Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2017) 25: 1251 – 1262
© TÜBİTAK
doi:10.3906/elk-1508-31

A clustering approach using a combination of gravitational search algorithm and k-harmonic means and its application in text document clustering

Mina MIRHOSSEINI[*]
Department of Computer Science, Faculty of Mathematics and Computing, Higher Education Complex of Bam, Bam, Iran

Received: 04.08.2015 • **Accepted/Published Online:** 17.04.2016 • **Final Version:** 10.04.2017

Abstract: Data clustering is one of the most popular techniques of information management, which is used in many applications of science and engineering such as machine learning, pattern reorganization, image processing, data mining, and web mining. Different algorithms have been suggested by researchers, where the evolutionary algorithms are the best in data clustering and especially in big datasets. It is illustrated that GSA-KM, which is a combination of the gravitational search algorithm (GSA) and K-means (KM), is superior over some other comparative evolutionary methods. One of the drawbacks of this approach is dependency on the initial seeds. In this paper, a combination method of GSA and K-harmonic means, called GSA-KHM, has been proposed, in which the dependency on the initialization has been improved. The proposed GSA-KHM method has been applied to data clustering. As a special application, it has also been used on the text document clustering application. The simulation results show that the proposed method works better than the GSA-KM and other comparative methods in both data clustering and text document clustering applications.

Key words: Clustering, gravitational search algorithm, K-means, K-harmonic means, text clustering

1. Introduction

Clustering aims to partition data into clusters whereby data inside a cluster have the most similarity to each other and the most differences from the objects of the other clusters. Many methods have been presented and applied in various applications. One of the most applicable and popular methods is K-means [1], which has the drawback of falling into local optima and dependency on the initial seed setting. The other method is K-harmonic means (KHM) [2], which aims to minimize the harmonic means of distances between data and their cluster centers. This method solves the problem of initialization in K-means; however, there is a risk of falling into the local optimum. Many heuristic methods have been proposed for solving this problem. In [3] a genetic algorithm was used for clustering of several datasets. A combinational method of K-means and the genetic algorithm was proposed in [4]. Two clustering methods based on ant colony and honeybee mating were respectively proposed in [5] and [6]. Particle swarm optimization was used in clustering in [7] and [8]. In [9], a simulated annealing algorithm for the clustering problem was developed. In [10] and [11], the gravitational search algorithm (GSA) was described and used in data clustering application. The authors of [12] suggested a new clustering method called sparse subspace clustering, which was applied to synthetic data as well as motion segmentation and face clustering applications. In [13] a method based on fuzzy relevance clustering was proposed for multilabel text classification in which a document can belong to one or more than one category.

[*] Correspondence: mirhosseini@bam.ac.ir

A new metaheuristic optimization algorithm named symbiotic organisms search was introduced in [14], which can be used as an optimization method for clustering applications.

KHM data clustering with the tabu search method was proposed in [15]. In 2011, a hybrid of KHM and the GSA was proposed in [16]. Using a combination of K-means and GSA, a method called GSA-KM was suggested in [17], where a comparison with all the mentioned methods illustrated the performance of this algorithm over the previous ones. In this method, a K-means algorithm is executed first, and its solutions enter as a part of the initial population of the GSA. This approach explores a more appropriate search space and also helps to prevent falling into the local optimum. The algorithm could converge to the optimal solution faster than before. The drawback of this algorithm is its sensitivity to the initial seeds.

In line with the theorem of “no free lunch”, there is no metaheuristic algorithm to optimally solve all optimizing problems [14]. Therefore, we aim to investigate a high-performance method to handle the clustering issue as an optimization problem. With this goal, we used KHM instead of K-means as in the proposed method in [17]. The novelty of this work is proposing a new clustering method by combining the GSA and KHM. The proposed method exploits the advantages of both KHM and the GSA. In this method, the KHM is used to help generate the initial population of the GSA to speed up the convergence and subsequently achieve more optimal results. Experiments have been conducted on five well-known UCI [18] datasets and, as a case study, it is applied on four text document datasets. The simulation results for both applications show the superiority of GSA-KHM over GSA-KM [17] and other previous methods including K-means [1], GA [3], ACO [5], HBMO [6], PSO [7,11], SA [9], and the original GSA [10,11].

