

Article

# Clustering Using an Improved Krill Herd Algorithm

Qin Li <sup>1</sup> and Bo Liu <sup>2,\*</sup>

<sup>1</sup> School of Information Science and Technology, Jinan University, Guangzhou 510630, China; jnuliqin@gmail.com

<sup>2</sup> Department of Information Science and Technology, Jinan University, Guangzhou 510630, China

\* Correspondence: tlxldd@jnu.edu.cn or ddxllb@163.com; Tel.: +86-20-8522-0227

Academic Editor: Javier Del Ser Lorente

Received: 27 March 2017; Accepted: 12 May 2017; Published: 17 May 2017

**Abstract:** In recent years, metaheuristic algorithms have been widely used in solving clustering problems because of their good performance and application effects. Krill herd algorithm (KHA) is a new effective algorithm to solve optimization problems based on the imitation of krill individual behavior, and it is proven to perform better than other swarm intelligence algorithms. However, there are some weaknesses yet. In this paper, an improved krill herd algorithm (IKHA) is studied. Modified mutation operators and updated mechanisms are applied to improve global optimization, and the proposed IKHA can overcome the weakness of KHA and performs better than KHA in optimization problems. Then, KHA and IKHA are introduced into the clustering problem. In our proposed clustering algorithm, KHA and IKHA are used to find appropriate cluster centers. Experiments were conducted on University of California Irvine (UCI) standard datasets, and the results showed that the IKHA clustering algorithm is the most effective.

**Keywords:** data clustering; krill herd; improved algorithm; mutation operators

## 1. Introduction

Clustering is an important research direction in data analysis. This method does not make any statistical hypothesis on data and, thus, is called unsupervised learning in pattern recognition and data mining. Clustering is mainly used in text clustering [1], search engine optimization [2], landmark selection [3], face recognition [4], and medicine and biology [5].

Clustering is one of the most difficult and challenging problems in machine learning. The variety of clustering algorithms is roughly divided into three main types, namely, overlapping (so-called non-exclusive) [6], partitional [7], and hierarchical [8]. Regardless of the type of clustering algorithm applied, the main goal is to maximize homogeneity within each cluster and heterogeneity among different clusters. In other words, objects that belong to the same cluster should be more similar to each other than objects that belong to different clusters.

Although present algorithms have their own advantages, they are sensitive to the initialization parameters and it is difficult to find their optimal clusters. In recent years, optimization methods inspired by natural phenomena have provided new ways to solve clustering problems. A swarm of individuals are employed to explore the search space and obtain an optimal solution, such as genetic algorithms (GA) [9], particle swarm optimization algorithms (PSO) [10], and ant colony optimization (ACO) [11], among others. Other novel swarm intelligence algorithms have been proposed, such as harmony search (HS) [12], honeybee mating optimization algorithm (HBMO) [13], artificial fish swarm algorithm (AFSA) [14], artificial bee colony (ABC) [15], firefly algorithm (FA) [16], monkey algorithm (MA) [17], bat algorithm (BA) [18], and many others.

The krill herd algorithm (KHA) [19] is a novel swarm algorithm that is based on the simulation of the herding behavior of krill individuals and the minimum distances of each individual krill from

food and from the highest density of the herd, which are considered as the objective functions for krill movement. Although proposed recently, KHA has quickly been applied to multiple scenarios. Amudhavel et al. [20] used KHA to optimize a peer-to-peer network. KHA is applied in a smartphone ad hoc network [21]. Kowalski et al. [22] used KHA for learning an artificial neural network. In [23], KHA demonstrated better performance compared with well-known algorithms, such as PSO and GA. Gandomi and Alavi [19] illustrated that KHA with a crossover operator is superior to other well-known algorithms, including differential evolution (DE) [24], biogeography-based optimization (BBO) [25], and ACO.

Although KHA outperforms many other swarm intelligent algorithms [19], the algorithm cannot search globally particularly well [26]. In [27], a free search KHA for function optimization was proposed to improve the feasibility and effectiveness of KHA. An improved KHA with a linear decreasing step was proposed by Li et al. [28]. Furthermore, a new KHA that improved the original genetic operator by modifying the mutation mechanism and adding a new updated scheme will be demonstrated in our paper.

In this paper, we apply KHA as an optimization means to transform clustering into an optimization problem. In other words, we use individual krill to represent  $K$  cluster centers ( $K$  is the number of clusters), and KHA is used to search for the optimal clustering center. According to the principle of minimum distance of centers, all of the objects of the dataset are divided into different clusters, which leads to obtaining clustering results.

The rest of the paper is organized as follows: In Section 2, details of KHA are introduced. Section 3 briefly explains the improved KHA. In Section 4 clustering with the IKHA approach is proposed. Section 5 presents the experimental results of our proposed algorithm. Finally, the summary and future works are provided in Section 6.

## 2. Introduction to Krill Herd Algorithm

KHA is based on the simulation of the herding of krill swarms in response to specific biological and environmental processes. Nearly all necessary coefficients for KHA are obtained from real-world empirical studies [19].

