

Research Article

Combinatorial Clustering Algorithm of Quantum-Behaved Particle Swarm Optimization and Cloud Model

Mi-Yuan Shan,¹ Ren-Long Zhang,¹ and Li-Hong Zhang²

¹ College of Business Administration, Hunan University, No. 11 Lushan South Road, Changsha 410082, China

² Liverpool Business School, Liverpool John Moores University, Redmonds Building, Brownlow Hill, Liverpool L3 5UX, UK

Correspondence should be addressed to Ren-Long Zhang; zhangrenlonggj@163.com

Received 14 April 2013; Revised 22 September 2013; Accepted 7 October 2013

Academic Editor: T. Warren Liao

Copyright © 2013 Mi-Yuan Shan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a combinatorial clustering algorithm of cloud model and quantum-behaved particle swarm optimization (COCQPSO) to solve the stochastic problem. The algorithm employs a novel probability model as well as a permutation-based local search method. We are setting the parameters of COCQPSO based on the design of experiment. In the comprehensive computational study, we scrutinize the performance of COCQPSO on a set of widely used benchmark instances. By benchmarking combinatorial clustering algorithm with state-of-the-art algorithms, we can show that its performance compares very favorably. The fuzzy combinatorial optimization algorithm of cloud model and quantum-behaved particle swarm optimization (FCOCQPSO) in vague sets (IVSs) is more expressive than the other fuzzy sets. Finally, numerical examples show the clustering effectiveness of COCQPSO and FCOCQPSO clustering algorithms which are extremely remarkable.

1. Introduction

Clustering is a popular data analysis method and plays an important role in data mining. Clustering is an important class of unsupervised learning techniques. How to get the best or a satisfying solution quickly has great significance for improving productivity and the development of society. So far, it has been widely applied in many fields, like web mining, pattern recognition, resources allocation, machine learning, spatial database analysis, artificial intelligence, and so on. The existing clustering algorithms can be simply classified into the following two categories: hierarchical clustering and partitional clustering [1]. Though the K -means algorithm is widely used to solve problems in many areas, KM is very sensitive to initialization, the better centers we choose, the better results we get. Also, it was easily trapped in local optimal [2]. Recently much work was done to overcome these problems. Traditional cluster analysis requires every point of the data set to be assigned into a cluster precisely, so we call it hard clustering. But in fact, most things exist in ambiguity in the attribute, there are no explicit boundaries among the things, and no the nature of either-or. So, the theory

of the fuzzy clustering is more suitable for the nature of things, and it can more objectively reflect the reality. To date, development of clustering has focused on data processing speed and efficiency. How appropriate the number of clusters may affect the clustering results. For example, a smaller number of clusters can help clearly identify the original data structure but key hidden information may not be obtained.

Data clustering was an exploratory or descriptive process for data analysis, where similar objects or data points were identified and grouped into a set of objects called a cluster [3, 4]. The clustering problem was defined as the problem of classifying a collection of objects into a set of natural clusters without any prior knowledge. The main goal of cluster analysis, an unsupervised classification technique was to analyze similarity of data and divide it into several cluster sets of similar data [5]. In terms of internal homogeneity and external separation, data within a cluster had high similarity, but there was less similarity between clusters. As a result, clustering is utilized to divide the data into several cluster sets for analysis and reduce data complexity. Generally, clustering has several steps: (1) proper features were selected from data as the basis for clustering; (2) a suitable clustering algorithm

was selected based on data type; (3) the evaluation principle determines the clustering results, which were provided for experts to interpret [6]. Basically, clustering algorithms can be hierarchical or partitional. Among the unsupervised clustering algorithms, the hierarchical method was most typical [7, 8]. But the problem with partitional methods was necessary to set the number of clusters before clustering as in K -means or K -medoids [9, 10]. In addition, Gath and Geva [11] proposed an unsupervised clustering algorithm based on the combination of fuzzy C -means and fuzzy maximum likelihood estimation. Lorette et al. [12] proposed an algorithm based on fuzzy clustering to dynamically determine the number of clusters in a data set. Furthermore, SYNERACT [13], an alternative approach to ISODATA [14] combining K -means with hierarchical descending approaches, did not require specifying the number of clusters. Based on K -means, other unsupervised clustering algorithms have also been developed, such as X -means and G -means [15, 16]. Omran et al. proposed a new dynamic clustering approach based on binary-PSO (DCPSO) algorithm. Paterlinia and Krink (2006) applied PSO for partitional clustering algorithms partition the data set into a specified number of clusters. These algorithms try to minimize certain criteria and can therefore be treated as optimization problems. In addition, Ouadfel et al. (2010) presented a modified PSO algorithm for automatic image clustering algorithm. Where each cluster groups together similar patterns. According to conducted experiments, the proposed approach outperforms K -means, FCM and KHM algorithm [17, 18]. Such an algorithm can directly generate a suitable number of clusters. In addition, it is critical to have a suitable measure for the effectiveness of a clustering method. Though some measures have been proposed in the area of classification, like Wallace's information measure, few have been proposed for clustering algorithms [19]. The most frequently applied measures were cohesion, within-cluster variance, separation, and between cluster variance [20]. The DBSCAN clustering algorithm can also be used in four applications that were using 2D points, 3D points, 5D points and 2D polygons of real world problems [21]. Those clustering algorithms have been applied in the clustering analysis for many years, but those algorithms are also unilateral.

Now, many improved velocity update rules have been developed, including the inertia weight (W) method [22], constriction factor method [23], guaranteed convergence method [24], improvement social model [25], global-local best value based PSO method, and GLbest-PSO method [26]. In addition, Montes de Oca et al. [27] proposed a novel PSO algorithm combining a number of algorithmic components that showed distinct advantages for optimization speed and reliability in their experimental study.

In addition to these methods, Kennedy and Eberhart [28] proposed the binary-PSO (BPSO), which is applied to optimization problems requiring discrete or binary variables for solutions. A novel optimization algorithm, based on a modified binary-PSO with mutation (MBPSOM) combined with a support vectormachine (SVM), was proposed to select the fault feature variables for fault diagnosis [29]. Chuang et al. [30] presented improved BPSO for feature

selection using gene expression data. In addition, Xie et al. [31] suggested the distributive-PSO (DPSO) algorithm. The DPSO algorithm was used to solve problems in which it was easy for the PSO to fall into a local value from which it cannot escape. In particle searching, the DPSO algorithm was utilized to randomly select particles for mutation. Liu et al. [32] proposed PSO with mutation based on similarity, in which similarity and collectivity are introduced. Most recently, Nani and Lumini [33] used the Nearest Neighbor approach for prototype reduction using PSO and they found the Nearest Neighbor approach to be good. In order to achieve a dynamic clustering, where the optimum number of clusters was also determined within the process, the so-called multidimensional PSO (MDPSO) method extended the native structure of PSO particles in such a way that they can make interdimensional pass with a dedicated dimensional PSO process [34]. In addition, Ouadfel et al. [35] presented an automatic clustering using a modified PSO (ACMPPO) algorithm. Also, Martinez et al. [36] proposed a swarm intelligence feature selection algorithm based on the initialization and update of only a subset of particles in the swarm. When compared to many of the other population-based approaches such as other algorithms such as GA, ACO, and DE, the convergence rate of the population was much slower for PSO algorithm [37]. Meanwhile, the PSO algorithms was inspired by social behavior among individuals, for instance, bird flocks. Intelligence particles representing a potential problem solution move through the search space.

