International Journal of Hybrid Information Technology
Vol.9, No.2 (2016), pp. 183-194
http://dx.doi.org/10.14257/ijhit.2016.9.2.16

Clustering with a Novel Global Harmony Search Algorithm for Image Segmentation

Lei Wang[1], Yufeng YI[2,3] and Jiao Liu[1]
[1]*Department of Public Order, National Police University of China, Shenyang, 110854, China*
[2]*Academy of Opt-Electronics, China Electronic Technology Group Corporation, Tianjin, 300000, China*
[3]*Key Laboratory of Electronics Information Control and Security Technology, China Electronic Technology Group Corporation, Sanhe, 065201, China*
Correspondent: Lei Wang, E-mail: leonwang521@126.com

Abstract

A state-feedback based harmony search (SFHS) algorithm to clustering data is presented to solve the problems that the traditional evolutionary-based clustering algorithms easily trap into the local optimum. Firstly, the state-feedback mechanism is introduced into HS algorithm, and the harmony memory difference metric is defined to adaptively adjust harmony memory considering rate and pitch adjusting bandwidth, which makes the convergence and efficiency of the harmony search (HS) algorithm improved obviously. Secondly, in the SFHS-based clustering algorithm, the decision variables in the harmony vector represent cluster centers, and the harmony vector represents a partition of data, the best partition of data can be obtained by updating the harmony memory. Finally, a novel validity metric is presented to determine the right number of clusters. Simulation experiments have been carried out on remote sensing images and animal images, and the relevant results are compared with the ANT-based clustering algorithm and the traditional HS-based clustering algorithms, it shows that the SFHS-based clustering algorithm has better convergence.

Keywords*: State-feedback, harmony search algorithm, clustering, ant algorithm*

1. Introduction

Clustering [1-3] is one of the most important unsupervised data classification method, which concerns with finding the best partition in the data sets so that the data in the same cluster is more similar to each other than to those in other clusters. Clustering has widespread applications such as in machine learning [4], pattern recognition [5], image analysis [6-7], information retrieval [8-9], and so on. Since traditional clustering approaches are sensitive to initial cluster centers, optimization methods have been successfully proposed to solve this problem [10-11]. For instance, genetic algorithm (GA) and ant algorithm (ANT) are used to find initial cluster centers. In addition, a fuzzy partition validity metric is used as the optimization criterion to determine the right number of clusters in data sets.

It is obvious that there is an always great need for more efficient optimization algorithm for clustering. Inspired by GA-based clustering and ANT-based clustering methods, the state-feedback based harmony search (SFHS) algorithm is proposed and used in clustering data. Harmony search (HS) algorithm [12-15] is a meta-heuristic global optimization method, and it has been successfully applied in many areas and shows better performance compared to genetic algorithm and ant algorithm in solving many problems.

In this paper, the state-feedback mechanism is introduced into HS to adaptively

ISSN: 1738-9968 IJHIT

adjust HS parameters, which makes the HS algorithm converge to the global best optimum quickly. In HS-based clustering algorithm, decision variables in harmony vector denote cluster centers, and harmony vector represents a partition of data. The best optimized partition can be obtained by updating harmony memory. In addition, a novel validity metric is proposed to find the right number of clusters. Experiments have been carried out on large-scale data sets such as remote sensing images and animal images. The analysis of the results indicates the superiority of SFHS-based clustering approach over those using ANT-based clustering approach and traditional HS-based clustering approach.

The paper is organized as follows. In Section 2, SFHS algorithm is presented. Section 3 provides an overview of FCM. Section 4 contains a description of applying SFHS in clustering. In Section 5, results of experiments are presented and discussed. We end the paper with some conclusions in Section 6.

