IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 20, 2020, accepted May 3, 2020, date of publication May 6, 2020, date of current version May 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2992903

A Novel Clustering Algorithm Based on DPC and PSO

JIANGHUI CAI[1,2], HUILING WEI[1], HAIFENG YANG[1,2], AND XUJUN ZHAO[1]
[1]School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan 030024, China
[2]Shanxi Key Laboratory of Advanced Control and Equipment Intelligence, Taiyuan 030024, China

Corresponding author: Haifeng Yang (hfyang@tyust.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant U1731126 and Grant U1931209, and in part by the Shanxi Province Key Research and Development Program under Grant 201803D121059 and Grant 201903D121116.

ABSTRACT Analyzing the fast search and find of density peaks clustering (DPC) algorithm, we find that the cluster centers cannot be determined automatically and that the selected cluster centers may fall into a local optimum and the random selection of the parameter cut-off distance d_c value. To overcome these problems, a novel clustering algorithm based on DPC & PSO (PDPC) is proposed. Particle swarm optimization (PSO) is introduced because of its simple concept and strong global search ability, which can find the optimal solution in relatively few iterations. First, to solve the effect of the selection of the parameter d_c on the calculation density and the clustering results, this paper proposes a method to calculate that parameter. Second, a new fitness criterion function is proposed that iteratively searches K global optimal solutions through the PSO algorithm, that is, the initial cluster centers. Third, each sample is assigned to K initial center points according to the minimum distance principle. Finally, we update the cluster centers and redistribute the remaining objects to the clusters closest to the cluster centers. Furthermore, the effectiveness of the proposed algorithm is verified on nine typical benchmark data sets. The experimental results show that the PDPC can effectively solve the problem of cluster center selection in the DPC algorithm, avoiding the subjectivity of the manual selection process and overcoming the influence of the parameter d_c. Compared with the other six algorithms, the PDPC algorithm has a stronger global search ability, higher stability and a better clustering effect.

INDEX TERMS Clustering, density peak, particle swarm optimization, fitness function.

I. INTRODUCTION

In 2014, Alex Rodriguez *et al.* proposed a new algorithm, the clustering by fast search and find of density peaks (DPC) algorithm [1]. Because DPC has the advantages of simple algorithmic principles, easy implementation and the ability to quickly find clusters of arbitrary shapes, many researchers have studied and applied it since the algorithm was published. The advantages of the DPC clustering algorithm are outstanding, but its disadvantages are also obvious. The DPC algorithm has the following disadvantages:

(1) It is difficult to determine the value of the parameter cut-off distance d_c, which mainly depends on subjective experience and lacks a certain basis for selection;

(2) The selection of cluster centers requires human participation and easily falls into a local optimum, which cannot guarantee the objectivity and accuracy of the clustering results.

The associate editor coordinating the review of this manuscript and approving it for publication was Anandakumar Haldorai.

For the shortcomings of the DPC algorithm, the particle swarm optimization (PSO) algorithm is introduced; it has a simple concept and a strong global search ability that can find the optimal solution in a relatively small number of iterations [2]. In this paper, a new fitness function based on the DPC algorithm is proposed, and a method for calculating the parameter d_c is proposed. On these bases, a novel clustering algorithm based on DPC & PSO (PDPC) is proposed. The effectiveness and advantages of the PDPC algorithm are verified by experiments on typical benchmark data sets. Experiments show that our algorithm can effectively solve the problem of cluster center selection in the DPC algorithm, avoiding the subjectivity of the manual selection process and overcoming the influence of the parameter d_c.

A. MOTIVATIONS

The motivation of this study can be summarized as follows:

- There is a parameter cut-off distance d_c in the DPC algorithm that is selected according to an empirical value,

which may affect the clustering results. Therefore, it is necessary to propose a new method of calculating d_c.
- The cluster centers selected by the DPC algorithm are likely to fall into a local optimum. This problem also impacts the clustering results and needs to be solved.
- Since the DPC algorithm visually identifies the cluster centers on the decision diagram (See Section III.A.2)), it may directly affect the clustering results. Therefore, it is necessary to overcome the influence of human factors and achieve the automatic identification of cluster centers.

Motivation 1: For the calculated density formula in the DPC algorithm, there is a parameter cut-off distance d_c, which is 1% to 2% of the size of the data set [1]. This empirically chosen value is uncertain and unreliable, which may affect the calculation of density and in turn affect the clustering results. Therefore, a new method for calculating d_c is proposed based on the Gaussian distance.

Motivation 2: The deficiencies of the DPC clustering algorithm must be overcome; its selected cluster centers may fall into a local optimum, and its initial centers may be located in the same cluster or may not be found. These issues can affect the clustering results. Considering the above problem, this paper introduces an intelligent optimization algorithm for clustering analysis.

Motivation 3: The DPC algorithm selects cluster centers visually and intuitively on the decision diagram. Some of the improved clustering methods use the same strategy, such as DP_K-medoids [3] and DPNM_K-medoids [3]. These methods show good performance on different data sets. However, there are human factors in the process of selecting cluster centers that may directly affect the clustering results. This insufficiency motivates us to propose a method that automatically identifies the cluster centers in the data set.

B. CONTRIBUTIONS

Inspired by the above motivations, the PDPC clustering algorithm is proposed. First, to solve the influence of the parameter d_c, this paper proposes a method to calculate the parameter d_c. Second, a new fitness criterion function based on the DPC algorithm is proposed, and it iteratively searches K initial cluster centers by the PSO algorithm. Then, each sample is assigned to K initial center points according to the minimum distance principle. Finally, we update the cluster centers and redistribute the remaining objects to the clusters closest to the cluster centers. The process iterates until the reallocation of objects no longer changes in any cluster or reaches the termination condition of iteration. The experimental results show that compared to the other methods, the PDPC algorithm has a stronger global search ability, higher stability and a better clustering effect on the benchmark data sets.

The main contributions of this work are summarized as follows:
- To solve the influence of the parameter cut-off distance d_c on the clustering results, a method of calculating the parameter d_c is proposed. First, the Gaussian distance between the data points is calculated. Second, the maximum and minimum Gaussian distances are found. Finally, the parameter d_c is proposed based on the mean value of the maximum and minimum Gaussian distances.
- Aiming at the problem that the cluster centers selected by the DPC algorithm easily fall into a local optimum, the PSO intelligent optimization algorithm is introduced for clustering analysis, and the global search ability of PSO can be used to find K approximate optimal solutions. We use the optimal solutions as the initial cluster centers. The PDPC algorithm achieves the purpose of automatically selecting the cluster centers, avoids the subjectivity of the manual selection process.
- Literature [1] proposed that the cluster center has the characteristics of high density ρ_i and long distance δ_i. According to this feature in the DPC algorithm, a new fitness function is proposed. Setting the fitness function is a key step in solving the optimization problem, and the design of the fitness function should be as simple as possible. Therefore, we use the inverse of the product of density and distance as the fitness function.
- We use multiple typical benchmark data sets to test the performance of the PDPC algorithm, and use three well-known evaluation cluster quality indicators (the accuracy, the precision and the recall) to evaluate the clustering results. The comparison experiments with other six algorithms show the effectiveness and correctness of the proposed clustering algorithm.