The remainder of this study has been organized as follows. The clustering methods of KM and KHM are described and compared in Section 2. In Section 3, the GSA is discussed. The proposed GSA-KHM clustering method is introduced in Section 4. The experiments and simulation results of GSA-KHM in comparison with eight other methods on five UCI datasets are presented in Section 5. In Section 6 a case study on the text document clustering by the proposed GSA-KHM algorithm is explained, and finally a summary of the study and the future works are discussed in Section 7.

2. The KM and KHM clustering methods

KM, which was first proposed by MacQueen in 1967 [19], is a simple unsupervised method for solving clustering problems. This procedure classifies the set through a predefined number of clusters (k). This algorithm aims to minimize the objective function shown as Eq. (1).

$$d = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 \tag{1}$$

Here, $\left\| x_i^{(j)} - c_j \right\|$ is the distance between data $x_i^{(j)}$ and the cluster center c_j for which $x_i^{(j)}$ is associated to the cluster represented by c_j and n is the number of data inside cluster j. The procedure of the algorithm is as follows:

1. Place random k points as the initial group centers.
2. Assign each datum to a group that has the nearest center.
3. Update the centers by taking the average of the data within each cluster.

4. Repeat steps 2 and 3 until there is no longer change in the center.

The result of the KM algorithm strictly depends on the initial chosen centers and the data have a close relation to their centers. In KM, each datum is assigned to just one center, so in the high density regions, the centers may not be able to move away from their data, although another center may exist there. Moving a center may lead to a small negative change on the local optimum, but its total effect on improving clustering accuracy would be useful. This movement could not be down by KM; however, KHM solved this problem using membership degrees.

KHM, like KM, is a center-based clustering algorithm proposed in 1999 [2]. This algorithm uses the harmonic averages of the distances from each data point to all centers. If there is no center near some data, this algorithm is able to move some centers near them. Considering $X = \{x_1 x_2, \ldots, x_n$ as the data points and $C = \{c_1 c_2, \ldots, c_k$ as the set of centers of the k clusters, the KHM algorithm does the following steps:

1. Generate the cluster centers randomly.
2. Calculate the objective function using Eq. (2).

$$KHM\left(X,C\right) = \sum_{i=1}^{n} \frac{k}{\sum\limits_{j=1}^{K} \frac{1}{\|x_i - c_j\|^p}}, \quad p \geq 2 \tag{2}$$

3. Recompute the center c_j as:

$$c_j = \frac{\sum\limits_{i=1}^{n} m\left(c_j \,|\, x_i\right) w(x_i) x_i}{\sum\limits_{i=1}^{n} m\left(c_j \,|\, x_i\right) w(x_i)} \quad , \tag{3}$$

4. where $w(x_i)$ is a weighting parameter that is computed by Eq. (4) and is representative of the effect of data x_i in recalculating the center parameters. $m(c_j|x_i)$ is a function that determines the membership degree of x_i to the center c_j and is obtained using Eq. (5) [2].

$$w\left(x_i\right) = \frac{\sum\limits_{j=1}^{k} \|x_i - c_j\|^{-p-2}}{(\sum\limits_{j=1}^{k} \|x_i - c_j\|^{-p})^2} \tag{4}$$

$$m\left(c_j \,|\, x_i\right) = \frac{\|x_i - c_j\|^{-p-2}}{\sum\limits_{j=1}^{k} \|x_i - c_j\|^{-p-2}} \tag{5}$$

Repeat step 3 until there is no significant change in $KHM(XC)$ value.

The data x_i will be assigned to cluster j, which has the maximum membership value of $m(c_j|x_i)$.

In this paper, KHM is used in generating the initial population in the GSA in order to produce faster convergence to the optimal solutions and obtain more optimal results [2,15].