2. The Harmony Search Algorithm

Harmony search algorithm is proposed by Z.W Geem, which is inspired by the improvisation process of musicians. In HS algorithm, a harmony (solution vector) is played by many musicians (decision variables) together, and each musician adjusts a pitch (a value) of their musical instrument to find a perfect state (global optimum). As a relatively new meta-heuristic algorithm, it has grown rapidly and many studies on HS have been published. In general, the steps of harmony search algorithm are as follows:

2.1. Initialize the Problem and HS Parameters

The optimization problem is defined as optimizing (Minimize or Maximize) objective function f(x), where x is a N-dimensional candidate solution vector consisting of decision variables x_i（i=1,2,…N）which is in ranges [x_i^L , x_i^U]. Both x_i^L and x_i^U are lower and upper bounds of each decision variable, respectively. In addition, harmony memory size (HMS), harmony memory considering rate (HMCR), pitch adjusting rate (PAR), the number of improvisations (NI) and pitch adjusting bandwidth (*bw*), those parameters are also specified in this step.

2.1.1. Title Initialize the Harmony Memory

The initial harmony memory matrix is randomly generated in search space, as shown in Eq.(1):

$$\mathrm{HM}=\begin{bmatrix} x_1^1 & \cdots & x_N^1 & | & f\left(\boldsymbol{x}^1\right) \\ \vdots & \ddots & \vdots & | & \vdots \\ x_1^{\mathrm{HMS}} & \cdots & x_N^{\mathrm{HMS}} & | & f\left(\boldsymbol{x}^{\mathrm{HMS}}\right) \end{bmatrix} \quad (1)$$

2.1.2. Improvise a New Harmony

Generating a new harmony vector $x^{'}$=($x_1^{'}$, $x_2^{'}$,⋯, $x_N^{'}$), which is determined by three rules: memory consideration, pitch adjustment and random selection. The procedure works as follows:

```
for each i∈[1,N] do
   r=rand()
   if r≤HMCR then
      x_i' = x_i^j (j=ceil(rand()×HMS)) %memory consideration
       if r≤PAR then
          x_i' = x_i'±rand()×bw %pitch adjustment
```



```
        if x_i'<x_i^L
          x_i'=x_i^L
        elseif x_i'>x_i^U
          x_i'=x_i^U
        end
      end
   else %random selection
      x_i' = x_i^L+ rand()×(x_i^U -x_i^L)
   end
end
```

where ceil(x) rounds elements of x to the nearest integers greater than or equal to x, and rand() is uniformly generated random number in ranges [0,1].

2.1.3 Update Harmony Memory

If the objective function value of current harmony vector x' is better than the worst harmony in the HM, the worst harmony in the HM will be replaced by current harmony x'.

2.1.4 Check the Stopping Criterion

The procedure is terminated if the stopping criterion (maximum number of iterations) is satisfied. Otherwise, go to 2.1.2.

2.2. Global-best Harmony Search

HMCR, PAR and *bw* are very important parameters in finding global optimal solution vectors in HS algorithm. Since traditional HS algorithm utilizes a constant for the parameters HMCR、PAR and *bw* in the whole iteration, efficiency and successful rate of HS algorithm reduced. Mahdavi et al (2007) proposed an improve harmony search algorithm (IHS), which dynamically updates pitch adjusting rate (PAR) and pitch adjusting bandwidth (*bw*) to adjust convergence rate of algorithm to optimal solution. The IHS dynamically updates PAR and *bw* using Eq. (2) and Eq. (3).

$$\mathrm{PAR}(t) = \mathrm{PAR}_{\min} + \cdots$$

$$\cdots \frac{(\mathrm{PAR}_{\max} - \mathrm{PAR}_{\min}) \times t}{T} \qquad (2)$$

$$\mathrm{bw}(t) = \mathrm{bw}_{\max} \exp\left(\frac{\ln\left(\dfrac{\mathrm{bw}_{\min}}{\mathrm{bw}_{\max}}\right) \times t}{T} \right) \qquad (3)$$

where $\mathrm{PAR}_{\min}$ and $\mathrm{PAR}_{\max}$ are minimum adjusting rate and maximum adjusting rate, respectively; $bw_{\min}$ and $bw_{\max}$ are minimum bandwidth and maximum bandwidth, respectively; t is the current iteration number; T is maximum iteration number.

