2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

A Hybrid EA for High-dimensional Subspace Clustering Problem

Lin Lin
School of Software Technology
Dalian University of Technology
Dalian, China
Fuzzy Logic Systems Institute
Fukuoka, Japan
lin@dlut.edu.cn

Mitsuo Gen
Tokyo University of Science
Tokyo, Japan
Fuzzy Logic Systems Institute
Fukuoka, Japan
mitsuogen@gmail.com

Yan Liang
School of Software Technology
Dalian University of Technology
Dalian, China

Abstract—**Considering Particle Swarm Optimization (PSO) could enhance solutions generated during the evolution process by exploiting their social knowledge and individual memory, we used PSO as a local search strategy in Genetic Algorithm (GA) framework for fine tuning the search space. GA is to make sure that every region of the search space is covered so that we have a reliable estimate of the global optimal solution and PSO is for further pruning the good solutions by searching around the neighborhood. In this paper, proposed approach is used for subspace clustering, which is an extension of traditional clustering that seeks to find clustering in different subspaces within a dataset. Subspace clustering is to find a subset of dimensions on which to improve cluster quality by removing irrelevant and redundant dimensions in high dimensions problems. The experimental results demonstrate the positive effects of PSO as a local optimizer.**

Keywords—particle swarm optimization; hybrid evolutionary algorithm; high-dimensional subspace clustering

I. INTRODUCTION

High dimension clustering is one of important thesis in data mining and it has been widely used in marketing analysis, information security, economic, medicine and engineering. Current technology can solve low-dimensional data clustering problems, however when considering high-dimensional clustering problems, the distributions of data have big difference with low-dimensional ones. So at this circumstance, many algorithms failed to give good solutions. The main reasons are: firstly, always happening on the sparse distribution data sets, the relative distance between data points is going to be zero since the number of dimensions is increasing; secondly, in the data space, there are some irrelevant attributes belying the clusters which are what we are finding; thirdly, the time cost grows exponentially as the dimension increases. So it has real meaning to come up a valid high dimension clustering method.

The main search methods include subspace clustering and dimension reduction. Subspace clustering is to extract the clusters from subspace. The classical algorithms in this category are PROCLUS, DOC, CLIQUE and etc [1]. Dimension reduction usually means to use attributes choosing or attributes changing to get relative low-dimensional data set from previous high-dimensional one, then to use traditional ways to complete the clustering.

While solving the high-dimensional problems, there are special difficulties from three aspects. (1) The problem is high-dimensional problem, the time cost grows exponentially as the dimension increases. (2) Which attribute sets are effective attributes to compose subspace clusters is unknown in such big data sets. (3) How many attributes should be chose is unknown.

In this paper, we firstly use GA as a global search way to get subspace clustering and then use PSO as a local search optimizer to prune the search space. Global reliability and local pruning is two competing goals governing the design of global optimizing provided by Torn and Zilinskas [2]. Their goals are to cover as much as possible huge search space to give a reliability of global optimum and to improve the good solutions by searching around the space near the solutions got from the global search. Many researchers add local search strategy to global search method to adopting a combination of two goals and they achieved good results.

The remainder of this paper is organized as follows. Section 2 gives relative work about GA and PSO. In Section 3, we describe the GA framework with PSO as a local search strategy. Numerical analysis of case study shows the effectiveness of proposed approaches in Section 4. Finally, in Section 5, we give the conclusion.

II. RELATIVE WORK

A. Genetic Algorithm and Clustering

Genetic Algorithm is a self-adapted global search algorithm by simulating the process of heredity and evolution in the nature environment and it is widely used in resolving complex optimizing problems [3]. It is based on basic concepts from the evolution biological model. It starts from some initial population representing possible potential solution set. Every possible solution is called individual which comes from decoding of possible potential solutions. After initial

978-1-4799-1488-3/14/$31.00 ©2014 IEEE

population is generated, according to survival of the fittest, in each generation, choose the individuals according to the fitness depending on different problems and then have crossover and mutation process. At last we get the solution having a best fitness value as the optimal solution in the optimizing problems. The strategy of GA lies in their population-based search strategy that will generate higher diversity in the search space, also reducing the likelihood to converge to local optimum.