2. Combinatorial Optimization Algorithm of Cloud Model and Quantum-Behaved Particle Swarm Optimization

2.1. Quantum-Behaved Particle Swarm Optimization. Genetic clustering algorithm is randomized search and optimization techniques guided by the principles of evolution and natural genetics, having a large amount of implicit parallelism. When the clustering algorithm is terminated in the limited time, one is used to find the methodology exhibiting rather poor search precision. Therefore, this unfavorable characteristic may to some extent indicate that the global optimality of the solution given by GA is dubious. Ant Colony clustering algorithm requires a longer period of time to run. At this stage, ants consider only the similarity without considering the density to pick up and down the object data, as results, data objects will be orderly dispersed. Because the ant colony clustering algorithm is a stochastic algorithm. Class of the results of cluster center reselection has the potential to separate from each other class of cluster center. Particle swarm optimization clustering algorithm is a newly rising evolutionary computation technique based on swarm intelligence, pso possesses the better convergent speed and computational precision compared with the traditional algorithms such as GA, PSO, ACO, and DE, it can effectively search out the global optimal solution in the space of solution [38–40]. Let us denote M as the swarm size and define n as the dimensionality of the search space. Particle swarm optimization is a global

optimization algorithm. At the same time, the convergence rate is accelerated by the modified PSO algorithm. From the analysis of quantum physics point of view, the particle state can be described the energy and momentum, measurement results of concrete can be used wave function $\psi(x, t)$. The wave function model of the square is a point particle in the space of probability. Therefore, in the QPSO algorithm, each particle is not expressed by the speed and position but as a quantum state. Particle probability distribution at a location space can be obtained by the probability density function of wave function $|\psi(x, t)|$. Once we determine the probability density function of particle position, we can obtain the probability distribution. The particle position update equation under the guidance of Monte Carlo method is expressed as

$$x_{ij}(t+1) = p_{ij} \pm 0.5 \cdot L_{ij} \cdot \ln\left(\frac{1}{u_{ij}}\right), \quad (1)$$

where u_{ij} is random numbers uniformly distributed at the interval (0, 1). The QPSO algorithm employs local version constriction factor method and global version inertia weight method simultaneously to achieve relatively high performance and $p_{ij}(t)$ is local version constriction factor. It is a random point between $p_{best_i,j}$ and $p_{gbest,j}$. The formula of $p_{ij}(t)$ is calculated as

$$\begin{aligned} p_{ij}(t) &= \varphi_{ij} \cdot P_{best_{i,j}}(t) + (1 - \varphi_{ij}) \cdot G_{best_i}(t), \\ P_{best_{i,j}} &= [P_{best_{i,1}}, P_{best_{i,2}}, \dots, P_{best_{i,n}}], \\ G_{best} &= [G_{best_1}, G_{best_2}, \dots, G_{best_n}]. \end{aligned} \quad (2)$$

The analysis results are as follows:

$$x_{ij}(t+1) = p_{ij}(t) \pm \beta \cdot |p_{ij}(t) - x_{ij}(t)| \cdot \ln\left(\frac{1}{u_{ij}}\right). \quad (3)$$

The method introduced the concept of average optimal position M_{best} . M_{best} Variables are expressed in the current moment group at all the best location history as follows:

$$\begin{aligned} P_{best_i} &= [P_{best_{i,1}}, P_{best_{i,2}}, \dots, P_{best_{i,n}}], \\ M_{best} &= [M_{best_1}, M_{best_2}, \dots, M_{best_n}] \\ &= \left[\frac{1}{M} \sum_{i=1}^M P_{best_{i,1}}, \frac{1}{M} \sum_{i=1}^M P_{best_{i,2}}, \dots, \frac{1}{M} \sum_{i=1}^M P_{best_{i,n}} \right]. \end{aligned} \quad (4)$$

So, the particle position updates formula transformed can be expressed as

$$x_{ij}(t+1) = p_{ij}(t) \pm \beta \cdot |M_{best_j}(t) - x_{ij}(t)| \cdot \ln\left(\frac{1}{u_{ij}}\right). \quad (5)$$

Parameter β which is set to linear gradient from 1.0 to 0.5 can generally get good results.

Values k for the uniform distribution random number between (0, 1), get basic expressions of QPSO algorithm as follows:

$$\begin{aligned} x_i(t+1) &= \varphi_i \cdot P_{best_i}(t) + (1 - \varphi_i) \cdot G_{best}(t) \\ &\quad + \beta \cdot |M_{best}(t) - x_i(t)| \cdot \ln\left(\frac{1}{u}\right), \\ &\quad k \geq 0.5, \\ x_i(t+1) &= \varphi_i \cdot P_{best_i}(t) + (1 - \varphi_i) \cdot G_{best}(t) \\ &\quad - \beta \cdot |M_{best}(t) - x_i(t)| \cdot \ln\left(\frac{1}{u}\right), \\ &\quad k < 0.5. \end{aligned} \quad (6)$$

Necessary and sufficient conditions for a point not algorithm global convergence of the PSO, but it can make the algorithm have better convergence speed and accuracy, the reasons not only involve the analysis of the convergence of the algorithm, but also research on search mechanism and complexity more involved algorithm. How to control search mechanism of the PSO by these design means the controlling of $L_{i,j}(t)$. The first algorithm method is evaluated in detail by the following formula:

$$L_{ij} = 2 \cdot \beta \cdot |M_{best_j} - x_{ij}|, \quad (7)$$

or

$$L_{i,j}(t) = 2\beta \cdot |p_{i,j}(t) - x_{i,j}(t)|. \quad (8)$$

The evolution equation of particle deduction into is expressed as follows:

$$\begin{aligned} X_{i,j}(t+1) &= p_{i,j}(t) \pm \alpha \cdot |p_{i,j}(t) - X_{i,j}(t)| \\ &\quad \cdot \ln\left[\frac{1}{u_{i,j}(t)}\right], \quad u_{i,j}(t) \sim U(0, 1). \end{aligned} \quad (9)$$