3. The gravitational search algorithm

The GSA is a novel developed search algorithm based on gravity and mass interactions. According to Eq. (6), the gravitational force (F) between two objects is directly proportional to the product of their masses and inversely proportional to the square of the distance between them.

$$F = G\frac{M_1.M_2}{R^2} \tag{6}$$

Here, M_1 and M_2 are respectively the mass of the first and the second particles, which are placed at a distance of R from each other. G is a constant equal to 6.67259×10^{-11}. According to Newton's second law (Eq. (7)), by applying the force F to a particle, its acceleration (a) depends only on its mass (M).

$$a = \frac{F}{M} \tag{7}$$

In the GSA, N objects are considered as N masses, which are in an n-dimensional search space. The position of mass i is defined as in Eq. (8), where x_i^d is the position of mass i in the dimension d in n-dimensional search space.

$$X_i = \left(x_i^1, \ldots, x_i^d, \ldots, x_i^n\right),\ i = 1,\, 2,\, \ldots, N \tag{8}$$

The optimal position of the particle corresponds to the optimal solution. The mass of each particle is computed using Eq. (9).

$$M_i(t) = \frac{fit_i\,(t) - worst(t)}{\sum\limits_{j=1}^{N}\left(fit_j(t) - worst\,(t)\right)} \tag{9}$$

In this equation, $M_i(t)$ is the mass of particle i, $fit_i\,(t)$ is its fitness value at iteration t, and $worst\,(t)$ is computed as in Eq. (10). The bigger mass is representative of the better solution. It means a bigger mass moves slower in the search space.

$$worst\,(t) = {}_{j\in\{1,\ldots,N\}}^{max} fit_j\,(t) \tag{10}$$

Regarding Eqs. (6) and (7), the summation of all entered forces to a mass is required to compute the acceleration. Eq. (11) is the total of all entered forces to particle i. By replacing Eq. (11) in Eq. (7), the acceleration is obtained as in Eq. (12).

$$F_i^d\,(t) = \sum_{j\in kbest,\, j\neq i} rand_j G\,(t)\,\frac{M_i\,(t)\,.M_j\,(t)}{R_{ij}\,(t) + \varepsilon}\left(x_j^d\,(t) - x_i^d\,(t)\right) \tag{11}$$

$$a_i^d\,(t) = \frac{F_i^d(t)}{M_i(t)} = \sum\nolimits_{j\in kbest,\, j\neq i} rand_j G\,(t)\,\frac{M_j\,(t)}{R_{ij}\,(t) + \varepsilon}(x_j^d\,(t) - x_i^d\,(t)\,) \tag{12}$$

Here, ε is a small constant to avoid dividing by zero. $rand_j$ is a random value in the interval [0,1]. The $kbest$ is a set of k best solutions whose size is N at first and reaches 1 finally. $G\,(t)$ is a decreasing function with respect to time, which starts from 1 and ends at 0. R_{ij} is the Euclidian distance between particles i and j, computed as in Eq. (13).

$$R_{ij} = \sqrt{\sum_{p=1}^{d}(x_i^p - x_j^p)^2} \tag{13}$$

After computing acceleration, the new velocity and position are obtained respectively using Eqs. (14) and (15).

$$v_i^d\,(t+1) = rand_i \times v_i^d\,(t) + a_i^d(t) \tag{14}$$

$$x_i^d\,(t+1) = x_i^d\,(t) + v_i^d(t+1) \tag{15}$$

The steps of the GSA are summarized in Algorithm 1 [10,11,16,20,21].

Algorithm 1. The steps of the gravitational search algorithm.

```
1. Identify the search space.
2. Generate random initial population.
3. Evaluate the agents' fitness.
4. Update $G(t)$, $worst(t)$, and $M_i\,(t)$ for $i = 1, 2, ..., N$ using Eq. (9) and Eq. (10).
5. Calculate the total force in different directions by Eq. (11).
6. Calculate the acceleration and velocity by Eq. (12) and Eq. (14).
7. Update the position of agents using Eq. (15).
8. Repeat steps 3 to 7 until the stopping criterion (specified number of iterations) is satisfied.
```