In nature, the adaptability of an individual is judged by its distance to food and the maximum density of the krill population. Thus, based on the assumption of an imaginary distance, the fitness is the value of the objective function. Within a two-dimensional space, the specific location of the individual krill varies with time depending on the following three actions [19]:

- movement induced by other krill individuals;
- foraging activity; and
- random diffusion.

KHA uses the Lagrangian model to extend the search space to an  $n$ -dimensional decision space as:

$$\frac{dX_i}{dt} = N_i + F_i + D_i \quad (1)$$

where  $N_i$  is the motion of the  $i$ th krill induced by other krill individuals,  $F_i$  represents the foraging activity, and  $D_i$  denotes the physical diffusion of the krill individuals.

The explanations for basic KHA are given as follows:

### (1) Motion induced by other krill individuals

According to theoretical arguments, individual krill maintain a high density and move due to mutual effects. The direction of motion induced,  $\alpha_i$ , is estimated from the local swarm density (local

effect), target swarm density (target effect), and repulsive swarm density (repulsive effect). For an individual krill, the motion can be defined as:

$$N_i^{new} = N^{max} \alpha_i + \omega_n N_i^{old} \quad (2)$$

where:

$$\alpha_i = \alpha_i^{local} + \alpha_i^{target} \quad (3)$$

$N^{max}$  is the maximum induced speed,  $\omega_n$  is the inertia weight of the motion induced in the range  $[0, 1]$ s,  $N_i^{old}$  is the last motion induced,  $\alpha_i^{local}$  is the local effect provided by the neighbors, and  $\alpha_i^{target}$  is the target direction effect provided by the best individual krill. According to the measured values of the maximum induced speed ( $N^{max}$ ),  $N^{max}$  is taken as  $0.01 \text{ (ms}^{-1}\text{)}$  in [19].

Different strategies can be used in choosing the neighbor. Based on the actual behavior of krill individuals, a sensing distance ( $d_s$ ) should be determined around a krill individual and the neighbors should be found.

The sensing distance for each krill individual can be determined by using different heuristic methods. Here, the sensing distance is determined by using the following formula for each iteration:

$$d_{s,i} = \frac{1}{5N} \sum_{j=1}^N \|X_i - X_j\| \quad (4)$$

where  $d_{s,i}$  is the sensing distance for the  $i$ th krill individual and  $N$  is the number of the krill individuals, and  $X_i$  represents the related positions of  $i$ th krill. If the distance of  $X_i$  and  $X_j$  is less than the defined sensing distance ( $d_{s,i}$ ),  $X_j$  is a neighbor of  $X_i$ .

### (2) Foraging motion

This movement is intended to comply with two criteria. The first is food location, and the second is previous experience about the food location. For the  $i$ th krill, the foraging motion can be expressed as:

$$F_i = V_f \beta_i + \omega_f F_i^{old} \quad (5)$$

where:

$$\beta_i = \beta_i^{food} + \beta_i^{best} \quad (6)$$

where  $V_f$  is foraging speed,  $\omega_f$  is inertia weight of the foraging motion in the range  $[0, 1]$ ,  $\beta_i^{food}$  is the attractive food, and  $\beta_i^{best}$  is the effect of the best fitness of the  $i$ th krill so far. According to measured values of the foraging speed,  $V_f$  is taken as  $0.02 \text{ ms}^{-1}$  in [19].

Food effect is defined in terms of its location. The center of food should be found and then formulated for food attraction. This solution cannot be determined, but can be estimated. In this study, the virtual center of food concentration is estimated according to the fitness distribution of krill individuals, which is inspired by the "center of mass" concept. The center of food for each iteration is formulated as:

$$X^{food} = \frac{\sum_{i=1}^N \frac{1}{K_i} X_i}{\sum_{i=1}^N \frac{1}{K_i}} \quad (7)$$

where  $K_i$  is the objective function value of the  $i$ th krill individual.

### (3) Physical diffusion

The physical diffusion of the krill individuals is considered a random process. This motion can be expressed in terms of a maximum diffusion speed and a random directional vector. The formula is as follows:

$$D_i = D^{max} \left(1 - \frac{I}{I_{max}}\right) \delta \quad (8)$$

where  $D^{\max}$  is the maximum diffusion speed, and  $\delta$  is the random directional vector and its arrays are random values between  $-1$  and  $1$ .  $I$  is the actual iteration number and  $I_{\max}$  is the maximum number of iterations.

(4) Motion process of KHA

Defined motions regularly change the krill position toward the best fitness. The foraging motion and motion induced by other krill individuals contain two local ( $\alpha_i^{local}, \beta_i^{best}$ ) and two global strategies ( $\alpha_i^{target}, \beta_i^{food}$ ), which work simultaneously and create a powerful algorithm. Using diverse operative parameters of the motion throughout the time, the position vector of a krill individual during interval  $t$  to  $t + \Delta t$  is expressed by the following equation:

$$X_i(t + \Delta t) = X_i(t) + \Delta t \frac{dX_i}{dt} \tag{9}$$

where  $X_i(t + \Delta t)$  represents the updated krill individual position, and  $X_i(t)$  represents the current position. Note that  $\Delta t$  is considered the most important constant and should be tuned carefully based on the optimization problem. This is because this parameter works as a scale factor of the speed vector, and  $\Delta t$  can be obtained from the following formula:

$$\Delta t = C_t \sum_{j=1}^{NV} (UB_j - LB_j) \tag{10}$$

where  $NV$  is the total number of variables, and  $LB_j$  and  $UB_j$  are the lower and upper bounds of the  $j$ th variables ( $j = (1, 2, \dots, NV)$ ), respectively. Therefore, the absolute of their subtraction shows the search space. It is empirically found that  $C_t$  is a constant number between  $[0, 2]$ . It is also obvious that low values of  $C_t$  let the krill individuals search the space carefully.