C. ROADMAP

The rest of this paper is organized as follows. Section II summarizes the related work relevant to this work. Section III gives the theoretical basis and some related concepts. In Section IV, a novel clustering algorithm based on DPC & PSO (PDPC) is proposed, and the algorithm is introduced in detail. Section V analyzes the experimental results on typical benchmark data sets, then analyzes the characteristics of the proposed algorithm. The six improved clustering algorithms (DP_K-medoids [3], DPNM_K-medoids [3], Improve K-means [4], K-means [5], Hybrid PSO and K-means [6] and DPC [1]) were selected for comparison. And finally, a summary of this work is given in Section VI.

II. RELATED WORKS

Clustering is a dynamic research field in data mining. It is also an important unsupervised learning technique in machine learning. Clustering is the process of grouping a set of data objects into multiple groups or clusters so that objects within a cluster have high similarity but are very dissimilar to objects in other clusters. Clustering as a data mining tool has its roots in many application areas such as biology, security, business intelligence, pattern recognition, Web search [7]–[9], trajectory clustering [10], [11] and astronomy [12]–[14].

Traditional approaches in clustering can be broadly categorized into partition-based, hierarchical-based, density-based, model-based, grid-based and soft computing methods [15]. Partitioning methods such as K-means [5] and K-medoids [16] relocate points by moving them from one category to another according to distance. These methods always need the number of clusters to be set in advance, and they are sensitive to initial cluster centers. For the problem of cluster center selection, [17] proposed a novel algorithm for initial cluster center selection, which uses MNN (M nearest neighbors), density and distance to determine the initial cluster centers. The authors show that the method obtains high-quality initial cluster centers. Hierarchical methods [18] structure categories by recursively classifying the data in either a top-down or bottom-up fashion. Density-based methods assume that the points that belong to each cluster are drawn from a specific probability distribution [19]. Clusters of arbitrary shape can be discovered by density-based methods such as DBSCAN [20] and Denclue [21]. Model-based methods [22] can obtain the clustering results by optimizing the fit between the given data and certain mathematical models. Reference [23] developed a simple clustering model inspired by the way in which the human visual system associates patterns spatially. And the approach is based on Cellular Neural Networks (CNNs), similar to the biological model. In grid-based methods, the data space is divided into a finite number of unit grid structures [24]. Therefore, such methods have a high processing speed. The evolutionary approaches that belong to the soft computing method [25], [26] are also used to deal with clustering problems. These algorithms such as the genetic algorithm (GA), artificial bee colony (ABC) and PSO [27], [28] can obtain satisfactory results by optimizing the objective function.

In 2014, there was a large breakthrough in density-based clustering approaches. Rodriguez and Laio proposed the DPC algorithm [1]. DPC is based on the concept that cluster centers are characterized by a higher density than that of their neighbors and by a relatively larger distance from points with higher densities. This algorithm uses these two features to obtain a scatter graph called a decision diagram, which is used to visually judge the potential cluster centers. Finally, each remaining point is assigned to a cluster according to its nearest neighbor of higher density. The algorithm is simple, and the clustering results can be completed in one step without iteration. However, the algorithm has human factors when selecting the cluster centers, which may directly affect the clustering results.

In response to the problems of the DPC algorithm, researchers have proposed many different algorithms. Reference [3] used DPC to optimize the initial medoids of the K-medoids clustering algorithm. To obtain better clustering, a new measure function is proposed as the ratio of the intra-distance of clusters to the inter-distance between clusters. The authors proposed two new K-medoids clustering algorithms: the DP_K-medoids algorithm and the DPNM_K-medoids algorithm. In [29], the new clustering algorithm, by finding density peaks based on Chebyshev's inequality (CDP), can obtain a judgment index by screening density and distance, which are normalized. The points whose judgment indexes are above the upper bound based on Chebyshev's inequality will be selected as the cluster centers. Then, the remaining points are assigned by their nearest neighbor of higher density. Inspired by the visual selection rule of DPC, reference [30] proposed a judgment index that approximately follows the generalized extreme value (GEV) distribution, and each cluster center's judgment index is much higher. Hence, it is reasonable that points are selected as cluster centers if their judgment indexes are larger than the upper quantile of GEV. This proposed method is called density peaks clustering based on generalized extreme value distribution (DPC-GEV).

Reference [31] introduced the idea of K-nearest neighbors (KNN) and principal component analysis (PCA) into DPC to improve the performance of the DPC algorithm. Reference [32] used the technique of K-nearest neighbors and fuzzy weighted K-nearest neighbors to overcome the deficiencies of the DPC algorithm. Reference [33] enhanced the DPC to make it suitable for hyperspectral band selection. The proposed approach is named the enhanced FDPC (E-FDPC), and it can use an exponential-based learning rule to adjust different numbers of cut-off thresholds and determine cluster centers automatically. Reference [34] presented a density peak based hierarchical clustering method (DenPEHC), which directly generates clusters on each possible clustering layer, and introduced a grid granulation framework to enable the clustering of large-scale and high-dimensional (LSHD) data sets.

To solve the shortcomings of initial cluster center selection of the clustering algorithm and being easily falling into a local optimum, some researchers try to use the intelligent optimization algorithm for clustering analysis and the clustering problem as the solution to the optimization problem. Among these strategies, the PSO algorithm is very popular due to its flexibility, robustness, discreteness and self-organization. PSO clustering focuses on solving clustering problems by using group behavior. Therefore, the global search ability of the PSO algorithm is used to find an approximate optimal solution.

PSO is a group intelligent optimization method proposed by Kennedy and Eberhart in 1995 [2]. It is derived from bird predation behavior research and is an iteration-based optimization tool. The system is initialized to a set of random solutions that search for the optimal value by iteration. The PSO algorithm is simple, easy to implement, and does not have many parameters to adjust. It has been widely used in function optimization, neural network training, and fuzzy system control.

In recent years, the PSO optimization algorithm and improved clustering methods for PSO have been studied and applied. Reference [35] proposed a PSO clustering algorithm based on different learning methods. The author proposed two improved fitness functions, which greatly improved the

classification accuracy of the clustering algorithm. Reference [36] proposed an effective PSO clustering method. In view of the shortcomings of PSO when applied to large data sets, particles on the boundary of the search space cannot be moved to a better position, and a mapping method is proposed. Reference [37] proposed an approach for document clustering using the particle swarm optimization method. This method is applied before K-means for finding optimal points in the search space, and these points are used as initial cluster centroids for the K-means algorithm to find the final clusters of documents. Reference [38] used K-medoids clustering to provide a fitness metric for the particle swarm optimization procedure to distinguish between active and inactive pixels in a scheme. Reference [6] proposed two new approaches to using PSO to cluster data, and it is shown how PSO can be used to find the centroids of a user-specified number of clusters. The algorithm is then extended to use K-means clustering to seed the initial swarm. This second algorithm primarily uses PSO to refine the clusters formed by K-means.

