



HAL
open science

Fuzzy Rules for Ant Based Clustering Algorithm

Amira Hamdi, Nicolas Monmarché, Mohamed Slimane, Adel Alimi

► **To cite this version:**

Amira Hamdi, Nicolas Monmarché, Mohamed Slimane, Adel Alimi. Fuzzy Rules for Ant Based Clustering Algorithm. *Advances in Fuzzy Systems, 2016, Lecture Notes in Networks and Systems, 2016, pp.1-16. 10.1155/2016/8198915 . hal-03587095*

HAL Id: hal-03587095

<https://hal.science/hal-03587095>

Submitted on 21 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Research Article

Fuzzy Rules for Ant Based Clustering Algorithm

Amira Hamdi,^{1,2} Nicolas Monmarché,² Mohamed Slimane,² and Adel M. Alimi¹

¹REGIM-Lab.: Research Groups in Intelligent Machines, University of Sfax, ENIS, BP 1173, 3038 Sfax, Tunisia

²Polytech Tours, University of Tours, Tours, France

Correspondence should be addressed to Amira Hamdi; amira.hamdi7@ieee.org

Received 8 July 2016; Accepted 8 September 2016

Academic Editor: Katsuhiko Honda

Copyright © 2016 Amira Hamdi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper provides a new intelligent technique for semisupervised data clustering problem that combines the Ant System (AS) algorithm with the fuzzy c -means (FCM) clustering algorithm. Our proposed approach, called F-ASClass algorithm, is a distributed algorithm inspired by foraging behavior observed in ant colony. The ability of ants to find the shortest path forms the basis of our proposed approach. In the first step, several colonies of cooperating entities, called artificial ants, are used to find shortest paths in a complete graph that we called graph-data. The number of colonies used in F-ASClass is equal to the number of clusters in dataset. Hence, the partition matrix of dataset founded by artificial ants is given in the second step, to the fuzzy c -means technique in order to assign unclassified objects generated in the first step. The proposed approach is tested on artificial and real datasets, and its performance is compared with those of K -means, K -medoid, and FCM algorithms. Experimental section shows that F-ASClass performs better according to the error rate classification, accuracy, and separation index.

1. Introduction

How do ants optimize food search? How do social spiders build communal nest? Why does a flock of birds fly in a v-shaped formation? How do termites build collectively their sophisticated nest structure? How do honey bee swarms cooperatively select their new nesting site? How does firefly flash its light in a wonderful pattern? How does a colony coordinate its behavior? How is it possible for social insects and animals to coordinate their actions and create complex patterns? How do such agents perform complex tasks without any direction and coordination between themselves? How agents in colony perform a work locally for global goal with sufficient flexibility as they are not controlled centrally? Collective behaviors in swarms of insects or animals have attached the attention of researches. They have proposed several intelligent models to solve a wide range of complex problems. This branch of artificial intelligence is addressed as swarm intelligence. The key components of swarm intelligence are self-organization, emergence, and stigmergy.

Self-organization is “a process whereby pattern at the global level of a system emerges solely from numerous interactions

among the lower-level components of the system. Moreover, the rules specifying interactions among the system's components are executed using only local information, without reference to the global pattern” [1]. In short it can be “a set of dynamical mechanisms whereby structures appear at the global level of a system from interactions of its lower-level components” [2].

Emergence seems to be the explication of what self-organizing systems produce. In this context the whole is not just the sum of its parts; it gets a surplus meaning that it is not captured by its part alone. The idea of emergence was firstly developed in [3] to explain indirect task coordination in the context of building behavior of termites. Grassé [3] showed that the coordination of building activities does not depend on the workers themselves but is mainly achieved by the nest structure.

The underlying idea of this paper is to propose a new approach to data clustering problem. We will show that the use of fuzzy logic combined with swarm intelligence technique yields robust results.

The remainder of the paper is organized as follows. In Section 2, we present an overview of data clustering problem

and swarm intelligence tools for data clustering. A general description of our proposed approach called F-ASClass is given in Section 3. Experimental results and comparison with hard and fuzzy algorithms are reported in Section 4. Finally concluding remarks are drawn in Section 5.

2. Literature Review

2.1. Problem Definition. Cluster analysis is a technique that organizes data by abstracting underlying structure either as a grouping of objects. Each group consists of objects that are similar between themselves and dissimilar to objects of other groups.

Each object corresponds to a vector of M numerical values which correspond to the M numerical attributes. The relationships between objects are generated into a dissimilarity matrix in which rows and columns correspond to objects. As objects can be represented by points in numerical space, the dissimilarity between two objects can be defined as distance between the two corresponding points. Any distance can be used as dissimilarity measures. The most commonly used dissimilarity matrix is the *Minkowski* metric:

$$d_r(O_i, O_j) = \sum_{k=1}^m (w_k |x_{ik} - x_{jk}|^r)^{1/r}, \quad (1)$$

where w_k is a weighting factor that will be set to 1 thereafter. According to the value of r ($r \geq 1$), the following measures are obtained: Manhattan distance ($r = 1$), Euclidean distance ($r = 2$), and *Chebyshev* distance ($r = \infty$). As mentioned in [4], Euclidean distance is the most common of the *Minkowski* metric.

2.2. Clustering Algorithms. The grouping step can be performed in a number of ways. In [5] different approaches to clustering data are described:

- (i) *Partitioning/Hierarchical Classification.* Partitional clustering technique identifies the partition that optimizes a clustering criterion defined on a subset of objects (locally) or over all of the objects (globally). Hierarchical clustering technique builds a sequence of nested partitions that are visualized, by example, by a dendrogram.
- (ii) *Hard/Fuzzy Classification.* A hard clustering algorithm allocates each object to a single cluster during its operations. Hence the clusters are disjoint. A fuzzy clustering algorithm associates each object with every cluster using a membership function. The output of such algorithm is a clustering but not a partition.
- (iii) *Deterministic/Stochastic.* Optimization in partitional approach can be accomplished using traditional technique or through a random search of the state space consisting of all possible labeling.
- (iv) *Supervised/Unsupervised Classification.* An unsupervised classification uses only the dissimilarity matrix. No information on the object class is provided to the

method (objects are said unlabeled). In supervised classification, objects are labeled while knowing their dissimilarities. The problem is then to construct hyperplanes separating objects according to their class. The unsupervised classification objective is different from that of the supervised case: in the first case, the goal is to discover groups of objects while in the second, known groups are considered and the goal is to discover what makes them different or to classify new objects whose class is unknown.

Our proposed technique presented in this paper, which we call F-ASClass, belongs to fuzzy-semisupervised partitional clustering technique. It uses fuzzy rules and stochastic behavior to partition dataset into specified number of clusters. For the present paper, it suffices to note that the following techniques (K -means, K -medoid, and FCM) are used to improve F-ASClass algorithm. A comparative study between us will be presented in Section 4.

K -means algorithm is a hard-unsupervised learning algorithm that appears to partition dataset into a specified number of clusters. The technique presented in [6] consists of starting with k groups, each of which consists of a single randomly selected object, and thereafter adding each new object to its closest cluster center. After an object is added to a group, the mean of that group is adjusted in order to take account of the new added object. The algorithm is deemed to have converged when the assignments no longer change.

The K -medoid algorithm described in [7] is based upon the search of representative objects of each cluster (called medoid), which should represent the various aspects of the structure of the data. K -medoid algorithm is related to the K -means algorithm. The main difference between K -means and K -medoid stands in calculating the cluster center. The medoid is a statistic which represents that data member of a dataset whose average dissimilarity to all the other members of the set is minimal. Therefore a medoid unlike mean is always a member of the dataset. It represents the most centrally located data item of the dataset.

The fuzzy c -means (FCM) algorithm firstly presented in [8] and improved in [9, 10] allows one samples to belong to two or more one clusters. The first idea which aims to characterize an individual objects' similarity to all the clusters was introduced in [11]. In this context, the similarity an object shares with each cluster is represented with a membership function whose values are between zero and one. Object in dataset will have a membership in every cluster; memberships close to unity indicate a high degree of similarity between the object and a cluster while memberships close to zero involve little similarity between the object and that cluster.

FCM method differs from previously presented K -means and K -medoid algorithms by the fact that the centroid of a cluster is the mean of all samples in the dataset, weighted by their degree of belonging to the cluster. The degree of belonging is presented by a function of the distance of the sample from the centroid, which includes a parameter

controlling for the highest weight given to the closest sample. All these techniques are sensitive to initial condition.

Another fuzzy classification model is studied in [12] which constructs the membership function on the basis of available statistical data by using an extension of the well-known contamination neighborhood. Reference [13] presents a new fuzzy technique using an adaptive network of fuzzy logic connectives to combine class boundaries generated by sets of discriminant functions in order to address the “curse of dimensionality” in data analysis and pattern recognition.

