Proceedings of the 2004 IEEE
International Conference on Networking, Sensing & Control
Taipei, Taiwan, March 21-23, 2004

Particle Swarm Optimization Algorithm and Its Application to Clustering Analysis

Ching-Yi Chen
Department of Electrical Engineering,
Tamkang University,
Tamsui, Taipei Hsien, Taiwan, ROC
ccy@ee.tku.edu.tw

Fun Ye
Department of Electrical Engineering,
Tamkang University,
Tamsui, Taipei Hsien, Taiwan, ROC
fyee@mail.tku.edu.tw

Abstract - *Clustering analysis is applied generally to Pattern Recognition, Color Quantization and Image Classification. It can help the user to distinguish the structure of data and simplify the complexity of data from mass information. The user can understand the implied information behind extracting these data. In real case, the distribution of information can be any size and shape. A particle swarm optimization algorithm-based technique, called PSO-clustering, is proposed in this article. We adopt the particle swarm optimization to search the cluster center in the arbitrary data set automatically. PSO can search the best solution from the probability option of the Social-only model and Cognition-only model[1,2,3]. This method is quite simple and valid, and it can avoid the minimum local value. Finally, the effectiveness of the PSO-clustering is demonstrated on four artificial data sets.*

Keywords: Clustering analysis, PSO.

1 Introduction

Cluster analysis has become an important technique in exploratory data analysis, pattern recognition, machine learning, neural computing, and other engineering. The clustering aims at identifying and extracting significant groups in underlying data. In the field of clustering, K-means algorithm is a very popular algorithm[4]. It is used for clustering where clusters are of crisp and spherical. Here, clustering is based on minimization of overall sum of the squared error between each pattern and the corresponding cluster center. Although K-means is extensively used in literature, it suffers from several drawbacks. The objective function of the K-means is not convex and hence it may contain local minima. Consequently, while minimizing the objective function, there is possibility of getting stuck at local minima (also at local maxima and saddle point)[5]. The performance of the K-means algorithm depends on the initial choice of the cluster centers.

Particle swarm optimization (PSO) is a population-based algorithm. This algorithm simulates bird flocking or fish schooling behavior to achieve a self-evolution system. It can search automatically the optimum solution in the vector space. But the searching process isn't randomness. According to the different problems, it decides the searching way by the fitness function. This article will develop a clustering method based on evolutionary computation. It can distinguish automatically the cluster central position of K groups data set. For testing the performance of this architecture, this paper will show the experience results by using four artificial data sets.

2 Particle swarm optimization

Particle swarm optimization (PSO) is an evolutionary computation technique developed by Kenney and Eberhart in 1995[1]. The method has been developed through a simulation of simplified social models. PSO is based on swarms such as fish schooling and bird flocking. According to the research results for bird flocking, birds are finding food by flocking (not by each individual). Like GA[6,7], PSO must also have a fitness evaluation function that takes the particle's position and assigns to it a fitness value. The position with the highest fitness value in the entire run is called the global best (P_g). Each particle also keeps track of its highest fitness value. The location of this value is called its personal best (P_i). The basic algorithm involves casting a population of particles over the search space, remembering the best (most fit) solution encountered. At each iteration, every particle adjusts its velocity vector, based on its momentum and the influence of both its best solution and the best solution of its neighbors, then computes a new point to examine. The studies shows that the PSO has more chance to "fly" into the better solution areas more quickly, so it can discover reasonable quality solution much faster than other evolutionary algorithms. The original PSO formulate is described as[1,2,3]:

1) Social-Only Model:

$$V_{id} = V_{id} + c_2 \square rand(\ ) \square (P_{gd} - X_{id}) \tag{1}$$

2) Cognition-Only Model:

$$V_{id} = V_{id} + c_1 \cdot rand(\ ) \cdot (P_{id} - X_{id}) \tag{2}$$

3) PSO Combine Model:

$$V_{id} = V_{id} + c_1 \cdot rand(\ ) \cdot (P_{id} - X_{id})$$
$$+ c_2 * rand(\ ) \cdot (P_{gd} - X_{id}) \tag{3}$$

$$X_{id} = X_{id} + V_{id} \tag{4}$$

where d is the number of dimensions (variables), i is a particle in the population, g is the particle in the neighborhood with the best fitness, V is the velocity vector, X is the location vector, and P is the position vector for a particle's best fitness yet encountered. Parameters c_1 and c_2 are the cognitive and social learning rates, respectively. These two rates control the relative influence of the memory of the neighborhood to the memory of the particle.

