Available online at www.sciencedirect.com

ScienceDirect

Applied Mathematics and Computation 190 (2007) 1502–1513

APPLIED MATHEMATICS AND COMPUTATION

www.elsevier.com/locate/amc

Application of honey-bee mating optimization algorithm on clustering

Mohammad Fathian, Babak Amiri *, Ali Maroosi

Iran University of Science and Technology, Tehran, Iran

Abstract

Cluster analysis is one of attractive data mining technique that use in many fields. One popular class of data clustering algorithms is the center based clustering algorithm. K-means used as a popular clustering method due to its simplicity and high speed in clustering large datasets. However, K-means has two shortcomings: dependency on the initial state and convergence to local optima and global solutions of large problems cannot found with reasonable amount of computation effort. In order to overcome local optima problem lots of studies done in clustering. Over the last decade, modeling the behavior of social insects, such as ants and bees, for the purpose of search and problem solving has been the context of the emerging area of swarm intelligence. Honey-bees are among the most closely studied social insects. Honey-bee mating may also be considered as a typical swarm-based approach to optimization, in which the search algorithm is inspired by the process of marriage in real honey-bee. Honey-bee has been used to model agent-based systems. In this paper, we proposed application of honeybee mating optimization in clustering (HBMK-means). We compared HBMK-means with other heuristics algorithm in clustering, such as GA, SA, TS, and ACO, by implementing them on several well-known datasets. Our finding shows that the proposed algorithm works than the best one.

Keywords: Clustering; Meta-heuristic; K-means; Honey-bee

1. Introduction

Data mining, extracting interesting and valuable information such as trends, features, or patterns from hidden predictive data, is the multidisciplinary field that is at the intersection of statistics, machine learning, database management, and data visualization, to provide a new perspective on data analysis.

Clustering analysis, which is the subject of active research in several fields such as statistics, pattern recognition, machine learning, and data mining, is to partition a given set of data or objects into clusters (or called groups, classes). It also applied in a large variety of applications, for example, image segmentation, objects and character recognition, document retrieval, etc [17]. There are many methods applied in clustering analysis, like

* Corresponding author.
E-mail address: Amiri_babak@ind.iust.ac.ir (B. Amiri).

0096-3003/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.amc.2007.02.029

hierarchical clustering, partition-based clustering, density-based clustering, and artificial intelligence-based clustering.

One popular class of data clustering algorithms is the center based clustering algorithm. K-means used as a popular clustering method due to its simplicity and high speed in clustering large datasets [6]. However, K-means has two shortcomings: dependency on the initial state and convergence to local optima [20] and global solutions of large problems cannot found with reasonable amount of computation effort [7]. In order to overcome local optima problem lots of studies done in clustering.

Mualik and Bandyopadhyay [22] proposed a genetic algorithm based method to solve the clustering problem and experiment on synthetic and real life datasets to evaluate the performance. The results showed that GA-based method might improve the final output of K-means.

Krishna and Murty [11] proposed a novel approach called genetic K-means algorithm for clustering analysis. It defines a basic mutation operator specific to clustering called distance-based mutation. Using finite Markov chain theory, it proved that GKA converge to the best-known optimum.

In [21] Selim and Al-Sultan discussed the solution of the clustering problem usually solved by the K-means algorithm. The problem known to have local minimum solutions, which are usually what the K-means algorithm obtains. The simulated annealing approach for solving optimization problems described and proposed for solving the clustering problem. The parameters of the algorithm discussed in detail and it shown that the algorithm converges to a global solution of the clustering problem.

According to [5], researchers considered a clustering problem where a given data set partitioned into a certain number of natural and homogeneous subsets such that each subset is composed of elements similar to one another but deferent from those of any other subset. For the clustering problem, a heuristic algorithm exploited by combining the tabu search heuristic with two complementary functional procedures, called packing and releasing procedures. The algorithm numerically tested for its electiveness in comparison with reference works including the tabu search algorithm, the K-means algorithm and the simulated annealing algorithm.

Over the last decade, modeling the behavior of social insects, such as ants and bees, for the purpose of search and problem solving has been the context of the emerging area of swarm intelligence. Using ant colony is a typical successful swarm-based optimization approach, where the search algorithm inspired by the behavior of real ants.

Kuo et al. [17] proposed a novel clustering method, ant K-means (AK) algorithm. AK algorithm modifies the K-means as locating the objects in a cluster with the probability, which updated by the pheromone, while the rule of updating pheromone is according to total within cluster variance (TWCV).

In [16] presents an ant colony optimization, methodology for optimally clustering N objects into k clusters. The algorithm employs distributed agents who mimic the way real ants find a shortest path from their nest to food source and back. They compared result with other heuristic algorithms in clustering, GA, Tabu search, SA. They showed that their algorithms are better than other algorithms in performance and time.

Honey-bees are among the most closely studied social insects. Honey-bee mating may also be considered as a typical swarm-based approach to optimization, in which the search algorithm is inspired by the process of marriage in real honey-bee. Honey bee has been used to model agent-based systems [1]. In a recent work, Abbass [8,9] developed an optimization algorithm based on the honey-bee marriage process.