As a valid global search method, GA has been used in clustering problem by many researchers. Maulik [4] proposed G-clustering, which is using GA as a global search method to optimizing the cluster centers to increase the accurate. But this method is used in all dimensions clustering, so it cannot be used in high dimension problems. In this paper, we will use evolutionary algorithm to deal with high-dimension problem and we will use subspace attributes to do clustering analysis.

B. Particle Swarm Optimization as LS Strategy

Particle swarm optimization (PSO) is a well-known evolutionary computation model developed by Kennedy and Eberhart [5]. PSO mimics the behavior of a swarm of a flock of birds or fishes. A swarm of particles is moving within the search space in PSO [6]. Each particle has a vector of velocity V_i, a vector of position x_i, the position P_{gbest}, which is the best one among all the particles in the population and the position P_{pbest} which is the best previous one of the each particle. During the evolution process of the PSO, position of each particle and the velocity are updated towards to P_{gbest} and its P_{pbest} based on the following equations.

$$v_i(k+1)=\omega\, v_i(k)+c_1\mathrm{rand}_1[p_{pbest}(k)-x_i(k)] + c_2\mathrm{rand}_2[p_{gbest}(k)-x_i(k)] \tag{1}$$

$$x_i(k+1)=x_i(k)+v_i(k+1) \tag{2}$$

The parameter $\omega\in[0,1]$ is called the inertia weight that is to determine the influence degree of the previous velocity. Acceleration coefficients are positive constants c_1 and c_2 and rand_1, rand_2 are random value in [0, 1].

In contrary to that GA performs according to evolution from generation to generation by dealing with solutions from different generations obviously. PSO performs based on social adaptation of information by allowing each particle accompanying the knowledge of good solutions, such as, previous best position and the global best solution which would retain throughout the whole evolution. This is a constructive and valid cooperation between particles to get a better solution.

Vesterstrøm and Thomsen [7] showed that the reliability of global convergence of GA with enough generations and the fast convergence of PSO can be hybridized. Farsangi, Nezamabadi-Pour, and Lee [8] used some parameters and problems setting, which will improve their algorithm performance. Many different GA-PSO hybrids were proposed in papers, such as Grosan et al exploited the fast convergence of PSO and global reliability of GA [9], to form a two-step implementation of GA and PSO to solve geometrical place problems. It was also used in the optimization of Profiled Corrugated Horn antenna [10]. Chiam et al use GA-PSO to solve finance applications [11].

III. GA-PSO CLUSTERING

A. Framework of Proposed GA-PSO

The proposed GA-PSO Clustering has two phases: the global search strategy GA and the local optimizer PSO. And it can be summarized as following steps as shown in Fig. 1:

```
procedure: GA-PSO clustering
begin
  input problem data, GA-PSO parameters;
  output the optimal solutions;
    step 1: t =0;
    step 2: get grid based on the density of the data;
    step 3: initialize population P(t) randomly from the high density
            data points;
    step 4: clustering the data points based on relative distance and
            calculating the fitness according to the relative distance of
            all data points;
    step 5: t=t+1;
    step 6: if meet the terminating condition, then go to step 10;
    step 7: get 10% good individuals and select left individuals to
            compose the next generation population P(t) by roulette
            selection routine;
    step 8: apply mutation operation on P(t);
    step 9: goto step 4;
    step 10: t =0;
    step 11: get the 30% solutions as the PSO population Q(t);
    step 12: t=t+1;
    step 13: if meet the terminal condition, then goto step 15;
    step 14: PSO evolution process;
    step 15: output the optimal solutions;
end;
```

Fig. 1. The procedure of GA-PSO clustering

The parts of how to form density-based grid, encoding method, mutation operation, selection operation and PSO process and will be detailed explanation in the next parts of this section.

B. Preprocess of Density-based Grid

In the first step, a standard clustering algorithm is applied to each attribute alone to construct a grid. So for a data set, the clustering algorithm should be called once per attribute. The cluster center will be composed by different grid and the number of centroids is varied as a function of the data distribution. There are two normal algorithms always being used in this situation, more accurate but slower EM [12] proposed by Dempster, Laird, & Rubin and faster but less accurate X-means [13] proposed by Pelleg & Andrew. We can use any of them. Fig.2. shows the result of preprocess of density based grid.