4. The proposed clustering approach using GSA-KHM

The main idea of this method is the GSA, in which generating the initial population is improved. This leads to a reduction in iteration number of the GSA; therefore, this method can enhance the performance of the GSA in data clustering. In this algorithm, KHM is executed on the data first and the result of its clustering enters as an individual into the initial population of the GSA. The minimum, the maximum, and the average of the data enter as the other part of the initial population. The remaining part of the initial population is generated randomly. This strategy of generating the initial population aims to accelerate obtaining the optimal clustering. In this method, each particle is represented as an array in the size of $(d \times k)$, where k and d are respectively the number of clusters and the number of features in each dataset. $P_i = \{Z_1, Z_2, \ldots, Z_k\}$ as the ith particle or a possible solution consists of the cluster centers, where $Z_j = (z_j^1 z_j^2, \ldots, z_j^d)$ is the center of the jth cluster $j = 1, 2, \ldots, k$ and $i = 1, 2, ..m$. In the latter notation, m is the size of the population. The steps of the algorithm are summarized in Algorithm 2.

5. Experimental results

The proposed method described in the previous section has been applied to five well-known UCI datasets of [18] including Iris, Wine, Glass, CMC, and Cancer, whose properties are summarized in Table 1. In this table, n is the size of the dataset, d is the number of features, and k is the number of clusters. The proposed GSA-KHM clustering approach is compared with the K-means [1], GA [3], ACO [5], HBMO [6], PSO [7,11], SA [9], original GSA [10,11], and GSA-KM [17] algorithms.

The best, the average, the worst, and the standard deviation of the distance of the data within each cluster in all methods are reported in the Table 2 for the five datasets. The results obviously show the superiority

Algorithm 2. The steps of the GSA-KHM.

```
• Phase 1: Executing KHM on the dataset.
• Phase 2: Generating the initial population $P_1 P_2, \ldots, P_m$ of the GSA such that:
    ○ $P_1$ is the output of KHM.
    ○ $P_2$ is the minimum of the dataset.
    ○ $P_3$ is the average of the dataset.
    ○ $P_4$ is the maximum of the dataset.
    ○ $P_5, \ldots, P_m$ are the randomly generated solutions.
• Phase 3: Running the GSA:
    ○ Evaluating all particles by the objective function.
    ○ Calculating $M$, $F$, and $a$ for each particle respectively according Eqs. (9), (11), and (12).
    ○ Updating the velocity and the position of particles based on Eqs. (14) and (15) respectively.
    ○ Phase 3 will be iterated until the stopping criterion (the predefined number of iterations)
      is reached.
```

Table 1. The properties of the used datasets.

Dataset	n	d	k
Iris	150	4	3
Wine	178	13	3
Glass	214	9	6
CMC	1473	10	3
Cancer	683	9	2

of GSA-KHM over the other comparative methods. The results are averaged over 20 independent runs of the algorithms on a population size of 50.

The results of [17] show the performance of GSA-KM in comparison with K-means [1], GA [3], ACO [5], HBMO [6], PSO [7,11], SA [9], and the original GSA [10,11]. In this study, our experimental results indicate the superiority of the proposed GSA-KHM in comparison with GSA-KM and the other comparative methods. Generally, both GSA-KM and GSA-KHM prevent falling into the local optimum and converge to the optimal solution faster than GSA because of providing a part of the initial population by KM or KHM. However, since KHM is not sensitive to the placing of the initial population, GSA-KHM improves the drawback of GSA-KM. In addition, the GSA-KHM clustering results are more desirable than those of GSA-KM.

6. Case study: GSA-KHM in text document clustering

6.1. Methodology

Here we use the proposed GSA-KHM to categorize text documents. In the text document application, some preprocessing is needed before clustering. For instance, there are many words in the text documents that do not describe the topic of the text, such as a, the, is, and are. These words are called stop-words and should be removed from the text. For this purpose, we have used a presupplied list used in the Weka machine learning workbench (http://jmlr.csail.mit.edu/papers/volume5/lewis04a/a11-smart-stoplist/english.stop) that contains

Table 2. The clustering results of GSA-KM and GSA-KHM on five UCI datasets.