Reference [14] is intended to solve the problem of dependence of clustering results on the use of simple and predetermined geometrical models for clusters. In this context, the proposed algorithm computes a suited convex hull representing the cluster. It determines suitable membership functions and hence represents fuzzy clusters based on the adopted geometrical model that it is used during the fuzzy data partitioning within an online sequential procedure in order to calculate the membership function.

2.3. Swarm Intelligence Tools for Data Clustering Problem.

We start with an illustration of swarm intelligence tools that have been developed to solve clustering problems: Particle Swarm Optimization [15], Artificial Bee Colony [16], Firefly algorithm [17], Fish swarm algorithm [18], and Ant Colony Algorithm. In [19] the basic data mining terminologies are linked with some of the works using swarm intelligence techniques. A comprehensive review of the state-of-the-art ant based clustering methods can be found in [20].

The first model of ants’ sorting behavior has been done by Deneubourg et al. [21] where a population of ants are randomly moving in a 2-dimensional grid and are allowed to drop or load objects using simple local decision rules and without any central control. The general idea is that isolated items should be picked up and dropped at some other locations where more items of that type are present. Based on this existing work, Lumer and Faieta [22] have extended it to clustering data problems. The idea is to define dissimilarity between objects in the space of object attributes. Each ant remembers a small number of locations where it has successfully picked up an object. And so, when depositing a new item this memory is used in order to bias the direction in which the ant will move: ant tends to move towards the location where it last dropped a similar item. From these basic models, in [23] Monmarché has proposed an ant based clustering algorithm, namely, AntClass which introduces clustering in a population of artificial ants capable to carry heaps of objects. Furthermore, this ant algorithm is hybridized with the K -means algorithm. In [24], a number of modifications have been introduced on both LF and AntClass algorithm and authors have proposed AntClust, which is an ant based clustering algorithm for image segmentation. In AntClust, a rectangular grid was replaced by a discrete array of cells. Each pixel is placed in a cell and all cells of the array are connected to the nest of ants’ colony. Each ant performs a number of moves between its nest and the array and decides

with a probabilistic rule whether or not to drop its pixel. If the ant becomes free, it searches for a new pixel to pick up [24].

According to [25], another important real ant’s collective behavior, namely, the chemical recognition system of ants, was used to resolve an unsupervised clustering problem. In [26], Azzag et al. considers another biologically observed behavior in which ants are able to build mechanical structure thanks to a self-assembling behavior. This can be observed through the formation of drops constituted of ants or the building of chains by ants with their bodies in order to link leaves together. The main idea here is to consider that each ant represents a data and is initially placed on a fixed point, called the support, which corresponds to the root of the tree. The behavior of an ant consists of moving on already fixed ants to fix itself to a convenient location in the tree. This behavior is directed by the local structure of the tree and by the similarity between data represented by ants. When all ants are fixed in the tree, this hierarchical can be interpreted as a partitioning of the data [26].

Bird flocks and schools clearly display structural order and appear to move as single coherent entity [27]. In [28, 29], it has been demonstrated that flying animal can be used to solve data clustering problem. The main idea is to consider that individuals represent data to cluster and that they move following local behavior rule in a way; after few movements, homogeneous individual clusters appear and move together. In [30] Abraham et al. propose a novel fuzzy clustering algorithm, called MPSO, which is based on a deviant variety of the PSO algorithm.

Social phenomena also exists in the case of spiders: in the *Anelosimus eximius* case, individuals live together, share the same web, and cooperate in various activities such as collective web building: spiders are gathered in small clusters under the vegetal leaves included in the web and distributed on the whole silky structure. In [31], the environment models the natural vegetation and is implemented as a square grid in which each position corresponds to a stake. Stakes can be of different heights to model the environmental diversity of the vegetation. Spiders are always situated on top of stakes and behave according to three several independent items (a movement item, a silk fixing item, and a return to web item). This model was transposed to region detection in image.

In [32] Hamdi et al. propose a new swarm-based algorithm for clustering, based on the existing work of [21, 23, 28] which uses ants’ segregation behavior to group similar objects together; birds’ moving behavior to control next relative positions for a moving ant; and spiders’ homing behavior to manage ants’ movements with conflicting situations.

In [33] we proposed using the stochastic principles of ant colonies in conjunction with the geometric characteristics of the bee’s honeycomb. This algorithm was called AntBee and it was improved in [34]. In this context, we used fractal rules to improve the convergence of the algorithm.

Another example of Ant Clustering algorithm, called AntBee algorithm, is developed in [29]. The proposed approach uses the stochastic principles of ant colonies in conjunction with the geometric characteristics of the bee’s

honeycomb and the basic principles of stigmergy. An improved AntBee called FractAntBee was proposed [24] that incorporated main characteristics of fractal theory.

According to [35], a novel approach to image segmentation based on Ant Colony System (ACS) is proposed. In ACS algorithm an artificial ant colony is capable of solving the traveling salesman problem [36]. As in ACO for the TSP, in ACO-based algorithms for clustering, each ant tries to find a cost-minimizing path, where the nodes of the path are the data points to be clustered. Like in the TSP, the cost of moving from data point x_i to x_j is the distance $d(i, j)$ between these points, measured by some appropriate dissimilarity metric. Thus, the next point to be added to the path tends to be similar to the last point on the path. An important way in which these algorithms deviate from ACO algorithms is that the ants do not necessarily visit all data points [37].

3. Proposed Methodology

In ASClass algorithm, we have assumed that a graph $G = \{O_1, \dots, O_n\}$ of n object has been collected by the domain expert where each object is a vector of k numerical values v_1, \dots, v_k . For measuring the similarity between objects we will use in the following Euclidean distance between two vectors, denoted by D , which is used for edges in graph G [38]. The complete set of parameters of our model will be presented in Section 3.3.

Initially all the objects will be scattered randomly on the graph G ; each node in the graph represents an object O in the datasets. The edge that connects two objects in the graph-data represents a measure of dissimilarity between these objects in the database. A class is represented by a route connecting a set of objects. In ASClass we chose to use more than one colony of ants, the number of colonies needed here is equal to the number of classes in the database. Initially, for each colony, m artificial ants are placed on a selected object, called “nest-object.” For each cluster we randomly chose one and only one “nest-object.” The simulation model is detailed in Figure 1.

The objective is to find the shortest route between the given objects and return to nest-object, while keeping in mind that each object can be connected to the path only once. The path traced by the ant represents a cluster in the dataset. Figure 2 shows a possible result of ASClass algorithm execution on the graph of Figure 1. It may be noted that, at the end of algorithm, each colony gives a collective traced-path that represent a cluster in the partition.

For each colony, an artificial agent possesses three behavioral rules:

- (i) a movement item inspired by ant foraging behavior;
- (ii) an object fixing item inspired by the collective weaving in social spiders;
- (iii) a return to web item.

3.1. Movement Item. An artificial ant k is an agent which moves from an object i to an object j on a dataset graph. It

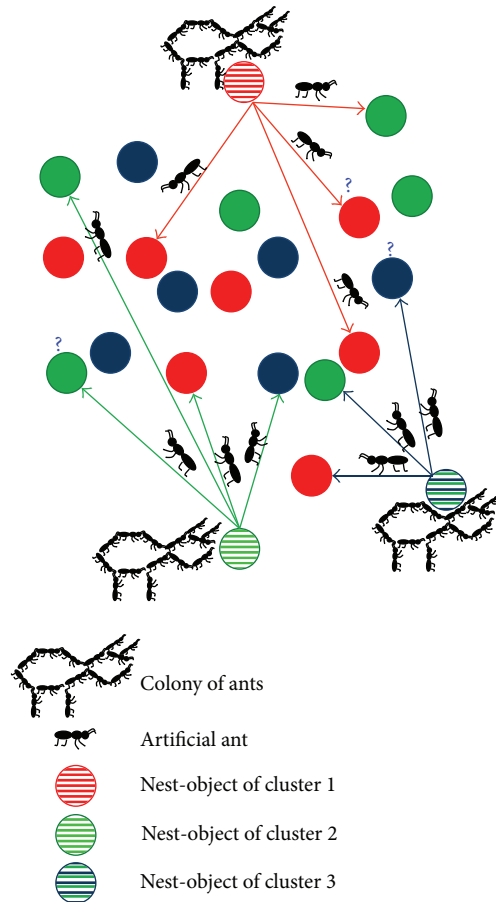


FIGURE 1: Graph used in ASClass algorithm [38].