(5) Genetic operators

Crossover operation is the use of a binomial crossover scheme to update the  $m$ th components of the  $i$ th krill by the following formula:

$$X_{i,m} = \begin{cases} x_{r,m} & rand_{i,m} < C_r \\ x_{i,m} & else \end{cases} \tag{11}$$

$$C_r = 0.2 \hat{K}_{i,best} \tag{12}$$

where  $C_r$  is crossover probability, which is a random number between 0 and 1,  $r \in \{1, 2, \dots, i - 1, i + 1, \dots, N\}$ . Mutation is controlled by mutation probability ( $M_u$ ). The adaptive mutation scheme used is formulated as

$$X_{i,m} = \begin{cases} x_{gbest,m} + \mu(x_{p,m} - x_{q,m}) & rand_{i,m} < Mu \\ x_{i,m} & else \end{cases} \tag{13}$$

$$Mu = 0.05 / \hat{K}_{i,best} \tag{14}$$

where  $p, q \in \{1, 2, \dots, i - 1, i + 1, \dots, N\}$  and  $\mu$  is a number between 0 and 1. In  $\hat{K}_{i,best}$ , the nominator is  $K_i - K_{best}$ . Based on this new mutation probability, the mutation probability for the global best is equal to zero, which increases as fitness decreases.

### 3. Improved KHA

The KHA algorithm considers various motion characteristics of individual krill, as well as the global exploration and local exploitation ability. Through simulation and experiments [19], the

performance of the algorithm is better than that of the majority of swarm intelligence algorithms. However, recent studies show that the KHA algorithm has excellent local exploitation ability, but global exploration ability is not as strong, especially in the treatment of high-dimensional multimodal function optimization [29], because the algorithm cannot always converge rapidly. To solve the problem, selection and crossover operators are added to the basic KHA in [29], and [30] used a local search to explore around the solution obtained by the KHA. Inspired by these developments, we propose the improved KHA algorithm (IKHA) based on a modified mutation scheme and a new updated mechanism.

The main ideas of IKHA are as follows: First, we sorted the individuals of each generation according to the fitness value in ascending order. The first part included individuals with good fitness (individuals with fitness value among the top 10%, but apart from the global best), and the rest comprised the second part. For the first part, which we call sub-optimal individuals, the fitness value was close to the optimal individual, but worse than the optimal solution. In the process of optimization of this part, the individual does not have much effect. Another noteworthy point, based on Equatio (14) in the previous section, is that mutation probability ( $M_u$ ) for the global best is equal to zero and increases with decreasing fitness. In other words, the smaller the fitness value, the higher the probability of mutation. Thus, we can improve the mutation mechanism to use this part of the individual and allow them to find the potential solution in the vicinity of the optimal solution.

For the first part of the sub-optimal individuals, we use the individual's own neighbors  $x_a$  (a neighbor of  $x_i$ ) to optimize the mutation program instead of the original stochastic selection  $x_p, x_q$ . Specific operations observed the following formula, where SN is the abbreviation of sub-optimal individuals and  $\mu_{nn}$  is a number between 0 and 1:

$$X_{i,m} = \begin{cases} x_{i,m} + \mu_{nn}(x_{a,m} - x_{i,m}) & x_i \in SN, rand_{i,m} \leq Mu \\ x_{i,m} & x_i \in SN, rand_{i,m} > Mu \end{cases} \quad (15)$$

For the second part of the individuals, we only had to use good individuals to guide them toward a better direction of evolution. Therefore, we chose sub-optimal individuals to optimize the mutation program. The specific formula is as follows:

$$X_{i,m} = \begin{cases} x_{gbest,m} + \mu(x_{b,m} - x_{c,m}) & x_i \notin SN, rand_{i,m} \leq Mu \\ x_{i,m} & x_i \notin SN, rand_{i,m} > Mu \end{cases} \quad (16)$$

where  $x_b, x_c \in \{SN \mid SN\}$  are sub-optimal individuals.