Reference [39] proposed a new method named MSSE-PSO (master-slave swarm shuffling evolution algorithm based on particle swarm optimization). MSSE-PSO combines the strengths of the particle swarm optimization, competitive evolution and sub-swarm shuffling, which greatly enhances survivability by sharing the information gained independently by each swarm. Besides, MSSE-PSO adopts the hierarchical idea, by which the master swarm guides the whole group to the optimal direction to control the balance between exploration and exploitation.

In summary, the main problems of the DPC algorithm are as follows: (1) there is a parameter cut-off distance d_c in the local density calculation formula that is selected according to an empirical value, but this value is unreliable and may have an impact on the clustering results; (2) the selected cluster centers may fall into a local optimum, and the initial centers may be located in the same cluster or not be found; and (3) selecting the cluster center has human factors, which may directly affect the clustering results.

In contrast, the main advantages of the PDPC algorithm are as follows: (1) a new method for calculating the parameter cut-off distance d_c is proposed. When calculating the density of data points, d_c does not need to be randomly selected according to the empirical value; (2) this study introduces the PSO intelligent optimization algorithm because it has strong global search ability, which prevents the cluster centers selected by the DPC algorithm from falling into a local optimum; (3) a new fitness function is proposed based on the DPC algorithm, which iteratively searches K global optimal solutions by the PSO algorithm, that is, the initial cluster centers. This approach overcomes the influence of human factors and realizes the purpose of automatically identifying cluster centers; (4) compared with the other six algorithms, the proposed algorithm improves clustering performance and computational efficiency and has a good clustering effect.

III. THEORETICAL BASIS

In this section, we introduce two classic algorithms: the density peak clustering algorithm and the particle swarm optimization algorithm.

A. DENSITY PEAK CLUSTERING ALGORITHM

Reference [1] describes a new density-based clustering algorithm, the density peak clustering algorithm, which uses novel ideas and is simple and clear. The premise of the algorithm is that the cluster center is surrounded by points whose densities are smaller than itself and that the center has a larger distance from other high-density points. The algorithm defines two parameters for each data point i: one is the density ρ_i of the data point, and the other is the distance δ_i from the data point to a local high-density point. It uses these two features to obtain a scatter graph called the decision diagram, selects the points where ρ_i and δ_i are both large as the cluster centers on the decision diagram, and assigns the remaining points to the cluster of the high-density point closest to them.

1) DENSITY AND DISTANCE CALCULATION

We define a density for each data point i based on the distance between the data points.

$$\rho_i = \sum_j \chi\left(d_{ij} - d_c\right) \tag{1}$$

where if we assume $x = d_{ij} - d_c$, then $\chi(x) = \begin{cases} 1 & x < 0 \\ 0 & x \geqslant 0 \end{cases}$. i and j are different data points; d_{ij} is the Euclidean distance between data points; and d_c is the cut-off distance and is a hyperparameter. d_c is 1% to 2% of the total number of points in the data set. ρ_i is equivalent to i as the center, d_c is the radius and the number of points in this range.

The minimum distance from each data point to a high local density point is

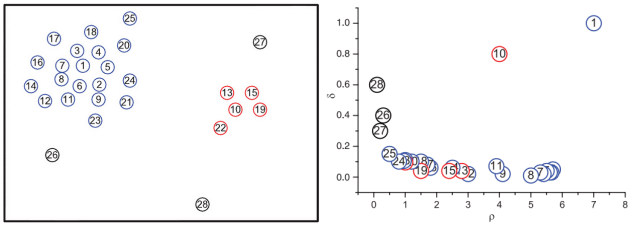
$$\delta_i = \min_{j:\rho_j > \rho_i} \left(d_{ij}\right) \tag{2}$$

For the distance between global high-density points, the opposite is true; we take the maximum distance between the two highest-density samples. Note that δ_i is much larger than the typical nearest-neighbor distance only for points that are local or global maxima in the density. Thus, cluster centers are recognized as points for which the value of δ_i is unusually large.

2) DECISION DIAGRAM

A decision diagram is a novel method for identifying the cluster centers of the data set proposed in [1]. This method determines cluster centers by constructing a decision diagram of the local density ρ and distance δ of each sample point in the data set. When both the density value ρ and the distance value δ of the point are large, the point may be a cluster center point.

Based on the values of the local density and distance of sample points, cluster centers can be selected intuitively.



(a) Point distribution. (b) Decision diagram.

FIGURE 1. (a) Point distribution in two dimensions. (b) Decision diagram for each sample point.

Reference [1] uses the data set shown in Figure 1 to illustrate the process of selecting cluster centers in the decision diagram. There are 28 sample points arranged in descending order of density in Figure 1(a), and the sample points can be divided into two clusters. Figure 1(b) is the decision diagram drawn with ρ as the horizontal axis and δ as the vertical axis. It can be seen that sample points 1 and 10 are located at the upper right corner of the decision diagram. The local density and distance are both large, so these two points are the cluster centers. Points 26, 27, and 28 have a relatively high δ and a low ρ because they are isolated, and they can be considered outliers.

3) CLUSTERING PROCESS

Points with relatively large local density ρ_i and distance δ_i are considered the cluster centers. These points are inherently dense and surrounded by neighbors with a density that is relatively large and at a relatively large distance from other higher-density points. Such points are selected as cluster centers. After the cluster centers are determined, the remaining other points are attributed to the cluster of the highest-density point closest to them, and the final clustering result is obtained.

The DPC algorithm has the advantages of simple principles, easy implementation and can quickly find clusters of any shape. However, in the clustering process, the decision diagram plays a decisive role in determining cluster centers using qualitative selection instead of quantitative analysis. Selecting the data point has the characteristics that both ρ_i and δ_i are larger, which is subjective. Sometimes, for the same decision diagram, different people may make different choices. As a result, the selected cluster centers may be located in the same cluster or may not be found.

B. PARTICLE SWARM OPTIMIZATION ALGORITHM

PSO is a common evolutionary algorithm based on the concepts of group and fitness [2]. An individual of the particle swarm represents a possible solution to the problem. Starting from a random solution, the PSO algorithm uses iteration to find a possible optimal solution and uses fitness to determine the quality of the solution. The algorithm randomly initializes a group of particles and then iterates to find the optimal solution. Each iteration of the particle tracks the individual extremum and the global extremum to dynamically update its velocity and position.

