluster distances Av.accuracy, Av.sensitivity, Av.specificity & macro-average F-score |
| 50 Wu et al. [84]-Kmeans-FFA-KELM | K-means + FFA + Kernel Extreme Learning Machine Model | non-Automatic | Evapotranspiration Estimation | Building various sub-models | Decomposition of training dataset into multiple subsets | meteorological data (Tave, Tmax, Tmin), wind speed, relative humidity and sunshine | FFA-KELM | Coefficient of determination, RMSE, MAE, SI and NSE |
| 51 Behera et al. [77]-FCM-FA | Fuzzy C-Means + FireFly Algorithm | tackles Fuzzy C-Means problems | Non-Automatic | | | | | |
| 52 Nayak, Naik & Behera [79]-FA-K-means | Firefly + K-means | global search capacity for K-means | Non-Automatic | | | | | |
| 53 Hassanzadeh & Meybodi [74]-K-FA | K-means + Firefly | Finding initial centroids | Non-Automatic | Finding initial centroids | Refining the centroids | Standard data set from UCI (Iris, WDBC, Sonar, Glass, and Wine) | K-means, PSO, KPSO | Intra-cluster distance and clustering error |
| 54 Tripathi, Sharma & Bala [87]-DBPKBA | Bat algorithm + K-means | the parallelized approach in a distributed environment | Non-Automatic | | | | | Best and Average intra-cluster distance |
Table 1. Cont.

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|-----------|--------------|--------------------------|--------------|---------------------|
| Chaudhary & Banati [90]-HESB | EShBAT + K-medoids + K-means | Leveraging optimisation capabilities | Non-Automatic | Dividing populations into groups | starting population and refining solutions | | BA, EShBAT, K-medoids | |
| Gan, & Lai [99]-KMBA | K-means + Bat Algorithm | Classification of EBN | Non-Automatic | EBN Classification | Basic Bat Algorithm Clustering | Initiating Initial points for BA | Three classes of data (Grades AA, A, and B) | Classification Accuracy |
| Payez, Altimiras, & Villavicencio [88] | K-means + Binary Bat Algorithm | demonstrate K-means technique utility in binarization | Non-Automatic | Multidimensional Backpack Problem | | | |
| Sood & Bansal [86] | K-Medoids + Bat Algorithm | Automatic | | Generating initial cluster center for K-Medoids | K-Medoids clustering | K-Medoids | |
| Jensi & Jiji [92]-FPAKM | K-means + FPA | Combining the advantages of the two algorithms | Non-Automatic | Provide Initial seedlings for K-means | K-means Clustering | Artificial dataset iris, thyroid, wine, CMC, crude oil, and glass | FPA, K-means | Mean-square quantization error (MSE) |
| Kumari, Rao & Rao [93] | K-means + FPA | optimum solutions in Image compression | Non-Automatic | Image Compression | | | Peak signal to noise ratio (PSNR), mean square error (MSE) and fitness function. |
| Armano & Farmani [95]-AABC | K-means + ABC | finding a global optimum solution | Non-Automatic | ABC Clustering | Use K-means for initial seedlings | Iris, Wine, and Contraceptive Method Choice (CMC) | K-means | Distortion Criterion, Computational Cost, SD and F-Measure |
| Cao & Xue [102]-MABC-K-means | Modified ABC + K-means | hybridized framework for cluster analysis. | Non-Automatic | Customer Relationship Management | Provide initial cluster center | Basic K-means Clustering | Simple dataset of customers and their orders of an e-commerce platform in the first quarter. | Differential Evolution algorithm (DEA), standard Genetic algorithm (GA) and standard Artificial Bee Colony algorithm (ABC) | The mean and variance of Griewank, Rastrigin, Rosenbrock, Ackley and Schwefel functions. |
| Wang et al. [101]-ABC-KM | ABC + K-means | Improving the effectiveness of Wind farm clustering | Non-Automatic | Modeling of Farms with DFIGs | Provide initial cluster center | Basic K-means Clustering | MW DFIG | K-means | Wind speed disturbances and short-circuit faults |
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing Compared with Performance Measure |
|-------------------------|-----------|-------------|--------------------------------|-----------|--------------|----------------------------------------------------------|
| 64 Huang [100] ABC + Accelerated K-means | Non-Automatic | Colour Image Quantization | Provide initial cluster center | Basic K-means Clustering | Lena, Baboon, Lake, Peppers, and Airplane with a size of 512 × 512 | SFLA-CQ Average mean square error, the standard deviation, and average computation time. |
| 65 Tran et al. [96] Enhanced ABC + K-means | Non-Automatic | colour Image Quantization | Generate initial cluster center | Basic K-means Clustering | Artificial datasets (Iris, Wine, Glass, E.coli, Liver disorder, Vowel, Fiona, WDBC, and CMC) | ABC, CABC, K-means, HABC, K-means++ and FAPSO-ACO-K Mean square error (MSE) and Euclidean distance. |
| 66 Bonab et al. [97] Modified K-means + ABC + DE | Non-Automatic | colour Image Quantization | Generate initial points for K-means clustering | Iris, Balance-Scale, Wine, E.coli, Glass, Abalone, Musk, Pendigits, Skin Seg, CMC, and Cancer | ABC, IABC, HABC, CABC, DFSABCelite and PSO+K-means Sphere, Rosenbrock, Rastrign, Alpine and Ackley |
| 67 Jin, Lin & Zhang [98] -CAABC-K-means | Non-Automatic | Colour Image Quantization | Generate initial points for K-means clustering | Iris, Balance-Scale, Wine, E.coli, Glass, Abalone, Musk, Pendigits, Skin Seg, CMC, and Cancer | ABC, IABC, HABC, CABC, DFSABCelite and PSO+K-means Sphere, Rosenbrock, Rastrign, Alpine and Ackley |
| 68 Dasu, Reddy & Reddy [99] K-means + ABC | Non-Automatic | Colour Image Quantization | Classification | Remote sensing Images | PSO Sensitivity, Specificity, Overall accuracy and Kappa Coefficient. |
| 69 Pambudi, Budhuradun & Wacukowo [100] GWO-K-means | Non-Automatic | Colour Image Quantization | Generate initial points for K-means clustering | Brain MRI K-means | Sum of Square Error (SSE) |
| 70 Korayem, Khoras & Kassirn [105] K-GWO | Non-Automatic | Colour Image Quantization | Generate initial points for K-means clustering | Benchmark problems downloaded from the web http://www.branchandcut.org/ accessed on 17 October 2021. Compared three different versions of the proposed algorithm | Total distance travelled |
### Table 1. Cont.

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|-----------|--------------|--------------------------|---------------|---------------------|
| Katarya & Verma [104]  | Fuzzy-C-Means + GWO | Non-Automatic Recommender System | Generating Initial Clusters and initial clusters centroids | Classification by similarity of user ratings | Movie lens dataset | PCA, PCA-SOM, K-means, PCA-K-means, K-means improved, SOM-Cluster, FCM, KM-PSO-FCM, PCA-GAKM and GARM-Cluster | Mean absolute error, standard deviation, precision and recall |
| Mohammed et al. [107] - KMGWO | K-means + GWO | Classical Engineering problem | Performance Enhancement of GWO using K-means | CEC2019 benchmark test functions | GWO, CSO, WOA-BAT, WOA | Sine-Cosine Algorithm (SCA) |
| Moorthy & Pabitha [109] - SCAK-means | SCA + K-means | resource discovering for cloud resources | Non-Automatic Cloud computing | Updating initial centroid position | Cloud resources K-means | Intra Cluster similarity, Inter-Cluster similarity, the similarity of cloud resources, and convergence rate |
| Garcia, Yepes & Marti [118] | CSO + K-means | solving combinatorial optimization problems | Non-Automatic Design of counterfort retaining walls | Production of new solution in continuous space | The emission and cost values obtained from [34,66] | K-means, HS | Wilcoxon signed-rank, the Shapiro–Wilk or Kolmogorov–Smirnov-Lilliefors normality test |
| Manju & Fred [120] | CSO + K-means | Optimization-based segmentation and compression | Compound images segmentation &compression | Generate initial centroids | NSL-KDD dataset IGNB chi square selection, and COFS | Image quality index, PSNR, RMSE, SSIM and SDME |
| Deepa, & Sumitra [121] - CSOAKM | CSO + K-means | optimal global solution | Non-Automatic Intrusion Detection System | Generate initial cluster centroid | RIDER breast dataset | | |
| Arjmand et al. [117] | Cuckoo Search Algorithm + Multiple Kernel-based Fuzzy C-Means | an automatic tumor segmentation algorithm | Non-Automatic Breast tumor segmentation | Generate initial Centroids for K-means algorithm | RIDER breast dataset | K-means and Fuzzy C-Means | Cluster accuracy, rand coefficient, jaccard coefficient and computational time. |
| Binu, Selvi & George [119] - MKF-Cuckoo | Cuckoo Search Algorithm + Multiple Kernel-based Fuzzy C-Means | Searching for the best cluster centroids | Non-Automatic | | Iris and wine datasets | | |
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|----------|--------------|--------------------------|----------------------------------|
| Girsang, Yunanto & Aslamiah [113]-FCSA | Cuckoo search algorithm + K-means | faster cluster analysis | Non-Automatic | Exploration | Convergence | Iris, Wine, Yeast, Abalone, Breast cancer, Glass, E.coli, Haberman, Sonar, and Parkinson | K-means | Mean and Standard Deviation |
| Tarkhaneh, Isazadeh & Khamenei [115]-HCSPSO | CS + PSO + K-means | More optimized cluster result | Non-Automatic | Clustering | PSO and K-means produces new nest for CS | Standard benchmark datasets | CS, K-means, PSO, Improved Cuckoo Search, ICS, ESA, BFGSA and EBA |
| Ye et al. [111]-ICS-Kmeans | Improved Cuckoo search + K-means | Better clustering, accuracy, and faster convergence rate | Non-Automatic | initial centroids for K-means algorithm | Basic K-means Clustering | UCI standard dataset (Iris, Wine, Seeds, and Haberman) | CS-Kmeans, K-Means, PSO-Kmeans | Sum of Square Error (SSE) |
| Lanying & Xiaolan [114] | Cuckoo Search + K-means | Optimization of cluster center in K-Means | Non-Automatic | Optimizing the clustering center | Basic K-Means Clustering | Movie Lens dataset | K-Means, PSO-Kmeans and GA-Kmeans | Clustering accuracy and convergence speed |
| Saida, Kamel & Omar [112] | Cuckoo Search + K-Means | Reduction of the number of CS iteration | Non-Automatic | Document clustering | Generate Initial Cluster Centroids | Reuters 21578 Text Categorization Dataset and the UCI Dataset | Clustering | F-Measure |
| Singh & Solanki [116] | K-means + Modified Cuckoo Search | Global optimum convergence | Non-Automatic | Initial centroids for K-means algorithm | K-means clustering | Iris, Wine, Seed, and Page Blocks | Clustering | Sum of Square Error (SSE) |
| Kwelilo [124]-DE-KM | DE + K-means | High-quality clustering solutions | Non-Automatic | Production of candidate solutions | initial centroids fine-tuning solution | UCI dataset, TSPLIB library, USC-SIPI repository | Global K-means, DE, two K-means variants algorithm, Genetic K-means algorithm | Sum of Square Errors (SSE) |
| Wang [129] | DE + K-means | Non-Automatic | Determine the initial cluster centers | Clustering using weighted K-means algorithm | Iris, Wine, Seed, and Page Blocks | Davies Bouldin Index (DBI) and Cosine Similarity (CS) measure |
| Silva et al. [130] | ACDE + K-means | Automatically determine k activation threshold | Automatic | DE approach | Automatic determination of cluster number | UCI standard dataset | Clustering | |
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|-----------|--------------|--------------------------|-------------|---------------------|
| Cai et al. [125]-CDE   | DE + one-step K-means | Improvement of DE | Automatic | DE approach | Clustering | multi-parent crossover operator | unconstrained single-objective benchmark functions with different characteristic | DE | Number of fitness function evaluations (NFFEs) and quality of the final solutions. |
| Mustafi & Sahoo [36]  | GA + DE + K-means | To improve the initial cluster centroids | Automatic | Test Document Clustering | DE approach | Generating improving cluster centers | Basic K-means clustering | Basic implementations of K-Means | |
| Bonab et al. [97]      | ABC + DE + Modified K-means | To solve initialization problems | Automatic | DE clustering | Clustering | Standard UCI dataset | |
| Siyera, Cobos, & Corrales [127] | DE + K-Means | A hybrid for continuous optimization | Non-Automatic | Generation of initial groups for DE | A large set of test functions | DE and PSO | |
| Shung et al. [131]-DE-ANS-AKO | DE + Adaptive niching + K-means | Dynamic adjustment of niche size to prevent premature convergence | Non-Automatic | DE clustering | Use of one iteration of k-means for fine-tuning the initial solution | Synthetic datasets, Letter, Connectist, Shuttle, MFC,Cs, Isolde1, Isolde2, HARs, Flowers17, Mnist, Cancer728, Yeast2945 | DE-AKO, DE-ANS-KO, GKA, MiQPSO, EPSONS, PSQKM, CGABC, SHADE, TSMPSO, ICMPKHM, FPGCA | Mean ICV Mean ARI Mean AC, Mean runtimes and Wilcoxon’s rank-sum tests. |
| Kuo, Suryani & Yasid [126]-CDE-K-Means | ACDE + K-means | An Automatic clustering algorithm | Automatic | Clustering | Tuning cluster centroids to improve performance | Iris and Wine | DE | |
| Hu et al. [128]        | DEFOA + K-means | Improving K-means | Non-Automatic | Clustering | Sales database | K-Means | F-Measure | |
| Fan et al. [134]-IWO-KMEANS | IWO + K-means | Improve global optimization while utilizing local optimization power | Non-Automatic | Text Clustering | Selection of initial cluster center | Basic K-means clustering | Chinese documents (history, transportation, medical, and sports) from the corpus of Fudan University | K-Means, DE-K-Means | F-Measure |
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|--------------------------------|----------|--------------|--------------------------|---------------|---------------------|
| Pan et al. [135] | IWO + K-means | Overcome the drawbacks of K-Means | Non-Automatic | Direct-K-means search for definite evolution direction | Clustering | | |
| Razi [137] | DEA based K-means + IWO | Clustering algorithm for better facility location | Non-Automatic | Facility Location problem | Determining the Pareto solution for the bi-objective model | Clustering | | |
| Boobord, Othman, & Abubakar [136] | PCA + IWO + K-means | PCA for dimensionality reduction | Non-Automatic | WK-means for clustering | Wine, Cancer, USCensus90, SPECTF Heart and Musk2000 | PCAK | Sum of Square Error (Best, Average, Worst and Standard deviation) |
| Emami & Derakhshan [65] | ICA + Fuzzy K-means | Escape from local optimal and increased convergence speed | Non-Automatic | Clustering in an alternate manner with the FKM | Iris, Glass, Sample, Contraceptive Method Choice (CMC), and Wine | ICA, PSOKHM, PSO, FKM and HABC | F-measure and runtime metrics |
| Abdeyazdan [140] | Modifier ICA + K-Harmonic means | Compensate existing problems in cluster analysis | Non-Automatic | Milling Machines classification | Iris, Glass, Contraceptive Method Choice, and Wine | ICAKM, KHM, GSOKHM and PSOKHM methods | F-measure, KHM (X, C), Runtime (s) |
| Niknam et al. [139] | K-means + Modified Imperial Competitive Algorithm | Optimum clustering | Non-Automatic | Generates initial empires for the modified ICA | Iris, Vowels, Wine and Contraceptive method choice | ACO, MICA, SA, PSO, GA, HBMO, TS and K-Means | The best, average, worst of the fitness function and Standard deviation of the fitness function. |
| Nazer, Sebastian & Kumar [145] | Harmony Search + K-Means | Better cluster accuracy | Non-Automatic | Determining the initial cluster centroids | Human Fibroblast Serum data and the Rat CNS data | K-Means, SOM, HCM, VCA, CRC | silhouette Index |
| Forsati et al. [141] | Harmony Search + K-Means | Less dependent on initial parameters | Non-Automatic | Initial centroids | Obtain the best vector from the HS | K-Means, HSCLUST | F measure |
| Chandran & Nazer [144] | Enhanced K-Means + Harmony Search | Better cluster solution | Non-Automatic | Determining the initial cluster centroids | UCI Machine Learning Repository dataset (Iris, New-Thyroid and Breast Cancer) | K-Means, HS-K-means | Cluster Purity metric. |
### Table 1. Cont.