The second kind of method is introduced in the algorithm of the average best position as follows:

$$\begin{aligned} C(t) \cdot C(t) &= (C_1(t), C_2(t), \dots, C_n(t)) \\ &= \frac{1}{M} \sum_{i=1}^M P_i(t) \\ &= \left(\frac{1}{M} \sum_{i=1}^M P_{i,1}(t), \frac{1}{M} \sum_{i=1}^M P_{i,2}(t), \dots, \frac{1}{M} \sum_{i=1}^M P_{i,N}(t) \right). \end{aligned} \quad (10)$$

The evolution equation of particle deduction into can be expressed as

$$X_{i,j}(t+1) = p_{i,j}(t) \pm \beta \cdot |C_j(t) - X_{i,j}(t)| \cdot \ln \left[\frac{1}{u_{i,j}(t)} \right], \quad u_{i,j}(t) \sim U(0,1). \quad (11)$$

The process and the concrete steps of the QPSO algorithm are described as follows:

Step 1. Set $t = 0$, initializing the particle swarm in the position of each particle and individual best position $p_i(0) = x_i(0)$.

Step 2. Calculate the average best position of particle swarm.

Step 3. For each particle group i ($1 \leq i \leq M$), Steps 4 to 7.

Step 4. Calculate the fitness of particle in current position $x_i(t)$, according to equation individual best position update particle. If $f[x_i(t)] < f[p_i(t-1)]$, $p_i(t) = x_i(t)$. If not, $p_i(t) = p_i(t-1)$.

Step 5. For the particle, the adaptation values $p_i(t)$ and the global best position values $G(t-1)$ are compared. If $f[p_i(t)] < f[G(t-1)]$, $G(t) = p_i(t)$. If not, $G(t) = G(t-1)$.

Step 6. According to each dimension of the particles, calculate equation at random point position.

Step 7. Calculate the new position calculating particle.

Step 8. If the end condition is not satisfied, $t = t + 1$.

Return to Step 2; otherwise, the algorithm terminates.

2.2. Cloud Model. Cloud model which is proposed based on the traditional fuzzy mathematics and probability statistics theory is a conversion model between qualitative concept and quantitative values. The cloud model which was proposed by Li et al. (1998), is novel uncertainty reasoning technology. Cloud model is an effective tool in uncertain transforming between qualitative concepts and their quantitative expressions. There are various implementation approaches of the cloud model, resulting in different kinds of cloud models. The normal cloud model is the most commonly used model, which is based on the normal distribution and the Gaussian membership function. It can be described as follows. They integrate fuzziness and randomness using digital characteristics, such as expected value E_x , entropy E_n and hyper entropy He.

In the domain of cloud droplets, E_x is the most representative of the qualitative concept, the expectation is that it is the domain of the center value. Entropy E_n is the qualitative concept of randomness and fuzziness, is jointly determined, capable of measuring particle size is represents a qualitative concept. E_n is a measure of the randomness of qualitative concept and reflects the discrete degree of the cloud droplet. The excess entropy E_n is the entropy of the uncertainty of

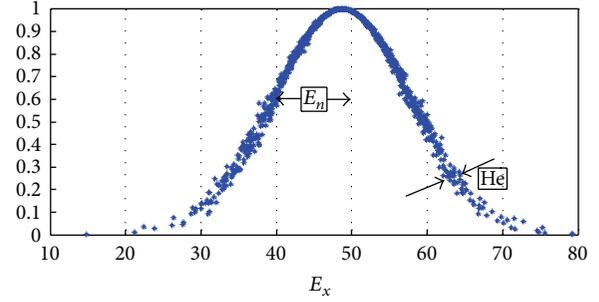


FIGURE 1: The digital characteristics of the cloud.

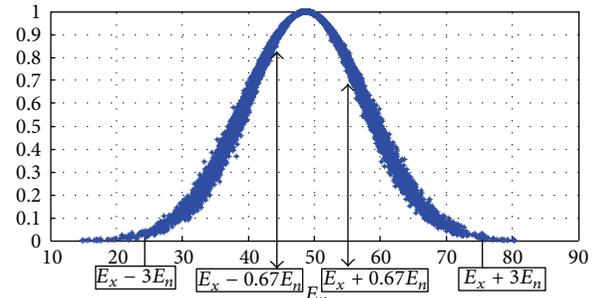


FIGURE 2: Cloud droplets contribute to qualitative concept.

measurement. The digital characteristics of the cloud are shown in Figure 1.

The hybrid sets which contain fuzziness and probabilities are defined over continuous universe of discourse. Domain $(-\infty, +\infty)$, all elements are in total contribution to the concept of C and expression formula can be expressed as

$$C = \frac{\int_{-\infty}^{+\infty} \mu_A(x) dx}{\sqrt{2\pi}E_n} = \frac{\int_{-\infty}^{+\infty} e^{-(x-E_x)^2/(2E_n^2)} dx}{\sqrt{2\pi}E_n} = 1, \quad (12)$$

$$\Delta C \approx \mu_A(x) * \frac{\Delta x}{(\sqrt{2\pi}E_n)}.$$

Domain $(E_x - 3E_n, E_x + 3E_n)$, all elements in total contribution to the concept of

$$C_{E_x \pm 3E_n} \cdot C_{E_x \pm 3E_n} = \frac{1}{\sqrt{2\pi}E_n} \int_{E_x - 3E_n}^{E_x + 3E_n} \mu_A(x) dx = 99.74\%. \quad (13)$$

The contribution of different areas of the cloud droplet swarm on the qualitative concept is different in Figure 2.

X , E_n , and He^2 obey normal distribution. Probability density functions of E_n . Consider,

$$f_{E_n}(x) = \frac{1}{\sqrt{2\pi}He} e^{-(x-E_n)^2/2He^2}, \quad (14)$$

$$f_x(x | E_n) = \frac{1}{\sqrt{2\pi}E_n'} e^{-(x-E_x)^2/2E_n'^2}.$$

It can be inferred that

$$\begin{aligned}
 f(x) &= \frac{1}{\sqrt{2\pi}E_n} e^{(x-E_x)^2/2E_n^2}, \\
 f_x(x) &= f_{E'_n}(x) \times f_x(x | E_n) \\
 &= \int_{-\infty}^{+\infty} \frac{1}{2\pi\text{He} |y|} e^{((x-E_x)^2/2y^2) - ((y-E_n)^2/2\text{He}^2)} dy, \\
 f_{X,\mu}(x) &= f_\mu(y) f_X(x | \mu = y) \\
 &= \begin{cases} \frac{1}{2\pi\text{He} \ln y} e^{(x-E_x - \sqrt{-2 \ln y} E_n)^2 / 4\text{He}^2 \ln y}, \\ (0 < y \leq 1, E_x \leq x < +\infty), \\ \frac{1}{2\pi\text{He} \ln y} e^{(x-E_x + \sqrt{-2 \ln y} E_n)^2 / 4\text{He}^2 \ln y}, \\ (0 < y \leq 1, -\infty < x \leq E_x). \end{cases}
 \end{aligned} \tag{15}$$

We can calculate the digital characteristic entropy estimate values $\hat{E}_n = (\bar{z}^2 - (S^2/2))^{1/4}$. At the same time, we can calculate the digital characteristic. The excess entropy estimate values $\hat{H}_e = \sqrt{(\bar{z} - (\bar{z}^2 - (S^2/2))^{1/2})}$.