1) PARTICLE VELOCITY AND POSITION

In a D-dimensional target search space, the PSO algorithm refers to individuals as ''particles''. The position of each particle represents a solution to the problem. A particle constantly adjusts its position x to search for a new solution [2]. The total number of particles is set to m, where the position of the i-th particle in the d-th dimension is x_{id}, the flying velocity is v_{id}, the current optimal position the particle has searched is P_{id}, and the current optimal position of the particle swarm as a whole is P_{gd}. The update formulas for velocity and position are as follows:

$$v_{id}(t+1) = w \times v_{id}(t) + c_1 r_1 [P_{id}(t) - x_{id}(t)] + c_2 r_2 \left[P_{gd}(t) - x_{id}(t)\right] \quad (3)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (4)$$

where $i = 1, 2, \ldots, m$; $d = 1, 2, \ldots, D$; w is the inertia weight; c_1 and c_2 are learning factors, which are nonnegative constants and usually take $c_1 = c_2 = 2$; r_1 and r_2 are random numbers in (0, 1); P_{id} is an individual extremum; and P_{gd} is the global extremum.

In the velocity update formula (3), the first term is the product of the inertia weight and the particle's current velocity, which represents the degree of trust of the particle in its current movement and is based on the inertial motion of the original velocity; the second term indicates the situation of self-awareness, which is the particle's judgment of its own history; and the third item represents social awareness, which is the mutual cooperation and information sharing of particles in the group.

2) ALGORITHM STEP

The flowchart of the PSO algorithm is shown in Figure 2.

The PSO algorithm is initialized as a group of random particles (random solution), and then, the optimal solution is found through iteration. In each iteration, the particle updates itself by tracking two extreme values, which are the individual extreme value P_{id} and the global extreme value P_{gd}. All particles have a fitness value determined by the optimized function, and each particle also has the velocity that determines the direction and distance of the flight. Then, the particles follow the current optimal particle and search in the solution space until the maximum number of iterations is reached; otherwise, execution continues.

IV. PDPC CLUSTERING ALGORITHM

In this section, based on the advantages of the PSO algorithm, a novel clustering algorithm based on DPC & PSO (PDPC) is proposed. The rest of this section is organized as follows. In Section IV.A, a method of calculating the parameter d_c is proposed to solve the problem of randomly selecting d_c according to empirical values in the DPC algorithm. In Section IV.B, a fitness function is proposed based on

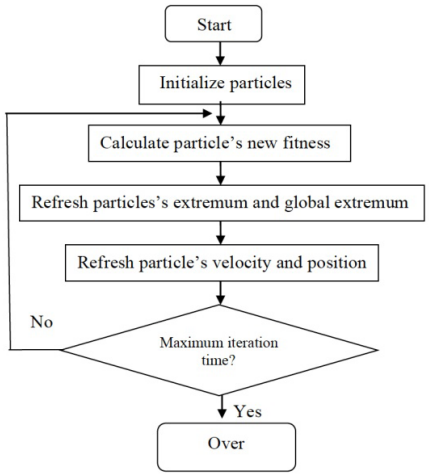


FIGURE 2. The PSO algorithm flowchart.

the DPC algorithm. Setting the fitness function is a crucial step in solving the optimization problem. In Section IV.C, the parameters of the velocity update formula are redefined in the PSO algorithm. In Section IV.D, the proposed PDPC algorithm is introduced in detail and the algorithm steps are given. Finally, in Section IV.E, the time complexity of PDPC and comparison algorithms is analyzed.

A. SETTING THE PARAMETER

In the density peak clustering algorithm proposed in [1], the parameter cut-off distance d_c is difficult to determine; it mainly relies on subjective experience, generally has approximately 1% to 2% of the size of the data set, and lacks a definite selection basis. Therefore, the impact on the clustering results is great.

To solve the influence of the parameter d_c value on the clustering results, a new method for calculating d_c is proposed in this paper. The specific steps are as follows:

1: Calculate the Gaussian distance between data points;

$$Distance = 1 - e^{-\frac{d_{ij}^2}{2}} \tag{5}$$

2: Get the maximum and minimum values of the Gaussian distance, expressed by $max_{Distance}$ and $min_{Distance}$ respectively;

3: Take the mean value of $max_{Distance}$ and $min_{Distance}$ to obtain the value of d_c.

$$d_c = \frac{max_{Distance} + min_{Distance}}{2} \tag{6}$$

B. FITNESS FUNCTION

The pros and cons of the current position of the particle are measured by the fitness function, and the fitness function obtains the corresponding fitness value of the particle. We hope that the algorithm will automatically recognize cluster centers. The probability that a particle is ultimately identified as a cluster center is proportional to the product of density ρ_i and distance δ_i [40].

Here, we use the density formula as follows.

$$\rho_i = \sum_j e^{-\left(\frac{d_{ij}}{d_c}\right)^2} \tag{7}$$

From the analysis of equations (1) and (7), we know that (1) calculates a discrete value and that (7) calculates a continuous value. In comparison, the probability of conflict in (7) is small; that is, the probability that different data points have the same local density will be small. The value of d_c in (7) can be calculated by (6), and is no longer selected according to the empirical value. Therefore, the local density calculated by (7) is better. In view of this consideration, we design the fitness function as follows.

$$f\left(d_{ij}\right) = \frac{1}{\rho \times \delta} = \frac{1}{\sum_j e^{-\left(\frac{d_{ij}}{d_c}\right)^2} \times \lim\limits_{j:\rho_j > \rho_i} \left(d_{ij}\right)} \tag{8}$$

where i and j denote different particles, d_{ij} is the Euclidean distance between particles, and d_c is the cut-off distance mentioned in Section IV.A. For a general particle, $\delta = min\left\{d_{ij}\right\}$, however, for a particle with the largest density value, $\delta = max\left\{d_{ij}\right\}$. The smaller the value of $f\ (d)$ is, the greater the probability that the particle becomes a cluster center point. If $f\ (d)_n < f\ (d)_{n-1}$, the optimal position needs to be updated.

We set a convergence condition as the termination condition of iteration for the PSO algorithm to ensure the performance of the proposed algorithm. The convergence formula is as follows:

$$|f(d)_n - f(d)_{n-1}| \leq \varepsilon, \quad n \geq 2 \tag{9}$$

where ε is the convergence parameter and n is the number of iterations. When a certain number of iterations is reached, the difference between $f\ (d)_n$ and $f\ (d)_{n-1}$ is very small, and it is determined that the particle swarm algorithm has reached convergence.

