FOUNDATIONS OF COMPUTING AND DECISION SCIENCES
Vol. 41 (2016) No. 2

DE GRUYTER OPEN

ISSN 0867-6356
e-ISSN 2300-3405

DOI: 10.1515/fcds-2016-0006

AN OPTIMIZED K-HARMONIC MEANS ALGORITHM COMBINED WITH MODIFIED PARTICLE SWARM OPTIMIZATION AND CUCKOO SEARCH ALGORITHM

Asgarali Bouyer *

Abstract. Among the data clustering algorithms, k-means (KM) algorithm is one of the most popular clustering techniques due to its simplicity and efficiency. However, k-means is sensitive to initial centers and it has the local optima problem. K-harmonic-means (KHM) clustering algorithm solves the initialization problem of k-means algorithm, but it also has local optima problem. In this paper, we develop a new algorithm for solving this problem based on an improved version of particle swarm optimization (IPSO) algorithm and KHM clustering. In the proposed algorithm, IPSO is equipped with Cuckoo Search algorithm and two new concepts used in PSO in order to improve the efficiency, fast convergence and escape from local optima. IPSO updates positions of particles based on a combination of global worst, global best with personal worst and personal best to dynamically be used in each iteration of the IPSO. The experimental result on five real-world datasets and two artificial datasets confirms that this improved version is superior to k-harmonic means and regular PSO algorithm. The results of the simulation show that the new algorithm is able to create promising solutions with fast convergence, high accuracy and correctness while markedly improving the processing time.

Keywords: k-means; k-harmonic means clustering; Particle swarm optimization; Lévy flight; Local Minimum

1. Introduction

Data clustering is a popular data mining technique, applied for extracting the reasonable organization of objects in a given dataset. This technique classifies the similar objects into different groups, or more precisely, the partitioning of a dataset into subsets, in which each part (subset) has some similarities and common characters. In fact, a set of patterns are gathered into clusters based on similarity among each cluster. Clustering is an important technique applied in many application domains including document clustering [19]fraud

* Faculty of Information Technology and Communications, Azarbaijan Shahid Madani University, Tabriz, Iran, *a.bouyer@azaruniv.edu*

detection [18], flow shop scheduling [29], machine learning [3], wireless mobile sensor networks [31], biomedical data [13], image processing [49], demand forecast [42]and financial classifications [34]. Many data clustering algorithms have been presented in the previous literatures with different approaches. Clustering algorithms can be generally divided into two groups: hierarchical algorithms and partitional algorithms. Hierarchical clustering algorithms recursively find nested clusters either in agglomerative or in divisive [20]; and partitional algorithm divides the datasets into some clusters whose members have nothing in common with each other [17, 25, 44]. The most popular and extensively used algorithm among partitioning algorithms is K-means algorithm. It easily clusters the large data set with best runtime. However, the results of K-means algorithm are very sensitive to positions of the initial cluster centers in the problem space[51]. It also has local optimum problem[21] and does not have any criterion for computing the number of clusters. K-Harmonic Means (KHM) is an alternative algorithm to solve the sensitivity to initialization problem of K-means methods [52]. This algorithm minimizes the harmonic average from all points in N to all centers in K. This approach proposes more robust results than K-means with different initial configurations. KHM solves the problem of initialization using a built-in boosting function[48]. However, it easily runs into local optima like K-means algorithm. To overcome the shortcomings of the KM and KHM algorithms some heuristic algorithms have been combined with these methods.

Recently, evolutionary and meta-heuristics like genetic algorithms[14], Ant Colony Optimization[39, 40], Artificial Bee Colony[24, 47], Particle Swarm Optimization (PSO)[8], Bacterial Foraging Optimization (BFO) [37], Cuckoo Search optimization[46] and other optimization algorithms have been hybridized with standard and basic clustering algorithms including K-means, Fuzzy K-means (FKM) and KHM to reach the required quality and performance in clustering process. These algorithms try to solve the weaknesses of the KM and KHM algorithms. However, they also put up with several limitations. For example, Tabu search and simulated annealing algorithms suffer from low quality results and low convergence speed problems [15]. Yang and Deb proposed Cuckoo Search (CS) algorithm via Levy flight in 2009 [46]. Cuckoo Search via Levy flights is based on the interesting breeding behavior such as brood parasitism of certain species of cuckoos. The basic ideas applied are the aggressive reproduction strategy of cuckoo and usage of Levy flights. Cuckoo Search algorithm is being widely used in engineering optimization problems [2] with exceptionally good results. PSO is a versatile population based stochastic optimization technique. The algorithm maintains a population of particles, where each particle represents a potential solution to an optimization problem. In the regular PSO [26], the diversity loss is mainly due to the strong desirability of the global best particle, which results in quick converge of all particles on local or global optimum where the global best particle locates [43]. In addition, PSO and ACO have the convergence problem.

Based on the weaknesses of KHM such as tendency to obtain local optima, this work investigates a new approach to solve the clustering problem, using a combination of KHM clustering with improved version of PSO (IPSO), and cuckoo search optimization. The goal of this combination is to obtain solutions that are closer to the global optima of the problem. The proposed method, called KHM-IPSO, maintains good convergence of PSO to global optimum because it uses cuckoo algorithm generation along with improved version of PSO to escape from local optimum. In IPSO, we have added two new concepts which Gworst as the “global worst” and Pworst as the “personal worst”. These two concepts add a new mutation phase to PSO to escape from local optimum with high probability. Furthermore, in

IPSO, the best generation of cuckoo is used by PSO to reach the global optimum with fast convergence. Generally, in most clustering algorithms, the main goals are to meet the required quality in clusters such as accuracy, processing time, standard deviation parameters, and F-measure [28]. In this paper, the following performance metrics is used in the comparative analysis: 1) the accuracy of final clustering results and 2) the speed of convergence. The results were obtained with five real datasets and two artificial data sets. The proposed approach was generally better than the other algorithms tried on the experiments. Based on the experimental results, it is found that the proposed KHM-IPSO performs cluster analysis with better quality and performance in comparison to PSO, KHM, and PSOKHM algorithms.

The rest of this paper is organized as follows: Section 2 briefly presents the current related works in clustering analysis and combined methods. The Cuckoo Search via Levy flight, regular PSO, and KHM algorithm are represented in Section 3. In section 4, the proposed KHM-IPSO clustering algorithm is explained. Section 5 shows the described experiment setting and results. Finally, the conclusion is presented in Section 6.

2. Related works

The particle swarm optimization (PSO) has been used for clustering in many studies. An efficient hybrid clustering based on fuzzy PSO, ACO and K-means algorithms, called FAPSO-ACO–K, is presented in[36]. The obtained results of this method are very notable in performance improvement of information clustering. PSOKHM data clustering algorithm proposed a hybrid algorithm based on KHM and PSO [48]. This algorithm solves the KHM's local optima problem and PSO's slow convergence speed. MOIMPSO clustering algorithm is a hybrid of multi-objective clustering algorithm and PSO that was presented to obtain a single best solution from the Pareto optimal archive [35]. By combining two genetic and PSO algorithms, Kao et al. invented a new method in which it has benefitted from jump and junction operator for genetic[22]. This approach could solve different problems of continual functions. In addition, significant changes have been obtained in finding the response to general optimization and convergence ratio. They also combined K-means algorithm, Nelder–Mead simplex search, and particle swarm optimization, called K–NM–PSO [23]. The K–NM–PSO searches for cluster centers of an arbitrary dataset as does the KM algorithm, but it can effectively find the global optima. They used *k*-means algorithm alone to generate one particle in the initial population. It implemented a Nelder–Mead search only on the best $m + 1$ particles in each iteration, where m is the number of attributes, and then the rest of the population is moved toward the best particle of the whole population and toward the best neighbor. Merwe et al. used PSO algorithm to solve *k*-means clustering problem. The algorithm is extended to use *k*-means clustering to seed the initial swarm [45].

FC-MOPSO is another research that combined multi-objective particle swarm (MOPSO) approach with Fuzzy Clustering technique [4]. In FC-MOPSO, the migration concept is used to exchange information among different sub-swarms and to ensure their diversity. A new approach based on PSO and Radial basis function (RBF) neural networks, PSO–OSD, has been developed in [12]. PSO–OSD used PSO algorithm, which is not sensitive to the initial values of the cluster centers. Chuang et al., combined chaotic map particle swarm optimization (CPSO) with an accelerated convergence rate strategy, and introduced this accelerated chaotic map particle swarm optimization (ACPSO) in their research [9].