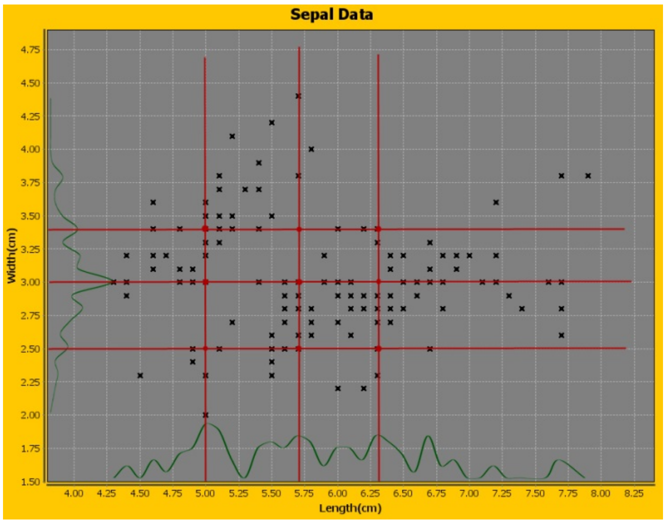
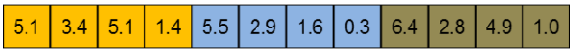


Fig. 2. An example of density based-grid

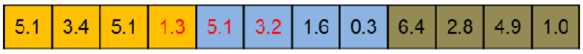
C. *Representation for Clustering Solutions*

In this paper, we directly use the composition of coordinates of the clustering centroids as a chromosome since the latter PSO can also share such encoding method. It is also concise to the clustering problems and does not need any change process, so the coordinate is just the gene of the encoding chromosome. When the clustering centroids are determined, all the data points can be clustered to the relative centroids according to the relative distances. Fig.2 shows a chromosome of a solution that the problem has four attributes and there are three clusters in this solution. In Fig.3, each color represents one clustering centroid coordinate.



5.1	3.4	5.1	1.4	5.5	2.9	1.6	0.3	6.4	2.8	4.9	1.0

Fig. 3. Representation of a clustering solution



5.1	3.4	5.1	1.3	5.1	3.2	1.6	0.3	6.4	2.8	4.9	1.0

Fig. 4. A mutation example

D. *Mutation Operation*

Variation operations are the basic genetic material diversifying parts within an evolutionary method. They make guided or random changes to individuals to exploit or explore the search space for optimal solution. Mutation is one of these, and it is to simulate the mutation operation to change some gene in the chromosome in nature. It tries to make a random change to a parent to explore a different area of the search space. We applied simple mutation in this algorithm. We choose a gene from the chromosome randomly and check if it can meet the mutation possibility. If so, we transfer the coordinate of the certain attribute to another one. The new one also comes from the density-based grid processed by the reprocess. Fig.3 shows that the third, fourth and fifth genes in the Fig. 4 mutate to new ones.

E. *Selection Operation*

Selection operation is the evolutionary operation of GA, which provides the driving force to let the population evolve. When the force is too much, the genetic search will be slower instead of necessary. Normally, we should use a lower selection pressure at the whole GA process to have a wide exploration of the search space, while in the latter PSO process, the genetic search toward promising regions in the search space.

In this paper, we directly choose 10% elitist individuals to be next generation according to the objective function. Then we use Fitness Proportionate Championships to choose from the left individuals.

Before evaluating the fitness of a cluster solution, we should complete the assignment of data points based on nearest neighbor assignment. As in self-organizing maps [14], average link agglomerative clustering [15] and k-means [16], our object is to form compact spherical clusters by minimizing the distance between cluster centroids and data points. Objective function is the direct basis of search of GA so the objective function would effect on the search direction and the degree of convergence. In high-dimensional clustering problem, the objective clusters has relative with a few but not all the attributes. In order to inspect the characters of the feature attribute in subspace clustering problems, we considering the distances normalized over the number of the feature attribute.

Assuming that there are j feature attributes in a certain subspace and k subspace solutions $\{ss_1,ss_2,ss_3,\dots ss_k\}$, which m clusters are $\{c_1,c_2,c_3,\dots,c_m\}$ and m clustering centroids are $\{cc_1,cc_2,cc_3,\dots,cc_m\}$, the fitness function is as follows:

$$\mathrm{Com}(x)=\Sigma_{x\in ss1}\mathrm{DIS}(ss_i) \tag{3}$$

$$\mathrm{DIS}(ss)=\Sigma_{x\in c1}\mathrm{dis}(c_j) \tag{4}$$

$$\mathrm{dis}(c)=\Sigma_{x\in cc1}\Sigma_{y\in j}\mathrm{dis}(x^j, p_k^j) \tag{5}$$

To each cluster c_i, we have the function (4) and to each cluster centroids we have function (5). P_k is all the data points belonging the ith cluster and *dis* is the sum of distance between data points and the cluster centroids in a cluster.