In early iterations, small PAR values and large *bw* values can help to improve global search capabilities and improve the convergence rate to global best solution. In final iterations, large PAR values and small *bw* values can increase the fine-tuning of solution vectors and cause the improvement of best solutions.

2.3. State-Feedback based Harmony Search Algorithm

Inspired by the variant of HS (known as IHS), the state-feedback based harmony search (SFHS) algorithm is proposed in this paper. In proposed algorithm, the state-feedback mechanism is introduced into HS to adaptively adjust parameters HMCR and *bw*. Specifically, parameters in HMCR and *bw* in HS are seen as controlled variables in automatic control system. With a feedback regulation, parameters in HMCR and *bw* can meet actual needs for different iterations, and the ability of SFHS for finding the global optimum improved obviously.

The main difference between traditional HS and SFHS algorithm is the way of adjusting HMCR and *bw*. To improve the performance of traditional HS algorithm, SFHS algorithm uses parameters in HMCR and *bw* as controlled variables in improvisation step. Firstly, we define harmony memory difference metric T to adaptively update HMCR and *bw*, and it can be expressed as follow:

$$T = \sqrt{\sum_{i=1}^{N}\left(x_i^{best} - x_i^{worst}\right)^2} \qquad (4)$$

where x^{best} and x^{worst} are best harmony and worst harmony in harmony memory, respectively; subscript i denotes ith decision variables. HMCR and *bw* are updated adaptively with iteration number can be expressed as follow:

$$\mathrm{HMCR}(t) = \mathrm{HMCR}_{\min} + \cdots$$
$$\cdots \frac{(\mathrm{HMCR}_{\max} - \mathrm{HMCR}_{\min}) \times T(t)}{T_{\max}} \qquad (5)$$

$$bw(t) = bw_{\min} + \frac{(bw_{\max} - bw_{\min}) \times T(t)}{T_{\max}} \qquad (6)$$

where $\mathrm{HMCR}_{\max}$ and $\mathrm{HMCR}_{\min}$ are maximum harmony memory considering rate and minimum harmony memory considering rate, respectively; $bw_{\max}$ and $bw_{\min}$ are maximum adjusting bandwidth and minimum adjusting bandwidth, respectively; $T_{\max}$ is maximum harmony memory difference metric; t represents current iteration number.

The HS algorithm is easy to trap into local optimum when the difference between best harmony and worst harmony is smaller. Let the value of HMCR be smaller, the probability that current harmony updates randomly in the solution space is greater, which can effectively prevent SFHS from trapping into local optimum. In addition, the probability that HS algorithm converges to the global optimum is greater when the difference between best harmony and worst harmony is smaller. Let the value of *bw* be smaller, and small *bw* increases the fine-tuning of solution vectors.

3. Fuzzy c-mean Clustering (FCM)

The aim of clustering is to find groupings with the same or similar data. By minimizing objective function, a dataset $X=(x_1, x_2,\ldots, x_j,\ldots x_n)$ is clustered into c groups. The standard FCM objective function can be expressed as follow:

$$J = \sum_{i=1}^{c}\sum_{j=1}^{n} \mu_{ij}^{m} D_{ij} \qquad (7)$$

where n is number of data in dataset; c is the number of clusters; $m \geq 1$ is a weighting exponent; $\mu_{i,j}$ is the membership of the jth object in the ith cluster center; D_{ij} is a distance metric which measures the difference between the data x_j and the cluster center v_i.

$$D_{ij} = \left\| x_j - v_i \right\| \qquad (8)$$

where the data x_j is a feature vector consisting of l real-valued measurements, which could be color, texture, coordinates, *etc*.; $\| \cdot \|$ represents Euclidean norm.

