IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 30, 2020, accepted June 27, 2020, date of publication July 1, 2020, date of current version July 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3006173

A Comparative Performance Study of Hybrid Firefly Algorithms for Automatic Data Clustering

ABSALOM EL-SHAMIR EZUGWU[1], (Member, IEEE), MOYINOLUWA B. AGBAJE[1], NAHLA ALJOJO[2], ROSANNE ELS[1], HARUNA CHIROMA[3], (Member, IEEE), AND MOHAMED ABD ELAZIZ[4]

[1]School of Computer Science, University of KwaZulu-Natal, Pietermaritzburg Campus, Pietermaritzburg 3201, South Africa
[2]Department of Information System and Technology, College of Computer Science and Engineering, University of Jeddah, Jeddah 23218, Saudi Arabia
[3]Future Technology Research Center, National Yunlin University of Science and Technology, Douliu 64002, Taiwan
[4]Department of Mathematics, Zagazig University, Zagazig 14459, Egypt

Corresponding authors: Absalom El-Shamir Ezugwu (ezugwua@ukzn.ac.za) and Haruna Chiroma (freedonchi@yahoo.com)

ABSTRACT In cluster analysis, the goal has always been to extemporize the best possible means of automatically determining the number of clusters. However, because of lack of prior domain knowledge and uncertainty associated with data objects characteristics, it is challenging to choose an appropriate number of clusters, especially when dealing with data objects of high dimensions, varying data sizes, and density. In the last few decades, different researchers have proposed and developed several nature-inspired metaheuristic algorithms to solve data clustering problems. Many studies have shown that the firefly algorithm is a very robust, efficient and effective nature-inspired swarm intelligence global search technique, which has been successfully applied to solve diverse NP-hard optimization problems. However, the diversification search process employed by the firefly algorithm can lead to reduced speed and convergence rate for large-scale optimization problems. Thus this study investigates the application of four hybrid firefly algorithms to the task of automatic clustering of high density and large-scaled unlabelled datasets. In contrast to most of the existing classical heuristic-based data clustering analyses techniques, the proposed hybrid algorithms do not require any prior knowledge of the data objects to be classified. Instead, the hybrid methods automatically determine the optimal number of clusters empirically and during the program execution. Two well-known clustering validity indices, namely the Compact-Separated and Davis-Bouldin indices, are employed to evaluate the superiority of the implemented firefly hybrid algorithms. Furthermore, twelve standard ground truth clustering datasets from the UCI Machine Learning Repository are used to evaluate the robustness and effectiveness of the algorithms against those of the classical swarm optimization algorithms and other related clustering results from the literature. The experimental results show that the new clustering methods depict high superiority in comparison with existing standalone and other hybrid metaheuristic techniques in terms of clustering validity measures.

INDEX TERMS Automatic clustering, firefly algorithm, firefly-based hybrid algorithms, clustering validity index.

I. INTRODUCTION

Data clustering is an important unsupervised classification technique, which involves the process of grouping data so that similar items are grouped into clusters based on some similarity metric [1]–[4]. Clustering is often used for a variety of fascinating real-world applications such as in marketing, biology, image analysis, libraries, insurance, data mining, medicine, statistical data analysis, community detection, and other fields of science and engineering [5]–[7]. Although cluster analysis was first used in two social sciences domains, namely, anthropology and psychology [8], furthermore, it was also used for trait theory classification in personality psychology by Cattell in early 1943 [8], [9]. The method of data clustering has since spread with significant relevance in application to other new research areas such as data science and machine learning. It is equally noteworthy to mention here that clustering data into meaningful groups is an important task of both artificial intelligence and data mining. In general term, clustering is considered

The associate editor coordinating the review of this manuscript and approving it for publication was Shaoyong Zheng.

to be an unsupervised classification of data, of which the results of the analysis greatly depend on the superiority and effectiveness of the clustering algorithms or methods employed.

In the past few decades, several heuristic-based algorithms have been proposed to solve clustering problems. Each of these algorithms is designed and implemented based on the two main classifications of clustering methods, namely, hierarchical and partitioning clustering algorithms [10], [11]. Hierarchical clustering algorithms generate a tree-like hierarchical structure which represents a nested grouping of data points. The most popular of these algorithms are the single-link and complete-link algorithms [12]. In the other hand, partitioning clustering algorithms distribute data points into non-overlapping clusters such that each data points belongs to only one cluster. In other words, the partitioning clustering algorithms produce single data partitions instead of constructing a tree-like structure, as it is the case for hierarchical clustering algorithms [13]. One major challenge with these algorithms is how to select an appropriate number of output clusters. The k-means algorithm seems to be the most popular among these algorithms. However, the success of the algorithms mentioned above in solving clustering analyses problems highly rely on having predetermined information about the data objects and the initial solution, which in most case can easily lead the algorithms into getting trapped around local optima [8]. These are serious drawbacks that have led data mining researchers to improvise and come up with other effective means of overcoming these defects among which includes the use of several evolutionary and swarm intelligence algorithms to deal with more complex and high dimensional data clustering problems.

Some of the evolutionary algorithms that have been employed to handle data clustering problem are genetic algorithm (GA) and differential evolution (DE), while several swarm intelligence techniques such as particle swarm optimization (PSO), ant colony optimization (ACO), firefly algorithm (FA), invasive weed optimization (IWO), artificial bee colony optimization (ABC), and teaching learning-based optimization (TLBO) have as well been effectively applied to solve clustering problems [14], [15]. For examples, Zabihi and Nasiri [16] proposed the use of a history-driven artificial bee colony algorithm to solve data clustering problem, for which a memory mechanism that is based on a binary space partitioning was incorporated into the ABC algorithm to improve its clustering performance. Merwe and Engelbrecht were the first to propose the use of PSO to solve clustering problems [17]. Similarly, Zhao *et al.* [18], worked on improving the performance of the k-mean algorithm by hybridizing it with PSO to avoid the algorithm's performance from directly being affected by the original cluster centers. Liu *et al.* [15] develop a genetic algorithm-based automatic clustering method that was able to find good quality clustering solutions for an unknown cluster. Niknam *et al.* [19] proposed an efficient hybrid evolutionary algorithm that combined ACO and simulated annealing (SA) algorithms to solve clustering analysis problem. The simulation results of the ACO-SA showed that the hybrid algorithm outperformed the basic SA, ACO and k-means respectively for partitional clustering problem. Satapathy and Naik [20] developed a TLBO algorithm that was used to find the centroids of a user-specified number of clusters. In another related study, Sahoo and Kumar [21] proposed two different modifications for the TLBO method to enhance its performance in clustering domain, in which instead of random initialization a predefined method previously used to exploit initial cluster centers was exploited. Zhao and Zhou [22] proposed an improved kernel possibilistic fuzzy c-means algorithm based on IWO algorithm for clustering analysis problem, while Liu *et al.* [23] employed multi-objective IWO algorithm to solve clustering problem. In the study carried out by Wang *et al.* [24], a flower pollination algorithm (FPA) with bee pollinators was proposed to solve cluster problem, while Agarwal and Mehta [25] studied application an enhanced flower pollination algorithm to solve data clustering problem. In recent times different authors have also considered. Senthilnath *et al.* [26] conducted a performance evaluation study on the use of standard FA to solve clustering problem and its results compared with that of the PSO, ABA, and other classical based clustering algorithms from the literature. Furthermore, a similar study on the performance analysis of the firefly algorithm for data clustering was also considered in [27] by Banati and Bajaj. At the same time, in 2012, Abshouri and Bakhtiary [28] proposed a new hybrid FA that combines FA and K-Harmonic Means algorithm to solve data clustering problem.

However, most of the clustering problem where the algorithms mentioned above have been tested and proved to yield superior quality solutions required the algorithms to be supplied with specific prior knowledge of the data objects characteristics and features. For example, specifying the number of clusters and other related dataset attributes. Unfortunately, in many real-life datasets, the number of clusters is not always known a priori, especially for large data objects. More so, determining automatically the exact number of clusters that would provide the appropriate clustering analysis under this condition can be extremely challenging [15]. Therefore, the specific objective of this paper is to develop an improved FA based clustering method that would automatically provide the proper clustering partition without any prior knowledge of the characteristics of the dataset. Also, the study proposes the implementation of four hybrid FA algorithms to solve a wide range of clustering analyses problems automatically. The newly developed hybrid algorithms include firefly algorithm particle swarm optimization (FAPSO), firefly algorithm artificial bee colony (FAABC), firefly algorithm invasive weed optimization (FAIWO), and firefly algorithm teaching-learning-based optimization (FATLBO). For the improved FA algorithm, a mutation selection operator is incorporated into the standard FA algorithm to maintain the balance between selection pressure and population diversity of the algorithm. Two cluster analysis validity

indices, namely Davies–Bouldin (DB) [35] and Compact-Separated (CS) [36] are employed as a measure of determining the validity of clustering solutions. Experimental results on real-life datasets are illustrated to validate the superior performances of the proposed improved and hybrid FA algorithms over other existing clustering methods.

The outline of this paper is as follows. Section II presents a more detailed and comprehensive literature review on state-of-the-art clustering algorithms. Section III elaborates on the methodology of FA algorithmic design concept and the details of the proposed FA-based hybrid algorithms design for solving data clustering problems, afterword's some of the preliminary mathematical concepts relating to clustering analysis is discussed. Section IV describe a series of numerical and comparison experiments. Finally, concluding remarks and future research directions are provided in Section V.

II. RELATED WORK

Firefly algorithm due to its robustness, efficiency, ability to handle problems in different fields, including NP-hard, versatility, and other great benefits, has been successfully applied to solve problems in various domains. A comprehensive review of FA that discusses the diverse areas where the algorithm has been successfully used to a broad spectrum of real-world applications with satisfactory results was done by Fister *et al.* in 2014 [41]. In both works of literature, the authors went further to suggest future directions to the algorithm. FA although has been studied and traced to have good track records in diverse domains; however, its implementation in data clustering and automatic data clustering scopes is still very much shallow. Very few works have been done in the application of the firefly algorithm to data clustering, and quite a more difficult challenge in finding previous studies in its application to automatic data clustering.

A performance study on the firefly algorithm (FA) for data clustering was carried out by Senthilnath *et al.* in [26]. They acknowledged the strengths of FA and applied classification error percentage (CEP) to generate optimal cluster centroids. The standard FA was implemented for data clustering by focusing more on the attractiveness, light absorption, population size, and distance, and CEP was applied to check the method that generates the optimal number of clusters. Further, FA was compared with ABC, PSO, and nine other clustering methods. Results showed that the classification efficiency of FA compared to others is more superior in terms of reliability, efficiency, excellent global performance, and robustness.

Hassanzadeh and Meybodi presented a hybrid approach based on FA and k-means for data clustering [42]. The proposed model called K-FA was implemented such that, FA was used to find cluster centroids to a user-specified number of clusters, then the FA was extended using the k-means algorithm. The extension with the k-means algorithm was done in order to aid the refining of the cluster centroids detected by FA. Also, global optima were used to improve the standard FA. Experimental results showed that

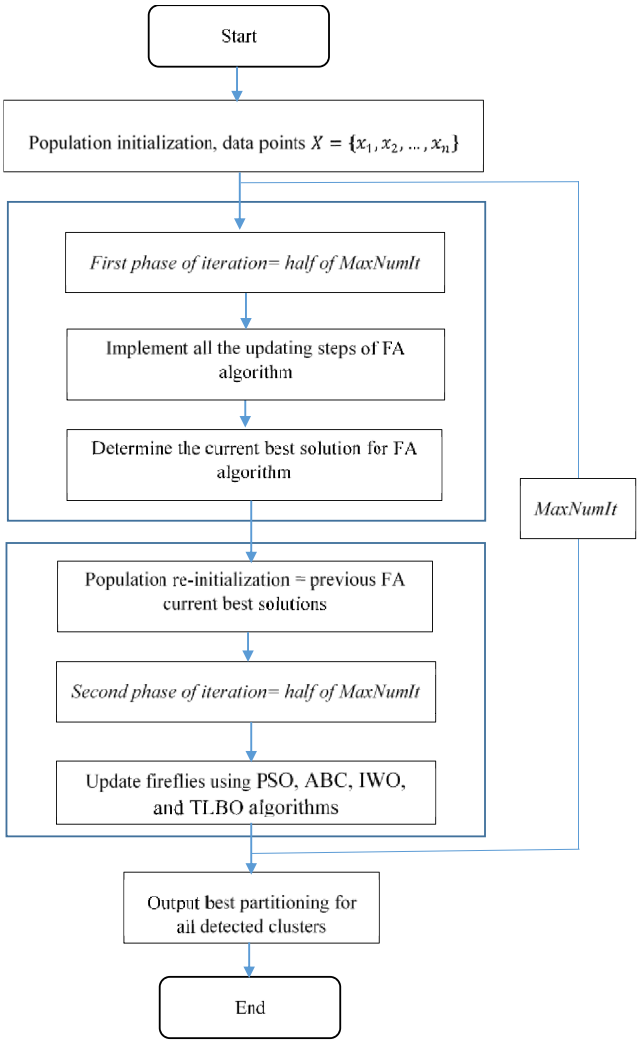


FIGURE 1. A compartmentalized flowchart of the hybrid firefly algorithms.

K-FA outperformed three other clustering algorithms in terms of better efficiency, and a decrease in intra-cluster distances which allowed the k-means method to have a proper initialization.

Banati and Bajaj conducted a viability performance analysis of FA for data clustering in [27]. The proposed method, called FClust, which is centroid-based, adopted the flashing behaviour of fireflies with the objective function of the clustering problem to obtain the optimal solution. The performance of FClust was evaluated using two statistical criteria, namely, trace within criteria (TWR) and variance ratio criteria (VRC) [43]. For simulation results comparison of FClust with standard PSO and DE showed that the FClust achieved the best mean fitness and standard deviation values on the VRC measure. Further, the quality of solutions obtained by FClust was also evaluated using the number of function evaluations via the run length distribution (RLD) approach [44]. RLD for FClust showed that it achieved the best function evaluation value and a faster convergence rate.

TABLE 1. A summary of metaheuristic algorithms that have been applied to automatic clustering problems.

Authors	Clustering method	Application area	Cluster validity index (CVI)
Omran et al. [53]	DCPSO	Cluster analysis	Dunn index (DI)
			Turi index
		Image segmentation	S_Dbw index
Masoud et al. [54]	CPSOII	Cluster analysis	Variance Ratio Criterion (VRC)
		Combinatorial optimization problem	Davies-Bouldin (DB) index
Ling et al. [55]	PLDC	Cluster analysis	Rand index (RI)
Kuo and Zulvia [56]	ACPSO	Cluster analysis	VI index
Satyasai and Ganapati [57]	MOIMPSO	Cluster analysis	-
		3D human models	
Das et al. [58]	MEPSO, Kernel_MEPSO	Cluster analysis	-
Kao and Chen [59]	PSOAC	Cluster analysis	-
		Cell formation	
Abubaker et al. [60]	MOPSOSA	Cluster analysis	DB index
			Symmetry (Symm) index
			Conn index
Das et al. [34]	ACDE	Cluster analysis	DB index
		Image segmentation	Compact-Separated (CS) index
Lee and Chen [61]	ACDE-O	Cluster analysis	*I* index
Saha et al. [62]	ADEFC	Cluster analysis	Xie-Beni (XB) index
Maulik & Saha [63]	MoDEAFC	Cluster analysis	XB index
		Image segmentation	
Suresh et al. [64]	MODE	Cluster analysis	XB index
			FCM index
			Rand index (RI)
			Silhouette index (SI)
Kundu et al. [65]	GADE	Cluster analysis	XB index
			FCM index
Zhong et al. [66]	AFCMDE	Cluster analysis	XB index
		Remote sensing	J_m index
Liu et al. [15]	AGCUK	Cluster analysis	DB index
			Calinski–Harabasz (CH) index

TABLE 1. ***(Continued.)*** **A summary of metaheuristic algorithms that have been applied to automatic clustering problems.**

He and Tan [67]	TGCA	Cluster analysis	Rand index (RI) Adjusted rand index (ARI)
Rahman and Islam [68]	GenClust	Cluster analysis	XB index
Karaboga et al. [69]	IDisABC	Cluster analysis	VI index Correct Classification Percentage (CCP)
Kuo et al. [70]	AKC-BCO	Cluster analysis Medicine (Prostate Cancer)	CS_{kernel} VI index
Kuo and Zulvia [71]	iABC	Cluster analysis Customer Segmentation	VI index
Das et al. [72]	ACBEA	Cluster analysis	CS index
Peng et al. [73]	Membrane system	Cluster analysis Membrane computing	CS index
Kumar et al. [74]	ACPAHS	Cluster analysis Image segmentation	Inter-intra cluster ratio Fitness function evaluation
Liu et al. [75]	DLSIAC	Cluster analysis Image segmentation	PMB index
Kumar et al. [76]	ACGSA	Cluster analysis Image segmentation	Inter-intra cluster ratio Fitness function evaluation
Kapoor et al. [77]	GWO	Cluster analysis Image segmentation	Inter-intra cluster ratio DB index
Anari et al. [78]	ACCALA	Cluster analysis Image segmentation	S_Dbw index
Zhou et al. [8]	SOS	Cluster analysis	Fitness function evaluation
Pacheco et al. [78]	Anthill	Cluster analysis	SI index
Elaziz et al. [79]	ASOCSA	Cluster analysis	Dunn index SI index DB index Calinski–Harabasz (CH) index
Agusti et al. [80]	GGA	Cluster analysis	DBI index
Salcedo-Sanz et al. [81]	GGA-based model	Cluster analysis	DBI index
Raposo et al. [82]	ACGA	Cluster analysis	Calinski–Harabasz (CH) index
Sharma and Chhabra [13]	HPSOM AHPSOM	Cluster analysis Mobile network data	Inter-cluster distance Intra-cluster distance Adjusted Rand Index (ARI) F-measure
Aliniya and Mirroshandel [83]	AC-ICA	Cluster analysis Face recognition	Purity Entropy RI ARI F-measure
Rajah and Ezugwu [91]	SOS	Cluster analysis	DB index and CS index

