Ant-based sorting and ACO-based clustering approaches: A review

Ayad Mohammed Jabbar[1], Ku Ruhana Ku-Mahamud[2], Rafid Sagban[3]
[1]College of Arts and Sciences, Universiti Utara Malaysia, Kedah, Malaysia
[2] Data Science Research Lab, School of Computing, Universiti Utara Malaysia, Kedah, Malaysia
[3] Computer Science Department, University of Babylon, Babylon, Iraq
E-mail: ayad_mohammed@ahsgs.uum.edu.my, ruhana@uum.edu.my, rsagban@uobabylon.edu.iq

***Abstract*— Data clustering is used in a number of fields including statistics, bioinformatics, machine learning exploratory data analysis, image segmentation, security, medical image analysis, web handling and mathematical programming. Its role is to group data into clusters with high similarity within clusters and with high dissimilarity between clusters. This paper reviews the problems that affect clustering performance for deterministic clustering and stochastic clustering approaches. In deterministic clustering, the problems are caused by sensitivity to the number of provided clusters. In stochastic clustering, problems are caused either by the absence of an optimal number of clusters or by the projection of data. The review is focused on ant-based sorting and ACO-based clustering which have problems of slow convergence, un-robust results and local optima solution. The results from this review can be used as a guide for researchers working in the area of data clustering as it shows the strengths and weaknesses of using both clustering approaches.**

Keywords*— *Data mining; Data clustering; Swarm intelligence; Optimization based-clustering; Ant Colony Optimization.

I. INTRODUCTION

Clustering is an unsupervised learning approach to discover homogeneous groups [1]. In general, clustering approaches are classified into stochastic and deterministic clustering approaches. The differences between both approaches are broad. However, they also share some common characteristics and problems that need to be solved. Valuable characteristics from both approaches can be used to produce a powerful clustering approach [2]. The deterministic approach can be classified into hierarchical and partitional approaches [3]. Partitional clustering includes density-based clustering, grid-based clustering and model-based clustering, whereas a hierarchical clustering approach can be classified into divisive and agglomerative approaches. The stochastic approach employs metaheuristic algorithms to produce enhanced solutions for optimization. However, the aim of any clustering approach is to construct clusters that are compact, connected and separated. Compactness is related to intra-cluster similarity. Intra-cluster similarity determines the degree of similarity among data objects in each cluster. Compactness has two main measurements, namely point-based and edge-based measures. The representative measures the similarity in each cluster based on the distance between each point within a cluster and its centroid. A spherical cluster is produced when a number of clusters is given a priori [4, 5]. The edge-based measurement employs distances between pairwise points within a single cluster, including pairwise distance total summation and connected graph maximum length. This measurement is suitable in clustering datasets formed in different densities. Connectivity measures the relation between points based on point neighbors. Connectivity assumes those points are related to the same neighbors of points, so each point is assigned to a neighbor of the same cluster. Separation presents dissimilarity between each cluster as inter-cluster dissimilarity, which indicates the amount of dissimilarity for each cluster regarding objects within each single cluster. The total inter-cluster shows how much the clusters are different and located far from each other. However, using a deterministic approach or stochastic approach in achieving compactness, connectedness and well-separated clusters means producing clusters that contain objects that are similar to each other within clusters and clusters that are separated sufficiently. Thus, high-quality clusters can be obtained.

II. CLUSTERING PROBLEMS AND PERFORMANCE

Determining the optimal number of clusters plays an essential role for the performance of clustering. Clustering approaches, such as deterministic approaches, require a user interaction and corporation of prior knowledge about the shape and number of clusters [6]. The deterministic algorithm experiences difficulties in obtaining a global optimal solution due to various reasons including random initial centroid selection, wrong number of clusters provided by a user and high-dimensional space problem [7]. The selected initial centroids and number of clusters are sensitive, thereby results are un-robust because they initialize different numbers of clustering and different initial centroids lead to different results [8]. However, in deterministic approaches, getting an optimal number of clusters is a general problem in every clustering approach. For example, partition clustering approaches are sensitive and require the number of clusters as a predefined value [9, 10]. In density-based clustering approaches and model-based clustering, an optimal number of clusters is a critical value produced via algorithm parameter selection [11]. In grid-based clustering, the size of the grid affects the number of clusters produced. Hierarchical clustering does not require a prior number of clusters, which is the drawback of partition clustering, but it fails to determine the final cut of hierarchical trees, which produce the optimal number of clusters [12]. Table I summarizes the performance of the clustering technique. It can be seen that more advantages can be obtained using hierarchical clustering techniques and optimization-based clustering approaches, which can lead to good performance, while other techniques should be given less priority because of their disadvantages.