Accelerated chaotic particle swarm optimization (ACPSO) searches through arbitrary datasets for appropriate cluster centers. Yang et al. introduced a hybrid method (called PSOKHM) based on combining PSO and *k*-harmonic means to enhance the global search ability of their algorithm [48]. PSOKHM repeats KHM four times in each generation for which it employs eight generations to improve particles within the population. Furthermore, PSO algorithm repeats eight times in each generation. A new approach for clustering based on particle swarm optimizer for dynamic optimization problems, CPSO, is presented in [7].CPSO employs hierarchical clustering method to track multiple peaks based on a nearest neighbor search strategy. Serkan et al., proposed a PSO algorithm and Fractional Global Best Formation (FGBF) technique for multidimensional search in dynamic environment [27]. GAI-PSO method is the combination of PSO, Genetic Algorithm (GA), and K-means algorithm to find global optimum and the fast convergence [1]. GAI-PSO algorithm searches the solution space to find the optimal initial cluster centroids for the next phase. The next phase is a local refining stage utilizing the *k*-means algorithm that can efficiently converge to the optimal solution. GSOKHM algorithm is another method that has been presented to improve the efficiency of KHM using PSO algorithm and genetic algorithm (GA) [11]. Xin-She et.al., has applied Cuckoo Search (CS) algorithm for clustering [46]. They have evaluated CS with GA and PSO using standard benchmark functions. In their study, CS algorithm is used with Levy flight and is found to be performing better compared to the other two methods. ICAKHM is a novel method on the basis of hybrid K-harmonic means algorithm and an improved version of the imperialist competitive algorithm (ICA) [2]. This version of ICA method uses the genetic operators of crossover and mutation to prevent the premature convergence helping KHM to evade local optima problem similar to many other evolutionary algorithms. The evaluation result of ICAKHM method [33] reveals that its results are often suitable. However, this algorithm usually is unstable and its result may or may not be improved. HCSDE is another method that focuses on the partitional clustering algorithm. HCSDE benefits from DE operators such as mutation and crossover to produce better solutions instead of random solutions in standard CS algorithm. It also improves the standard CS algorithm in terms of convergence speed and accuracy [6]. However, this method is not more stable than PSOKHM and ICAKHM. In addition, [33] presents a survey of the relevant literature in this field.

We have compared our proposed algorithm with PSO, KHM, PSOKHM and ICAKHM methods in section 5.

3. The regular Cuckoo, PSO and KHM clustering algorithms

The k-means clustering problem is to divide the *n* instances or objects into *k* clusters with the clusters partitioning the instances ($x_1 \ldots x_n$) into the subsets. Each subset is a $n_k \times m$ matrix, where n_k shows the number of assigned instances to cluster Kth*(K:1..k).* Therefore, each solution has K subsets that each subset has its particular instances. K-means uses a vector C ($C_1 \ldots C_k$) where C_i is the centroid of the i^{th} cluster. Data Clustering is aimed to find out a reasonable organization for the instances of a given dataset by identifying and quantifying similarities or dissimilarities among the instances [32]. In fact, the best clustering algorithm is the one that divides a dataset into clusters where the instances of each cluster have the most similarities with each other and the least similarities with instances of other clusters. The goal of data clustering is to minimize the objective function, in this case a squared error function [41]. The squared error function shows the total intra-cluster variance as objective of K-Means clustering. The minimized squared error means the high quality clustering. Equation (1) indicates the squared error function for k-means clustering.

$$f(k) = \sum_{k=1}^{K} \sum_{i=1}^{n_k} (x_i - C_k)^2 \tag{1}$$

Among the different solution of k-means, if the error function *f (k)* is minimum for a solution S_j, then the solution S_j is the best result. It means that S_j has the best clusters. Generally, k-means clustering consists of two main steps:

Repeat Steps 1 & 2:

Step 1) The assignment step: instances are placed in the closest cluster as defined by the distance function (Euclidean distance).

Step 2) The re-estimation step: the cluster centroids are recalculated from the instances assigned to the cluster according to Equation(2).

$$C_k = \frac{1}{n_k} \sum_{i=1}^{n_k} x_i \tag{2}$$

Until *convergence criterion is met.*

Where *k=1,2,...K* is the number of clusters, *f(k)* is the objective function, x_i, i=1,2,... n_k is the patterns in the *k*th cluster, n_k is the number of instances in cluster *k*, and C_k is center of the *k*th cluster.

In the improved PSO algorithm (IPSO), we have also used a one-dimensional array to encode cluster centers as particles. Every particle or candidate solution in the population consists of a one-dimensional array with the length of *d* × *k* cells to show all cluster centers. KHM-IPSO tries to find an optimal partitions of *k* well-separated clusters.

Before explaining the proposed hybrid method (KHM-IPSO), we have briefly discussed the regular PSO, *k*-harmonic means algorithm, and cuckoo search via Lévy Flight for immediate reference.

3.1. Particle Swarm Optimization

Particle swarm optimization (PSO), introduced by Kennedy and Eberhart in 1995 [26], uses a group of cooperating particles where each particle represents a candidate solution for exploring solutions which can allows for optimizing problems. It is a population-based optimization tool and can be carried out easily to solve various optimization problems. In PSO, a swarm of particles “fly” through the search space. Each particle follows the previous best position found by its neighbor particles and the previous best position found by itself[33]. Particles move through an n-dimensional search space. Each particle *i* maintains a record of the position of its previous best performance in a vector called *pbest*. Initial positions and velocities of the particles are chosen randomly. Each particle's position is updated at each iteration step according to its own personal best position and the best solution of the swarm. When a particle takes the entire population as its topological neighbors, the best value is a global best and is called *gbest*. All particles can share information about the search space. Representing a possible solution to the optimization problem, each particle moves in the direction of its best solution and the global best position discovered by any particles in the swarm. The particles in the swarm update their velocities according to the equation (3):

$$v_i^{(t+1)} \leftarrow \omega v_i^{(t)} + c_1 rand_1 (pbest_i^{(t)} - x_i^{(t)}) + c_2 rand_2 (gbest^{(t)} - x_i^{(t)}) \qquad (3)$$

Where x_i (t) is the position of the i^{th} particle at the *t* moment and v_i (*t*) is the velocity of the i^{th} particle at the *t* moment. The factor ω is the inertia weight that denotes a proportion of the previous velocity, *pbest* is the best position of the particle, and *gbest* is the global best position of the swarm that has been found by the whole population so far. In addition, $rand_1$and$rand_2$ are variables ranging random values between 0 and 1. The constants$c_1$andc_2 are positive constants that determine the impact of the personal best solution and the global best solution on the search process, respectively.

The new position of a particle is updated using the equation (4):

$$x_i^{(t+1)} \leftarrow v_i^{(t+1)} + x_i^{(t)}) \qquad (4)$$

The updating of the particle position is performed with Eq.(4). Both equations (3) and (4) are iterated until convergence of the search process is reached. PSO algorithm is very fast, simple and easy to understand and implement. Nevertheless, it has some shortcomings. PSO gives good results and accuracy for single objective optimization, but for multi objective problem, it stuck into local optima. Another PSO problem is its nature to a fast and premature convergence in mid optimum points[38].

3.2. K-Harmonic-Means clustering

K-means (KM) is one of the classical and widely used approaches due to its advantages such as simplicity and fastness. However, the main drawback of k-means is sensitivity to initial starting points (initial centers). Since k-means select initial starting points randomly, the results (solutions) are not trustful. In each running of k-means, the result may be different from previous results. K-harmonic means (KHM) algorithm was therefore proposed in 1999 to solve the high sensitivity problem to initial centers[52]. KHM algorithm uses harmonic average of distances from each data point to the cluster center instead of the minimum distance in k-means algorithm. The basic KHM algorithm is shown as follows:

X = (x_1,. . . ,x_n): the data to be clustered.

C = (c_1, . . . , c_k): the set of cluster centers.

$m(c_j \mid x_i)$: The membership function defining the proportion of data point that belongs to center c_j.

$w(x_i)$: The weight function defining how much influence data point x_i has in re-computing the center parameters in the next iteration.