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|----------|--------------|--------------------------|---------------|---------------------|
| 105 Raval, Raval & Valiveti [146] | Harmony Search + K-Means | Cluster Analysis Optimization | Non-Automatic | Sensor Network Energy Utilization | Finding initial cluster centers called Clustering Hierarchy (CH) | Dataset simulation using NS2 simulator | K-Means, HSA | Energy dissipation, Total data transfer in number of packets |
| 106 Cobos et al. [143]-IGHFSK | Global best Harmony Search + K-Means | Hybriding Global best Harmony Search with K-Means | Automatic | Web document clustering | Using BIC or Davies-Bouldin index | Provides global search strategy in the solution space | Finds the optimum value in a local search space | Datasets based on Reuters-21578 and DMOZ | Carrot2 |
| 107 Mahdavi & Abolhassani [142]-HKA | K-means + Harmony Search | An algorithm based on HS optimization | Non-Automatic | Web document clustering | Global search for optimum solutions | Localize search in the proximity of the obtained global solution | TREC-5, TREC-6, TREC-7, DMOZ, and 20 Newsgroup | K-Means, GA, PSO AND GM | Quality and speed of convergence, F-Measure |
| 108 Kim et al. [147] | Harmony Search + K-means | Clustering-based SDN load balancing scheme | Non-Automatic | SDN load balancing | Fine-tuning the solution from K-means clustering | Basic clustering | 100 to 1000 switches and 10 to 100 controllers are randomly placed in an area of 100 x 100 | HS, P-HS, and P-HS-K | Measure of accuracy |
| 109 Eskandarzadehalamdary, Masoumi, & Sojodishijani [151]-BH BK | Black Hole + Bisecting K-means | Improve performance of bisecting K-means | Non-Automatic | Generates initial cluster centroids for BK-means | Basic clustering and refinement | Iris, Glass, Vowel, and Contraceptive Method Choice (CMC) | Bisecting K-means, BH, PSO | Sum of intra-cluster distances and Error Rate (ER) |
| 110 Feng, Wang & Chen [153] | Black Hole + K-means | Initial cluster centers for K-means | Non-Automatic | Image Classification | Determining the initial cluster center for K-means | Basic clustering and refinement | | |
| 111 Pal & Pal [152] | Black Hole + K-means | Improved cluster analysis | Non-Automatic | Clustering | Partly generates initial cluster center | K-Means | | |

**Black Hole (BH) Algorithm**

| 112 Jiang, Zang & Liu [155] | K-means + DNA genetic Algorithm + P system | K-means based on DNA genetic algorithm and P system | Non-Automatic | Analyze the initial cluster center with P system | Randomly generated dataset | Convergence rate, Measure of accuracy and intra cluster distance |
| 113 Wang, Xiang & Liu [159] | K-means + K-medoids + Tissue-like P system | Handling noises and outliers | Non-Automatic | Tissue-like P system to present parallel operation | UCI dataset | K-means and K-medoids |
| 114 Zhao, Liu & Zhang [158] | P system + K-medoids | Using P system to realize K-medoids algorithm | Non-Automatic | Provide parallel operation for lower time complexity | Clustering | | |
Table 1. Cont.

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|-----------|--------------|--------------------------|-----------------------------------|
| Weisun & Liu [157]      | MDE K-means + P system | Improved initial cluster center for K-means | Non-Automatic | Evolve the objects with MDE | Clustering | Artificial data sets, the iris, wine | K-means algorithm and DE-K-means algorithm | Cluster validity index, Xie-Beni index, the PBMF index |
| Zhao & Liu [156]        | K-Means + GA + Tissue-like P system | Improved initial cluster center for K-means | Non-Automatic | P system for parallelism and GA for good convergence | Clustering | UCI datasets | K-means, CCIA, k-d tree, K-means++, PSO, Rais,' Khan's | No of initialisation cells |
| Wang, Liu & Xiang [160] | K-means + Tissue-like P system | Improved initial cluster center for K-means | Non-Automatic | Selection of initial cluster centers | Clustering | UCI datasets—Wine, Glass, Haberman, Soybean-small, and Zoo | No of initialisation cells |
| Angelin [162]           | K-means + Dragonfly | Outlier detection | Non-Automatic | Optimizing the generated clusters | Arthymia, Diabetes and Epileptic seizure | SAR Images | No of initialisation cells |
| Kumar, Reddy & Rao [163] | WHIDA-FCM | SAR Images change detection | Non-Automatic | SAR Image Change detection | SAR Images | SAR Images | No of initialisation cells |
| Chen et al. [163]       | Quantum-inspired ant lion optimizer + K-Means | An efficient algorithm for intrusion detection | Non-Automatic | Intrusion detection | Intrusion detection | KDD Cup datasets and Iris, Glass, Wine, Cancer, Vowel, CMC and Vehicle | No of initialisation cells |
| Murugan & Babaraj [164] | ALPSOC | Improved K-medoids + Ant lion + PSO | Non-Automatic | Optimized the generated initial clusters | K-Means datasets—Glass, Leaf, Seeds, Soybean and Ionosphere | No of initialisation cells |
| Dhand & Sheoran [165]   | Ant Lion Optimizer + K-Means algorithm | Energy-efficient routing protocol | Non-Automatic | Energy-efficient routing protocol | Clustering | No of initialisation cells | No of initialisation cells |

**Dragonfly Algorithm (DA)**

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|-----------|--------------|--------------------------|-----------------------------------|
| Whiddon & Liu [157]     | MDE K-means + P system | Improved initial cluster center for K-means | Non-Automatic | Evolve the objects with MDE | Clustering | Artificial data sets, the iris, wine | K-means algorithm and DE-K-means algorithm | Cluster validity index, Xie-Beni index, the PBMF index |
| Zhao & Liu [156]        | K-Means + GA + Tissue-like P system | Improved initial cluster center for K-means | Non-Automatic | P system for parallelism and GA for good convergence | Clustering | UCI datasets | K-means, CCIA, k-d tree, K-means++, PSO, Rais,' Khan's | No of initialisation cells |
| Wang, Liu & Xiang [160] | K-means + Tissue-like P system | Improved initial cluster center for K-means | Non-Automatic | Selection of initial cluster centers | Clustering | UCI datasets—Wine, Glass, Haberman, Soybean-small, and Zoo | No of initialisation cells |
| Angelin [162]           | K-means + Dragonfly | Outlier detection | Non-Automatic | Optimizing the generated clusters | Arthymia, Diabetes and Epileptic seizure | SAR Images | No of initialisation cells |
| Kumar, Reddy & Rao [163] | WHIDA-FCM | SAR Images change detection | Non-Automatic | SAR Image Change detection | SAR Images | SAR Images | SAR Images |
| Chen et al. [163]       | Quantum-inspired ant lion optimizer + K-Means | An efficient algorithm for intrusion detection | Non-Automatic | Intrusion detection | Intrusion detection | KDD Cup datasets and Iris, Glass, Wine, Cancer, Vowel, CMC and Vehicle | No of initialisation cells |
| Murugan & Babaraj [164] | ALPSOC | Improved K-medoids + Ant lion + PSO | Non-Automatic | Optimized the generated initial clusters | K-Means datasets—Glass, Leaf, Seeds, Soybean and Ionosphere | No of initialisation cells |
| Dhand & Sheoran [165]   | Ant Lion Optimizer + K-Means algorithm | Energy-efficient routing protocol | Non-Automatic | Energy-efficient routing protocol | Clustering | No of initialisation cells | No of initialisation cells |

**Ant Lion Optimizer (ALO)**

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|-----------|--------------|--------------------------|-----------------------------------|
| Whiddon & Liu [157]     | MDE K-means + P system | Improved initial cluster center for K-means | Non-Automatic | Evolve the objects with MDE | Clustering | Artificial data sets, the iris, wine | K-means algorithm and DE-K-means algorithm | Cluster validity index, Xie-Beni index, the PBMF index |
| Zhao & Liu [156]        | K-Means + GA + Tissue-like P system | Improved initial cluster center for K-means | Non-Automatic | P system for parallelism and GA for good convergence | Clustering | UCI datasets | K-means, CCIA, k-d tree, K-means++, PSO, Rais,' Khan's | No of initialisation cells |
| Wang, Liu & Xiang [160] | K-means + Tissue-like P system | Improved initial cluster center for K-means | Non-Automatic | Selection of initial cluster centers | Clustering | UCI datasets—Wine, Glass, Haberman, Soybean-small, and Zoo | No of initialisation cells |
| Angelin [162]           | K-means + Dragonfly | Outlier detection | Non-Automatic | Optimizing the generated clusters | Arthymia, Diabetes and Epileptic seizure | SAR Images | No of initialisation cells |
| Kumar, Reddy & Rao [163] | WHIDA-FCM | SAR Images change detection | Non-Automatic | SAR Image Change detection | SAR Images | SAR Images | SAR Images |
| Chen et al. [163]       | Quantum-inspired ant lion optimizer + K-Means | An efficient algorithm for intrusion detection | Non-Automatic | Intrusion detection | Intrusion detection | KDD Cup datasets and Iris, Glass, Wine, Cancer, Vowel, CMC and Vehicle | No of initialisation cells |
| Murugan & Babaraj [164] | ALPSOC | Improved K-medoids + Ant lion + PSO | Non-Automatic | Optimized the generated initial clusters | K-Means datasets—Glass, Leaf, Seeds, Soybean and Ionosphere | No of initialisation cells |
| Dhand & Sheoran [165]   | Ant Lion Optimizer + K-Means algorithm | Energy-efficient routing protocol | Non-Automatic | Energy-efficient routing protocol | Clustering | No of initialisation cells | No of initialisation cells |
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|------------------------|-----------|-------------|---------------------------------|----------|--------------|------------------------|--------------|---------------------|
| Majhi & Biswal [164]   | K-Means + Ant Lion Optimizer | Optimal cluster analysis | Non-Automatic | Optimized the generated clusters | Generate initial clusters | Glass, vowel, sponorhoe, leaf, gene expression cancer RNA-seq, waveform database generator (version 2), immunotherapy, and soybean | K-Means, KMeans-PSO, KMeans-FA, DBSCAN and Revised DBSCAN | Sum of intra-cluster distances and F-measure |
| Naem & Ghali [167]-K-median Modularity ALO | Social network community detection | Non-Automatic | Social Networks community detection | Optimized the generated clusters | Generate initial clusters | Zachary karate Club, Bottlenose Dolphins network, American College football network, Polbooks network | K-means Modularity PSO, K-means Modularity Bat optimization, K-means Modularity CSO, K-median Modularity PSO, K-median Modularity Bat optimization, K-median Modularity CSO, GN, F1, BGLL, HSCDA | Normalized Mutual Information (NMI), Measure of Modularity for community quality |
| Thiruvenkatasuresh & Venkatachalam [171] | Social Spider Algorithm + Fuzzy C-means | classify and segment Brain tumor images | Non-Automatic | Tumor detection in Brain images | Optimizing Centroid | UCI datasets (Iris, Glass, Vowel, Wine, Ruspini, and Cancer) | ANFIS and FCMGWO |
| Chandran, Reddy, & Janet [170]-SSOKC | Balance local and global searches with improved convergence speed | Non-Automatic | | To find the vicinity of optimal solution | initial centroid refinements and final optimal solution | Kbat, KFA, KPA, CPU Elapse Time |
| Sharma & Patel [173]-K-Means-FFO | Optimal clustering quality | Non-Automatic | Optimize initial Clusters | Generate initial clusters | 20NewsGroup, Reuters-21578, and Classic-4 dataset | K-means, K-means-PSO and K-means-ALO | Intra-cluster distance, Purity Index, F-Measure and Standard Deviation |
| Jiang et al. [174] | K-means + FOA | Optimal clustering quality | Non-Automatic | Optimize initial Clusters | Generate initial clusters | Integrated data of affected areas | RWFOA and MFOA | Weighted sum of construction costs, transportation costs and penalty costs of emergency rescue centers |
| Metheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|------------------------|-----------|-------------|----------------------------------|----------|--------------|-------------------------|---------------|----------------------|
| Wang et al. [176] -FOAKFCM | Kernel-based Fuzzy C-means + FOA | Integrating kernel-based fuzzy c-means and FOA | Non-Automatic | Initialize initial cluster centroids | Classifying/Clustering the data | Iris, Glass, and Seeds | FCM, KFCM | Classification evaluation index (XB index) |
| Hu et al. [128] | DEFOA + K-means | Improving K-means for universal continuous optimization | Non-Automatic | Generate initial cluster centroids | Optimize the initial clustering | Sales database | K-means | Convergence performance |
| Aboubi, Drias & Kamel [179] -BSO-CLARA | BSO + K-medoids | Effective and efficient algorithm | Non-Automatic | | | PAM, CLARA and CLARANS |
| Djenouri, Habbas & Aggoune-Mtalaa [180] | Using K-means as decomposition | Non-Automatic | | | Clustering | DIMACS |
| Djenouri, Bellahi & Belkheir [178] | BSO + K-means | Document Information Retrieval Problem | Non-Automatic | Exploration of already created clusters | Clustering | CACM collection, TREC, Webdocs and Wikilinks | PTM, SVMIR, KNNIR and ARMIR | F-measure, Runtime |
| Revathi, Eswaramurthy, & Padmaravathi [182] -BCO + KM | BCO + K-means | Reduced computational cost | Non-Automatic | Selection of initial cluster centroids | Optimizing the initial clusters for optimal solution | 2 Artificial datasets; UCI datasets (CMC, Glass WRC, Heart, Iris, Wine, Vowel, Balance) | K-means, PSO, BFO and BCO | Sum of Square Errors (SSE) |
| Vijayakumari & Deepa [183] -HFCA | FCM + Fuzzy BCO | High efficiency | Non-Automatic | Selection of initial cluster centroids | Optimizing the initial clusters for optimal solution | Iris, WBC, Glass, Wine, Vowel, and CMC | FBBO, FBCO, FCM and FPSO | IntraCluster distance |
| Kartik, Tamhachagan, & Narayana [185] -SS-KMeans | SDS + K-means | Finding optimal clustering points | Non-Automatic | Select initial centroid for clustering | Clustering | S | True Positive Rate (TPR) |
| Teimoury et al. [187] -HMBK | HBMO + KMeans | An optimized hybrid clustering algorithm | Non-Automatic | Selection of Initial Cluster centroids | Clustering | Wine, Iris and B.C. | SA, PSO, TS, ACO, GA, K-means | Sum of Square Error (SSE) |
| Table 1. Cont. |
|----------------|
| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|--------------------------|-----------|-------------|---------------------------------|----------|--------------|-------------------------|---------------|----------------------|
| Aghaebrahimi, Golkhandan & Ahmadnia [188] | HBMO + KMeans | Localization and sizing of flexible AC transmission system | Non-Automatic | Determining the best fitness function | Data Classification—Clustering | TCSC, UPFC AND SVC | Average Installation Cost, total generation cost and cost of power transmission losses |
| Senthilkumar & Chitra [190] | MCSO + K-means | Load balance in cloud networks | Non-Automatic | Measuring the load ratio | Clustering | | Overall Response time and Processing time |
| Onan & Korukoglu [193] | K-means + GSO | An efficient and effective hybrid algorithm | Non-Automatic | Find initial Cluster Centroids | Clustering | Iris, Breast Cancer, E.coli, Diabetes, Haberman’s survival data | K-means, Fuzzy C-Means, GSO | F-measure and Rand Index |
| Tang et al. [194] | Improved GSO + K-means | Multi-modal optimization for optimal cluster analysis | Non-Automatic | Generates the initial cluster center | Clustering | Iris dataset | K-means, K-means++, GSO + K-means, | Run time, minimum number of iterations, SSE, NMI, Purity and Rand Index |
| Zhou et al. [192] | GSO + K-means | Avoid the effect of the initial condition | Non-Automatic | Generates the initial cluster center | Clustering | Pepper, Lena, and Mandrill | K-means, Fuzzy C-Means | Quantization error, the maximum intra-distance, the minimum inter-distance |
| Das, Das & Dey [194] | MBCO + K-means | Faster convergence | Non-Automatic | Either generate initial centroids or does the clustering | Either generate initial centroids or does the clustering | Glass, Wine, Vowel, CMC, Cancer, HV, Iris | MBCO, K-NM-PSO, K-PSO, K-HS, KIBCOCLUST, IBCOCLUST | PSO | Percentage Error (PE) |
| Forsati, Kerikha & Shamstand [197] | Improved BCO + K-means | An efficient algorithm for large and high dimensional dataset | Non-Automatic | Generates initial cluster centroids | Clustering | Wine, Iris, Glass, Vowel, Cancer Document dataset (Politics, TREC, I2OZ, 20 Newsgroup and Web Ace) | GA, ACO, K-means, PSO, CABE, IBCOCLUST, HSCLUST, K-NM-PSO, K-PSO, K-GA, K-HS, K-ABC | Cluster Quality and Rate of Convergence |
| Niu, Duan & Liang [201] | BFO + K-means | Efficient algorithm with global and parallel search capacities | Non-Automatic | Generates initial cluster centroids | Clustering | | | |
Table 1. Cont.