TABLE I. PERFORMANCE OF CLUSTERING TECHNIQUE

Cluster technique		Advantage	Disadvantage
Deterministic approach	Partitional clustering	No computational complexity time.	Sensitive and requires a predefined number of clusters & initial centroids.
	Density-based clustering	i) Does not require the number of cluster as a predefined value. ii) Able to detect outliers and identify clustering shape.	i) Sensitive to parameters selection. ii) Computational complexity time is high iii) Sensitive to dataset that contains different levels of density.
	Model-based clustering	i) Does not require a number of clusters as predefined values. ii) Small computational complexity time.	i) Curse of dimensionality problem. Ii) Sensitive to parameter selection.
	Hierarchical clustering	i) Suitable for text clustering. ii) Does not require the number of clusters as predefined values.	i) High dimensionality problem. ii) Does not provide best partition in hierarchical clustering tree.
	Grid-based clustering	Clustering performance is based on grid size.	Computational complexity time is based on grid size
Stochastic approach	Optimization-based clustering	i) No need of a priori information. ii) Includes strategies to avoid the local optima problem. iii) Deals efficiently with high dimensionality problem.	i) Results based on projection of data. ii) Results based on objective function.

Regardless of the placement of the problem in the deterministic approach, the stochastic approach via optimization-based clustering is relatively different. It induces the power of optimization to work with any dataset with no a priori information. The partition of clustering is conducted only if the clustering criterion is optimized. Experimental results of this approach are generally better than deterministic approaches. It includes strategies to avoid the local optima problem and deals efficiently with high dimensionality [13]. The optimization approach performs better than the deterministic approach because the clustering task is considered as an optimization problem where an optimal solution is searched. The objective function is to group similar objects and achieve high similarity among objects within a cluster and low similarity between clusters. The optimization approaches are classified into three: exact, estimation, and approximation [13]. The exact algorithm can produce an optimal solution to any optimization problem within an instance depending on the run time. Unfortunately, the exact algorithms require exponential time, especially with hard optimization problems (NP-hard) [14]. The estimation approach does not guarantee an optimal solution because the approach produces results based on a predefined range of inputs. The approximate algorithms may not find an optimal solution but, in practice, they are often able to find good solutions, albeit possibly suboptimal, in a relatively short time. The approximate algorithms can be classified as either heuristics algorithms or metaheuristics algorithms [14]. The heuristics algorithm rather than classic method is preferable for solving optimization problems because it takes a shorter time [15]. It is applied when the exact method is too slow and when the optimal solution is too difficult to obtain. The heuristics algorithm depends on the problem by applying exhaustive search that evaluates all possible solutions. The metaheuristics algorithm is a generalized, high-level problem with independent methods that can be applied to a wide range of problems. It is a framework used to control the heuristics method [16]. In the literature of metaheuristics, different ways are used to classify and describe the algorithms [16, 17]. Researchers classify them as nature- versus non-nature-inspired, population-based versus single point search, dynamic versus static objective function, one versus various neighborhood structures and memory usage versus memory-less methods [17]. The major difference between them is that the single solution, such as tabu search and simulated annealing (SA), produces results based on modifying and improving a single candidate solution, whereas the population-based solution, such as swarm-based algorithm and evolutionary algorithm, are frequently used to improve multiple candidate solutions. The advantages of these choices are their suitability and robustness for large problems [18]. Exploration and exploitation are the key algorithmic components of the metaheuristic algorithm [19, 20]. Exploration is a global search space that produces diverse solutions while exploitation generates information based on the regions exploited in the search on the local region [21]. However, any optimization approach can successfully solve any problem when the balance between the two components is optimal [22, 23]. Optimization-based clustering, applied successfully to solve the clustering problem in different optimization problems, includes Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC), Simulated Annealing (SA), Firefly (FA) and Ant Colony Optimization (ACO); hybridized with external heuristic algorithms includes K-means, agglomerative and bisect K-means algorithm. This integration carries the advantage of providing information about the clusters, such as the number of clusters and cluster shape.

In the literature, the number of clusters is determined automatically using a predefined range of clusters by employing an estimation approach [19]. PSO is integrated with K-means to improve the clustering performance. PSO searches the optimal number of clusters in a range of clusters and, later, K-means adjusts the generated centroids of each particle. The proposed algorithm outperforms the deterministic algorithm in terms of accuracy, cost and time but is highly time consuming.

Perim et al. [24] presents two initialization methods which improves the SA search algorithm by finding close optimal solutions. Deterministic and stochastic procedures, namely PCA-Part and K-means++, are hybrids of the SA algorithm. These methods are used for choosing the initial solution of SA. PCA-Part generates good results and K-means++ produces solutions close to the global optima. PCA-Part is a hierarchical approach based on the PCA technique minimizing the sum square error (SSE) between the elements in each iteration. K-means++ randomly select elements as the initial centroid and iteratively select the other centroids based on the SSE between elements. The proposed method performs better than K-means and the classical SA algorithm. Phanendra et al. [25] use SA to select optimal initial seeds as inputs of the K-means algorithm instead of randomly selecting initial seeds. This hybrid method outperforms the standard k-means algorithm.