C. SELECTION OF PARAMETERS IN VELOCITY UPDATE FORMULA

The velocity and position update formulas use (3) and (4) in the PDPC algorithm proposed in this paper. However, we redefine the parameters in the velocity update formula (3). The selected values for the parameters are as follows:

1: w is the inertia weight. Shi and Eberhart added the inertia weight to make the particles gradually slow down, which also affects exploration and exploitation [41]. High values of w prevents particles from slowing down more than lower values do, which is good for exploring the search space. Lower values of w allow particles to exploit a good region without overshooting positions too much. It was found that linearly decreasing the inertia weight from 0.9 to 0.4 produces good results [41]. The inertia weight

is decreased according to

$$w = w_{max} - \frac{t \times (w_{max} - w_{min})}{t_{max}} \quad (10)$$

where w_{max} and w_{min} are the initial and final values of the inertia weight, respectively, t is the number of iterations and t_{max} is the maximum number of iterations.

2: c_1 and c_2 are learning factors. Ratnaweera *et al.* modified PSO by changing the acceleration coefficients over time [42]. This variant is called time-varying acceleration coefficient PSO (TVAC-PSO). The cognitive acceleration (c_1) starts with a higher value than c_2 and linearly decreases, while the social acceleration (c_2) starts with a lower value and linearly increases. The ranges of values are the following: c_1 decreases linearly from 2.5 to 0, and c_2 increases linearly from 0 to 2.5. The linear change is performed using

$$c_1\,(t+1) = \left(c_{1,final} - c_{1,initial}\right) \times \frac{t}{t_{max}} + c_{1,final} \quad (11)$$

$$c_2\,(t+1) = \left(c_{2,final} - c_{2,initial}\right) \times \frac{t}{t_{max}} + c_{2,final} \quad (12)$$

where c_{final} and $c_{initial}$ are the final and initial values of the acceleration coefficient, respectively, t is the number of iterations and t_{max} is the maximum number of iterations. Note that c_1 and c_2 are now functions of time.

3: r_1 and r_2 are random numbers obeying the $U(0, 1)$ distribution.

D. PDPC ALGORITHM

This paper proposes the PDPC clustering algorithm, which is mainly developed to address the defects of the DPC algorithm. Its main contribution is to introduce the PSO intelligent optimization algorithm for clustering analysis.

1) ALGORITHM IDEA

The shortcomings of the DPC algorithm urgently need to be addressed. In this paper, first, to mitigate the influence of the selection parameter d_c on the clustering results, a method for calculating d_c is proposed that uses the mean value of the maximum and minimum values of the Gaussian distance. Second, the PDPC algorithm introduces the PSO intelligent optimization algorithm for clustering analysis. Based on the density and distance of the data points, a new fitness function is proposed. The global search ability of the PSO algorithm is used to find the K-approximate optimal solutions. Then, each sample is assigned to K initial center points according to the minimum distance principle. Finally, we update the cluster centers and redistribute the remaining objects to the clusters closest to the cluster centers. The process iterates until the reallocation of objects no longer changes in any cluster or reaches the termination condition of iteration. The experimental results show that the PDPC algorithm has a strong global search ability, high stability and a good clustering effect.

2) ALGORITHM DESCRIPTION

According to the above description, the specific steps of the PDPC algorithm are shown in Algorithm 1.

Algorithm 1 PDPC: A Novel Clustering Algorithm Based on DPC & PSO

Input: A data set containing n objects, clusters number K.
Output: K cluster center points, the final clustering results.
Begin:

1: Initialization; set the number of particles m and convergence condition.
2: Calculate the fitness function value of each particle according to (8).
3: P_{id} and P_{gd} are updated by comparing the fitness of each particle with the fitness of the best position P_{id} and the fitness of the optimal position P_{gd}.
4: Update the velocity and position of each particle using (3) and (4).
5: Verify that the final condition is met. If the termination condition of iteration is satisfied, the iteration is stopped; otherwise, Step 2 is performed.
6: Consider the K optimal points given by the PSO algorithm as the initial cluster centers.
7: The distance of each data point to each cluster centers is calculated.
8: According to the current position, each sample is assigned to K initial cluster centers according to the principle of minimum distance.
9: Based on the new classification, calculate the new cluster center using (13) in each cluster.
10: Perform an iterative process of assigning the remaining data points and updating cluster centers. Stop iteration when the clustering results remain the same or the termination condition of iteration is reached.
11: Output the final clustering results.

End

In order to overcome the defects of DPC algorithm, this paper proposes PDPC clustering algorithm. First, a new fitness function based on DPC algorithm is proposed. Second, the K optimal solutions are searched by the PSO method as the initial cluster centers. Finally, perform the iterative process and create the clusters. Steps 1-6 are the process of the PSO optimization, where the fitness function used in step 2 is based on the DPC algorithm, and steps 7-11 are clustering processes. Step 9 determines new centers using formula (13), it computes the new mean using the objects assigned to the cluster. All the objects are then reassigned using the updated means as the new cluster centers. The iterations continue until the assignment is stable, that is, the clusters formed in the current round are the same as those formed in the previous round, or the termination condition of iteration is reached.

$$Center_i = \frac{1}{n_i} \sum_{\forall x_i \in C_i} x_i \quad (13)$$

TABLE 1. Summary of the time complexity for each of the seven algorithms.

	DP_K-medoids	DPNM_K-medoids	Improved K-means	K-means	Hybrid PSO and K-means	DPC	**PDPC**
Distance matrix	$O(n^2)$	$O(n^2)$	$O(n^2)$	—	$O(n^2)$	$O(n^2)$	$\boldsymbol{O(n^2)}$
Calculating object density	$O(n^2)$	$O(n^2)$	$O(n^2)$	—	—	$O(n^2)$	$\boldsymbol{O(n^2)}$
Selecting the initial centers	$O(nlogn)$	$O(nlogn)$	$O(n^2)$	$O(1)$	$O(n)$	$O(nlogn)$	$\boldsymbol{O(tn)}$
Remaining sample allocation	$O(tnK)$	$O(tnK)$	$O(tnK)$	$O(tnK)$	$O(tnK)$	$O(n)$	$\boldsymbol{O(tnK)}$
Total complexity(magnitude)	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(n)$	$O(n^2)$	$O(n^2)$	$\boldsymbol{O(n^2)}$

[1]Note: — indicates no.

where $Center_i$ is a new center, x_i is the data point that belongs to cluster C_i, and n_i is the number of data points that belong to cluster C_i.

The particle swarm optimization algorithm first divides the particle swarm into several ''subgroups'' according to the clustering algorithm and finds the optimal position of each ''subgroup''; then, the particles in the particle swarm update their velocity and position values based on their individual extremum and the optimal position in each ''subgroup''. By clustering the particle swarm, the algorithm exchanges information between the particles and finds the optimal solution in the iterative process, which makes the global convergence of the algorithm stronger.

E. COMPLEXITY ANALYSIS

In this subsection, the calculation costs are analyzed for PDPC, DP_K-medoids [3], DPNM_K-medoids [3], Improved K-means [4], K-means [5], Hybrid PSO and K-means [6] and DPC [1], as shown in Table 1. However, each method differs in its calculation complexity. In addition, the total cluster complexity includes updating the centers and calculating the distance between each pair of objects.