Cloud model is also used in two-order Gaussian distribution to produce cloud drops whose distribution displays high kurtosis and fat tail with power-law decay. In sociology and economics, many phenomena have been found to share high kurtosis and fat tail because of preferential attachment in evolution processes. This paper tries to investigate the relation between Gaussian distribution and fat-tail distribution, and to construct fat-tail distribution based on Gaussian distribution with iterations to depict more uncertain phenomena. If the Gaussian distribution as 1 order Gaussian distribution, and then order 1 x_1 probability density function is expressed as a random variable. Consider,

$$f(x) = \left(\frac{1}{\sqrt{2\pi}\sigma} \right) e^{(x-u)^2/2\sigma^2}. \tag{16}$$

For three of the higher order Gaussian iteration with simple parameters, we study the changing trend of its mathematical properties and make lots of experimental analysis.

When $u_i = 0$ ($i = 1, 2, \dots, n$), $u_i = 0$ and $\sigma \neq 0$; then

$$\begin{aligned}
 E(X_i) &= u_i = 0, \\
 \text{Var}(x_i) &= \sum_{i=1}^{n-1} u_i^2 + \sigma^2 = \sigma^2, \\
 \frac{E(x_i - E(x_i))^4}{\text{Var}(x_i)} &= \frac{3n\sigma^4}{\sigma^4}.
 \end{aligned} \tag{17}$$

When $u_i = u$ ($i = 1, 2, \dots, n$) and $r = \sigma/u$; then

$$\begin{aligned}
 E(X_i) &= u_i = u, \\
 \text{Var}(x_i) &= \sum_{i=1}^{n-1} u_i^2 + \sigma^2 = \sigma^2 + (n-1) \frac{\sigma^2}{r^2}, \\
 \frac{E(x_i - E(x_i))^4}{\text{Var}(x_i)} &= \frac{3^p r^4 + 6r^2 \sum_{i=1}^{n-1} 3^{n-i} + \sum_{i=1}^{n-1} (6i-5) 3^{n-i}}{r^4 + (n-1)^2 + 2(n-1)r^2}.
 \end{aligned} \tag{18}$$

When $u_i = r^i \sigma$ ($i = 1, 2, \dots, n$), then $E(X_i) = u_i = r^i \sigma$,

$$\begin{aligned}
 \text{Var}(x_i) &= \sum_{i=1}^{n-1} u_i^2 + \sigma^2 = \left(\sum_{i=1}^{n-1} r^{2i} + 1 \right) \sigma^2 \\
 &= \sigma^2 \left(\frac{r^{2n} - 1}{r^2 - 1} \right), \\
 \frac{E(x_i - E(x_i))^4}{\text{Var}(x_i)} &= \left(3^p + 6 \sum_{i=1}^{n-1} 3^{n-i} r^{2i} + 6 \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} 3^{n-i} r^{2(i+j)} \right. \\
 &\quad \left. + \sum_{i=1}^{n-1} 3^{n-i} r^{4i} \right) \\
 &\quad \times \left(1 + 2 \sum_{i=1}^{n-1} r^{2i} + \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} r^{2(i+j)} \right)^{-1}.
 \end{aligned} \tag{19}$$

In short, the combinatorial clustering algorithms based on the Cloud, transform a qualitative question, the effectiveness evaluation of the combinatorial clustering algorithms into a quantitative question, and consider the fuzzy factors and random factors effects appeared in the evaluating cause. In the cloud model, E_n determines the steepness of the normal cloud, the greater the E_n , the wider the cloud covered the level range, otherwise the more narrow. According to “ $3E_n$ ” rule of normal cloud, take $\theta_1 = 3$ in the algorithm, which affect the degree of discrete cloud droplets.

2.3. Combinatorial Optimization Clustering Algorithm. Combination with the basic principle of particle swarm optimization, a rapid evolutionary algorithm is proposed based on the characteristics of the cloud model on the process of transforming a qualitative concept to a set of quantitative numerical values, namely cloud hyper mutation particle swarm optimization algorithm. Its core idea is to achieve the evolution of the learning process and the mutation operation by the normal cloud particle operator. The COCQPSO can be modeled naturally and uniformly, which makes it easy and natural to control the scale of the searching space. The simulation results show that the proposed algorithm has fine capability of finding global optimum, especially for multimodal functions.

TABLE 1: Solutions of sphere functions with different dimensionality N values.

Dimensionality	$N = 2$	$N = 5$	$N = 10$	$N = 30$
5	$8.785e - 165$	$1.650e - 144$	$4.176e - 129$	$4.666e - 120$
10	$2.979e - 129$	$1.237e - 125$	$4.412e - 117$	$5.301e - 111$
30	$1.192e - 067$	$3.888e - 071$	$1.766e - 068$	$3.069e - 060$
50	$4.201e - 049$	$3.038e - 050$	$3.121e - 048$	$1.191e - 041$
100	$5.127e - 029$	$6.976e - 031$	$4.380e - 030$	$7.082e - 026$

TABLE 2: Dimension, initial value scope, and target values of 6 test functions.

Number	Functions	Dimension	The scope of initial values (10^4)	Acceptable level (%)
F1	Sphere	30	$[-100, 100]$	0.0100
F2	Rosenbrock	30	$[-100, 100]$	100.00
F3	Rastrigrin	30	$[-100, 100]$	100.00
F4	Griewank	30	$[-600, 600]$	0.1000
F5	Schaffer	2	$[-100, 100]$	0.00001
F6	Ackley	30	$[-32, 32]$	0.1000

Choose global extreme value G_{best} in mutation as E_x , because at this point it may have been trapped in local optimal algorithm. And according to the principle of sociology, the current outstanding individuals usually also exist better individuals. There are two positions are used in the particle for updating the velocity; one is the global experience position of all particles, which memorizes the global best solution obtained through all particles; the other is each particle's local optimal point, which memorizes the best position that particle has ever moved to. Namely the local optimal point in its surroundings more chance to find the optimal solution. Parameter K is set too large and the variation is too frequent, which influence the operational efficiency of algorithm. Set is too small, the precision will be reduced. And because of the particle swarm algorithm in the early evolutionary convergence speed, convergence speed gradually slows down late, so it is difficult to give parameters K set a perfectly reasonable fixed values. In this paper, we make the $K = G_{best}/2$, the K value along with the global optimal value G_{best} dynamic decreases, and realized adaptive adjustment.