Pelleg et al. [26] enhance K-means by estimating a range of clusters instead of a fixed number. The algorithm starts with the lower estimated range value and keeps tracking until reaching the upper estimated range value. During this process, the algorithm records the best centroids based on the obtained score. The results show an improvement in terms of quality compared with the standard K-means algorithm. Rahman et al. [27] propose a hybrid algorithm comprising a genetic algorithm (GA) and K-means algorithm. This hybrid can generate high-quality clustering solutions and prevent the local minima issue. GA automatically finds the right number of clusters through the clustering process and then obtains centers used as initial seeds in the K-means algorithm. The proposed algorithm utilizes an initial population selection technique systematically to select chromosomes in a deterministic and random way. The idea of a deterministic and random technique increases the exploration of ultimately good quality genes. The experiments show better results when compared to standard K-means algorithm and GA algorithm.

The major problems in the deterministic approach and hybrid algorithm are bad random initial centroid, selecting the right number of clusters and high-dimensional data, which cause slow convergence and local optima solutions. The hybrid algorithm produces an enhanced solution, but is constrained by the deterministic clustering approach. Thus, the optimization-based clustering approach is a unique way to produce better solutions.

III. Ant Algorithms And Their Application To Clustering

Swarm Intelligence (SI) is the collective behavior of natural or artificial decentralized, self-organization, which aims to utilize the collective behavior of social insects in solving optimization problems [28, 29]. Two main approaches, classified as ant-based sorting and ACO-based clustering, exhibit major differences in dealing with the clustering problem [30]. Ant-based sorting is inspired by the clustering of cemetery, larvae sorting [31] and colonial odor [32]. ACO-based clustering, inspired by the foraging behavior of real ants looking for food [33, 34], has been presented based on different natural behaviors. The first is based on concept aggregations which reflect the ability of ants to aggregate objects and aggregate themselves. This is the main key in this clustering approach. This concept has been investigated in *Lasius niger* ants by Chrétien [35], *Pheidole pallidula* ants by Deneubourg et al. [31] and brood sorting by Franks and Sendova [36]. This concept is later developed as a model presented in the form of the population of ants scattered into a 2D grid to perform clustering by picking up or dropping off objects in different regions. This model represents the original Standard Ant Clustering Algorithm (SACA). The typology is then extended to another model inspired by the aggregation of ants towards safer areas, where ants depart their locations searching for safer regions, which reflects self-aggregation. The safe region is an area which contains ants that share high amounts of similarities. Algorithms classified under this type are called self-aggregation within a 2D grid. Further extension of self-aggregation is inspired from an important concept known as stigmergy. This concept is modeled as an aggregation of pheromones that cause the conglomeration of ants, or clustering behavior. The stigmergy adopted in SACA reflects when ants move from one location to another based on changes in pheromones from its current cell towards the most likely cell it can travel to. The algorithm is classified under the concept known as ant aggregation through pheromone in a 2D grid. The final model was inspired from the idea of colonial or “Gestalt” odor theory. The ants can determine an ant nest based on odor of the colony. Odor colony membership is the key to the clustering process.

The ACO-based clustering models mentioned above are inspired from the foraging behavior of real ants looking for food. Algorithms that belong to this category require the ants to deposit an amount of pheromone on the graph structure which represents the problem that needs to be clustered. It encourages other ants to increasingly choose edges with higher pheromones. This category includes several approaches such as single objective, multi-objective, multi-colony and multi-pheromone implementations using ACO. The problem of clustering can be solved based on two predominant approaches classified as either an optimal assignment problem or clustering within a graph problem.

IV. Ant-Based Sorting Versus ACO-Based Clustering Approaches

The main difference between both approaches is that the ACO-based clustering treats the clustering problem as an optimization task, whereas in ant-based sorting is defined as an implicit objective function [3]. Ant-based sorting may result in too many clusters as objects may be left alone in the 2D grid and objects can still be carried by ants when the algorithm run-time stops [37]. Therefore, other algorithms can be combined to minimize categorization errors. In ACO-based clustering, an ant contacts an indirect connection that helps the ants communicate with one another [38]. Communication is undirected and performed through pheromone substance [39]. However, both approaches suffer from drawbacks which include slow convergence, premature convergence and rough clustering with the risk of staying in local optima. Table II depicts the comparison of both approaches.

TABLE II. ACO-BASED CLUSTERING VERSUS ANT-BASED SORTING

ACO-based clustering	Ant-based sorting
Clustering is explicitly defined	Clustering is implicitly defined
Pheromone is an essential component	Pre-processing is essential and highly required
Low time consumption	High time consumption
Post processing is not required	Post-processing is highly required
Most algorithms require pre-defined knowledge (for optimal assignment problems)	Most algorithms require pre-defined knowledge
Clustering is based on foraging behavior	Clustering is based on several behaviors which include cemetery, larvae sorting and colonial odor
Ant movement is probabilistic	Ant movement is random.
Less sensitivity to its parameter selection	Very sensitive to its parameter selection
Free data is forbidden	Produces free data
Clustering problem is modelled as graph structure	Clustering problem is modelled as 2D grid structure or as chains of objects