Dataset	Criteria	K-means	GA	SA	ACO	HBMO	PSO	GSA	GSA-KM	GSA-KHM
Iris	Best	97.333	113.98	97.45	97.10	96.75	96.894	96.698	96.679	95.631
	Average	106.050	125.19	99.95	97.17	96.95	97.232	96.723	96.689	96.653
	Worst	120.450	139.77	102.01	97.80	97.75	97.897	96.764	96.705	96.698
	Std	14.631	14.563	2.018	0.367	0.531	0.347	0.0123	0.0076	0.0074
Wine	Best	16,555.68	16,530.53	16,473.48	16,530.53	16,357.28	16,345.96	16,315.35	16,294.25	16,289.4
	Average	18,061.00	16,530.53	17,521.09	16,530.53	16,357.28	16,417.47	16,376.61	16,294.31	16,290.30
	Worst	18,563.12	16,530.53	18,083.25	16,530.53	16,357.28	16,562.31	16,425.58	16,294.64	16,293.50
	Std	793.21	0	753.084	0	0	85.49	31.34	0.0406	0.0498
Glass	Best	215.74	278.37	275.16	269.72	245.73	270.57	220.78	211.47	201.44
	Average	235.50	282.32	282.19	273.46	247.71	275.71	225.70	214.22	200.35
	Worst	255.38	286.77	287.18	280.08	249.54	283.52	229.45	216.08	211.76
	Std	12.47	4.138	4.238	3.584	2.438	4.55	3.4008	1.1371	0.0498
CMC	Best	5842.20	5705.63	5849.03	5701.92	5699.26	5700.98	5698.15	5697.03	5691.68
	Average	5893.60	5756.59	5893.48	5819.13	5713.98	5820.96	5699.84	5697.36	200.35
	Worst	5934.43	5812.64	5966.94	5912.43	5725.35	5923.24	5702.09	5697.87	5695.20
	Std	47.16	50.369	50.867	45.634	12.690	46.95	1.724	0.2717	0.2878
Cancer	Best	2999.19	2999.32	2993.45	2970.49	2989.94	2973.50	2967.96	2965.14	2951.08
	Average	3251.21	3249.46	3239.17	3046.06	3112.42	3050.04	2973.58	2965.21	2948.67
	Worst	3521.59	3427.43	3421.95	3242.01	3210.78	3318.88	2990.83	2965.30	2949.07
	Std	251.14	229.734	230.192	90.500	103.471	110.80	8.1731	0.0670	0.0677

527 stop-words. In addition, in the text documents, there are words that are thematically similar but have different morphological concepts. These words should be mapped into their stem to be treated as a single word. For example, the words "computing" and "computation" have similar concepts. Although their morphology is different, these words are stemmed to "compute" [22]. We have used Porter's suffix-stripping algorithm for the stemming process [23].

We modeled the documents as a bag of words and used word-by-word comparison. If $T = \{t_1 t_2, \ldots, t_l$ is the set of all occurred terms in the corpus, each document can be represented as a feature vector $d_i = \{w_{1i} w_{2i}, \ldots, w_{li}$, where $w_{ki}$is the weight of term t_k in document d_i. We have used a well-known weighting schema named TF-IDF, which is defined in Eq. (16) [24,25].

$$w_{ki} = TFIDF\left(d_i, t_k\right) = TF\left(d_i, t_k\right) \times \log\left(\frac{|D|}{DF\left(t_k\right)}\right) \tag{16}$$

Here, $TF(d_i t_k)$ is the frequency of term t_k in document d_i, D is the number of documents in the corpus, and $DF(t_k)$ is the number of documents in which term t_k occurs. The words whose weights are less than a predefined value are then removed.