Bozorg Haddad and Afshar [15] presented an optimization algorithm based on honeybee mating that successfully applied to a single reservoir optimization problem with discrete decision variables. Later, Bozorg Haddad et al. [14] applied the same algorithm to three benchmark mathematical problems.

In [2] the honey bee mating optimization algorithm is presented and tested with a nonlinear, continues constrained problem with continues decision and state variables to demonstrate the efficiency of the algorithm in handling the single reservoir operation optimization problems. They showed that the performance of the model is quite comparable with the result of the well-developed traditional linear programming solvers.

This paper presents application of honey-bee mating optimization (HBMO) algorithm for clustering. The paper organized as follow: in Section 2 we discussed cluster analysis problems. Section 3 introduce honeybee mating nature and application of honey-bee mating algorithm in clustering, and then in Section 4 experimental result of proposed clustering algorithm in comparison with other heuristics clustering algorithms showed.

2. Clustering

Data clustering, which is an NP-complete problem of finding groups in heterogeneous data by minimizing some measure of dissimilarity, is one of the fundamental tools in data mining, machine learning and pattern classification solutions [13]. Clustering in N-dimensional Euclidean space R^N is the process of partitioning a given set of n points into a number, say k, of groups (or, clusters) based on some similarity (distance) metric in clustering procedure is Euclidean distance, which derived from the Minkowski metric (Eqs. (1) and (2)).

$$\mathrm{d}(x,y) = \left(\sum_{i=1}^{m} |x_i - y_j|^r\right)^{1/r}, \tag{1}$$

$$\mathrm{d}(x,y) = \sqrt{\sum_{i=1}^{m} (x_i - y_j)^2}. \tag{2}$$

In this study, we will also use Euclidian metric as a distance metric. The existing clustering algorithms can be simply classified into the following two categories: hierarchical clustering and partitional clustering. The most class of popular class of partitional clustering methods is the center based clustering algorithms [23].

The K-means algorithms, is one of the most widely used center based clustering algorithms [6].

To find K centers, the problem is defined as an optimization (minimization) of a performance function, Perf(X, C), defined on both the data items and the center locations. A popular performance function for measuring goodness of the k clustering is the total within-cluster variance or the total mean-square quantization error (MSE), Eq. (3) [23]

$$\mathrm{Perf}(X,C) = \sum_{i=1}^{N} \mathrm{Min}\{\|x_i - c_l\|^2 | l = 1,\ldots,K\}. \tag{3}$$

The steps of the K-means algorithm are as follow [22]:

Step 1: Choose K cluster centers randomly from n points.
Step 2: Assign each point to clusters.
Step 3: Compute new cluster centers.
Step 4: If termination criteria satisfied, stop otherwise continues from step 2.

Note that in case the process close not terminates at step 4 normally, then it executed for a mutation fixed number of iterations.

3. Application of honey-bee mating optimization algorithm in clustering

3.1. Honey-bee modeling

A honeybee colony typically consists of a single egg-laying long-lived queen, anywhere from zero to several thousands drones (depending on the season) and usually 10,000–60,000 workers [9]. Queens are specialized in egg laying. A colony may contain one queen or more during its life cycle, which named monogynous and/or polygynous colonies, respectively. Only the queen bee is fed "royal jelly," which is a milky-white colored, jelly-like substance "Nurse bees" secrete this nourishing food from their glands, and feed it to their queen. The diet of royal jelly makes the queen bee bigger than any other bee in the hive. A queen bee may live up to 5 or 6 years, whereas worker bees and drones never live more than 6 months. There usually several hundred drones that live with the queen and worker bees. Mother Nature has given the drones' just one task, which is to provide the queen with some sperm. After the mating process, the drones die [10].

Drones are the fathers of the colony. They are haploid and act to amplify their mothers' genome without altering their genetic composition, except through mutation. Therefore, drones considered as agents that propagate one of their mother's gametes and function to enable females to act genetically as males. Workers specialized in brood care and sometimes lay eggs. Broods arise either from fertilized or unfertilized eggs. The former represent potential queens or workers, whereas the latter represent prospective drones [2].

In the marriage process, the queen(s) mate during their mating flights far from the nest. A mating flight starts with a dance performed by the queen who then starts a mating flight during which the drones follow the queen and mate with her in the air. In each mating, sperm reaches the spermatheca and accumulates there to form the genetic pool of the colony [19].

Each time a queen lays fertilized eggs, she randomly retrieves a mixture of the sperm accumulated in the spermatheca to fertilize the egg [18].

The queen is pursued by a large swarm of drones (drone comets), when copulation occurs. Insemination ends with the eventual death of the drone, and the queen receiving the "mating sign." The queen mates multiple times but the drone, inevitably, only once. These features make bee mating the most spectacular mating among insects [2].

The mating flight may considered as a set of transitions in a state-space (the environment) where the queen moves between the different states in some speed and mates with the drone encountered at each state probabilistically. At the start of the flight, the queen initialized with some energy content and returns to her nest when the energy is within some threshold from zero to full spermatheca. [2].