In both approaches, clustering is performed based on different representative including data point-to-cluster assignment, cluster representatives, direct point-agent matching and search agent. In data point-to-cluster assignments, each agent has knowledge about the number of clusters and assigns each data object based on similarity to its appropriate cluster. This representative use in both approaches when the problem of clustering is considered as an optimal assignment problem. The number of clusters is known while the centroids are iteratively adjusted until optimization of the centroid (central point) is reached. In cluster representation, the agent is represented as the center of each cluster. In direct point-agent matching, each agent corresponds to one data object. This repressive method is commonly used in ant-based sorting, where the number of agents is equal to the number of objects. Thus, the algorithm achieves worse performance when the size of the dataset is high. In search agent, there is no matching between data objects and agent. Thus, the number of agents does not necessarily equal the number of objects. Each agent searches in the search space to construct a clustering solution based on the probabilistic model. ACO-based clustering involves a search agent while ant-based sorting can represent the other reprehensive. The summary of the representations is indexed according to years as depicted in Figures 1 and 2.

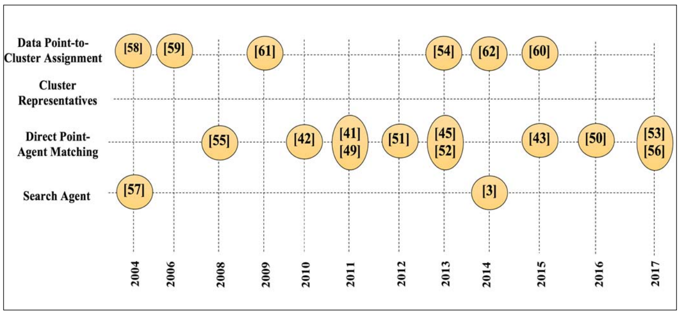

Fig. 1. Distribution of agent representation research over a time period

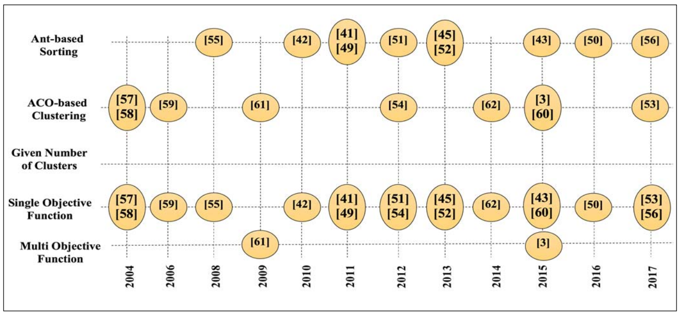

Fig. 2. Distribution of ant-based sorting versus ACO-based clustering research over a time period

Ant-based sorting performs clustering on a 2D grid structure containing sets of objects. Those objects are randomly projected into the grid [40].This random projection is a major impediment causing low similarity, multiple objects which belong to the wrong cluster and production of excessive clusters. These disadvantages have led researchers to combination with other algorithms. On contact, ACO-based clustering performs clustering with no attention to the way objects' projection is formed. Thus, its combination with other algorithms is not fundamental but may improve results. Initial centroid selection of ant-based sorting is critical depending upon the results of the projection of objects, where the similarity of objects in the local neighborhood is low, leading to difficulty in selecting or discovering the best initial centroid. This situation affects the efficiency of the algorithm and convergence speed, especially when similar objects are located in different clusters [41]. Occasionally, imbalanced dataset classifications may occur where similar data objects are projected into different areas on the grid [42, 43]. This makes the algorithm unable to compile data correctly and in a timely manner, thus an additional pre-processing step increases the similarity between objects before performing clustering.

Kernel principal component analysis (KPCA) is employed to minimize the degree of randomness [41]. KPCA extracts efficient features from a dataset to be linearly modelled. Extracted features will then perform in the feature space using ant-based sorting. Objects that are close in the feature space are also close in the projection plane. Therefore, objects in the local neighborhood become highly similar and running time is significantly reduced. However, KPCA suffers from robustness when outliers are presence in a dataset [44]. A similar technique, based on kernel entropy component analysis (KECA) [45], was employed to deal with the problem of initial centroid selection. KECA, in the initial projection stage, can modify random projections by extracting features with a distinct angular structure. It can reveal cluster structure and information about the underlying labels of the data [46]. The similarity measure is calculated as previous work by the kernel function based on Euclidean distance. Both KECA and KPCA methods produce high-dimensional data especially on large datasets. Memory issues and imbalance classification problems are notably difficult in these methods [47]. In recent years, data have progressively increased. DBSCAN [48] is an algorithm of density-based clustering suffering from high dimensionality problems when datasets contain different levels of densities. DBSCAN has poor memory and is unable to handle clusters with different shapes when poor parameters are selected. To solve the problem of high dimensionality, partitioning-based DBSCAN algorithm (PDBSCAN) is proposed for a large dataset. In this algorithm, the dataset is partitioned into small partitions. Each small part is processed by the DBSCAN algorithm and then merged to one dataset. However, PDBSCAN remains sensitive to data because it is processed based on the results of partitioning. Jiang et al. [49] propose a hybrid algorithm to solve the problem of sensitive initial parameters in both DBSCAN and PDBSCAN. This hybrid algorithm performs clustering based on dimensional size. The proposed algorithm employs two more algorithms, namely point density (PD) and modified ant-based sorting partitioning algorithm (PACA). For 2D data, the algorithm uses PD to partition data directly based on the k-means algorithm. In multi-dimensional data, the algorithm will partition data with the PACA algorithm. PD is used to partition the dataset based on the value of density using the k-means algorithm. The PACA algorithm deals with multi-dimensional data based on hybrid PD with ant-based sorting. The ant-based sorting algorithm is used to present multi-dimensional data on a 2D grid. PD is then employed to calculate and partition the objects. However, using a k-means algorithm with high-dimension data results may cause a local optima solution.