Beyond the modified mutation mechanism, an updated operator is added in our approach. After many iterations, the KHA tends to stagnate. To avoid premature convergence in the early run phase, we added an updated mechanism to overstep the local extremum. In our approach, a parameter, the maximum number of stalls ( $S_{max}$ ), is added. Suppose that the  $K_{gbest}$  (the fitness value of the global best individual of the population) remains unchanged, and  $num_{samebest}$  (the number of unchanged iterations) is greater than  $S_{max}$ , then the updated formula is shown as follows:

$$X_{best}^{new1} = X_{best} + v_{best}(X_{best} - X_{\overline{SN}}), \text{ if } (num_{samebest} > S_{max}) \quad (17)$$

$$X_{best}^{new2} = X_{best} - v_{best}(X_{best} - X_{\overline{SN}}), \text{ if } (num_{samebest} > S_{max}) \quad (18)$$

where  $X_{\overline{SN}}$  is the average position of the SN, and  $v_{best}$  is a number between 0 and 1. If the fitness value of  $X_{best}^{new1}$  or  $X_{best}^{new2}$  is less than  $K_{gbest}$ , we replace the old position with the new position.  $S_{max}$ , which is defined as follows, and is a positive integer greater than zero and decreases with the increase of the iteration number:

$$S_{max} = \left\lceil s_{max} \left( 1 - \frac{I}{I_{max}} \right) \right\rceil \quad (19)$$

In IKHA, the optimized mutation scheme abandons the original randomly-selected individuals for mutations, and uses different mutations for individuals with different fitness values. With such a divide-and-rule strategy, we take full advantage of all individuals, as opposed to the KHA. For example, sub-optimal individuals can be used to find potentially better values, thereby preventing the algorithm from falling into a local optimum. For the remaining individuals, excellent individuals could guide them, thereby speeding up optimization. The purpose of the updated operation is to find the potential for the escape from the local solution at the later run phase of the process.

The time computational complexity of IKHA is the same as KHA, and the analysis is as follows: In KHA, for each krill in an iteration, the time complexity of calculating the sensing distance  $d_{s,i}$  is  $O(N)$ , so KHA's time computational complexity is  $O(I_{\max} \bullet N^2)$ ; in IKHA, the added updated operating is mainly according to Equations (17) and (18), and time computational complexity is  $O(1)$ . Moreover, with the improved mutation mechanism, we need to sort the individuals according to their fitness value, and we use a quick sort algorithm, whose time computational complexity is  $O(N \log N)$  in the average case, or  $O(N^2)$  in the worst case, but for every generation, one sorting operation is added, thus, the time computational complexity of IKHA is still  $O(I_{\max} \bullet N^2)$ .

To test IKHA further, we conducted the following experiments by using the Ackley function [31]. The Ackley function is defined as follows and its graph is shown in Figure 1:

$$f(x) = \sum_{i=1}^n -20 \exp \left( -0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} - \exp \left( \frac{1}{n} \sum_{i=1}^n \cos 2\pi x_i \right) \right) + 20 + e \quad (20)$$

The convergence graphs for the Ackley function is drawn in Figure 2. In our experiment, the number of iterations is set to 100, the population size is 50, and the results are obtained after 50 trials. For the KHA and the proposed IKHA, we set the same parameters  $N^{\max} = 0.01$ ,  $V_f = 0.02$ ,  $D^{\max} = 0.005$ ,  $S_{\max} = 5$ , and  $v_{best} = 0.5$  at the beginning and these parameters linearly decreased to 0.1 at the end in IKHA [32,33]. Regarding the convergence behavior of KHA and IKHA, both IKHA and KHA converged quickly in the early run phase, but IKHA converged faster than KHA. During the latter run, KHA began to stagnate after rapid convergence, but IKHA continued to find a better value. Thus, IKHA can quickly converge in the early iterations and jump out of the local optimum to find a better solution.

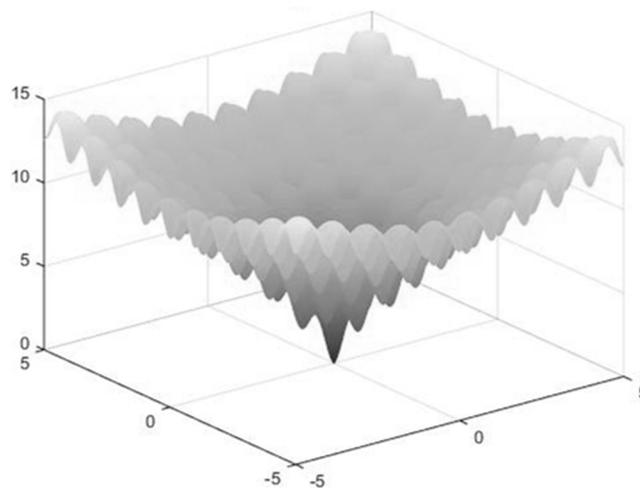


Figure 1. Ackley function graph.

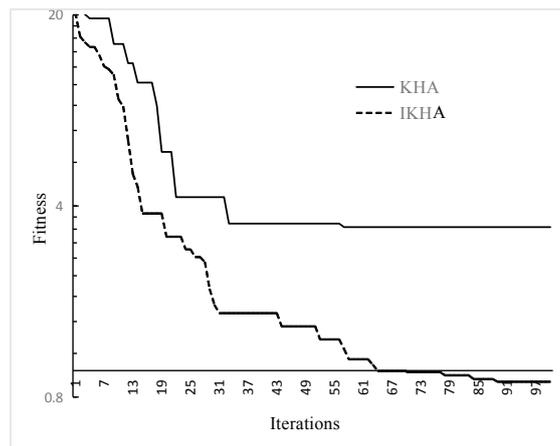


Figure 2. Comparison of convergence of the KHA and IKHA for Ackley (D = 20).