3 A clustering analyzed technology applying the PSO algorithm

3.1 Clustering analysis

Clustering analysis is a technology which can classify the similar sample points into the same group from a data set[8]. It is a branch from multi-variable analysis and unsupervised learning rule in the pattern recognition. For space S which has the K groups and the N points $\{x_1, x_2, ..., x_N\}$, the definition of the clustering analysis is as follow:

1) Point vector set $X = \{x_1, x_2, ..., x_N\}$, the *ith* point vector x_i is a vector in n-dimensional space, the number of the pixel vector N □□.

2) Cluster set $C = \{C_1, C_2, ..., C_K\}$, K represents the cluster number by partitioning X. These K empty sets completely disjoint. Then

$$\begin{cases} C_i \neq \emptyset \quad for \quad i = 1,2,...K \\ C_i \cap C_j = \emptyset, i \neq j, \\ \bigcup_{i=1}^{K} C_i = X, \end{cases} \tag{5}$$

where $\emptyset$ is an empty set

The target of cluster analysis is the highest similar characteristic data in each cluster C_i and the least similar characteristic data in the other clusters. Each cluster C_i can get a n-dimension cluster center z_i. It is the center of the whole data in C_i. The iteration algorithm to calculate the cluster center z_i is as below[9].

Step1) Given a cluster center set $Z_m = \{z_1, z_2, ..., z_K\}$, obtained from the m^{th} iteration, assign each data point to the closet cluster center.

Step2) Obtain the new cluster center Z_{m+1} by computing the cluster center of each cluster based on partitioning of Step1.

Note that the iteration algorithm will be terminated when the moving value of cluster center is lower than the default value.

In many clustering techniques, the hard c-means (HCM, generally called K-means) algorithm is one of well-known hard clustering techniques. It can centralize the data point x_i to the closest cluster center z_i. The formula to decide the weight of the similarity is by using Euclidean distances.

$$D = \|x_i - z_j\|, \ i = 1,2,..., N, \ j = 1,2,..., K \tag{6}$$

A good method of cluster analysis can partition properly a data cluster set $X = \{x_1, x_2, ..., x_N\}$ to K clusters, where $2 \leq K \leq N-1$. When we have an unlabelled data set, it is very important to define a objective function for a cluster analysis method. Intuitively, each cluster shall be as compact as possible. Thus, the objective function of the hard c-means is created with the Euclidean norm. It represents as below:

$$J_{HCM} = \sum_{j=1}^{K} \sum_{i=1}^{N} \|x_i - z_j\|^2 \tag{7}$$

where z_j is the jth cluster center. The necessary condition of the minimum J_{HCM} is

$$z_j = \frac{1}{n_j} \sum_{x_i \in C_j} x_i, \ j = 1,2,..., K \tag{8}$$

where n_j is the whole sample number of group C_j. This equation is the basic of the K-means algorithm to iterate the cluster center of each group.

3.2 PSO-clustering

We will now introduce how to utilize PSO-clustering to decide the vector of the cluster center. PSO-clustering has much powerful distinguished capability in multi-dimensions space. In n-dimensions of Euclidean space R^n, it can distinguish the N data points into K groups and decide the center of each cluster. Let us set the encoding value of each particle as the string sequence which is constructed by real value. It represents K cluster centers. For n-dimensions space, the length of each particle is $K \square n$ words. The initial population produces randomly and represents the vector of the different cluster centers.

Fig. 1 is an example of the encoding of the single particle in the PSO initial population. Let $n=2, K=3$, i.e., the search space is two-dimension and the number of clusters is three. The string of this particle represents three cluster centers [(-4.5, 9), (23, 15) and (3.46, 5.23)].

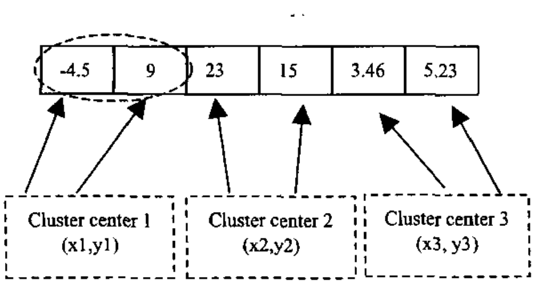

Fig.1 The encoding of the single particle in the PSO initial population.