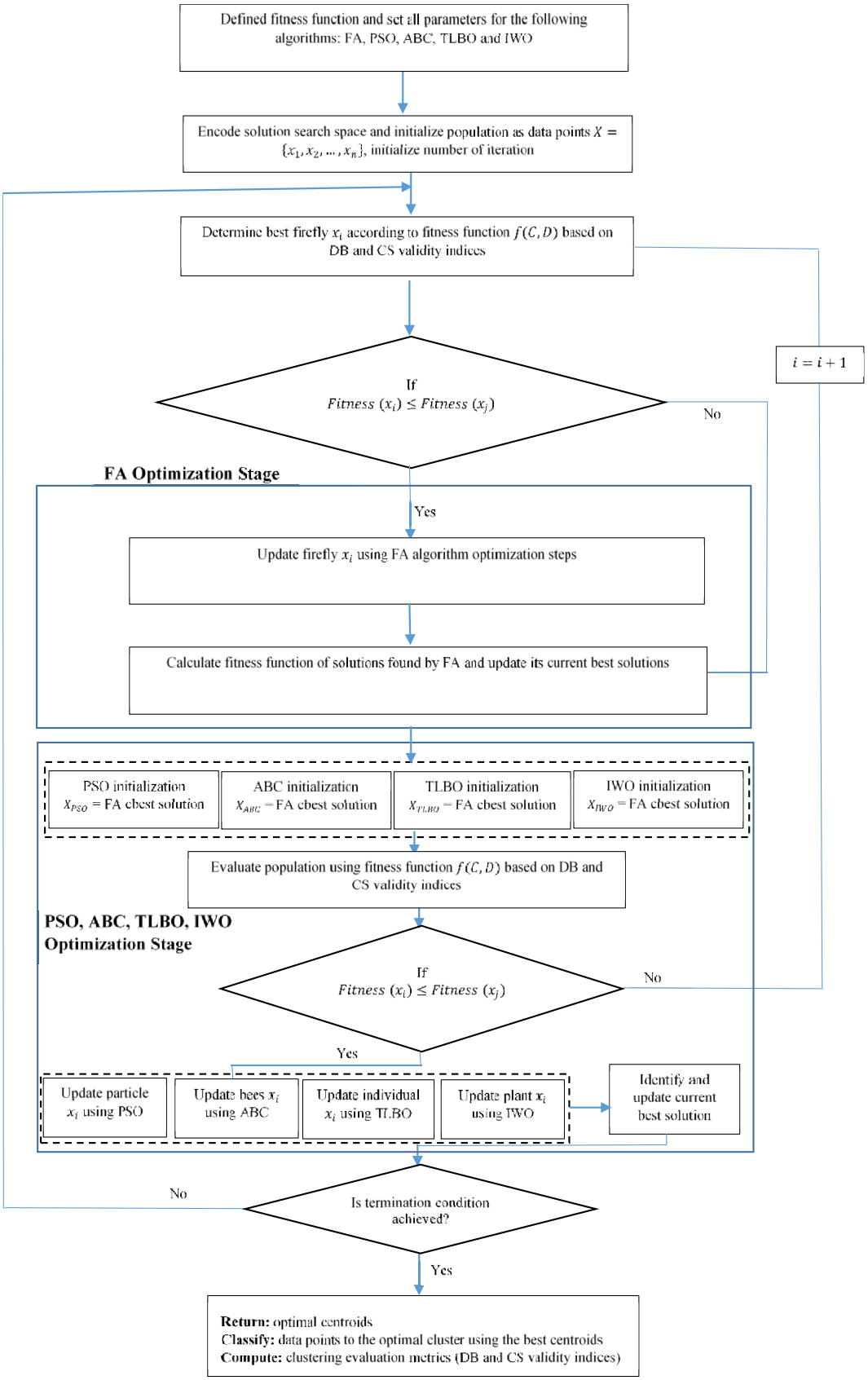


FIGURE 2. Flowchart of the hybrid firefly algorithms.

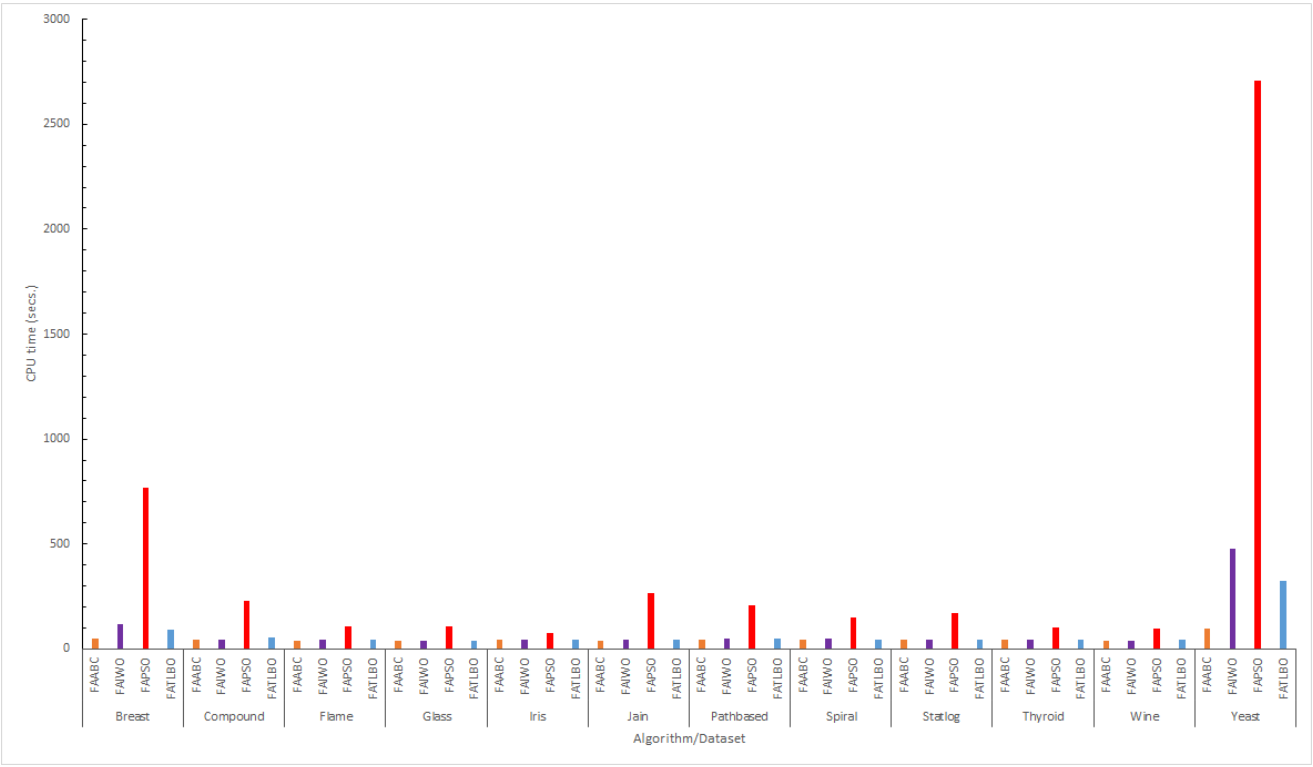


FIGURE 3. Average computational time consumed by the four hybrid firefly algorithms on CS measure for all the datasets for 40 replications.

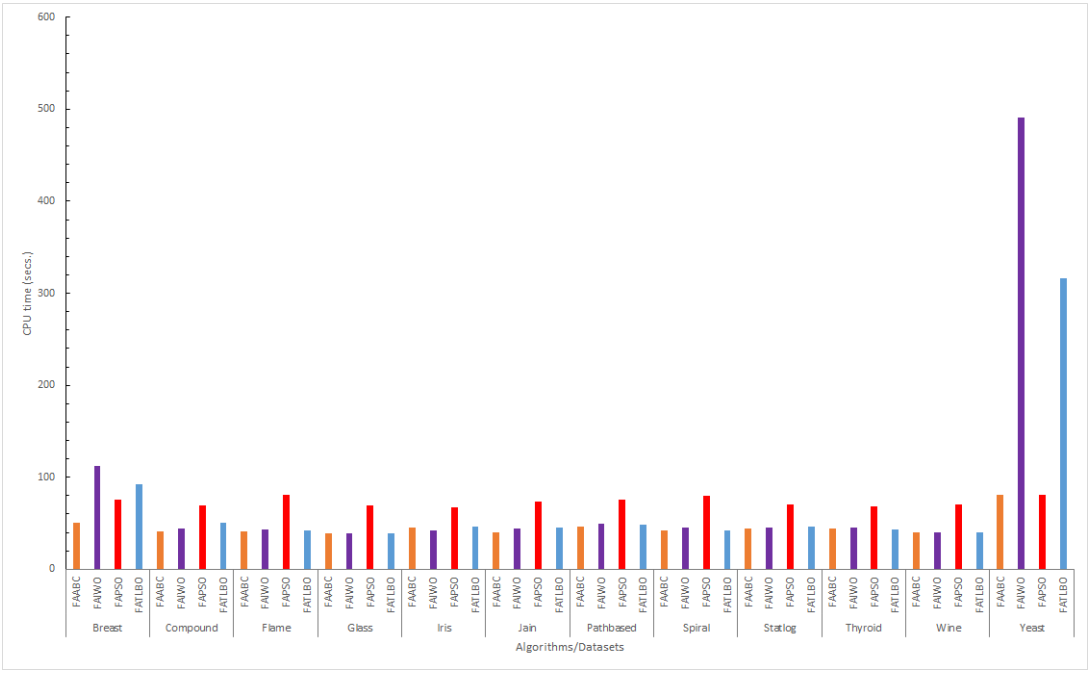
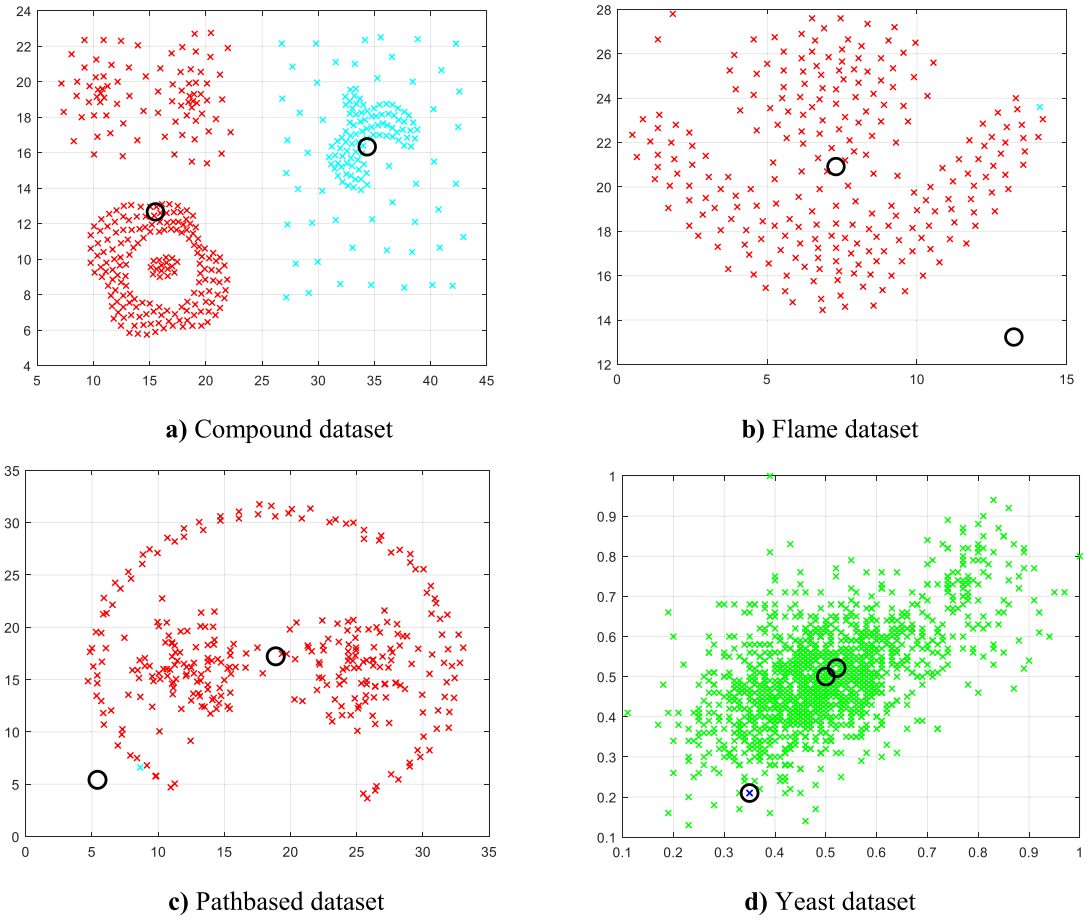


FIGURE 4. Average computational time consumed by the four hybrid firefly algorithms on DB measure for all the datasets for 40 replications.

In 2015, Kaushik and Arora integrated FA with an improved genetic algorithm [45], called FGA. The proposed model selects its initial population from a pool of population which is based on firefly algorithms, i.e. the initial population is generated from the global best solutions of the firefly algorithm. FAG operates in two ways. First, the classical FA is



a) Compound dataset

b) Flame dataset

c) Pathbased dataset

d) Yeast dataset

FIGURE 5. Clustering results of hybrid FAABC of some datasets on CS-index.

applied to sets of a randomly selected initial population which generates chromosomes of a set, and secondly, the chromosomes are then positioned in the mating pool from where they partake in the mutation and crossover operations of the genetic algorithm. Also, at the initialization stage of FGA, it results in global optimization, which prevents the solutions from getting trapped within the local optima. The test results, when compared to the basic genetic algorithm and firefly algorithm, showed that FGA had better inter-cluster and intra-cluster distances, and better satisfactory results.

Nayak *et al.* [47] implemented an improved FA with a fuzzy c-means algorithm called FAFCM and improved FAFCM for real-world clustering datasets. The improved FA addressed the shortfalls of the fuzzy c-means method, of local optima entrapment and high sensitivity to initialization. FAFCM was incorporated in two stages, firstly, a standard firefly algorithm with fuzzy c-means clustering, and secondly, an improved firefly algorithm with fuzzy c-means clustering. The first handled the limitations of the fuzzy c-means algorithm by minimizing the objective function. In contrast, the second phase refined the cluster centers that were identified from the first phase, and it also helped in further minimization of the objective function. FAFCM was compared with three other clustering algorithms, and the results showed that FAFCM had consistent results over the test datasets, a faster convergence speed, as well as a minimized objective function. However, the number of clusters was predefined before centroid assignment by FAFCM.

An efficient hybrid method based on a modified FA and a dynamic k-means algorithm for data clustering were developed by Sundararajan and Karthikeyan in [48]. The proposed algorithm is called a hybrid modified firefly and dynamic k-means algorithm. The dynamic k-means algorithm was incorporated so that it can adequately find the optimal number of clusters during execution time, as well as to improve the cluster quality and optimality. The model works in such a way that; it determines new centroids by adding one to the cluster counter in each iteration until the required cluster quality is attained since the model works well for a predefined number of clusters. Experimental results showed that the proposed model found better clusters quality in less time with increased optimality, against the compared algorithm.

Ezugwu [40] presented an extensive survey study of major nature-inspired metaheuristic algorithms that have been applied to solve automatic data clustering problems. Furthermore, the author carried out a comparative study

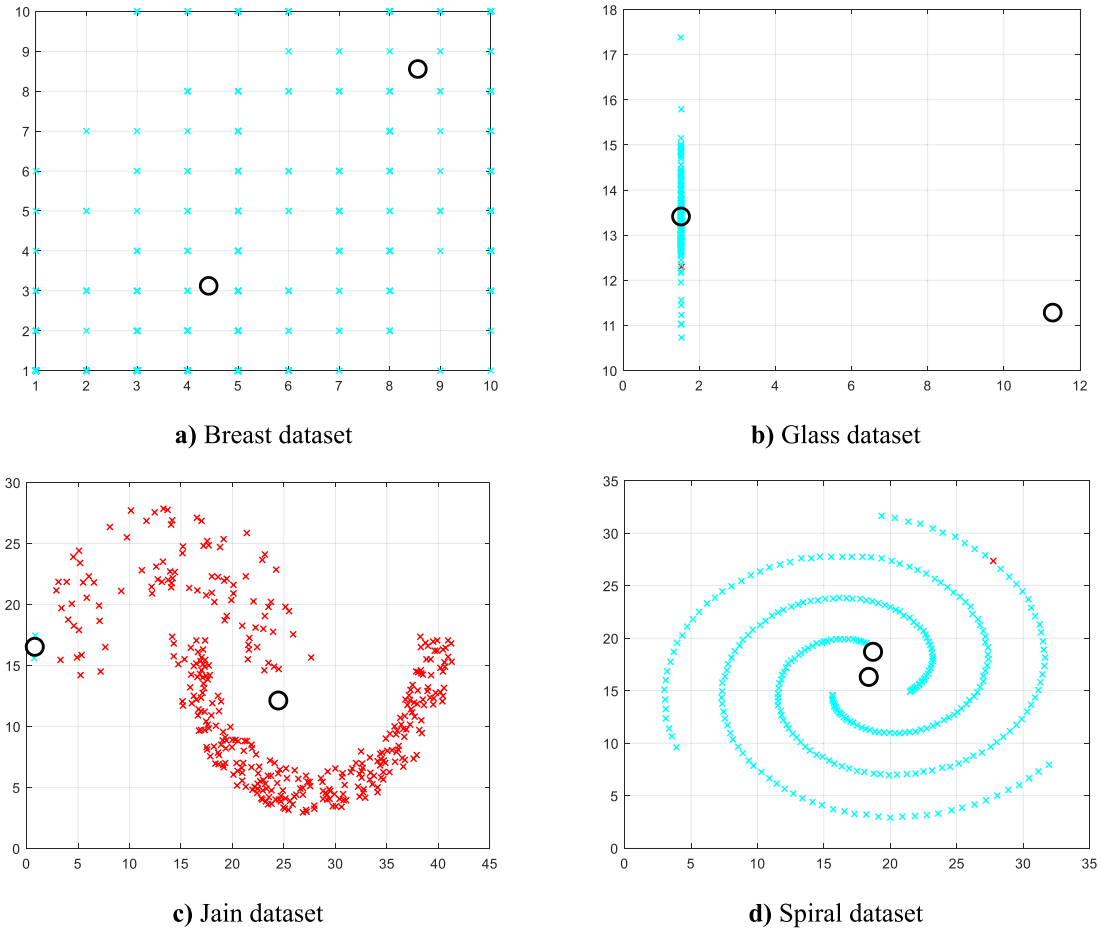


FIGURE 6. Clustering results of hybrid FAABC of some datasets on DB-index.

of several modified well-known global metaheuristic algorithms to solve automatic clustering problems, of which three hybrid swarm intelligence and evolutionary algorithms, namely, particle swarm differential evolution algorithm, firefly differential evolution algorithm and invasive weed optimization differential evolution algorithm, were employed to deal with the task of automatic clustering. The experimental results revealed that the firefly algorithm was more appropriate for better clustering of both low and high dimensional data objects than were other state-of-the-art algorithms.