In developing the algorithm, the functionality of workers is restricted to brood care and therefore, each worker may be represented as a heuristic which acts to improve and/or take care of a set of broods (i.e., as feeding the future queen with royal jelly). A drone mates with a queen probabilistically using an annealing function as follows [9]:

$$\mathrm{Prob}(Q,D) = \exp[-\Delta(f)/S(t)], \tag{4}$$

where Prob (Q,D) is the probability of adding the sperm of drone D to the spermatheca of queen Q (that is, the probability of a successful mating); $\Delta(f)$ is the absolute difference between the fitness of D (i.e., $f(D)$) and the fitness of Q (i.e., $f(Q)$); and $S(t)$ the speed of the queen at time t. It is apparent that this function acts as an annealing function, where the probability of mating is high when the queen is still at the beginning of her mating flight, therefore her speed is high, or when the fitness of the drone is as good as the queen'. After each transition in space, the queen's speed and energy decays according to the following equations:

$$S(t+1) = \alpha(t) \times S(t), \tag{5}$$

where α is a factor $\in [0,1]$ and is the amount of speed reduction after each transition.

Workers that used to improve the brood's genotype may represent a set of different heuristics. The rate of improvement in the brood's genotype, as result of a heuristic application to that brood, defines the heuristic fitness value.

The fitness of the resulting genotype is determined by evaluating the value of the objective function of the brood genotype and/or its normalized value. It is important to note that a brood has only one genotype.

Thus, an HBMO algorithm maybe constructed with the following five main stages:

- The algorithm starts with the mating flight, where a queen (best solution) selects drones probabilistically to form the spermatheca (list of drones). A drone then selected from the list randomly for the creation of broods.
- Creation of new broods (cluster centers) by crossover the drone's genotypes with the queens.
- Use of workers (heuristics) to conduct local search on broods (trial solutions).
- Adaptation of worker's fitness, based on the amount of improvement achieved on broods.
- Replacement of weaker queens by fitter broods.

The algorithm starts with three user-defined parameters and one predefined parameter. The predefined parameter is the number of workers (W), representing the number of heuristics encoded in the program. However, the predefined parameter may be used as a parameter to alter the number of active heuristics if required; that is, the user may choose the first heuristic, where W is less than or equal to the total number of heuristics encoded in the program. The three user-defined parameters are the number of queens, the queen's spermatheca size representing the maximum number of mating per queen in a single mating flight, and the number of broods that will be born by all queens. The speed of each queen at the start of each mating flight initialized at random. A set of queens then initialized at random. A randomly selected heuristic then used to improve the genotype of each queen, assuming that a queen is usually a good bee. A number of mating flights are undertaken. In each mating flight, all queens fly based on the speed of each, where speed generated at random for each queen before each mating flight commences. At the start of a mating flight, a drone generated randomly and the queen positioned over that drone. The transition made by the queen in space based on her speed that represents the probability of flipping each gene in the drone's genome. At the start of a mating flight, the speed may be higher and the queen may make very large steps in space. While the energy of the queen decreases, her speed decreases, and as a result, the neighborhood covered by the queen, decreases. At each step in the space, the queen mates with the drone encountered at that step using the probabilistic rule in Eq. (4). If the mating is successful (i.e., the drone passes the probabilistic decision rule), the drone's sperm is stored in the queen's spermatheca. To sum up, the algorithm starts with a mating flight where a queen selects a drone with a predefined probabilistic rule. By cross-overing the drone's genotypes with the queen's, a new brood (trial solution) is formed which later can be improved, employing workers to conduct local search.

When all queens complete their mating flight, they start breeding. For a required number of broods, a queen selected in proportion to her fitness and mated with a randomly selected sperm from her spermatheca. A worker chosen in proportion to its fitness to improve the resultant brood. After all broods have been generated, they are sorted according to their fitness. The best brood replaces the worst queen until there is no brood that is better than any of the queens are. Remaining broods then killed and a new mating flight begins until all assigned mating flights are completed or convergence criteria met. The main steps of the HBMO algorithm presented in Fig. 1. In addition, a full-scale computational flowchart illustrated in Fig. 2.

3.2. Application of HBM algorithm in clustering

The search capability of HBM algorithm used in this article for the purpose of appropriately determining a fixed number of K cluster centers in R^N; thereby suitably clustering the set of n unlabelled points the clustering metric that has been adopted is the sum of the Euclidean distance of the points of the points from their respective cluster centers. The steps of the proposed algorithm are shown in Fig. 2, there are now described in detail.