### 4. Clustering Algorithms with IKHA

#### 4.1. Basic Idea of Clustering

Data clustering, which is a NP-complete problem, finds heterogeneous data by minimizing some measure of dissimilarity. Given  $Dataset = \{data_1, data_2, \dots, data_n\}$ , clustering aims to divide the whole data into K clusters ( $K \leq n$ ),  $n$  is the total number of data objects, and the data objects of the same cluster are similar according to the similarity criteria. The similarity measure uses Euclidean distance:

$$dis(data_i, data_j) = \sqrt{\sum_{d=1}^D (data_{i,d} - data_{j,d})^2} \tag{21}$$

where  $i, j \in \{1, 2, \dots, n\}$ , and  $data_{i,d}$  is the  $d$ th attribute of the  $i$ th datum in  $\mathbb{R}^D$ ,  $dis(data_i, data_j)$  denotes the distance of  $data_i$  and  $data_j$ , and  $D$  is the number of attributes for each data object.

#### 4.2. Clustering Based on IKHA

Clustering is in accordance with appropriate indexes to find an optimal clustering process. The essence of clustering is the optimization process. What is important is finding ways to combine the optimization algorithm IKHA with clustering. By representing each krill in the IKHA as a clustering scheme, we find the optimal clustering scheme by choosing the appropriate objective function. A clustering scheme can be expressed by all clustering centers. That is, every krill  $X_i$  represents the  $K$  clustering centers:

$$X_i = \{C_1, C_2, \dots, C_k, \dots, C_{K-1}, C_K\} \tag{22}$$

$$C_k = \begin{pmatrix} C_k^1 \\ C_k^2 \\ \vdots \\ C_k^d \\ \vdots \\ C_k^{D-1} \\ C_k^D \end{pmatrix} \tag{23}$$

where  $d$  denotes the number of parameters of the data that will be clustered, and  $C_k^1$  represents the first parameter of the first cluster center. Each krill individual can be expressed as the following matrix:

$$X_i = \left\{ \begin{array}{ccccccc} C_1^1 & C_2^1 & \dots & C_k^1 & \dots & C_{K-1}^1 & C_K^1 \\ C_1^2 & C_2^2 & \dots & C_k^2 & \dots & C_{K-1}^2 & C_K^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ C_1^d & C_2^d & \dots & C_k^d & \dots & C_{K-1}^d & C_K^d \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ C_1^{D-1} & C_2^{D-1} & \dots & C_k^{D-1} & \dots & C_{K-1}^{D-1} & C_K^{D-1} \\ C_1^D & C_2^D & \dots & C_k^D & \dots & C_{K-1}^D & C_K^D \end{array} \right\} \quad (24)$$

In this study, one krill is used to represent a candidate solution to a problem, and the selected  $K$  initial cluster centers are potential solutions. One krill and  $K$  initial clustering center play similar roles in our algorithms. Thus, the mapping between a krill individual and  $K$  initial clustering centers can be established. In the coding method of the krill location structure, a set of initial cluster centers are generated randomly from the dataset points.

The whole krill population represents a variety of clustering schemes. In this manner, our aim is to find the optimal clustering centers. According to the principle of the minimal distance, data are categorized into the appropriate cluster. The description of the improved krill-herd clustering algorithm (IKHCA) is shown in Algorithm 1.

---

**Algorithm 1. Improved Krill Herd Clustering Algorithm (IKHCA)**

---

- (1) Define the parameters ( $K, I_{max}, N, N^{max}, V_f, D^{max}$ , and so on).
  - (2) Initialize  $N$  krills randomly as the initial clustering center.
  - (3) Evaluate each krill individual by fitness function.
  - (4) For each krill individual:
    - Perform three motions (motion induced by another individual, foraging motion, and physical diffusion).
    - Then, implement the crossover operator and the modified mutation operator (two mutation schemes were performed for individuals with different fitness levels).
    - Calculate the fitness according the krill's new position; if the new fitness is better than the older, update the krill individual position in the search space.
  - (5) Use the updated mechanism to update the krill's position if the new position is superior to the old position.
  - (6) Repeat Steps 4 and 5 until the stopping criteria are satisfied.
  - (7) Return to the best clustering solution.
- 

**5. Simulation and Experiment**

To investigate the performance of IKHCA, five clustering algorithms, namely, K-means [34], ACO [35], PSO [36], KHCA I in [30], and KHCA II, were compared. KHCA II is a clustering algorithm based on KHA [19]. Five datasets obtained from UCI Machine Learning Repository [37] were used in our experiment. The details of the data sets, including the name, number of classes, attributes, and records are presented in Table 1. Our experiments were conducted on Eclipse 4.6.0 with Windows 7 environment using Intel Core i7, 3.40 GHz, and 4 GB RAM.