The remaining words are assumed as the key words and constitute the property vector. Afterwards, the similarity between all pairs of documents should be computed by an appropriate similarity measure. Then the GSA-KHM clustering approach of Algorithm 1 is applied to the extracted property vectors. Measuring the similarity between two text documents is one of the important issues in the text clustering field [26]. In this study we have used the cosine similarity measure, which is seen in Eq. (17). It computes the similarity between document $d_i = \{w_{1i}w_{2i}, \ldots, w_{ni}$ and document $d_i = \{w_{1j}w_{2j}, \ldots, w_{nj}$ as follows [24,25,27]:

$$Co\sin e - Sim(d_i, d_j) = \frac{\sum_{k=1}^{n} w_{ki}w_{kj}}{\sqrt{\sum_{k=1}^{n} w_{ki}^2 \sum_{k=1}^{n} w_{kj}^2}}. \tag{17}$$

The clustering results are evaluated by two known measures named entropy and F-measure, which are represented in Eqs. (18) and (20), respectively [23].

$$ECS = \sum_{j=1}^{m} \frac{n_j \times E_j}{n} \tag{18}$$

Here, n is the size of the dataset, m is the number of clusters, n_j is the size of cluster j, and E_j is the entropy of cluster j, which is computed by Eq. (19).

$$E_j = -\sum_{i} p_{ij} log(p_{ij}) \tag{19}$$

In this equation, p_{ij} is the probability of a member belonging to cluster j to class i. The F-measure is computed as follows:

$$F\,(i,\, j) = \frac{2 \times Recall(i,\, j) \times Precision(i, j)}{Recall\,(i,\, j) + Precision(i, j)} \tag{20}$$

where $Recall(ij)$ and $Precision(ij)$ are calculated by Eqs. (21) and (22), respectively.

$$Recall\,(i, j) = \frac{n_{ij}}{n_i} \tag{21}$$

$$Precision\,(i, j) = \frac{n_{ij}}{n_j} \tag{22}$$

In these equations, n_{ij} is the number of class i members in cluster j, and n_i is the size of class i and n_j is the size of cluster j. Generally, higher values of F-measure and lower values of entropy are representative of better results in clustering. The Figure illustrates the procedure of text document clustering by GSA-KHM.

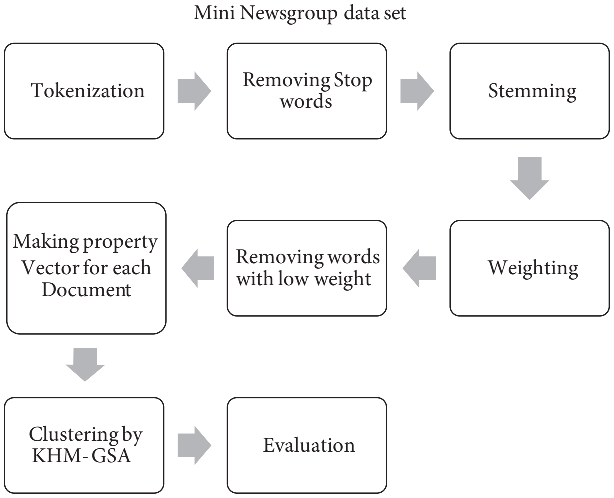

Figure. Steps of text document clustering using GSA-KHM.

6.2. Experimental results

We have used the Reu_01, Re0, and Re1 datasets from Reuters-21587 (http://archive.ics.uci.edu/ml/datasets/Reuters21578+Text+Categorization+Collection) and Mini Newsgroup from 20Newsgroup (http://people.csail.mit.edu/jrennie/20Newsgroups) sources. The properties of these datasets are given in Table 3. After preprocessing of the text documents, the proposed GSA-KHM has been applied to the mentioned datasets. The results of the proposed GSA-KHM in comparison with GSA-KM have been reported for datasets Re01, Re0, Re1, and Mini Newsgroup respectively in Tables 4–7 by varying the number of

Table 3. The properties of text document datasets.

Dataset	Source	The number of selected documents	The number of classes
Reu_01	Reuters-21587	1000	5
Re0	Reuters-21587	1504	13
Re1	Reuters-21587	1657	25
Mini_Newsgroup	20Newsgroup	2000	20

Table 4. The comparison between GSA-KHM and GSA-KM clustering methods on Reu_01 dataset.