All the different literature and comparative analyses results do point to the fact that the FA is a very efficient and robust metaheuristic algorithm for solving real-world problems. More so, the findings from Ezugwu [40] and Agbaje *et al.* [49] on the promising performance of the FA for automatic clustering compelled us to go into this research to investigate further the superior performances of both the improved nutation based firefly algorithm and its hybrid variants for automatic data clustering.

After extensive analysis that was carried out, we have compiled the following possible clustering methods, application areas, and clustering validity index types for the respective identified automatic metaheuristic techniques, which is presented in Table 1 above.

III. THE FIREFLY ALGORITHM

Firefly Algorithm is a nature-inspired optimization algorithm that was developed by Xin-She Yang in the late 2007 and early 2008 [29], [30]. The FA algorithmic design concept was inspired by the dynamic illumination of the light attribute from the fireflies, which are commonly found in most tropical and temperate regions. There are approximately 2000 species of fireflies, of which many of them produce short, rhythmic flashes of illuminations at regular intervals. The flashlights produced by these insects often act as communication signals that are used to entice other fireflies and also to send warnings to potential prey [31]. As a novel swarm intelligence population-based metaheuristic algorithm, FA has been used for solving different nonlinear engineering design optimization problems, as reported in [32]. Furthermore, studies have also shown that FA is very promising in terms of solving the most difficult NP-hard numerical optimization problems in both continuous and discrete spaces [33]. The mathematical modelling and representation of the standard FA algorithm are represented in equations (1) to (5). In equation (1),

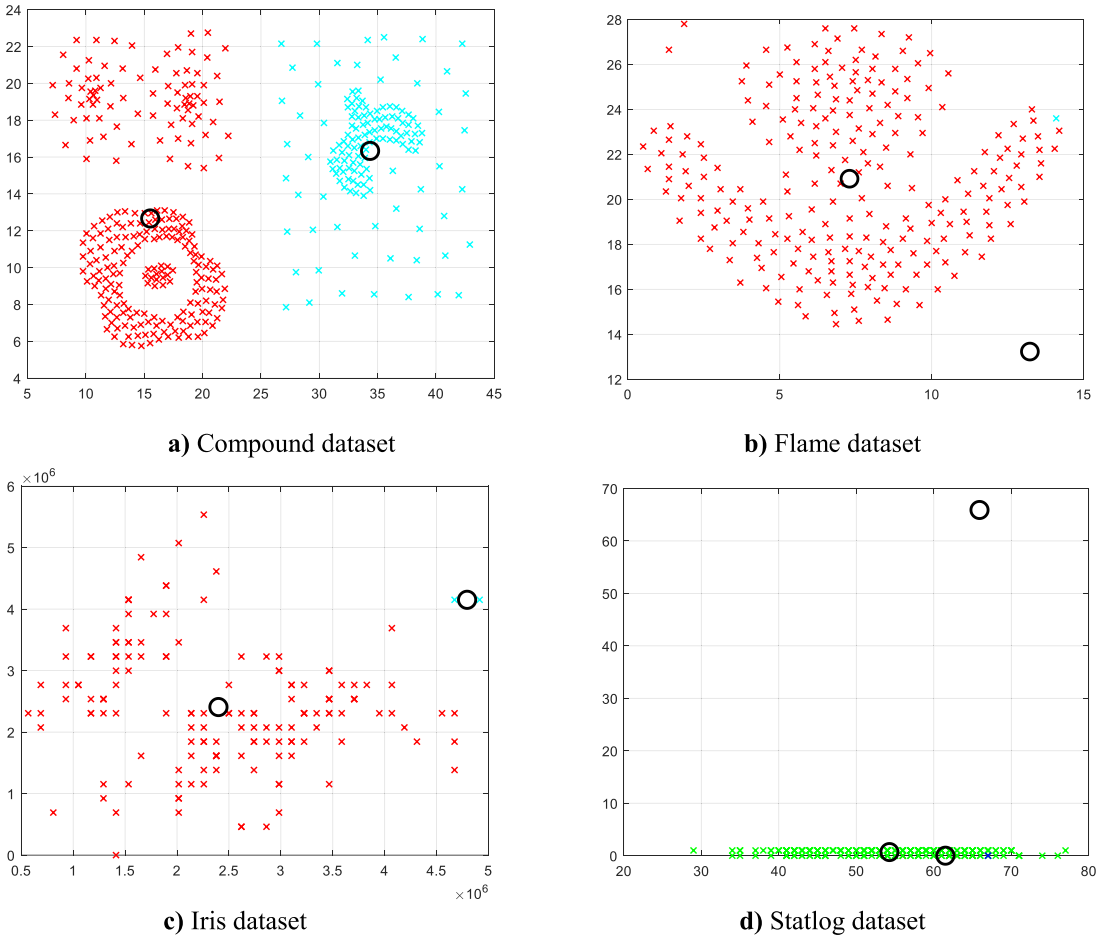


FIGURE 7. Clustering results of hybrid FAIWO of some datasets on CS-index.

the light intensity I of a firefly flashlight is said to be inversely proportional to the square of its distance denoted by r. This implies that the light intensity of the individual firefly diminishes with an increase in distance. However, this is because as the distance increases, the flashlight is released into the atmosphere [33].

$$I \propto 1/r^2 \tag{1}$$

Aligning the problem landscape to the FA algorithm design, the optimization model can be formulated in such a manner that the firefly flashlight is proportional to the fitness function value to be optimized. The following design principles were used to formulated basic FA [31]: it was assumed that all firefly species are identical in sex, the attractiveness of every firefly is directly proportional to the quality of its light intensity produced, the intensity of flashlight produced by any firefly is determined by the fitness function landscape that is to be optimized. In the FA algorithm design, light intensity and attractiveness are considered to play a vital role in the algorithm implementation and performance. Usually, in the case of maximization problems, the light intensity, produced at a specified point (y) is directly proportional to the fitness value of the fitness function, that is $I(y) \propto F(y)$. As shown in eq. (2), the light intensity changes with respect to distance and intensity of light emitted into the atmosphere.

$$I(r) = I_0 e^{-\gamma r^2} \tag{2}$$

where I_0 denotes initial light intensity at $r = 0$, γ is the light absorption coefficient, while r is the distance. From eq. (2), by combining the effect of the inverse square law and absorption, the singularity at $r = 0$ is circumvented in the expression $1/r^2$ [30], [33]. Based on eq. (3), the attractiveness of a firefly (β) is proportional to the light intensity of the firefly.

$$\beta = \beta_0 e^{-\gamma r^2} \tag{3}$$

where β_0 refers to the attractiveness at $r = 0$.

The distance measure between any two fireflies x_i and x_j is determined in terms of Euclidean distance:

$$r_{ij} = \| x_i - x_j \| = \sqrt{\sum_{k=1}^{d} \left(x_{i,k} - x_{j,k}\right)^2} \tag{4}$$

where d is the problem dimension. The movement of firefly from one point i to another point j is formulated as shown

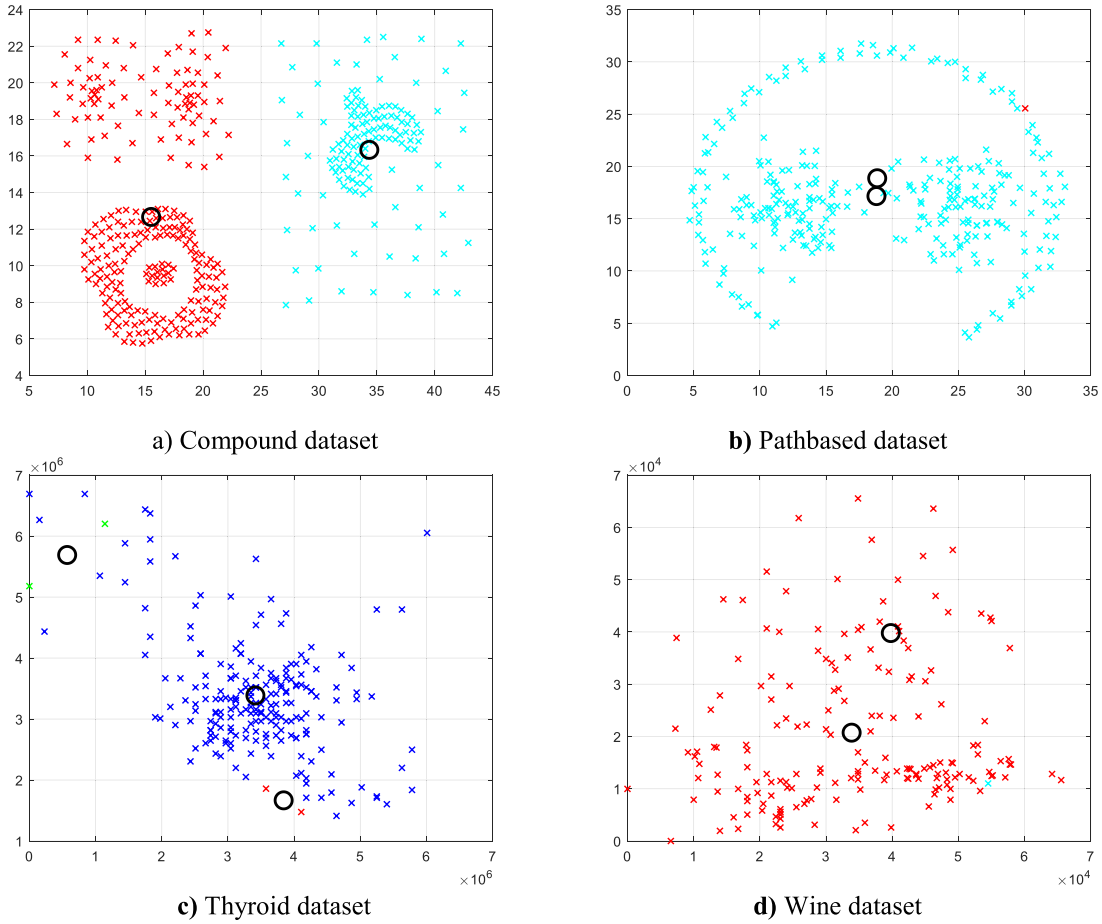


FIGURE 8. Clustering results of hybrid FAIWO of some datasets on DB-index.

in eq. (5):

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \alpha\epsilon_i \tag{5}$$

where $\alpha \in [0, 1]$, $\gamma \in [0, \infty)$. The parameter ϵ_i is a random number obtained from a Gaussian distribution. ϵ_i can be replaced with $rand - 0.5$, where $rand \in [0, 1]$. The third term ($\alpha\epsilon_i$) in eq. (5) shows firefly movement from one point to another, with regards to their attractiveness.

In this paper, to improve the exploration and exploitation capability of the FA, so that the algorithm can handle clustering tasks of high dimensionality more efficiently, the concept of mutation strategy is introduced into the FA searching process. Ideally, modified FA mutation strategy explores and exploits the search space by leveraging more desirable features from attractive fireflies and adding such functionality to enhance the attractiveness of the less bright fireflies. The extent of the enhancement feature modification that is required for any identified firefly with weak light brightness is determined by calculating the mutation probability (MP) of that firefly. Therefore, it is expected that those fireflies with excellent brightness will have lower MP, while those fireflies with low light intensity will have higher MP. In general, the concept of using MP is that there is a high probability of improving low-quality solutions and a low likelihood of reducing good quality solutions. The mutation operator probability used to introduce additional diversity among the firefly swarm is commutated as follows.

$$MP = f(x_{new}) - f(x_{old}) \tag{6}$$

where $f(x_{new})$ is the new firefly fitness and $f(x_{old})$ is the fitness of the first firefly. The main steps of the mutated FA are summarised as illustrated in Algorithm listing 1.

A. FIREFLY-BASED HYBRIDS AND CLUSTERING PROBLEM DESCRIPTION

The proposed hybridization methods described in this paper focuses on exploiting the various advantage of both the FA and other representative algorithms, namely, PSO, ABC, IWO, and TLBO algorithms. It is equally interesting to note that all the algorithms mentioned above work well for a wide range of global optimization problems. In this study, we propose a set of new hybrid firefly-based algorithms by combining some of the advantages of all the above mentioned individual algorithms. The proposed hybrid algorithms combine the attraction mechanism of FA with the effective fraternization capabilities of PSO, ABC, IWO, and TLBO to

Algorithm 1 Improved Firefly Algorithm

```
Define initial values of firefly parameters: β_o, α, n and γ
Define Fitness function f(x), x = (x_1, x_2 ... x_D)t
Initialize n positions of firefly (i = 1, 2, 3, ... .n)
Evaluate f(x) to determine light intensity L_i of firely x_i
while (t < MaxGeneration)
  for i = 1: n
    for j = 1:n
      if (L_i < L_j)
Move firefly i towards firefly j according to eq. (5);
      end
       Calculate MP = f(x_new) − f(x_old)
       Perform mutation ()
    end
    Calculate attractiveness variance with distance r using
exp(−γr);
      Calculate new fitness values for all fireflies;
      Accept new solution with best fitness;
  end
  Update firefly light intensity L_i;
  Update iteration counter t = t + 1;
  Reduce α by a factor;
end
```

maintain a good balance between exploration and exploitation of the problem search space. Also, the combination is done so as also to increase the solution accuracy, speed of convergence and the diversity of the population. We implemented four hybrid algorithms, namely, FAPSO, FAABC, FAIWO, and FATLBO, to solve data clustering problems. It is noteworthy to mention that the improved FA and other four metaheuristics are executed in parallel to specifically promote information sharing among the swarm population and thus enhance searching efficiency [37].

The implementation strategy employed by the four new hybrid algorithms begins its search process by using FA as the global optimization search algorithm, because of its strong exploration ability and then subsequently introducing the other four single algorithms separately and then using them as a local search optimization algorithm to enhance the intensification capability of the new hybrid methods. The local search mechanism is suggestively important in the design of the new hybrid algorithm, especially when the search process descends the paths of the local optimal solutions, it will prevent the algorithms from entrapment into local minima. Therefore, the advantage, as mentioned above, is leveraged to improve both the exploitation and exploration ability of the proposed FA-based hybrid algorithms. Furthermore, one of the main enhancement quality of such hybridization and regrouping mechanism of the new algorithms is to ensure that the search for candidate solutions is concentrated only on the promising region of the solution search spaces. This mechanism is significant, as it aids the proposed method not to search for a candidate solution within less promising regions of the search space. A similar technique was implemented in [37], where FA was combined with the differential evolution algorithm.

The effectiveness and efficiency of the proposed FA-based hybrid methods are evaluated using the CS and DB validity indices discussed in section III of this paper. These two validity indices also help to determine the appropriate optimal number of clusters and find the best partitioning for the detected clusters. For the first phase of the hybrid algorithm implementation, the FA-based hybrid algorithms start their search optimization processes with the generation of initialization population of fireflies. After that, the fitness function of each candidate solution found by the FA is computed and determined using the two clustering validity measures. Iteratively, these new solutions with the best fitness values are updated using the operators of FA. In the second optimization phase, the same process is iteratively repeated using now the operators of PSO, ABC, IWO, and TLBO algorithms, respectively to re-optimize the solutions obtained in the first phase. Note that the two phases of optimization techniques form the first cycle of the evaluation phase for FAPSO, FAABC, FAIWO, and FATLBO implementation. It is interesting to mention here that the four FA-based hybrids use the best solution generated by the FA search results in the first phase as its initial search population. As for the evaluation process, the previous local best and global best within the new population are compared, and the candidate solution with the best fitness values is updated accordingly. As stated earlier, the CS and DB indices are used by the four methods to compute the final fitness function of each solution, which the FA-based hybrids use to determine the best candidate solution and make the necessary updates. Finally, the best solution is determined based on which solution has the smallest CS-index value or DB-index value. The entire process of the FA-based hybrid algorithms is repeated until the termination criteria are reached. The Algorithm listing 2 shows the steps mentioned above for the FA-hybrids algorithms. Figure 1 illustrates the compartmentalized flowchart of the proposed method, while Figure 2 illustrates the implementation flowchart of the generalized hybrid methods. In general, the figure also represents the clustering processes of the four hybrid algorithms implementations. However, it is noteworthy to mention that part of the main contribution of the current paper is the proposal of a critical performance study and evaluation of several hybrid firefly algorithms for the task of automatic clustering. No record of a similar research focus in the literature exist as of the time of writing this paper.

As aforementioned earlier, the hybrid algorithm implementation methods comprise of two stages. The first stage engages the modified FA algorithm by randomly generating initial swarm, where the number of fireflies equal to the number of clusters and the swarm population is uniformly distributed across the dimension of the dataset, which in this case is the clustering problem search space. After the swarm initialization, the next task is the evaluation of the best swarm according to the fitness function determined by the DB and

Algorithm 2 Pseudocode for the Four FA-Based Hybrid Algorithms

```
Input: Data points X = {x1, x1, ..., xn}
Output: Optimal cluster centres D = {d1, d1, ..., dC}
Begin
Generate initialize population with K random cluster centres
Determine the objective function using CS and DB validity indices
For i = 1 to n
        Evaluate cost function using Euclidean distance metrics to get the best individual
            If current value of population (i) .Cost <= BestSolution.Cost
              Update the current population (i) as the best solution;
            End If
        End For
    While maximum iteration is not reached do
        For i = 1 to n
          For j = 1 to n
            If pop (j) < population (i) .Cost
            Move population (i) towards population (j) using FA operators
            If newsolution.Cost <= newpopulation (i) .Cost
               newpopulation (i) becomes the new solution;
                 If newpopulation (i) .Cost <= bestSolution.Cost
                Update newpopulation (i) as the new solution;
                 End If
            End If
           End If
        Apply PSO updating formula (see [40]) on the current newpopulation (i)
        Apply ABC updating formula (see [50]) on the current newpopulation (i)
        Apply IWO updating formula (see [40]) on the current newpopulation (i)
        Apply TLBO updating formula (see [51]) on the current newpopulation (i)
        Update the global best solution in the whole population
        Evaluate the fitness value of each individual candidate solution
        Update the new value as the global best
      End For
    End For
  End While
End
```

CS validity indices [40]. Note that the best swarm position, for example, represents the data point that achieves the minimum distance to the swarm from its previous searches. The PSO, ABC, IWO, and TLBO operate on the new set of the solution generated by the FA updating equation given in (5). The parameters of the respective logarithms are used to determine next movement patterns of their optimization strategies as also explained earlier. Iteratively the various position of the new populations is updated until the case of a satisfactory termination condition is met, and the algorithm simulation process is terminated.