After the encoding of the string of the particles, the execution of PSO clustering is as follow:

Step1) Initialize positions vector X and associated velocity V of all particles in the population randomly. Here the position X of the particle is the center position of each cluster.

Step2) Evaluate the fitness function for each particle. The method is assign point $x_i, i=1,2,...,N$ to cluster C_j, $j \square \{1,2,...,K\}$ iff

$$\|x_i - z_j\| \square \|x_i - z_p\|, p = 1,2,...,K \text{, and } j \square p \text{ .(9)}$$

And the fitness function for the PSO-clustering is given by :

$$J = \sum_{j=1}^{K}\sum_{i=1}^{N}\|x_i - z_j\|^2 \tag{10}$$

$$fitness = k/(J + J_o) \tag{11}$$

where the k is a positive constant, and J_o is a small-valued constant.

Step3) Compare particle's fitness evaluation with particle's best solution P_i. If current value is better than P_i, then set P_i value equal to the current value, and the P_i position equal to the current position in n-dimensional space.

Step4) Compare fitness evaluation with the population's overall previous best. If current value is better than the P_g (the global version of the best value), then reset P_g to the current particle's value and position.

Step5) Change velocities and position using equation(3) and (4).

Step6) Repeat Step2)-Step5) until a stop criterion is satisfied or a predefined number of iterations is completed.

4 Simulation results

The four artificial data sets (Data1_3C_2D, Data2_3C_2D, Data3_3C_3D, and Data4_2C_3D) have been used to verify the capability of PSO-clustering. Data1_3C_2D and Data2_3C_2D are the two-dimension data sets. Data3_3C_3D and Data4_2C_3D are the three-dimension data sets. In these experiment, we default the parameters of PSO-clustering $c_1 = c_2 = 1.5$, $k = 50$, $J_o = 0.1$. For getting the better convergence results, we set $V(t+1) = 0.75 \square V(t)$ during the process of the every iteration. It can adjust the moving speed value of the particles. The experiment result is described as follows.

Fig.2 shows the circular clustering architecture. The clustering result is expected. Fig.3 shows the clustering results by the data set in Fig.2. It represents the characteristic of circle partition of Euclidean distances. Fig.4 shows the mixture of spherical and ellipsoidal clusters which is constructed by 579 points of 2-D data. Fig.5 shows the final clustering results.

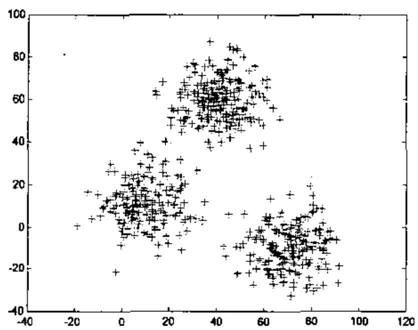

Fig.2 The 2-D artificial data set consisting of 600 data points. (Data1_3C_2D)

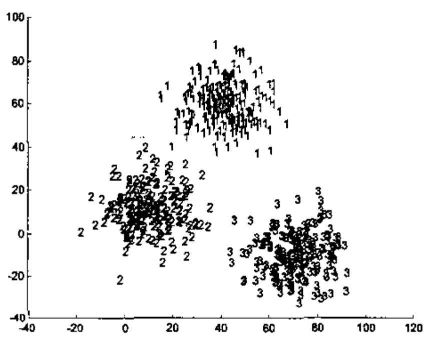

Fig.3 Clustered Data1_3C_2D by using PSO-clustering. The centers are shown with "○".

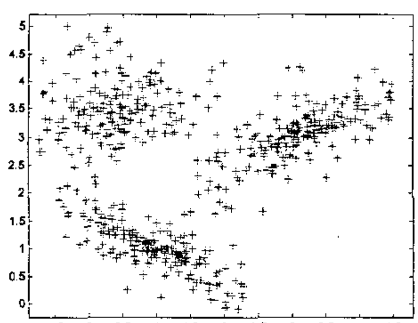

Fig.4 The 2-D artificial data set consisting of 579 data points (Data2_3C_2D).

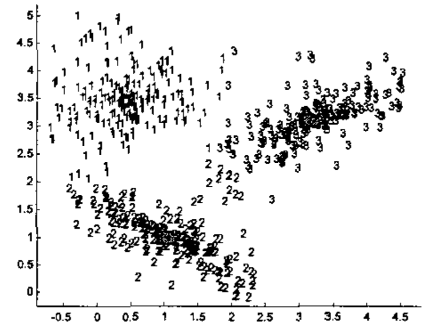

Fig.5 Clustered Data2_3C_2D by using PSO-clustering. The centers are shown with "○".