The objective function is minimized when each data is close to their cluster center. By updating cluster centers and membership matrix, the objective function J will be minimized. The cluster center and the fuzzy membership matrix can be updated using Eq. (9) and Eq. (10).

$$v_i = \frac{\sum_{j=1}^{n} \mu_{ij}^m v_j}{\sum_{j=1}^{n} \mu_{ij}^m} \qquad (9)$$

$$\mu_{ij} = \left(\sum_{k=1}^{c} \left(\frac{D_{ij}}{D_{kj}} \right)^{2/(m-1)} \right)^{-1} \qquad (10)$$

The FCM algorithm proceeds by updating cluster center and membership matrix until the following termination criterion is satisfied:

$$\left\| J_{new} - J_{old} \right\| < \varepsilon \qquad (11)$$

According to the value of the membership between the data and each cluster center, the data can be assigned to a class which belongs to the cluster center with the highest membership value.

4. SFHS-based Clustering Algorithm

The FCM algorithm is one of the most widely used methods for cluster analysis, and it can be successfully applied in economy, military, astronomy, medical, and so on. However, since traditional FCM algorithm is sensitive to the initial cluster centers and can't determine the right number of clusters in data sets, optimization-based clustering algorithms have been successfully proposed to solve this problem.

4.1. Preparation Work for using SFHS to FCM

Ant-based clustering algorithms are discussed in [11] by relocating ants to cluster centers in feature space, moving objects in a 2-D space and merging them to form clusters. The proposed algorithm possesses advantage of FCM algorithm and ants algorithm. With a group of ant cooperation, the cluster centers can be found for the optimal partition.

Inspired by ant-based clustering algorithms, a novel SFHS algorithm is used in FCM clustering. The proposed algorithm includes two important stages: harmony search stage and clustering stage. The former enables the harmony vector updating adaptively in the feature space to find good cluster centers with a fixed number of iterations. The latter can utilizes the cluster centers obtained in the harmony search stage for FCM clustering.

The objective function of FCM is reformulated as the optimization criterion. The reformulated version of J is denoted as R and our SFHS-based clustering algorithm minimizes the following objective function to find a good partition.

$$R = \sum_{j=1}^{n} \left(\sum_{i=1}^{c} \left(D_{ij} \right)^{1/(1-m)} \right)^{1-m} \qquad (12)$$

The objective function R is an equivalent reformulation of J that eliminates the membership function. The membership matrix for the FCM clustering case is given as follows:

$$\mu_{ij} = \left(D_{ij}^{1/1-m} \right) / \left(\sum_{j=1}^{c} D_{ij}^{1/1-m} \right) \qquad (13)$$

where D is a distance metric that describe the difference between data vector and cluster center.

In the FCM clustering, a partition of data set can be obtained by calculating the membership matrix that each data belongs to c cluster centers. In the SFHS-based clustering algorithm, the harmony vector V_i is a $c \times s$ matrix of cluster centers where c is the number of cluster centers and s is the features number of each data. The initial decision variables values of the harmony vector in HM are randomly assigned in the range between the lower and upper bounds of each feature. The decision variables are updated in feature space, in such a way as to position the cluster centers and create a data partition. The current harmony is compared with the worst harmony in the HM, and the harmony memory will be updated that the worst harmony in the HM is replaced by the current harmony if the current harmony is better than the worst harmony in the HM. After a fixed number of iterations, the best partition will be obtained by updating the HM.

4.2. Finding the Right Number of Clusters

The traditional clustering approach is required to input the number of clusters, and the different input number of clusters leads to quite different clustering results. In ant-based clustering approach, a validity metrics is used on the clustering results created with different numbers of clusters to determine the right number of clusters. However, since the determination of the right number of clusters requires the experiments more than once, the efficiency of the algorithm is relatively low, and it is inapplicable in practice. In this paper, we propose a novel validity metric to determine the right number of clusters. The right number of clusters can be automatically calculated when the input number of clusters is greater than the right number of clusters.

A binary matrix M is defined to calculate the number of data in each class, and it can be described by the following equation.

$$M_{ij} = \begin{cases} 0 & U_{ij} < \max\left(U_{1j}, U_{2j}, \ldots, U_{cj}\right) \\ 1 & \text{otherwise} \end{cases} \tag{14}$$

where U is the membership matrix calculated by the SFHS-based clustering approach.