| Metaheuristic Algorithm | Objective | Application | Method for Automatic Clustering | MOA Role | K-Means Role | Dataset Used for Testing | Compared with | Performance Measure |
|-------------------------|-----------|-------------|---------------------------------|-----------|--------------|-------------------------|--------------|---------------------|
| Cuckoo Optimization Algorithm (COA) | | | | | | | |
| 146 Lashkari & Moattar [202]-ECOA-K | ECOA + K-means | Fast convergence algorithm with intelligent operators | Non-Automatic | Generate initial cluster centroids | Clustering | UCI dataset (CMC, Iris, and Wine) | BH, Big Bang Big Crunch (BBBC), CSA, COA, K-means | Purity Index, Convergence rate, Coefficient of Variance, time complexity |
| Symbiotic Optimization Search (SOS) | | | | | | | |
| 147 Yang, & Sutrisno [198]-CSOS | SOS + K-means | An automatic hybrid clustering algorithm | Automatic | Assigning half the population size as the number of clusters | Clustering | 28 benchmark functions | CRPS, SaNSDE, rCMA-ES, GA, SOS and GWO | Number of successful runs, Average computational time, and Average number of evaluations |
3.3. RQ3. What Were the Various Automatic Clustering Approaches Adopted in the Reported Hybridization?

Different authors have varied approaches in achieving automatic clustering in integrating K-means with the corresponding MOA in the reviewed literature. Zhou et al. [45] adopted the Noise Method [203] and the K-means++ method [204]. Dai, Jiao, and He [28] achieved automatic clustering through dynamic optimization of cluster number $k$ through heredity, mutation with parallel evolution, and community intermarriage of the parallel genetic algorithm coupled with variable-length chromosome encoding. From the work of Li et al. [40], an optimal K-value was generated from the initial seed of chromosomes ranging between 1 and MaxClassVal, expressing the K-value by a byte classified into 255 kinds. Kuo et al. [39] employed the self-organizing feature map (SOM) neural network method [205,206] which involves the projection of high dimensional input space into a low-dimensional topology for the visual determination of the cluster number. An improved canopy [207] with K-means++ [204] techniques were used by Zhang and Zhou [35], where the canopy technique leverages domain-specific attributes to design a cheap distance metric for creating canopies using Euclidean distance. Mohammadreza Pourahmad [46] generated the initial number of clusters from a uniform distribution over a specified range of 2 to M, where M is the number of objectives in a multi-objective optimization algorithm [208]. Patel, Raghuwanshi, and Jaiswal [200] used the approach of determining the female chromosomes using the sex determination method (SDM) in the genetic algorithm and assigning the number of females as $k$.

In Barekatain, Dehghani & Pourzaferani [44], the dataset was segmented into nonequivalent cells, and the nodes whose residual energy is more than the average of its cell were selected as cluster heads. The number of cluster heads is then taken as $k$. The use of Mahalanobis distance to consider the covariance between data points for better representation of initial data and the number of generated groups using the MapReduce framework forms the number of clusters was adopted by Sinha & Jana [33]. In Kapil, Chawla & Ansari [3], data objects act as candidates for cluster centroids. The GA operators are executed to find the fittest instance that serves as the initial cluster centroids. The number of fittest instances obtained automatically determines the number of clusters. Rahman and Islam [32] used a fixed number of chromosomes (half selected deterministically and the other half randomly) for the initial population for the GA process from which the fittest instance is obtained as cluster centroids. The method of allocating a range of values for $k$ (between 2 and 10) and selecting the best value that produced the optimal solution was used by Islam et al. [34]. Mustafi and Sahoo [36] combined the GA framework with differential evolution for obtaining the number of clusters, while Xiao et al. [31] employed a GA-based method that adopts a Q-bit representation for the dataset pattern with a single run of the conventional K-means on each chromosome. Omran, Salman, and Engelbrecht [58] used PSO to find the best set of cluster centroids among the existing data object to produce the optimum number of clusters, and Kao and Lee [61] used discrete PSO in optimizing the number of clusters. In the case of Sood and Bansal [86], the Bat algorithm was employed in optimizing the initial representative objects for each cluster.

The idea of using a manual strategy to find $k$ activation threshold by DE to automatically determine the number of clusters was adopted by Silva et al. [130]. At the same time, Cai et al. [125] used the idea of random generation of $k$ values, where $k$ is an arbitrarily generated integer number [36,97]. Kuo, Suryani, and Yasid [126] also used the DE approach in obtaining the number of clusters. The use of Bayesian information criterion (BIC) [209] or the Davies–Bouldin Index (BDI) [210] in automatically finding the number of clusters was employed by Cobos et al. [143]. Yang and Sutrisno [198] used the idea of specifying the initial number of clusters as half of ecosize generated as sub-ecosystems, in which CSOS then optimizes to generate the correct cluster number in a dataset. Table 2 present the list of adopted automatic clustering approaches, which have been reported in the literature.
### Table 2. List of adopted automatic clustering approaches.

| S/N | Authors | Adopted Automatic Clustering Approach |
|-----|---------|---------------------------------------|
| 1   | Zhou et al. [45] | Noise method combined with K-means++ |
| 2   | Dai, Jiao, and He [28] | Dynamic optimization through heredity, mutation with parallel evolution, and community intermarriage |
| 3   | Li et al. [40] | Determined optimal number of k from the initial seed of chromosomes ranging between 1 and MaxClassVal |
| 4   | Kuo et al. [39] | Self-organizing feature map (SOM) neural network method |
| 5   | Zhang & Zhou [35] | An improved canopy with K-means++ |
| 6   | Mohammadrezapour, Kisi, and Pourrahmad [46] | Optimizing a uniform distribution over a specified range of values |
| 7   | Patel, Raghuwanshi, and Jaiswal [200] | Sex determination method |
| 8   | Barekatain, Dehghani, & Pourzaferani [44] | Segmented into nonequivalent cells and selection of nodes whose residual energy is more than the cell’s average |
| 9   | Sinha & Jana [33] | The use of Mahalanobis distance and MapReduce framework |
| 10  | Kapil, Chawla, & Ansari [3] | Executing GA operators on data objects as candidates for cluster centroids to find the fittest instance |
| 11  | Rahman and Islam [32] | Selecting a fixed number of chromosomes (half selected deterministically and the other half randomly) as the initial population for the GA process to obtain the fittest instances |
| 12  | Islam et al. [34] | Allocating a range of values for k (between 2 and 10) and selecting the best value that produced the optimal solution |
| 13  | Mustafi and Sahoo [36] | Combining GA framework with differential evolution |
| 14  | Xiao et al. [31] | Employing GA-based method that adopts Q-bit representation for dataset pattern with a single run of the conventional K-means on each chromosome |
| 15  | Omran, Salman, and Engelbrecht [58] | Using PSO to find the best set of cluster centroids among the existing data objects |
| 16  | Kao and Lee [61] | Using discrete PSO in optimizing the number of clusters |
| 17  | Seod and Bansal [86] | Using Bat algorithm to optimize the initial representative objects for each cluster |
| 18  | Silva et al. [130] | Using a manual strategy to find a activation threshold by DE |
| 19  | Cai et al. [125] | Random generation of k value as k = rndint [2] where NP is the population size and rndint is a random integer number |
| 20  | Kuo, Suryani, and Yasid [126] | DE approach in obtaining the number of clusters |
| 21  | Cobos et al. [143] | Optimizing Bayesian information criterion (BIC) or the Davies–Bouldin index (DBI) |
| 22  | Yang and Sutrisno [198] | Specifying the initial number of clusters as half of ecosize generated as sub-ecosystems which CSOS then optimizes |

### 3.4. RQ4: What Were the Contributions Made to Improve the Performance of the K-Means Clustering Algorithm in Handling Automatic Clustering Problems?

Zhou et al. [45], in their hybridization of K-means with the corresponding MOA, were able to achieve an automatic selection of high-quality initial seeds without specifying the number of clusters to be generated as well as avoidance of premature convergence. From the work of Dai, Jiao, and He [28], the blind estimate of the cluster number by the K-means algorithm was avoided ensuring precision and reducing the influence of the cluster number; the algorithm search time was also reduced. The use of SOM in determining the number of clusters and starting points made the resulting integrated clustering algorithm more robust [39]. Rahman and Islam [32] and Zhang and Zhou [35] reported high-quality cluster results in their proposed clustering algorithm but with higher time complexity. Further work by Islam et al. [34] reportedly yielded higher-quality clusters with equivalent computational resources.

Patel, Raghuwanshi, and Jaiswal [200] reportedly achieved well distributed and well-separated clusters, which evolved faster with fewer functions evaluation for obtaining the optimal. Kapil, Chawla, and Ansari [3] obtained correct clusters from their k-means/GA integrated clustering algorithm. Mustafi and Sahoo [36] observed a significant reduction in the possibility of convergence of the K-means algorithm to local optimal. Xiao et al. [31] in their Q-bit-based GA/K-means integrated clustering algorithm, was able to achieve effective clustering without knowing cluster numbers beforehand. Omran, Salman, and Engelbrecht [58] obtained the correct number of clusters with the corresponding clusters with minimum interference from a user using their proposed integrated K-means/PSO clustering algorithm. According to Kao and Lee [61], combining K-means with discrete
PSO enhanced the performance of K-means in finding an optimal solution to dynamic clustering problems. Sood and Bansal [86] achieved better and efficient cluster analysis while integrating K-medoids with the bat algorithm. According to Silva et al. [130] and Kuo, Suryani, and Yasid [126], the integration of K-means with DE yielded an excellent cluster result. Cai et al. [125] reported a balance between exploration and exploitation in the search algorithm and improving the quality of the final cluster result. A superior and higher performance of K-means clustering integrated with ABC and DE was reported by Bonab et al. [97]. Cobos et al. [143] reported promising experimental results in their automatic hybridized clustering algorithm that combined global best harmony search with K-means. In the same vein, Yang and Sutrisno [198] reported promising performance of their automatic K-means algorithm hybridized with SOS, which was found faster in high dimensional problems alleviating the dimensionality effect.

In summary, the performance of the K-means clustering algorithm in handling automatic clustering problems was substantially improved in terms of determination of the correct number of clusters, high-quality cluster results, performance enhancements and computational efficiency, and avoidance of convergence into local optimal.

3.5. RQ5. What Is the Rate of Publication of Hybridization of K-Means with Nature-Inspired Meta-Heuristic Algorithms for Automatic Clustering?

This section examines the rate of publications of articles on hybridization of K-means with nature-inspired meta-heuristic based on the selected article.

Publications Trend of K-Means Hybridization with MOA

Figure 2 presents the publication trend of K-means hybridization with MOA in the last 20 years. There is significant growth in research involving hybridization of K-means with MOA, with 2020 having the highest number of articles. The bifurcated distribution of this publication is presented in Table 3, showing at least each MOA having a publication on its hybridization with K-means with reference to its proposed year. K-means hybridization with CS having the highest number of publications (4) in the year 2019. The total of each publication per MOA as well as per year is shown on the last but one column and last row of Table 3, respectively, with GA having the highest number of articles (25) followed by PSO (16), FA (12), CS (11), DE (10), ABC (8), HS (7), and MC (6). ALO and BAT have the same number of articles (four each) followed by GWO, IWO, and FFO, each having four articles; ICA, BH, and GSO came next with three articles each; FPA, DA, Bacterial CO, HBMO, and BCO has two articles each while the rest has one article each. Due to the fact that each algorithm has a different year of proposal, the normalized rate of publication of each MOA is presented in Figure 3. The normalized rate of publication was calculated using the equation below, where \(N_j\) is the number of publications in a year, with \(j\) and \(i\) representing the current year and MOA proposal year, respectively

\[
\text{Normalize rate of publication} = \left( \frac{\sum_{i}^{j} N_i}{(j - i)} \right)
\]

The normalized rate of publication of K-means hybridization with MOA is displayed in the last column of Table 3. The rate of publication of hybridization of K-means with MOA for automatic clustering is shown in Figure 3.
Figure 2. Rate of Publications on K-means hybridization with MOA.