For proving the ability of this architecture, we use these two 3-D data sets Data3_3C_3D and Data4_2C_3D to experiment. The data set contains 600 data points distributed on three groups as shown in Fig.6. The clustering results are given in Fig.7. Fig.8 shows the 3-D data set consisting of 228 data points. Fig.9 shows the final clustering results.

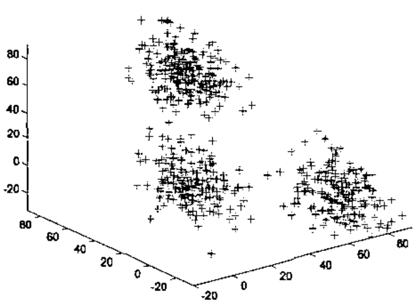

Fig.6 The 3-D artificial data set consisting of 600 data points. (Data3_3C_3D)

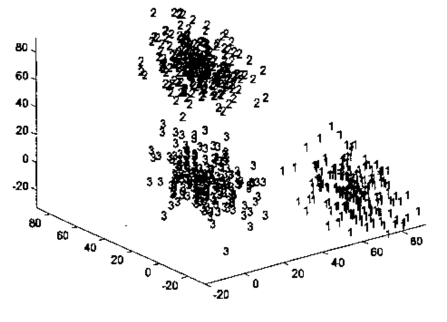

Fig.7 Clustered Data3_3C_3D by using PSO-clustering. The centers are shown with "○".

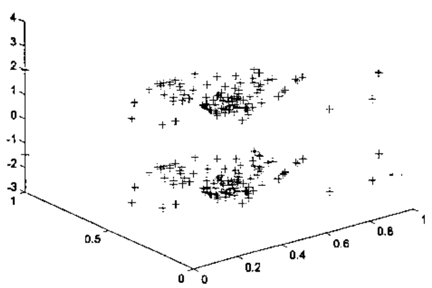

Fig.8 The 3-D artificial data set consisting of 228 data points. (Data4_2C_3D)

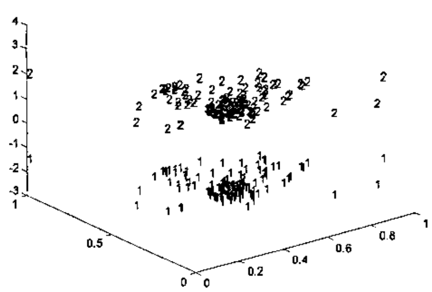

Fig.9 Clustered Data4_2C_3D using PSO-clustering. The centres are shown with "○".

Fig.10 and Fig.11 show the convergence situation of the system by using PSO-clustering in the different population sizes. From the result of Data2_3C_2D and Data3_3C_3D, it is easy to watch the best performance of PSO-clustering in different initial condition. The different population sizes may impact the speed to approach the best solution. But the system can approach the best solution after the enough iterations.

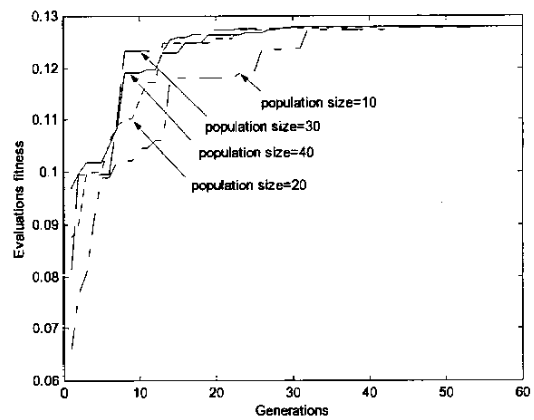

Fig.10 In different population sizes, the fitness function curve after the process of iteration (By using Data2_3C_2D).

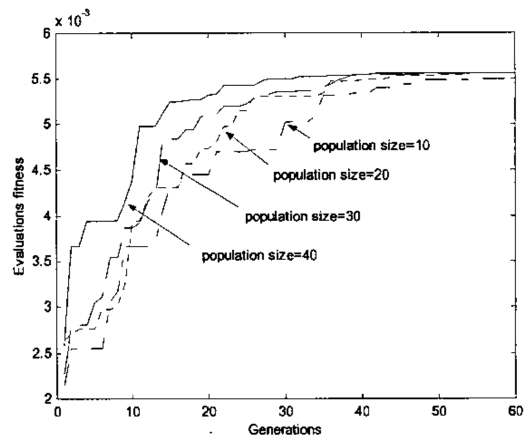

Fig.11 In different population sizes, the fitness function curve after the process of iteration (By using Data3_3C_3D).