The number of clusters	GSA-KHM		GSA-KM	
	Entropy	F-measure	Entropy	F-measure
$K = 2$	0.42	0.52	0.49	0.45
$K = 3$	0.40	0.44	0.47	0.40
$K = 4$	0.39	0.45	0.45	0.36
$K = 5$	0.38	0.40	0.46	0.34

Table 5. The comparison between GSA-KHM and GSA-KM clustering methods on the Re0 dataset.

The number of clusters	GSA-KHM		GSA-KM	
	Entropy	F-measure	Entropy	F-measure
$K = 2$	0.34	0.48	0.38	0.44
$K = 3$	0.34	0.46	0.37	0.45
$K = 4$	0.33	0.45	0.36	0.43
$K = 5$	0.31	0.42	0.35	0.40
$K = 6$	0.30	0.39	0.33	0.37
$K = 7$	0.28	0.38	0.33	0.35
$K = 8$	0.28	0.35	0.32	0.32
$K = 9$	0.27	0.32	0.30	0.30
$K = 10$	0.25	0.31	0.29	0.29
$K = 11$	0.23	0.30	0.29	0.27
$K = 12$	0.23	0.27	0.26	0.26
$K = 13$	0.21	0.26	0.24	0.24

clusters (K). It is clearly observable that the results of GSA-KHM are better that those of GSA-KM in terms of entropy and F-measure for all used datasets.

Table 6. The comparison between GSA-KHM and GSA-KM clustering methods on the Re1 dataset.

The number of clusters	GSA-KHM		GSA-KM	
	Entropy	F-measure	Entropy	F-measure
K = 2	0.52	0.62	0.55	0.59
K = 4	0.48	0.59	051	0.57
K = 6	0.46	0.56	0.50	0.55
K = 8	0.41	0.52	0.44	0.49
K = 10	0.39	0.48	0.41	0.47
K = 12	0.36	0.46	0.39	0.44
K = *14*	0.34	0.42	0.36	0.39
K = 16	0.31	0.41	0.35	0.37
K = 18	0.29	0.39	0.32	0.36
K = 20	0.27	0.36	0.29	0.33
K = 22	0.24	0.32	0.27	0.30
K = 24	0.23	0.30	0.24	0.27

Table 7. The comparison between GSA-KHM and GSA-KM clustering methods on Mini Newsgroup dataset.

The number of clusters	GSA-KHM		GSA-KM	
	Entropy	F-measure	Entropy	F-measure
K = 2	0.22	0.81	0.28	0.75
K = 3	0.26	0.79	0.29	0.73
K = 4	0.27	0.77	0.31	0.71
K = 5	0.28	0.76	0.31	0.69
K = 6	0.28	0.69	0.32	0.64
K = 7	0.29	0.64	0.32	0.63
K = 8	0.29	0.62	0.33	0.61
K = 9	0.30	0.61	0.33	0.58
K = 10	0.30	0.57	0.33	0.54
K = 11	0.31	0.56	0.34	0.51

7. Conclusions

In this study, a clustering method has been presented by combination of the GSA and KHM. The proposed method has better clustering results than GSA-KM. In addition, unlike GSA-KM, this method is not dependent on the initial centers. This method was applied to five well-known UCI datasets and four textual datasets. The simulation results of both cases show better performance of the proposed GSA-KHM in comparison with GSA-KM. Therefore, this method can work as an appropriate tool in data clustering in some applications.

References

[1] Jain AK. Data clustering: 50 years beyond K-means. J Pattern Recogn 2010; 31: 651-666.

[2] Zhi XB, Fan JI. Some notes on k-harmonic means clustering algorithm. Advances in Intelligent and Soft Computing 2010; 82: 375-384.

[3] Maulik U, Bandyopadhyay S. Genetic algorithm-based clustering technique. J Pattern Recogn 2000; 33: 1455-1465.

[4] Krishna K, Narasimha M. Genetic k-means algorithm. IEEE T Syst Man Cy B 1999; 29: 433-439.

[5] Shelokar PS, Jayaraman VK, Kulkarni BD. An ant colony approach for clustering. Anal Chim Acta 2004; 509: 187-195.