Subhadra et al. [43] propose a post-pruning step on results produced by ant-based sorting after a number of iterations. Using post-pruning finds some non-member objects located in different, wrong, clusters. Determining those non-members and redistributing them will produce better results and less running time. However, high-dimension is the main drawback with this proposed algorithm. Gao [50] proposes a data combination mechanism to improve ant movement in ant-based sorting. The results are more robust compared to standard ant-based sorting.

An efficient splitting rule is proposed to the redistributed data object after a number of iterations. Those objects occupy dropping regions for other members and are surrounded by other members, thus ant-based sorting addresses those objects [42]. Another technique uses parallelization to solve the problem of high dimensionality with Hadoop technology [51]. The map function splits a dataset into a set of groups, and the results are produced by a reduced function. However, the main criterion to measure the quality of clustering is not discussed in detail in this work. The said methodology does not explain how a dataset is separated.

Hybrid ant-based sorting with k-means algorithm is proposed to improve ant-based sorting clustering results. Ant-based sorting produces a clustering process with free data. The free data is individual data located in a 2D grid without processing by the algorithm. K-means algorithm, used for post-processing, minimizes the error iteratively. [52]. However, ant-based sorting produces random movements which may result in incorrect initial centroids. Thus the k-means algorithm also produces local optima solutions. Similar research proposes a hybrid method to improve the convergence of ant-based sorting. The presented algorithm combines the characteristics of ant-based sorting which are stochastically and exploratory and k-means algorithms [53]. Modification is done by frequently applying both algorithms in sequence stages with mandatory conditions. The condition controls the movement of ants on the grid. The experimental results indicate that the presented algorithm outperforms other clustering methods including K-means, PSO and ant-class algorithms. Another piece of research proposes a new algorithm called (FCACA) with a new modification of the original ant-based clustering objective function. The modification improves the convergence algorithm [54]. The evaluated algorithm showed less complexity than the original SACA algorithm.

Ghosh et al. [55] present an aggregation of pheromone density. Ant movements, governed by the deposited pheromone intensity, are aggregated at different points of the search space by other ants. The movement of ants will form homogenous groups. Although this method achieved good results compared to others, it produced more than the desired number of clusters. Another similar study improved ant movements based on intelligent behavior [56]. The presented algorithm called (FlyAntClass) contains additional behaviors inspired from birds and spiders to control ant movements.

Ant colony optimization with different favor (ACODF) is the first ACO-based clustering algorithm established by Tsai and his colleagues [57]. Several strategies applied include: differently favorable ants to solve the clustering problem, the algorithm adopts the SA algorithm concept for decreasing the number of ants iteratively and increases exploration using a tournament selection technique. Shelokar et al. [58] were pioneers in introducing an ACO-based clustering algorithm as presented in the original Dorigo framework to perform clustering as an optimization problem. The algorithm mainly relies on pheromone trails only as guides for ants to construct clustering based on a pre-defined number of clusters provided by the user. However, the local optima solution easily occurs because the algorithm depends on pheromones only. Kao and Cheng extend the Shelokar algorithm by introducing dynamic cluster centers, namely ACOC [59]. The ACOC algorithm mainly relies on pheromone information and heuristic information in constructing a clustering solution instead of only using pheromone information. However, the algorithm is better than the Shelokar algorithm in terms of accuracy but involves longer computation time than k-means algorithms as well as needing a given number of clusters. Thus, convergence is premature, and the algorithm is easily trapped in local optima. Another extended study proposed a new algorithm combining the ACOC algorithm with spectral Laplacian called (SACOC). The new algorithm generates a new search space, producing more robust results [60]. The algorithm showed better results when evaluated against ACOC and k-means algorithms. However, the algorithm requires pre-defined numbers of clusters and suffers from memory consumption and running time.