STEPS:

1. Initialize the algorithm with guessed centers C, i.e., randomly choose the initial centers.
2. Calculate objective function value according to equation (5):

$$KHM(X,C)=\sum_{i=1}^{n}\frac{k}{\sum_{j=1}^{k}\frac{1}{\left\|x_i-c_j\right\|^{p}}} \tag{5}$$

Where p is an input parameter and typically p≥2.

3. For each data point x_i, compute its membership $m(c_j \mid x_i)$ in each center c_j according to equation (6):

$$m(c_j \mid x_i)=\frac{\left\|x_i-c_j\right\|^{-p-2}}{\sum_{j=1}^{k}\left\|x_i-c_j\right\|^{-p-2}} \quad , \; m(c_j \mid x_i)\in[0,1]. \tag{6}$$

4. For each data point x_i, compute its weight $w(x_i)$ according to equation (7):

$$w(x_i)=\frac{\sum_{j=1}^{k}\left\|x_i-c_j\right\|^{-p-2}}{\left(\sum_{j=1}^{k}\left\|x_i-c_j\right\|^{-p}\right)^{2}} \tag{7}$$

5. For each center c_j, re-compute its location from all data points x_i according to their memberships and weights using equation (8):

$$c_j=\frac{\sum_{j=1}^{n}m(c_j \mid x_i).w(x_i).x_i}{\sum_{j=1}^{n}m(c_j \mid x_i).w(x_i)} \tag{8}$$

6. Repeat steps 2–5 predefined number of iterations or until KHM(X,C) does not change significantly.
7. Assign data point x_i to cluster j with the biggest $m(c_j \mid x_i)$.

Objective function of KHM algorithm introduces conditional probability of cluster center to data points and dynamic weights of data points at each iteration. Due to employing the membership function $m(c_j \mid x_i)$, KHM algorithm is particularly useful when the boundaries of clusters are ambiguous and not well-separated.

KHM algorithm alleviate the weakness of k-means algorithm. However, like KM, KHM still converges to local optimum. To solve this problem, we need to use combinatorial optimization problems. In this paper, KHM is combined with Cuckoo search optimization and an improved version of PSO algorithms. Since Cuckoo search algorithm is a part of this study, it is explained in detail in the following section.

3.3. Cuckoo Search via Lévy flight

To deal with the fast convergence of PSO to local optima, we have used the cuckoo search algorithm (CS) with PSO. The CS is an optimization algorithm based on the breeding pattern of parasitic cuckoos [46]. This optimization algorithm has been confirmed to deliver outstanding performance in function optimization. The proposed IPSO algorithm combines the advantages of CS and PSO and overcomes the main disadvantage of PSO, easily becoming trapped in the local minima, which performs the local search faster than the PSO. In the multi-dimensional space where the optimal solution is sought, the CS is carried out for a maximization problem, where the quality or fitness of a solution can simply be proportional to the value of the objective function. The CS is based on three idealized rules:

1. Each cuckoo lays a single egg into a randomly chosen host nest from among *n* nests;
2. The nests with better quality eggs (implying better fitness value of the function concerned), if not detected, would be hatched to grow into the cuckoo chicks, who would join the next generation
3. The number of available host's nests is fixed, and the host bird discovers the egg laid with a probability p_a. The worst nests are discovered and dumped from further calculations.

Based on these three rules, the basic steps of the Cuckoo Search (CS) can be summarized as the pseudo code shown in Fig 1. When generating new solutions $x_i^{(t+1)}$ from the old one ($x_i^{(t)}$), Lévy flight is performed for a cuckoo *i*th with the parameter 1<λ<3 as follow:

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus \text{Lévy}(\lambda) \qquad (9)$$

$$Lévy{\sim}u = t^{-\lambda} \qquad 1 < \lambda < 3 \qquad (10)$$

where α>0 is the step size that should be related to the scales of the problem of interests. In most cases, we can use α=1. The product $\oplus$ means entry wise multiplications. This entry wise product is similar to those used in PSO, but here the random walk via Lévy flight is more efficient in exploring the search space as its step length is much longer in the long run. The Lévy flight essentially provides a random walk while the random step length is drawn from a Lévy distribution (Equation 10). Pseudo-code of CS is shown in Figure 1.

```
Cuckoo search algorithm
        Input: CN as stopping criterion
        Output: optimal fitness value
    1.  Objective function f(x), where $x = (x_1, x_2, ..., x_d)^T$
        Generate initial population of n host nests $x_i$ (i = 1, 2, .., n)
        (One of the host nests is produced by the KHM at the first time)
    2.  Set cycle to 1
    3.  Repeat
            a.  Randomly select a cuckoo by Levy flight using Eq.3
            b.  Calculate its fitness value ($F_c$) by the objective function
            c.  Randomly select a nest
            d.  Calculate its fitness value ($F_n$) by the objective function
            e.  If ($F_c$ <$F_n$) then Replace the nest with the cuckoo
            f.  A fraction $p_a$ of nest are replaced by new nests
            g.  Calculate fitness and keep best nests
            h.  Rank the solutions and find the current best
            i.  Store the best nest as optimal fitness value
    4.  cycle=cycle+1
    5.  Until cycle≤CN
```

Figure 1. Pseudo-code of Levy flight Cuckoo Search Algorithm

Levy distribution has infinite variance and infinite mean with a power-law step size of a heavy tail, shown in Figure 2. It chooses a random nest from the given population of nests and evaluates its fitness function using Equation 3. If the fitness of the new solution is better than the older one then it replaces the older one with the new one. The best nests with the fittest egg (solution) are carried-on to the next generation. It should be kept in mind as a warning that the system must not be trapped in a local optimum. For this reason, a fraction of the new solutions must be cropped up from far field randomization with locations far enough from the current best solution [46].

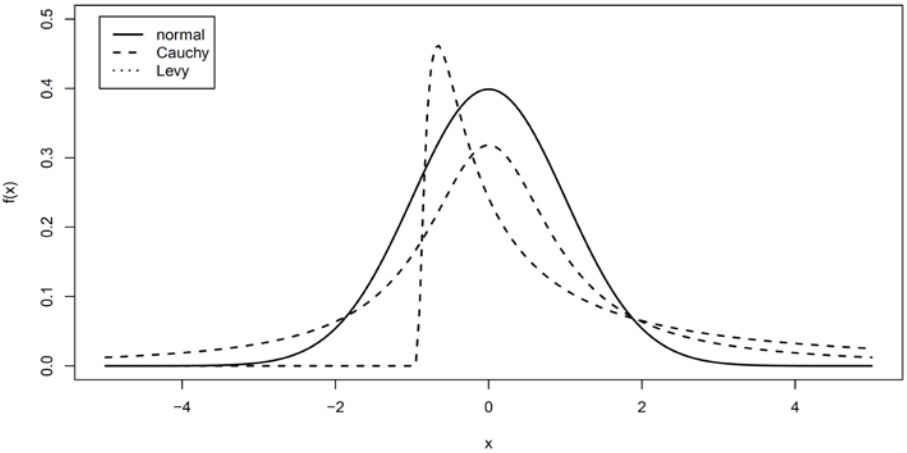


Figure 2. The Levy, Cauchy, and Normal distribution [30].

4. The proposed clustering algorithm

In this section, we describe an improved clustering algorithm based on an improved version of particle swarm optimization (IPSO) algorithm and *k*-harmonic means, called KHM-IPSO. The improved particle swarm optimization (IPSO) provides a partition of data points without any prior knowledge. Meanwhile, the KHM algorithm can obtain high-quality initializations from the IPSO, and provides better input to IPSO to accelerate its convergence.

4.1. Improved particle swarm optimization

In regular PSO, we had global best and personal best, which were the best value for objective function found by all particles in the swarm and the best value for objective function found by every particle so far. Improved particle swarm optimization (IPSO) uses two new concepts in comparison to basic or regular PSO,: 1) *Gworst*: the worst point in the current population or “global worst” and 2)*Pworst*: the worst point in the memory of each particle" or “personal worst”. The global worst is the fitness value of that candidate solution which has the worst value for objective function (maximum value in minimization problems). This value is found by all particles in the swarm. The second concept is the worst place that every particle of the population has seen during their move. These concepts have been differently used in our previous research work with different impact and objective functions[16]. In IPSO, the position of particles depend on their own current worst solution and their group's previous worst solution. In IPSO algorithm, the *Gbest* with *Gworst* and *Pbest* with *Pworst* can dynamically be used instead of one another. For each particle, the worst fitness values, *Pworst*, and for the whole swarm, *Gworst*, are computed in each iteration. Figure 3 shows a particle.