For threshold selection of N , Sphere function, for example, to verify the influence of different N values of CHPSO algorithm precision. The experiment parameter settings are as follows: population sizes are 100, scope of the initial value for $(-5, 5)$, the largest iterative algebra are 1,000, respectively, for, 10, 30, and 50 dimension 5 and 100 Sphere function, in the case of N take 2, 5, 10, and 20 independent running 50 times averaging, measure parameter $K = G_{best}/2$. The experimental results are shown in Table 1.

We can see from Table 1, the smaller is the threshold N of dimensionality than 10 low dimensional function, the higher is the accuracy of the solution. For functions test, the acceptable result proportion of the value is shown in Table 2.

Test functions dimension, the range of initial value, the acceptable values of the experimental results of the COCQPSO algorithm, the Cloud genetic algorithm (CGA), the adaptive particle swarm optimization (APSO), Gaussian-Dynamic PSO, GDPSO), are shown in Table 3. All

experimental population size is 20, the number of iterations is 1,000, 100 times and each function operate independently and then we compare the optimal value, average value, and the success rate. To fully embody the effect of mutation operator, COCQPSO. Dynamically adjusted inertia weight w is not in, take a fixed value 0.725. Accelerate the constants $C_1 = 1$ and $C_2 = 1$, threshold value $N = 2$.

Use the knowledge of probability theory to prove the convergence of the COCQPSO algorithm. The combinatorial algorithm could be adaptive control under the guidance of the scope of the search space, and they are best under the conditions of the larger search space to avoid the local optimal solution. A typical functions of comparative experiment results show that the algorithm can avoid trapping in local optimal solution, and enhance the ability of global optimization at the same time to be able to more quickly converge to the global optimal solution. At the same time, the quality and efficiency of optimization is better than the other algorithms. The simulation results show that the complex constrained optimization problem, optimization algorithm, and excellent performance, especially for the super high-dimensional constraint optimization problem, the combinatorial algorithm obtained the higher accuracy of the solution.

3. Simulation Experiment Results

3.1. The Comparison Results of the Algorithms. PSO is inspired by social behavior among individuals, for instance, bird flocks. Particles (individuals) representing a potential problem solution move through the search space. The COCQPSO and FCOCQPSO algorithms focus on the collective behaviors that result from the local interactions of the individuals and interactions with their environment. The algorithms experimental results prove that the hybridization strategies are effective. The sequence algorithms with the enlarged pheromone table are superior to the other algorithms because the enlarged pheromone table diversifies the generation of

TABLE 3: Performance comparison among APSO, GDPSO, and COCQPSO.

Functions	APSO			GDPSO		
	Average	Success rate	Optimal value	Average	Success rate	Optimal value
Sphere	1.43e - 18	100.00	3.99e - 29	1.21e - 08	100.00	1.08e - 25
Rosenbrock	36.0900	95.00	0.0003	25.659	98.00	24.9651
Rastrigrin	65.4700	100.00	0.0000	171300	98.00	0.000
Griewank	0.00824	100.00	0.0000	1.80e - 03	100.00	0.000
Schaffer	0.99951	98.00	0.0000	6.06e - 04	74.00	0.000
Ackley	1.12e - 5	100.00	—	—	—	—

Functions	CGA			COCQPSO		
	Average	Success rate	Optimal value	Average	Success rate	Optimal value
Sphere	1.4949e - 007	100.00	1.3147e - 09	1.01e - 10	100.00	4.2461e - 225
Rosenbrock	25.749e - 008	98.00	9.6253e - 09	26.4879	98.00	9.0725e - 008
Rastrigrin	3.000000	100.00	3.000000	3.000000	100.00	3.000000
Griewank	0.0000	100.00	0.0000	0.0000	100.00	0.0000
Schaffer	-1.031628	100.00	-1.031628	-1.03156	100.00	-1.031601
Ackley	0.998004	100.00	0.998004	3.90e - 12	100.00	4.44e - 15

TABLE 4: Comparison of error rates for the algorithms.

Data set	Criteria	GA	ACO	PSO	COCQPSO	FCOCQPSO
Data 1	Best	2.9230	2.6720	2.9220	2.8520	2.8500
	Worst	3.0000	2.7030	3.0000	2.9300	2.9282
	Average	2.9600	2.6900	2.9510	2.8810	2.8800
Data 2	Best	0.3440	1.5780	0.3430	0.2740	0.2730
	Worst	0.3760	1.6100	0.3750	0.3060	0.3045
	Average	0.3600	1.5940	0.3590	0.2895	0.2890
Data 3	Best	3.3380	2.9840	3.3280	2.6300	2.6100
	Worst	3.3850	3.0470	3.3750	3.3000	3.2980
	Average	3.3750	3.0155	3.3650	3.3020	3.3000
Data 4	Best	11.8530	18.7340	11.8430	11.1050	11.0045
	Worst	11.9160	18.9060	11.9060	11.2040	11.1800
	Average	11.8730	18.8150	11.8630	11.1500	11.1020

new solutions of the COCQPSO and FCOCQPSO algorithms, which prevent traps into the local optimum.

To compare the performance of the COCQPSO and FCOCQPSO algorithms with those of other clustering algorithms such as Genetic algorithm (GA) and ant colony optimization (ACO), we have made lots of clustering experiments such as four different types of artificial data sets to get the average and standard deviation. Four different types of artificial data sets have been randomly generated from a multivariate uniform distribution. Data set 1: the wine dataset. This dataset contains chemical analyses of 178 wines derived from three different cultivars. Data set 2: the iris dataset. This dataset contains three categories of 50 objects. Data set 3: the glass identification dataset. This dataset contains 214 objects with nine attributes. Data set 4: the CMC dataset. Clustering experimental results of comparison of error rates for the algorithms is given in Table 4.

From the results given in Table 4, we can see that none of the previously proposed algorithms outperform the COCQPSO and FCOCQPSO algorithms in terms of the average

and best objective function values for the four datasets used here. The proposed algorithms of this study are effective, robust, easy to tune, and tolerably efficient as compared with other algorithms.

The COCQPSO and FCOCQPSO algorithms have been tested in comparison with the GA and PSO for artificial and real world data. Finally, good experimental results show the feasibility of the proposed algorithms. In all our experiments, both the data is randomly or heuristically initialized. For complex problems, it is advised to choose the FCOCQPSO algorithm for the initialization, in order to reach the best solutions. Clustering results of the FCOCQPSO algorithm are shown in Figure 3.