[6] Fathian M, Amiri B, Maroosi A. Application of honey-bee mating optimization algorithm on clustering. J Appl Math Comput 2007; 190: 1502-1513.

[7] Chen CY, Fun Y. Particle swarm optimization algorithm and its application to clustering analysis. In: IEEE 2004 Networking, Sensing and Control International Conference; 21–23 March 2004; Taipei, Taiwan. New York, NY, USA: IEEE. pp. 789-794.

[8] Jin P, Zhu YL, Hu KY. A clustering algorithm for data mining based on swarm intelligence. In: IEEE 2007 Machine Learning and Cybernetics International Conference; 19–22 August 2007; Hong Kong. New York, NY, USA: IEEE. pp. 803-807.

[9] Selim SZ, Alsultan K. Simulated annealing algorithm for the clustering problem. Pattern Recogn 1991; 24: 1003-1008.

[10] Hatamlou A, Abdullah S, Nezamabadi-pour H. Application of gravitational search algorithm on data clustering. Lect Notes Comp Sci 2011; 6954: 3379-3466.

[11] Hatamlou A, Abdullah S, Othman Z. Gravitational search algorithm with heuristic search for clustering problems. In: IEEE 2011 Data Mining and Optimization Conference; 28–29 June 2011; Putrajaya, Malaysia. New York, NY, USA: IEEE. pp. 190-193.

[12] Elhamifar E, Vidal R. Sparse subspace clustering: algorithm, theory, and applications. IEEE T Pattern Anal 2013; 35: 2765-2781.

[13] Lee SJ, Jiang JY. Multilabel text categorization based on fuzzy relevance clustering. IEEE T Fuzzy Syst 2014; 22: 1457-1471.

[14] Cheng MY, Prayogo D. Symbiotic organisms search: a new metaheuristic optimization algorithm. Comput Struct 2014; 139: 98-112.

[15] Güngör Z, Ünler A. K-Harmonic means data clustering with tabu-search method. Appl Math Model 2008; 32: 141-151.

[16] Yin M, Hu Y, Yang F, Li X, Gu W. A novel hybrid k-harmonic means and gravitational search algorithm approach for clustering. Expert Syst Appl 2011; 38: 9319-9324.

[17] Gungor Z, Unler A. A combined approach for clustering based on K-means and gravitational search algorithms. Swarm and Evolutionary Computation 2012; 6: 47-52.

[18] Blake CL, Merz CJ. UCI Repository of Machine Learning Databases. Irvine, CA, USA: University of California-Irvine, 2013.

[19] MacQueen JB. Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability; 21 June–18 July 1967; Berkeley, CA, USA. Berkeley, CA, USA: University of California Press. pp. 281-297.

[20] Rashedi E, Nezamabadi-pour H, Saryazdi S. GSA: A gravitational search algorithm. Inform Sciences 2009; 179: 2232-2248.

[21] Kahraman HT. Meta-heuristic linear modeling technique for estimating excitation current of synchronous motor. Turk J Elec Eng & Comp Sci 2014; 22: 1637-1652.

[22] Huang A. Similarity measures for text document clustering. In: Proceedings of the Sixth New Zealand Computer Science Research Student Conference; 14–18 April 2008; Hamilton, New Zealand. pp. 49-56.

[23] Porter MF. An algorithm for suffix stripping. Program 1980; 14: 130-137.

[24] Karol S, Mangat V. Evaluation of a text document clustering approach based on particle swarm optimization. International Journal of Computer Science & Network Security 2013; 13: 69-90.

[25] Berry MW, Browne M. Lecture Notes in Data Mining. Singapore: World Scientific, 2006.

[26] Lin YS, Jiang JY, Lee SJ. A similarity measure for text classification and clustering. IEEE T Knowl Data En 2014; 26: 1575-1590.

[27] Mirhosseini M, Mashinchi M, Nezamabadi-pour H. Improving n-similarity problem by genetic algorithm and its application in text document resemblance. Fuzzy Information and Engineering 2014; 6: 263-278.