X_{11}	X_{12}	...	X_{1d}	...	X_{k1}	X_{k2}	...	X_{kd}

Figure 3. The representation of a particle.

The best previous position of the particle in i^{th} iteration is calculated as:

$Pworst_i(t) = [Pworst^t_{i,1}, Pworst^t_{i,2},...., Pworst^t_{i,m}]$ m is the number of solutions

Also the *Pbest, Gbest* and *Gworst* is recalculated as follow:

$Pbest_i(t) = [Pbest^t_{i,1}, Pbest^t_{i,2},...., Pbest^t_{i,m}]$ m is the number of solutions

$Gbest_i(t) = [Gbest^t_1, Gbest^t_2,...., Gbest^t_m]$ m is the number of solutions

$Gworst_i(t) = [Gworst^t_1, Gworst^t_2,...., Gworst^t_m]$ m is the number of solutions

(get the worst in the whole particles till iteration i,)

At each iteration, after finding these specific positions, the particles move in the presence of two distinct steps. The logic of considered movements is to escape from bad points and areas, then converging to appropriate places. At the first step, which we call it acceleration step, particles find suitable area of search by moving away from unsuitable areas. In fact, this move cause particles to spread in search space and search for good solutions in a wide area, and in case of entering local optima they can bypass it. At the next step, which we also call it convergence step, all particles try to move towards global optimum based on their personal memory and best particle. From the mathematical inference of PSO, larger inertia weight performs more efficient global search and smaller one performs more effective local search. Thus, Shi and Eberhart [19] used Eq. (11) that decreases inertia weight with increasing the number of iterations linearly:

$$\omega_{(t+1)} = \omega_{MAX} - \frac{\omega_{MAX} - \omega_{Min}}{t_{MAX}} \times t \tag{11}$$

This equation has also been used in IPSO. The pseudo- code of the proposed IPSO algorithm is shown in Figure 4. In this algorithm $x_i(t)$ and $v_i(t)$ respectively show the position and the velocity of particle *i* at time or iteration *t*. $\omega v_i(t+1)$ and $Sv_i(t+1)$ respectively calculate the velocity of particle *i* based on *Pworst* and *Pbest* solution. The $Pbest_i(t)$ is the best position found by particle *i* which keeps the fitness value of best candidate solution encountered by considered particle so far. $Gbest_i(t)$ is the best position found by the whole swarm so far; and ω is an inertia weight scaling the previous time step velocity. The $Pworst_i(t)$ is the worst position found by particle *i* which keeps the fitness value of worst candidate solution. The c_1 and c_2 coefficients are two constant coefficients ([0,2]) which control the influence of the best personal position of the particle ($Pbest_i(t)$) and the best global position ($Gbest_i(t)$) where c1+c2≤4. $rand_1$, and $rand_2$ are random values in the range [0, 1]. *K* is a constriction factor that is for updating particle's flying velocity.

Through constriction factor, the algorithm can have better convergence and stability. ω_{MAX} and ω_{Min} are the maximum and minimum of the inertia weights, respectively, and *t* is the iteration counter.

```
Algorithm: Pseudo-code for IPSO
1. Initialize n particles (one of the particles is produced by KHM algorithm at the
   first time).
   - Set psize (population size), ω (the inertia factor), $c_1$, $c_2$ (the weight between
   the attraction to Pbest and Gbest).
2. Repeat
    a. Calculate fitness of each particle by the objective function
    b. Select the best particle of PSO (global best position) based on best fitness
       value($F_{best\_pso\_particle}$)
    c. Find a best nest by Cuckoo search optimization (algorithm in Fig.1)
    d. If ($F_{best\_pso\_particle}$<$F_{best\_Cuckoo\_nest}$) Then // Use the PSO generation
        i. Select solution from PSO
        ii. Store this solution as a best nest for Cuckoo search.
    e. else // Use the Cuckoo generation
        i. Select solution of best nest by Cuckoo search algorithm
        ii. Store this solution as a best particle of PSO
    f. global best position is the best fit particle
    g. Update the velocity and position for each particle $i^{th}$ based on following steps:
        i. $K = 2/|2-(c_1+c_2+\sqrt{(c_1+c_2)-4(c_1+c_2)}|)$
        ii. $\omega v_i(t+1) \leftarrow K[\omega v_i(t)+c_1 rand_1(x_i(t)-Pworst_i(t))+c_2 rand_2(x_i(t)-Gworst_i(t))]$
            $Sv_i(t+1) \leftarrow K[\omega v_i(t)+c_1 rand_1(pbest_i(t)-x_i(t))+c_2 rand_2(gbest(t)-x_i(t))]$
        iii. Update the velocity using: $v_i(t+1) = MAX\{\omega v_i(t+1), Sv_i(t+1)\}$
        iv. Update the position using: $x_i(t+1) \leftarrow v_i(t+1)+x_i(t)$
        v. Recalculate $Pbest_i(t)$, $Pworst_i(t)$, $Gbest_i(t)$, $Gworst_i(t)$ and $\omega_{(t+1)}$
        vi. For each particle if (fitness of current position < fitness of personal best) then
        vii. Personal best = current position
        viii. Update $Pbest_i(t)$, $Pworst_i(t)$, $Gbest_i(t)$, and $Gworst_i(t)$ if the new values are
              better than the old ones.
3. Until stopping criteria met
4. Global best position is retained (as a cluster centre to KHM algorithm).
```

Figure 4. Pseudo-code of Improved PSO using Cuckoo Search Algorithm

4.2. The proposed hybrid clustering algorithm

We have combined KHM and IPSO to form a hybrid-clustering algorithm (KHM-IPSO), which maintains the qualities of IPSO, CS and KHM besides solving their convergence and sensitivity problems. The pseudo code of the proposed KHM-IPSO algorithm is represented in Figure 5.

```
Algorithm: Pseudo-code for KHM-MPSO hybrid algorithm
 1. Set the initial parameters as follows:
      a. Set Max-Itr: maximum number of iterations (It often set to 10)
      b. Set psize (population size).
 2. Initialize a population of size Psize.
 3. Set iterative count count1= 0;
 4. Set iterative counts count2=0, count3=0;
 5. Execute the classic KHM algorithm for the first time
      a. Choosing the initial centers
      b. Apply KHM algorithm (in section 3.2)
      c. Assign the result as one of the particles for PSO and one of the host nests
         for Cuckoo.
         (Other particles and host nests are initialized randomly)
 6. Use the Improved PSO (IPSO) algorithm in Fig.3 to:
      a. Apply the IPSO operator to update the psize objects.
      b. count2 = count2 + 1. If count2 < 8, go to Step 6.a
 7.  (KHM Method) For each object i
      a. Take the result of IPSO algorithm as the initial cluster centers of the KHM
         algorithm.
      b. Recalculate each cluster center using the KHM algorithm.
      c. count3 = count3 + 1. If count3 < 4, go to Step 7.b.
 8. count1 = count1 + 1. If count1 < Max-Itr, go to Step4.
 9. Assign data point $x_i$ to cluster $j$ with the biggest $m(c_j|x_i)$.
```

Figure 5. The pseudo code of the proposed KHM-IPSO algorithm

The proposed algorithm is built based on two main steps whereat each step, only a type of a move is done by particles. First step is to escape from local optimums and migration away from unsuitable places in search space. The second stage is to converge to the global optimum. These two steps are repeated consecutively until termination criteria satisfied (e.g., maximum number of iteration achieved or no change occurs in certain number of iterations). KHM-IPSO applies KHM to the particles in the swarm every 10 generations such that the fitness value of each particle is improved. In the proposed algorithm, Cuckoo Search via Levy flight has been used which is efficient to find new suitable neighbors and better solutions [50]. Sometimes, the best particle of PSO is selected by Levy flight Cuckoo Search instead of PSO algorithm, if the objective function of the generated PSO solution is weaker than the generated solution by Levy flight Cuckoo. Furthermore, at the first iteration, IPSO and CS get an initial solution from KHM algorithm along with producing some random solutions. It helps IPSO and CS to improve their next generations.