B. CLUSTERING PROBLEM DESCRIPTION

In this performance study, we propose a series of hybrid firefly algorithm to solve automatic data clustering problems. As described in [34] to handle automatic data clustering problems, we adopt the same approach for the implementation of the variants of the hybrid firefly algorithms. Given that a set of dataset F is defined as $F = \{f_1, f_2, \ldots, f_n\}$ which is divided into non-overlapping groups of cluster $G = \{g_1, g_2, \ldots, g_n\}$, such that the dimension $w_i (i = 1, 2, \ldots, n)$ is p. For each of the cluster $G = \{g_1, g_2, \ldots, g_n\}$, there is a centroid $d_i = (i = 1, 2, \ldots, C)$ represented for each of the clusters, that is, $D = (d_1, d_2, \ldots, d_C)$ are the centres of $G = \{g_1, g_2, \ldots, g_C\}$. For a p-dimensional data vector, the following conditions must take place:

$$V_i \cap V_j = \emptyset \quad \text{where } i, j = 1, 2, \ldots, C \text{ and } i \neq j \tag{7}$$

$$V_1 \cup V_2 \cup \ldots \cup V_C = F \tag{8}$$

$$V_i \subseteq F \quad \text{and } V_i \neq 0,\ i = 1, 2, \ldots, C \tag{9}$$

At the initialization phase of each of the hybrid algorithms, the population (swarm) size K is defined as $W = (w_1, w_2, \ldots, w_K)$. As described above, let each member a_i in the population be a $Q \times p$-dimensional vector, $F_{n \times p}$, which is defined as $W_i = w_1^*, w_2^*, \ldots, w_q^*$ $(w_{11}, w_{12}, \ldots, w_{1p}), (w_{21}, w_{22}, \ldots, w_{2p}), \ldots, (w_{Q1}, w_{Q2}, \ldots, w_{Qp})$. The main goal of the optimization method over

TABLE 2. (a) Parameter configurations of ABC, IWO, PSO, and TLBO algorithms. (b) Characteristics of the twelve benchmark datasets.

ABC		IWO		PSO		TLBO	
parameter	value	parameter	value	parameter	value	parameter	Value
Population size	25	Population size	25	Population size	25	Population size	25
$MaxIt$	200	$MaxIt$	200	$MaxIt$	200	$MaxIt$	200
a	0.009	m	2	c_1	1.5	m	2
# $nOnlooker$ bees	25	S_{min}	0	c_1	2.0		
m	2	S_{max}	5	m	2		
		E	2	$wdamp$	0.99		
		$sigma\_initial$	0.5	w	0.8		
		$sigma\_final$	0.001				

(a)

Datasets	Type of dataset	Number of data points (N)	Dimension of dataset (D)	Number of clusters (k)
Breast	UCI dataset	699	10	2
Compound	Shape set	399	2	6
Flame	Shape set	240	2	2
Glass	UCI dataset	214	10	7
Iris	UCI dataset	150	4	3
Jain	Shape set	373	2	2
Pathbased	Shape set	300	2	3
Spiral	Shape set	312	2	3
Statlog	UCI dataset	270	13	2
Thyroid	UCI dataset	215	5	2
Wine	UCI dataset	178	13	3
Yeast	UCI dataset	1,484	8	10

(b)

the four proposed hybrids of the firefly algorithm in this study is minimization, where we employed the two common and most used cluster validity indices namely, CS and DB indices, to minimize the sum of the distances between the datasets $f_i(i = 1, 2, \ldots, n)$ and centers $d_i(i = 1, 2, \ldots, C)$. The upper and lower boundaries of the number of groups in the population are respectively defined as, Var_{min} represented as $k_j^* = min\{F_1, F_2, \ldots, F_p\}$ and Var_{max} denoted as $m_j^* = max\{F_1, F_2, \ldots, F_p\}$. In general, the lower boundary is $k = (k_1^*, k_2^*, \ldots, k_C^*)$ and the upper boundary is $m = (m_1^*, m_2^*, \ldots, m_C^*)$, for the solution space. To solve the automatic clustering problem, the ith particle W_i is evaluated as follows:

$$W_i = rand\ (1, Q \times p)^*.\ (m - k) + k \tag{10}$$

where $rand\ (1, Q \times p)$ is a vector of a uniformly distributed random number which returns an integer between 0 and 1.

C. CLUSTERING VALIDITY INDEX

In this section, we discuss the two validity indices that are used across the study to measure and analyze the effectiveness of the four proposed hybrids of the firefly algorithm, as well as the quality of the clustering solution obtained. Generally, a good cluster validity index offers two significant purposes; firstly, it helps to determine the number of clusters and, secondly, it determines the best (optimal) partition [35]. Likewise, a good cluster validity index is expected to handle two key areas of portioning namely cohesion and separation. *Cohesion*: in this case simply means that the objects or data points in a cluster should be compact and identical (similar) and as possible. A deviation in the fitness variance of the objects in a cluster indicates good compactness of such a cluster. On the other hand, *separation* in contrast to cluster compactness should be different and distinct to each other. This step can be, however, seen in the distance among cluster centers, which indicates the cluster separation. Davis and Bouldin [36] further stated that a clustering validity index should as well exhibit the following properties:

1. Ability to involve minimal or no human interference or parameter specification during its operation.
2. Ability to be scalable computational-wise for large datasets.
3. Ability to produce accurate results for datasets with arbitrary dimensions.

TABLE 3. Numerical results comparison of average solutions obtained by muted FA and the four hybrid algorithms based on the CS and DB indices over 40 replications.

Dataset/Algorithm	CS index					DB index				
	FA	FAABC	FAIWO	FAPSO	FATLBO	FA	FAABC	FAIWO	FAPSO	FATLBO
Breast	**0.6423**	0.6757	0.7860	0.8513	0.6879	**0.6519**	0.7035	0.7889	0.6808	0.7158
Compound	0.6976	0.6374	0.7227	**0.5827**	0.5731	**0.4932**	0.7047	0.7025	0.5024	0.6035
Flame	**0.3880**	**0.3880**	0.4002	0.4488	0.3874	0.7229	0.3913	0.3991	0.6427	**0.3870**
Glass	0.0608	0.0608	0.0608	0.0608	0.0608	0.6091	0.0608	0.0608	**0.5140**	0.0608
Iris	0.5582	0.5515	0.6949	**0.4879**	0.5351	0.5700	0.5475	0.6965	0.5725	**0.5408**
Jain	0.6537	0.6508	0.6470	**0.5232**	0.6546	0.6490	0.6518	0.6537	**0.6442**	0.6515
Pathbased	0.5801	0.6508	0.7038	**0.5524**	0.5750	0.6683	0.5816	0.7269	0.6416	**0.5571**
Spiral	0.6816	0.6638	0.7478	**0.6340**	0.6752	0.7480	0.6833	0.7337	0.7373	**0.6795**
Statlog	0.2945	0.2937	0.2937	**0.2937**	0.2937	**0.2039**	0.2945	0.2937	0.2040	0.2937
Thyroid	0.5271	0.5301	0.5611	**0.4271**	0.5267	0.4937	0.5297	0.5532	**0.4896**	0.5238
Wine	0.8426	0.8593	0.9121	**0.6429**	0.8274	0.8050	0.8712	0.9154	**0.7981**	0.8385
Yeast	0.4770	0.4631	0.4323	**0.3512**	0.4373	0.6026	0.4759	0.4260	0.5888	**0.4343**
Average	0.5336	0.5354	0.5802	**0.4880**	0.5195	0.6015	0.5413	0.5792	0.5847	**0.5239**

For a crisp or hard clustering, some of the most used and well-known validity indices are CS index [35] and DB index [36], which were also used in this study as aforementioned. For most of the validity indices, they are considered as either minimization or maximization optimization technique by default. Similarly, their implementation outputs demonstrate a good clustering partition. As a result of their optimizing strategy, the clustering validity indices are best adopted with optimization algorithms such as the PSO, DE, GA, etc. In this study, we define the cluster validity index as a function J, such that a given clustering B, and a similarity measure V it is defined as $J(B, V)$. The function $J(B, V)$ returns a real number which indicates the cluster validity index or the fitness of the clustering task B. The two validity indices adopted for our study are further discussed in the next section.

1) COMPACT-SEPARATED INDEX

This cluster validity measure estimates the ratio of the sum of within-cluster scatter to between-cluster separation, which is similar to how the DB index operates. It has been studied that the CS index offers more efficiency in handling clusters having different dimensions, densities or sizes. Although, it is computationally more intensive than the DB index in terms of execution time, however, it does produce more good quality solutions. Furthermore, a large value of a CS index indicates weak compactness or separation, while a lesser value means a good and better clustering. Let the within-cluster scatter be denoted as Y_i and the between-cluster separation be represented as Y_j, such that the distance measure V is given as $V\left(Y_i, Y_j\right)$. Hence, the CS index for a clustering B is computed as given in equation 11.

$$J_{CS}\left(B, V\right) = \frac{\frac{1}{K}\sum_{i=1}^{K}\left[\frac{1}{B_i}\sum_{Y_i \in B_i} max_{Y_j \in B_i}\{V(Y_i, Y_j)\}\right]}{\frac{1}{K}\sum_{i=1}^{K}\left[min_{j \in K, j \neq i}\left\{V\left(x_i, x_j\right\}\right]\right.}$$
$$J_{CS}\left(B, V\right) = \frac{\sum_{i=1}^{K}\left[\frac{1}{B_i}\sum_{Y_i \in B_i} max_{Y_j \in B_i}\{V\left(Y_i, Y_j\right)\}\right]}{\sum_{i=1}^{K}\left[min_{j \in K, j \neq i}\left\{V\left(x_i, x_j\right\}\right]\right.} \quad (11)$$

where K is the number of clusters in B.

2) DAVIS-BOULDIN INDEX

The DB index estimates the quality of clustering by evaluating the intra-cluster (average distances of all data points within a cluster from the centroid) to inter-cluster (the distance between two centroids) distances. Likewise, for DB index, the smaller the index value, the better the compactness or separation, and otherwise for a large value. Let W_i be defined as the average distance of all the data points within a cluster B_i to their centroids x_i. The average distance is calculated as:

$$W_i = \left[\frac{1}{B_i}\sum\nolimits_{X \in B_i} V(R, x_i)^t\right]^{\frac{1}{t}} \quad (12)$$

where $V(R, x_i)$ is the distance between a data point R in B_i and its centroid x_i, and $t \geq 1$ is an integer that can be selected independently. If $t = 1$, W_i equates to the average Euclidean distance of the vectors within the cluster. On the other hand, if $t = 2$, W_i equates to the standard deviation of the distances of objects within a cluster to their respective

TABLE 4. Numerical results for the four hybrid firefly algorithms based on the CS and DB indices on over 40 replications.

Dataset	Algorithm	CS index				DB index			
		Best	**Worst**	**Average**	**StDev.**	**Best**	**Worst**	**Average**	**StDev.**
Breast	FAABC	**0.5996**	1.0352	0.6757	0.1125	**0.5996**	1.0718	0.7035	0.1551
	FAIWO	**0.5996**	1.0715	0.7860	0.1947	**0.5996**	1.0715	0.7889	0.1886
	FAPSO	0.6025	1.0795	0.8513	0.1402	0.6519	1.0881	0.6808	0.1032
	FATLBO	**0.5996**	1.0352	0.6879	0.1535	**0.5996**	1.0352	0.7158	0.1826
Compound	FAABC	**0.4962**	0.7732	0.6374	0.1280	0.4962	0.7732	0.7047	0.1122
	FAIWO	0.5032	0.7732	0.7227	0.0858	0.5032	0.7732	0.7025	0.1070
	FAPSO	0.5032	0.7732	0.5827	0.0558	**0.4932**	0.6770	0.5024	0.0365
	FATLBO	**0.4962**	0.7732	0.5731	0.1007	0.5032	0.7732	0.6035	0.1066
Flame	FAABC	0.3846	0.4079	0.3880	0.0064	**0.3846**	0.4079	0.3913	0.0091
	FAIWO	0.3846	0.7142	0.4002	0.0592	**0.3846**	0.7142	0.3991	0.0519
	FAPSO	**0.3683**	0.5366	0.4488	0.0350	0.5969	0.7073	0.6427	0.0405
	FATLBO	0.3846	0.3948	0.3874	0.0043	**0.3846**	0.3948	0.3870	0.0041
Glass	FAABC	**0.0608**	0.0608	0.0608	0.0000	0.0608	0.0608	0.0608	0.0000
	FAIWO	**0.0608**	0.0608	0.0608	0.0000	0.0608	0.0608	0.0608	0.0000
	FAPSO	**0.0608**	0.0608	0.0608	0.0000	**0.3336**	0.6136	0.5140	0.1206
	FATLBO	**0.0608**	0.0608	0.0608	0.0000	0.0608	0.0608	0.0608	0.0000
Iris	FAABC	0.5311	0.7396	0.5515	0.0507	**0.5311**	0.7396	0.5475	0.0430
	FAIWO	0.5725	0.8040	0.6949	0.0595	0.5577	0.8040	0.6965	0.0668
	FAPSO	**0.4260**	0.5999	0.4879	0.0478	0.5689	0.6640	0.5725	0.0149
	FATLBO	0.5311	0.5577	0.5351	0.0048	**0.5311**	0.7191	0.5408	0.0300
Jain	FAABC	0.5647	0.6546	0.6508	0.0172	0.5393	0.6546	0.6518	0.0182
	FAIWO	0.5393	0.7012	0.6470	0.0293	0.7687	0.5307	0.6537	0.0324
	FAPSO	**0.4673**	0.5867	0.5232	0.0288	0.6514	0.6326	0.6442	0.0062
	FATLBO	0.6546	0.6546	0.6546	0.0000	**0.5307**	0.6546	0.6515	0.0196
Pathbased	FAABC	0.5647	0.6546	0.6508	0.0172	0.5000	0.7888	0.5816	0.0871
	FAIWO	0.5000	0.8789	0.7038	0.0858	0.5000	0.9048	0.7269	0.1073
	FAPSO	**0.4988**	0.6360	0.5524	0.0350	0.6238	0.6695	0.6416	0.0167
	FATLBO	**0.4988**	0.6548	0.5750	0.0733	**0.4988**	0.6548	0.5571	0.0729
Spiral	FAABC	**0.5277**	0.7910	0.6638	0.0726	**0.5287**	0.7910	0.6833	0.0621
	FAIWO	0.5442	0.9291	0.7478	0.0876	0.5635	0.8023	0.7337	0.0548
	FAPSO	0.5567	0.7226	0.6340	0.0367	0.7270	0.7988	0.7373	0.0182
	FATLBO	0.5357	0.6862	0.6752	0.0363	0.5383	0.6862	0.6795	0.0296
Statlog	FAABC	**0.2937**	0.2937	0.2937	0.0000	0.2937	0.3280	0.2945	0.0054
	FAIWO	**0.2937**	0.2937	0.2937	0.0000	0.2937	0.2937	0.2937	0.0000
	FAPSO	**0.2937**	0.2937	0.2937	0.0000	**0.2038**	0.2052	0.2040	0.0003
	FATLBO	**0.2937**	0.2937	0.2937	0.0000	0.2937	0.2937	0.2937	0.0000
Thyroid	FAABC	0.5238	0.6409	0.5301	0.0259	0.5238	0.6409	0.5297	0.0258
	FAIWO	0.5238	0.6409	0.5611	0.0431	0.5238	0.6409	0.5532	0.0343
	FAPSO	**0.4271**	0.4271	0.4271	0.0000	**0.4813**	0.5027	0.4896	0.0097
	FATLBO	0.5238	0.6409	0.5267	0.0185	0.5238	0.5238	0.5238	0.0000
Wine	FAABC	0.6570	0.8829	0.8593	0.0628	**0.6570**	0.8829	0.8712	0.0407
	FAIWO	0.8829	1.0240	0.9121	0.0425	0.8829	1.0301	0.9154	0.0486

TABLE 4. *(Continued.)* **Numerical results for the four hybrid firefly algorithms based on the CS and DB indices on over 40 replications.**

	FAPSO	**0.5085**	0.7940	0.6429	0.0724	0.8850	0.5639	0.7981	0.0408
	FATLBO	0.6570	0.8829	0.8274	0.0818	**0.6570**	0.8829	0.8385	0.0801
Yeast	FAABC	0.3897	0.5205	0.4631	0.0496	**0.3897**	0.5205	0.4759	0.0520
	FAIWO	0.3897	0.5205	0.4323	0.0398	**0.3897**	0.5205	0.4260	0.0441
	FAPSO	**0.3120**	0.4169	0.3512	0.0290	0.4379	0.7770	0.5888	0.0978
	FATLBO	0.3897	0.5205	0.4373	0.0498	**0.3897**	0.5205	0.4343	0.0530

TABLE 5. **Mean ranks achieved by the Friedman test for the four proposed hybrid firefly algorithms.**

Dataset	CS-Index				DB-Index			
	FAABC	FAIWO	FAPSO	FATLBO	FAABC	FAIWO	FAPSO	FATLBO
Breast	2.14	2.70	3.26	**1.90**	2.41	3.03	2.53	**2.04**
Compound	2.55	3.38	2.18	**1.90**	3.29	3.19	**1.10**	2.43
Flame	2.08	2.10	3.78	**2.05**	2.18	2.06	3.98	**1.79**
Glass	**2.50**	**2.50**	**2.50**	**2.50**	**2.00**	**2.00**	4.00	**2.00**
Iris	2.61	3.95	**1.19**	2.25	1.70	3.90	2.93	**1.48**
Jain	2.99	2.95	**1.00**	3.06	2.96	2.98	**1.10**	2.96
Pathbased	2.30	3.73	**1.93**	2.05	2.04	3.64	2.60	**1.73**
Spiral	2.35	3.60	**1.48**	2.58	1.99	2.91	3.48	**1.63**
Statlog	**2.50**	**2.50**	**2.50**	**2.50**	3.03	2.99	**1.00**	2.99
Thyroid	2.75	3.58	**1.00**	2.68	2.71	3.66	**1.00**	2.63
Wine	2.75	3.63	**1.15**	2.48	2.76	3.55	**1.33**	2.36
Yeast	3.30	2.90	**1.10**	2.70	2.59	**1.69**	3.85	1.88

centroids. Taking H_{ij} to represent the inter-cluster distance between two centroids x_i and x_j, we have that,

$$H_{ij} = V\left(x_i, x_j\right), \quad i \neq j \tag{13}$$

Let V_i be defined as

$$V_i = \max\left\{\frac{W_i + W_j}{H_{ij}} | 1 \leq i, j \leq K, i \neq j\right\} \tag{14}$$

Thus, the DB index is expressed as:

$$J_{DB}\left(B, V\right) = \frac{1}{K}\sum\nolimits_{K}^{1} V_i \tag{15}$$

where K is the number of clusters.

In summary, it is important to note that the data clustering problem described in this paper is modelled as an optimization problem. For example, given an instance of data points with x attributes and a predetermined number of clusters g, the objective function aims to determine an optimal cluster setting such that the sum of squared Euclidean distances between each data object and the center of the belonging cluster is minimized. Therefore, by so doing, each data point should belong to a unique cluster, and no cluster must be left empty.