Figure 3. Publication rate of K-means hybridization with MOA for automatic clustering.
### Table 3. The year-wise bifurcated K-means hybridization with MOA Publication Report.

| MOA                          | 2002 | 2003 | 2006 | 2007 | 2008 | 2009 | 2010 | 2011 | 2012 | 2013 | 2014 | 2015 | 2016 | 2017 | 2018 | 2019 | 2020 | 2021 | Total | Norm. Ra |
|-----------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|--------|----------|
| ALO (2015)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1     | 1      | 2      | 1      | 5       | 0.83    |
| ABC (2005)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | 3    | -    | -    | -    | -    | -    | 3     | 1      | 8      | 0.50    |
| BAT (2010)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | 1    | -    | -    | -    | -    | -    | 1     | 1      | 2      | 0.45    |
| Bacterial CO (2012)         | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1     | -      | 1      | 2       | 0.22    |
| BFO (2000)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | -    | -    | -    | -      | -      | 1      | 0.05    |
| BCO (2012)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | 1    | -    | -    | -    | -    | -      | -      | 1      | 0.22    |
| BSO (2012)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | -    | -    | -    | -    | -      | -      | 1      | 0.33    |
| BH (2013)                   | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | 1    | -    | -    | -    | -      | -      | 3      | 0.38    |
| CS (2009)                   | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | 1    | -    | -    | 3    | -    | 4    | 2      | -      | 11     | 0.92    |
| Cockroach SO (2010)         | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | -    | -    | -    | -    | -    | -    | -    | -      | -      | 1      | 0.09    |
| DE (2013)                   | -    | -    | -    | -    | -    | -    | -    | 1    | 1    | 1    | 1    | 1    | 1    | -    | -    | -    | -    | -      | -      | 10     | 1.25    |
| DA (2015)                   | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | -    | -    | -    | -    | -      | -      | 1      | 0.33    |
| COA (2011)                  | -    | -    | -    | -    | 1    | 1    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -      | -      | 10     | 0.10    |
| FFA (2008)                  | -    | -    | -    | -    | -    | -    | -    | -    | 1    | 1    | 2    | 2    | 1    | -    | 1    | -    | -    | -      | -      | 4      | 0.19    |
| FFO (2000)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 2      | -      | 4       | 0.19    |
| FPO (2012)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | -    | -    | -    | -      | -      | 1      | 0.22    |
| GA (1988)                   | 1    | 3    | 2    | 1    | 1    | 3    | -    | -    | -    | -    | 1    | 1    | 2    | 1    | 3    | 2    | 2      | -      | 25     | 0.76    |
| GSO (2009)                  | -    | -    | -    | -    | -    | -    | -    | -    | 1    | 1    | -    | -    | -    | -    | 1    | -    | -    | -      | -      | 3      | 0.25    |
| GWO (2014)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | 1    | -    | -      | -      | 4      | 0.36    |
| HS (2003)                   | -    | -    | -    | -    | 1    | 1    | 1    | -    | -    | 1    | -    | 1    | -    | -    | -    | -    | -    | -      | -      | 7      | 0.35    |
| ICA 92007                   | -    | -    | -    | -    | -    | 1    | -    | -    | -    | -    | -    | 1    | -    | -    | -    | -    | -    | -      | -      | 3      | 0.21    |
| IWO (2010)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -      | -      | 2      | 0.20    |
| MC (1998)                   | -    | -    | -    | -    | -    | -    | -    | 1    | 3    | -    | -    | -    | -    | 1    | -    | 1    | -    | -      | -      | 4      | 0.20    |
| HBMO (2011)                 | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 2    | -    | -    | -    | -    | -    | -      | -      | 2      | 0.20    |
| HS (1995)                   | 1    | 1    | 1    | 1    | 2    | 1    | -    | -    | 1    | 1    | 2    | 1    | -    | 1    | -    | 1    | -      | -      | 16     | 0.62    |
| PSO (2012)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | -    | -    | -    | -    | -      | -      | 2      | 0.20    |
| SCA (2016)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -      | -      | 2      | 0.20    |
| SOD (2011)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -      | -      | 2      | 0.20    |
| SOS (2014)                  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -      | -      | 2      | 0.20    |
| Social Spider O (2015)       | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 1    | -    | 1    | -      | -      | 2      | 0.33    |
| **Total/Year**              | 1    | 1    | 4    | 3    | 3    | 4    | 5    | 4    | 2    | 8    | 12   | 15   | 10   | 7    | 15   | 19   | 22   | 11    | 147   |        |         |

**Note:** Norm. Ra refers to the normalized relevance score, calculated by dividing the total number of publications by the total number of years.
The highest number of articles published with respect to this was recorded in the year 2010, 2018, and 2019 with three articles each; 2006, 2009, 2013, and 2015 had two articles each while the remaining years had only one article each. The automatic/non-automatic K-means hybridization per MOA is illustrated in Figure 4. Moreover, Figure 4 reveals that most of the publications on K-means hybridization with MOA addressed general clustering with less attention paid to automatic clustering. Only 23 articles out of 147 selected articles reported on automatic clustering. This shows that only 16% of the total articles published in the last two decades on K-means hybridization with MOA addressed the problem of automatic clustering. Among the MOA hybridized with K-means, only 7 MOAs (GA, PSO, BA, ABC, DE, HS, and SOS) out of the 28 reviewed MOA, which amounts to 20.6% that directed their hybridization towards solving automatic clustering problems. In general, it can be observed that the rate of publication on K-means hybridization with particular MOA is relatively low. There is a need for more research in this aspect to explore more possibilities of improving the performance of the existing hybridized algorithm. This implies that hybridizing the K-means with these other MOAs for solving automatic clustering problems needs to be explored. Table 3 shows the year-wise bifurcated K-means hybridization with MOA publication report. Similarly, the details of the articles selected and used in the analysis of the study are presented in Table 4.

Figure 4. MOA-based automatic vs. non-automatic K-means hybridization (Total number of articles considered = 147).
| Authors                        | Publishers | Journal/Conference                        | Indexing | Citation | Impact Factor |
|-------------------------------|------------|------------------------------------------|----------|----------|---------------|
| Abdeyazdan [140]              | Springer   | Journal of Supercomputing                | √        | 15       | 2.474         |
| Aboubi, Drias & Kamel [179]   | Springer   | Conf                                     |          |          |               |
| Aghaebrahimi, Golshikan & Ahmadnia [188] | IEEE     | Conf                                     |          |          | 8             |
| Angelin, B. [162]             | IEEE       | Turkish Journal of Computer & Mathematics Education | √        | 0        | 0.33          |
| Arjmand et al. [317]          | IEEE       | Conf                                     | √        |          | 5             |
| Armano & Farmani [95]         | IRIS       | Int'l Journal of Computer Theory & Engineering | √        |          | 36            |
| Bandypadhyay & Maulik [24]    | Elsevier   | Information Sciences                     | √        | 465      | 6.795         |
| Barekatain, Dehghani & Pourzafarani [44] | Elsevier | Procedia Computer Science                | √        | 38       | 2.09          |
| Behera et al. [77]            | Inderscience | Int'l Journal of Fuzzy Computation & Modelling | √        |          | 2             |
| Binu, Selvi & George [119]    | Elsevier   | AASRI Procedia                          | √        |          | 17            |
| Bonab et al. [97]             | Springer   | Computational Intelligence in Information Systems | √        |          | 12            |
| Boobord, Othman, & Abubakar [136] | i-srs.org | Intl. Journal of Advance Soft Computer Appl | √        |          | 1             | 0.79          |
| Cai et al. [125]              | Elsevier   | Applied Soft Computing                   | √        |          | 120           | 6.725         |
| Cao & Xue [102]               | IEEE       | Int'l Conf on Network & Information Systems for Computers | √        |          | 4             |
| Chandran & Nazeer [144]       | IEEE       | Recent Adv. in Intelligent Computational Systems | √        |          | 13            |
| Chandran, Reddy, & Janet [170] | IEEE     | Second Int'l Conf on Intelligent Computing & Control Systems | √        |          | 1             |
| Chaudhary & Banati [90]       | Inderscience | Int'l Journal of Advanced Intelligence Paradigms | √        |          | 0             | 0.63          |
| Chen & Zhang [59]             | IEEE       | Int'l Conf on Wireless Comms, Networking & Mobile Computing | √        |          | 31            |
| Chen et al. [166]             | Elsevier   | Knowledge-Based Systems                  | √        |          | 13            |
| Chen, Miao & Bu [72]          | IEEE       | Int'l Conf on Power, Intelligent Computing & Systems | √        |          | 2             |
| Cheng et al. [25]             | IEEE       | Int'l Conf on Pattern Recognition (ICPR'06) | √        |          | 23            |
| Cobos et al. [142]            | IEEE       | IEEE congress on evolutionary computation | √        |          | 13            |
| Dai, Jiao & He [28]           | IEEE       | Int'l Conf on Intelligent Information HIDing & Multimedia Signal Proc | √        |          | 10            |
| Das, Das & Dey [196]          | Elsevier   | Applied Soft Computing                   | √        |          | 25            | 6.725         |
| Dasu, Reddy & Reddy [99]      | Springer   | Adv. in Cybernetics, Cognition, & Machine Learning for Comm Tech | √        |          | 1             |
| Deepa, & Sumitra [121]        | IOPScience | Int'l Conf on Mechanical, Electronics & Computer Engineering | √        |          | 0             |
| Dhand & Sheoran [168]         | Elsevier   | Materials Today: Conf Proceeding          | √        |          | 4             | 31.04         |
| Authors | Publishers | Journal/Conference | Indexing | Citation | Impact Factor |
|---------|------------|--------------------|----------|----------|---------------|
| Djenouri, Belhadi & Belkebir [178] | Elsevier | Expert Systems with Appl | √ | 48 | 6.954 |
| Djenouri, Habbas & Aggoune-Mtalaa [180] | SCITEPRESS | Int'l Conf on Agents & Artificial Intelligence | √ | 5 | |
| El-Shorbagy et al. [177] | Springer | Computational Statistics | √ | 14 | 1 |
| Emami & Derakhshan [65] | Springer | Arabian Journal of Science & Engineering | √ | 28 | 2.334 |
| Eshlaghy & Razi [42] | Inderscience | Int'l Journal of Business Systems | √ | 15 | 0.53 |
| Eskandarzadehalsamrdy, Masoumi, & Sojodishjani [151] | IEEE | Iranian Conf on Elect Engineering | | 11 | |
| Fan et al. [134] | IEEE | Int'l Conf on Autonomic & Trusted Computing | | 2 | |
| Feng, Wang & Chen [154] | Springer | International Conf on Scalable Computing & Comm | | 0 | |
| Forsati et al. [141] | Elsevier | Neurocomputing | √ | 60 | 5.719 |
| Forsati, Keikha & Shamsfard [197] | IEEE | IEEE/WIC/ACM Int'l Conf on Web Intelligence & Intelligent Agent Tech | | 69 | |
| Gan, & Lai [89] | IEEE | Int'l Conf on Automatic Control & Intelligent Systems (I2CACIS) | | 3 | |
| García, Yepes & Marti [118] | MDPI | Mathematics | √ | 21 | 2.258 |
| Ghezelbash, Maghsoudi & Carranza [38] | Elsevier | Computer & Geoscience | √ | 23 | 3.372 |
| Girsang, Yunanto & Aslamiah [113] | IEEE | Int'l Conf on Elect Engineering & Computer Science | | 6 | |
| Hassanzadeh & Meybodi [74] | IEEE | Int'l Conf on Elect Engineering & Computer Science | | 99 | |
| Himabindu et al. [83] | Elsevier | Materials Today: Conf Proceeding | √ | 0 | 31.04 |
| Hu et al. [128] | IEEE | Int'l Conf on Computer Science & Education (ICCSE) | | 11 | |
| Huang [100] | MDPI | Symmetry | √ | 1 | 2.713 |
| Islam et al. [34] | Elsevier | Expert Systems with Appl | √ | 2 | 6.954 |
| Jensi & Jiji [92] | arXiv Cornell University | Advanced Computational Intelligence | | 37 | |
| Jiang et al. [174] | IEEE | IEEE Int'l Conf on Artificial Intelligence & Computer Appl | | 0 | |
| Jiang, Zang & Liu [155] | Springer | Int'l Conf on Human Centered Computing | | 2 | |
| Jie & Yibo [71] | IEEE | Int'l Conf on Power & Renewable Energy | | 0 | |
| Jin, Lin & Zhang [98] | MDPI | Algorithms | √ | 0 | 2.27 |
| Jitpakdee, Aimmanee & Uyyanonvara [79] | World Academy of Science, Engineering & Tech | Int'l Journal of Computer & Information Engineering | | 7 | |
| Kao & Lee [61] | Elsevier | Expert Systems with Appl | √ | 53 | 6.954 |
| Authors | Publishers | Journal/Conference | Indexing | Citation | Impact Factor |
|---------|------------|--------------------|----------|----------|---------------|
| Kao, Zahara & Kao [60] | IEEE | Int’l Conf on Intelligent Computing & Intelligent Systems | SCI, WOS | 400 | |
| Kapil, Chawla & Ansari [3] | IEEE | Int’l Conf on Parallel, Distributed & Grid Computing | | 67 | |
| Karegowda et al. [41] | Springer | Int’l Conf on Adv. in Computing | | 16 | |
| Karthik, Tamizhazhagan, & Narayana [185] | Elsevier | Materials Today: Conf Proceeding | SCI | 0 | 31.04 |
| Katarya & Verma [104] | Springer | Neural Computer & Appl | SCI, WOS, Scopus | 58 | 5.606 |
| Kaur, Pal & Singh [81] | Springer | Int’l Journal of System Assurance Engineering & Mgt | SCI, WOS, Scopus | 15 | 1.72 |
| Kim et al. [147] | IEEE | IEEE Annual Consumer Commns & Networking Conf | SCI, WOS, Scopus | 3 | |
| Korayem, Khorsid & Kasem [105] | IOP Publishing | IOP Conf series: materials science & engineering | SCI, WOS, Scopus | 44 | 0.51 |
| Kumar, Reddy & Rao [164] | Elsevier | Journal of Computational Design & Engineering | SCI, WOS, Scopus | 4 | 6.6 |
| Kumari, Rao & Rao [93] | Inderscience | Int’l Journal of Advanced Intelligence Paradigms | SCI, WOS, Scopus | 0 | 0.63 |
| Kuo & Li [80] | Elsevier | Computers & Industrial Engineering | SCI, WOS, Scopus | 28 | 5.431 |
| Kuo et al. [39] | Elsevier | Expert Systems with Appl | SCI, WOS, Scopus | 120 | 6.954 |
| Kuo, Suryani & Yasid [126] | Springer | Institute of Industrial Engineers Asian Conf | SCI, WOS, Scopus | 16 | |
| Kwedlo [124] | Elsevier | Pattern Recognition Letters | SCI, WOS, Scopus | 128 | 3.756 |
| Langari et al. [82] | Elsevier | Expert Systems with Appl | SCI, WOS, Scopus | 17 | 6.954 |
| Lanying & Xiaolan [115] | ACM | Int’l Conf on Intelligent Information Proc | SCI, WOS, Scopus | 0 | |
| Lashkari & Moattar [202] | Iran Journals | Journal of AI & Data Mining | SCI, WOS, Scopus | 5 | 0.127 |
| Laszlo & Mukherjee, [26] | IEEE | Transactions on pattern analysis & machine intelligence | SCI, WOS, Scopus | 153 | |
| Laszlo & Mukherjee, [27] | Elsevier | Pattern Recognition Letters | SCI, WOS, Scopus | 192 | |
| Li et al. [40] | IEEE | Int’l Conf on Digital Content, Multimedia Tech & its Appl | SCI, WOS, Scopus | 153 | |
| Lu et al. [43] | Springer | Int’l Conf on Bio-Inspired Computing: Theories & Appl | SCI, WOS, Scopus | 11 | |
| Mahdavi & Abedbassani [142] | Springer | Data Mining & Knowledge Discovery | SCI, WOS, Scopus | 168 | 3.67 |
| Majhi & Biswal [164] | Elsevier | Karbala Int’l Journal of Modern Science | SCI, WOS, Scopus | 44 | 2.93 |
| Manju & Fred [120] | Springer | Multimedia Tools & Appl | SCI, WOS, Scopus | 1 | 2.757 |
| Mathew & Vijayakumar [75] | IEEE | Int’l Conf on High Performance Computing & Appl | SCI, WOS, Scopus | 15 | |
| Mohammadrezaeezad, Kisi & Pourahmadi [46] | Springer | Neural Computer & Appl | SCI, WOS, Scopus | 12 | 5.606 |
| Mohammed et al. [107] | Emerald Publishing | World Journal of Engineering | SCI, WOS, Scopus | 0 | 1.2 |
| Moorthy & Patnitha [109] | IEEE | Int’l Conf on High Performance Computing & Appl | SCI, WOS, Scopus | 3 | |
| Mursuun & Baburaj [166] | IEEE | Int’l Conf on Smart Tech in Computing, Elect & Electronics | SCI, WOS, Scopus | 0 | |
| Mustafi & Sahoo [36] | Springer | Soft Computing | SCI, WOS, Scopus | 15 | 3.643 |
| Authors                          | Publishers | Journal/Conference                          | Indexing | SCI | WOS | Scopus | Google Scholar | DBLP | Citation | Impact Factor |
|---------------------------------|------------|--------------------------------------------|----------|-----|-----|--------|----------------|------|----------|--------------|
| Naem & Ghali [167]              | BEEI       | Indonesia Journal                          | √        | √   | √   | √      | 2              |      |          |              |
| Nayak et al. [66]               | Springer   | Int’l Conf on Computer & Comm Tech         |          |     |     |        | 20             |      |          |              |
| Nayak et al. [77]               | Springer   | Computational Intelligence in Data Mining  |          |     |     |        | 1              |      |          |              |
| Nayak, Naik & Behera [79]       | Springer   | Adv. in Intelligent Systems & Computing    |          |     |     |        | 4              |      |          |              |
| Nazee, Sebastian & Kumar [145]  | PMC        | Bioinformatics                             | √        | √   | √   | √      | 8              | 244  | 6.212    |              |
| Niknam et al. [139]             | Elsevier   | Engineering Appl                           | √        | √   | √   | √      | 476            | 6.725 |          |              |
| Niknam, & Amiri [53]            | Elsevier   | Applied Soft Computing                     | √        | √   | √   |        | 22             | 6.212 |          |              |
| Niu et al. [48]                 | Elsevier   | Engineering Appl                           | √        | √   | √   | √      | 325            | 2.58 |          |              |
| Niu, Duan & Liang [201]         | Springer   | Int’l Conf on Intelligent Data Engineering & AutoLearning |          |     |     |        | 6              |      |          |              |
| Omran, Salman & Engelbrecht [58]| Springer   | Pattern Analysis & Appl                    |          |     |     |        | 325            | 2.58 |          |              |
| Onan & Korukoglu [193]          | ProQuest   | Int’l Symposium on Computing in Science & Engineering |          |     |     |        | 1              |      |          |              |
| Pal & Pal [152]                 | Springer   | Computational Intelligence in Data Mining  |          |     |     |        | 3              |      |          |              |
| Pambudi, Badharudin & Wicaksono [106] | ICTACT     | Journal on Soft computing                  |          |     |     |        | 0              | 0.787 |          |              |
| Pan et al. [135]                | World Scientific | Int’l Journal of Pattern Recognition & Artificial Intelligence | √        | √   | √   |        | 18             | 1.375 |          |              |
| Patel, Raghuvanshi & Jaiswal [200] | IEEE       | IEEE Int’l Advance Computing Conf          |          |     |     |        | 13             |      |          |              |
| Paul, De & Dey [70]             | IEEE       | Int’l Conf on Electronics, Computing & Comm Tech |          |     |     |        | 11             |      |          |              |
| Pavez, Altimiras, & Villavicencio [88] | Springer | Proc of the Computational Methods in Systems & Software |          |     |     |        | 26             |      |          |              |
| Prabha & Visalakshi [64]        | IEEE       | Int’l Conf on Intelligent Computing Appl   |          |     |     |        | 52             | 8.038 |          |              |
| Rahman & Islam [32]             | Elsevier   | Knowledge-Based Systems                    | √        | √   | √   | √      | 5              |      |          |              |
| Ratanavisagul [69]              | IEEE       | Int’l Conf on Computational Intelligence & Appl |          |     |     |        | 1              |      |          |              |
| Raval, Raval & Valiveti [146]   | IEEE       | Int’l Conf on Recent Trends in Information Tech |          |     |     |        | 0              |      |          |              |
| Razi [137]                      | Springer   | Journal of Industrial Engineering Int’l     |          |     |     |        | 5              | 2.02 |          |              |
| Revathi, Eswaramurthy, & Padmavathi [182] | IOP Publishing | In IOP Conf Series: Materials Science & Engineering |          |     |     |        | 0              | 0.51 |          |              |
| Saida, Kamel & Omar [112]       | Springer   | Recent Adv. on Soft Computing & Data Mining |          |     |     |        | 11             |      |          |              |
| Senthilkumar & Chitra [190]     | IEEE       | Int’l Conf on Smart Systems & Inventive Tech |          |     |     |        | 0              |      |          |              |
| Sharma & Patel [173]            | ACADEMIA   | Int’l Journal of Computer Science & Information Security (IJCSIS) | √        | √   | √   |        | 0              | 0.702 |          |              |
| Sheng et al. [131]              | IEEE       | Transactions on Cybernetics               |          |     |     |        | 3              | 11.45 |          |              |
| Sheng, Tucker & Liu [30]        | Springer   | Soft Computing                             | √        | √   | √   |        | 2              | 3.643 |          |              |
| Authors | Publishers | Journal/Conference | Indexing | Citation | Impact Factor |
|---------|------------|--------------------|----------|----------|---------------|
| Sierra, Cobos, & Corrales [127] | Springer | Ibero-American Conf on Artificial Intelligence | SCI, WOS, Scopus | 6 |
| Silva et al. [130] | Springer | Int’l Conf on Green, Pervasive, & Cloud Computing | 4 |
| Singh & Solanki [116] | Springer | Emerging Research in Electronics, Computer Science & Tech | 8 |
| Sinha & Jana [33] | Springer | Journal of Supercomputing | 21 | 2.474 |
| Sood & Bansal [86] | Citeseer | Int’l Journal of Applied Information Systems | 21 |
| Tang et al. [194] | IEEE | Int’l Symposium on Distributed Computing & Appl | 1 |
| Tarkhanêb, Isazadeh & Khamnei [115] | Inderscience | Int’l Journal of Computer Appl | √, √, √ | 11 | 1.55 |
| Teimoury et al. [187] | MDPI | Sensors | √, √, √ | 4 | 3.576 |
| Thiruvenkatasuresh & Venkatachalam [171] | Inderscience | Int’l Journal of Biomedical Engineering & Tech | √, √ | 1 | 1.01 |
| Tran et al. [96] | IEEE | Chinese Journal of Electronics | √, √, √ | 38 | 0.941 |
| Tripathi, Sharma & Bala [87] | Springer | Int’l Journal of System Assurance Engineering & Mgt | √, √, √ | 32 | 1.72 |
| Tsai & Kao [63] | IEEE | Int’l Conf on Systems, Man, & Cybernetics | 91 |
| Van der Merwe & Engelbrecht [49] | IEEE | Congress on Evolutionary Computation | 953 |
| Vijayakumari & Deepa [185] | Infokara | | 0 |
| Wang et al. [101] | IEEE | Access | 2 | 3.367 |
| Wang et al. [176] | IEEE | Int’l Conf on Grey Systems & Intelligent Services | 5 |
| Wang [129] | IEEE | Adv. Infor Mgt, Comm, Electronic & Auto Control Conf | 3 |
| Wang, Liu & Xiang [160] | Taylor & Francis | Int’l Journal of Parallel, Emergent & Distributed Systems | √, √, √ | 2 | 1.51 |
| Wang, Xiang & Liu [159] | IEEE | Int’l Conf on Intelligent Science & Big Data Engineering | 0 |
| Weisun & Liu [157] | WSEAS | | 0 |
| Wu et al. [84] | Elsevier | Agricultural Water Mgt | √, √ | 6 | 4.516 |
| Xiao, Yan, Zhang, & Tang [31] | Elsevier | Expert Systems with Appl | √, √, √ | 108 | 6.954 |
| Xie et al. [74] | Elsevier | Applied Soft Computing | √, √ | 40 | 6.725 |
| Yang, Sun & Zhang [62] | Elsevier | Expert Systems with Appl | √, √, √ | 228 | 6.954 |
| Yang, & Sutrisno [198] | Elsevier | Applied Soft Computing | √, √ | 1 | 6.725 |
| Ye et al. [111] | IEEE | Int’l Conf on Convergence & Hybrid Information Tech | 10 |
| Zhang & Zhou [35] | IEEE | Int’l Conf on Artificial Intelligence & Big Data | 11 |
| Zhang, Leung & Ye [199] | IEEE | Int’l Conf on Convergence & Hybrid Information Tech | 26 |
| Zhao & Liu [58] | Springer | Int’l Conf on Human Centered Computing | 0 |
| Zhao, Liu & Zhang [158] | TELKOMNIKA | Indonesian Journal of Elect Engineering, | 5 |
| Zhou et al. [192] | Guangxi Key Laboratory | | 19 |
| Zhou et al. [45] | MDPI | ISPRS Int’l Journal of Geo-Information | √, √, √, √ | 20 | 2.899 |
4. Results and Discussions