5. Experimental results

We test our proposed algorithm on seven data sets and compare with other well-known algorithms. These data sets are five real data sets and two artificial data sets that are named as Iris, Wine, Wisconsin breast cancer (denoted as Cancer), Contraceptive Method Choice (denoted as CMC) and Ripley’s glass with different number of clusters, data objects and features for every data object [5]. These data sets cover low, medium and high dimensions. A brief description of these data sets is explained below.

5.1. The datasets

- **ArtSet1 (n = 300, d = 2, k = 3):** This is an artificial data set. It is a two-featured problem with three unique classes. A total of 300 patterns are drawn from three independent bivariate normal distributions, where classes are distributed according to

  $$N2 = \left(\mu = \begin{pmatrix} \mu_{i1} \\ \mu_{i2} \end{pmatrix}, \sum = \begin{bmatrix} 0.4 & 0.04 \\ 0.04 & 0.4 \end{bmatrix} \right), i = 1,2,3$$

  $\mu_{11} = \mu_{12} = -2, \quad \mu_{21} = \mu_{22} = 2, \quad \mu_{31} = \mu_{32} = 6$

  and μ being mean vector and Σ is the covariance matrix.
- **ArtSet2 (n = 300, d = 3, k = 3):**This is an artificial data set with three-features and three classes and 300 patterns, where every feature of the classes is distributed according to Class1~ Uniform (10, 25), Class2 ~ Uniform (25, 40), Class3 ~ Uniform (40, 55).
- **Iris Data set (N = 150, d = 4, K = 3):**This is perhaps the best-known database to be found in the pattern recognition literature. Fisher's paper is a classic in the field and is 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 two; the latter are NOT linearly separable from each other.
- **Wine Data set (N = 178, d = 13, K = 3):**These data are the results of a chemical analysis of wines grown in Institute of Pharmaceutical and Food Analysis and Technologies 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.
- **Wisconsin Breast Cancer Data set (N = 683, d = 9, K = 2):**In this data set, features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image. A few of the images can be found at http://www.cs.wisc.edu/~street/images/.
- **Ripley's glass Data set (N = 214, d = 9, K = 6):**The study of classification of types of glass was motivated by criminological investigation. At the scene of the crime, the glass left can be used as evidence, if it is correctly identified!
- **The Contraceptive Method Choice Data set (N = 1473, d = 10, K = 3):**The samples consist of the married women who were either not pregnant or not sure of their pregnancy at the time the interviews were conducted. It predicts the choice of the current contraceptive method (no contraception has 629 objects, long-term methods have 334 objects, and short-term methods have 510 objects) of a woman based on her demographic and socioeconomic characteristics.

5.2. Simulation setups

We compare the performance of the proposed algorithm on the selected datasets with traditional PSO, k-harmonic means, PSOKHM, and ICAKHM algorithms. The quality of solutions is compared by the sum of the intra cluster distances, i.e. the distances between data objects within a cluster and its center. It is clear, the smaller the sum of the distances is, the higher the quality of clustering. The parameters of the proposed algorithm are adjusted based on Table 2.

Table 1. The properties of five real data sets from UCI data repository. and two artificial data sets

	Number of Attributes	Number of Classes	Missing Data	Number of Instances
ArtSet1	2	3	No	300
ArtSet2	3	3	No	300
Iris	4	3	No	150
Wine	13	3	No	178
Wisconsin Breast Cancer	30	2	No	569
Ripley's Glass	9	6	No	214
CMC	9	3	No	1437

Table 2. Simulation setups for PSO parameters

C1	C2	ω_{Min}	ω_{MAX}
1	1	0.4,0.3	0.9, 1
1.5	1	0.4,0.5	0.9, 1
1	1.5	0.4	0.9, 1
2	1	0.4	0.9, 1
2	2	0.4	1, 0.9, 0.8, 0.7
2	2	0.3	1, 0.9, 0.8
2	2	0.2	1, 0.9, 0.8

In the simulation process, the c_1 and c_2 is adjusted with different values between [1, 2]; and ω_{Min} , ω_{MAX} is set with different values between [0, 1]. For instance, Table 2 shows the various assigned values for these parameters. These parameters have just been tested on Iris and Wine datasets to find appropriate values. The obtained results show that in all PSO based algorithm the best setup for these parameters is: c_1=2, c_2=2, $\omega_{Min} = 0.4$, and $\omega_{MAX} = 0.9$. Therefore, in the following comparisons, we evaluate our proposed algorithm (KHM-IPSO***)*** with other algorithms based on this mentioned setting. These algorithms are implemented using Matlab 2012, and evaluated based on the following measures:

1- The most common quality measurement for clustering algorithms is the F-Measure criterion[10]. The F-Measure uses the ideas of precision and recall from information retrieval. In other words, F-measure is provided to show the clustering accuracy of the algorithms. The higher the F-measure, the better the clustering due to the higher accuracy of the resulting clusters mapping to the original classes. Each class *i* (as given by the class labels of the used benchmark data set) is regarded as the set of n_i items desired for a query; each cluster j (generated by the algorithm) is regarded as the set of nj items retrieved for a query; n_{ij} gives the number of elements of class *i* within cluster *j*. For each class *i* and cluster *j* F-Measure, precision (*p*), and recall (*r*) are defined as follow:

$$F(i,j) = ((b^2+1).P(i,j).r(i,j))/(b^2P(i,j)+r(i,j)) \qquad (12)$$

Where b = 1 to obtain equal weighting for P and r

$$\mathrm{p(i,\, j) = (n_{ij}/n_{j})} \qquad (13)$$

$$\mathrm{r(i,\, j) = (n_{ij}/n_{j})} \qquad (14)$$

$$\mathrm{F} = \sum_{\mathrm{i}} \frac{\mathrm{n_i}}{\mathrm{n}} MAX_j \{F(i,j)\} \qquad \text{(overall F - Measure)} \qquad (15)$$

Clearly, the larger value for F-Measure reveals the better quality for a clustering algorithm.

2- The average standard deviation, *stdev,* is another criterion measure that is defined as follows:

$$tdev = \frac{1}{c}\sqrt{\sum_{i=1}^{n_c} \| \sigma(v_i) \|} \qquad (16)$$

Where *c* is the number of clusters and v_i is the center of cluster *i*th.

3- Objective function value in best, average and worst values: Best is the minimum objective function value among all runs, average is the average objective function value of all runs, Worst is the is the maximum value among all times. The smaller value for objective function is the higher the quality of clustering algorithm.

5.3. Performance evaluation

To compare the performance of our algorithm with those of other approaches, each algorithm is 100 times for each of the datasets and averaged at the end. The simulation results are demonstrated in Table 3.

The simulation results given in Tables 3 show that KHM-IPSO and PSOKHM are very precise, and on average, KHM-IPSO is more precise than PSOKHM. Furthermore, in all other datasets, our algorithm has a small standard deviation compared to the other algorithms except cancer data set(for PSOKHM). For instance, the results obtained on the Iris dataset show that KHM-IPSO converges to the global optimum of 96.6228 in most of runs whereas the best solutions of KHM, PSO, ICAKHM and PSOKHMare97.8396, 98.7741, 96.6362 and 96.6301, respectively. Additionally, the obtained best, average and worst solutions of KHM, PSO, PSOKHM, and KHM-IPSO algorithms indicate that the KHM-IPSO is the best one for all data sets, except cancer dataset. Nevertheless, the obtained results for best and average solutions by PSOKHM algorithm are good and close to KHM-IPSO's results, whereas worst solution of KHM-IPSO is high quality than other algorithms. In short, KHM-IPSO has minimum values of KHM function in Iris, Wine, CMC and glass datasets.