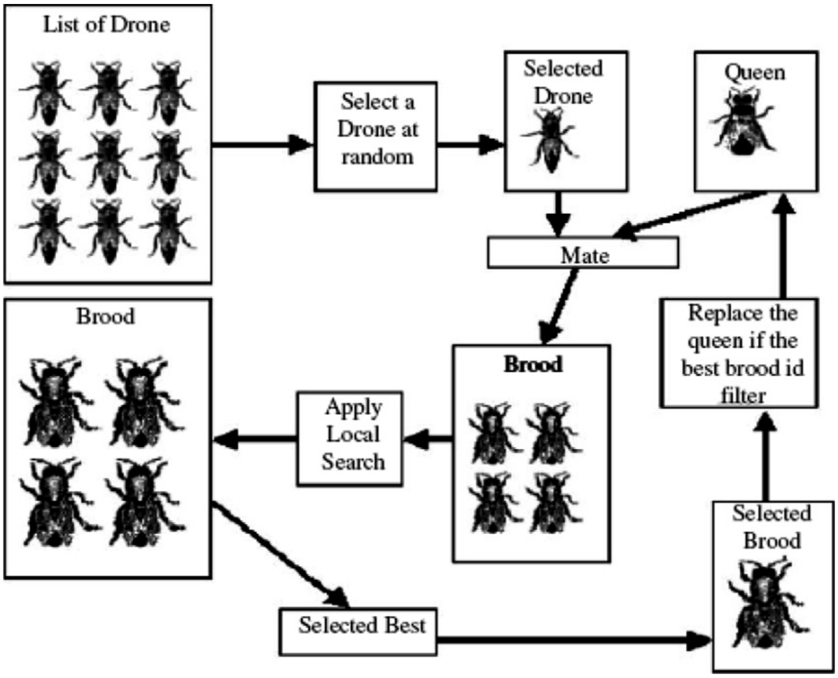


Fig. 1. The HBMO algorithm [9].

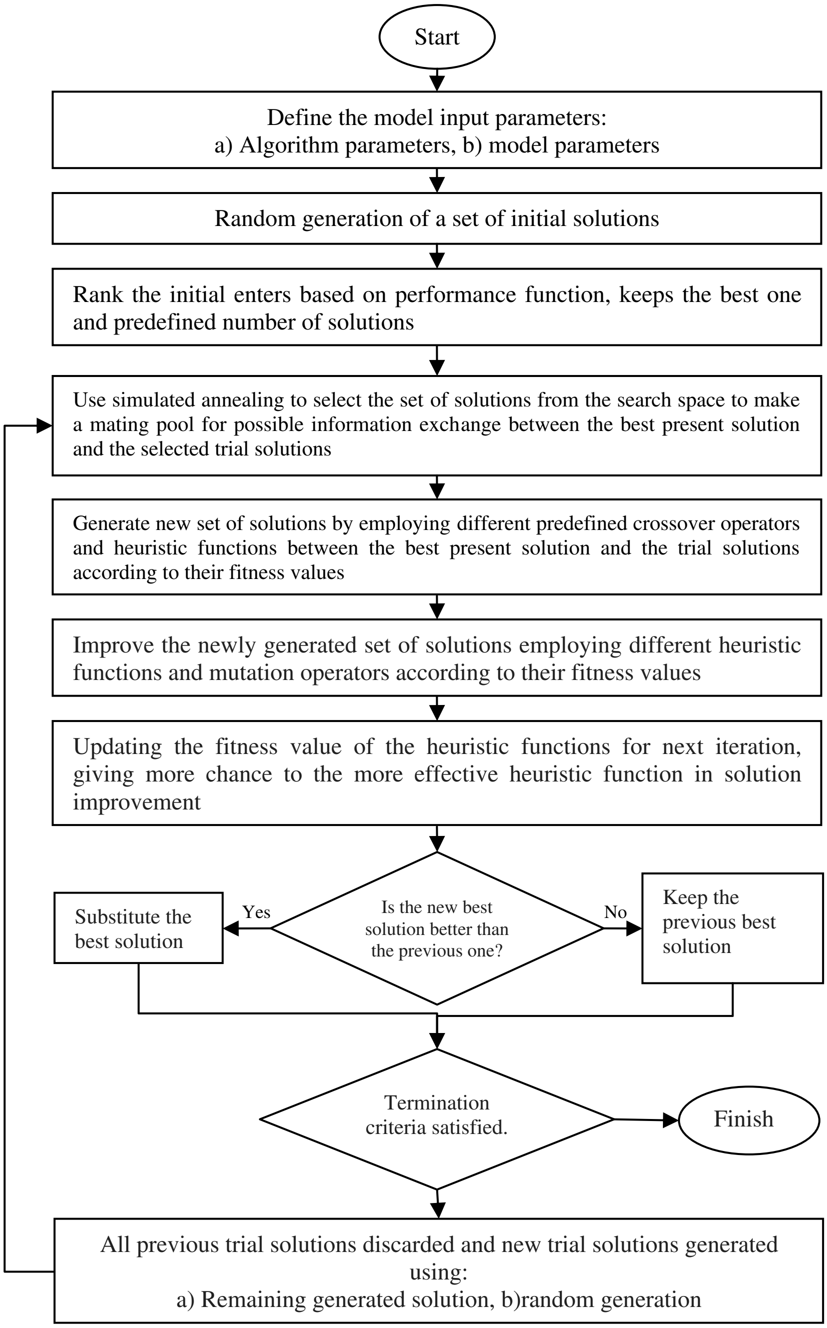

Start
Define the model input parameters:
a) Algorithm parameters, b) model parameters
Random generation of a set of initial solutions
Rank the initial enters based on performance function, keeps the best one and predefined number of solutions
Use simulated annealing to select the set of solutions from the search space to make a mating pool for possible information exchange between the best present solution and the selected trial solutions
Generate new set of solutions by employing different predefined crossover operators and heuristic functions between the best present solution and the trial solutions according to their fitness values
Improve the newly generated set of solutions employing different heuristic functions and mutation operators according to their fitness values
Updating the fitness value of the heuristic functions for next iteration, giving more chance to the more effective heuristic function in solution improvement
Is the new best solution better than the previous one?
Yes
Substitute the best solution
No
Keep the previous best solution
Termination criteria satisfied.
Finish
All previous trial solutions discarded and new trial solutions generated using:
a) Remaining generated solution, b)random generation

Fig. 2. HBM algorithm representation.