**Table 1.** The details of selected datasets.

Name	Number of		
	Clusters	Parameters	Elements
Iris	3	4	150
Wine	3	13	178
Glass	6	9	214
Cancer	2	9	683
CMC	3	10	1473

Before the experiment, the setting of the parameters and the selection of the objective functions in KHCA II and IKHCA were specified. In KHCA II and IKHCA, we used the sum of squared error ( $I_{SSE}$ ) as the objective function directly, the formula is indicated in Equation (25). The low value of  $I_{SSE}$ , the higher the quality of the clustering is

$$I_{SSE} = \sum_{k=1}^K \sum_{i=1}^n \omega_{ik} (\text{dis}(\text{data}_i, C_k))^2 \tag{25}$$

$$\omega_{ik} = \left\{ \begin{array}{l} 1, \text{ if data } i \text{ belongs to cluster } k \\ 0, \text{ else} \end{array} \right\} \tag{26}$$

The parameters are set in accordance with [19,38]:

$$N^{\max} = 0.01;$$

$$V_f = 0.02; \text{ and}$$

$$D^{\max} = 0.005.$$

Here,  $C_t$  is set to 0.5, and the inertia weights ( $\omega_n, \omega_f$ ) are equal to 0.9 at the beginning of the search, and linearly decreased to 0.1 at the end to encourage exploitation. The size of the population is set to 25,  $s_{\max} = 5$ ,  $v_{best} = 0.5$  at the beginning, and linearly decreased to 0.1 at the end in IKHCA.

We compared the performance of different clustering algorithms from two aspects. First, we compared the objective function value of the different clustering algorithms in Table 2, and then we compared the accuracy of different clustering algorithms in Table 3. Accuracy is specifically expressed as follows:

$$\text{accuracy} = \left( \frac{\text{number of correctly placed data}}{\text{total number of data}} \right) \times 100 \tag{27}$$

Table 2 lists the best and worst means of the solution, and ranks the algorithms based on the mean values for all datasets in Table 1. As compared, algorithm results are directly taken from [30]. KHCA II and IKHCA algorithms were executed 100 times independently with the same parameters described in this paper, except that the maximum number of generations was set to 200. As shown in Table 2, IKHCA obtained better solutions for the best and worse than other algorithms on the Wine, Glass, Cancer, and CMC datasets, but not on Iris. KHCA II obtained the first solution for best on the Iris dataset. However, KHCA II generated a poor solution for the worst with respect to the Iris dataset. Then, we observed that IKHCA achieved the best solutions from mean values on all datasets, except Glass. However, IKHCA is very close to the results obtained by the KHCA II algorithm on the Glass dataset. From the experimental results, our proposed algorithm achieved better optimal solutions with improved stability in a limited number of iterations. IKHCA ranked first in all algorithms.

**Table 2.** Objective function values obtained by the algorithms.

Data Set	Criteria	K-means	ACO	PSO	KHCA I	KHCA II	IKHCA
Iris	Best	98.5	97.4	97.1	96.4	96.65	96.66
	Worst	117.4	99.2	100.5	103.1	97.68	96.67
	Mean	104.7	97.8	98.8	98.6	96.67	96.66
	Rank	6	3	5	4	2	1
Wine	Best	16,562.6	16,510.3	16,336.4	16,328.1	16,293.01	16292.12
	Worst	17,995.9	16,535.8	16,426.4	16,430.9	17,710.16	16,589.23
	Mean	17,101.5	16,528.5	16,396.3	16,384.2	16,490.17	16,305.51
	Rank	6	5	3	2	4	1
Glass	Best	225.3	219.8	271.1	216.2	210.85	210.30
	Worst	263.1	258.1	286.8	255.9	246.16	223.03
	Mean	248.2	241.3	279.3	238.3	215.86	215.90
	Rank	5	4	6	3	1	2

Table 2. Cont.

Data Set	Criteria	K-means	ACO	PSO	KHCA I	KHCA II	IKHCA
Cancer	Best	2994.9	2966.6	2974.4	2945.8	2964.39	2964.39
	Worst	3651.5	3098.9	3289.1	3088.8	3571.53	2971.15
	Mean	3131.3	2984.9	3102.8	2981.4	2995.87	2968.16
	Rank	6	3	5	2	4	1
CMC	Best	5891.3	5721.8	5795.5	5711.2	5700.16	5692.20
	Worst	5989.4	5836	5866	5821.3	5791.52	5695.02
	Mean	5945.1	5773	5823.1	5759.4	5760.29	5694.91
	Rank	6	4	5	2	3	1
	Mean Rank	5.8	3.8	4.8	2.6	2.8	1.2
	Final Rank	6	4	5	2	3	1

In Table 3, the clustering accuracies of IKHCA and other clustering algorithms are given, and part of the results were obtained directly from [30], with the bold font indicating the best results. At a glance, one can easily see that the last three clustering algorithms (KHCA I, KHCA II, and IKHCA) by using KHA are obviously better than the K-means, ACO, and PSO algorithms. It can be seen that the introduction of KHA into the clustering problem is reasonable and effective. Based on these results, IKHCA is proved to be the best algorithm with respect to objective function value and accuracy.

Table 3. Accuracy obtained by the algorithms.