A data set containing n objects, for all algorithms except K-means, the time complexity of calculating the distance matrix is $O(n^2)$. The K-means algorithm does not need to calculate the distance matrix and density between data points during the implementation process. The time complexity of the algorithm for calculating the distance from each sample point to the ''cluster center'' is $O(n)$.

For all algorithms except K-means and Hybrid PSO and K-means algorithms, the time complexity for calculating all sample densities is $O(n^2)$. Hybrid PSO and K-means clustering algorithm first executes the K-means once. The result of the K-means algorithm is then used as one of the particles, while the rest of the swarm is initialized randomly. Therefore, the algorithm does not need to calculate the density between data points, and the total time complexity is $O(n^2)$.

The time complexity of the cluster center iterative process of the six algorithms, except DPC, is $O(tnK)$, where t is the number of iterations of the algorithm, n is the number of data points, and K is the number of clusters. After obtaining the initial cluster centers, the DPC algorithm assigns each remaining point to the cluster of the nearest neighbor samples whose density is larger than that of the sample, so the sample allocation time complexity is $O(n)$. Therefore, the time complexity of the DPC algorithm in calculating all objects is $O(n^2)$ without accounting for the process of determining the cluster centers artificially [32].

For the PDPC algorithm, the number of particles in each iteration does not change. Assume that the number of particles in the $i-th$ iteration is n_i, where $i = 1, 2, \cdots, t$, t represents the maximum number of iterations, so $n_1 = n_2 = \cdots = n_t = n$. The complexity in calculating the distance matrix is $O(n^2)$, and the time complexity for calculating all sample densities is $O(n^2)$. It can be concluded that the time complexity of selecting the initial center using the PSO algorithm is $O(tn)$. In the center-updating phase, the K centers updating complexity is $O(tnK)$. From this, we can determine that the total time complexity of the PDPC algorithm is $O(n^2)$.

The complexity of each of the seven algorithms is summarized in Table 1. The time complexity of the K-means algorithm is small, but K-means iterates multiple times during the running process. Intuitively, our PDPC has the same time complexity as the DP_K-medoids, DPNM_K-medoids, Improve K-means, Hybrid PSO and K-means and DPC algorithms. However, we introduced the PSO algorithm, which reduces the number of iterations because of its strong global search capabilities. Overall, the running time of the proposed algorithm is less based on the following experimental analysis.

V. EXPERIMENTAL RESULTS AND DISCUSSION

All experiments are performed on an Intel Xeon E-2186M processor with 2.90 GHz and 32.0GB RAM running Windows 10 Ultimate. All programs are compiled and executed using Eclipse 4.3.2 on a Java HotSpot 64-bit server Virtual Machine.

In this section, we discuss the testing and verification of the proposed PDPC algorithm clustering performance and compare the results with those of the other six algorithms (DP_K-medoids [3], DPNM_K-medoids [3], Improved K-means [4], K-means [5], Hybrid PSO and K-means [6] and DPC [1]) using both classical synthetic data sets and real data sets. The clustering results of the algorithms were evaluated using the clustering time, the number of iterations, the accuracy of the clustering [45], and the precision and recall of external validity evaluation indicators.

TABLE 2. Characteristics of the data sets.

Data sets	Points	Attributes	Clusters
Spiral	312	2	3
Aggregation	788	2	7
Wdbc	569	30	2
Wireless	2000	7	4
Waveform	5000	21	3
Waveform(noise)	5000	40	3
Frogs-MFCCs	7195	22	4
Electrical Grid	10000	13	2
Pendigits	10992	16	10

A. DATA SET SELECTION AND INTRODUCTION

The data sets are divided into two groups, synthetic and real-world data sets. To verify the validity of the algorithm, we used two synthetic data sets and seven real data sets to test the performance of the clustering algorithms, as shown in Table 2. The synthetic data sets come from the research published in [43], [44]. Spiral has three clusters in the 3-spiral data sets. Aggregation consists of seven distinct groups that are non-Gaussian clusters. These data sets are labeled, and their descriptions are as follows.

Wdbc: Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image.

Wireless: These data were collected to perform experimentation on how WiFi signal strength can be used to determine an indoor location.

Waveform and Waveform(noise): These are different versions of a waveform with 3 classes of waves. Waveform contains 5000 instances with 21 attributes, and Waveform (noise) has 19 more noise attributes.

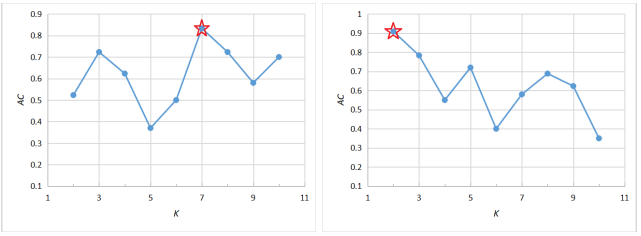
Frogs-MFCCs: This data set was used in several classification tasks related to the challenge of anuran species recognition through their calls.

Electrical Grid: This is a local stability analysis of a 4-node star system (where the electricity producer is in the center) implementing the decentral smart grid control concept.

Pendigits: This is a digit database that collects 250 samples from 44 writers.

A detailed description of the nine experimental data sets is shown in Table 2. In Table 2, column "Points" specifies the number of sample points in each data set; "Attributes" gives the dimension of each data sets; "Clusters" denotes the number of clusters in each data set. There were differences in data size, attribute number and/or cluster number for each data set. We use labeled data sets to test the performance of the algorithm, which is helpful for evaluating the clustering quality of the algorithm. Therefore, K represents the number of clusters, and the K value is directly input as a constant.

Generally, the number of clusters K can not be set too large. Therefore, for the data sets with unknown distribution, we determine the number of clusters K by experiment.



(a) Aggregation. (b) Wdbc.

FIGURE 3. Determination of the K value on the aggregation and Wdbc data sets.

The specific methods are as follows. First, the range of K can be set to 2-10. Then, by running the algorithm on each K value and calculating the accuracy (AC) (See Section V.B) of the current algorithm to determine the optimal number of clusters, the K corresponding to the value with the highest accuracy is regarded as the final number of clusters.Taking the Aggregation and Wdbc data sets as examples, as shown in Figure 3, it can be seen that the final K value is the same as the actual number of clusters.