IV. SIMULATION EXPERIMENTS

Experiments were carried out using a 3.60 GHz Intel(R) Core(TM) i7-7700 processor and 16 GB memory on Windows 10 operating system. The entire algorithm was programmed in MATLAB R2018b and statistical analysis conducted using IBM SPSS Statistics Version 25.

A. PARAMETER SETTING

In this section, we present the settings of the control parameters for the respective four firefly-based algorithms that are studied in this paper. The control parameter settings are described in Table 2(a). For each of the proposed algorithm, we initialize an equal number of populations and number of iterations, as well as the same number of replications, which in this case is 40 runs for all our experiments. The FA being the control algorithm has the following parameter settings: The population size is set as 25, a maximum number of iteration *MaxIt* is set as 200, light absorption coefficient γ is set as 1, attraction coefficient β is set as 2, mutation coefficient m is set as 2, and finally the mutation coefficient damping ratio α is set as 1. The parameter configurations of ABC, IWO, PSO, and TLBO are further detailed in Table 2a.

TABLE 6. *p*-values produced by the Wilcoxon rank-sum test for equal medians on CS index.

Dataset	FAPSO vs FAABC	FAPSO vs FAIWO	FAPSO vs FATLBO	FAABC vs FAIWO	FAABC vs FATLBO	FAIWO vs FATLBO
Breast	0	0.109	0	0.02	0.683	0.012
Compound	0.009	0	0.683	0.001	0.007	0
Flame	0	0	0	0.71	0.553	0.626
Glass	1	1	1	1	1	1
Iris	0	0	0	0	0.008	0
Jain	0	0	0	0.398	0.18	0.08
Pathbased	0.018	0	0.116	0	0.315	0
Spiral	0.033	0	0	0	0.355	0
Statlog	1	1	1	1	1	1
Thyroid	0	0	0	0	0.18	0
Wine	0	0	0	0	0.034	0
Yeast	0	0	0	0.015	0.016	0.857

TABLE 7. *p*-values produced by the Wilcoxon rank-sum test for equal medians on DB index.

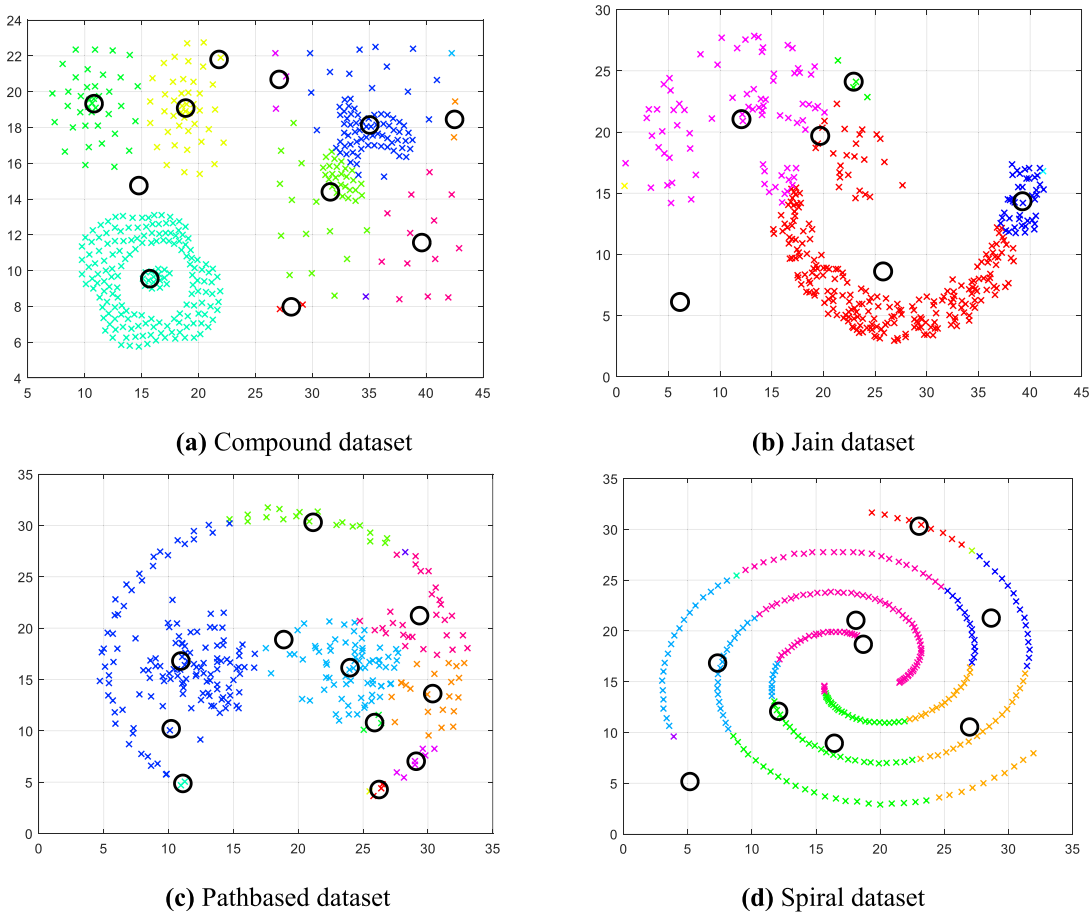
Dataset	FAPSO vs FAABC	FAPSO vs FAIWO	FAPSO vs FATLBO	FAABC vs FAIWO	FAABC vs FATLBO	FAIWO vs FATLBO
Breast	0.353	0.032	0.135	0.024	0.591	0.01
Compound	0	0	0	0.84	0	0.001
Flame	0	0	0	0.841	0.02	0.011
Glass	0	0	0	1	1	1
Iris	0	0	0	0	0.119	0
Jain	0	0	0	0.893	0.655	0.786
Pathbased	0.001	0	0	0	0.161	0
Spiral	0	0.717	0	0.001	0.844	0
Statlog	0	0	0	0.317	0.317	1
Thyroid	0	0	0	0	0.157	0
Wine	0	0	0.012	0	0.049	0
Yeast	0	0	0	0	0.001	0.381

Parameter Key Terms: The parameter a is the acceleration coefficient upper bound, S_{min} and S_{max} are the minimum and maximum number of seeds, E is the variance reduction exponent, *sigma_initial* and *sigma_final* are the values of initial and final standard deviations, c_1 and c_1 are the personal and global learning coefficients, *wdamp* is the inertia weight damping ratio, while w is the inertia weight defined as $w = w_{max} - \frac{(w_{max}-w_{min})*t}{MaxIt}$, where t denotes the number of iterations. Note that the value of w is dynamically adjusted relative to iteration t to avoid the hybrid FAPSO from plunging into premature convergence.

B. DATASETS DESCRIPTION

The twelve datasets used are well-known and well-used benchmark datasets from the UCI Machine Learning Repository. A brief description of some of the datasets are presented as follows:

- **Breast Cancer Wisconsin (Original) dataset**: this dataset was obtained from the diagnosis of breast cancer from the University of Wisconsin hospital. It contains two classes of tumour (2 benign and 4 malignant), 699 data points with 10 attributes.
- **Glass dataset**: this was obtained from the USA Forensic Science Service, defined in terms of their oxide



(a) Compound dataset

(b) Jain dataset

(c) Pathbased dataset

(d) Spiral dataset

FIGURE 9. Clustering results of hybrid FAPSO of some datasets on CS-index.

contents. The classification of this dataset was motivated as a result of criminal investigation from crime scenes, where glasses left can be used as a source of evidence if correctly identified. Nine different chemical measures, (Refractive index (RI), Sodium (Na), Aluminium (Al), Silicon (Si), Potassium (K), Calcium (Ca), Barium (Ba), Iron (Fe), and Magnesium (Mg)), were used as a standard for identifying a glass, which belongs to one of six types of glasses. It consists of 214 data points with ten attributes.

- **Iris dataset**: this dataset consists of three different variants of the iris flower, namely, Iris Setosa, Iris Versicolor and Iris Virginica. The three different species are comprised of 150 instances with four attributes.
- **Statlog (Heart) dataset:** this dataset is based on the diagnosis of heart disease from four different databases, which was generated based on 13 different attributes. It consists of 250 instances and 13 attributes.
- **Wine dataset:** the wine dataset was obtained by using chemical analysis to determine the origin of wines grown in the same region, but from three different cultivators in Italy. The analysis was able to determine the quantities made up in the 13 constituents that were found in each type of the three varieties of wines. It contains 178 patterns with 13 attributes.
- **Yeast dataset:** the yeast dataset was used to predict the localization sites of protein in cells. It contains 1484 patterns and 8 attributes.

The details of the remaining datasets namely, Jain dataset, Pathbased dataset, Spiral dataset, and Thyroid can be obtained in [38] for Chang and Yeung [39] for both Pathbased and Spiral, and [40] for the Thyroid dataset. The twelve datasets configurations are summarized in Table 2b.

C. RESULTS AND DISCUSSION

In this section, we present and discuss the average numerical results obtained by the standard FA and other four FA-based hybrid firefly algorithms. The algorithms were compared based on their computed average CS and DB indices values. In Table 3, the bolded values indicate the algorithm that obtained the best solution as compared to other competing algorithms. All the results presented in this study are in reported in four decimal places, and we focused mainly on the quality of solution produced by each of the algorithms, as well as execution time taken for each algorithm to search for the

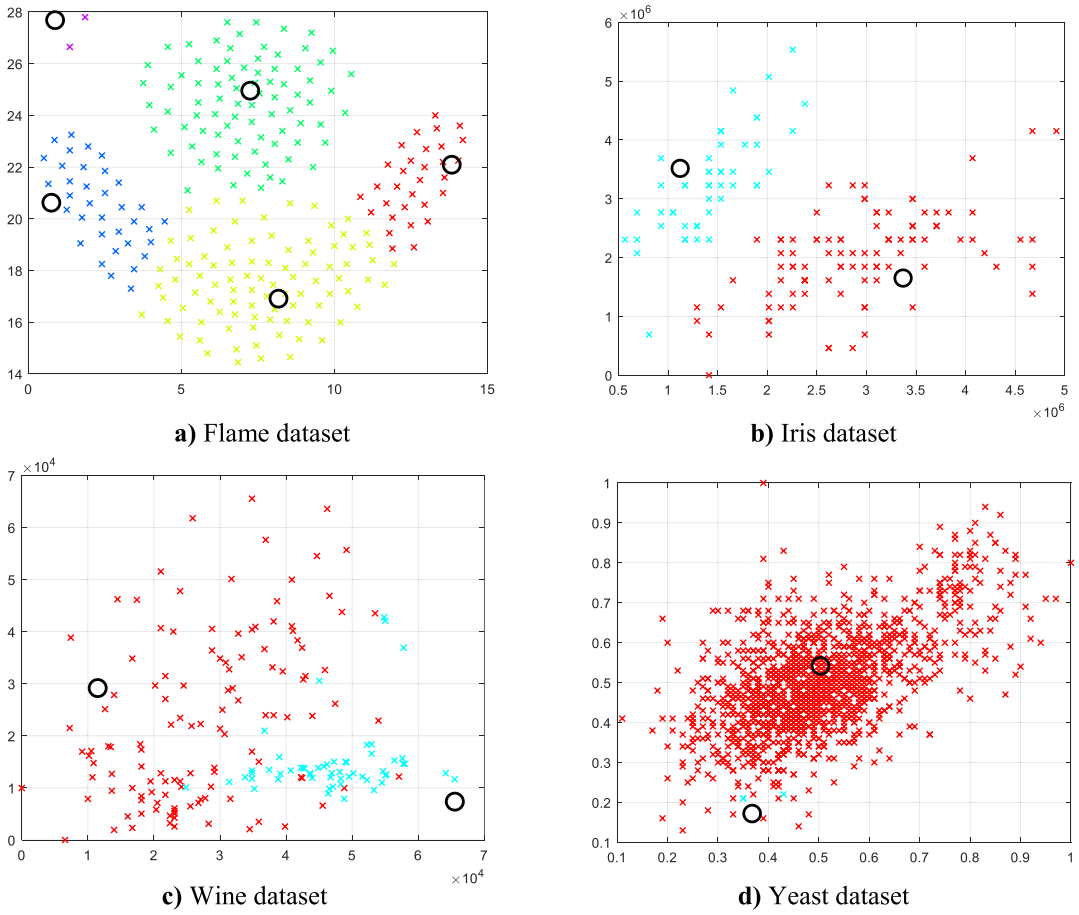


FIGURE 10. Clustering results of hybrid FAPSO of some datasets on DB-index.

near-optimal solutions. For the CS measure, it is shown that FA performed well on the Breast dataset and Flame dataset. Likewise, FAABC did well on Flame dataset. Furthermore, the FAPSO recorded most of the best performance on nine of the twelve datasets, namely, Compound, Iris, Jain, Pathbased, Spiral, Statlog, Thyroid, Wine and, Yeast datasets. On the contrary for the DB index, the best performance is seen with the FATLBO algorithm, in which it obtained best results in five datasets, namely, Flame, Iris, Pathbased, Spiral and, Yeast datasets. This is closely followed by the FAPSO which achieved the best performance in four datasets, Glass, Jain, Thyroid, and Wine datasets. Although, the standard FA did reasonably well on three datasets which are Breast, Compound, and Statlog datasets. Both FAABC and FAIWO had no outperforming solutions on the DB index. It was observed that the FA outperformed all the other algorithms on the Breast dataset in both instances of the cluster validity measure. In contrast, FAIWO did not exceed any of the different approaches for either CS or DB validity measures.

In general, the comparisons between the standard FA and its hybrid variants, show that the optimal fitness solutions achieved by the FAPSO on the CS index are lesser in values, which signifies better performance. More so, the performance of the FAPSO algorithm was able to attain excellent performance across more datasets than any other algorithms, thus making it the most superior algorithm. However, for the DB index, FATLBO emerged the best performed algorithm with the best minimum average clustering results, and the FAPSO closely follows it, then the standard FA. Therefore, since the FAPSO algorithm showed excellent performance in both instances of the validity measure, we can deduce that the FAPSO is an efficient and effective automatic clustering algorithm.

Next, we present and discuss the results of the four proposed hybrid firefly algorithms using the following descriptive statistics, namely, the best solution, worst solution, average solution and standard deviation. The highlighted values in bold indicate where an algorithm outperformed the rest of the compared algorithms or have the same results with them. As seen in the CS index column, FAABC, FAIWO and FATLBO achieved the same results on the Breast dataset, as well as with FAABC and FATLBO on Compound dataset. Likewise, FAPSO had the best performance on Flame, Iris, Jain, Thyroid, Wine and Yeast datasets. FAPSO and FATLBO achieved the best identical values for Pathbased dataset, while FAABC obtained the best solution for Spiral dataset. A level

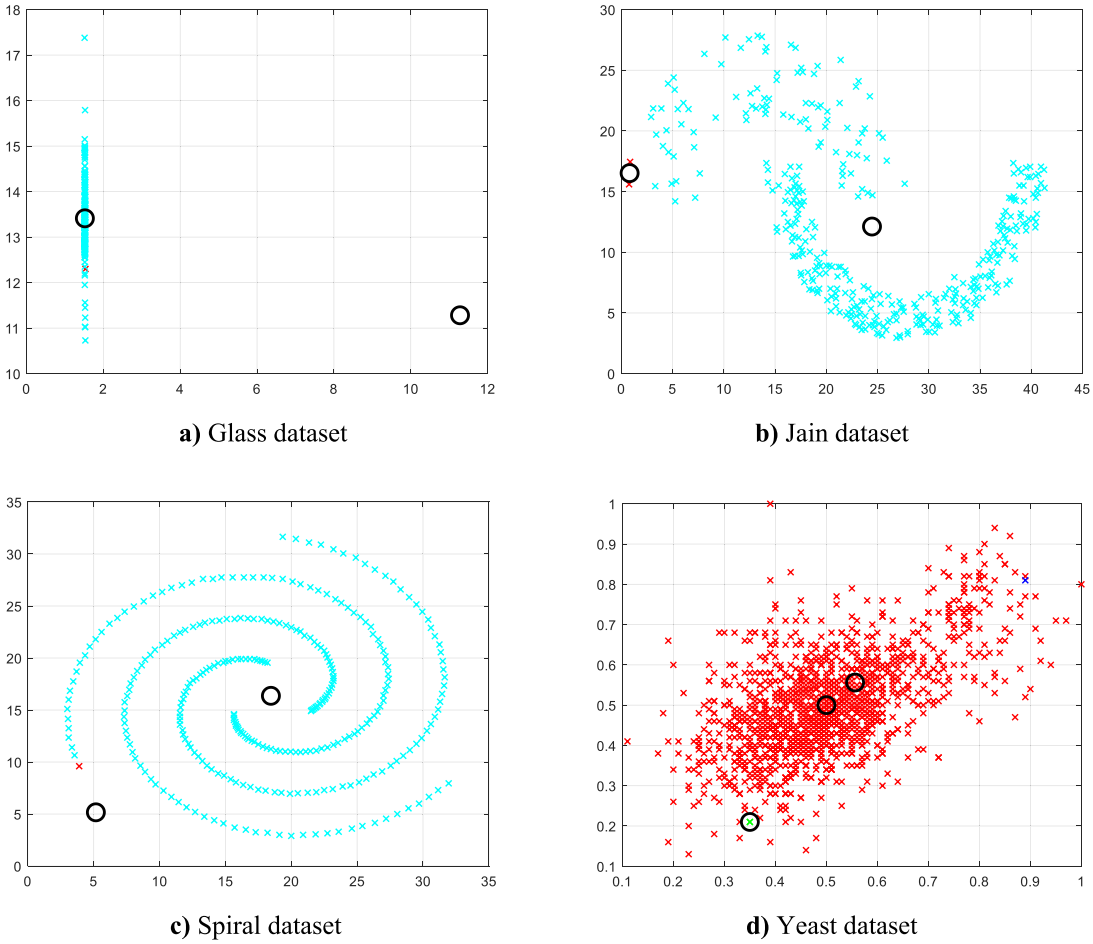


FIGURE 11. Clustering results of hybrid FATLBO of some datasets on CS-index.

of consistency and stability is shown in the results obtained in Glass and Statlog datasets, across all the four hybrid methods. Hence, FAPSO clearly shows performance superiority over the other hybrid algorithms on the CS index.