4.1. Metrics

The articles that were selected for this study were based on metrics such as article publishers, journals, citation numbers, and the impact factors. Articles from conferences proceedings were also considered. The details of the articles selected are presented in Table 4. The largest number of articles were selected from IEEE with 46 articles, followed by Springer and Elsevier with 37 articles and 30 articles, respectively. Inderscience, MDPI, and IOP publishing, respectively, had six, five, and three articles each. PMC, ProQuest, and ScitePress have two articles each, while all other publishers have one each. Thirty-two of the articles were indexed in Science, twenty-four in WOS, sixty-one in Scopus, sixty-six in Google Scholar, and twenty-two in DBLP. All the articles were gathered between 19 May 2021 and 23 June 2021.

4.2. Strength of This Study

A comprehensive analysis of hybridization of K-means with nature-inspired meta-heuristic optimization algorithms is presented in this study. It includes a hundred and forty-seven hybridized K-means with different MOA clustering algorithms. Recent publications from 2019, 2020, and 2021 are also considered. The role of K-means algorithms and the corresponding MOA in the hybridized algorithms were highlighted, including the dataset used for testing and the criteria for their performance measure. This detail is presented in Table 4. The algorithms that actually handled automatic clustering are also identified among the lot. The various automatic clustering approaches adopted in the reported automatic K-means hybridization are also identified and presented. Current challenges, as well as future directions, are also discussed.

4.3. Weakness of This Study

In order to incorporate details of the relevant manuscripts, a maximum effort has been expended, and most available articles in the last two decades were considered. Nevertheless, it is an impossible task to cover all the manuscripts in a single study. All non-English-based related manuscripts were not included in this study. Some other metaheuristic optimization algorithms were not considered as well.

4.4. Hybridization of K-Means with MOA

From this study, it can be observed that the K-means clustering algorithm has been widely hybridized with various MOA to improve the process of data clustering. The advantages of K-means in terms of simplicity and low computational complexity have been harnessed to improve the clustering capability of many of the MOA. The ability of many of the MOA in global optimum search enhanced the performance of K-means in escaping local optimal convergence leveraging on their optimization capability. Hybridizing K-means with MOA provides a balance between exploration and exploitation in the search algorithm to improve the quality of the final cluster result. There are noticeable improvements in general clustering performance and efficiency in relation to cluster results.

Specification of the number of clusters as a user parameter is a major challenge in cluster analysis. The various hybridization of nature-inspired meta-heuristics techniques with K-means clustering algorithms that handled automatic clustering problems were presented. From the study, it can be seen that only a few of the hybrid algorithms addressed the problem of automatic clustering. Different methods were adopted in estimating the optimal number of clusters in any given dataset. In most of the automatic hybrid algorithms, the correct number of clusters were optimized from the initial population, which were either randomly generated or deterministically selected from the data objects.

Automatic specification of cluster number in the K-means with MOA hybrid algorithm conspicuously enhanced the performance of the former algorithm by reducing the number of iteration operations required to obtain an optimal result compared with the traditional
algorithm. Most initialization problems associated with traditional K-means, such as user-specified parameters of k and random selection of cluster centers, were resolved through the generation of optimized initial cluster centroids, which was made possible by the optimization process of the MOA. The number of optimum cluster centers invariably gives the number of clusters to be generated.

In some of the hybrid algorithms, parallelization of the K-means algorithm and quantum processing was made possible for faster convergence, handling distributed datasets, improved multidimensional datasets clustering, and reducing computational complexity. The issues of outlier detection, noise handling, discovering non-globular clusters, and non-linear partitioning were solved by some of the hybrid algorithms, as well as efficient clustering of large and high dimensional datasets.

Furthermore, the various hybridized algorithms were tested on either synthetically generated datasets, UCI datasets, or some real-life datasets. The datasets used with the corresponding hybrid algorithm can be found in Table 2. The performance of the hybridized algorithms was also measured using different cluster analysis performance metrics. This is also included in Table 2.

4.5. Impact of Automatic Hybridized K-Means with MOA

Hybridization of K-means with MOA for automatic clustering has been found to improve the performance of these algorithms in handling cluster analysis. Automatic determination of cluster numbers assists in avoiding the sensitivity of initial seeds in the initial population [45]. In most cases, it helps select near optimum initial cluster centroids for the clustering process instead of the usual random selection of the initial cluster centroids.

Determining the number of clusters automatically also enhances the convergence speed of the resultant hybridized clustering algorithm due to fewer iterations required to obtain the optimal cluster result. The impact of automatic hybridized algorithms is more pronounced when handling real-life datasets. An accurate guess of the correct number of clusters in real-life datasets is an assiduous task, if not impossible, due to its high dimensionality and density. Improving traditional K-means to solve real-life automatic clustering problems through hybridization is of great impact in cluster analysis.

4.6. Trending Areas of Application of Hybridized K-Means with MOA

The trending areas of application of K-means with MOA hybrid algorithms reported in the reviewed literature include cluster analysis optimization, image segmentation, social network community detection, localization and sizing of flexible AC transmission system, routing protocols, color quantization, forecasting models, image compression, satellite image classification, facility location, intrusion detection, document information retrieval, and cloud networks load balancing. A summarized list of all the application areas identified in the cause of the study that are associated with the hybrid K-means algorithms is listed in Table 1.

4.7. Research Implication and Future Directions

The major emphasis of this study is to identify the K-means hybrid developed for the purpose of automatic clustering. However, most of the reviewed articles concentrated efforts on finding solutions to the initial cluster centroid problems of the traditional K-means algorithm and the problem of local optimum convergence. In some other cases, the attention was not on improving the K-means clustering algorithm. Instead, the attention was on improving the performance of the corresponding MOA in handling the clustering problem. For the few that proposed improving the K-means clustering algorithm, their performances’ limitations, such as increased d number of user-dependent variables and algorithm complexity, limit their performances. The same drawbacks also affect the hybridized algorithm extending the K-means algorithm for handling automatic clustering.
Moreover, the number of research papers on the hybridization of K-means with MOA is relatively small compared with the number of existing MOAs and still smaller when the issue of automatic clustering is considered. There is a need for further research on finding new K-means hybridization that will enhance its performance in handling automatic clustering for big data clustering while maintaining its desirable quality of linear order complexity. In most hybridized algorithms, a higher execution time is required to obtain higher quality clustering results. Further, they are more computationally expensive due to the increase in the necessary iteration operation to achieve convergence. A computationally less expensive hybridized K-means algorithm that can handle automatic clustering will be highly desirable.

5. Conclusions

In this study, hybridization of the K-means clustering algorithm with different MOAs has been presented. The primary objective of each hybridization was considered with the role of corresponding MOA and K-means in the resultant hybridized algorithm. The various dataset used for testing as well as the criteria used for performance evaluation were similarly extracted. The various existing MOA and hybrids used for comparison purposes for judging the performance of the hybridized algorithm were also presented. The publication rate of research on K-means hybridization with some MOA has also been presented as well as the normalized rate of the publications. The critical analysis of the findings from the study revealed the normalized publication rate of the different extracted articles on integrating K-means with MOAs. Five research questions were designed, and the corresponding answers were provided in this extensive literature analysis of the different hybridization methods incorporating the K-means clustering algorithm with MOA.

From the response to the first research question, twenty-nine metaheuristics optimization algorithm, most of which are nature-inspired, were considered with a hundred and forty-seven articles reviewed that reports the various hybridization with K-means clustering algorithm or any of its variants. In the provided answers to the second research question, the various articles whose primary objective was to solve the problem of automatic clustering were identified among the reviewed articles. These articles were relatively small compared with the total number of articles selected for the study. Various areas of application where these hybridized algorithms have been deployed are also listed. The reviewed hybridized algorithm’s various approaches to automatic clustering were discussed in response to the third research question. The response to the fifth question presented a thorough analysis of the publication trend with reference to K-means hybridization with MOA in the last two decades. A bifurcation presentation of the reviewed algorithms reveals that there is a generally low rate in research publication involving the hybridization of K-means with MOA in most of the reviewed literature. This indicates a great need for more attention in this area of research, most especially for handling automatic clustering problems. This was further verified by the graphical report obtained from the normalization of the publication rate. Finally, the study further reveals that the existing hybridized K-means algorithms with MOAs still require higher execution time when applied to the clustering of a big dataset to obtain higher quality clustering results.