To sum up, simulation experiment results show a superiority of the FCOCQPSO algorithms over other counterpart algorithms in terms of accuracy. In the COCQPSO and FCOCQPSO algorithms, the particle's search is guided by the position which may be not the global position but may lie in a promising search region so that the particles have a big chance to search this region and find out the global optimal solution.

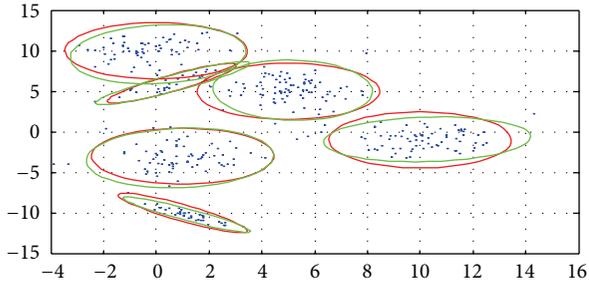


FIGURE 3: Clustering results of the FCOCQPSO algorithm.

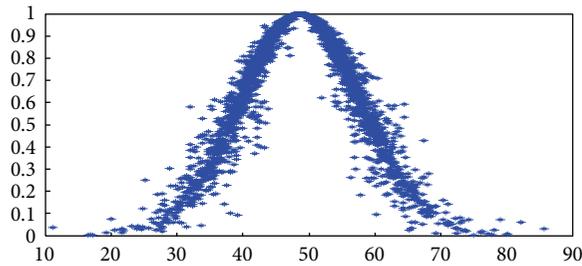


FIGURE 4: He = 1.89 clustering algorithm.

As a result, the COCQPSO and FCOCQPSO algorithms may have stronger global search ability and better overall performance than the other algorithms such as GA, PSO, ACO, and DE, particularly for the hard optimization problems. The experimental results show that the combinatorial clustering algorithms improve the precision and the result is stable.

3.2. Experimental Results. Fuzzy particle swarm optimization clustering algorithm is a novel method for solving real problems by using both the fuzzy rules and the characteristics of particle swarm optimization. In this paper, we successfully solve multiattribute groups' allocation problems by fuzzy combinatorial optimization clustering algorithm of cloud model and quantum-behaved particle swarm optimization.

Hyper Entropy H_e reflects the dispersion of the Cloud drops. The bigger the Hyper Entropy is, the bigger of its dispersion and the randomness of degree of membership, and so is the thickness of Cloud. Two figures contrast can be seen, big super entropy diagram cloud droplets of discrete degree are big, and the entropy of small cloud droplets is relatively concentrated. The clustering algorithm result of the cloud droplets experiments phenomenon is shown in Figure 4.

Fuzzy clustering algorithm of data to some extent, overcomes the inner shape: the distribution of the dependent on, and can correctly in different shape distribute data clustering, compute speed fast, and overcome the sensitivity to noise data and outliers, enhancing the algorithm robustness. According to fuzzy clustering, algorithm is sensitive to initial value, the shortcomings of easy to fall into local optimum, this paper proposes a method of fuzzy clustering based on particle swarm optimization. The design of fitness function according to the clustering criterion, using particle swarm optimization

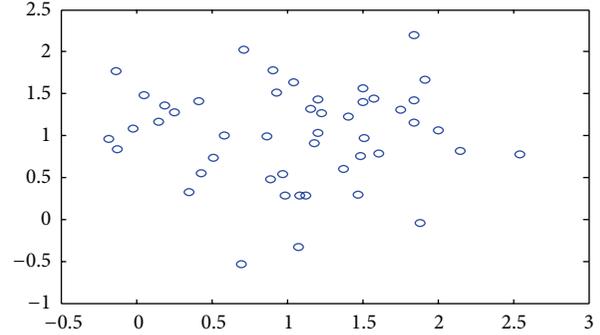


FIGURE 5: Combinatorial optimization clustering algorithm result.

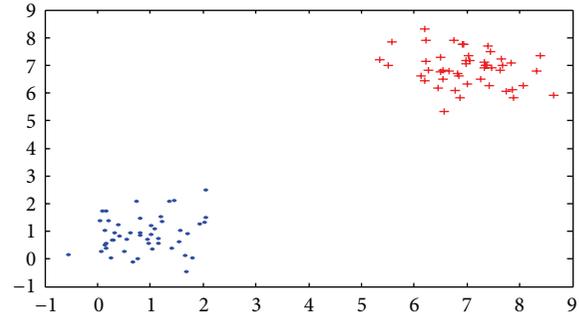


FIGURE 6: The corresponding dynamic clustering algorithm result.

algorithm to optimize clustering center, behind the simulation experiment proves the feasibility and effectiveness of the algorithm.

According to different thresholds of multiproject, multiattribute groups, we get the corresponding dynamic cluster result by algorithm of cloud model and quantum-behaved particle swarm optimization as follows in Figure 5.

According to different thresholds of multiproject, multiattribute groups, we get the corresponding dynamic cluster result by combinatorial optimization algorithm of cloud model and quantum-behaved particle swarm optimization clustering algorithm as follows in Figure 6.

According to different thresholds of multiproject, multiattribute groups, we get the corresponding 200-dynamic combinatorial cluster result as follows in Figure 7.

According to different thresholds of multiproject, multiattribute groups, we get the corresponding dynamic combinatorial cluster result by 200-quantum-behaved particle swarm optimization and cloud model clustering algorithm as follows in Figure 8.

According to different thresholds of multiproject, multiattribute groups we get the corresponding 1000-dynamic combinatorial cluster result as follows in Figure 9.

According to different thresholds of multiproject, multiattribute groups, we get the corresponding dynamic cluster result by 1000-quantum-behaved particle swarm optimization and cloud model combinatorial clustering algorithm as follows in Figure 10.

According to different thresholds of multiproject, multiattribute groups, we get the corresponding 10000-dynamic

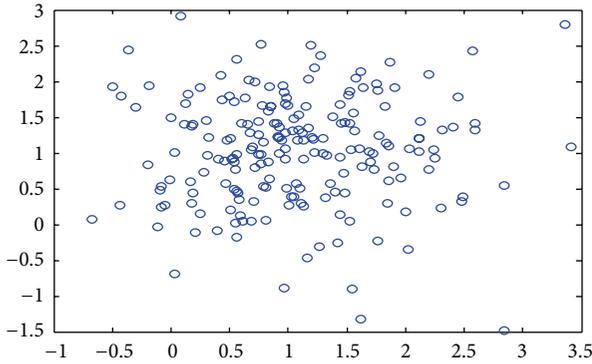


FIGURE 7: 200-valued dynamic clustering algorithm result.