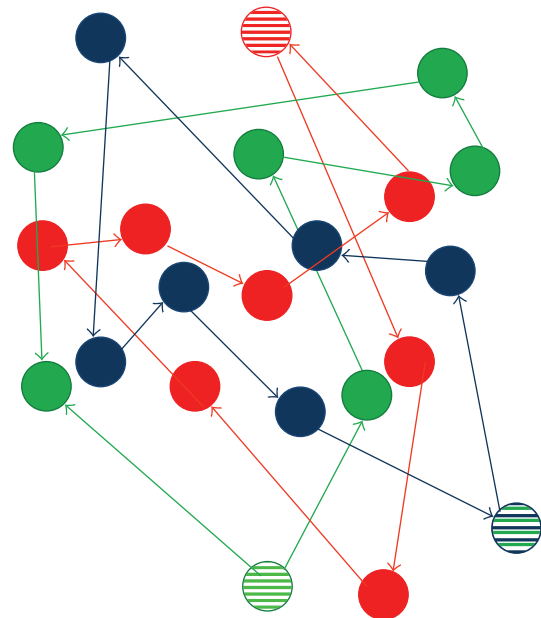


FIGURE 2: Results of ASClass algorithm on graph used in Figure 1: tree clusters are drawn; each of them is created by a colony of ants in the ASClass algorithm.

decides which object to reach among the objects it can access from its current position. The agent selects the object to move on according to a probability density P_{ij} , depending on the trail accumulated on edges and on the heuristic value, which was chosen here to be a dissimilarity measures.

$$P_{ij}^k(t) = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta} \quad \text{if } j \in N_i^k. \quad (2)$$

α and β are two parameters which determine the relative influence of the pheromone trail and the dissimilarity measures and N_i^k is the set of objects which ant k has not yet connected to its tour.

3.2. Object Fixing Item. When an ant reaches an object it can fix it on its path according to a contextual probability P_{fix} ; if a decision is made, ants draw a new edge between the current object and the last fixed object; otherwise, it returns to web and updates its memory (it decides to delete the object among the objects it can access from its current position).

The probability to fix the object to the path is defined as

$$P_{\text{fix}i}^k(t) = \begin{cases} 2f(O_i) & \text{if } f(O_i) < k_{\text{fix}} \\ 1 & \text{if otherwise,} \end{cases} \quad (3)$$

where k_{fix} is constant.

$f(O_i)$ is measure of the average similarity of the object O_i with the objects O_j forming the path created by the ant k and it is calculated as follows:

$$f(O_i) = \max \left\{ \frac{1}{N_{TK}} \sum_{O_j \in T_k} 1 - \frac{d(O_i, O_j)}{\theta}; 0 \right\}, \quad (4)$$

where θ is a scaling factor determining the extent to which the dissimilarity between two objects is taken into account and N_{TK} is the number of objects forming the tour constructed by the ant k .

At each time step, if an ant decides to fix new objects, it updates the pheromone trail on the arcs it has crossed in its tour. This is achieved by adding a quantity $\Delta\tau_{ij}$ to its arc and it is defined as follows:

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{1}{C_k} & \text{if edge } (i, j) \text{ belongs to } T^k \\ 0 & \text{otherwise,} \end{cases} \quad (5)$$

$$\tau_{ij} = \tau_0 + \sum_{k=1}^m \Delta\tau_{ij}^k(t) \quad \forall (i, j) \in T_k,$$

where C_k is the length of the tour T_k built by the ant k and τ_0 is constant.

After all ants have constructed their tours, the pheromone trails are lowered. This is done by the following rules:

$$\tau_{ij} = (1 - \rho) \tau_{ij}, \quad (6)$$

where $0 < \rho < 1$ and it is the pheromone trail evaporation. Equation (6) becomes

$$\tau_{ij} = \tau_0 + \sum_{k=1}^m \Delta\tau_{ij}^k(t) + (1 - \rho) \tau_{ij} \quad \forall (i, j) \in T_k. \quad (7)$$

3.3. Return to Web Item. If the decision is made, ant returns to the last fixed object and selects the object to move on according to its updated memory.

This process is achieved for each colony. At the end of process we obtain C routes which represent the C clusters in the datasets.

All notations used in ASClass algorithm are shown as follows:

$\tau_{ij}(t)$: the amount of pheromone on edge which connects city i to city j at time t .

α : parameter that control the relative importance of the trail intensity τ_{ij} .

β : parameter that control the visibility η_{ij} .

η_{ij} : the visibility.

ρ : the coefficient of decay.

N_{Tk} : the number of objects forming the tour constructed by the ant k .

J^k : set of objects which ant k has not yet connected to its tour.

$P_{ij}(t)$: transition probability from object i to object j .

θ : scaling factor.

Q : constant.

L_k : the length of the tour built by the k th ant.

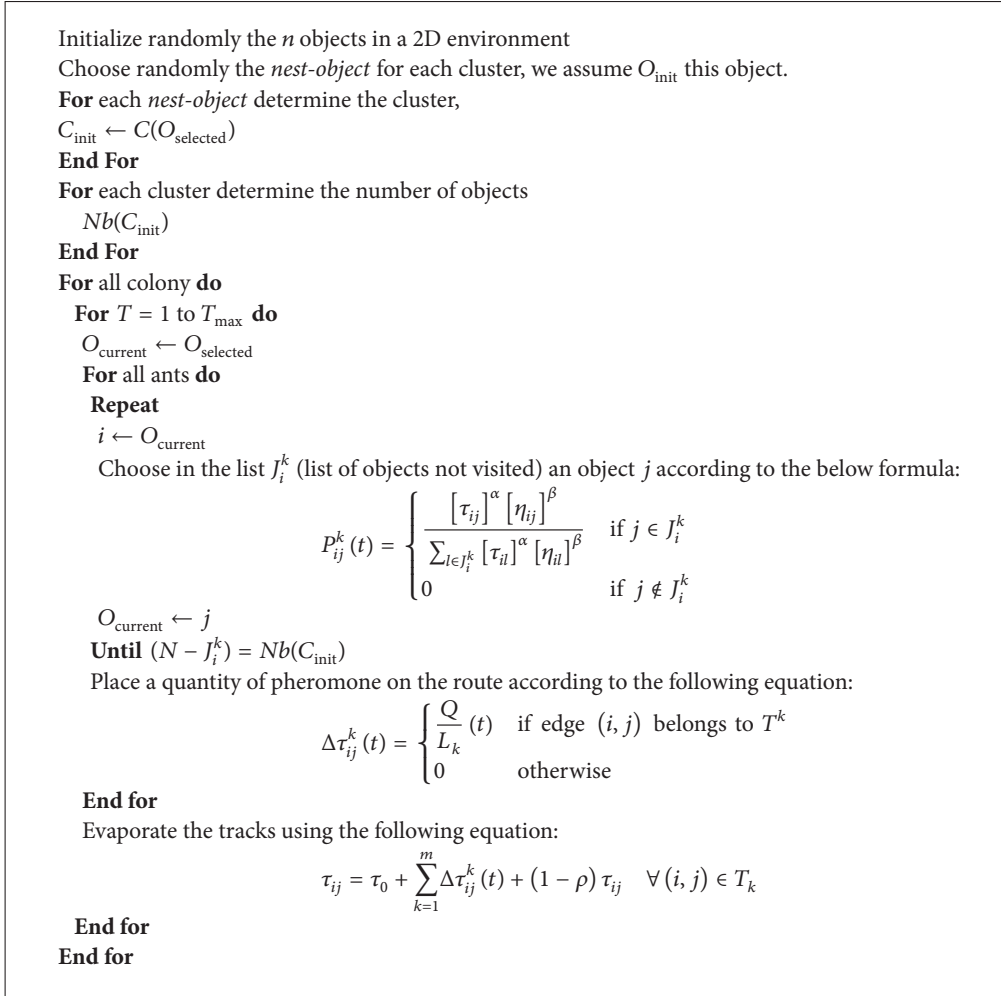
k_{fix} : constant.

m : number of ants in colony.

Its general principle is defined as shown in Algorithm 1.

3.4. Improvements of ASClass Algorithm. In studying the asymptotic behavior of ASClass algorithm, we make the convenient assumption: there are still some objects which are not assigned to any cluster when the ASClass algorithm stops. We called these objects *Outliers*. This phenomenon is caused by the fact that an object can belong to two or more clusters (the same object can be added by several colonies to their paths). Solutions that we propose in this paper are presented in the following.

3.4.1. First Solution: ASClassⁱ Algorithms. In order to find partition for unclassifiable objects (*Outliers*), we propose applying, respectively, K -means algorithm, K -medoid algorithm, and FCM algorithm on dataset of *Outliers* samples.