In the clustering results, if the number of data in the class is significantly less than the maximum number of data in all classes, the class will be merged into another class and their cluster center distance is closer than others. Repeat this process until the number of data in each class meets the following conditions

$$\sum_{k=1}^{n} M_{ik} > \alpha \times \max\left(\sum_{k=1}^{n} M_{ik}\right), i = 1, 2 \ldots c \tag{15}$$

where M is a binary matrix; α is a proportional coefficient, and α=0.15 in this paper. The class does not meet the conditions will be merged into another class. Finally, we get c_n cluster centers (c_n<=c and c_n reflects the right number of clusters in data set).

4.3. The Main Steps of SFHS-based Clustering are as Follows:

Step1: Initialize the problem parameters: Normal the feature values in ranges [0, 1].

Step2: Initialize the HS parameters: the harmony vector is a $c \times s$ matrix of cluster centers in FCM notation where s is the number of features for each data and c is the number of clusters; harmony memory size HMS = 5; lower and upper bounds of harmony memory consideration rate $HMCR_{min}$ = 0.45 and $HMCR_{max}$ = 0.99, respectively; the lower and upper bounds of the pitch adjusting bandwidth are bw_{min} = 0.003 and bw_{max} = 0.02, respectively; the maximum number of iterations T for different data sets and the initial iteration to be 1; initializing harmony memory and calculating the objective function value of the harmony in the HM.

Step3: updating the current harmony

3.1. Calculating the harmony memory considering rate and the pitch adjusting bandwidth for the current iteration.

3.2. The decision variables are randomly selected in the HM or plus $bw(t)$ to it if the randomly generated probability is less than HMCR(t).

3.3. The decision variables are randomly selected in the solution space if the randomly generated probability is more than HMCR(t).

Step4: Evaluate the objective function value of the current harmony.

4.1. The worst harmony is removed from the harmony memory and the current harmony is copied to the harmony memory if the objective function value of the current harmony is better than the worst harmony in the harmony memory. Otherwise, go to step3 to update the current harmony for a better partition.

4.2. The current iteration number t=t +1.

Step5: Repeat Step3 and Step4 until the maximum number of iteration is attained.

Step6: Find the right number of clusters.

5. Experiments and Discussion

To evaluate the performance of SFHS algorithm for cluster analysis, four algorithms (ANT, HS, IHS and SFHS algorithms) are taken into account. These algorithms are tested on remote sensing images and animal images using Matlab7.0 on an Intel IV processor (2.66GHz clock). Four performance measures including: the best, worst, mean and average time are used to compare the proposed algorithm with ANT-based clustering approach and traditional HS-based clustering approaches. In addition, a validity metrics is used to find the right number of clusters.

5.1. Performance Measures

Figure 1 and Figure 2 show two examples to compare the performance of the proposed method with the other three methods. Moreover, we conduct 50 independent runs of four algorithms, and four performance measures of the different algorithms are listed in table 1-4. The parameters of the four algorithms for Figure 1 and Figure 2 are as follows.

In Figure 1 and Figure 2, the feature number of each data l=1 and l=3, respectively. For the ANT-based clustering algorithm, the number of ants (N_a=6 for figure 1 and N_a =9 for Figure 2), the maximum number of iterations (NI=7 for Figure 1 and NI =10 for Figure 2). For the HS-based clustering algorithm, harmony memory consideration rate HMCR = 0.9, pitch adjusting rate PAR=0.45, pitch adjusting bandwidth bw=0.003, NI=120 for Figure 1 and NI=300 for Figure 2. For the IHS-based clustering algorithm, HMCR = 0.9, the maximum pitch adjusting rate PAR_{max}=0.99, the minimum pitch adjusting rate PAR_{min}=0.45, the maximum pitch adjusting bandwidth bw_{max}=0.02, the minimum pitch adjusting bandwidth bw_{min}=0.003, NI=120 for figure 1 and NI=300 for Figure 2. For the SFHS-based clustering algorithm, the maximum memory consideration rate $HMCR_{max}$ = 0.99, the minimum memory consideration rate $HMCR_{min}$ = 0.49, PAR = 0.45, bw_{max}=0.02, bw_{min}=0.003, NI=120 for Figure 1 and NI=300 for Figure 2.