ACO-based clustering employs the multi-objective function in clustering problems [61]. The ant algorithm, which consists of two colonies, works in parallel and contains one objective function in each colony. The number of clusters must be input by the user, as in the Shelokar and ACOC algorithm. Evaluation of this proposed algorithm was not comprehensive. Furthermore, time complexity is high. A study on ACO-based clustering using the multi-objective function in clustering, without requiring external knowledge about the dataset, number of clusters and density of clusters, was performed [3]. Adjusted compactness and relative separation are two different objective functions employed in this study. However, the proposed work has shortcomings in terms of time complexity and handling noise data. The Medoid-based ACO Clustering Algorithm (MACOC) which is an extension of the ACOC algorithm is proposed. MACOC is medoid-based instead of centroid-based which improves the algorithm to be more robust in the presence of noise [62]. The algorithm exceeds ACOC algorithm performance but suffers from sensitivity for a pre-defined number of clusters.

To the best of our knowledge, no study has compared the performance of all ant-based clustering approaches. Comparison that has been performed and reported is based on individual work with one or two clustering approaches [58]. Several works consider the Shelokar dataset as a benchmark dataset in conducting their studies on ACO-based clustering [59, 60– 62].

V. Conclusion

This paper has presented a review of research on clustering which includes a deterministic and stochastic approach. The review highlights the two classes of swarm-based clustering algorithm: ant-based sorting and ACO-based clustering. Studies have been focused on improvement of the algorithms and issues that have been studied include slow convergence, local optima and low similarity, explored as limitations in both approaches. The review concludes that limitations are caused by several problems which include random projection of data, bad initial centroids and pre-assigning of an inappropriate number of clusters. Studies in the suggested areas can push the boundaries of knowledge in achieving more optimal clustering results. Optimal clustering results can be determined by establishing an explicit objective function which considers the search space as clustering within a graph and not clustering as a search for optimal assignment, which is the major issue in the swarm-based clustering algorithm.

Acknowledgments

The authors thank the Ministry of Higher Education Malaysia for funding this study under Fundamental Research Grant Scheme (S/O code 13240).

References

[1] S. J. Nanda and G. Panda, “A survey on nature inspired metaheuristic algorithms for partitional clustering,” *Swarm Evol. Comput.*, vol. 16, pp. 1–18, 2014.

[2] C. Tsai, H. Wu, and C. Tsai, “A new data clustering approach for data mining in large databases,” *Proc. Int. Symp. Parallel Archit. Algorithms Networks. I-SPAN’02*, pp. 315–320, 2002.

[3] T. İnkaya, S. Kayalıgil, and N. E. Özdemirel, “Ant Colony Optimization based clustering methodology,” *Appl. Soft Comput.*, pp. 301–311, 2015.

[4] Y. Jung, H. Park, D. Z. Du, and B. L. Drake, “A decision criterion for the optimal number of clusters in hierarchical clustering,” *J. Glob. Optim.*, vol. 25, no. 1, pp. 91–111, 2003.

[5] M. Ester, H. Kriegel, X. Xu, and D.- Miinchen, “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise,” *Proc. 2nd Int. Conf. Knowl. Discov. Data Min.*, pp. 226–231, 1996.

[6] R. Guan, X. Shi, M. Marchese, C. Yang, and Y. Liang, “Text Clustering with Seeds Affinity Propagation,” *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 4, pp. 627–637, 2011.

[7] A. K. Jain, “Data clustering: 50 years beyond K-means,” *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, 2010.

[8] O. Tanaseichuk and A. H. Khodabakshi, “An Efficient Hierarchical Clustering Algorithm for Large Datasets,” *Austin J. Proteomics, Bioinforma. Genomics*, vol. 2, pp. 1–6, 2015.

[9] G. Tzortzis and A. Likas, “The MinMax k-Means clustering algorithm,” *Pattern Recognit.*, vol. 47, no. 7, pp. 2505–2516, 2014.

[10] B. K. Mishra, N. R. Nayak, A. Rath, and S. Swain, “Far Efficient K-Means Clustering Algorithm,” pp. 106–110, 2012.

[11] M. Haghir, H. Abolhassani, and M. Haghir, “Improving density-based methods for hierarchical clustering of web pages,” *Data Knowl. Eng.*, vol. 67, pp. 30–50, 2008.

[12] N. A. Mokhtar, Y. Z. Zubairi, and A. G. Hussin, “A clustering approach to detect multiple outliers in linear functional relationship model for circular data,” *J. Appl. Stat.*, pp. 1–11, 2017.

[13] S. Desale, A. Rasool, S. Andhale, and P. Rane, “Heuristic and Meta-Heuristic Algorithms and Their Relevance to the Real World: A Survey,” *Int. J. Comput. Eng. Res. Trends*, vol. 351, no. 5, pp. 2349–7084, 2015.

[14] M. Birattari, “Tuning Metaheuristics: A Machine Learning Perspective,” in *Tuning Metaheuristics: A Machine Learning Perspective*, Second edi., vol. 197, Berlin: Springer, 2009, p. 37.