Table 1 is the performance comparison between PSO-clustering, K-means, and Fuzzy C-means[10]. The object function J calculated by PSO-clustering is better than other algorithm calculated. In many times test, PSO can approximate the final convergence results. It can avoid such K-means easy to enter the local optimal solution due to different initial value.

Table 1 Object function and cluster center vector by using K-means, Fuzzy C-means, and PSO-clustering.

		K-means		Fuzzy C-means		PSO-clustering (pop. Size=40)
	J	Cluster centers	J	Cluster centers	J	Cluster centers
Data1_3C_2D	7.4132e+003	(70.1612,-9.8843) (9.7295,11.3981) (40.6229,59.8819)	7.4099e+003	(70.4208,-10.1129) (9.3373,11.1162) (40.5974,59.9508)	7.4091e+003	(70.3192,-10.1183) (8.9575,10.8130) (40.3797,59.5856)
Data2_3C_2D	391.1567	(1.0386,1.0425) (0.3817,3.4866) (3.1856,3.1137)	391.4012	(1.0585,1.0226) (0.3767,3.4592) (3.2414,3.1391)	390.8406	(1.0610,1.0080) (0.4070,3.4432) (3.1977,3.1059)
Data3_3C_3D	8.9943e+003	(70.1612,-9.8843,9.8843) (40.5252,59.7952,59.7952) (9.6824,11.2399,11.2399)	8.9938e+003	(70.1586,-10.3445,-10.3445) (40.4632,60.0401,60.0401) (9.6721,11.0839,11.0839)	8.9904e+003	(70.1750,-10.0240,-10.2396) (40.1818,59.8128, 59.9821) (8.9577, 10.8288,10.8596)
Data4_2C_3D	38.8497	(0.4018, 0.3440,-1.4507) (0.4018,0.3440,1.9618)	38.8646	(0.4020,0.3444,-1.4508) (0.4020,0.3444,1.9619)	38.4240	(0.3893,0.3208,-1.4547) (0.3904,0.3197,1.9624)

5. Conclusion

This paper provides a clustering analysis algorithm based on PSO, called PSO-clustering. PSO will base on the minimum object function J to search automatically the data cluster centers of n-dimension Euclidean space R^n. Traditional cluster algorithm such as K-means may get stuck at local optimal solution, depending on the choice of the initial cluster centers. It can't make sure to solve the global optimal solution every time. Related to the other evolution algorithm, PSO needs the less parameter to decide. When it be executed, it can avoid to enter the local optimal solution. The experiment results by using four artificial data sets show the PSO-clustering algorithm has better performance than the traditional clustering analysis algorithm.

References

[1] J. Kennedy, R. Eberhart, "Particle Swarm Optimization," *Proc. of IEEE international Conference on Neural Networks (ICNN)*, Vol.IV, pp.1942-1948, Perth, Australia, 1995.

[2] R. Eberhart, J. Kennedy, "A new optimizer using particle swarm theory," *Proc. 6th Int. Symposium on Micro Machine and Human Science*, pp.39-43, 1995.

[3] Russell C. Eberhart, Yuhui Shi, "Particle Swarm Optimization: Developments, Applications and Resources," *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2001)*, Seoul, Korea, 2001.

[4] J.T. Tou, R.C. Gonzalez, *Pattern Recognition Principles*, Addison-Wesley, Reading, 1974.

[5] S.Z. Selim, M.A. Ismail, "K-means type algorithms: a generalized convergence theorem and characterization of local optimality," *IEEE Trans. Pattern Anal. Mach. Intell.6*, pp. 81-87, 1984.

[6] J. L. R. Filho, P.C. Treleaven, C. Alippi, "Genetic algorithm programming environments," *IEEE Comput.* 27(1994) 28-43..

[7] Ujjwal Maulik, Sanghamitra Bandyopadhyay, "Genetic algorithm-based clustering technique," *Pattern Recognition* 33, pp. 1455-1465, 2000.

[8] M. R. Anderberg, *Cluster Analysis for Application*, Academic Press, New York, 1973.

[9] Allen Gersho, Robert M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, 1992.

[10] J.C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Plenum Press, New York, 1981.