Tables 1-4 compare the performance measures obtained by the SFHS-based clustering algorithm for Figure 1 and Figure 2 with those of three other algorithms (the ANT-based clustering algorithm, the IHS-based clustering algorithm and the HS-based clustering algorithm) in this paper. The best、worst and mean results obtained by the ANT-based clustering are all very close to each other in each case, which means that the ANT-based clustering algorithm is easy to trap into the local optimum. Since the traditional HS algorithms are global optimization search algorithms, they can effectively prevent the HS from trapping into the local optimum. We can see from the Tables 1-4 that the results obtained by the HS-based and HIS-based clustering algorithms are better than the ANT-based clustering algorithm. In this paper, the state-feedback mechanism is introduced into HS to adaptively adjust harmony memory consideration rate (HMCR) and pitch adjusting bandwidth (bw), which makes the proposed algorithm can converge to the global optimum quickly. As show in tables 1-4, the performance of the proposed method is most outstanding compared with the three other methods.

(a) (b) (c) (d)

Figure 1. Remote Sensing Images

Table 1. The experiment Results for Figure 1 (*c*=4)

Image	Algorithm	Best (10^3)	Worst (10^3)	Mean (10^3)	Time (*s*)
a	ANT	2.2305	2.2383	2.2347	7.9940
	HS	1.5028	1.6529	1.5552	5.0818
	IHS	1.4997	1.5795	1.5233	5.0538
	SFHS	1.4993	1.5864	1.5183	4.9824
b	ANT	2.5275	2.5362	2.5329	8.0661
	HS	1.7862	1.9116	1.8439	5.0857
	IHS	1.7796	1.8328	1.7949	5.0201
	SFHS	1.7757	1.8368	1.7945	4.9692
c	ANT	3.8948	3.9069	3.9031	8.0722
	HS	2.3276	2.4600	2.3708	5.0727
	IHS	2.3159	2.4090	2.3397	5.0588
	SFHS	2.3150	2.4060	2.3344	4.9879
d	ANT	3.0208	3.0322	3.0276	8.0016
	HS	1.8205	1.9317	1.8706	5.0639
	IHS	1.8154	1.9164	1.8421	5.0669
	SFHS	1.8170	1.9039	1.8384	5.0149

Table 2. The Experiment Results for Figure1 (*c*=5)

Image	Algorithm	Best (10^3)	Worst (10^3)	Mean (10^3)	Time (*s*)
a	ANT	1.7800	1.7885	1.7853	8.2109
	HS	1.1494	1.2465	1.1773	6.3387
	IHS	1.1438	1.1903	1.1660	6.3727
	SFHS	1.1430	1.1976	1.1642	6.2931
b	ANT	2.0183	2.0277	2.0240	8.2139
	HS	1.3674	1.4531	1.3957	6.3593
	IHS	1.3634	1.4271	1.3852	6.3640
	SFHS	1.3635	1.4233	1.3851	6.2453
c	ANT	3.1196	3.1242	3.1220	8.2518
	HS	1.7424	1.8290	1.7847	6.3158
	IHS	1.7381	1.8231	1.7606	6.3437
	SFHS	1.7371	1.7742	1.7503	6.2550
d	ANT	2.4138	2.4262	2.4192	8.1892
	HS	1.3931	1.5107	1.4431	6.3445
	IHS	1.3814	1.5249	1.4129	6.3259
	SFHS	1.3799	1.4943	1.4004	6.2584

(a) (b) (c) (d)

Figure 2. Animal Images

Table 3. The Experiment Results for Figure 2 (*c*=4)