Table 3. Simulation results of 100run of following clustering algorithms (p=2.5)

Data set	***Criteria***	***KHM***	***PSO***	***ICAKHM***	***PSOKHM***	***KHM-IPSO***
Iris	Best solution	97.8396	98.7741	96.6362	96.6301	**96.6228**
	Average	102.235	99.1629	96.6664	96.6355	**96.6323**
	Worst solution	108.4184	102. 9339	96.6919	96.6630	**96.6382**
	stdev	13.2517	0.3882	**0.01055**	0.09128	**0.01055**
	F-Measure	0.8853	0.8861	0.356710	0.8891	**0.8924**
Wine	Best solution	16552.38	16,344.38	16293.9	16,297.17	**16,293.15**
	Average	18057.74	16415.51	16295.6	16302.59	**16293.43**
	Worst solution	18560.84	16560.82	16296.94	16314.37	**16293.69**
	stdev	789.998	82.55	1.002372	0.62	**0.49**
	F-Measure	0.669	0.6781	0.6802	0.671	**0.6885**
Cancer	Best solution	2989.72	2964.50	2962.42	**2961.98**	2962.10
	Average	3233.46	3029.21	3022.81	**3024.47**	3024.49
	Worst solution	3545.81	3338.66	3150.15	3149.82	**3148.90**
	stdev	250.1	108.11	0.396	**0.380**	**0.380**
	F-measure	0.9617	0.9339	0.841	0.9617	**0.9647**
CMC	Best solution	5,847.88	5,701. 53	5,699.2183	5,698.73	**5,691.16**
	Average	5,899.48	5,822.94	5,705.1485	5,700.04	**5,694.41**
	Worst solution	5,942.06	5,918.93	5721.1779	5,702.11	**5,695.72**
	stdev	47.16	46. 96	1.268275	0.92	**0.81**
	F-measure	0.45034	0.4633	0.4446	0.4524	**0.4731**
Glass (Ripley's glass)	Best solution	215.23	271.63	199.86	199.47	**199.425**
	Average	234.95	276.85	202.41	199.503	**199.438**
	Worst solution	257.541	284.912	209.778	199.549	**199.452**
	stdev	12.465	4.551	0.26	0.141	**0.139**
	F-measure	0.6637	0.6429	0.6695	0.6648	**0.6835**

On the other hand, the simulation results of Table 3 show that the F-Measure of the proposed algorithm absolutely is better than those of obtained by others in all data sets. It reveals that the clusters are spatially well separated by to KHM-IPSO algorithm.

The standard deviation (stdev) of the proposed algorithm is less than the other algorithms. It means that, KHM-IPSO can find optimal solutions in most of the cases, while other algorithms may trap in local optima. Moreover, it often can find high quality solutions compared to the other algorithms. The best standard deviation in Iris dataset (with low dimensional) belongs to ICAKHM and our proposed KHM-IPSO algorithm. The standard deviation of the fitness function for these algorithms is 0.01055 in Iris dataset, which is significantly less than other methods. However, ICAKHM does not have better stdev in all other datasets. For Cancer dataset (with high dimensional) the PSOKHM and KHM-IPSO algorithms has better stdev than other algorithms. Furthermore, KHM-IPSO algorithm has better stdev than other algorithms in Wine, CMC and Glass datasets.

In general, the simulation results shown in Table3indicate that the proposed KHM-IPSO algorithm converges to the global optimum with an improved standard deviation and less function evaluations. This leads logically to the end that the KHM-IPSO is a feasible and a robust clustering algorithm.

Table 4. Obtained results for PSOKHM method on seven data sets for p = 2.5, p=3, and p=3.5 based on KHM(X,C), F-Measure, and Runtimes (for 100 independent run).

	PSOKHM Algorithm		
	PSOKHM(p=2.5)	**PSOKHM(p=3)**	**PSOKHM(p=3.5)**
ArtSet1			
KHM(X,C)	703.509 (0.050)	741.3861 (0.0023)	806.644 (0.0074)
F-measure	**1.000 (0.000)**	**1.000 (0.000)**	**1.000 (0.000)**
Runtime	0.7054 (0.0097)	**0.7052 (0.0048)**	0.7043 (0.0044)
ArtSet2			
KHM(X,C)	109525.941 (0.152)	256953.240(13.183)	679549.738 (283.234)
F-measure	**1.000 (0.000)**	**1.000 (0.000)**	**1.000 (0.000)**
Runtime	0.7461 (0.00440)	**0.7411 (0.00532)**	0.7363 (0.00443)
Iris			
KHM(X,C)	149.521122 (0.220046)	126.356025 (0.051715)	**111.496432 (0.371611)**
F-measure	0.889365 (0.001704)	0.891125 (0.000616)	0.890476 (0.000822)
Runtime	0.776514 (0.008286)	0.785798 (0.008178)	0.782373 (0.013236)
Wine			
KHM(X,C)	75642795.261(123127.311)	1075350475.505 (5934548.867)	15938236000.160 (375608578.016)
F-measure	**0.678695 (0.008791)**	0.647009 (0.008415)	0.631343 (0.007597)
Runtime	**1.198114 (0.006380)**	1.199792 (0.008373)	1.200532 (0.009848)
CMC			
KHM(X,C)	96730.543291 (205.878047)	187530.512796 (209.278023)	385242.257966 (1077.221514)
F-measure	0.464650 (0.003114)	0.454853 (0.003122)	0.455401 (0.004125)
Runtime	7.942413 (0.013390)	7.892877 (0.033972)	8.032246 (0.034444)
Cancer			
KHM(X,C)	**57167.360619 (0.626255)**	**113703.834625 (6.098736)**	232149.835544 (25.711355)
F-measure	0.961290 (0.000216)	0.964719 (0.000)	0.965644 (0.000188)
Runtime	**2.846137 (0.014304)**	2.848792 (0.026659)	2.860135 (0.012080)
Glass			
KHM(X,C)	1242.219883 (9.641556)	1741.945932 (19.074922)	2251.847572 (90.469432)
F-measure	0.647040 (0.019853)	0.663190 (0.017553)	**0.672230 (0.016469)**
Runtime	2.176758 (0.023808)	2.194541 (0.011367)	2.175558 (0.011030)

Due to the close similarity between PSOKHM and the proposed algorithm, we compare them considering more details that are presented in Tables 4 and 5. Tables 4 and 5 are the results of the objective function KHM(X,C),F-measure and Runtime criteria which are in accordance with different p values p = 2.5, p=3 and p = 3. The tables show means and standard deviations (in brackets) for 100 independent runs. Bold face indicates the best result out of the two algorithms.

Table 5. Obtained results for proposed KHM-IPSO clustering on seven data sets for p = 2.5, p=3, and p=3.5 based on KHM(X,C), F-Measure, and Runtimes (for 100 independent run).

Proposed KHM-IPSO Algorithm			
	KHM-IPSO (p=2.5)	**KHM-IPSO (p=3)**	**KHM-IPSO (p=3.5)**
ArtSet1 **KHM(X,C)** **F-measure** **Runtime**	**668.703 (0.0061)** **1.000 (0.000)** **0.7503 (0.0071)**	**711.365(0.2372)** **1.000 (0.000)** 0.7142 (0.0046)	**762.157 (0.301)** **1.000 (0.000)** **0.6959 (0.0040)**
ArtSet2 **KHM(X,C)** **F-measure** **Runtime**	**108046.437 (0.1762)** **1.000 (0.000)** **0.7397 (0.00466)**	**256469.498 (9.686)** **1.000 (0.000)** **0.7411 (0.0053)**	**646197.154072 (198.852)** **1.000 (0.000)** **0.7325 (0.00493)**

Iris
KHM(X,C)
F-measure
Runtime
	149.046166(0.047562)
	0.892268(0.001425)
	0.294792 (0.007)
		126.279684 (0.062980)
		0.891129(0.000616)
		0.281934(0.004)
			111.496936(0.472)
			0.891775(0.000)
			0.301901(0.008)

Wine
KHM(X,C)
F-measure
Runtime
	74944203.249(121621.908)
	0.678521(0.009)
	0.908909(0.005)
		1066602515.373(3911744.742)
		0.647854(0.010)
		0.9061102(0.007)
			15668540757.021(263948606)
			0.631675(0.006)
			1.044160(0.019)

CMC
KHM(X,C)
F-measure
Runtime
	96569.572990(125.758)
	0.464754(0.003)
	5.861913(0.012)
		187350.068520(132.273)
		0.464096(0.002978)
		5.910834(0.020)
			383568.825231(329.816)
			0.462089(0.002998)
			6.093088(0.077)