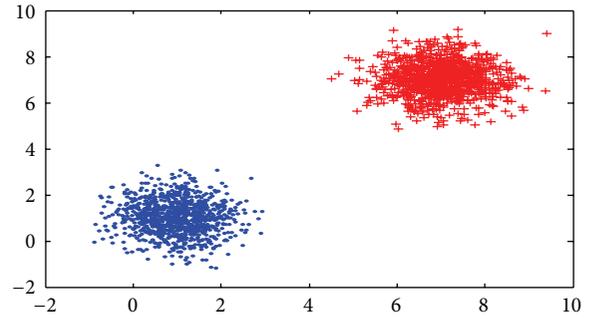


FIGURE 10: 1000-valued combinatorial clustering algorithm result.

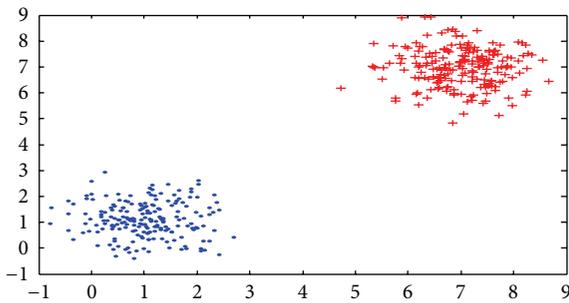


FIGURE 8: 200-valued combinatorial clustering algorithm result.

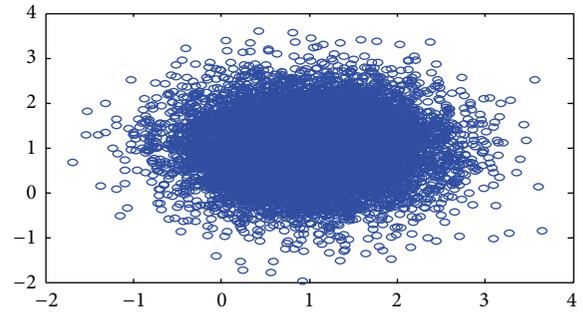


FIGURE 11: 10000-valued dynamic clustering algorithm result.

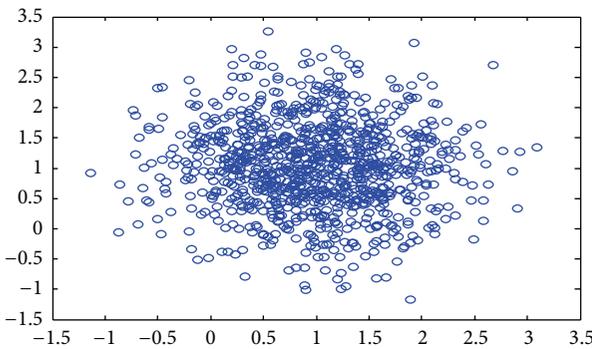


FIGURE 9: 200-valued dynamic clustering algorithm result.

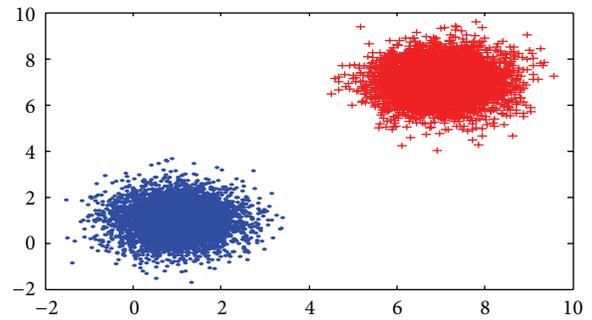


FIGURE 12: 10000-valued combinatorial clustering algorithm cluster result.

combinatorial cluster result by quantum-behaved particle swarm optimization clustering algorithm as follows in Figure 11.

According to different thresholds of multiproject, multi-attribute groups, we get the corresponding dynamic cluster result by 10000-quantum-behaved particle swarm optimization and cloud model combinatorial clustering algorithm as follows in Figure 12.

4. Conclusion

According to these studies, there are many different kinds of clustering algorithms which can be applied. In addition, it is critical to have a suitable algorithm for the effectiveness of

a clustering method. In this paper, we have developed the COCQPSO and FCOCQPSO algorithms. The results from various simulations using artificial data sets show that the proposed combinatorial clustering algorithms have better performance than that of the other clustering algorithms such as GA, PSO, and ACO. We demonstrated through several experiments that much better clustering results can be got by interval-valued combinatorial clustering algorithm than the corresponding dynamic clustering algorithm.

These experiments show that the combinatorial clustering algorithm based on similarity is better than conventional clustering method in terms of calculating complexity and clustering effect. In this work, we have given the full description of implementations and details of the combinatorial algorithms to improve its performance. We have tested the proposed algorithms in different synthetic and real clustering

problem, obtaining very good results that improve classical approaches. So, it is very important for us to research on combinatorial clustering algorithms. The combinatorial clustering algorithms are applied in several fields such as statistics, pattern recognition, machine learning, and data mining, to partition a given set of data or objects into clusters. It is also applied in a large variety of applications, for example, image segmentation, objects and character recognition, and document retrieval. The combinatorial clustering algorithms problems are the kind of important optimization problems in our real life which are difficult to find a satisfying solution within a limited time. In addition to the extension of the application domain, our future studies can investigate the combinatorial clustering algorithms to improve their solution quality.

Acknowledgments

This research is partially supported by the National Natural Science Foundation of China (Grant no. 70372039, Grant no. 70671037, and Grant no. 70971036) and the Doctoral Program Foundation of Institutions of Higher Education of China (Grant no. 20050532005).