[15] S. Desale, A. Rasool, S. Andhale, and P. Rane, “Heuristic and Meta-Heuristic Algorithms and Their Relevance to the Real World: A Survey,” *Int. J. Comput. Eng. Res. Trends*, vol. 351, no. 5, pp. 2349–7084, 2015.

[16] K. Sörensen, “Metaheuristics-the metaphor exposed,” *Int. Trans. Oper. Res.*, vol. 22, no. 1, pp. 3–18, 2015.

[17] C. Blum and A. Roli, “Metaheuristics in combinatorial optimization: overview and conceptual comparison,” *ACM Comput. Surv.*, vol. 35, no. 3, pp. 189–213, 2003.

[18] J. Handl and B. Meyer, “Ant-based and swarm-based clustering,” *Swarm Intell.*, vol. 1, no. 2, pp. 95–113, 2007.

[19] 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.

[20] J. G. J. March, “Exploration and exploitation in organizational learning,” *Organ. Sci.*, vol. 2, no. 1, pp. 71–87, 1991.

[21] A. M. Jabbar, “Controlling the Balance of Exploration and Exploitation in ACO Algorithm,” *J. Univ. Babylon*, 2018.

[22] R. Sagban, K. R. Ku-Mahamud, and M. S. A. Bakar, "Unified strategy for intensification and diversification balance in ACO metaheuristic," ICIT 2017 - 8th Int. Conf. Inf. Technol. Proc., pp. 139–143, 2017.

[23] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Inf. Sci. (Ny).*, vol. 237, pp. 82–117, 2013.

[24] G. T. Perim, E. D. Wandekokem, and F. M. Varejão, "K-means initialization methods for improving clustering by simulated annealing," in *n Ibero-American Conference on Artificial Intelligence*, 2008, pp. 133–142.

[25] N. M. Phanendra Babu, "Simulated Annealing for Optimal Initial Seed Selection in K- means Algorithm," *Indian J. Pure Appl. Math.*, pp. 85–94, 1994.

[26] D. Pelleg, D. Pelleg, A. W. Moore, and A. W. Moore, "X-means: Extending K-means with efficient estimation of the number of clusters," *Proc. Seventeenth Int. Conf. Mach. Learn. table contents*, pp. 727–734, 2000.

[27] M. A. Rahman and M. Z. Islam, "A hybrid clustering technique combining a novel genetic algorithm with K-Means," *Knowledge-Based Syst.*, vol. 71, no. August, pp. 345–365, 2014.

[28] U. Chandrasekhar and P. R. P. Naga, "Recent trends in Ant Colony Optimization and data clustering: A brief survey," *2011 2nd Int. Conf. Intell. Agent Multi-Agent Syst.*, pp. 32–36, 2011.

[29] C. Huang, W. Huang, H. Chang, C. Yeh, and C. Tsai, "Hybridization strategies for continuous ant colony optimization and particle swarm optimization applied to data clustering," *Appl. Soft Comput.*, vol. 13, no. 9, p. 38, 2013.

[30] J. Handl and B. Meyer, "Ant-based and swarm-based clustering," *Swarm Intell.*, vol. 1, no. 2, pp. 95–113, 2007.

[31] J. L. Deneubourg, S. Goss, N. Franks, A. Sendova-Franks, C. Detrain, and L. Chrétien, "The Dynamics of Collective Sorting: Robot-Like Ants and Ant-Like Robots," *From Anim. to Animat. 1st Int. Conf. Simul. Adapt. Behav.*, pp. 356–363, 1991.

[32] N. Labroche, N. Monmarché, and G. Venturini, "Antclust :Ant clustering and web usage mining," *Insectes Soc.*, pp. 25–36, 2003.

[33] M. Dorigo, "Optimization, Learning and Natural Algorithms (in Italian)," Politecnico di Milano, 1992.

[34] R. Sagban, K. R. Ku-Mahamud, and M. S. A. Bakar, "Reactive Max - Min Ant System: An Experimental Analysis of the Combination with K - Opt Local Searches," in Proceedings of 5th International Conference on Computing and Informatics, ICOCI 2015, 2015, pp. 300–305.

[35] Chrétien, "Organisation spatiale du mat´ eriel provenant de l'excavation du nid chez Messor barbarus et des cadavres d'ouvri´ eres chez Lasius niger (Hymenopterae: Formicidae)," 1996.

[36] Sendova and Franks, "Spatial relationships within nests of the antLeptothorax unifasciatus(Latr.) and their implications for the division of labour," *Anim. Behav.*, vol. 50, no. 1, pp. 121–136, 1995.

[37] Y. He, S. C. Hui, and Y. Sim, "A novel ant-based clustering approach for document clustering," *3rd Asia Inf. Retr. Symp. AIRS 2006 Oct. 16 2006 Oct. 18*, vol. 4182 LNCS, pp. 537–544, 2006.

[38] R. Sagban, K. R. Ku-Mahamud, and M. S. Abu Bakar, "Reactive max-min ant system with recursive local search and its application to TSP and QAP," *Intell. Autom. Soft Comput.*, vol. 23, no. 1, pp. 127–134, 2017.