Data Set	K-means	ACO	PSO	KHCA I	KHCA II	IKHCA
Iris	83.3	88.5	87.7	89.07	89.67	<b>90.67</b>
Wine	63.62	70.6	70.4	71.12	70.99	<b>73.03</b>
Glass	60.8	64.7	56.65	64.98	65.01	<b>65.88</b>
Cancer	93.37	94.1	94.62	95.01	95.02	<b>95.16</b>
CMC	41.8	45.5	45.2	45.5	45.55	<b>45.62</b>

## 6. Conclusions and Future Work

KHA is a good swarm intelligent heuristic algorithm that could be gradually applied to address real-world problems. For the original KHA algorithm that could not always converge rapidly and search globally particularly well, we proposed IKHA, which improved the original mutation mechanism to provide two different mutation schemes and introduced an updated mechanism. In IKHA, we were in accordance with the fitness of individuals, set different mutation schemes according to their own conditions, made outstanding individuals look for better solutions, and the rest moved closer to the good individual. Then, through the updated mechanism, optimal individuals looked for potential solutions in the surrounding space to avoid being stuck in the local optimal zone. Experimental results showed that IKHA performed better than KHA.

Several clustering algorithms depend highly on the initial states and always converge to the nearest local optimum from the starting position of the search. In order to find the optimal clustering center, we applied the IKHA to solve an actual clustering problem and proposed the improved krill-herd clustering algorithm (IKHCA). According to the experiments, the IKHCA had better efficiency than, and outperformed, other well-known clustering approaches. Moreover, the results of the experiments show that the IKHA can successfully be introduced in clustering problems and perform best in almost all experimental datasets. In the future, there are several issues that can be further studied, such as utilizing the optimization ability of the IKHA to find the optimal cluster number and apply the IKHA to other scenarios to solve a wide range of real-world problems.

**Acknowledgments:** This work was supported by the National Natural Science Foundation of China (U1431227); and the Guangzhou Scientific and Technological Project (201604010037).

**Author Contributions:** All of the authors contributed to the content of this paper. Qin Li participated in the algorithm analyses, design, algorithm implementation and draft preparation. Bo Liu analyzed the experimental data and revised this paper. All authors read and approved the manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Abualigah, L.M.Q.; Hanandeh, E.S. Applying Genetic Algorithms to Information Retrieval Using Vector Space Model. *Int. J. Comput. Sci. Eng. Appl.* **2015**, *5*, 19–28.
2. Carpineto, C.; Osiriński, S.; Romano, G.; Weiss, D. A Survey of Web Clustering Engines. *Acm Comput. Surv.* **2009**, *41*, 17. [[CrossRef](#)]
3. Rafailidis, D.; Constantinou, E.; Manolopoulos, Y. Landmark selection for spectral clustering based on Weighted PageRank. *Futur. Gener. Comput. Syst.* **2017**, *68*, 465–472. [[CrossRef](#)]
4. Wu, B.; Hu, B.G.; Ji, Q. A Coupled Hidden Markov Random Field Model for Simultaneous Face Clustering and Tracking in Videos. *Pattern Recognit.* **2016**, *64*, 361–373. [[CrossRef](#)]
5. Kaya, I.E.; Pehlivanlı, A.C.; Sekizkardeş, E.G.; Ibrikli, T. PCA Based Clustering for Brain Tumor Segmentation of T1w MRI Images. *Comput. Method. Progr. Biomed.* **2017**, *140*, 19–28. [[CrossRef](#)] [[PubMed](#)]
6. Macqueen, J. Some Methods for Classification and Analysis of MultiVariate Observations. *Proc. Berkeley Symp. Math. Stat. Probab.* **1967**, *1*, 281–297.
7. Jain, A.K. *Data Clustering: 50 Years Beyond K-Means*; Springer: Berlin, Germany, 2008.
8. Langfelder, P.; Zhang, B.; Horvath, S. Defining clusters from a hierarchical cluster tree: The Dynamic Tree Cut package for R. *Bioinformatics* **2008**, *24*, 719. [[CrossRef](#)] [[PubMed](#)]
9. Goldberg, D.E. Genetic Algorithms in Search, Optimization and Machine Learning. *Choice Rev. Online* **1989**, *27*, 2104–2116.
10. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the 1995 IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
11. Xu, B.; Zhu, J.; Chen, Q. *Ant Colony Optimization. New Advances in Machine Learning*; InTech: Jiangsu, China, 2010; pp. 1155–1173.
12. Zong, W.G.; Kim, J.H.; Loganathan, G.V. A New Heuristic Optimization Algorithm: Harmony Search. *Simul. Trans. Soc. Model. Simul. Int.* **2001**, *76*, 60–68.
13. Abbass, H.A. MBO: Marriage in honey bees optimization—a Haplometrosis polygynous swarming approach. In Proceedings of the 2001 Congress on Evolutionary Computation, Seoul, Korea, 27–30 May 2001; Volume 1, pp. 207–214.
14. Xiaolei, L.I.; Shao, Z.; Qian, J. An Optimizing Method Based on Autonomous Animats: Fish-swarm Algorithm. *Syst. Eng. Theory Pract.* **2002**, *11*, 32–38.
15. Karaboga, D. *An Idea Based on Honey Bee Swarm for Numerical Optimization*; Erciyes University, Engineering Faculty, Computer Engineering Department: Kayseri, Turkey, 2005.
16. Yang, X.S. Firefly Algorithms for Multimodal Optimization. *Mathematics* **2009**, *5792*, 169–178.
17. Zhao, R.; Tang, W. Monkey algorithm for global numerical optimization. *J. Uncertain Syst.* **2008**, *2*, 165–176.
18. Yang, X.S. A New Metaheuristic Bat-Inspired Algorithm. *Comput. Knowl. Technol.* **2010**, *284*, 65–74.
19. Gandomi, A.H.; Alavi, A.H. Krill herd: A new bio-inspired optimization algorithm. *Commun. Nonlinear Sci. Numer. Simul.* **2012**, *17*, 4831–4845. [[CrossRef](#)]
20. Amudhavel, J.; Sathian, D.; Raghav, R.S.; Pasupathi, L.; Baskaran, R.; Dhavachelvan, P. A Fault Tolerant Distributed Self Organization in Peer To Peer (P2P) Using Krill Herd Optimization. In Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology, Unnao, India, 6–7 March 2015.
21. Amudhavel, J.; Kumarakrishnan, S.; Gomathy, H.; Jayabharathi, A.; Malarvizhi, M.; Prem Kumar, K. An Scalable Bandwidth Reduction and Optimization in Smart Phone Ad hoc Network (SPAN) Using Krill Herd Algorithm. In Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology, Unnao, India, 6–7 March 2015.
22. Kowalski, P.A.; Lukasik, S. Training Neural Networks with Krill Herd Algorithm. *Neural Proc. Lett.* **2016**, *9463*, 5–17. [[CrossRef](#)]