References

- [1] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 316–323, 1999.
- [2] S. S. Khan and A. Ahmad, "Cluster center initialization algorithm for K-means clustering," *Pattern Recognition Letters*, vol. 25, no. 11, pp. 1293–1302, 2004.
- [3] S. Kotsiantis and P. Pintelas, "Recent advances in clustering: a brief survey," *WSEAS Transactions on Information Science and Applications*, vol. 1, no. 1, pp. 73–81, 2004.
- [4] J.-S. Lee and S. Olafsson, "Data clustering by minimizing disconnectivity," *Information Sciences*, vol. 181, no. 4, pp. 732–746, 2011.
- [5] H. Alizadeh, B. Minaei-Bidgoli, and S. K. Amirgholipour, "A new method for improving the performance of k nearest neighbor using clustering technique," *Journal of Convergence Information Technology*, vol. 4, no. 2, pp. 84–92, 2009.
- [6] R. Xu and D. Wunsch II, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.
- [7] H. Frigui and R. Krishnapuram, "A robust competitive clustering algorithm with applications in computer vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 450–465, 1999.
- [8] Y. Leung, J.-S. Zhang, and Z.-B. Xu, "Clustering by scale-space filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1396–1410, 2000.
- [9] E. Forgy, "Cluster analysis of multivariate data: efficiency vs. interpretability of classification," *Biometrics*, vol. 21, pp. 768–780, 1965.
- [10] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics, John Wiley & Sons, New York, NY, USA, 1990.
- [11] I. Gath and A. B. Geva, "Unsupervised optimal fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 773–780, 1989.
- [12] A. Lorette, X. Descombes, and J. Zerubia, "Fully unsupervised fuzzy clustering with entropy criterion," in *International Conference on Pattern Recognition*, vol. 3, pp. 3998–4001, 2000.
- [13] K.-Y. Huang, "A synergistic automatic clustering technique (SYNETRACT) for multispectral image analysis," *Photogrammetric Engineering and Remote Sensing*, vol. 68, no. 1, pp. 33–40, 2002.
- [14] G. H. Ball and D. J. Hall, "A clustering technique for summarizing multivariate data," *Behavioral Science*, vol. 12, no. 2, pp. 153–155, 1967.
- [15] D. Pelleg and A. Moore, "X-means: extending K-means with efficient estimation of the number of clusters," in *Proceedings of the 17th International Conference on Machine Learning*, pp. 727–734, Morgan Kaufmann, San Francisco, Calif, USA, 2000.
- [16] G. Hamerly and C. Elkan, "Learning the K in K-means," in *Proceedings of 7th Annual Conference on Neural Information Processing Systems (NIPS '03)*, pp. 281–288, 2003.
- [17] M. G. H. Omran, A. Salman, and A. P. Engelbrecht, "Dynamic clustering using particle swarm optimization with application in image segmentation," *Pattern Analysis and Applications*, vol. 8, no. 4, pp. 332–344, 2006.
- [18] S. Ouaifel, M. Batouche, and A. Taleb-Ahmed, "A new particle swarm optimization algorithm for dynamic image clustering," in *Proceedings of the 5th International Conference on Digital Information Management (ICDIM '10)*, pp. 546–551, July 2010.
- [19] C. S. Wallace and D. M. Boulton, "An information measure for classification," *Computer Journal*, vol. 11, no. 2, pp. 185–194, 1968.
- [20] P. N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, Pearson Education, 2006.
- [21] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: the algorithm GDBSCAN and its applications," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 169–194, 1998.
- [22] Y. Shi and R. Eberhart, "Modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 69–73, IEEE Press, Piscataway, NJ, USA, May 1998.
- [23] M. Clerc, "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization," in *Proceedings of IEEE Congress on Evolutionary Computation (ICEC '99)*, pp. 1951–1957, Washington, DC, USA, 1999.
- [24] F. Van den Bergh and A. P. Engelbrecht, "A new locally convergent particle swarm optimiser," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp. 94–99, October 2002.
- [25] B. Zhao, C. X. Guo, B. R. Bai, and Y. J. Cao, "An improved particle swarm optimization algorithm for unit commitment," *International Journal of Electrical Power and Energy Systems*, vol. 28, no. 7, pp. 482–490, 2006.
- [26] M. S. Arumugam and M. V. C. Rao, "On the improved performances of the particle swarm optimization algorithms with adaptive parameters, cross-over operators and root mean square (RMS) variants for computing optimal control of a class of hybrid systems," *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 324–336, 2008.
- [27] M. A. Montes de Oca, T. Stützle, M. Birattari, and M. Dorigo, "Frankenstein's PSO: a composite particle swarm optimization algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1120–1132, 2009.

- [28] J. Kennedy and R. C. Eberhart, "Discrete binary version of the particle swarm algorithm," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pp. 4104–4108, IEEE Service Center, Piscataway, NJ, USA, October 1997.
- [29] L. Wang and J. Yu, "Fault feature selection based on modified binary PSO with mutation and its application in chemical process fault diagnosis," in *Proceedings of the 1st International Conference on Advances in Natural Computation (ICNC '05)*, vol. 3612 of *Lecture Notes in Computer Science*, pp. 832–840, Changsha, China, August 2005.
- [30] L.-Y. Chuang, H.-W. Chang, C.-J. Tu, and C.-H. Yang, "Improved binary PSO for feature selection using gene expression data," *Computational Biology and Chemistry*, vol. 32, no. 1, pp. 29–37, 2008.
- [31] X. F. Xie, W. J. Zhang, and Z. L. Yang, "A dissipative particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computing (CEC '02)*, pp. 1666–1670, Honolulu, Hawaii, USA, 2002.
- [32] L. Wang, L. Zhang, and D.-Z. Zheng, "An effective hybrid genetic algorithm for flow shop scheduling with limited buffers," *Computers & Operations Research*, vol. 33, no. 10, pp. 2960–2971, 2006.
- [33] L. Nanni and A. Lumini, "Particle swarm optimization for prototype reduction," *Neurocomputing*, vol. 72, no. 4-6, pp. 1092–1097, 2009.
- [34] S. Kiranyaz, T. Ince, A. Yildirim, and M. Gabbouj, "Fractional particle swarm optimization in multidimensional search space," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 40, no. 2, pp. 298–319, 2010.
- [35] S. Ouadfel, M. Batouche, and A. Taleb-Ahmed, "A new particle swarm optimization algorithm for dynamic image clustering," in *Proceedings of the 5th International Conference on Digital Information Management (ICDIM '10)*, pp. 546–551, July 2010.
- [36] E. Martinez, M. M. Alvarez, and V. Trevino, "Compact cancer biomarkers discovery using a swarm intelligence feature selection algorithm," *Computational Biology and Chemistry*, vol. 34, no. 4, pp. 244–250, 2010.
- [37] Y.-T. Kao, E. Zahara, and I.-W. Kao, "A hybridized approach to data clustering," *Expert Systems with Applications*, vol. 34, no. 3, pp. 1754–1762, 2008.
- [38] R. J. Kuo, Y. J. Syu, Z.-Y. Chen, and F. C. Tien, "Integration of particle swarm optimization and genetic algorithm for dynamic clustering," *Information Sciences*, vol. 195, pp. 124–140, 2012.
- [39] H. Cheng, S. Yang, and J. Cao, "Dynamic genetic algorithms for the dynamic load balanced clustering problem in mobile ad hoc networks," *Expert Systems with Applications*, vol. 40, no. 4, pp. 1381–1392, 2013.
- [40] D. Chang, Y. Zhao, C. Zheng, and X. Zhang, "A genetic clustering algorithm using a message-based similarity measure," *Expert Systems with Applications*, vol. 39, no. 2, pp. 2194–2202, 2012.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