B. EVALUATE CLUSTERING QUALITY

To evaluate the performance of these different clustering algorithms, three metrics are adopted in this paper. The first measure is the accuracy (AC) of the clustering results, which was proposed by Huang and Ng [45]:

$$AC = \frac{\sum_{i=1}^{K} a_i}{|N|} \tag{14}$$

where a_i is a sample that is classified correctly, K is the number of clusters, and N is the number of data points in the data set. The remaining two metrics are precision (PR) and recall (RE):

$$PR = \frac{\sum_{i=1}^{K} \frac{a_i}{a_i+b_i}}{K} \tag{15}$$

$$RE = \frac{\sum_{i=1}^{K} \frac{a_i}{a_i+c_i}}{K} \tag{16}$$

where K is the number of classes of data; a_i is the number of objects that are correctly assigned to class C_i $(1 \le i \le K)$; b_i is the number of objects that are incorrectly assigned to class C_i; and c_i is the number of objects that should be in class C_i but are not correctly assigned to it.

For AC, PR and RE, higher values indicate better clustering quality. When their values are 1, it means that the clustering result is entirely correct. In addition, we used the clustering time and the the number of iterations to evaluate the efficiency of each algorithm.

C. TEST CONVERGENCE OF THE PDPC ALGORITHM

Observe the convergence change of PDPC algorithm on the different data sets and different number of iterations to determine the convergence parameter ε, as shown in Figure 4.

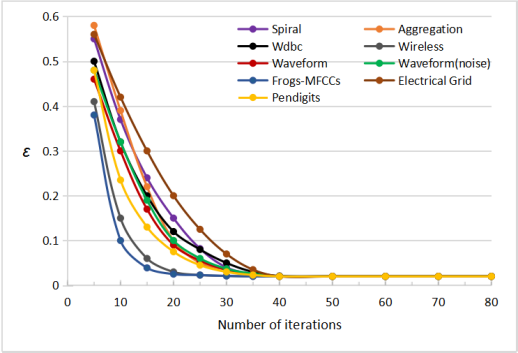


FIGURE 4. Test convergence.

TABLE 3. The threshold value d_c of each data set.

Data sets	d_c	Data sets	d_c
Spiral	0.00250	Waveform(noise)	1.00000
Aggregation	0.04996	Frogs-MFCCs	0.09259
Wdbc	1.00000	Electrical Grid	0.99685
Wireless	1.00000	Pendigits	1.00000
Waveform	0.99999		

From the experimental Figure 4 of the PDPC clustering algorithm, we can find that after the number of iterations reaches 40, the algorithm tends to converge, and the cluster centers no longer has obvious changes. Take the convergence parameter $\varepsilon = 0.02$. In the process of improved particle swarm optimization, the cluster centers that algorithm outputs are the cluster centers when the algorithm achieves convergence and stability.

D. PERFORMANCE ANALYSIS OF THE PDPC ALGORITHM

Before clustering, we used the method of calculating parameters proposed in this paper to get the threshold value d_c of each data set, as shown in Table 3. These values were adopted in the following experiments.

In this subsection, the PDPC algorithm is compared with the DP_K-medoids [3], DPNM_K-medoids [3], Improved K-means [4], K-means [5], Hybrid PSO and K-means [6] and DPC [1] on the data sets in Table 2. Twenty experiments were performed on each data set, the AC, PR, and RE of each experiment were statistically analyzed, and the best value, worst value and average value were recorded in 20 clustering experiments for each algorithm, as shown in Tables 4-6. The best results are in bold.

The experimental results in Tables 4-6 show that compared with other clustering algorithms, the average values of AC, PR, and RE of the PDPC clustering algorithm obtained relatively high values on most of the data sets listed in Table 2. This result shows that the proposed algorithm has a good clustering effect and high stability. First, the experimental results of each algorithm on two synthetic data sets are analyzed. For the Spiral data set, the DPC algorithm is optimal, and PDPC has higher values than other algorithms and ranks second. Whether the PDPC algorithm has the best value, the worst value or the average value, it is second only to DPC. The difference between the best and worst values of the PDPC algorithm is much smaller than that of DPC, indicating that the introduction of the PSO algorithm can improve the stability of the DPC algorithm. For the Aggregation data set, compared to the other six algorithms, the PDPC algorithm achieved the best clustering results. This result shows that the introduction of the PSO optimization algorithm in this paper overcomes the shortcoming of the DPC artificial selection center in which it easily falls into a local optimum.

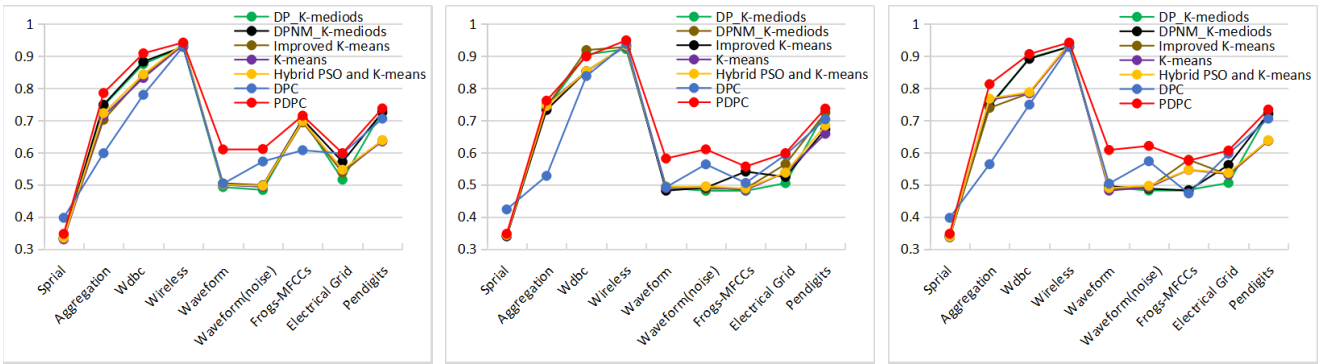
Furthermore, the experimental results of each algorithm on the real data set are analyzed. For the Wdbc data set, the PDPC algorithm achieves the optimal average value of AC and RE, while the DPNM_K-medoids algorithm obtains the optimal average value of PR. The PDPC algorithm has a lower average value of PR than the DP_K-medoids and DPNM_K-medoids algorithms but a higher value than the DPC. Similarly, the DPNM_K-medoids algorithm has also been performed 20 experiments on this data set. First, the DPNM_K-medoids algorithm selects cluster centers on the decision diagram. The center selected by the algorithm may be different in each experiment; Second, in the data object allocation stage, there are data points that originally belong to this cluster are not fully allocated to this cluster, and no data points that belong to other clusters are divided. Therefore, it can be known from the analysis of formula (15) that the DPNM_K-medoids algorithm may obtain a higher PR value in several experiments, that is, the optimal average value of PR may be obtained. For the remaining data sets, the PDPC algorithm performs well. The average values of the three indicators were optimal. In general, the algorithm proposed in this paper has a good clustering effect and high stability. Our proposed algorithm overcomes the shortcoming of the DPC algorithm in which it easily falls into a local optimum, and it achieves the purpose of automatically selecting cluster centers.