ALGORITHM 1

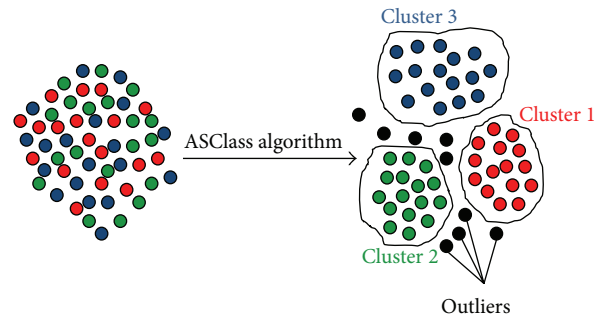
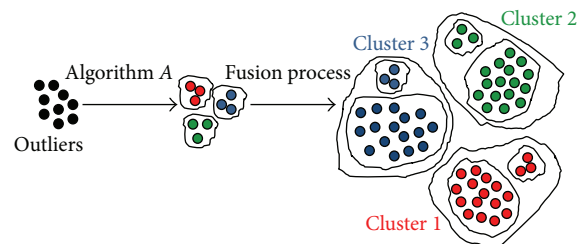
TABLE 1: Hybrid ASClassⁱ algorithms.

Index	Algorithm A (second step)	Proposed models
1	K-means	ASClass ¹
2	K-medoid	ASClass ²
3	FCM	ASClass ³

Our proposed method ASClass, presented in the previous section, can also be used as an initialization step for these algorithms. As can be seen in Table 1, we called, respectively, ASClass¹, ASClass², and ASClass³ our proposed versions of ASClass.

The ASClassⁱ procedure consists of simply starting with k groups. The initialization part is identical to ASClass. We are able to show in Figure 3 that the output of first step consists of classified and unclassified objects. Classified samples are represented by different symbol (red, green, and blue). Unclassified ones are represented by black point.

As can be seen in Figure 4, unclassified objects resulting at the first step (black point) are given as input parameter to ASClassⁱ, in the second step.

FIGURE 3: Step 1 of both ASClassⁱ and F-ASClass algorithms.FIGURE 4: Step 2 of ASClassⁱ algorithms.

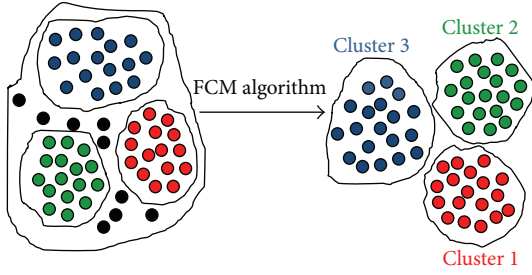


FIGURE 5: Step 2 of F-ASClass algorithm.

Algorithm A (different values of A are given in Table 1) is applied on dataset of unclassified objects to reassigning them to the appropriate cluster. A algorithm create new clusters with the same specified number K. These clusters will be merged with existing clusters created at the first step.

3.4.2. Second Solution: F-ASClass Algorithm. The result of the partition founded by ASClass in the first step is given as input parameter to FCM algorithm in the second step. An element u_{ij} of partition matrix represents the grade of membership of object x_i in cluster c_j . Here, u_{ij} is a value that described the membership of object i to class.

We will initialize the partition matrix (membership matrix) given to FCM function by 1 according to classified object and fz according to unclassified objects.

For the classified objects, if object i is in class j $u_{ij} = 1$; otherwise $u_{ij} = 0$. For the unclassified objects, $u_{ij} = fz$ for all class j ($j = 1, 2, 3, \dots, K$), meaning that an unclassified object i has the same membership to all class j . The sum of membership values across classes must equal one.

$$fz = \frac{1}{K}. \quad (8)$$

Figure 5 shows that the output of the first step (see Figure 3) will be given as input parameter in the second step.

4. Experimental Results

4.1. Artificial and Real Data. To evaluate the contribution of our method, we use several numerical datasets, including artificial and real databases from the Machine Learning Repository [39]. Concerning the artificial datasets, the databases *art1*, *art2*, *art3*, *art5*, and *art6* are generated with *Gaussian* laws and with various difficulties (classes overlay, nonrelevant attributes, etc.), and *art4* data is generated with uniform law. The general information about the databases is summarized in Table 2. For each data file, the following fields are given: the number of objects (N), the number of attributes (N_{Att}), and the number of clusters expected to be found in the datasets (N_C).

Art1 dataset is the type of data most frequently used within previous work on ant based clustering algorithm [22, 23].

TABLE 2: Main characteristics of artificial and real databases used in our tests.

Datasets	N_{Att}	N	K	Class distribution
Art1[4, 400]	2	400	4	(100, 100, 100, 100)
Art2[2, 1000]	2	1000	2	(500, 500)
Art3[4, 400]	2	1100	4	(500, 50, 500, 50)
Art4[2, 200]	2	200	2	(100, 100)
Art5[9, 900]	2	900	9	(100, 100, ..., 100)
Art6[4, 400]	8	400	4	(100, 100, 100, 100)
Iris[3, 150]	4	150	3	(50, 50, 50)
Thyroid[3, 215]	5	215	3	(150, 35, 30)
Pima[2, 768]	8	768	2	(500, 268)

The data are normalized in $[0, 1]$, the measure of similarity is based on Euclidian distance, and the algorithms parameters used in our tests were always the same for all databases and all algorithms.

As can be seen in Table 2, Fisher's Iris dataset contains 3 classes of 50 instances where each class refers to a type of Iris plant. An example of class discovery of ASClass on Iris dataset is shown in Figure 7.

Five versions of the ASClass algorithm were coded and tested with test data in order to determine accuracy or results. One implementation presented basic features of ASClass without improvements. ASClass¹, ASClass², and ASClass³ are improvements versions of the basic ASClass in which we propose applying, respectively, the first solution described in Section 3.4.1. F-ASClass is a fuzzy-ant clustering solutions described in Section 3.4.2.

4.2. Evaluation Functions. The quality of the clustering results of the different algorithms on the test sets are compared using the following performances measures. The first performance measure is the error classification (Ec) index. It is defined as

$$Ec = 1 - \frac{\sum_{k=1}^K \text{Recall}_k}{K}, \quad (9)$$

$$\text{Recall}_k = \frac{n_{kk}}{n_k}.$$

The second measure used in this paper is the *accuracy* coefficients. It determines the ratio of correctly assigned objects:

$$\text{Accuracy} = \frac{\sum_{k=1}^K n_{kk}}{N}, \quad (10)$$

where n_{kk} is the number of objects of the k th class correctly classified, n_k is the number of objects in cluster C_k , and N is the total number of objects in dataset. Not_{ass} represents the percentage of not assigned objects in the predicted partition.

TABLE 3: Error classification and accuracy coefficients found on Iris dataset. Average on 20 trials, 100 iterations per trial.

Datasets	m	Ec	Accuracy	Not_ass
Iris[3, 150]	5	0.310	0.663	28.70%
	10	0.320	0.651	28.13%
	20	0.309	0.662	30.43%
	30	0.275	0.696	28.37%
	40	0.292	0.667	29.43%
	50	0.249	0.724	26.80%
	100	0.251	0.729	28.70%

Moreover, we will use *separation* index. It is defined in [40] as

$$S = \frac{\sum_{i=1}^K \sum_{j=1}^N (\mu_{ij})^2 \|O_j - v_i\|^2}{N * \min_{i,j} \|O_j - v_i\|^2}. \quad (11)$$

In [5] u_{ij} is defined as follows: u_{ij} ($i = 1, \dots, K$; $j = 1, \dots, N$) is the membership of any fuzzy partition. The corresponding hard partition of u_{ij} is defined as $w_{ij} := 1$ if $\text{argmax}_i \{u_{ij}\}$; $w_{ij} := 0$ otherwise. Cluster should be well separated; thereby a smaller S indicates the partition in which all the clusters are overall compact and separate to each other.

The centroid of i is represented by the parameter v_i .

4.3. Results. This section is divided into two parts. First, we study the behavior of ASClass with respect to the number of ants in each colony using Iris dataset. Second, we compare F-ASClass with the K -means, K -medoid, FCM, and ASClass ^{i} ($i = 1, 2, 3$), using test datasets presented in Table 2. This comparison can be done on the basis of average classification error and average value of accuracy.

4.3.1. Parameters Setting of ASClass Algorithm. We propose studying the speed to find the optimal solution, defined as the number of iterations, as a function of the number of ants in ASClass. So we evaluate the performance of ASClass varying the number of ants from 5 to 100, given a fixed number of iterations (100 iterations per trial).

Results presented in Figures 6(a)–6(g) show that the length of the best tour made by each colony of ants is improved very fast in the initial phase of the algorithm. Once the second phase has been reckoned, new good solutions start to appear but a phenomenon of local optima is discovered in the last phase. In this stage, the system ceased to explore new solutions and therefore will not be improved any more. This process called, in the literature of ACO algorithms, *unipath behavior* and it would indicate the situation in which the ants follow the same path and so create the same cluster. This was due to a much higher quantities of artificial pheromones deposited by ants following that path than on all the others.