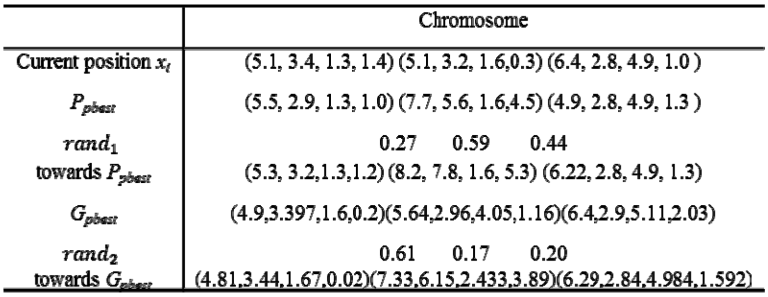
F. *Particle Position Update of PSO*

After the running GA as a global search method, we will directly choose some elitist individuals as the initial population for the local search strategy, PSO. That is one of the advantages coordinate encoding method. Such a local search method can prune the search space validly.

The position of each particle is updated towards to its personal best position P_{pbest} and the global best position G_{pbest}. The way adopted for updating the velocities and positions of each particle in the proposed PSO is based on the standard PSO introduced in Section 2. The procedure is as follows:

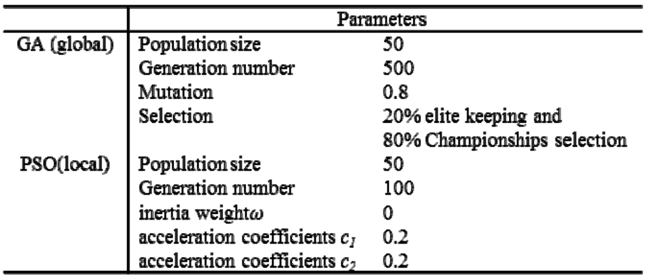
Advance x_i towards P_{pbest} with velocity V_1=c_1,
Advance x_i towards P_{gbest} with velocity V_2=c_2.

TABLE I. POSITION UPDATING FOR ONE PARTICLE



	Chromosome
Current position x_i	(5.1, 3.4, 1.3, 1.4) (5.1, 3.2, 1.6,0.3) (6.4, 2.8, 4.9, 1.0)
P_{pbest}	(5.5, 2.9, 1.3, 1.0) (7.7, 5.6, 1.6,4.5) (4.9, 2.8, 4.9, 1.3)
$rand_1$	0.27 0.59 0.44
towards P_{pbest}	(5.3, 3.2,1.3,1.2) (8.2, 7.8, 1.6, 5.3) (6.22, 2.8, 4.9, 1.3)
G_{pbest}	(4.9,3.397,1.6,0.2)(5.64,2.96,4.05,1.16)(6.4,2.9,5.11,2.03)
$rand_2$	0.61 0.17 0.20
towards G_{pbest}	(4.81,3.44,1.67,0.02)(7.33,6.15,2.433,3.89)(6.29,2.84,4.984,1.592)

TABLE II. ALGORITHM PARAMETER SETTINGS OF GA-PSO FOR THE EXPERIMENTAL STUDY



	Parameters	
GA (global)	Population size	50
	Generation number	500
	Mutation	0.8
	Selection	20% elite keeping and 80% Championships selection
PSO(local)	Population size	50
	Generation number	100
	inertia weightω	0
	acceleration coefficients c_1	0.2
	acceleration coefficients c_2	0.2

In this paper, PSO is worked as a local optimizer and its most important issue is to prune the search space validly. So we set parameter ω concerned with global search ability to 0 and the two parameters, c_1 and c_2 to apropos constants to enhance the local search ability. But we should notice that, for different data set, different value of c_1 and c_2 would influence the search result.

During the position updating referred to its personal best position, the difference between the current position x_i and personal best position P_{pbest} is found by comparing their each coordinate. Secondly, we get a random value as rand$_1$, which is to decide the degree of the position updating with the acceleration coefficients $c_{1.}$ And the position updating towards global best position G_{pbest} is same except that the relative points should be decided before Close Process.