23. Chaturvedi, S.; Pragma, P.; Verma, H.K. Comparative analysis of particle swarm optimization, genetic algorithm and krill herd algorithm. In Proceedings of the 2015 International Conference on Computer, Communication and Control, Indore, India, 10–12 September 2015.
24. Storn, R.; Price, K. Differential Evolution—A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [[CrossRef](#)]
25. Simon, D. Biogeography-Based Optimization. *IEEE Trans. Evol. Comput.* **2009**, *12*, 702–713. [[CrossRef](#)]
26. Wang, G.; Guo, L.; Gandomi, A.H.; Cao, L.; Alavi, A.H.; Duan, H.; Li, J. Lévy-Flight Krill Herd Algorithm. *Math. Probl. Eng.* **2013**, *2013*, 1–14. [[CrossRef](#)]
27. Li, L.; Zhou, Y.; Xie, J. A Free Search Krill Herd Algorithm for Functions Optimization. *Math. Probl. Eng.* **2014**, *2014*, 1–21. [[CrossRef](#)]
28. Li, J.; Tang, Y.; Hua, C.; Guan, X. An improved krill herd algorithm: Krill herd with linear decreasing step. *Appl. Math. Comput.* **2014**, *234*, 356–367. [[CrossRef](#)]
29. Wang, G.G.; Gandomi, A.H.; Alavi, A.H. Stud krill herd algorithm. *Neurocomputing* **2014**, *128*, 363–370. [[CrossRef](#)]
30. Nikbakht, H.; Mirvaziri, H. A new clustering approach based on K-means and Krill Herd algorithm. In Proceedings of the 2015 IEEE Electrical Engineering, Tehran, Iran, 10–14 May 2015; pp. 662–667.
31. Gandomi, A.H.; Yang, X.S. Benchmark Problems in Structural Optimization. In *Computational Optimization, Methods and Algorithms*; Springer: Berlin, Germany, 2011; pp. 259–281.
32. Price, H.J. Swimming behavior of krill in response to algal patches: A mesocosm study. *Limnol. Oceanogr.* **1989**, *34*, 649–659. [[CrossRef](#)]
33. Morin, A.; Okubo, A.; Kawasaki, K. Acoustic data analysis and models of krill spatial distribution. In *Scientific Committee for the Conservation of Antarctic Marine Living Resources; Selected Scientific Papers, Part I*; Scientific Committee: Tasmania, Australia, 1988; pp. 311–329.
34. Hartigan, J.A.; Wong, M.A. Algorithm AS 136: A K-Means Clustering Algorithm. *Appl. Stat.* **1979**, *28*, 100–108. [[CrossRef](#)]
35. Shelokar, P.S.; Jayaraman, V.K.; Kulkarni, B.D. An ant colony approach for clustering. *Anal. Chim. Acta* **2004**, *509*, 187–195. [[CrossRef](#)]
36. Chen, C.Y.; Ye, F. Particle swarm optimization algorithm and its application to clustering analysis. In Proceedings of the 2004 IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan, 21–23 March 2004; Volume 2, pp. 789–794.
37. Blake, C.L.; Merz, C.J. University of California at Irvine Repository of Machine Learning Databases. 1998. Available online: <http://www.ics.uci.edu/mllearn/MLRepository.html> (accessed on 15 May 2017).
38. Kowalski, P.A.; Lukasik, S. Experimental Study of Selected Parameters of the Krill Herd Algorithm. In *Intelligent Systems*; Springer: Cham, Switzerland, 2014; pp. 473–485.