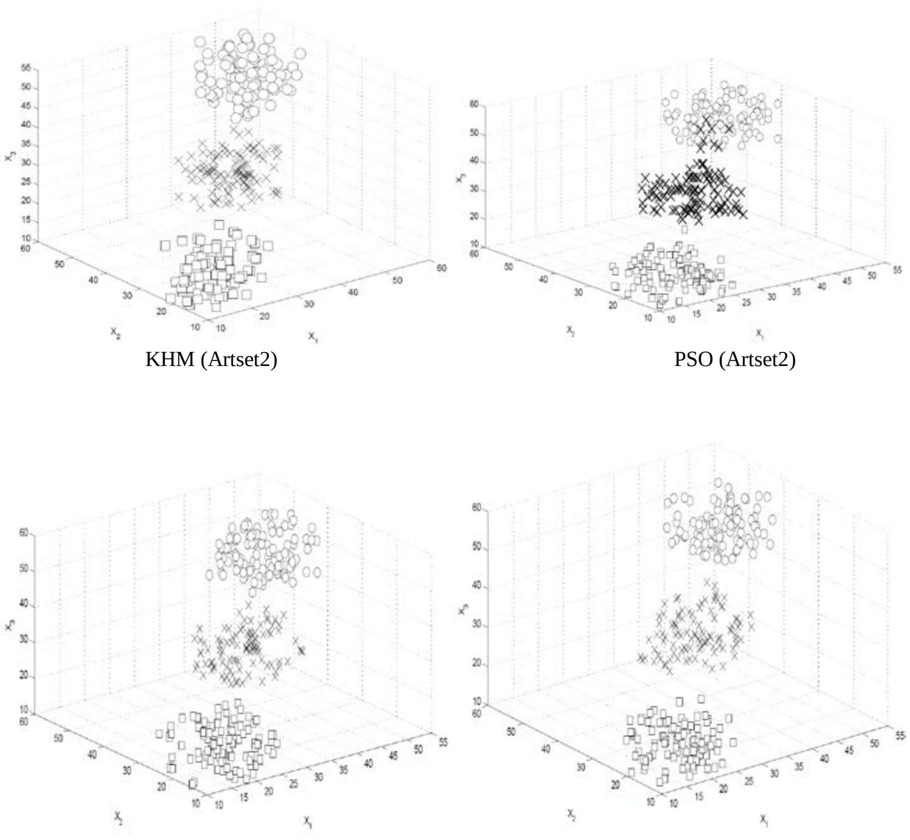
Cancer
KHM(X,C)
F-measure
Runtime
	57167.366038(0.625)
	0.961290(0.000)
	2.088459(0.013)
		113716.138(4.578)
		0.964719(0.000)
		2.0761(0.012)
			232137.293652(24.844)
			0.965901(0.000)
			2.283777(0.035)

Glass **KHM(X,C)** **F-measure** **Runtime**	**1241.6486 (9.659926)** **0.6471302 (0.019853)** **2.03842 (0.017369)**	**1740.79643 (18.84145)** **0.672816 (0.016377)** **1.904374 (0.01027)**	**2251.524907 (90.139592)** 0.67172101 (0.017489) **2.00720816 (0.010112)**

Due to the reduction of the value of KHM(X,C) function and the increase of the F-measure, KHM-IPSO algorithm generates better clustering quality than PSOKHM algorithm. In other words, KHM-IPSO improves the F-measure, runtime, and KHM(X,C) measures in most of the running with different P values. The proposed KHM-IPSO has best runtimes in most of evaluations with different *p* values. The evaluations shown in Tables 4 and Table 5 clear that KHM-IPSO algorithm has a small runtime in comparison with PSOKHM algorithm in all datasets, except cancer data set. Consequently, the accuracy, correctness, and convergence of our proposed algorithm are satisfactory and robust than the PSOKHM and other compared algorithms.

Finally, an execution of KHM-IPSO and other mentioned algorithms on Artset2 data set is shown at Figure 6. In Fig. 6, the experiment results on two artificial datasets demonstrate the improved quality of clustering by proposed KHM-IPSO algorithm. As you see, KHM-IPSO can cluster objects more clearly with best F-measure and KHM(X, C) value. The main drawback for KHM-IPSO, ICAKHM, and PSOKHM is their running time in comparison to

KHM algorithm. KHM algorithm has best running time. However, KHM-IPSO has better running time than ICAKHM and PSOKHM.



KHM (Artset2) PSO (Artset2)

PSOKHM (Artset2) **KHM-IPSO (Artset2)**

Figure 6. An execution of KHM, PSO, PSOKHM and KHM-IPSO clustering algorithms on Artset2 data set.

6. Conclusion

There are many developed algorithms to solve the drawbacks of KHM clustering. Some focus on generating well initialization methods, while others consider finding globally optimal solution. This paper proposed KHM-IPSO algorithm based on KHM, IPSO, and CS algorithms. The IPSO used Cuckoo search optimization and two new concepts *Pworts* and *Gworts* in regular PSO algorithm to improve the stability of discovering the best solutions and convergence to global optimum. Therefore, the combination of IPSO algorithm with KHM utilized the advantages of the KHM besides solved the KHM shortcomings. It overcame initialization sensitivity of KHM, and achieved the good convergence to global optimum, effectively. The new proposed algorithm was tested on several real and artificial data sets. The experiments confirmed that the proposed algorithm was accurate and robust compared to PSO, KHM, PSOKHM and ICAKHM algorithms. The proposed KHM-IPSO algorithm not only improved the F-measure and standard deviation parameters, it also helped KHM escape from local optima. Because of obtaining high-quality initializations from the IPSO, the KHM-IPSO algorithm provided a better output and performance. Our proposed algorithm clustered large datasets faster and more accurate than other algorithms. Yet, it should be mentioned that one drawback of KHM-IPSO is its runtime compared to KHM due to using PSO and Cuckoo search optimization. However, KHM-IPSO has better runtime than other combined methods because of fast convergence to global optimum, except KHM algorithm. As a future work, we investigate on combining PSO and Artificial Bee Colony into KHM to get a faster convergence, accuracy and runtime.

References

[1] Abdel-Kader, R.F. Genetically Improved PSO Algorithm for Efficient Data Clustering. *in Machine Learning and Computing (ICMLC), 2010 Second International Conference on*.2010, 71-75.

[2] Abdeyazdan, M., Data clustering based on hybrid K-harmonic means and modifier imperialist competitive algorithm. *Journal of Supercomputing*, **68**, 2, 2014, 574-598.

[3] Anaya, A.R. and J.G. Boticario, Application of machine learning techniques to analyse student interactions and improve the collaboration process. *Expert Syst. Appl.*, **38**, 2, 2011, 1171-1181.

[4] Benameur, L., J. Alami, and A. El Imrani. A New Hybrid Particle Swarm Optimization Algorithm for Handling Multiobjective Problem Using Fuzzy Clustering Technique. *in Computational Intelligence, Modelling and Simulation, 2009. CSSim '09. International Conference on*.2009, 48-53.

[5] Blake, C.L., D.J. Newman, and C.J. Merz, *UCI Repository of machine learning databases*, University of California, Irvine, Dept. of Information and Computer Sciences 1998.

[6] Bouyer, A., H. Ghafarzadeh, and O. Tarkhaneh, An Efficient Hybrid Algorithm using Cuckoo Search and Differential Evolution for Data Clustering. *Indian Journal of Science and Technology*, **8**, 24, 2015.

[7] Changhe, L. and Y. Shengxiang. A clustering particle swarm optimizer for dynamic optimization. *in Evolutionary Computation, 2009. CEC '09. IEEE Congress on*.2009, 439-446.

[8] Chen, C.-Y. and Y. Fun. Particle swarm optimization algorithm and its application to clustering analysis. *in Networking, Sensing and Control, 2004 IEEE International Conference on*.2004, 789-794 Vol.2.

[9] Chuang, L.-Y., C.-J. Hsiao, and C.-H. Yang, Chaotic particle swarm optimization for data clustering. *Expert Systems with Applications*, **38**, 12, 2011, 14555-14563.

[10] Dalli, A., *Adaptation of the F-measure to cluster based lexicon quality evaluation*, in *Proceedings of the EACL 2003 Workshop on Evaluation Initiatives in Natural*

Language Processing: are evaluation methods, metrics and resources reusable? Association for Computational Linguistics: Budapest, Hungary. 2003, 51-56.