IV. CASE STUDY

In this part we used the proposed algorithm to solve real data problems, we choose two data sets. All the experiments were performed on a Pentium (R) Dual-Core CPU 3.00 GHz 3.00 GHZ with 2.00 GB RAM. We have coded with Java. All data set were run with GA-PSO proposed in this paper and ESC. ESC is another evolutionary algorithm focusing on subspace clustering analysis, but it doesn't have a local optimizer strategy. The results of experiments are evaluated and compared by the error rate. In this paper, we run two well-known data sets. Both of them are from

http://archive.ics.uci.edu/ml/.

A. Irish Data Set

Iris data set contains 3 classes of 50 instances each of the classes are Iris Setosa, Iris Versicolour and Iris Virginica respectively. There are four attributes in the Iris data set, sepal length, sepal width, petal length, and petal width.

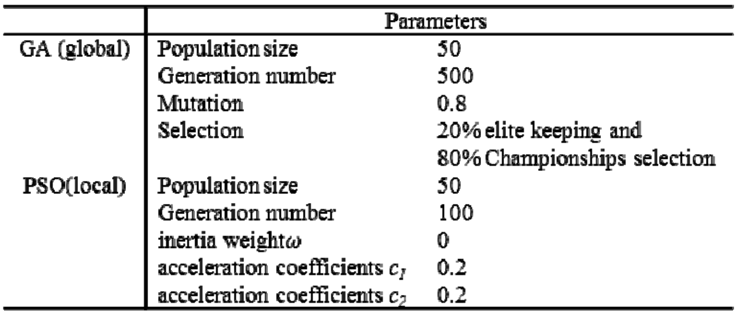
For this data set, the parameters we used are listed in Table II. We enhance the global research ability of GA by setting relatively big mutation probability. Fig. 5 and Fig. 6 show the convergence of solutions of GA-PSO and ESC respectively. In Fig. 5 we can know that after the global searching, we got some solutions which have relative good fitness value. But when go on running the algorithm after the generation is over 120, it doesn't change obviously so it cannot find better solutions. At this circumstance, we apply PSO and its effectiveness shown in Fig. 5. It can prune the search space as a local search optimizer.

Table IV shows the error rate of Iris data set with proposed GA-PSO and ESC. Error rate is the ratio of the number of wrong data points and the number of total data points in certain cluster, which is one valid indicator to show the effectiveness of algorithms. And it next part of the test case, we also use the error ratio to compare the proposed GA-PSO and ESC. Fig. 4 shows that for the average error rate, GA-PSO is better than ESC. But for the best solution, the advantage of GA-PSO is not so remarkable.

B. Wine Data Set

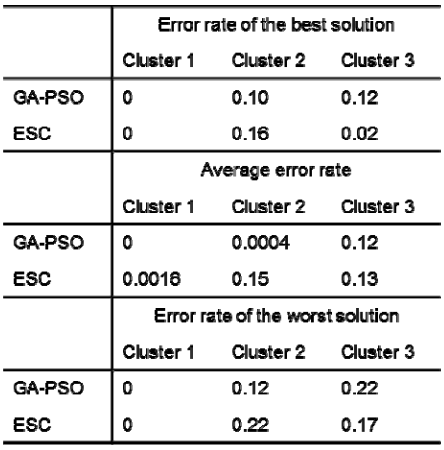
Wine data set is also a famous dataset which is the result of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. Wine data set has 13 attributes and contains 187 instances.

TABLE III. ALGORITHM PARAMETER SETTINGS OF GA-PSO FOR THE EXPERIMENTAL STUDY



	Parameters	
GA (global)	Population size	50
	Generation number	500
	Mutation	0.8
	Selection	20% elite keeping and 80% Championships selection
PSO(local)	Population size	50
	Generation number	100
	inertia weightω	0
	acceleration coefficients c_1	0.2
	acceleration coefficients c_2	0.2

TABLE IV. THE ERROR RATE OF IRIS DATA SET WITH PROPOSED GA-PSO AND ESC



	Error rate of the best solution		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0	0.10	0.12
ESC	0	0.16	0.02
	Average error rate		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0	0.0004	0.12
ESC	0.0016	0.15	0.13
	Error rate of the worst solution		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0	0.12	0.22
ESC	0	0.22	0.17

cluster number = 3; running times = 50;