Image	Algorithm	Best (10^3)	Worst (10^3)	Mean (10^3)	Time (*s*)
a	ANT	1.2384	1.2680	1.2544	4.0262
	HS	0.8862	1.2474	1.0499	2.6749
	IHS	0.8227	1.2842	0.9294	2.6555
	SFHS	0.7948	1.0623	0.8885	2.5945
b	ANT	1.8003	1.8062	1.8039	4.0019
	HS	1.1838	1.6378	1.4047	2.6824
	IHS	1.0104	1.3913	1.1707	2.6372
	SFHS	0.9820	1.4556	1.1337	2.6352
c	ANT	1.7427	1.7720	1.7625	4.0078
	HS	1.0326	1.3608	1.2000	2.6885
	IHS	0.9331	1.2224	1.0694	2.6162
	SFHS	0.9293	1.2189	1.0206	2.6157
d	ANT	1.4718	1.4767	1.4754	4.0029
	HS	1.0096	1.3598	1.1671	2.6519
	IHS	0.9915	1.2316	1.0902	2.6410
	SFHS	0.9719	1.2302	1.0591	2.6226

Table 4. The Experiment Results for Figure 2 (*c*=5)

Image	Algorithm	Best (10^3)	Worst (10^3)	Mean (10^3)	Time (*s*)
a	ANT	0.9900	1.0160	1.0027	4.1822
	HS	0.7411	1.0370	0.8432	3.8159
	IHS	0.6736	0.8689	0.7387	3.7248
	SFHS	0.6485	0.8115	0.7008	3.7160
b	ANT	1.4395	1.4452	1.4427	4.1792
	HS	0.8640	1.2588	1.0957	3.7396
	IHS	0.8051	1.1400	0.9660	3.7709
	SFHS	0.8255	1.1053	0.9130	3.7025
c	ANT	1.3973	1.4178	1.4088	4.2127
	HS	0.8439	1.0587	0.9492	3.7185
	IHS	0.7057	1.0886	0.8438	3.6798
	SFHS	0.7196	0.9350	0.8173	3.6701
d	ANT	1.1776	1.1814	1.1803	4.1886
	HS	0.8403	1.0558	0.9484	3.7962
	IHS	0.7967	0.9427	0.8514	3.7119
	SFHS	0.7416	0.9186	0.8303	3.6703

5.2. Finding the Number of Clusters

Clustering with optimization approach requires knowledge of the number of clusters. In our proposed algorithm, a validity metrics is proposed and used to determine the right number of clusters. In clustering results, the class with few data is merged into another class and their cluster center distance is closer than others. Experiments are done with two images as shown in Figure 3. The right number of clusters can be obtained when the input number of clusters is greater than the right number of clusters. For instance, as shown in Table 5, the right number of clusters (c_n=2,2,2 for Figure 3(a) and c_n=4,4,4 for Figure 3(b)) is automatically calculated when we input the different number of clusters (c=4,5,6 for Figure 3(a) and c=6,7,8 for Figure 3(b)).

(a) (b)

Figure 3. Images

Table 5. The Right Number of Clusters

image	c	c_n
3(a)	4	2
	5	2
	6	2
3(b)	6	4
	7	4
	8	4

6. Conclusion

This paper presents a state-feedback based harmony search (SFHS) algorithm for optimizing the FCM objective function. The proposed SFHS algorithm introduces the state-feedback mechanism to improve its convergence and efficiency. In addition, a novel validity metric is presented to determine the right number of clusters. The optimization performance of SFHS algorithm on cluster analysis has been investigated by several experimental studies. The experimental results illustrate that SFHS algorithm is more efficient in finding the best solutions than the existing ANT, IHS and HS algorithms, and SFHS algorithm is superior to other algorithms in literatures when it is used in clustering data. In the future, we will go on revising and updating this algorithm, and SFHS algorithm can be used to medical image processing, video image processing and crime scene image processing for actual applications.

Acknowledgments

This work is supported by National Natural Science Foundation of PR China under Grant 71231001 and PhD research Startup Foundation of National Police University of China. We also gratefully acknowledge the helpful comments and suggestions of the anonymous referees.