Step 1: String representation

A chromosome has used to represent a candidate solution to a problem where each gene in the chromosome represents a parameter of the candidate solution. In this study, a chromosome regarded as a set of K initial cluster centers and each gene is a cluster center dimension. Specifically, a chromosome can be represented as $C = [c_1 \ldots c_j \ldots c_K]$ where c_j is the jth gene and K is total number of genes. Fig. 3, illustrate a chromosome encoding example. $C_1 = (2,5,1), C_2 = (6,3,2),\ C_3 = (5,7,4)$.

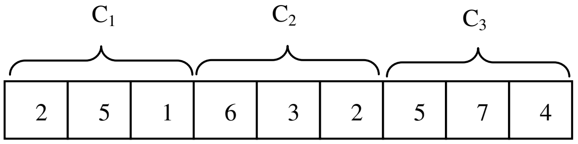


Fig. 3. A chromosome-encoding example.

Step 2: Define the model inputs parameters

The algorithm starts with three user-defined parameters and one predefined parameter. The predefined parameter is the number of worker (W), representing the number of heuristics encoded in the program. However, the predefined parameter may be used as a parameter to alter the number of active heuristics if required; that is, the user may choose the first heuristic, where W is less than or equal to the total number of heuristics encoded in the program.The user-defined parameters are number of queens, the queen's spermatheca size representing the maximum number of broods that will be born by all queens. The speed of each queen at the start of each mating flight initialized at random,

Step 3: Random generation of a set of initial solutions

In this stage, a set of initial cluster centers generated randomly from the dataset points. Each solution represents K cluster centers as shown in Fig. 1.

Step 4: Selection of queen

After generation of a set of initial solutions mentioned in the previous stage, rank the initial solutions based on performance function and set the best one as queen.

Step 5: Mating flight

Use simulated annealing to select the set of solutions from the search space to make a mating pool for possible information exchange between the best present solution (queen) and the selected trial solutions.

Step 6: Breeding process

Generate new set of solutions by employing predefined crossover operators and heuristic functions between the present solutions and the trial solution according to their fitness values. In this study, we adopt intermediate crossover. It creates children by taking a weighted average of parents. You can specify the weights by a single parameter, Ratio, which can be a scalar or a raw vector of length number of variables. The default is a vector of all 1's. The function creates the child from parent 1 and parent 2 using the following formula.

$$\text{Child} = \text{parent } 1 + \text{ran} * \text{Ratio} * (\text{parent } 2 - \text{parent } 1). \tag{6}$$

If all the entries of Ratio lie in the range [0, 1], the children produced are within the hypercube defined by placing the parents apposite vertices. If ratio is not in that range, the children might lie outside the hypercube. If ratio is a scalar, then all the children lie on the line between the parents.

Step 7: feeding selected broods and queen with the royal jelly by workers

Improve the newly generated set of solutions employing different heuristic functions and mutation operators according to their fitness values. For binary representation of chromosomes, a bit position mutated by simply flipping its value. Since we are considering floating point representation in this article, we use the following mutation. A number δ in the range [0, 1] generated with uniform distribution. If the value at a gene position is v, after mutation it becomes:

$$v \pm 2 * \delta * v, \quad v \neq 0,$$
$$v \pm 2 * \delta, \quad v = 0.$$

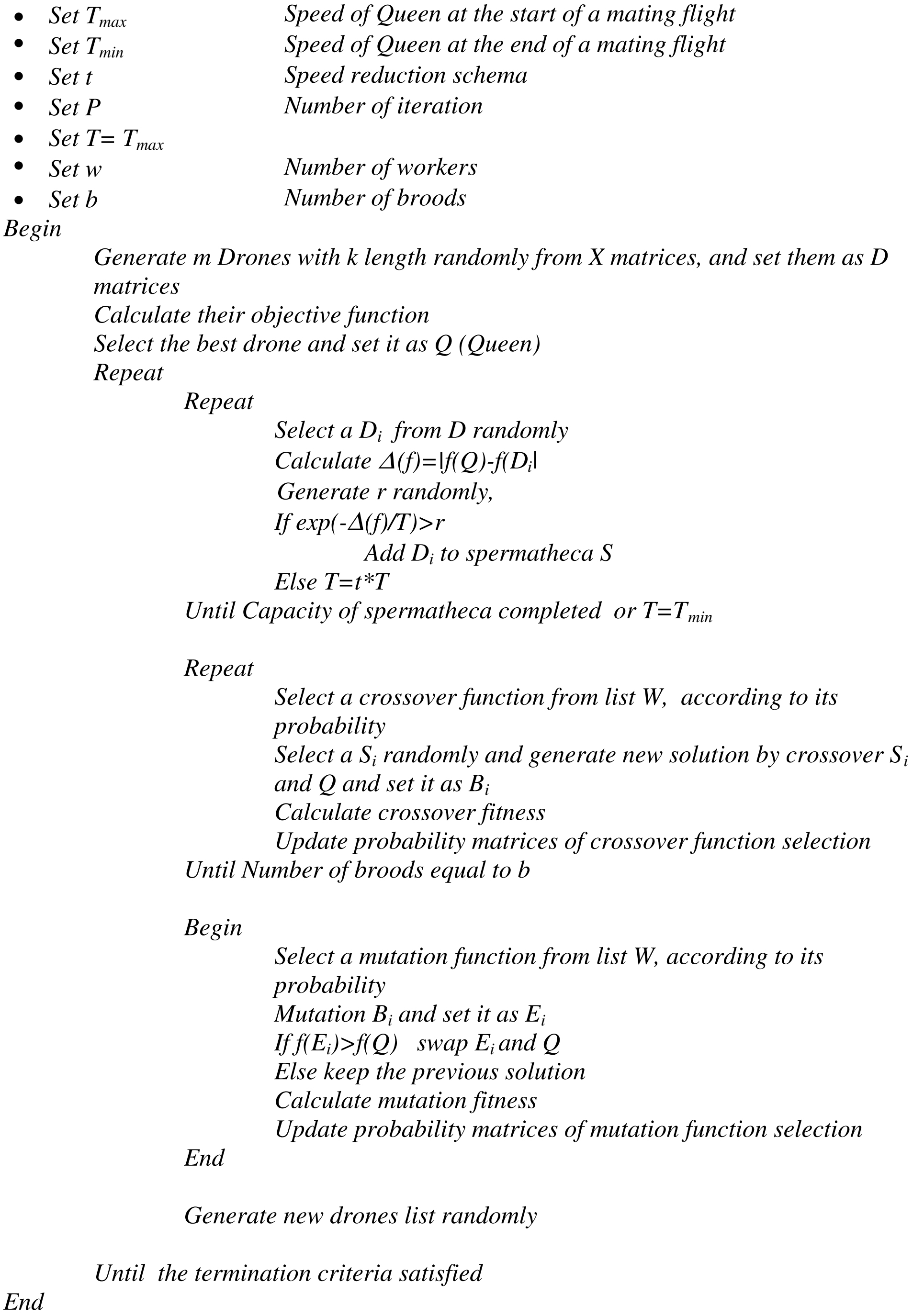
The $+$ or $-$ sign occurs with equal probability. Note that we could have implemented mutation as $v \pm \delta * v$.

However, one problem with this form is that if the values at a particular position in all the chromosomes of a population become positive (or negative), then we will never be able to generate a new chromosome having a negative (or positive) value at the position. In order to overcome this limitation, we have incorporated a factor of 2 while implementing mutation other form like

$$v \pm (\delta + \varepsilon) * v,$$

where $0 < \varepsilon < 1$ would also have satisfied our purpose.

Step 8: If the new best solution is, better than the queen replace it with queen.



```
• Set $T_{max}$            Speed of Queen at the start of a mating flight
• Set $T_{min}$            Speed of Queen at the end of a mating flight
• Set t                    Speed reduction schema
• Set P                    Number of iteration
• Set $T= T_{max}$
• Set w                    Number of workers
• Set b                    Number of broods
Begin
    Generate m Drones with k length randomly from X matrices, and set them as D
    matrices
    Calculate their objective function
    Select the best drone and set it as Q (Queen)
    Repeat
        Repeat
            Select a $D_i$ from D randomly
            Calculate $\Delta(f)=|f(Q)-f(D_i|$
            Generate r randomly,
            If $exp(-\Delta(f)/T)>r$
                Add $D_i$ to spermatheca S
            Else $T=t*T$
        Until Capacity of spermatheca completed  or $T=T_{min}$

        Repeat
            Select a crossover function from list W,  according to its
            probability
            Select a $S_i$ randomly and generate new solution by crossover $S_i$
            and Q and set it as $B_i$
            Calculate crossover fitness
            Update probability matrices of crossover function selection
        Until Number of broods equal to b

        Begin
            Select a mutation function from list W, according to its
            probability
            Mutation $B_i$ and set it as $E_i$
            If $f(E_i)>f(Q)$   swap $E_i$ and Q
            Else keep the previous solution
            Calculate mutation fitness
            Update probability matrices of mutation function selection
        End

        Generate new drones list randomly

    Until  the termination criteria satisfied
End
```

Fig. 4. Pseudo code for HBM clustering Algorithm.

Step 9: Check the termination criteria
If the termination criteria satisfied finish the algorithm, else discard all previous trial solutions and generate new trial solutions. Then go to step 5 until all assigned iteration (mating flights) completed or convergence criteria met.

The pseudo code for application of HBM algorithm in clustering illustrate in Fig. 4.

4. Experimental result

In this section, we present a set of experiments that shows goodness of our algorithm. We have done our experiments on a Pentium IV, 2.8 GHz, 512 GB RAM computer and we have coded with Matlab 7.0 software. We run all five algorithms on three different datasets. The datasets are all well-known iris, wine and breast-cancer datasets taken from Machine Learning Laboratory [4].

DataSet1: This is the Iris data set, which is perhaps the best-known database to found in the pattern recognition literature. Fisher's paper is a classic in the field and referenced frequently to this day. The data set contains three classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are not linearly separable from each other. There are 150 instances with four numeric attributes in iris data set. There is no missing attribute value. The attributes of the iris data set are; sepal length in cm, sepal width in cm, petal length in cm and petal width in cm.