[11] Danesh, M., M. Naghibzadeh, M. Totonchi, M. Danesh, B. Minaei, and H. Shirgahi, *Data Clustering Based on an Efficient Hybrid of K-Harmonic Means, PSO and GA, in Transactions on Computational Collective Intelligence IV,* N. Nguyen, Editor, Springer Berlin Heidelberg. 2011, 125-140.

[12] Fathi, V. and G.A. Montazer, An improvement in RBF learning algorithm based on PSO for real time applications. *Neurocomputing,* **111**, 0, 2013, 169-176.

[13] Fodeh, S.J., C. Brandt, T.B. Luong, A. Haddad, M. Schultz, T. Murphy, and M. Krauthammer, Complementary ensemble clustering of biomedical data. *Journal of Biomedical Informatics*, **46**, 3, 2013, 436-443.

[14] Ghaemi, R., N.B. Sulaiman, H. Ibrahim, and N. Mustapha, A review: accuracy optimization in clustering ensembles using genetic algorithms. *Artif. Intell. Rev.*, **35**, 4, 2011, 287-318.

[15] Hatamlou, A., In search of optimal centroids on data clustering using a binary search algorithm. *Pattern Recognition Letters,* **33**, 13, 2012, 1756-1760.

[16] Hatamlou, A., V. Beiranvand, A. Bouyer, A. Habiboghli, and R. Mostafaeia, *Application of modified PSO on clustering,* in *5th Postgraduate Annual Research Seminar 2009 (PARS'09)*: Malaysia. 2009.

[17] He, Q., A review of clustering algorithms as applied in IR. *Univ. Illinois at Urbana-Champaign, Tech. Rep. UIUCLIS-1999/6+ IRG*, 1999.

[18] Hilas, C.S. and P.A. Mastorocostas, An application of supervised and unsupervised learning approaches to telecommunications fraud detection. *Knowledge-Based Systems*, **21**, 7, 2008, 721-726.

[19] Hu, G., S. Zhou, J. Guan, and X. Hu, Towards effective document clustering: A constrained K-means based approach. *Information Processing & Management*, **44**, 4, 2008, 1397-1409.

[20] Jain, A.K., Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, **31**, 8, 2010, 651-666.

[21] Jain, A.K., M.N. Murty, and P.J. Flynn, Data clustering: a review. *ACM Comput. Surv.*, **31**, 3, 1999, 264-323.

[22] Kao, Y.-T. and E. Zahara, A hybrid genetic algorithm and particle swarm optimization for multimodal functions. *Appl. Soft Comput.*, **8**, 2, 2008, 849-857.

[23] Kao, Y.-T., E. Zahara, and I.W. Kao, A hybridized approach to data clustering. *Expert Systems with Applications*, **34**, 3, 2008, 1754-1762.

[24] Karaboga, D. and C. Ozturk, A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Appl. Soft Comput.*, **11**, 1, 2011, 652-657.

[25] Keller, F., Clustering. *Computer University Saarlandes, Tutorial Slides*.

[26] Kennedy, J. and R. Eberhart. Particle swarm optimization. *in Neural Networks, 1995. Proceedings., IEEE International Conference on.*1995, 1942-1948 vol.4.

[27] Kiranyaz, S., J. Pulkkinen, and M. Gabbouj, Multi-dimensional particle swarm optimization in dynamic environments. *Expert Systems with Applications*, **38**, 3, 2011, 2212-2223.

[28] Kumar, M. and N.R. Patel, Clustering data with measurement errors. *Computational Statistics & Data Analysis*, **51**, 12, 2007, 6084-6101.

[29] Kumar, S. and C.S.P. Rao, Application of ant colony, genetic algorithm and data mining-based techniques for scheduling. *Robotics and Computer-Integrated Manufacturing*, **25**, 6, 2009, 901-908.

[30] Lévy, P. *The Lévy Distribution*. Available from: http://www.math.uah.edu/stat/special/Levy.html.

[31] Liu, C.-M., C.-H. Lee, and L.-C. Wang, Distributed clustering algorithms for data-gathering in wireless mobile sensor networks. *Journal of Parallel and Distributed Computing*, **67**, 11, 2007, 1187-1200.

[32] Maimon, O.Z. and L. Rokach, Data mining and knowledge discovery handbook. Vol. 1. 2005: Springer.

[33] Mangat, V., *Survey on Particle Swarm Optimization Based Clustering Analysis*, *in Swarm and Evolutionary Computation*, L. Rutkowski, et al., Editor, Springer Berlin Heidelberg. 2012, 301-309.

[34] Marinakis, Y., M. Marinaki, M. Doumpos, and C. Zopounidis, Ant colony and particle swarm optimization for financial classification problems. *Expert Syst. Appl.*, **36**, 7, 2009, 10604-10611.

[35] Nanda, S.J. and G. Panda, Automatic clustering algorithm based on multi-objective Immunized PSO to classify actions of 3D human models. *Engineering Applications of Artificial Intelligence*, **26**, 5–6, 2013, 1429-1441.

[36] Niknam, T. and B. Amiri, An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. *Applied Soft Computing*, **10**, 1, 2010, 183-197.

[37] Passino, K.M., Biomimicry of bacterial foraging for distributed optimization and control. *Control Systems, IEEE*, **22**, 3, 2002, 52-67.

[38] Rana, S., S. Jasola, and R. Kumar, A review on particle swarm optimization algorithms and their applications to data clustering. *Artificial Intelligence Review*, **35**, 3, 2011, 211-222.

[39] Runkler, T.A., Ant colony optimization of clustering models: Research Articles. *Int. J. Intell. Syst.*, **20**, 12, 2005, 1233-1251.

[40] Saatchi, S. and C.C. Hung, *Hybridization of the ant colony optimization with the k-means algorithm for clustering*, in *Proceedings of the 14th Scandinavian conference on Image Analysis* Springer-Verlag: Joensuu, Finland. 2005, 511-520.

[41] Senthilnath, J., V. Das, S.N. Omkar, and V. Mani, *Clustering Using Levy Flight Cuckoo Search*, *in Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012)*, J.C. Bansal, et al., Editor, Springer India. 2013, 65-75.

[42] Servet, M.K., M.S. Kiran, E. ÖZceylan, M. Gündüz, and T. Paksoy, Swarm intelligence approaches to estimate electricity energy demand in Turkey. *Know.-Based Syst.*, **36**, 2012, 93-103.

[43] Shengxiang, Y. and L. Changhe, A Clustering Particle Swarm Optimizer for Locating and Tracking Multiple Optima in Dynamic Environments. *Evolutionary Computation, IEEE Transactions on*, **14**, 6, 2010, 959-974.

[44] Sung, C. and H. Jin, A tabu-search-based heuristic for clustering. *Pattern Recognition*, **33**, 5, 2000, 849-858.

[45] Van Der Merwe, D.W. and A.P. Engelbrecht. Data clustering using particle swarm optimization. *in Evolutionary Computation, 2003. CEC '03. The 2003 Congress on*.2003, 215-220 Vol.1.

[46] Xin-She, Y. and S. Deb. Cuckoo Search via Levy flights. *in Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*.2009, 210-214.

[47] Yan, X., Y. Zhu, W. Zou, and L. Wang, A new approach for data clustering using hybrid artificial bee colony algorithm. *Neurocomput.*, **97**, 2012, 241-250.

[48] Yang, F., T. Sun, and C. Zhang, An efficient hybrid data clustering method based on K-harmonic means and Particle Swarm Optimization. *Expert Syst. Appl.*, **36**, 6, 2009, 9847-9852.

[49] Yang, S., R. Wu, M. Wang, and L. Jiao, Evolutionary clustering based vector quantization and SPIHT coding for image compression. *Pattern Recogn. Lett.*, **31**, 13, 2010, 1773-1780.

[50] Yang, X.-S. and S. Deb. Cuckoo search via Lévy flights. *in Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*. IEEE,2009, 210-214.

[51] Žalik, K.R., An efficient k′-means clustering algorithm. *Pattern Recognition Letters*, **29**, 9, 2008, 1385-1391.

[52] Zhang, B., M. Hsu, and U. Dayal, *K-Harmonic Means -A Spatial Clustering Algorithm with Boosting, in Temporal, Spatial, and Spatio-Temporal Data Mining*, J. Roddick and K. Hornsby, Editor, Springer Berlin Heidelberg. 2001, 31-45.

Received 19.11.2015, Accepted 30.05.2016