References

[1] K. Rasha and M. S. Kamel, “Cooperative clustering”, Pattern Recognition, vol. 43, no. 6, **(2010)**, pp. 2315-2329.
[2] L. Wang, C. Leckie, R. Kotagiri and J. Bezdek, “Approximate pairwise clustering for large data sets via sampling plus extension”, Pattern Recognition, vol. 44, no. 2, **(2011)**, pp. 222-235.
[3] R. J. Kuo and L. M. Lin, “Application of a hybrid of genetic algorithm and particle swarm optimization algorithm for order clustering”, Decision Support Systems, vol. 49, no. 4, **(2010)**, pp. 451-462.
[4] A. Y. Al-Omary and M. S. Jamil, “A new approach of clustering based machine-learning algorithm”, Knowledge-Based Systems, vol. 19, no. 4, **(2006)**, pp. 248-258.

[5] A. M. Massone, L. Studer and F. Masulli, "Possibilistic clustering approach to trackless ring Pattern Recognition in RICH counters", International Journal of Approximate Reasoning, vol. 41, no. 2, **(2006)**, pp. 96-109.
[6] X. Y. Wang and J. Bu, "A fast and robust image segmentation using FCM with spatial information", Digital Signal Processing, vol. 20, no. 4, **(2010)**, pp. 1173-1182.
[7] C. C. Hung, S. Kulkarni and B. C. Kuo, "A New Weighted Fuzzy C-Means Clustering Algorithm for Remotely Sensed Image Classification", IEEE Journal of Selected Topics in Signal Processing, vol. 5, no. 3, **(2011)**, pp. 543-553.
[8] A. Tombros, R. Villa and C. J. Van Rijsbergen, "The effectiveness of query-specific hierarchic clustering in information retrieval", Information Processing & Management, vol. 38, no. 4, **(2002)**, pp.559-584.
[9] K. S. Lee, K. Kageura and K. S. Choi, "Implicit ambiguity resolution using incremental clustering in cross-language information retrieval", Information Processing & Management, vol. 41, no. 4, **(2004)**, pp. 145-159.
[10] L. O. Hall, I. B. Ozyurt and J. C. Bezdek, "Clustering with a genetically optimized approach", IEEE Transactions on Evolutionary Computation, vol. 3, no. 2, **(1999)**, pp. 103-112.
[11] P. M. Kanade, L. O. Hall, "Fuzzy Ants and Clustering", IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 37, no. 5, **(2007)**, pp. 758-769.
[12] Z. Geem, J. Kim and G. Loganathan, "A new heuristic optimization algorithm: harmony search", Simulation, vol. 76, no. 2, **(2001)**, pp. 60-68.
[13] G. H. O. Mahamed and M. Mehrdad, "Global-best harmony search. Applied Mathematics and Computation", vol. 198, no. 2, **(2008)**, pp. 634-656.
[14] Q. K. Pan, P. N. Suganthan and M. F. Tasgetiren, "A self-adaptive global best harmony search algorithm for continuous optimization problems", Applied Mathematics and Computation, vol. 216, no. 3, **(2010)**, pp. 830-848.
[15] M. Mahdavi, M. Fesanghary and E. Damangir, "An improved harmony search algorithm for solving optimization problems", Applied Mathematics and Computation, vol. 188, no. 2, **(2007)**, pp. 1567-1579.

Authors

Lei Wang, He received his Ph.D. degree from University of Science and Technology Beijing. He is currently a assistant professor in Department of Public Order, National Police University of China. He is primary research interests in Intelligent Algorithm, Imagine Segmentation and Information System.

Yufeng Yi, He received his Ph.D. degree from Northeastern University. He is currently a engineer in Northeast Institute of Electronic Technology, China. He is primary research interests in Intelligent Algorithm and Imagine Segmentation.

Jiao Liu, She received her Ph.D. degree from Dalian Maritime University. She is currently a associate professor in Department of Public Order, National Police University of China. She is primary research interests in Emergency Management, Emergency Information System.