[39] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 26, no. 1, pp. 29–41, 1996.

[40] U. Boryczka, "Finding groups in data: Cluster analysis with ants," *Appl. Soft Comput. J.*, vol. 9, no. 1, pp. 61–70, 2009.

[41] L. Zhang and Q. Cao, "A novel ant-based clustering algorithm using the kernel method," *Inf. Sci. (Ny).*, no. 20, pp. 4658–4672, 2011.

[42] W. Zhang, C. K. Chang, H. I. Yang, and H. Y. Jiang, "A hybrid approach to data clustering analysis with K-means and enhanced ant-based Template Mechanism," *Proc. - 2010 IEEE/WIC/ACM Int. Conf. Web Intell. WI 2010*, vol. 1, pp. 390–397, 2010.

[43] K. Subhadra, M. Shashi, and A. Das, "Extended ACO Based Document Clustering with hybrid Distance Metric," *Proc. 2015 IEEE Int. Conf. Electr. Comput. Commun. Technol.*, 2015.

[44] M. H. Nguyen and F. De Torre, "Robust Kernel Principal Component Analysis," *Adv. Neural Inf. Process. Syst.*, pp. 1185–1192, 2009.

[45] L. Zhang, Q. Cao, and J. Lee, "A novel ant-based clustering algorithm using Renyi entropy," *Appl. Soft Comput. J.*, vol. 13, no. 5, pp. 2643–2657, 2013.

[46] R. Jenssen, "Kernel Entropy Component Analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 847–860, 2010.

[47] T. Sato, "Kernel-based Principal Components Analysis on Large Telecommunication Data," *Proc. Eighth Australas. Data Min. Conf.*, pp. 109–115, 2009.

[48] X. Xu and J. Jäger, "A fast parallel clustering algorithm for large spatial databases," *High Perform. Data Min.*, pp. 263–290, 1999.

[49] H. Jiang, J. Li, S. Yi, X. Wang, and X. Hu, "A new hybrid method based on partitioning-based DBSCAN and ant clustering," *Expert Syst. Appl.*, vol. 38, no. 8, pp. 9373–9381, 2011.

[50] W. Gao, "Improved Ant Colony Clustering Algorithm and," *Hindawi Publ. Corp. Comput. Intell. Neurosci.*, 2016.

[51] Y. Yang, X. Ni, H. Wang, and Y. Zhao, "Parallel implementation of ant-based clustering algorithm based on hadoop," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, pp. 190–197, 2012.

[52] J. Lu and R. Hu, "A new hybrid clustering algorithm based on K-means and ant colony algorithm," *Proc. 2nd Int. Conf. Comput. Sci. Electron. Eng.*, pp. 1718–1721, 2013.

[53] A. Onan, H. Bulut, and S. Korukoglu, "An improved ant algorithm with LDA-based representation for text document clustering," *J. Inf. Sci.*, vol. 43, no. 2, pp. 275–292, 2017.

[54] P. Dziwiński, Ł. Bartczuk, and J. T. Starczewski, "Fully Controllable Ant Colony System for Text Data Clustering," in *Swarm and Evolutionary Computation*, 2012, pp. 199–205.

[55] A. Ghosh, A. Halder, M. Kothari, and S. Ghosh, "Aggregation pheromone density based data clustering," *Inf. Sci. (Ny).*, pp. 2816–2831, 2008.

[56] A. Hamdi, M. Slimane, N. Monmarche, and A. M. Alimi, "FlyAntClass: Intelligent move for ant based clustering algorithm," *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. AICCSA*, 2017.

[57] C. F. Tsai, C. W. Tsai, H. C. Wu, and T. Yang, "ACODF: a novel data clustering approach for data mining in large databases," *J. Syst. Softw.*, vol. 73, no. 1, pp. 133–145, 2004.

[58] P. S. Shelokar, V. K. Jayaraman, and B. D. Kulkarni, "An ant colony approach for clustering," *Anal. Chim. Acta*, no. 2, pp. 187–195, 2004.

[59] Y. Kao and K. Cheng, "An ACO-Based Clustering Algorithm," *ANTS Int. Work. Ant Colony Optim. Swarm Intell.*, vol. 4150/2006, pp. 340–347, 2006.

[60] H. D. Menéndez, F. E. B. Otero, and D. Camacho, "SACOC:A Spectral-Based ACO Clustering Algorithm," in *Intelligent Distributed Computing VIII*, 2015, pp. 185–194.

[61] D. S. Santos, D. de Oliveira, and A. L. Bazzan, "Data mining and multi-agent integration," *Data Min. Multi-Agent Integr.*, pp. 239–249, 2009.

[62] H. D. Menéndez, F. E. B. Otero, and D. Camacho, "MACOC: A Medoid-Based ACO Clustering Algorithm," in *Swarm Intelligence*, 2014, pp. 122–133.