DataSet2: This is the wine data set, which also taken from MCI laboratory. These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. There are 178 instances with 13 numeric attributes in wine data set. All attributes are continuous. There is no missing attribute value.

DataSet3: This is the Wisconsin Breast Cancer Database (January 8, 1991) which also taken from MCI laboratory. There are 699 instances with 10 numeric attributes in data set. Attributes 2 through 10 have been used to represent instances. Each instance has one of two possible classes: benign or malignant. There are 16 missing attribute values in data set.

To evaluate the performance of the HBM algorithm in clustering, we have compared it with several typical stochastic algorithms including the ACO algorithm [16], the simulated annealing approach [18], the genetic algorithms [3], and the tabu search approach [12]. The effectiveness of stochastic algorithms is greatly

Table 1
Result obtained by the five algorithms for 10 different runs on dataset 1

Method	Function value			Function evaluation	CPU time (s)
	F_{best}	F_{average}	F_{worst}		
HBM	96.752047	96.95316	97.757625	11214	35.25
ACO	97.100777	97.171546	97.808466	10998	33.72
GA	113.986503	125.197025	139.778272	38128	105.53
TS	97.365977	97.868008	98.569485	20201	72.86
SA	97.100777	97.134625	97.263845	29103	95.92

Table 2
Result obtained by the five algorithms for 10 different runs on dataset 2

Method	Function Value			Function Evaluation	CPU time (s)
	F_{best}	F_{average}	F_{worst}		
HBM	16257.284378	16257.284378	16257.284378	7238	55.18
ACO	16530.533807	16530.533807	16530.533807	9306	68.29
GA	16530.533807	16530.533807	16530.533807	33551	226.68
TS	16666.226987	16785.459275	16837.535670	22716	161.45
SA	16530.533807	16530.533807	16530.533807	7917	57.28

dependent on the generation of initial solutions. Therefore, foe every dataset, algorithms performed 10 times individually for their own effectiveness tests, each time with randomly generated initial solutions.

The comparison of results for each dataset based on the bet solution found in 10 distinct runs of each algorithm, the average number of evaluations required and the convergence processing time taken to attain the best solution.

The solution quality is also given in terms of the average and worst values of the clustering metric (Favg, F$_{worst}$, respectively) after 10 different runs for each of the five algorithms. Tables 1–3 show these results.

For clustering problem, on iris dataset results given in Table 1, show that the HBM provide the optimum value of 96.752047, the ACO and SA methods obtain 97.100777. The HBM was able to find the optimum 8 times as compared to that of nine times obtained by ACO and five times obtained by SA. The ACO required the least number of function evaluation (10998) and the processing time (33.72). The result obtained for the clustering problem, dataset 2 given in Table 2. The HBM find the optimum solution of 16246.352467 and the ACO, SA and GA methods provide 16530.533807. The HBM, ACO, SA and GA methods found the optimum solution in all their 10 runs. The function evaluation and the execution time taken by the HBM algorithm are less than other algorithms.

Table 3
Result obtained by the five algorithms for 10 different runs on dataset 3

Method	Function value			Function evaluation	CPU time (s)
	F_{best}	F_{average}	F_{worst}		
HBM	10099.297958	10100.13	10107.646802	23268	99.35
ACO	10111.827759	10112.126903	10114.819200	25626	102.15
GA	10116.294861	10128.823145	10148.389608	45003	153.24
TS	10249.72917	10354.315021	10438.780449	29191	114.01
SA	10111.827759	10114.045265	10115.934358	28675	108.22

Table 4
Values of parameters of each of five algorithms

HBM		ACO		GA		TS		SA	
Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
Number of queens	1	Ants (*R*)	50	Population size	50	Tabu list size	25	Probability threshold	0.98
Number of drones	150	Probability threshold for maximum trail (*q*0)	0.98	Crossover rate	0.8	Number of trial solutions	40	Initial temperature	5
Capacity of spermatheca	50	Local search probability (*p*ls)	0.01	Mutation rate	0.001	Probability threshold	0.98	Temperature multiplier	0.98
Speed of Queen at first of flight	Randomly ∈ [0.5 1]	Evaporation rate (ρ)	0.01	Maximum number of iterations	1000	Maximum number of iterations	1000	Final temperature	0.01
Minimum speed of Queen	Randomly ∈ [0 1]	Maximum number of iterations (*itermax*)	1000					Number of iterations detect steady state	100
Speed reduction schema (cooling schema)	0.98							Maximum number of iterations	30000
Cross over ratio	1.5								
Maximum number of iterations	1000								

The HBM algorithm for the human thyroid disease dataset, provide the optimum solution of 10099.297958 to this problem with success rate of 90% during 10 runs. In terms of the function, evaluations and the processing time the HBM performed better than other clustering algorithms as can observed from Table 3.

Shelokar et al. [16] performed several simulations to find the algorithmic parameters that result into the best performance of ACO, GA, SA and TS algorithms in terms of the equality of solution found, the function evaluations and the processing time required. In this study, we used their algorithmic parameters. In addition, we performed several simulations to find the algorithmic parameters for HBM algorithm. Algorithmic parameters for all algorithms illustrated in Table 4.