On one hand, we can observe from Table 3 that the best results with ASClass (corresponds to the smallest value of error of classification, $Ec = 0.249$, and the largest value of accuracy, $Ac = 0.729$) are obtained when the number of ants is equal to 50 (number of samples in each cluster = 50). It is clear here that the number of ants improve the efficiency of solution quality: a run with 100 ants is more search effective than with 5 ants. This can be explained by the importance of communication in colony through trail in using many ants (in case of 100 ants). This propriety, called synergistic effect, makes ASClass attractiveness on its ability of finding quickly an optimal solution.

As a semisupervised clustering algorithm, we can use confusion matrix as a visualization tool to evaluate the performance of ASClass. It contains information about the number per class of well classifies and mislabeled samples. Furthermore, it contains information about the capability of system to not mislabel one cluster as another. An example of confusion matrix on Iris dataset is shown in Figures 8(a)–8(g).

It is important to note that Iris dataset presents the following characteristic: one cluster is linearly separable from the other two; the latter are not linearly separable from each other. We have presented this characteristic in Figure 7. Results presented in Figures 8(a)–8(g) prove that ASClass may be capable of identifying this characteristic.

It is clear to see that for all examples in Figures 8(a)–8(g), the illustrated results show that all predicted samples of cluster 1 are correctly assigned as cluster 1. Besides no instance of cluster 1 is misclassified as class 2 and class 3. However, a large number of predicted samples of cluster 2 may be misclassified into cluster 3 and a few number are misclassified into cluster 1. In case of cluster 3, there are some samples misclassified into cluster 2 and no one is misclassified into cluster 1.

As can be seen in Figures 8(a)–8(g), which represent confusion matrix for only clustered data, the confusion matrix did not contain the total 150 instances distribution of Iris data: in the case of ASClass, there are still some objects which are not assigned to any cluster when the algorithm stops.

4.3.2. Comparative Analysis. In order to compare the effectiveness of our proposed improvements of ASClass, we apply them to datasets presented in Table 2. We also use tree indexes of cluster validity: error classification (Ec), accuracy, and separation (S) coefficient. “The index should make good intuitive sense, should have a basis in theory, and should be readily computable” [4].

Note that underline bold face indicates the best and bold face indicates the second best. In K -means, K -medoid, and FCM algorithms the predefined input number of clusters K is set to the known number of classes in each dataset.

Tables 4 and 5 contain, respectively, the mean, standard deviations, mode, minimal, and maximal value of error rate of classification and the accuracy obtained by the four

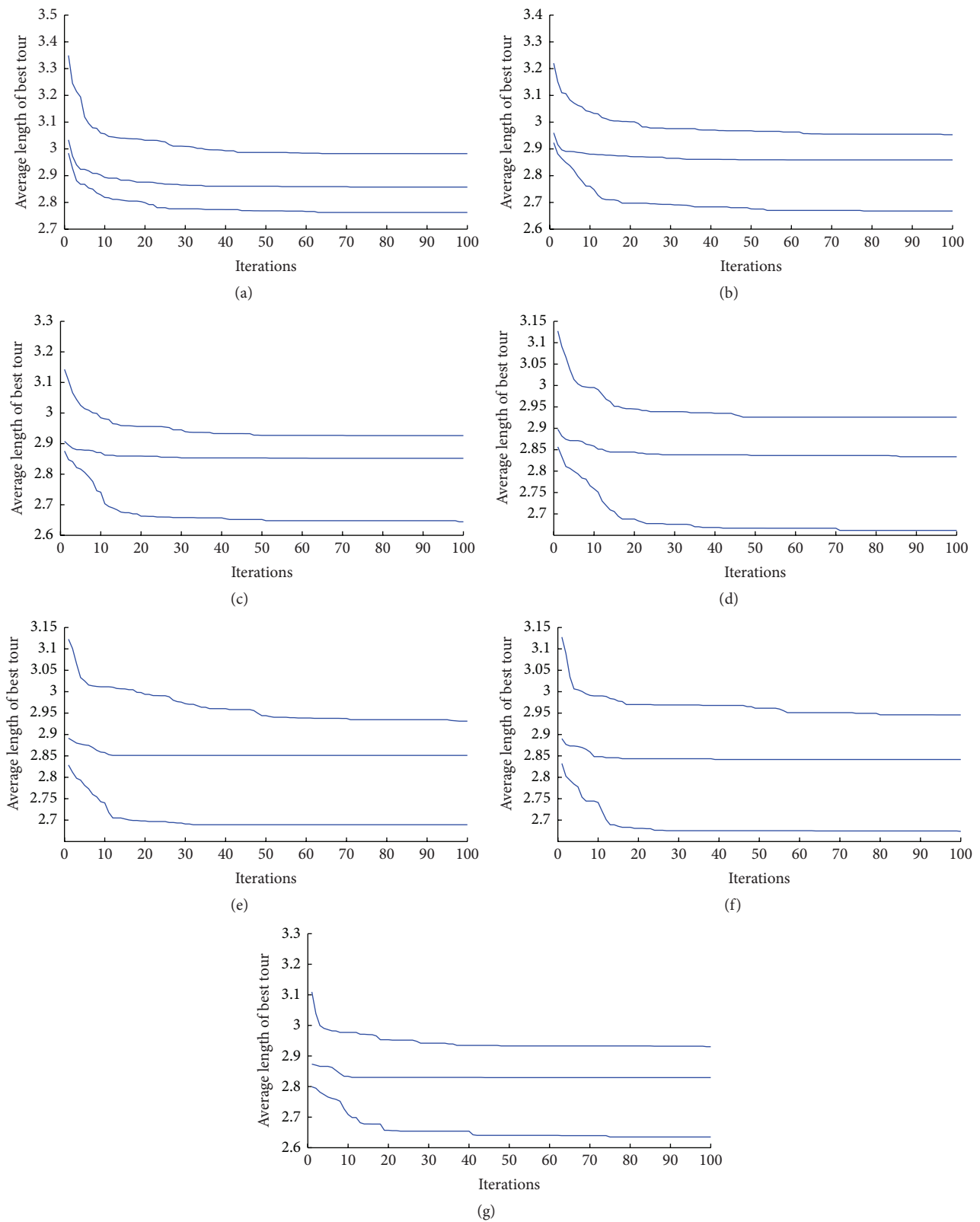


FIGURE 6: Results obtained by ASClass algorithm obtained on Iris dataset (a) $m = 5$; (b) $m = 10$; (c) $m = 20$; (d) $m = 30$; (e) $m = 40$; (f) $m = 50$; (g) $m = 100$.

TABLE 4: Mean, standard deviation, mode, min, and max values of *error rate* achieved by each clustering algorithm (*K*-means, *K*-medoid, FCM, and ASClass) over 20 trials on dataset presented in Table 2.

Datasets	Algorithm	Mean	Std	Mode	Min	Max
Art1[4, 400]	<i>K</i> -means	0.806	0.180	0.947	0.5225	0.9950
	<i>K</i> -medoid	0.773	0.158	0.760	0.5150	0.9750
	FCM	0.555	1.139×10^{-16}	0.555	0.5550	0.5550
	ASClass	<u>0.471</u>	<u>0</u>	<u>0.471</u>	<u>0.4713</u>	<u>0.4713</u>
Art2[2, 1000]	<i>K</i> -means	0.548	0.490	0.980	0.020	0.980
	<i>K</i> -medoid	0.404	0.482	0.020	0.020	0.980
	FCM	0.980	2.278×10^{-16}	0.980	0.980	0.980
	ASClass	<u>0.039</u>	<u>0</u>	<u>0.039</u>	<u>0.039</u>	<u>0.039</u>
Art3[4, 100]	<i>K</i> -means	0.749	0.227	0.585	0.222	0.989
	<i>K</i> -medoid	0.810	0.141	0.549	0.549	0.995
	FCM	0.644	0	0.644	0.644	0.644
	ASClass	<u>0.507</u>	8.183×10^{-16}	<u>0.508</u>	<u>0.506</u>	<u>0.508</u>
Art4[2, 200]	<i>K</i> -means	0.450	0.510	0	0	1
	<i>K</i> -medoid	0.572	0.494	1	0	1
	FCM	0.985	2.278×10^{-16}	0.985	0.985	0.985
	ASClass	<u>0.015</u>	<u>0</u>	<u>0.015</u>	<u>0.015</u>	<u>0.015</u>
Art5[9, 900]	<i>K</i> -means	0.908	0.065	0.873	0.802	0.990
	<i>K</i> -medoid	0.883	0.133	0.985	0.455	0.996
	FCM	0.905	2.278×10^{-16}	0.905	0.905	0.905
	ASClass	<u>0.4766</u>	<u>0</u>	<u>0.476</u>	<u>0.476</u>	<u>0.476</u>
Art6[4, 400]	<i>K</i> -means	0.641	0.273	0.750	0	1
	<i>K</i> -medoid	0.827	0.176	1	0.427	1
	FCM	0.750	0	0.750	0.750	0.750
	ASClass	<u>0.089</u>	0.002	0.088	0.085	0.095
Iris[3, 150]	<i>K</i> -means	0.576	0.336	0.480	0.113	1
	<i>K</i> -medoid	0.710	0.229	0.126	0.126	1
	FCM	0.746	2.278×10^{-16}	0.746	0.746	0.746
	ASClass	<u>0.257</u>	5.695×10^{-17}	0.257	0.257	0.257
Thyroid[3, 215]	<i>K</i> -means	0.774	0.166	0.922	0.239	0.922
	<i>K</i> -medoid	0.594	0.261	0.666	0.128	0.952
	FCM	0.847	2.278×10^{-16}	0.847	0.847	0.847
	ASClass	<u>0.476</u>	0	0.476	0.476	0.476
Pima[2, 768]	<i>K</i> -means	<u>0.461</u>	0.125	0.371	0.371	0.628
	<i>K</i> -medoid	0.504	0.116	0.616	0.325	0.641
	FCM	0.650	1.139×10^{-16}	0.650	0.650	0.650
	ASClass	0.543	0.008	0.538	0.530	0.565