However, for the DB measure, FAABC, FAIWO and FATLBO obtained the best but identical results for the Breast dataset. These results are also similar to those of the CS index, which is to say that FAABC, FAIWO and FATLBO performance are the same for the Breast dataset in both instances of the cluster validity measures. The FAPSO achieved the best scores for Compound, Glass, Statlog and Thyroid datasets. Similar to the results obtained by FAABC, FAIWO and FATLBO for the Breast dataset, the three algorithms also had identical results for Flame and Yeast datasets. The results achieved by FAABC and FATLBO are identical for Iris and Wine datasets. FAABC outperformed the other algorithms on Spiral dataset. The values obtained by FATLBO on Jain and Pathbased datasets are superior to those of the other algorithms. Although there are a few instances where two or more algorithms have similar results in some datasets, however, this does not rule out the apparent evidence that the FAPSO outperformed the other algorithms on four datasets. For example, in Compound, Glass, Statlog and Thyroid datasets, as aforementioned. Although the values obtained by FAABC, FAIWO and FATLBO for Statlog dataset are identical as those of the CS index, FAPSO, however, obtained the overall best clustering solution. Based on these evaluations, we can, therefore, say that on the average, for all the four algorithms and across the twelve datasets, the CS index is an efficient validity measure for clustering solutions than the DB index.

Figures 3 and 4 show the average computational time consumed by each of the algorithms using the two validity indices to complete their search for optimal solutions. For both time graphs, FAACB is represented in yellow bars, FAIWO by purple bars, FAPSO by red bars, and FATLBO by blue bars. The average time consumed is plotted against corresponding algorithms and datasets. For the CS index in Figure 3, it is observed that FAPSO has the highest (worst) execution run time across the twelve datasets. The FAIWO follows this, and then FATLBO. FAABC has the best (least) run time across all the twelve test datasets. As earlier discussed, although FAPSO achieved the best solutions on CS amidst all the methods, it, however, consumed considerable time in all its

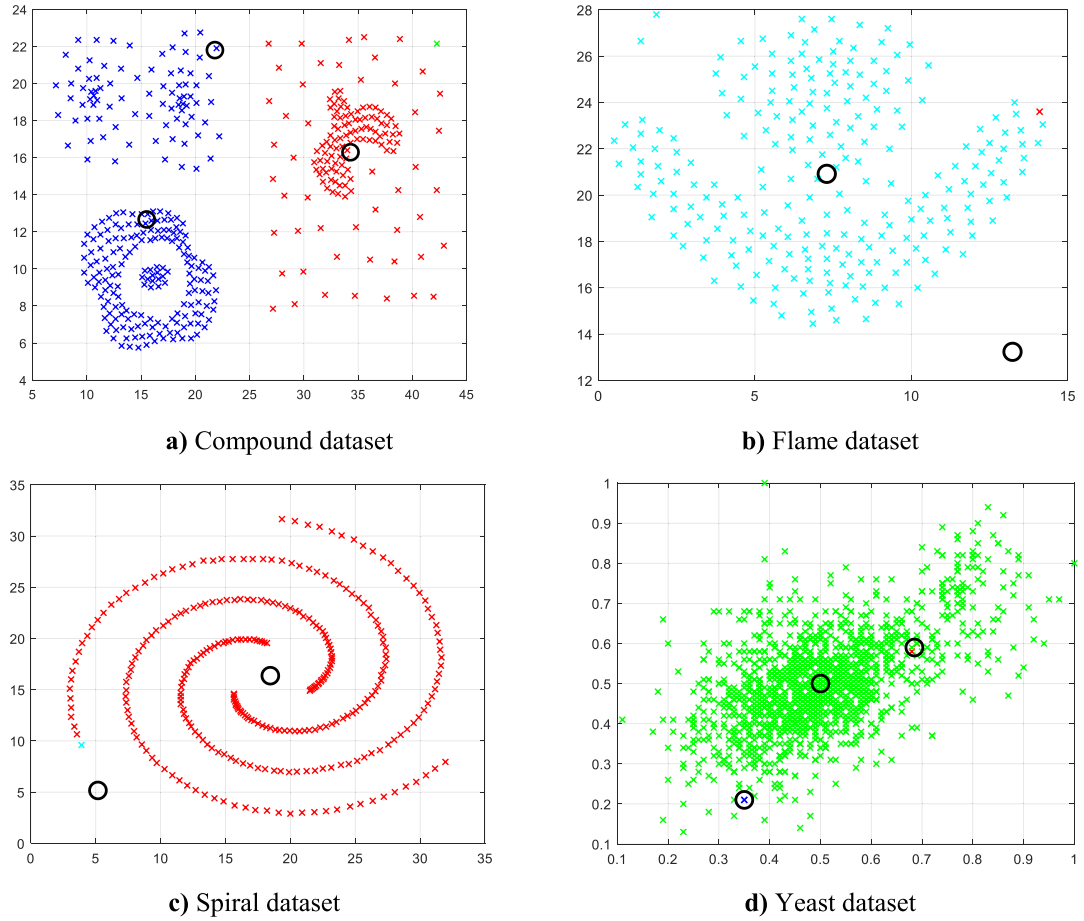


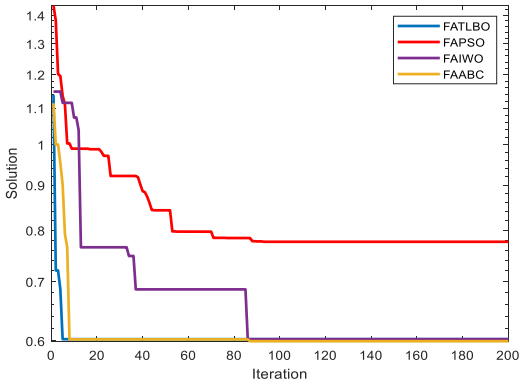
FIGURE 12. Clustering results of hybrid FATLBO of some datasets on DB-index.

search process on each dataset. Similarly, for the DB index in Figure 4, FAABC has the best execution time across all the datasets, followed by FATLBO, and then FAPSO. FAIWO has the worst (highest) run time among the four algorithms.

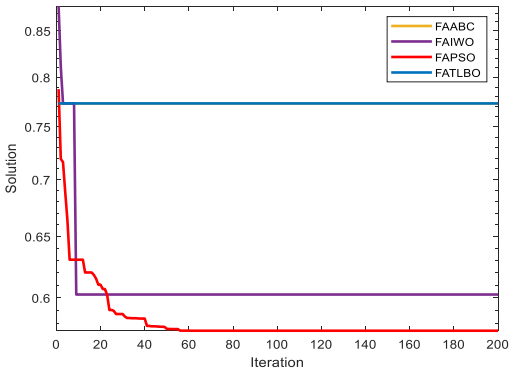
D. STATISTICAL ANALYSIS TEST

For further comparison, we performed a non-parametric statistical test called the Friedman rank-sum test, which can be used to identify the presence of any significant differences between the behaviour of two or more algorithms. As presented in Table 5, we observe that for the CS index, FAPSO particularly has the best rank on seven of the twelve datasets, namely, Iris, Jain, Pathbased, Spiral Thyroid, Wine and Yeast datasets. Similarly, there is an identical rank for all the four algorithms across Glass and Statlog datasets, as was seen in the numerical results presented above in Table 4. The FATLBO is ranked next to FAPSO in three datasets which include, Breast Compound and Flame datasets. However, strengthens the fact that FAPSO is a better efficient hybrid firefly algorithm for solving automatic data clustering problem. Yet, for DB index, FAPSO and FATLBO have a tie in their ranks on the equal number of datasets, namely Compound, Jain, Statlog, Thyroid, and Yeast datasets for FAPSO and Breast, Flame, Iris, Pathbased and Spiral datasets for the FATLBO. Finaly, FAABC, FAIWO and FATLBO have an identical mean rank-sum in Glass dataset.

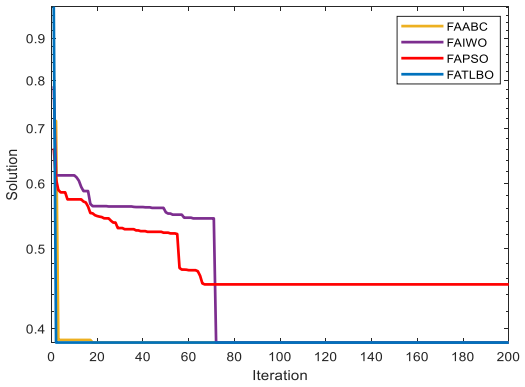
To further justify the mean ranks obtained by the Friedman test statistic in Table 5, we performed additional Wilcoxon post-hoc test to ascertain where significant statistical difference exists among the compared algorithms. Therefore, the Wilcoxon's statistics test is used in this case to aid us to draw a meaningful statistical conclusion. Tables 6 and 7 reports the p-values produced by this posthoc analysis for the pairwise comparison of FAPSO vs FAABC, FAPSO vs FAIWO, FAPSO vs FATLBO, FAABC vs FAIWO, FAABC vs FATLBO and, FAIWO vs FATLBO, for both the CS and DB validity indices respectively. Almost all the values are below our adjusted p-value of ($p \leq 0.0083$). We obtained a great number of statistically significant values on pairwise of FAPSO with other algorithms than the other pairwise in both cases of CS and DB indices. Hence, this further proves the superiority of FAPSO over other methods with a clear indication that the algorithm is a robust and efficient hybrid firefly algorithm for carrying out the task of automatic data clustering.



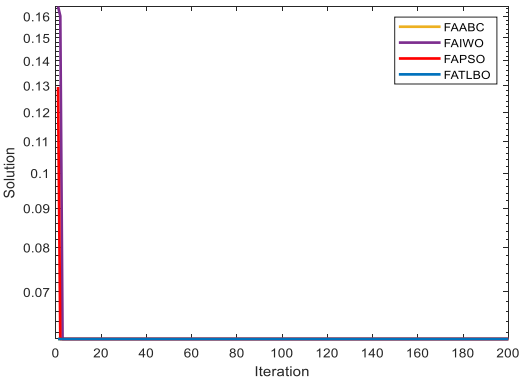
a) Breast dataset



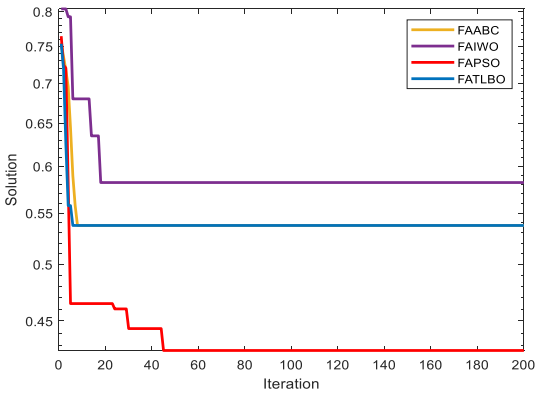
b) Compound dataset



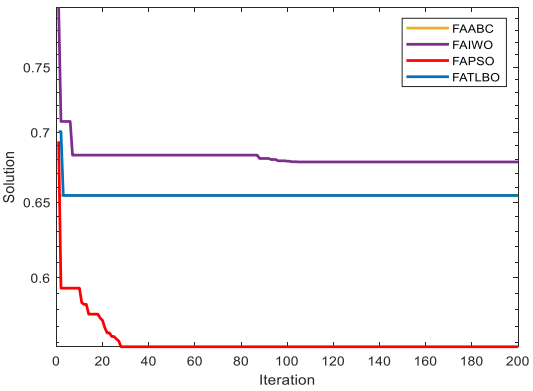
c) Flame dataset



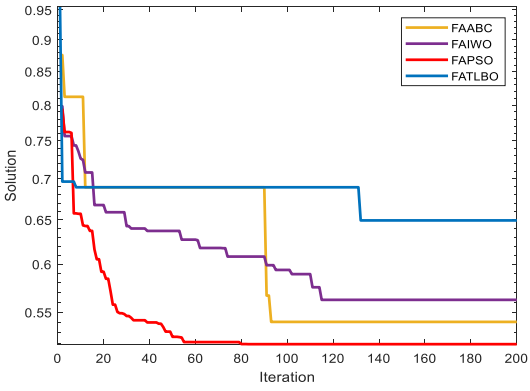
d) Glass dataset



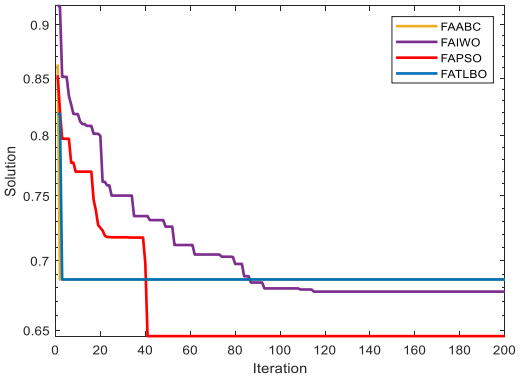
e) Iris dataset



f) Jain dataset



g) Pathbased dataset



h) Spiral dataset

FIGURE 13. **Convergence curves on CS-index of each algorithm on all the test datasets.**

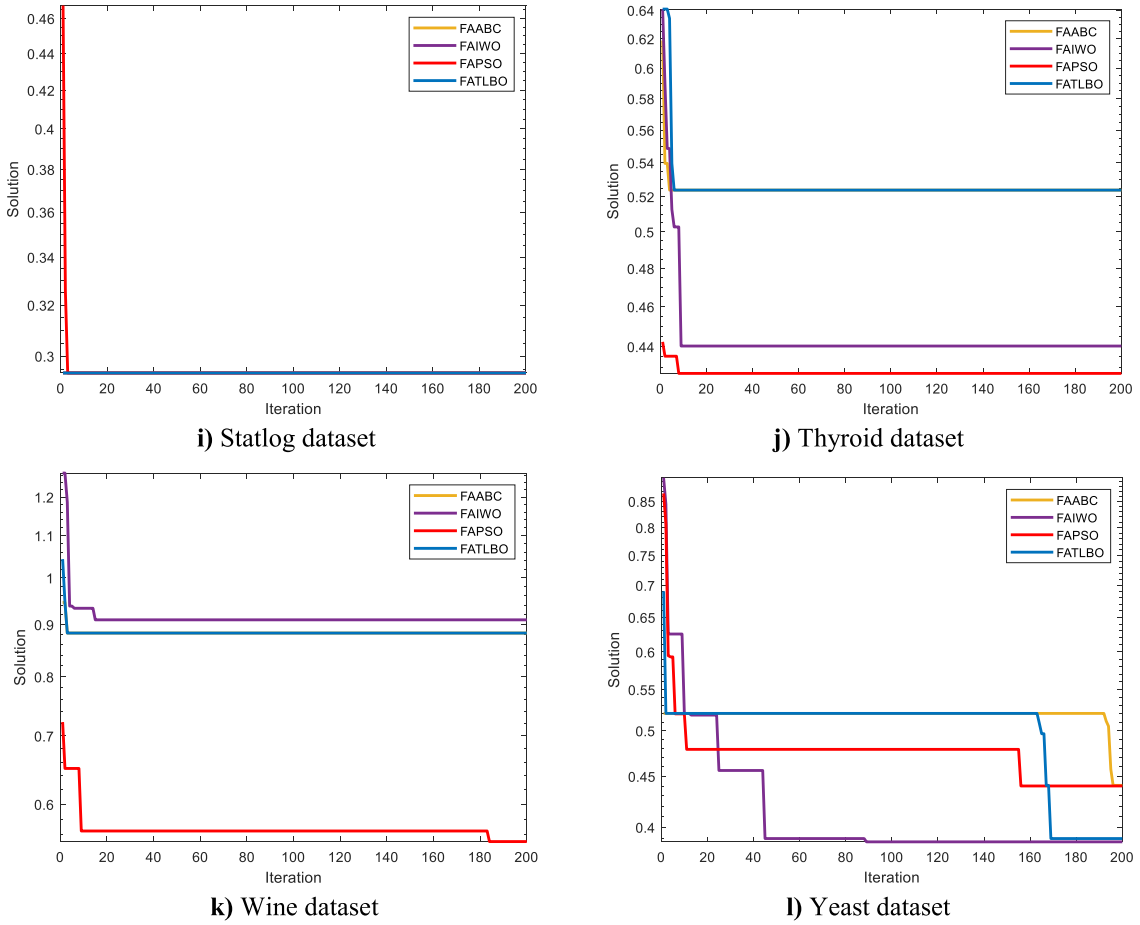


FIGURE 13. *(Continued.)* **Convergence curves on CS-index of each algorithm on all the test datasets.**

E. CLUSTERING PROCESS

The clustering results of some selected datasets for all the algorithm on CS and DB index across all the four hybrid algorithms are presented in Figures 5-12. In Figure 5 (FAABC based on CS index), we have three perfect clusters for the Compound dataset, while we have one cluster for Flame, Pathbased and Yeasts datasets, although with a blue string outlier on them which is not noticeable. Likewise, for Figure 6 (FAABC based on DB index), we have good clustering but with red stringed-outlier on the Glass and Jain dataset, and a red stringed-outlier in the Spiral cluster. Figures 7 and 8 show clustering results for FAIWO using CS and DB measures, respectively. The compound dataset has three exact clusters, while Statlog has one. A blue string of outlier is noticed in Flame and Iris datasets, as shown in Figure 7. Also in Figure 8, Compound dataset is well clustered into three groups, Pathbased into one group with a string of red outlier, Thyroid dataset was equally classified into three classes of blue, green and red. In contrast, the Wine dataset was classified into one class but with a blue exception class.

In Figure 9 on the Compound dataset, a small part of the magenta and yellow class are mixed with the blue class, but the dataset is divided into six classes. Also, some outliers were not properly grouped, which are present in the green class. For the Jain dataset, we had three distinct clusters with a few green outliers attached to the magenta class. Also, in Pathbased and Spiral datasets, we had five and six clearly separated classes, respectively. Likewise, for DB index as shown in Figure 10, all the selected datasets had perfect clustering which is well separated and presented on each graph, except the Yeast dataset that had a few outliers around it.

A good clustering result is presented for FATLBO, according to Figures 11 and 12. In Figure 11, FATLBO achieved one clustered distinct group on each of the selected datasets, with a few outliers of red, blue and green outliers that are not noticeable. While the Compound dataset has three definite clusters, one cluster for each of Flame, Spiral, and Yeast datasets, they, however, had green, red and blue outliers, as seen in Fig. 12.