The result illustrate that the proposed HBM optimization approach can be considered as a viable and an efficient heuristic to find optimal or near optimal solutions to clustering problems of allocating N objects to k clusters.

5. Conclusion

In summary, honey-bee mating optimization algorithm to solve clustering problems developed in this paper. Honey-bees are among the most closely studied social insects. Honey-bee mating may also be considered as a typical swarm-based approach to optimization, in which the search algorithm is inspired by the process of marriage in real honey-bee.

Honey bee has been used to model agent-based systems. The HBM algorithm for data clustering can be apply when the number of clusters known a priori and are crisp in nature. To evaluate the performance of the HBM algorithm, it compared with other stochastic algorithms viz. ant colony, genetic algorithm, simulated annealing and tabu search. The algorithm implemented and tested on several real datasets; preliminary computational experience is very encouraging in terms of the quality of solution found, the average number of function evaluation and the processing time required.

References

[1] A. Perez-Uribe, B. Hirsbrunner, Learning and foraging in robot-bees, in: SAB 2000 Proceedings Supplement Book, International Society for Adaptive Behavior, Honolulu, Hawaii, (2000) pp. 185–194.

[2] A. Afshar, O. Bozog Haddad, M.A. Marino, B.J. Adams, Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation, Journal of the Franklin Institute (2006).

[3] C.A. Murthy, N. Chowdhury, Pattern Recogn. Lett. 17 (1996) 825–832.

[4] C.L. Blake, C.J. Merz, UCI repository of machine learning databases. Available from: <http://www.ics.uci.edu/-mlearn/MLRepository.html>.

[5] C.S. Sung, H.W. Jin, A tabu-search-based heuristic for clustering, Pattern Recogn. 33 (2000) 849–858.

[6] E.W. Forgy, Cluster analysis of multivariate data: efficiency versus interpretability of classifications, Biometrics 21 (3) (1965) 768–769.

[7] H. Spath, Clustering Analysis Algorithms, Ellis Horwood, Chichester, UK, 1989.

[8] H.A. Abbass, A monogenous MBO approach to satisfiability, in: Proceeding of the International Conference on Computational Intelligence for Modelling, Control and Automation, CIMCA'2001, Las Vegas, NV, USA, 2001.

[9] H.A. Abbass, Marriage in honey-bee optimization (MBO): a haplometrosis polygynous swarming approach, in: The Congress on Evolutionary Computation (CEC2001), Seoul, Korea, May 2001, (2001) pp. 207–214.

[10] H.H. Laidlaw, R.E. Page, Mating designs, in: T.E. Rinderer (Ed.), Bee Genetics and Breeding, Academic Press Inc., NY, 1986, pp. 323–341.

[11] K. Krishna, Murty, Genetic k-means Algorithm, IEEE Trans. Syst. Man Cybernet. B Cybernet. 29 (1999) 433–439.

[12] K.S. Al-Sultan, Tabu-search-based heuristic for clustering, Pattern Recogn. 28 (1995) 1443–1451.

[13] M.R. Garey, D.S. Johnson, H.S. Witsenhausen, The complexity of the generalized Lloyd–Max problem, IEEE Trans. Inform. Theory 28 (2) (1982) 255–256.

[14] O. Bozorg Haddad, A. Afshar, M.A. Mariñ o, Honey bees mating optimization algorithm (HBMO); a new heuristic approach for engineering optimization, in: Proceeding of the First International Conference on Modeling, Simulation and Applied Optimization (ICMSA0/05), Sharjah, UAE, 1–3 February (2005).

[15] O. Bozorg Haddad, A. Afshar, MBO (Marriage Bees Optimization), A new heuristic approach in hydro systems design and operation, in: Proceedings of 1st International Conference On Managing Rivers In The 21st Century: Issues and Challenges, Penang, Malaysia, 21–23 September 2004, pp. 499–504.

[16] P.S. Shelokar, V.K. Jayaraman, B.D. Kulkarni, An ant colony approach for clustering, Anal. Chim. Acta 509 (2004) 187–195.

[17] R.I. kuo, H.S. Wang, Tung-Lai Hu, S.H. Chou, Application of ant K-means on clustering analysis, Comput. Math. Appl. 50 (2005) 1709–1724.

[18] R.E. Page, The evolution of multiple mating behaviors by honey-bee queens (*Apis mellifera* L.), J. Genet. 96 (1980) 263–273.

[19] R.F.A. Moritz, E.E. Southwick, Bees as Super organisms, Springer, Berlin, Germany, 1992.

[20] S.Z. Selim, M.A. Ismail, K-means type algorithms: a generalized convergence theorem and characterization of local optimality, IEEE Trans. Pattern Anal. Mach. Intell. 6 (1984) 81–87.

[21] Shokri Z. Selim, K. Al-Sultan, A simulated annealing algorithm for the clustering problem, Pattern Recogn. 24 (10) (1991) 1003–1008.

[22] Ujjwal Mualik, Sanghamitra Bandyopadhyay, Genetic algorithm-based clustering technique, Pattern Recogn. 33 (2000) 1455–1465.

[23] Zulal Gungor, Alper Unler, K-harmonic means data clustering with simulated annealing heuristic, Appl. Math. Comput. (2006).