Author Contributions: Conceptualization, A.E.E.; methodology, A.M.I. and A.E.E.; software, A.E.E.; investigation, A.M.I.; resources, A.E.E.; data curation, A.M.I.; writing—original draft preparation, A.M.I. and A.E.E.; writing—review and editing, A.M.I., A.E.E. and M.S.A.; supervision, A.E.E.; project administration, A.E.E. and M.S.A.; funding acquisition, M.S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

DataAvailabilityStatement: Not applicable.
Conflicts of Interest: The authors declare no conflict of interest regarding the publication of this manuscript.

Abbreviations

| Abbreviation | Description |
|--------------|-------------|
| ABC          | Artificial Bee Colony |
| ABC-KM       | Artificial Bee Colony K-Means |
| ABDWTF-FCM   | Artificial Bee Colony based discrete wavelet transform with fuzzy c-mean |
| AC           | Accuracy of Clustering |
| ACA-AL       | Agglomerative clustering algorithm with average link |
| ACA-CL       | Agglomerative clustering algorithm with complete link |
| ACA-SL       | Agglomerative clustering algorithm with single link |
| ACDE-K-means | Automatic Clustering-based differential Evolution algorithm with K-Means |
| ACN          | Average Correct Number |
| ACO          | Ant Colony Optimization |
| ACO-SA       | Ant Colony Optimization with Simulated Annealing |
| AGCUK        | Automatic Genetic Clustering for Unknown K |
| AGWDWTF-FCM  | Adaptive Grey Wolf-based Discrete Wavelet Transform with Fuzzy C-mean |
| ALO          | Ant Lion Optimizer |
| ALO-K        | Ant Lion Optimizer with K-Means |
| ALPSOC       | Ant Lion Particle Swarm Optimization |
| ANFIS        | Adaptive Network based Fuzzy Inference System |
| ANOVA        | Analysis of Variance |
| AR           | Accuracy Rate |
| ARI          | Adjusted Rand Index |
| ARMIR        | Association Rule Mining for Information Retrieval |
| BBBBC        | Big Bang Big Crunch |
| BCO          | Bacterial Colony Optimization |
| BCO+KM       | Bacterial Colony Optimization with K-Means |
| BFCA         | Bacterial Foraging Clustering Algorithm |
| BFGSA        | Bird Flock Gravitational Search Algorithm |
| BFO          | Bacterial foraging Optimization |
| BGLL         | A modularity-based algorithm by Blondel, Guillaume, Lambiotte, and Lefebvre |
| BH           | Black Hole |
| BH-BK        | Black Hole and Bisecting K-means |
| BKBA         | K-Means Binary Bat Algorithm |
| BPN          | Back Propagation Network |
| BPZ          | Bavarian Postal Zones Data |
| BSO          | Bees Swarm Optimization |
| BSO-CLARA    | Bees Swarm Optimization Clustering Large Dataset |
| BSOGD1       | Bees Swarm Optimization Guided by Decomposition |
| BTD          | British Town Data |
| C4.5         | Tree-induction algorithm for Classification problems |
| CAABC        | Chaotic Adaptive Artificial Bee Colony Algorithm |
| CAABC-K      | Chaotic Adaptive Artificial Bee Colony Algorithm with K-Means |
| CABC         | Chaotic Artificial Bee Colony |
| CCI          | Correctly Classified Instance |
| CCIA         | Cluster Centre Initialization Algorithm |
| CDE          | Clustering Based Differential Evolution |
| CFA          | Chaos-based Firefly Algorithm |
| CGABC        | Chaotic Gradient Artificial Bee Colony |
| CIEFA        | Compound Inward Intensified Exploration Firefly Algorithm |
| CLARA        | Clustering Large Applications |
| CLARANS      | Clustering Algorithm based on Randomized Search |
| CMC          | Contraceptive Method Choice |
| CMIWO K-Means| Cloud model-based Invasive weed Optimization |
| CMIWOKM      | Combining Invasive weed optimization and K-means |
| COA          | Cuckoo Optimization Algorithm |
| Abbreviation | Description |
|--------------|-------------|
| COFS         | Cuckoo Optimization for Feature Selection |
| CPU          | Central Processing Unit |
| CRC          | Chinese Restaurant Clustering |
| CRPSO        | Craziness based Particle Swarm Optimization |
| CS           | Cuckoo Search |
| CSA          | Cuckoo Search Algorithm |
| CS-K-means   | Cuckoo Search K-Means |
| CSO          | Cockroach Swarm Optimization |
| CSOAKM       | Cockroach Swarm Optimization and K-Means |
| CSOS         | Clustering based Symbiotic Organism Search |
| DA           | Dragonfly Algorithm |
| DADWT-FCM    | Dragonfly Algorithm based discrete wavelet transform with fuzzy c-mean |
| DBI          | Davies-Bouldin Index |
| DBSCAN       | Density-Based Spatial Clustering of Applications with Noise |
| DCPSO        | Dynamic Clustering Particle Swarm Optimization |
| DDI          | Dunn-Dunn Index |
| DE           | Differential Evolution |
| DEA-based K-means | Data Envelopment Analysis based K-Means |
| DE-AKO       | Differential Evolution with K-Means Operation |
| DE-ANS-AKO   | Differential Evolution with adaptive niching and K-Means Operation |
| DEFOA-K-means | Differential Evolution Fruit Fly Optimization Algorithm with K-means |
| DE-KM        | Differential Evolution and K-Means |
| DE-SVR       | Differential Evolution -Support Vector Regression |
| DFBPKBA      | Dynamic frequency-based parallel K-Bat Algorithm |
| DFSABCelite  | ABC with depth-first search framework and elite-guided search equation |
| DMOZ         | A dataset |
| DNA          | Deoxyribonucleic Acid |
| DR           | Detection Rate |
| DWT-FCM      | Discrete wavelets transform with fuzzy c-mean |
| EABC         | Enhanced Artificial Bee Colony |
| EABCK        | Enhanced Artificial Bee Colony K-Means |
| EBA          | Enhanced Bat Algorithm |
| ECOA         | Extended Cuckoo Optimization Algorithm |
| ECOA-K       | Extended Cuckoo Optimization Algorithm K-means |
| EFC          | Entropy-based Fuzzy Clustering |
| EPSONS       | PSO based on new neighborhood search strategy with diversity mechanism and Cauchy mutation operator |
| ER           | Error Rate |
| ESA          | Elephant Search Algorithm |
| EShBAT       | Enhanced Shuffled Bat Algorithm |
| FA           | Firefly Algorithm |
| FACTS        | Flexible AC Transmission Systems |
| FA-K         | Firefly-based K-Means Algorithm |
| FA-K-Means   | Firefly K-Means |
| FAPSO-ACO-K  | Fuzzy adaptive Particle Swarm Optimization with Ant Colony Optimization and K-Means |
| FA-SVR       | Firefly Algorithm based Support Vector Regression |
| FBCO         | Fuzzy Bacterial Colony Optimization |
| FBO          | Fractional Bacterial Foraging Optimization |
| FCM          | Fuzzy C-Means |
| FCM-FA       | Fuzzy C-Means Firefly Algorithm |
| FCMGWO       | Fuzzy c-means Grey Wolf Optimization |
| FCSA         | Fuzzy Cuckoo Search Algorithm |
| FFA-KELM     | Firefly Algorithm based Kernel Extreme Learning Machine |
| FFO          | Fruit Fly Optimization |
| FGKA         | Fast Genetic K-means Algorithm |
| Abbreviation | Full Form |
|--------------|-----------|
| FI           | F-Measure |
| FKM          | Fuzzy K-Means |
| FM           | F-Measure |
| FN           | A modularity-based algorithm by Newman |
| FOAKFCM      | Kernel-based Fuzzy C-Mean clustering based on Fruitfly Algorithm |
| FPA          | Flower Pollination Algorithm |
| FPAGA        | Flower Pollination Algorithm and Genetic Algorithm |
| FPAKM        | Flower Pollination Algorithm K-Means |
| FPR          | False Positive Rate |
| FPSO         | Fuzzy Particle Swarm Optimization |
| FSDP         | Fast Search for Density Peaks |
| GA           | Genetic Algorithm |
| GABEEC       | Genetic Algorithm Based Energy-efficient Clusters |
| GADWT        | Genetic Algorithm Discrete Wavelength Transform |
| GAEESP       | Genetic Algorithm Based Energy Efficient adaptive clustering hierarchy Protocol |
| GAGR         | Genetic Algorithm with Gene Rearrangement |
| GAK          | Genetic K-Means Algorithm |
| GAS3         | Genetic Algorithm with Species and Sexual Selection species and sexual selection using K-Means |
| GAS3KM       | species and sexual selection using K-Means |
| GA-SVR       | Genetic Algorithm based Support Vector Regression |
| GCUK         | Genetic Clustering for unknown K |
| GENCLUST     | Genetic Clustering |
| GENCLUST-F   | Genetic Clustering variant |
| GENCLUST-H   | Genetic Clustering variant |
| GGA          | Genetically Guided Algorithm |
| GKA          | Genetic K-Means Algorithm |
| GKM          | Genetic K-Means Membranes |
| GKMC         | Genetic K-Means Clustering |
| GM           | Gaussian Mixture |
| GN           | A modularity-based algorithm by Girvan and Newman |
| GP           | Genetic Programming |
| GPS          | Global Position System |
| GSI          | Geological Survey of Iran |
| GSO          | Glowworm Swarm Optimization |
| GSO-KHM      | Glowworm Swarm Optimization |
| GTD          | Global Terrorist Dataset |
| GWDWT-FCM    | Grey Wolf-based Discrete Wavelength Transform with Fuzzy C-Means |
| GWO          | Grey wolf optimizer |
| GWO-K-Means  | Grey wolf optimizer K-means |
| HABC         | Hybrid Artificial Bee Colony |
| HBMO         | Honeybees Mating Optimization |
| HCSPSO       | Hybrid Cuckoo Search with Particle Swarm Optimization and K-Means |
| HESB         | Hybrid Enhanced Shuffled Bat Algorithm |
| HFCA         | Hybrid Fuzzy Clustering Algorithm |
| HHMA         | Hybrid Heuristic Mathematics Algorithm |
| HKA          | Harmony K-Means Algorithm |
| HS           | Harmony Search |
| HSA          | Harmony Search Algorithm |
| HSCDA        | Hybrid Self-adaptive Community Detection algorithms |
| HSCLLUST     | Harmony Search clustering |
| HSKH         | Harmony Search K-Means Hybrid |
| HS-K-means   | Harmony Search K-Means |
| IABC         | Improved Artificial Bee Colony |
| IBCCOCLUST   | Improved Bee Colony Optimization Clustering |
| ICA          | Imperialist Competitive Algorithm |
| ICAFKM       | Imperialist Competitive Algorithm with Fuzzy K Means |
| Abbreviation | Description |
|--------------|-------------|
| ICAKHM       | Imperial Competitive Algorithm with K-Harmonic Mean |
| ICAKM        | Imperial Competitive Algorithm with K-Means |
| ICGSOSO      | Image Clustering Glowworm Swarm Optimization |
| ICMPKHM      | Improved Cuckoo Search with Modified Particle Swarm Optimization |
| IC          | Intracluster Variation |
| IFCM         | Interactive Fuzzy C-Means |
| IGBHSK      | Global Best Harmony Search K-Means |
| IGNB         | Information Gain-Naïve Bayes |
| IIIEFA       | Inward Intensified Exploration Firefly Algorithm |
| IPSO         | Improved Particle Swarm Optimization |
| IPSO-K- Means | Improved Particle swarm Optimization with K-Means |
| IWO          | Invasive weed optimization |
| IWO-K-Means  | Invasive weed Optimization K-means |
| kABC         | K-Means Artificial Bee Colony |
| KBat         | Bat Algorithm with K-Means Clustering |
| KCPHOSO      | K-Means and Combinatorial Particle Swarm Optimization |
| K-Fa         | K-Means Firefly Algorithm |
| KFCCA        | K-member Fuzzy Clustering and Firefly Algorithm |
| KFCM         | Kernel-based Fuzzy C-Mean Algorithm |
| KGA          | K-Means Genetic Algorithm |
| K-GWO        | Grey wolf optimizer with traditional K-Means |
| KHM          | K-Harmonic Means |
| K-HS         | Harmony K-Means Algorithm |
| KIBCLUS      | K-Means with Improved bee colony |
| KMBA         | K-Means Bat Algorithm |
| KMCLUS       | K-Means Modified Bee Colony K-means |
| KMMeans FFO  | K-Means Fruit fly Optimization |
| KMMeans-ALO  | K-Means with Ant Lion Optimization |
| K-Means-FFA-KELM | Kernel Extreme Learning Machine Model |
| KMGWO        | K-Means Grey wolf optimizer |
| K-MICA       | K-Means Modified Imperialist Competitive Algorithm |
| KMQGA        | Quantum-inspired Genetic Algorithm for K-Means Algorithm |
| KMVGGA       | K-Means clustering algorithm based |
| K-NM-PSO     | K-Means Nelder–Mead Particle Swarm Optimization |
| KNNIR        | K-Nearest Neighbors for Information Retrieval |
| KPA          | K-means with Flower pollination algorithm |
| KPSO         | K-means with Particle Swarm Optimization |
| KSRSO        | K-Means selective regeneration Particle Swarm Optimization |
| LEACH        | Low-Energy Adaptive Clustering Hierarchy |
| MABC-K       | Modified Artificial Bee Colony |
| MAE          | Mean Absolute Error |
| MAX-SAT      | Maximum satisfiability problem |
| MBCO         | Modified Bee Colony K-means |
| MC           | Membrane Computing |
| MCSCO        | Modified Cockroach Swarm Optimization |
| MEQPSO       | Multi-Elitist Quantum-behaved Particle Swarm Optimization |
| MFA          | Modified Firefly Algorithm |
| MFOA         | Modified Fruit Fly Optimization Algorithm |
| MPISO        | Modified Particle Swarm Optimization |
| MICA         | Modified Imperialist Competitive Algorithm |
| MKCLUS       | Modified Bee Colony K-means Clustering |
| MKF-Cuckoo   | Multiple Kernel-Based Fuzzy C-Means with Cuckoo Search |
| MN           | Multimodal Nonseparable function |
| Acronym   | Description                                                                 |
|-----------|------------------------------------------------------------------------------|
| MOA       | Meta-heuristic Optimization Algorithm                                        |
| MPKM      | Modified Point symmetry-based K-Means                                        |
| MSE       | Mean Square Error                                                             |
| MTSP      | Multiple Traveling Salesman Problem                                           |
| NaFA      | Firefly Algorithm with neighborhood attraction                               |
| NGA       | Niche Genetic Algorithm                                                      |
| NGKA      | Niching Genetic K-means Algorithm                                             |
| NM-PSO    | Nelder-Mead simplex search with Particle Swarm Optimization                  |
| NNGA      | Novel Niching Genetic Algorithm                                              |
| Noiseclust| Noise clustering                                                              |
| NR-ELM    | Neighborhood-based ratio (NR) and Extreme Learning Machine (ELM)             |
| NSE       | Nash-Sutcliffe Efficiency                                                    |
| NSL-KDD   | NSL Knowledge Discovery and Data Mining                                       |
| PAM       | Partitioning Around Medoids                                                   |
| PCA       | Principal component analysis                                                  |
| PCA-GAKM  | Principal Component Analysis with Genetic Algorithm and K-means              |
| PCAK      | Principal Component Analysis K-means                                           |
| PCA-SOM   | Principal Component Analysis and Self-Organizing Map                         |
| PCAWK     | Principal component analysis                                                  |
| PGAClust  | Parallel Genetic Algorithm Clustering                                         |
| PGKA      | Prototypes-embedded Genetic K-means Algorithm                                 |
| P-HS      | Progressive Harmony Search                                                    |
| P-HS-K    | Progressive Harmony Search with K-means                                       |
| PIMA      | Indian diabetic dataset                                                      |
| PNSR      | Peak Signal to Noise Ratio                                                    |
| PR        | Precision-Recall                                                             |
| PSC-RCE   | Particle Swarm Clustering with Rapid Centroid Estimation                     |
| PSDWT-FCM | Particle Swarm based Discrete Wavelength Transform with Fuzzy C-Means        |
| PSNR      | Peak Signal-to-Noise Ratio                                                    |
| PSO       | Particle Swarm Optimization                                                   |
| PSO-ACO   | Particle Swarm Optimization and Ant Colony Optimization                       |
| PSO-FCM   | Particle Swarm Optimization with Fuzzy C-means                                |
| PSOFKM    | Particle Swarm Optimization with Fuzzy K-means                                |
| PSOK      | Particle Swarm Optimization with K-means based clustering                     |
| PSOKHM    | Particle Swarm Optimization with K-Harmonic Mean                              |
| PSO-KM    | PSO-based K-means clustering algorithm                                        |
| PSOLF-KHM | Particle Swarm Optimization with Levy and K-Harmonic Mean Algorithm          |
| PSOM      | Particle Swarm optimization with mutation operation                          |
| PSO-SA    | Particle Swarm Optimization with Simulated Annealing                          |
| PSO-SVR   | Particle Swarm Optimization based Support Vector Regression                  |
| PTM       | Pattern Taxonomy Mining                                                      |
| QALO-K    | Quantum Ant Lion Optimizer with K-Means                                      |
| rCMA-ES   | restart Covariance Matrix Adaptation Evolution Strategy                       |
| RMSE      | Root Mean Square Error                                                        |
| ROC       | Receive Operating Characteristics                                             |
| RSC       | Relevant Set Correlation clustering model                                     |
| RVPSO-K   | K-Means cluster algorithm based on Improved velocity                         |
| RWFOA     | Fruit Fly Optimization based on Stochastic Inertia Weight                    |
| SA        | Simulated Annealing                                                          |
| SaNSDE    | Self-adaptive Differential Evolution with Neighborhood Search                |
| SAR       | Synthetic Aperture Radar                                                      |
| SCA       | Sine-Cosine Algorithm                                                        |
| SCAK-Means| Sine-Cosine Algorithm with K-means                                            |
| SD        | Standard Deviation                                                           |
| SDM       | Sexual Determination Method                                                   |
| SDME      | Second Derivative-like Measure of Enhancements                               |
SDN Software defined Network
SDS Stochastic Diffusion Search
SFLA-CQ Shuffled frog leaping algorithm for Color quantization
SHADE Success-History based Adaptive Differential Evolution
SI Scatter Index
SI Silhouette Index
SIM dataset Simulated dataset
SMEER Secure multi-tier energy-efficient routing protocol
SOM Self-Organizing Feature Maps
SOM+K Self-Organizing Feature Maps neural networks with K-Means
SRPSO Selective Regeneration Particle Swarm Optimization
SSB Sum of Square Between
SSE Sum of Square Error
SSIM Structural Similarity
SS-KMeans Scattering search K-Means
SSO Social Spider Optimization
SSOKC Social Spider Optimization with K-Means Clustering
SSW Sum of Square within
SVC Support Vector Clustering
SVM+GA Support Vector Machine with Genetic Algorithm
SVMIR Support Vector Machine for Information Retrieval
TCSC Thyristor Controlled Series Compensator
TKMC Traditional K-means Clustering
TP True Positivity Rate
TPR True Positivity Rate
TREC Text Retrieval Conference dataset
TS Tabu Search
TSMPSO Two-Stage diversity mechanism
in Multiobjective Particle Swarm Optimization
TSP-LIB-1600 dataset for Travelling Salesman Problem
TSP-LIB-3038 dataset for Travelling Salesman Problem
UCC U-Control Chart
UCI University of California Irvine
UN Unimodal Nonseparable function
UPFC Unified Power Flow Controller
US Unimodal Separable function
VGA Variable string length Genetic Algorithm
VSGSO-D K-means Variable Step-size glowworm swarm optimization
VSSFA Variable Step size firefly Algorithm
WDRC Wisconsin Diagnostic Breast Cancer
WHDA-FCM Wolf hunting based dragonfly with Fuzzy C-Means
WK-Means Weight-based K-Means
WOA Whale Optimization Algorithm
WOA-BAT Whale Optimization Algorithm withBat Algorithm
WSN Wireless Sensor Networks