F. ALGORITHM CONVERGENCE CURVES

The equivalent convergence comparison curves for the four hybrid algorithms are presented in Figures 13 and 14. The overall convergence evaluation for the respective algorithm on both the CS and DB measures show that the FAPSO

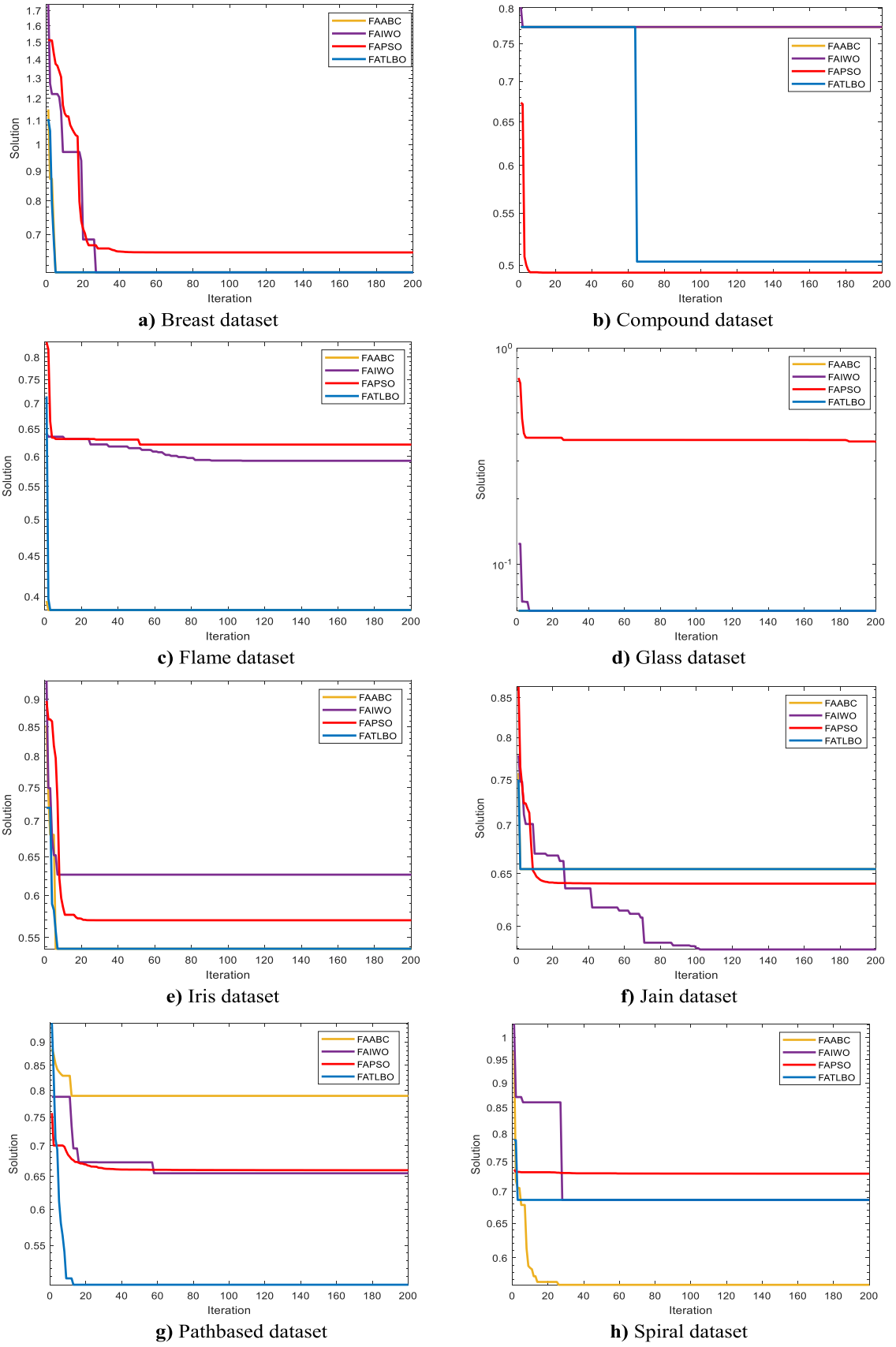


FIGURE 14. Convergence curves on DB-index of each algorithm on all the test datasets.

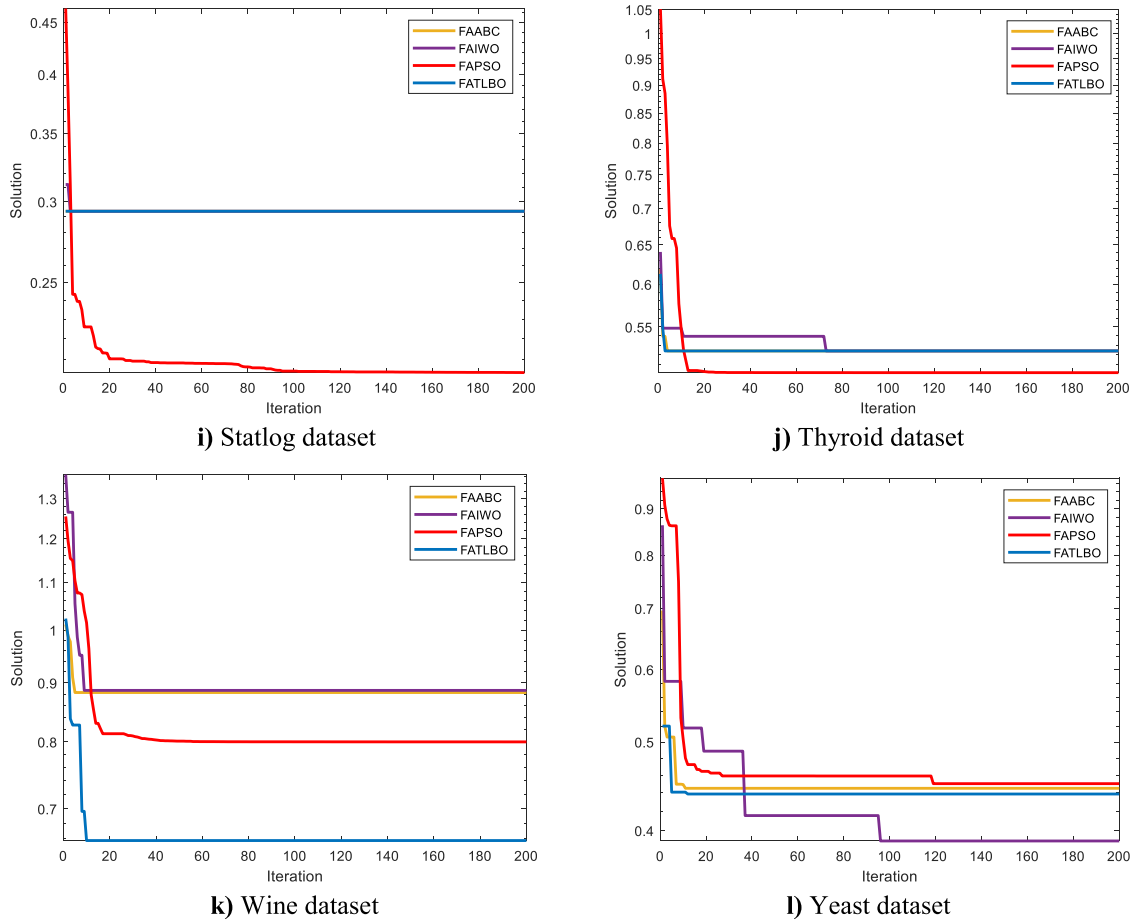


FIGURE 14. *(Continued.)* **Convergence curves on DB-index of each algorithm on all the test datasets.**

converges better than FAABC, FAIWO and FATLBO. Next to FAPSO is FATLBO, which obtained fair convergence than FAABC and FAIWO, while FAIWO converged the poorest in both instances of the validity measures.

G. HIGH-DIMENSIONAL DATASET AND PARAMETER FINE-TUNING

In this section, an additional experiment was carried out to determine the scaling performance behaviour of the two best performed algorithms proposed in this paper, namely, FAPSO and FATLBO on seven relatively high dimensional datasets. The performance of the two algorithms is further validated by fine-tuning their standard control parameters, which in this case is the population size. The population size of 50 and 100 were selected for the parameter tuning task. On the one hand, the parameter tuning measure assists in evaluating the impact of control parameters for the two algorithms, which might somewhat affect the performance of the individual algorithm either negatively or positively in terms of solution quality or computational cost complexity. The results of the fine-tuning experiment are shown in Tables 8 and 9, respectively. Note that the results of FAPSO and FATLBO are compared with those of three hybrid algorithms from literature namely, particle swarm optimization differential evolution (PSODE) [40], firefly algorithm differential evolution (FADE) [40], and invasive weed optimization differential evolution (IWODE) [40]. Each of these algorithms is implemented and executed under the same experimental conditions, which makes it logical to compare their clustering results and computational costs.

For the two algorithms, some noticeable performance improvements in the solution quality were observed as compared to the results of the hybrid methods from the literature [40]. However, the observed improvements were at the expense of computational time, which increased significantly as shown in the two tables 8 and 9 below. The FAPSO obtained the least average solution with 0.5411 and 0.5700, followed by FATLBO with 0.5719 and 0.6096 for both population sizes of 50 and 100 as compared to literature results. However, with an increase in the number of population size, there is no significant improvement in terms of clustering solution quality based on the results obtained by the hybrid FATLBO.

The results of the computational time complexity for the FAPSO and FATLBO algorithms implementation are presented alongside the obtained clustering solutions

TABLE 8. Results obtained by FAPSO and FATLBO for high-dimensional dataset using a population size of 50.

			FAPSO		FATLBO		PSODE [40]		FADE [40]		IWODE [40]	
Datasets	N	K	Solution	Time	Solution	Time	Solution	Time	Solution	Time	Solution	Time
Bridge	4,096	26	**0.5901**	103.1013	0.6108	90.9244	0.6734	27.0772	0.6109	97.1574	0.9997	23.402
Letter	20,000	256	**0.7146**	144.6228	0.7775	105.2789	0.9194	67.6421	0.7829	317.1155	1.1752	49.3524
Housec5	34,112	256	**0.4187**	197.6234	0.4518	110.2976	0.4987	19.037	0.5652	47.0923	0.6569	21.4191
Housec8	34,112	256	**0.4245**	221.2172	0.4559	128.8341	0.4559	47.9596	0.4559	244.5398	0.5485	40.823
Birch1	100,000	100	**0.6572**	271.1421	0.6952	185.0492	0.6997	163.7943	0.7189	818.3993	0.8085	120.1747
Birch2	100,000	100	**0.504**	282.3312	0.5163	190.2154	0.507	149.0613	0.507	637.688	0.5142	94.1641
Birch3	100,000	100	**0.6812**	291.4421	0.7596	201.1128	0.6789	174.5673	0.6911	876.4011	0.7372	104.7143
		Average	0.5700	215.9257	0.6096	144.5303	0.6333	92.7341	0.6188	434.0562	0.7772	64.8642

TABLE 9. Results obtained by FAPSO and FATLBO for high-dimensional dataset using a population size of 100.

			FAPSO		FATLBO		PSODE [40]		FADE [40]		IWODE [40]	
Datasets	N	K	Solution	Time	Solution	Time	Solution	Time	Solution	Time	Solution	Time
Bridge	4,096	26	**0.5821**	222.6785	0.6001	107.188	0.6116	38.2201	0.6109	221.3055	1.0494	28.6025
Letter	20,000	256	**0.7004**	1795.8947	0.7613	132.698	0.7872	124.1709	0.795	971.6205	1.1698	75.5472
Housec5	34,112	256	**0.4017**	257.9266	0.4415	144.7786	0.5598	23.3929	0.4987	184.3469	0.676	21.9806
Housec8	34,112	256	**0.4159**	287.0499	0.4619	189.7457	0.5209	86.9351	0.4559	1025.8784	0.5151	72.5537
Birch1	100,000	100	**0.6321**	357.5953	0.6510	217.4207	0.6875	351.0767	0.7186	1855.6013	0.7304	277.3295
Birch2	100,000	100	**0.4570**	233.9347	0.4836	295.4221	0.507	240.1282	0.507	1556.589	0.5103	284.7349
Birch3	100,000	100	**0.5988**	394.0758	0.6036	299.4511	0.6972	367.0218	0.6911	1710.561	0.7277	157.8827
		Average	0.5411	507.0222	0.5719	198.1006	0.6245	175.8494	0.6110	1075.1289	0.7684	131.2330

in Tables 8 and 9. One of the significant drawbacks of the parameter fine-tuning is that the running time considerable grows for each algorithm. For example, the FAPSO even though it produced the best clustering solution in terms of cohesion and compactness, the computational costs increased exponentially relative to population size. Although, similar characteristics behaviour was displayed in the computation cost obtained by other hybrid algorithms from the literature. However, this is expected because the hybrid implementation process incorporates additional subroutine processing overhead, which invariably increases the execution time complexities of the combined algorithms. Thus the high computational cost recorded by both FAPSO and FATLBO.

H. ALGORITHM COMPLEXITY

In determining the complexity of any metaheuristic algorithm, there is no one size fits all solution that can be applied. Although the detailed computational complexity may depend on the structure of the algorithm design and implementation [29]. However, for the five proposed metaheuristic algorithms used in this paper, their complexities can be easily estimated. For the improved FA algorithm, the time complexity is defined as $O(n^2t)$ where n denotes the number of population size used, which in this case is $n = 25$ and $\boldsymbol{t}$ represents the number of iterations. Also note that for the sake of simplicity in the implementation process, all the five proposed algorithms, including FA and the four hybrids, namely, FAPSO, FAABC, FATLBO, and FAIWO algorithms have two inner loops when going through the entire population n. Therefore, for the four proposed hybrid algorithms, the time complexity is defined as $O\left(\frac{n^2t}{4} + \frac{n^2t}{2}\right)$, this is because each section of the four single or individual representative algorithms only uses half of the population size. Also, as the values of n and t that were used for the experiments reported in this paper are small (typically, $n = 25$, $t = 200$), the computation cost is relatively inexpensive because the algorithm complexity is linear in terms of $\boldsymbol{t}$. Similarly, also note that the main computational cost relies on the evaluations of the defined clustering task objective function.

Further, similar to some other metaheuristic algorithms, the FA, which is used as the core representative algorithm for the proposed hybrid techniques have some limitations as follows: FA optimal performance highly depends on adequate parameter fine-tuning, diversification in FA can lead to reduced computational speed and convergence rate, FA is not very suitable for handling complex problems, because it can be trapped in many local optima in the event of searching for

possible candidate solutions [29]. However, because each of the hybrid methods depends on the FA, their performance can as well be restricted, precisely due to the parameter tuning effects and over-diversification or exploration mechanism of the FA base algorithm. These limitations were experienced when the hybrid algorithms were subjected to clustering task that involves the use of high dimensional datasets.

V. CONCLUSION

In this study, four new FA-based hybrid algorithms were implemented and successfully used to solve automatic data clustering problems. Subsequently, a performance study of the respective proposed algorithms was carried out. The simulation results obtained from the multiple experiments executed revealed that the FAPSO outperformed the other hybrid algorithms, including the FAABC, FAIWO and FATLBO, respectively, in terms of solution quality and convergence speed. On the other hand, the FATLBO seemed to have equally performed relatively well and was next to the FAPSO algorithm, as it was able to yield high clustering solutions and better computational speed as well. However, the FAIWO appeared to be the least superior methods in terms of clustering quality and speed of convergence. In future research, we intend to apply the proposed FA-based hybrid algorithms to solve other complex optimization problems with similar settings and possibly on variants of the clustering problem considered in this paper. Similarly, it will be interesting to see some high-level extension of the proposed hybrid clustering algorithms that would dynamically enable the individual algorithms to determine the set of optimal parameter configuration for maximum performance improvement of the individual process.

Finally, the possibility of combining FA algorithms with some recent deep learning clustering methods such as the deep embedding clustering [83], deep clustering network [84], pairwise constraints clustering [85], deep embedding network [86], joint unsupervised learning of deep representation for images [84], deep learning with non-parametric clustering [87], convolutional neural network clustering [88] and deep clustering with convolutional autoencoder embedding [90] can be investigated to solve real-world data clustering problems, specifically, those problems with high dimensionality and complex features.

CONFLICT OF INTERESTS

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

REFERENCES

[1] M. R. Anderberg, *Cluster Analysis for Applications: Probability and Mathematical Statistics: A Series of Monographs and Textbooks*, vol. 19. New York, NY, USA: Academic, 2014.
[2] J. A. Hartigan, *Clustering Algorithms*. New York, NY, USA: Wiley, 1975.
[3] P. A. Devijver and J. Kittler, *Pattern Recognition: A Statistical Approach*. Upper Saddle River, NJ, USA: Prentice-Hall, 1982.
[4] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1988.
[5] S. J. Roberts, "Parametric and non-parametric unsupervised cluster analysis," *Pattern Recognit.*, vol. 30, no. 2, pp. 261–272, Feb. 1997.
[6] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*, vol. 20. Philadelphia, PA, USA: SIAM, 2007.
[7] T. Soni Madhulatha, "An overview on clustering methods," 2012, *arXiv:1205.1117*. [Online]. Available: http://arxiv.org/abs/1205.1117
[8] Y. Zhou, H. Wu, Q. Luo, and M. Abdel-Baset, "Automatic data clustering using nature-inspired symbiotic organism search algorithm," *Knowl.-Based Syst.*, vol. 163, pp. 546–557, Jan. 2019.
[9] R. B. Cattell, "The description of personality: Basic traits resolved into clusters.," *J. Abnormal Social Psychol.*, vol. 38, no. 4, pp. 476–506, 1943.
[10] F. Murtagh, "A survey of recent advances in hierarchical clustering algorithms," *Comput. J.*, vol. 26, no. 4, pp. 354–359, 1983.
[11] S. J. Nanda and G. Panda, "A survey on nature inspired Metaheuristic algorithms for partitional clustering," *Swarm Evol. Comput.*, vol. 16, pp. 1–18, Jun. 2014.
[12] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, Jun. 2010.
[13] M. Sharma and J. K. Chhabra, "Sustainable automatic data clustering using hybrid PSO algorithm with mutation," *Sustain. Comput., Informat. Syst.*, vol. 23, pp. 144–157, Sep. 2019.
[14] A. José-García and W. Gómez-Flores, "Automatic clustering using nature-inspired metaheuristics: A survey," *Appl. Soft Comput.*, vol. 41, pp. 192–213, Apr. 2016.
[15] Y. Liu, X. Wu, and Y. Shen, "Automatic clustering using genetic algorithms," *Appl. Math. Comput.*, vol. 218, no. 4, pp. 1267–1279, Oct. 2011.
[16] F. Zabihi and B. Nasiri, "A novel history-driven artificial bee colony algorithm for data clustering," *Appl. Soft Comput.*, vol. 71, pp. 226–241, Oct. 2018.
[17] D. W. Van der Merwe, A. P. Engelbrecht, "Data clustering using particle swarm optimization," in *Proc. Congr. Evol. Comput. (CEC)*, vol. 1, Dec. 2003, pp. 215–220.
[18] M. Zhao, H. Tang, J. Guo, and Y. Sun, "Data clustering using particle swarm optimization," in *Future Information Technology*. Berlin, Germany: Springer, 2014, pp. 607–612.
[19] T. Niknam, J. Olamaei, and B. Amiri, "A hybrid evolutionary algorithm based on ACO and SA for cluster analysis," *J. Appl. Sci.*, vol. 8, no. 15, pp. 2695–2702, Dec. 2008.
[20] S. C. Satapathy and A. Naik, "Data clustering based on teaching-learning-based optimization," in *Proc. Int. Conf. Swarm, Evol., Memetic Comput.* Berlin, Germany: Springer, Dec. 2011, pp. 148–156.
[21] A. J. Sahoo and Y. Kumar, "Modified teacher learning based optimization method for data clustering," in *Advances in Signal Processing and Intelligent Recognition Systems*. Cham, Switzerland: Springer, 2014, pp. 429–437.
[22] X.-Q. Zhao and J.-H. Zhou, "Improved kernel possibilistic fuzzy clustering algorithm based on invasive weed optimization," *J. Shanghai Jiaotong Univ.*, vol. 20, no. 2, pp. 164–170, Apr. 2015.
[23] R. Liu, X. Wang, Y. Li, and X. Zhang, "Multi-objective invasive weed optimization algortihm for clustering," in *Proc. IEEE Congr. Evol. Comput.*, Jun. 2012, pp. 1–8.
[24] R. Wang, Y. Zhou, S. Qiao, and K. Huang, "Flower pollination algorithm with bee pollinator for cluster analysis," *Inf. Process. Lett.*, vol. 116, no. 1, pp. 1–14, Jan. 2016.
[25] P. Agarwal and S. Mehta, "Enhanced flower pollination algorithm on data clustering," *Int. J. Comput. Appl.*, vol. 38, nos. 2–3, pp. 144–155, Jul. 2016.
[26] J. Senthilnath, S. N. Omkar, and V. Mani, "Clustering using firefly algorithm: Performance study," *Swarm Evol. Comput.*, vol. 1, no. 3, pp. 164–171, 2011.
[27] H. Banati and M. Bajaj, "Performance analysis of firefly algorithm for data clustering," *Int. J. Swarm Intell.*, vol. 1, no. 1, pp. 19–35, 2013.
[28] A. A. Abshouri and A. Bakhtiary, "A new clustering method based on firefly and KHM," *J. Commun. Comput.*, vol. 9, no. 4, pp. 387–391, 2012.
[29] X.-S. Yang and X. He, "Firefly algorithm: Recent advances and applications," *Int. J. Swarm Intell.*, vol. 1, no. 1, pp. 36–50, 2013.
[30] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*. York, U.K.: Luniver Press, 2010.
[31] A. E. Ezugwu, O. J. Adeleke, A. A. Akinyelu, and S. Viriri, "A conceptual comparison of several metaheuristic algorithms on continuous optimisation problems," *Neural Comput. Appl.*, vol. 32, pp. 6207–6251, Mar. 2020.
[32] X.-S. Yang, "Firefly algorithm, stochastic test functions and design optimization," *Int. J. Bio-Inspired Comput.*, vol. 2, no. 2, pp. 78–84, 2010.