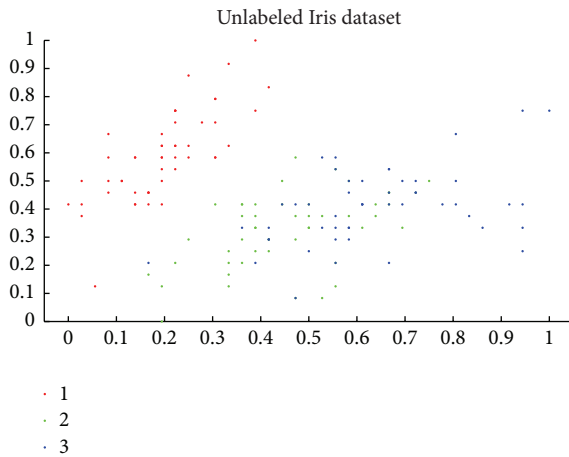


FIGURE 7: Two-dimensional projection of Iris dataset onto first two principal components.

automatic clustering algorithms (*K*-means, *K*-medoid, FCM, and ASClass).

In Tables 6 and 7, we report, respectively, the mean, standard deviations, mode, minimal, and maximal values of the value of error rate of classification and accuracy obtained by the four automatic clustering algorithms (ASClass¹, ASClass², ASClass³, and F-ASClass).

Table 8 contains, respectively, the mean, standard deviations, mode, minimal, and maximal value of separation obtained by all fuzzy approaches: FCM, ASClass³ and F-ASClass.

In general, the comparative results presented in Table 4 indicate that our proposed algorithm generates more compact clusters as well as lower error rates than the other clustering algorithms. The only exception is on the *Pima* dataset where ASClass does not achieve the minimum error rates.



FIGURE 8: Confusion matrix obtained with ASClass on Iris dataset for 100 iterations. (a) $m = 5$; (b) $m = 10$; (c) $m = 20$; (d) $m = 30$; (e) $m = 40$; (f) $m = 50$; (g) $m = 100$.

Table 5 also conforms to the fact that ASClass algorithm remains clearly and consistently superior to the other three techniques in terms of the clustering accuracy. The only exception is also seen on *Pima* dataset where the best value is generated by the *K*-means algorithm.

In Tables 6 and 7 we report, respectively, the mean, standard deviations, mode, minimal, and maximal values of

error rate classification and accuracy obtained by ASClass¹, ASClass², ASClass³, and F-ASClass algorithms. It is clear from these tables that F-ASClass algorithm can minimize the value error classification and maximize accuracy for all datasets. The only exception is for *Art3* and *Pima* datasets. The best values on error rate classification for these both datasets are given by ASClass¹. The best value for

TABLE 5: Mean, standard deviation, mode, min, and max values of *accuracy* classification achieved by each clustering algorithm (*K*-means, *K*-medoid, FCM, and ASClass) over 20 trials on dataset presented in Table 2.

Datasets	Algorithm	Mean	Std	Mode	Min	Max
Art1[4, 400]	<i>K</i> -means	0.193	0.180	0.052	0.052	0.477
	<i>K</i> -medoid	0.226	0.158	0.050	0.025	0.485
	FCM	0.445	1.139×10^{-16}	0.445	0.445	0.445
	ASClass	0.352	1.708×10^{-16}	0.352	0.352	0.352
Art2[2, 1000]	<i>K</i> -means	0.452	0.490	0.020	0.020	0.980
	<i>K</i> -medoid	0.596	0.482	0.980	0.020	0.980
	FCM	0.020	3.559×10^{-18}	0.020	0.020	0.020
	ASClass	0.965	4.556×10^{-16}	0.965	0.965	0.965
Art3[4, 100]	<i>K</i> -means	0.279	0.236	0.049	0.002	0.702
	<i>K</i> -medoid	0.201	0.195	0.344	9.090×10^{-4}	0.674
	FCM	0.066	1.423×10^{-19}	0.066	0.066	0.066
	ASClass	0.661	0.001	0.660	0.660	0.662
Art4[2, 200]	<i>K</i> -means	0.550	0.510	1	0	1
	<i>K</i> -medoid	0.427	0.494	0	0	1
	FCM	0.015	8.898×10^{-16}	0.015	0.015	0.015
	ASClass	0.984	1.139×10^{-16}	0.984	0.984	0.984
Art5[9, 900]	<i>K</i> -means	0.091	0.065	0.126	0.010	0.197
	<i>K</i> -medoid	0.116	0.133	0.014	0.003	0.544
	FCM	0.094	4.271×10^{-17}	0.094	0.094	0.094
	ASClass	0.493	2.278×10^{-16}	0.493	0.493	0.493
Art6[4, 400]	<i>K</i> -means	0.358	0.273	0.250	0	1
	<i>K</i> -medoid	0.172	0.176	0	0	0.572
	FCM	0.250	0	0.250	0.250	0.250
	ASClass	0.918	0.002	0.918	0.912	0.921
Iris[3, 150]	<i>K</i> -means	0.424	0.336	0.520	0	0.886
	<i>K</i> -medoid	0.289	0.229	0	0	0.873
	FCM	0.253	5.695×10^{-17}	0.253	0.253	0.253
	ASClass	0.745	0	0.745	0.745	0.745
Thyroid[3, 215]	<i>K</i> -means	0.212	0.281	0.032	0.032	0.888
	<i>K</i> -medoid	0.448	0.341	0.697	0.023	0.944
	FCM	0.074	0	0.074	0.074	0.074
	ASClass	0.650	1.139×10^{-16}	0.650	0.650	0.650
Pima[2, 768]	<i>K</i> -means	0.550	0.164	0.668	0.332	0.668
	<i>K</i> -medoid	0.505	0.158	0.334	0.320	0.686
	FCM	0.333	1.139×10^{-16}	0.3333	0.333	0.333
	ASClass	0.454	0.005	0.456	0.438	0.464

accuracy is given by ASClass² for *Art3* and by ASClass¹ for *Pima*.

Moreover, F-ASClass appears to be the most stable algorithm by having the lowest value of standard deviation for all indexes validity on all datasets.

Both ASClass¹ and ASClass² are unable to find new partition for *Art4* dataset where the number of unclassified objects is too low (unclassified objects = 3). However both ASClass³ and F-ASClass can be executed on small datasets.

This feature proves that fuzzy-hybrid techniques are more efficient than hard-hybrid technique.

Table 8 shows that the most partition is made by F-ASClass and the second best value is given by ASClass³; thus a smallest S indeed indicates a valid optimal partition. As can be seen in (11), the more separate the clusters, the larger ($\|O_j - O_i\|^2$) and the smaller S . Thus, the fuzzy clustering rule used in F-ASClass is to modify the compactness measure by minimizing S .

TABLE 6: Average on 20 trials. Mean, standard deviation, mode, min, and max values of *error rate* achieved by each clustering (ASClass¹, ASClass², ASClass³, and F-ASClass) algorithm over 20 trials on dataset presented in Table 2.