References

1. Ezugwu, A.E. Nature-inspired metaheuristic techniques for automatic clustering: A survey and performance study. SN Appl. Sci. 2020, 2, 273. [CrossRef]
2. MacQueen, J. Some Methods for Classification and Analysis of Multivariate Observations. Am. J. Hum. Genet. 1969, 21, 407–408.
3. Kapil, S.; Chawla, M.; Ansari, M.D. On K-Means Data Clustering Algorithm with Genetic Algorithm. In Proceedings of the 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC), Solan, India, 22–24 December 2016; pp. 202–206.
4. Ezugwu, A.E.-S.; Agbaje, M.B.; Aljojo, N.; Els, R.; Chiroma, H.; Elaziz, M.A. A Comparative Performance Study of Hybrid Firefly Algorithms for Automatic Data Clustering. IEEE Access 2020, 8, 121089–121118. [CrossRef]
5. Ezugwu, A.E.; Shukla, A.K.; Agbaje, M.B.; Oyelade, O.N.; José-García, A.; Agushaka, J.O. Automatic clustering algorithms: A systematic review and bibliometric analysis of relevant literature. Neural Comput. Appl. 2020, 33, 6247–6306. [CrossRef]
6. José-García, A.; Gómez-Flores, W. Automatic clustering using nature-inspired metaheuristics: A survey. Appl. Soft Comput. 2016, 41, 192–213. [CrossRef]

7. Hruschka, E.; Campello, R.J.G.B.; Freitas, A.A.; de Carvalho, A. A Survey of Evolutionary Algorithms for Clustering. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. 2009, 39, 133–155. [CrossRef]

8. Ezugwu, A.E.; Shukla, A.K.; Nith, R.; Akinyelu, A.A.; Agushaka, J.O.; Chiroma, H.; Muhuri, P.K. Metaheuristics: A comprehensive overview and classification along with bibliometric analysis. Artif. Intell. Rev. 2021, 54, 4237–4316. [CrossRef]

9. Rana, S.; Jasola, S.; Kumar, R. A review on particle swarm optimization algorithms and their applications to data clustering. Artif. Intell. Rev. 2010, 35, 211–222. [CrossRef]

10. Nanda, S.J.; Panda, G. A survey on nature inspired metaheuristic algorithms for partitional clustering. Swarm Evol. Comput. 2014, 16, 1–18. [CrossRef]

11. Alam, S.; Dobbie, G.; Koh, Y.S.; Riddle, P.; Rehman, S.U. Research on particle swarm optimization based clustering: A systematic review of literature and techniques. Swarm Evol. Comput. 2014, 17, 1–13. [CrossRef]

12. Mane, S.U.; Gaikwad, P.G. Nature Inspired Techniques for Data Clustering. In Proceedings of the 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), Mumbai, India, 4–5 April 2014; pp. 419–424.

13. Falkenauer, E. Genetic Algorithms and Grouping Problems; John Wiley & Sons, Inc.: London, UK, 1998.

14. Cowgill, M.; Harvey, R.; Watson, L. A genetic algorithm approach to cluster analysis. Comput. Math. Appl. 1999, 37, 99–108. [CrossRef]

15. Okwu, M.O.; Tartibu, L.K. Metaheuristic Optimization: Nature-Inspired Algorithms Swarm and Computational Intelligence, Theory and Applications; Springer Nature: Berlin/Heidelberg, Germany, 2020; Volume 927.

16. Malik, K.; Tayal, A. Comparison of Nature Inspired Metaheuristic Algorithms. Int. J. Electron. Electr. Eng. 2014, 7, 799–802.

17. Engelbrecht, A.P. Computational Intelligence: An Introduction; John Wiley & Sons: London, UK, 2007.

18. Agbaje, M.B.; Ezugwu, A.E.; Els, R. Automatic Data Clustering Using Hybrid Firefly Particle Swarm Optimization Algorithm. IEEE Access 2019, 7, 184963–184984. [CrossRef]

19. Rajakumar, R.; Dhavachelvan, P.; Vengattaraman, T. A Survey on Nature Inspired Meta-Heuristic Algorithms with its Domain Specifications. In Proceedings of the 2016 International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 21–26 October 2016; pp. 1–6.

20. Ezugwu, A.E. Advanced discrete firefly algorithm with adaptive mutation–based neighborhood search for scheduling unrelated parallel machines with sequence–dependent setup times. Int. J. Intell. Syst. 2021, 1–42. [CrossRef]

21. Holland, J.H. Genetic algorithms. Sci. Am. 1992, 267, 66–73. [CrossRef]

22. Sivanandam, S.N.; Deepa, S.N. Genetic algorithms. In Introduction to Genetic Algorithms; Springer: Berlin/Heidelberg, Germany, 2008; pp. 15–37.

23. Krishna, K.; Murty, M.N. Genetic K-means algorithm. IEEE Trans. Syst. Man Cybern. Part B 1999, 29, 433–439. [CrossRef] [PubMed]

24. Bandyopadhyay, S.; Maulik, U. An evolutionary technique based on K-means algorithm for optimal clustering in RN. Inf. Sci. 2002, 146, 221–237. [CrossRef]

25. Cheng, S.S.; Chao, Y.H.; Wang, H.M.; Fu, H.C. A prototypes-embedded genetic k-means algorithm. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR’06), Hong Kong, China, 20–24 August 2006; Volume 2, pp. 724–727.

26. Laszlo, M.; Mukherjee, S. A genetic algorithm using hyper-quadtrees for low-dimensional k-means clustering. IEEE Trans. Pattern Anal. Mach. Intell. 2006, 28, 533–543. [CrossRef] [PubMed]

27. Laszlo, M.; Mukherjee, S. A genetic algorithm that exchanges neighboring centers for k-means clustering. Pattern Recognit. Lett. 2007, 28, 2359–2366. [CrossRef]

28. Dai, W.; Jiao, C.; He, T. Research of K-Means Clustering Method based on Parallel Genetic Algorithm. In Proceedings of the Third International Conference on Intelligent Information Hiding and Multimedia Signal. Processing (IIH-MSP 2007), Kaohsiung, Taiwan, 26–28 November 2007; Volume 2, pp. 158–161.

29. Chang, D.-X.; Zhang, X.-D.; Zheng, C.-W. A genetic algorithm with gene rearrangement for K-means clustering. Pattern Recognit. 2009, 42, 1210–1222. [CrossRef]

30. Sheng, W.; Tucker, A.; Liu, X. A niching genetic k-means algorithm and its applications to gene expression data. Soft Comput. 2008, 14, 9–19. [CrossRef]

31. Xiao, J.; Yan, Y.; Zhang, J.; Tang, Y. A quantum-inspired genetic algorithm for k-means clustering. Expert Syst. Appl. 2010, 37, 4966–4973. [CrossRef]

32. Rahman, M.A.; Islam, M.Z. A hybrid clustering technique combining a novel genetic algorithm with K-Means. Knowl.-Based Syst. 2014, 71, 345–365. [CrossRef]

33. Sinha, A.; Jana, P.K. A Hybrid MapReduce-based k-Means Clustering using Genetic Algorithm for Distributed Datasets. J. Supercomput. 2018, 74, 1562–1579. [CrossRef]

34. Islam, M.Z.; Estivill-Castro, V.; Rahman, M.A.; Bossomaier, T. Combining K-Means and a genetic algorithm through a novel arrangement of genetic operators for high quality clustering. Expert Syst. Appl. 2018, 91, 402–417. [CrossRef]
35. Mustafi, D.; Sahoo, G. A hybrid approach using genetic algorithm and the differential evolution heuristic for enhanced initialization of the k-means algorithm with applications in text clustering. Soft Comput. 2019, 23, 6361–6378. [CrossRef]

36. El-Shorbagy, M.A.; Ayoub, A.Y.; Moussa, A.A.; El-Desoky, I.M. An enhanced genetic algorithm with new mutation for cluster analysis. Comput. Stat. Stat. 2019, 34, 1355–1392. [CrossRef]

37. Ghezelbash, R.; Magsoudi, A.; Carranza, E.J.M. Optimization of geochemical anomaly detection using a novel genetic K-means clustering (GKMC) algorithm. Comput. Geosci. 2019, 134, 104335. [CrossRef]

38. Kuo, R.; An, Y.; Wang, H.; Chung, W. Integration of self-organizing feature maps neural network and genetic K-means algorithm for market segmentation. Expert Syst. Appl. 2006, 30, 313–324. [CrossRef]

39. Li, X.; Zhang, L.; Li, Y.; Wang, Z. An Improved k-Means Clustering Algorithm Combined with the Genetic Algorithm. In the 6th International Conference on Digital Content, Multimedia Technology and Its Applications, Seoul, Korea, 16–18 August 2010; pp. 121–124.

40. Kao, Y.-T.; Zahara, E.; Kao, I.-W. A hybridized approach to data clustering. Expert Syst. Appl. 2013, 366–377. [CrossRef]

41. Karegowda, A.G.; Vidya, T.; Jayaram, M.A.; Manjunath, A.S. Improving Performance of k-Means Clustering by Initializing Cluster Centers using Genetic Algorithm and Entropy based Fuzzy Clustering for Categorization of Diabetic Patients. In Proceedings of International Conference on Advances in Computing; Springer: New Delhi, India, 2013; pp. 899–904.

42. Eshlaghy, A.T.; Razi, F.F. A hybrid grey-based k-means and genetic algorithm for project selection. Int. J. Bus. Inf. Syst. 2015, 18, 141. [CrossRef]

43. Lu, Z.; Zhang, K.; He, J.; Niu, Y. Applying k-Means Clustering and Genetic Algorithm for Solving MTSP. In International Conference on Bio-Inspired Computing: Theories and Applications; Springer: Singapore, 2016; pp. 278–284.

44. Barekatain, B.; Dehghani, S.; Pourazaferani, M. An Energy-Aware Routing Protocol for Wireless Sensor Networks Based on New Combination of Genetic Algorithm & k-means. Procedia Comput. Sci. 2015, 72, 552–560.

45. Zhou, X.; Gu, J.; Shen, S.; Ma, H.; Miao, F.; Zhang, H.; Gong, H. An Automatic K-Means Clustering Algorithm of GPS Data Combining a Novel Niche Genetic Algorithm with Noise and Density. ISPRS Int. J. Geo-Inf. 2017, 6, 392. [CrossRef]

46. Mohammadrezapour, O.; Kisi, O.; Pourahmad, F. Fuzzy c-means and K-means clustering with genetic algorithm for identification of homogeneous regions of groundwater quality. Neural Comput. Appl. 2018, 32, 3763–3775. [CrossRef]

47. Esmin, A.A.A.; Coelho, R.A.; Matwin, S. A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data. Artif. Intell. Rev. 2013, 44, 23–45. [CrossRef]

48. Niu, B.; Duan, Q.; Liu, J.; Tan, L.; Liu, Y. A population-based clustering technique using particle swarm optimization and k-means. Nat. Comput. 2016, 16, 45–59. [CrossRef]

49. Ni, X.; Zhu, H.; Lin, J.; Li, J.; Liu, S. An improved k-means clustering algorithm using the genetic algorithm. Expert Syst. Appl. 2017, 87, 125–134. [CrossRef]

50. Omran, M.G.H.; Salman, A.; Engelbrecht, A.P. Dynamic clustering using particle swarm optimization with application in image segmentation. Pattern Anal. Appl. 2005, 8, 332–344. [CrossRef]

51. Alam, S.; Dobbie, G.; Riddle, P. An Evolutionary Particle Swarm Optimization Algorithm for Data Clustering. In Proceedings of the 2008 IEEE Swarm Intelligence Symposium, St. Louis, MO, USA, 21–23 September 2008; pp. 1–7.