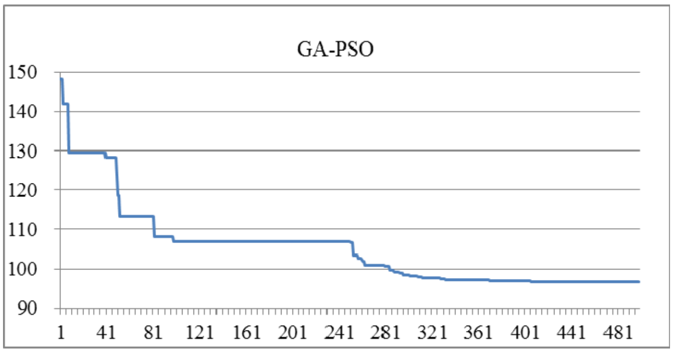


Fig. 5. The solution convergence of GA-PSO on Iris data set

Table III shows the algorithm parameter settings of GA-PSO for the experimental study. For wine data set, we consider three situation of different attribute number, and they are 5, 7 and 13 respectively. Tables V-VII shows the error rate of Wine data set with proposed GA-PSO and ESC, with different attribute number respectively. It shows that the error rate declines when the number of attributes are considers. For this problem, all the attributes are not irrelative. But when we just consider 7 attributes to get clustering analysis, it has already got a relative low error rate for the average value. We can also know well that, no matter how many of the attribute numbers is, the error of GA-PSO is lower than that of ESC of both best solution and average of all solutions.

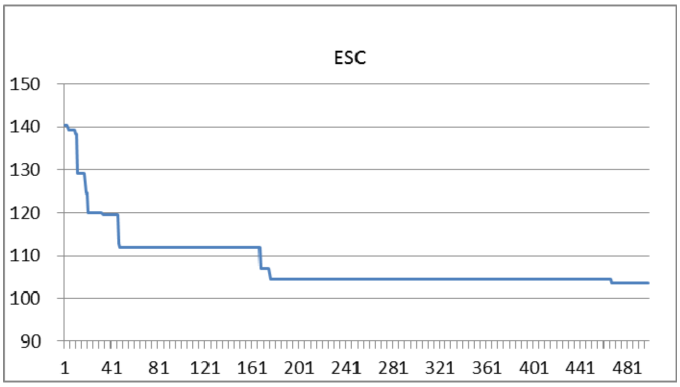


Fig. 6. The solution convergence of ESC on Iris data set

V. CONCLUSION

In this paper, we proposed a hybrid evolutionary algorithm GA-PSO, which GA is for global search and PSO is as a local search optimizing strategy. According to the setting of the parameters in the GA and PSO, we would enhance the global search ability and the search space pruning ability. Since GA is a stochastic search algorithm, and its genetic operation, such as, mutation operation and selection operation can expand the search space, so GA can handle global search. However, sometimes GA would have premature convergence problems. So in this situation, it cannot get optimal solutions. To resolve this problem, we add a PSO as local search strategy. We use some real data sets of clustering analysis to test the proposed algorithm and we also compared our algorithm with another evolutionary algorithm to prove its effectiveness.

Before evolution algorithm begins, we apply a density-based method to get grid. Such gird composed the resource pool of cluster centroids. The reason of using this method instead of a stochastic way to initialize the population of GA is that higher density points have higher possibility to be the cluster centroids. In GA, we used direct coordinate encoding method, it is concise to the clustering problems and don’t need any change process, the coordinate is the gene of the encoding chromosome. Moreover the latter PSO can also share the result

TABLE V. THE ERROR RATE OF WINE DATA SET WITH PROPOSED GA-PSO AND ESC (ATT. # = 5)

	Error rate of the best solution		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0.09	0.17	0.08
ESC	0.11	0.25	0.09
	Average error rate		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0.13	0.41	0.10
ESC	0.18	0.59	0.12
	Error rate of the worst solution		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0.27	0.59	0.16
ESC	0.30	0.67	0.17

cluster number = 3; running times = 50; attribute number = 5.