Datasets	Algorithm	Mean	Std	Mode	Min	Max
Art1[4, 400]	ASClass ¹	0.678	0.096	0.7625	0.434	0.782
	ASClass ²	0.686	0.095	0.755	0.462	0.790
	ASClass ³	0.585	2.278×10^{-16}	0.585	0.585	0.585
	F-ASClass	0.545	0	0.545	0.545	0.545
Art2[2, 1000]	ASClass ¹	0.140	0.017	0.148	0.106	0.173
	ASClass ²	0.144	0.035	0.084	0.084	0.195
	ASClass ³	0.145	5.695×10^{-17}	0.146	0.146	0.146
	F-ASClass	0.020	0	0.020	0.020	0.020
Art3[4, 100]	ASClass ¹	0.609	0.071	0.616	0.409	0.703
	ASClass ²	0.633	0.070	0.689	0.473	0.705
	ASClass ³	0.619	1.139×10^{-16}	0.619	0.619	0.619
	F-ASClass	0.676	1.139×10^{-16}	0.675	0.675	0.675
Art4[2, 200]	ASClass ¹	—	—	—	—	—
	ASClass ²	—	—	—	—	—
	ASClass ³	0.020	0	0.020	0.020	0.020
	F-ASClass	0.015	0	0.015	0.015	0.015
Art5[9, 900]	ASClass ¹	0.664	0.033	0.701	0.591	0.706
	ASClass ²	0.669	0.031	0.694	0.603	0.708
	ASClass ³	0.707	2.278×10^{-16}	0.707	0.707	0.707
	F-ASClass	0.523	0	0.523	0.523	0.523
Art6[4, 400]	ASClass ¹	0.178	0.031	0.155	0.092	0.207
	ASClass ²	0.175	0.025	0.207	0.114	0.207
	ASClass ³	0.179	8.542×10^{-17}	0.179	0.179	0.179
	F-ASClass	0	0	0	0	0
Iris[3, 150]	ASClass ¹	0.335	0.050	0.293	0.273	0.420
	ASClass ²	0.345	0.060	0.313	0.240	0.426
	ASClass ³	0.306	0	0.306	0.306	0.306
	F-ASClass	0.106	4.271×10^{-17}	0	0	0.004
Thyroid[3, 215]	ASClass ¹	0.504	0.111	0.350	0.350	0.678
	ASClass ²	0.536	0.121	0.564	0.313	0.678
	ASClass ³	0.573	1.139×10^{-16}	0.573	0.573	0.573
	F-ASClass	0.196	5.695×10^{-17}	0.196	0.196	0.196
Pima[2, 768]	ASClass ¹	0.518	0.038	0.538	0.456	0.582
	ASClass ²	0.537	0.034	0.501	0.480	0.594
	ASClass ³	0.551	0	0.551	0.551	0.551
	F-ASClass	0.650	1.139×10^{-16}	0.650	0.650	0.650

5. Conclusion

We have presented in this paper a new fuzzy-swarm-algorithm called F-ASClass for data clustering in a knowledge discovery context. F-ASClass introduces new fuzzy-heuristic for the ant colony inspired from the spider web construction. We have also proposed using several colonies of ants; the

number of colonies in F-ASClass depended on the number of clusters in databases to be classified. Fuzzy rules are introduced to find a partition for unclassified objects generated by the work of artificial ants. The experimental results show that the proposed algorithm is able to achieve an interesting result with respect to the well-known clustering algorithm,

TABLE 7: Mean, standard deviation, mode, min, and max values of *accuracy* classification achieved by each clustering algorithm (ASClass¹, ASClass², ASClass³, and F-ASClass) over 20 trials on dataset presented in Table 2.

Datasets	Algorithm	Mean	Std	Mode	Min	Max
Art1[4, 400]	ASClass ¹	0.321	0.096	0.2375	0.2175	0.565
	ASClass ²	0.313	0.095	0.245	0.210	0.5375
	ASClass ³	0.415	5.695×10^{-17}	0.415	0.415	0.415
	F-ASClass	0.455	5.695×10^{-17}	0.455	0.455	0.455
Art2[2, 1000]	ASClass ¹	0.859	0.017	0.852	0.826	0.894
	ASClass ²	0.855	0.035	0.805	0.805	0.915
	ASClass ³	0.854	4.556×10^{-16}	0.854	0.854	0.854
	F-ASClass	0.980	2.27×10^{-16}	0.980	0.980	0.980
Art3[4, 100]	ASClass ¹	0.476	0.086	0.430	0.390	0.680
	ASClass ²	0.483	0.101	0.384	0.384	0.663
	ASClass ³	0.397	1.139×10^{-16}	0.619	0.619	0.619
	F-ASClass	0.466	2.278×10^{-16}	0.466	0.466	0.466
Art4[2, 200]	ASClass ¹	—	—	—	—	—
	ASClass ²	—	—	—	—	—
	ASClass ³	0.980	2.278×10^{-16}	0.980	0.980	0.980
	F-ASClass	0.985	2.278×10^{-16}	0.985	0.985	0.985
Art5[9, 900]	ASClass ¹	0.335	0.033	0.293	0.293	0.408
	ASClass ²	0.330	0.031	0.305	0.291	0.396
	ASClass ³	0.292	1.139×10^{-16}	0.292	0.292	0.292
	F-ASClass	0.476	0	0.476	0.476	0.476
Art6[4, 400]	ASClass ¹	0.821	0.031	0.792	0.792	0.907
	ASClass ²	0.824	0.025	0.792	0.792	0.885
	ASClass ³	0.820	8.542×10^{-17}	0.179	0.179	0.179
	F-ASClass	1	0	1	1	1
Iris[3, 150]	ASClass ¹	0.664	0.050	0.706	0.580	0.726
	ASClass ²	0.655	0.060	0.580	0.573	0.760
	ASClass ³	0.693	0	0.693	0.693	0.693
	F-ASClass	0.893	1.139×10^{-16}	0.893	0.893	0.893
Thyroid[3, 215]	ASClass ¹	0.592	0.052	0.595	0.493	0.674
	ASClass ²	0.567	0.055	0.544	0.493	0.665
	ASClass ³	0.544	1.139×10^{-16}	0.544	0.544	0.544
	F-ASClass	0.906	1.139×10^{-16}	0.906	0.906	0.906
Pima[2, 768]	ASClass ¹	0.473	0.021	0.447	0.447	0.505
	ASClass ²	0.468	0.024	0.450	0.433	0.509
	ASClass ³	0.441	0	0.441	0.441	0.441
	F-ASClass	0.333	1.139×10^{-16}	0.333	0.333	0.333

K -means, K -medoid, and FCM and our proposed hybrid algorithms, ASClass, ASClass¹, ASClass², and ASClass³.

In the future, we will propose a new approach, called F-CIC or “Fuzzy-Collective Intelligence of Colonies,” to master the complexity. In this context we will propose new fuzzy rules to manage communication between colonies. However,

the remarkable success of our fuzzy-metaheuristic proposed in this paper can serve as a starting point for new approach in engineering and computer science. In our approach basic principles of swarm intelligence are still present but again untimely we will propose using a good improvement with fuzzy rules and fractal theory.

TABLE 8: Mean, standard deviation, mode, min, and max values of *separation* index achieved by each clustering algorithm (FCM, ASClass³, and F-ASClass) over 20 trials on dataset presented in Table 2.

Datasets	Algorithm	Mean	Std	Mode	Min	Max
Art1[4, 400]	FCM	0.003	4.449×10^{-19}	0.003	0.003	0.003
	ASClass ³	3.013×10^{-5}	6.026×10^{-4}	0	0	0.012
	F-ASClass	6.212×10^{-6}	1.242×10^{-4}	0	0	0.002
Art2[2, 1000]	FCM	0.001	4.449×10^{-19}	0.001	0.001	0.001
	ASClass ³	3.436×10^{-6}	1.086×10^{-4}	0	0	0.003
	F-ASClass	1.256×10^{-6}	3.972×10^{-5}	0	0	0.001
Art3[4, 100]	FCM	0.001	4.449×10^{-19}	0.001	0.001	0.001
	ASClass ³	4.383×10^{-6}	1.453×10^{-4}	0	0	0.004
	F-ASClass	1.180×10^{-6}	3.913×10^{-5}	0	0	0.001
Art4[2, 200]	FCM	0.009	3.559×10^{-18}	0.009	0.009	0.009
	ASClass ³	3.771×10^{-5}	5.333×10^{-4}	0	0	0.007
	F-ASClass	3.525×10^{-5}	4.985×10^{-4}	0	0	0.007
Art5[9, 900]	FCM	0.001	0	0.001	0.001	0.001
	ASClass ³	4.017×10^{-6}	1.205×10^{-4}	0	0	0.003
	F-ASClass	1.207×10^{-6}	3.623×10^{-5}	0	0	0.001
Art6[4, 400]	FCM	0.002	8.898×10^{-19}	0.002	0.002	0.002
	ASClass ³	8.116×10^{-6}	1.623×10^{-4}	0	0	0.003
	F-ASClass	4.203×10^{-6}	8.407×10^{-5}	0	0	0.001
Iris[3, 150]	FCM	0.006	1.779×10^{-18}	0.006	0.006	0.006
	ASClass ³	9.837×10^{-5}	0.001	0	0	0.014
	F-ASClass	3.171×10^{-5}	3.884×10^{-4}	0	0	0.004
Thyroid[3, 215]	FCM	0.0089	3.559×10^{-18}	0.008	0.008	0.008
	ASClass ³	3.857×10^{-4}	0.005	0	0	0.082
	F-ASClass	2.470×10^{-5}	3.623×10^{-4}	0	0	0.005
Pima[2, 768]	FCM	0.012	5.339×10^{-18}	0.012	0.012	0.012
	ASClass ³	2.863×10^{-5}	7.935×10^{-4}	0	0	0.021
	F-ASClass	6.369×10^{-6}	1.765×10^{-4}	0	0	0.004