52. Kao, I.W.; Tsai, C.Y.; Wang, Y.C. An effective particle swarm optimization method for data clustering. In Proceedings of the 2008 IEEE Congress on Evolutionary Computation, CEC’08, Beijing, China, 18–21 May 2008; Volume 1, pp. 111–118. [CrossRef]

53. Niknam, T.; Amirb, B. An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. Appl. Soft Comput. 2010, 10, 183–197. [CrossRef]

54. Thangaraj, R.; Pant, M.; Abraham, A.; Bouvry, P. Particle swarm optimization: Hybridization perspectives and experimental illustrations. Appl. Math. Comput. 2011, 217, 5208–5226. [CrossRef]

55. Chuang, L.-Y.; Hsiao, C.-J.; Yang, C.-H. Chaotic particle swarm optimization for data clustering. Expert Syst. Appl. 2011, 38, 14555–14563. [CrossRef]

56. Chen, C.-Y.; Ye, F. Particle Swarm Optimization Algorithm and its Application to Clustering Analysis. In Proceedings of the 17th Conference on Electrical Power Distribution, Tehran, Iran, 2–3 May 2012; pp. 789–794.

57. Yuwono, M.; Su, S.W.; Moulton, B.D.; Nguyen, H.T. Data clustering using variants of rapid centroid estimation. IEEE Trans. Evol. Comput. 2013, 18, 366–377. [CrossRef]

58. Omran, M.; Engelbrecht, A.P.; Salman, A. Particle swarm optimization method for image clustering. Int. J. Pattern Recognit. Artif. Intell. 2005, 19, 297–321. [CrossRef]

59. Chen, J.; Zhang, H. Research on Application of Clustering Algorithm based on PSO for the Web Usage Pattern. In Proceedings of the 2007 International Conference on Wireless Communications, Networking and Mobile Computing, Honolulu, HI, USA, 21–25 September 2007; pp. 3705–3708.

60. Kuo, R.; Kuo, I.-W. A hybridized approach to data clustering. Expert Syst. Appl. 2008, 34, 1754–1762. [CrossRef]

61. Kao, Y.; Lee, S.Y. Combining K-Means and Particle Swarm Optimization for Dynamic Data Clustering Problems. In Proceedings of the 2009 IEEE International Conference on Intelligent Computing and Intelligent Systems, Shanghai, China, 20–22 November 2009; Volume 1, pp. 757–761.
116. Singh, S.P.; Solanki, S. A Movie Recommender System Using Modified Cuckoo Search. In *Emerging Research in Electronics, Computer Science and Technology*. Springer: Singapore, 2019; pp. 471–482.

117. Arjmand, A.; Meshgini, S.; Afrouzian, R.; Farzamnia, A. Breast Tumor Segmentation Using K-Means Clustering and Cuckoo Search Optimization. In Proceedings of the 9th International Conference on Computer and Knowledge Engineering (ICCKE), Virtual, 24–25 October 2019; pp. 305–309.

118. Garcia, J.; Yepes, V.; Martí, J.V. A Hybrid k-Means Cuckoo Search Algorithm Applied to the Counterfertaining Walls Problem. *Mathematics* **2020**, *8*, 555. [CrossRef]

119. Binu, D.; Selvi, M.; George, A. MKF-Cuckoo: Hybridization of Cuckoo Search and Multiple Kernel-based Fuzzy C-means Algorithm. *AASRI Procedia* **2013**, *4*, 243–249. [CrossRef]

120. Manju, V.N.; Fred, A.L. An efficient multi balanced cuckoo search k-means technique for segmentation and compression of compound images. *Multimed. Tools Appl.* **2019**, *78*, 14897–14915. [CrossRef]

121. Deepa, M.; Sumitra, P. Intrusion Detection System Using K-Means Based on Cuckoo Search Optimization. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *993*, 012049. [CrossRef]

122. Storn, R.; Price, K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]

123. Brest, J.; Maučec, M.S. Population size reduction for the differential evolution algorithm. *Appl. Intell.* **2007**, *29*, 228–247. [CrossRef]

124. Kwedlo, W. A clustering method combining differential evolution with the K-means algorithm. *Pattern Recognit. Lett.* **2011**, *32*, 1613–1621. [CrossRef]

125. Mehrabian, A.; Lucas, C. A novel numerical optimization algorithm inspired from weed colonization. *Ecol. Inform.* **2006**, *1*, 355–366. [CrossRef]

126. Fan, C.; Zhang, T.; Yang, Z.; Wang, L. A Text Clustering Algorithm Hybirding Invasive Weed Optimization with K-Means. In Proceedings of the 12th International Conference on Autonomic and Trusted Computing and 2015 IEEE 15th International Conference on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom), Beijing, China, 10–14 August 2015; pp. 1333–1338. [CrossRef]

127. Singh, S.P.; Solanki, S.; Varela, N.B.P.; Varela, N.; Guiliany, J.G.; Sanabria, E.S.; Otero, M.S.; Rojas, V. U-Control Chart Based Differential Evolution Clustering for Determining the Number of Cluster in k-Means. In *International Conference on Green, Pervasive, and Cloud Computing*. Springer: Cham, Switzerland, 2019; pp. 31–41. [CrossRef]

128. Sheng, W.; Wang, X.; Wang, Z.; Li, Q.; Zheng, Y.; Chen, S. A Differential Evolution Algorithm with Adaptive Niching and K-Means Operation for Data Clustering. *IEEE Trans. Cybern.* **2020**, *1–15*. [CrossRef]

129. Mustafi, D.; Mustafi, A.; Sahoo, G. A novel approach to text clustering using genetic algorithm based on the nearest neighbour heuristic. *Int. J. Comput. Appl.* **2020**, *1–13*. [CrossRef]

130. Breit, J.; Yasid, A. Automatic clustering combining differential evolution algorithm and k-means algorithm. In *Proceedings of the 29th International Conference on Industrial Engineers Asian Conference 2013*. Springer: Singapore, 2013; pp. 1207–1215.

131. Sierra, L.M.; Cobos, C.; Corrales, J.C. Continuous Optimization based on a Hybridization of Differential Evolution with K-Means. In *IBERO-American Conference on Artificial Intelligence*. Springer: Cham, Switzerland, 2014; pp. 381–392.

132. Hu, J.; Wang, C.; Liu, C.; Ye, Z. Improved K-Means Algorithm based on Hybrid Fruit Fly Optimization and Differential Evolution. In Proceedings of the 12th International Conference on Computer Science and Education (ICCSE), Houston, TX, USA, 22–25 August 2017; pp. 464–467.

133. Wang, F. A Weighted K-Means Algorithm based on Differential Evolution. In Proceedings of the 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control. Conference (IMCEC), Xi’an, China, 25–27 May 2018; pp. 1–2274.

134. Silva, J.; Lezama, O.B.P.; Varela, N.; Guiliany, J.G.; Sanabria, E.S.; Otero, M.S.; Rojas, V. U-Control Chart Based Differential Evolution Clustering for Determining the Number of Cluster in k-Means. In *International Conference on Green, Pervasive, and Cloud Computing*. Springer: Cham, Switzerland, 2019; pp. 31–41. [CrossRef]

135. Pan, G.; Li, K.; Ouyang, A.; Zhou, X.; Xu, Y. A hybrid clustering algorithm combining cloud model IWO and k-means. *Int. J. Pattern Recogn. Artif. Intell.* **2014**, *28*, 1450015. [CrossRef]

136. Boobord, F.; Othman, Z.; Abubakar, A. PCAWK: A Hybridized Clustering Algorithm Based on PCA and WK-means for Large Size of Dataset. *Int. J. Adv. Soft Comput. Appl.* **2015**, *7*, 3.

137. Razi, F.F. A hybrid DEA-based K-means and invasive weed optimization for facility location problem. *J. Ind. Eng. Int.* **2018**, *15*, 499–511. [CrossRef]

138. Atashpaz-Gargari, E.; Lucas, C. Imperialist Competitive Algorithm: An Algorithm for Optimization Inspired by Imperialistic Competition. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation, Singapore, 25–28 September 2007.

139. Niknam, T.; Fard, E.T.; Pourjafarian, N.; Rousta, A. An efficient hybrid algorithm based on modified imperialist competitive algorithm and K-means for data clustering. *Eng. Appl. Intell. Artif.* **2011**, *24*, 306–317. [CrossRef]

140. Abdeyazdan, M. Data clustering based on hybrid K-harmonic means and modifier imperialist competitive algorithm. *J. Supercomput.* **2014**, *68*, 574–598. [CrossRef]

141. Forsati, R.; Meybodi, M.; Mahdavi, M.; Neiat, A. Hybridization of K-Means and Harmony Search Methods for Web Page Clustering. In Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Melbourne, Australia, 14–17 December 2020; Volume 1, pp. 329–335.
169. Cuevas, E.; Cienfuegos, M.; Zaldívar, D.; Pérez-Cisneros, M. A swarm optimization algorithm inspired in the behavior of the social-spider. *Expert Syst. Appl.* 2013, 40, 6374–6384. [CrossRef]

170. Chandran, T.R.; Reddy, A.V.; Janet, B. Performance Comparison of Social Spider Optimization for Data Clustering with Other Clustering Methods. In Proceedings of the 2018 Second International Conference on Intelligent Computing and Control. Systems (ICICCS), Madurai, India, 14–15 June 2018; pp. 1119–1125.

171. Thiruvennikateshwar, M.P.; Venkatachalam, V. Analysis and evaluation of classification and segmentation of brain tumour images. *Int. J. Biomed. Eng. Technol.* 2019, 30, 153–178. [CrossRef]

172. Xing, B.; Gao, W.J. Fruit fly optimization algorithm. In *Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms*; Springer: Cham, Switzerland, 2014; pp. 167–170.

173. Sharma, V.K.; Patel, R. Unstructured Data Clustering using Hybrid K-Means and Fruit Fly Optimization (KMeans-FFO) algorithm. *Int. J. Comput. Sci. Inf. Secur.* (IJCIS) 2020, 18.

174. Jiang, X.Y.; Pa, N.Y.; Wang, W.C.; Yang, T.T.; Pan, W.T. Site Selection and Layout of Earthquake Rescue Center Based on K-Means Clustering and Fruit Fly Optimization Algorithm. In Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 27–29 June 2020; pp. 1381–1389.

175. Gowdhams, D.; Thangavel, K.; Kumar, E.S. Fruit Fly K-Means Clustering Algorithm. *Int. J. Scient. Res. Sci. Eng. Technol.* 2016, 2, 156–159.

176. Wang, Q.; Zhang, Y.; Xiao, Y.; Li, J. Kernel-based Fuzzy C-Means Clustering based on Fruit Fly Optimization Algorithm. In Proceedings of the 2017 International Conference on Grey Systems and Intelligent Services (GSIS), Stockholm, Sweden, 8–11 August 2017; pp. 251–256.

177. Drias, H.; Sadeg, S.; Yahi, S. Cooperative Bees Swarm for Solving the Maximum Weighted Satisfiability Problem. In *International Work-Conference on Artificial Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 318–325.

178. Djennouri, Y.; Belhadi, A.; Belkebir, R. Bees swarm optimization guided by data mining techniques for document information retrieval. *Expert Syst. Appl.* 2018, 94, 126–136. [CrossRef]

179. Aboubi, Y.; Drias, H.; Kamel, N. BSO-CLARA: Bees Swarm Optimization for Clustering Large Applications. In *International Conference on Mining Intelligence and Knowledge Exploration*; Springer: Cham, Switzerland, 2015; pp. 170–183.

180. Djennouri, Y.; Habbas, Z.; Aggounne-Mtalaa, W. Bees Swarm Optimization Metaheuristic Guided by Decomposition for Solving MAX-SAT. *ICAART 2016*, 2, 472–479.

181. Li, M.; Yang, C.W. Bacterial colony optimization algorithm. *Control Theory Appl.* 2011, 28, 223–228.

182. Revathi, J.; Eswaramurthy, V.P.; Padmavathi, P. Hybrid data clustering approaches using bacterial colony optimization and k-means. *IOP Conf. Ser. Mater. Sci. Eng.* 2021, 1070, 012064. [CrossRef]

183. Vijayakumari, K.; Deepa, V.B. Hybridization of Fuzzy C-Means with Bacterial Colony Optimization. 2019. Available online: [http://infokara.com/gallery/118-dec-3416.pdf](http://infokara.com/gallery/118-dec-3416.pdf) (accessed on 18 September 2021).

184. Al-Rifaie, M.M.; Bishop, J.M. Stochastic Diffusion Search Review. *Paladyn J. Behav. Robot.* 2012, 1, 155–173. [CrossRef]

185. Karthik, J.; Tamizhazhagan, V.; Narayana, S. Data leak identification using scattering search K Meas in social networks. *Mater. Today Proc.* 2021. [CrossRef]

186. Fathian, M.; Amiri, B.; Maroosi, A. Application of honey-bee mating optimization algorithm on clustering. *Appl. Math. Comput.* 2007, 190, 1502–1513. [CrossRef]

187. Teodorović, D. Bee colony optimization (BCO). In *Innovations in Swarm Intelligence*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 39–60.
196. Das, P.; Das, D.K.; Dey, S. A modified Bee Colony Optimization (MBCO) and its hybridization with k-means for an application to data clustering. *Appl. Soft Comput.* 2018, 70, 590–603. [CrossRef]

197. Forsati, R.; Keikha, A.; Shamsfard, M. An improved bee colony optimization algorithm with an application to document clustering. *Neurocomputing* 2015, 159, 9–26. [CrossRef]

198. Yang, C.-L.; Sutrisno, H. A clustering-based symbiotic organisms search algorithm for high-dimensional optimization problems. *Appl. Soft Comput.* 2020, 97, 106722. [CrossRef]

199. Zhang, D.; Leung, S.C.; Ye, Z. A Decision Tree Scoring Model based on Genetic Algorithm and k-Means Algorithm. In Proceedings of the Third International Conference on Convergence and Hybrid Information Technology, Busan, Korea, 11–13 November 2008; Volume 1, pp. 1043–1047.

200. Patel, R.; Raghuwanshi, M.M.; Jaiswal, A.N. Modifying Genetic Algorithm with Species and Sexual Selection by using K-Means Algorithm. In Proceedings of the 2009 IEEE Advanced Computing Conference, Patiala, India, 6–7 March 2009; pp. 114–119.

201. Niu, B.; Duan, Q.; Liang, J. Hybrid Bacterial Foraging Algorithm for Data Clustering. In *International Conference on Intelligent Data Engineering and Automated Learning*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 577–584.

202. Charon, I.; Hudry, O. The noising method: A new method for combinatorial optimization. *Oper. Res. Lett.* 1993, 14, 133–137. [CrossRef]

203. Arthur, D.; Vassilvitskii, S. K-means++: The advantages of careful seeding. In Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms, New Orleans, LA, USA, 7–9 January 2007.

204. Pykett, C. Improving the efficiency of Sammon’s nonlinear mapping by using clustering archetypes. *Electron. Lett.* 1978, 14, 799–800. [CrossRef]

205. Lee, R.C.T.; Slagle, J.R.; Blum, H. A triangulation method for the sequential mapping of points from N-space to two-space. *IEEE Trans. Comput.* 1977, 26, 288–292. [CrossRef]

206. McCallum, A.; Nigam, K.; Ungar, L.H. Efficient clustering of high-dimensional data sets with application to reference matching. In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Boston, MA, USA, 1 August 2000; pp. 169–178.

207. Wikaisuksakul, S. A multi-objective genetic algorithm with fuzzy c-means for automatic data clustering. *Appl. Soft Comput.* 2014, 24, 679–691. [CrossRef]

208. Neath, A.A.; Cavanaugh, J.E. The Bayesian information criterion: Background, derivation, and applications. *Wiley Interdiscip. Rev. Comput. Stat.* 2012, 4, 199–203. [CrossRef]

209. Davies, D.L.; Bouldin, D.W. A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.* 1979, 224–227. [CrossRef]