[33] A. E. Ezugwu and F. Akutsah, "An improved firefly algorithm for the unrelated parallel machines scheduling problem with sequence-dependent setup times," *IEEE Access*, vol. 6, pp. 54459–54478, 2018.
[34] S. Das, A. Abraham, and A. Konar, "Automatic clustering using an improved differential evolution algorithm," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 38, no. 1, pp. 218–237, Jan. 2008.
[35] C.-H. Chou, M.-C. Su, and E. Lai, "A new cluster validity measure and its application to image compression," *Pattern Anal. Appl.*, vol. 7, no. 2, pp. 205–220, Jul. 2004.
[36] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979.
[37] L. Zhang, L. Liu, X.-S. Yang, and Y. Dai, "A novel hybrid firefly algorithm for global optimization," *PLoS ONE*, vol. 11, no. 9, Sep. 2016, Art. no. e0163230.
[38] A. K. Jain and M. H. Law, "Data clustering: A user's dilemma," in *Proc. Int. Conf. Pattern Recognit. Mach. Intell.* Berlin, Germany: Springer, Dec. 2005, pp. 1–10.
[39] H. Chang and D.-Y. Yeung, "Robust path-based spectral clustering," *Pattern Recognit.*, vol. 41, no. 1, pp. 191–203, Jan. 2008.
[40] A. E. Ezugwu, "Nature-inspired Metaheuristic techniques for automatic clustering: A survey and performance study," *Social Netw. Appl. Sci.*, vol. 2, no. 2, p. 273, Feb. 2020.
[41] I. Fister, X. S. Yang, and D. Fister, "Firefly algorithm: A brief review of the expanding literature," in *Cuckoo Search and Firefly Algorithm*. Cham, Switzerland: Springer, 2014, pp. 347–360.
[42] T. Hassanzadeh and M. R. Meybodi, "A new hybrid approach for data clustering using firefly algorithm and K-means," in *Proc. 16th CSI Int. Symp. Artif. Intell. Signal Process. (AISP)*, May 2012, pp. 007–011.
[43] D. Karim, H. Salah, and C. Adnen, "Hybridization DE with K-means for speaker clustering in speaker diarization of broadcasts news," *Int. J. Speech Technol.*, vol. 22, no. 4, pp. 893–909, Dec. 2019.
[44] S. Paterlini and T. Krink, "Differential evolution and particle swarm optimization in partitional clustering," *Comput. Statist. Data Anal.*, vol. 50, no. 5, pp. 1220–1247, 2006.
[45] K. Panetta, C. Gao, and S. Agaian, "No reference color image contrast and quality measures," *IEEE Trans. Consum. Electron.*, vol. 59, no. 3, pp. 643–651, Aug. 2013.
[46] H. H. Hoos and T. Stützle, *Stochastic Local Search: Foundations and Applications*. Amsterdam, The Netherlands: Elsevier, 2004.
[47] K. K. Maheshwar and V. Arora, "A hybrid data clustering using firefly algorithm based improved genetic algorithm," *Procedia Comput. Sci.*, vol. 58, pp. 249–256, Aug. 2015.
[48] J. Nayak, M. Nanda, K. Nayak, B. Naik, and H. S. Behera, "An improved firefly fuzzy C-means (FAFCM) algorithm for clustering real world data sets," in *Advanced Computing, Networking and Informatics*, vol. 1. Cham, Switzerland: Springer, 2014, pp. 339–348.
[49] S. Sundararajan and S. Karthikeyan, "An efficient hybrid approach for data clustering using dynamic K-means algorithm and firefly algorithm," *J. Eng. Appl. Sci.*, vol. 9, no. 8, pp. 1348–1353, 2014.
[50] M. B. Agbaje, A. E. Ezugwu, and R. Els, "Automatic data clustering using hybrid firefly particle swarm optimization algorithm," *IEEE Access*, vol. 7, pp. 184963–184984, 2019.
[51] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Global Optim.*, vol. 39, no. 3, pp. 459–471, Oct. 2007.
[52] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems," *Comput.-Aided Des.*, vol. 43, no. 3, pp. 303–315, 2011.
[53] M. G. H. Omran, A. Salman, and A. P. Engelbrecht, "Dynamic clustering using particle swarm optimization with application in image segmentation," *Pattern Anal. Appl.*, vol. 8, no. 4, pp. 332–344, Feb. 2006.
[54] H. Masoud, S. Jalili, and S. M. H. Hasheminejad, "Dynamic clustering using combinatorial particle swarm optimization," *Int. J. Speech Technol.*, vol. 38, no. 3, pp. 289–314, Apr. 2013.
[55] H.-L. Ling, J.-S. Wu, Y. Zhou, and W.-S. Zheng, "How many clusters? A robust PSO-based local density model," *Neurocomputing*, vol. 207, pp. 264–275, Sep. 2016.
[56] R. J. Kuo and F. E. Zulvia, "Automatic clustering using an improved particle swarm optimization," *J. Ind. Intell. Inf.*, vol. 1, no. 1, pp. 46–51, 2013.
[57] S. J. Nanda and G. Panda, "Automatic clustering algorithm based on multi-objective immunized PSO to classify actions of 3D human models," *Eng. Appl. Artif. Intell.*, vol. 26, nos. 5–6, pp. 1429–1441, May 2013.
[58] A. Abraham, S. Das, and A. Konar, "Kernel based automatic clustering using modified particle swarm optimization algorithm," in *Proc. 9th Annu. Conf. Genetic Evol. Comput. (GECCO)*, Jul. 2007, pp. 2–9.
[59] Y. Kao and C. C. Chen, "Automatic clustering for generalized cell formation using a hybrid particle swarm optimization," *Int. J. Prod. Res.*, vol. 52, no. 12, pp. 3466–3484, 2014.
[60] A. Abubaker, A. Baharum, and M. Alrefaei, "Automatic clustering using multi-objective particle swarm and simulated annealing," *PLoS ONE*, vol. 10, no. 7, Jul. 2015, Art. no. e0130995.
[61] W.-P. Lee and S.-W. Chen, "Automatic clustering with differential evolution using cluster number oscillation method," in *Proc. 2nd Int. Workshop Intell. Syst. Appl.*, May 2010, pp. 1–4.
[62] I. Saha, U. Maulik, and S. Bandyopadhyay, "A new differential evolution based fuzzy clustering for automatic cluster evolution," in *Proc. IEEE Int. Advance Comput. Conf.*, Mar. 2009, pp. 706–711.
[63] U. Maulik and I. Saha, "Automatic fuzzy clustering using modified differential evolution for image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 9, pp. 3503–3510, Sep. 2010.
[64] K. Suresh, D. Kundu, S. Ghosh, S. Das, and A. Abraham, "Automatic clustering with multi-objective differential evolution algorithms," in *Proc. IEEE Congr. Evol. Comput.*, May 2009, pp. 2590–2597.
[65] D. Kundu, K. Suresh, S. Ghosh, S. Das, A. Abraham, and Y. Badr, "Automatic clustering using a synergy of genetic algorithm and multi-objective differential evolution," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.* Berlin, Germany: Springer, Jun. 2009, pp. 177–186.
[66] Y. Zhong, S. Zhang, and L. Zhang, "Automatic fuzzy clustering based on adaptive multi-objective differential evolution for remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 5, pp. 2290–2301, Oct. 2013.
[67] H. He and Y. Tan, "A two-stage genetic algorithm for automatic clustering," *Neurocomputing*, vol. 81, pp. 49–59, Apr. 2012.
[68] M. A. Rahman and M. Z. Islam, "A hybrid clustering technique combining a novel genetic algorithm with K-Means," *Knowl.-Based Syst.*, vol. 71, pp. 345–365, Nov. 2014.
[69] C. Ozturk, E. Hancer, and D. Karaboga, "Dynamic clustering with improved binary artificial bee colony algorithm," *Appl. Soft Comput.*, vol. 28, pp. 69–80, Mar. 2015.
[70] R. J. Kuo, Y. D. Huang, C.-C. Lin, Y.-H. Wu, and F. E. Zulvia, "Automatic kernel clustering with bee colony optimization algorithm," *Inf. Sci.*, vol. 283, pp. 107–122, Nov. 2014.
[71] R. J. Kuo and F. E. Zulvia, "Automatic clustering using an improved artificial bee colony optimization for customer segmentation," *Knowl. Inf. Syst.*, vol. 57, no. 2, pp. 331–357, Nov. 2018.
[72] S. Das, A. Chowdhury, and A. Abraham, "A bacterial evolutionary algorithm for automatic data clustering," in *Proc. IEEE Congr. Evol. Comput.*, May 2009, pp. 2403–2410.
[73] H. Peng, J. Wang, P. Shi, A. Riscos-Núñez, and M. J. Pérez-Jiménez, "An automatic clustering algorithm inspired by membrane computing," *Pattern Recognit. Lett.*, vol. 68, pp. 34–40, Dec. 2015.
[74] V. Kumar, J. K. Chhabra, and D. Kumar, "Automatic data clustering using parameter adaptive harmony search algorithm and its application to image segmentation," *J. Intell. Syst.*, vol. 25, no. 4, pp. 595–610, 2016.
[75] R. Liu, B. Zhu, R. Bian, Y. Ma, and L. Jiao, "Dynamic local search based immune automatic clustering algorithm and its applications," *Appl. Soft Comput.*, vol. 27, pp. 250–268, Feb. 2015.
[76] V. Kumar, J. K. Chhabra, and D. Kumar, "Automatic cluster evolution using gravitational search algorithm and its application on image segmentation," *Eng. Appl. Artif. Intell.*, vol. 29, pp. 93–103, Mar. 2014.
[77] S. Kapoor, I. Zeya, C. Singhal, and S. J. Nanda, "A grey wolf optimizer based automatic clustering algorithm for satellite image segmentation," *Procedia Comput. Sci.*, vol. 115, pp. 415–422, Jan. 2017.
[78] B. Anari, J. Akbari Torkestani, and A. M. Rahmani, "Automatic data clustering using continuous action-set learning automata and its application in segmentation of images," *Appl. Soft Comput.*, vol. 51, pp. 253–265, Feb. 2017.
[79] M. A. Elaziz, N. Nabil, A. A. Ewees, and S. Lu, "Automatic data clustering based on hybrid atom search optimization and sine-cosine algorithm," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2019, pp. 2315–2322.
[80] L. E. Agustín-Blas, S. Salcedo-Sanz, S. Jiménez-Fernández, L. Carro-Calvo, J. D. Ser, and J. A. Portilla-Figueras, "A new grouping genetic algorithm for clustering problems," *Expert Syst. Appl.*, vol. 39, no. 10, pp. 9695–9703, Aug. 2012.

[81] S. Salcedo-Sanz, L. Carro-Calvo, A. Portilla-Figueras, L. Cuadra, and D. Camacho, "Fuzzy clustering with grouping genetic algorithms," in *Proc. Int. Conf. Intell. Data Eng. Automated Learn.* Berlin, Germany: Springer, Oct. 2013, pp. 334–341.
[82] C. Raposo, C. H. Antunes, and J. P. Barreto, "Automatic clustering using a genetic algorithm with new solution encoding and operators," in *Proc. Int. Conf. Comput. Sci. Appl.* Cham, Switzerland: Springer, Jun. 2014, pp. 92–103.
[83] Z. Aliniya and S. A. Mirroshandel, "A novel combinatorial merge-split approach for automatic clustering using imperialist competitive algorithm," *Expert Syst. Appl.*, vol. 117, pp. 243–266, Mar. 2019.
[84] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2016, pp. 478–487.
[85] J. Yang, D. Parikh, and D. Batra, "Joint unsupervised learning of deep representations and image clusters," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5147–5156.
[86] Y.-C. Hsu and Z. Kira, "Neural network-based clustering using pairwise constraints," 2015, *arXiv:1511.06321*. [Online]. Available: http://arxiv.org/abs/1511.06321
[87] P. Huang, Y. Huang, W. Wang, and L. Wang, "Deep embedding network for clustering," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 1532–1537.
[88] G. Chen, "Deep learning with nonparametric clustering," 2015, *arXiv:1501.03084*. [Online]. Available: http://arxiv.org/abs/1501.03084
[89] Y. Lukic, C. Vogt, O. Durr, and T. Stadelmann, "Speaker identification and clustering using convolutional neural networks," in *Proc. IEEE 26th Int. Workshop Mach. Learn. for Signal Process. (MLSP)*, Sep. 2016, pp. 1–6.
[90] K. G. Dizaji, A. Herandi, C. Deng, W. Cai, and H. Huang, "Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 5736–5745.
[91] V. Rajah and A. E. Ezugwu, "Hybrid symbiotic organism search algorithms for automatic data clustering," in *Proc. Conf. Inf. Commun. Technol. Soc. (ICTAS)*, Mar. 2020, pp. 1–9.

ABSALOM EL-SHAMIR EZUGWU (Member, IEEE) received the B.Sc. degree in mathematics (computer science) and the M.Sc. and Ph.D. degrees in computer science from Ahmadu Bello University, Zaria, Nigeria. He is currently a Senior Lecturer with the School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal, Durban, South Africa. He has published articles relevant to his research interest in internationally refereed journals and edited books, conference proceedings, and local journals. His main research interests include parallel algorithms design in cloud and grid computing environments, artificial intelligence with a specific interest in computational intelligence, and metaheuristic solutions to real-world global optimization problems. He is a member of IAENG and ORSSA.

MOYINOLUWA B. AGBAJE received the B.Sc. degree (Hons.) in computer science from the Federal University of Technology, Akure, Nigeria, in 2015. She is currently pursuing the master's degree (research) with the University of KwaZulu-Natal, Durban, South Africa. Her research interests include deep learning, machine learning, big data analytics, artificial intelligence, computer vision, and nature inspired optimization.

NAHLA ALJOJO received the bachelor's degree in computer science from King Abdulaziz University, Jeddah, Saudi Arabia, the master's degree in computer system and information technology from Washington International University, Prussia, PA, USA, and the Ph.D. degree in computing from the University of Portsmouth. She was an Associate Professor with the Information System Department, Faculty of Computing and Information Technology, King Abdulaziz University. She is currently an Associate Professor with the Information Systems Department, Faculty of Computing and Information Technology, University of Jeddah, Jeddah. Her research interests include adaptivity in web-based educational systems, e-business, leadership's studies, information security, e-learning, and education.

ROSANNE ELS received the M.Sc. degree in computer science and the P.G.C.E. degree from the University of KwaZulu-Natal, Pietermaritzburg, South Africa, in 1997 and 2008, respectively. She is currently a Lecturer of computer science with the School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal. Her research interests include parallel computing, deep learning, machine learning, data mining, artificial intelligence, computer science education, and nature-inspired optimization.

HARUNA CHIROMA (Member, IEEE) received the B.Tech. degree in computer science from Abubakar Tafawa Balewa University, Nigeria, the M.Sc. degree in computer science from Bayero University, Kano, Nigeria, and the Ph.D. degree in artificial intelligence from the University of Malaya, Malaysia. He is currently an Associate Professor with the National Yunlin University of Science and Technology, Douliu, Taiwan. He has published articles relevant to his research interest in international refereed journals, edited books, conference proceedings, and local journals. His main research interests include metaheuristic algorithms in energy modeling, decision support systems, data mining, machine learning, soft computing, human-computer interaction, social media in education, computer communications, software engineering, and information security. He is a member of ACM, NCS, INNS, and IAENG. He serves on the Technical Program Committee of several international conferences.

MOHAMED ABD ELAZIZ received the B.S. and M.S. degrees in computer science and the Ph.D. degree in mathematics and computer science from Zagazig University, Egypt, in 2008, 2011, and 2014, respectively. Since 2014, he has been a Lecturer with the Mathematical Department, Zagazig University, where he is currently an Associate Professor of computer science. He is the author of more than 70 articles. His research interests include machine learning, signal processing, and image processing.