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

- [1] S. Camazine, J. L. Deneubourg, N. R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau, *Self-Organization in Biological Systems*, Princeton University Press, 2001.
- [2] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, New York, NY, USA, 1999.
- [3] P. P. Grassé, "La reconstruction du nid et les coordinations inter-individuelles chez *Bellicositermes natalensis* et *Cubitermes* sp. La théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs," *Insectes Sociaux*, vol. 6, pp. 41–81, 1959.
- [4] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice Hall Advanced Reference Series, Prentice Hall, Englewood Cliffs, NJ, USA, 1988.
- [5] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.
- [6] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, University of California Press, 1967.
- [7] L. Kaufman and P. J. Rousseeuw, "Clustering by means of Medoids," in *Statistical Data Analysis Based on the L1-Norm and Related Methods*, Y. Dodge, Ed., pp. 405–416, North-Holland, New York, NY, USA, 1987.
- [8] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *Journal of Cybernetics*, vol. 3, no. 3, pp. 32–57, 1973.
- [9] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, NY, USA, 1981.
- [10] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: the fuzzy c-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2-3, pp. 191–203, 1984.

- [11] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338–353, 1965.
- [12] L. V. Utkin, "Fuzzy one-class classification model using contamination neighborhoods," *Advances in Fuzzy Systems*, vol. 2012, Article ID 984325, 10 pages, 2012.
- [13] N. J. Pizzi and W. Pedrycz, "Classifying high-dimensional patterns using a fuzzy logic discriminant network," *Advances in Fuzzy Systems*, vol. 2012, Article ID 920920, 7 pages, 2012.
- [14] L. Liparulo, A. Proietti, and M. Panella, "Fuzzy clustering using the convex hull as geometrical model," *Advances in Fuzzy Systems*, vol. 2015, Article ID 265135, 13 pages, 2015.
- [15] M. Omran, A. Salman, and A. Engelbrecht, "Image classification using particle swarm optimization," in *Proceedings of the Conference on Simulated Evolution and Learning*, vol. 1, pp. 370–374, Singapore, November 2002.
- [16] D. Karaboga and C. Ozturk, "A novel clustering approach: Artificial Bee Colony (ABC) algorithm," *Applied Soft Computing*, vol. 11, no. 1, pp. 652–657, 2011.
- [17] J. Senthilnath, S. N. Omkar, and V. Mani, "Clustering using firefly algorithm: performance study," *Swarm and Evolutionary Computation*, vol. 1, no. 3, pp. 164–171, 2011.
- [18] L. Xiao, "A clustering algorithm based on artificial fish school," in *Proceedings of the 2nd International Conference on Computer Engineering and Technology (ICCET '10)*, pp. 766–769, Chengdu, China, 2010.
- [19] C. Grosan, A. Abraham, and M. Chis, *Swarm Intelligence in Data Mining. Studies in Computational Intelligence (SCI)*, vol. 34, Springer, Berlin, Germany, 2006.
- [20] A. Hamdi, N. Monmarché, A. M. Alimi, and M. Slimane, "Artificial ants for automatic classification," in *Artificial Ants: From Collective Intelligence to Real-Life Optimization and Beyond*, N. Monmarché, F. Guinand, and P. Siarry, Eds., pp. 265–290, ISTE-WILEY, Hoboken, NJ, USA, 2010.
- [21] J. Deneubourg, J. S. Goss, S. Pasteels, J. Fresneau, and D. Lachaud, "Self-organization mechanisms in ant societies (II): learning in foraging and division of labor," in *From Individual to Collective Behavior in Social Insects*, Experientia Supplementum, pp. 177–196, Birkhäuser, 1987.
- [22] E. Lumer and B. Faieta, "Diversity and adaptation in populations of clustering ants," in *Proceedings of the 3rd International Conference on Simulation of Adaptive Behavior: From Animals to Animats (SAB '94)*, pp. 501–508, MIT Press, Cambridge, Mass, USA, 1994.
- [23] N. Monmarché, "On data clustering with artificial ants," in *Proceedings of the Workshop on Data Mining with Evolutionary Algorithms: Research Directions (AAAI '99 & GECCO '99)*, A. A. Freitas, Ed., pp. 23–26, Orlando, Fla, USA, 1999.
- [24] S. Ouadfel and M. Batouche, "An efficient ant algorithm for swarm-based image clustering," *Journal of Computer Science*, vol. 3, no. 3, pp. 162–167, 2007.
- [25] N. Labroche, N. Monmarché, and G. Venturini, "AntClust: ant clustering and web usage mining," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '03)*, pp. 25–36, Chicago, Ill, USA, July 2003.
- [26] H. Azzag, N. Monmarché, M. Slimane, G. Venturini, and C. Guinot, "AntTree: a new model for clustering with artificial ants," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 2642–2647, December 2003.
- [27] G. W. Reynolds, "Flocks, herds and schools: a distributed behavioral model," in *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '87)*, pp. 25–34, 1987.
- [28] G. Proctor and C. Winter, "Information flocking: data visualisation in virtual worlds using emergent behaviours," in *Proceedings of the 1st International Conference on Virtual Worlds (VW '98)*, pp. 168–176, Paris, France, 1998.
- [29] S. Aupetit, N. Monmarché, M. Slimane, C. Guinot, and G. Venturini, "Clustering and dynamic data visualization with artificial flying insect," in *Genetic and Evolutionary Computation—GECCO 2003*, vol. 2723 of *Lecture Notes in Computer Science*, pp. 140–141, Springer, Berlin, Germany, 2003.
- [30] A. Abraham, S. Das, and S. Roy, "Swarm intelligence algorithms for data clustering," in *Soft Computing for Knowledge Discovery and Data Mining*, pp. 279–313, Springer, Berlin, Germany, 2008.
- [31] C. Bourjot, V. Chevrier, and V. Thomas, "A new swarm mechanism based on social spiders colonies: from web weaving to region detection," *Web Intelligence and Agent Systems*, vol. 1, no. 1, pp. 47–64, 2003.
- [32] A. Hamdi, N. Monmarché, A. M. Alimi, and M. Slimane, "SwarmClass a novel data clustering approach by a hybridization of an ant colony with flying insects," in *Ant Colony Optimization and Swarm Intelligence*, vol. 5217 of *Lecture Notes in Computer Science*, pp. 412–414, Springer, Berlin, Germany, 2010.
- [33] A. Hamdi, N. Monmarché, A. M. Alimi, and M. Slimane, "AntBee: clustering with artificial ants in hexagonal grid," in *Proceedings of the International Conference of Artificial Evolution*, pp. 462–472, Angers, France, October 2011.
- [34] A. Hamdi, N. Monmarché, M. Slimane, and A. M. Alimi, "FractAntBee algorithm for data clustering problem," *Journal of Computer Science and Information Security (IJCSIS)*, vol. 14, no. 7, 2016.
- [35] S. Ouadfel and M. Batouche, "MRF-based image segmentation using Ant Colony System," *Electronic Letters on Computer Vision and Image Analysis*, vol. 2, no. 1, pp. 12–24, 2003.
- [36] M. Dorigo and L. Maria Gambardella, *Ant Colonies for the Traveling Salesman Problem*, BioSystems Press, 1997.
- [37] A. Hamdi, N. Monmarché, A. M. Alimi, and M. Slimane, "TspClass: a novel data clustering approach by artificial ants," in *Proceedings of the International Conference on Metaheuristics and Nature Inspired Computing (META '10)*, pp. 27–31, Djerba, Tunisia, 2010.
- [38] A. Hamdi, N. Monmarché, M. Slimane, and A. M. Alimi, "Spider's behavior for ant based clustering algorithm," in *Proceedings of the International Conference on Intelligent Systems Design and Applications (ISDA '15)*, pp. 254–259, Marrakech, Morocco, December 2015.
- [39] M. Lichman, *UCI Machine Learning Repository*, University of California, School of Information and Computer Science, Irvine, Calif, USA, 2013, <http://archive.ics.uci.edu/ml>.
- [40] A. M. Bensaid, L. O. Hall, J. C. Bezdek et al., "Validity-guided (Re)clustering with applications to image segmentation," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 2, pp. 112–123, 1996.