TABLE VI. THE ERROR RATE OF WINE DATA SET WITH PROPOSED GA-PSO AND ESC (ATT. # = 7)

	Error rate of the best solution		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0.03	0.07	0.02
ESC	0.07	0.09	0.04
	Average error rate		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0.07	0.14	0.03
ESC	0.13	0.17	0.08
	Error rate of the worst solution		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0.15	0.26	0.05
ESC	0.25	0.29	0.15

cluster number = 3; running times = 50; attribute number = 7

TABLE VII. THE ERROR RATE OF WINE DATA SET WITH PROPOSED GA-PSO AND ESC (ATT. # = 13)

	Error rate of the best solution		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0	0.05	0
ESC	0	0.08	0
	Average error rate		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0.04	0.10	0.00
ESC	0.10	0.12	0.01
	Error rate of the worst solution		
	Cluster 1	Cluster 2	Cluster 3
GA-PSO	0.17	0.19	0.02
ESC	0.23	0.18	0.04

cluster number = 3; running times = 50; attribute number = 13

of such encoding method. We apply a mutation operation in GA, which is to try to get a random change to a parent to explore different areas of the search space. The fitness function in this algorithm is the compactness of data sets which has been used in many clustering algorithms. Concerned with the PSO, we set the inertia weight ω=0, to stop the global search, and set the acceleration coefficients c1=c2=0.2, to enhance the local search ability. However, the fitness function needs to be improved since it doesn't have positive correlation with the error rate all the time. In further research, it would be our focus.

ACKNOWLEDGMENT

This work is partly supported by New Teacher Fund of Ministry of Education of China (no. 20120041120053) and supported by Japan Society of Promotion of Science (JSPS): Grant-in-Aid for Scientific Research (C) (No. 24510219.0001).

REFERENCES

[1] Parsons L., Haque E. & Liu H.: Subspace Clustering for High Dimensional Data: A Review. SIGKDD Explorations, 2004, 6(1): 90-105.

[2] Gen M., Cheng R. & Lin L.: Network Models and Optimization: Multiobjective Genetic Algorithm Approach, Springer, 2008.

[3] Torn A. & Zilinskas A.: Global optimization, Lecture Notes in Computer Science 350, Springer, 1989.

[4] Maulik U. & Bandyopadh yay S.: Genetic Algorithm- Based Clustering, Technique Pattern Recognition, 2000, 33(9): 455-1465.

[5] Kennedy J. & Eberhart R., Particle swarm optimization, In Proceedings of IEEE int. conf. on neural networks, 1995, 4: 1942–1948.

[6] Yu X.J. & Gen M.: Introduction to Evolutional Algorithms, Springer, 2010.

[7] Vesterstrøm J. & Thomsen R.: A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems, In Proceedings of the congress on evolutionary computation, 2004, 2: 1980–1987.

[8] Farsangi M.M., Nezamabadi-Pour H. & Lee K.Y.: Multi-objective VAr planning with SVC for a large power system using PSO and GA, In Proceedings of the IEEE Power Systems Conference and Exposition, 2006, 274–279.

[9] Grosan C., Abraham A. & Nicoara M.: Performance tuning of evolutionary algorithms using particle sub swarms, In Proceedings of the 7th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, 2005, 25-29.

[10] Ahmat-Samii Y.: Genetic algorithm (GA) and particle swarm optimization (PSO) in engineering electromagnetics, In Proceedings of the 17th international conference on Applied Electromagnetics and Communications, 2003, 1-5.

[11] Chiam S.C., Tan K.C. & Mamun A.: A mimetic model of evolutionary PSO for computational finance applications, Expert Systems with Applications, 2009, 36(2): 3695–3711.

[12] Dempster A. P., Laird N. M. & Rubin D. B.: Maximum Likelihood from Incomplete Data via the EM Algorithm, Journal of the Royal Statistical Society, Series B (Methodological), 1977, 39(1): 1-38.

[13] Pelleg D. & Andrew M.: X-means: Extending K-means with Efficient Estimation of the Number of Clusters. ICML-2000 (2000).

[14] Vorhees E.: The effectiveness and efficiency of agglomerative hierarchical clustering in document retrieval. Ph.D. dissertation, Department of Computer Science, Cornell University, 1986.

[15] Forgy E.: Cluster analysis of multivariate data: efficiency vs. interpretability of classifications. Biometrics, 1965, 21: 768-780.

[16] MacQueen J. B.: Some Methods for classification and Analysis of Multivariate Observations, In Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967: 281–297