Based on the above analysis, we show the average value of each indicator (AC, PR, and RE) in a line chart in Figure 5. Taking the data sets as the x-axis values and the evaluation index results as the y-axis values, the data set index value curves can be constructed. The purpose is to test the effectiveness of the proposed algorithm for clustering performance.

According to the AC value curve shown in Figure 5(a), the PDPC algorithm (red line) achieves the best clustering accuracy of all algorithms on eight of the nine data sets. PDPC is followed by the DPC algorithm, which achieves the best clustering accuracy on one data set. The worst methods are the DP_K-medoids, DPNM_K-medoids, Improved K-means, K-means and Hybrid PSO and K-means algorithms, which do not obtain the best evaluation index value in any data set. The most significant improvement achieved by using the PDPC algorithm was observed for the Aggregation data set, and there was an improvement from 0.5977 using the DPC algorithm to 0.7850 using the PDPC algorithm. We also find that for the Waveform data set, the AC values of the

TABLE 4. The AC tested by seven algorithms on each data set.

	Spiral			*Aggregation*			*Wdbc*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-medoids[3]	0.3429	0.3218	0.3323	0.7652	0.7344	0.7453	**0.9279**	0.8318	0.8753
DPNM_K-medoids[3]	0.3429	0.3223	0.3326	0.7665	0.7369	0.7485	**0.9279**	0.8533	0.8826
Improved K-means[4]	0.3462	0.3143	0.3358	0.7259	0.6934	0.7016	0.8541	0.8062	0.8387
K-means[5]	0.3462	0.3043	0.3297	0.7855	0.6578	0.7137	0.8541	0.7922	0.8319
Hybrid PSO and K-means[6]	0.3498	0.3083	0.3345	0.7904	0.6747	0.7225	0.8607	0.8156	0.8431
DPC[1]	**0.4516**	**0.3558**	**0.3969**	0.7183	0.4734	0.5977	0.8770	0.6731	0.7794
PDPC(This study)	0.3558	0.3429	0.3470	**0.8731**	**0.7678**	**0.7850**	0.9196	**0.8541**	**0.9085**
	Wireless			*Waveform*			*Waveform(noise)*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-medoids[3]	0.9395	0.9086	0.9285	0.5032	0.4636	0.4924	0.5110	0.4558	0.4834
DPNM_K-medoids[3]	0.9395	0.9104	0.9290	0.5116	0.4798	0.5035	0.5116	0.4614	0.4980
Improved K-means[4]	0.9545	0.9167	0.9387	0.5018	0.4621	0.4997	0.5136	0.4390	0.4936
K-means[5]	0.9545	0.9123	0.9345	0.5018	0.4578	0.4997	0.5146	0.4362	0.4938
Hybrid PSO and K-means[6]	0.9561	0.9123	0.9388	0.5056	0.4572	0.5004	0.5168	0.4366	0.4967
DPC[1]	0.9595	0.9031	0.9294	0.5044	0.5012	0.5028	0.5720	**0.5720**	0.5720
PDPC(This study)	**0.9609**	**0.9241**	**0.9420**	**0.6382**	**0.5018**	**0.6095**	**0.6302**	0.5664	**0.6107**
	Frogs − MFCCs			*Electrical Grid*			*Pendigits*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-medoids[3]	0.7308	0.6589	0.7038	0.5243	0.5067	0.5155	0.7434	0.6967	0.7295
DPNM_K-medoids[3]	0.7305	0.6534	0.7034	**0.6388**	0.5173	0.5709	0.7416	0.6934	0.7247
Improved K-means[4]	0.7208	0.6439	0.6937	0.5922	0.5197	0.5375	0.6665	0.6179	0.6377
K-means[5]	0.7247	0.6462	0.6983	0.5957	0.5056	0.5447	0.6725	0.6150	0.6342
Hybrid PSO and K-means[6]	0.7246	0.6462	0.6980	0.5971	0.5075	0.5463	0.6769	0.6186	0.6378
DPC[1]	0.6272	0.5720	0.6071	0.6215	0.5712	0.5964	0.7271	0.6762	0.7054
PDPC(This study)	**0.7359**	**0.6645**	**0.7147**	0.6072	**0.5922**	**0.5979**	**0.7690**	**0.7152**	**0.7371**



(a) AC (b) PR (c) RE

FIGURE 5. The AC, PR and RE of seven algorithms on synthetic and real data sets.

six algorithms other than PDPC are very close, and PDPC is greatly improved. However, for the Sprial data set, the AC value of the PDPC algorithm is 0.3471, which is significantly lower than that of the DPC algorithm but still higher than those of the other five algorithms. The results indicate that the proposed algorithm may not be suitable for the Sprial data set. It is related to the distribution of the data set, because Sprial is a path-based spectral clustering result for 3-spiral data set.

Figure 5 shows similar trends in the metrics of different algorithms on different data sets. Compared with the other six algorithms, the PDPC algorithm showed the best clustering performance on most data sets. However, there are subtle differences. For example, the DPNM_K-medoids algorithm achieved top clustering performance for one data set when using the PR (Figure 5(b)), compared to no any data sets when using the AC (Figure 5(a)). Alternatively, the DP_K-medoids and DPNM_K-medoids algorithms had similar clustering performance for all of the indexes on all data sets except Electrical Grid. This is because the initial cluster center selection methods of these two clustering algorithms are the same; the difference is that the clustering criterion function is

TABLE 5. The PR tested by seven algorithms on each data set.

	Spiral			*Aggregation*			*Wdbc*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-medoids[3]	0.3427	0.3348	0.3392	0.7485	0.6864	0.7386	0.9341	0.8467	0.9046
DPNM_K-medoids[3]	0.3427	0.3357	0.3392	0.7485	0.6864	0.7418	**0.9383**	**0.8548**	**0.9186**
Improved K-means[4]	0.3462	0.3276	0.3402	0.7355	0.6580	0.7322	0.9026	0.8046	0.8525
K-means[5]	0.3462	0.3198	0.3414	0.7677	0.6497	0.7483	0.9017	0.7956	0.8525
Hybrid PSO and K-means[6]	0.3490	0.3257	0.3440	0.7848	0.6544	0.7446	0.9021	0.8033	0.8527
DPC[1]	**0.4988**	**0.3548**	**0.4229**	0.6230	0.4464	0.5273	0.9128	0.6529	0.8377
PDPC(This study)	0.3552	0.3425	0.3471	**0.8225**	**0.7490**	**0.7612**	0.9026	0.8503	0.8990
	Wireless			*Waveform*			*Waveform(noise)*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-medoids[3]	0.9468	0.9046	0.9217	0.5039	0.4715	0.4903	0.5109	0.4548	0.4809
DPNM_K-medoids[3]	0.9468	0.9167	0.9284	0.5041	0.4728	0.4941	0.5115	0.4593	0.4872
Improved K-means[4]	0.9597	0.9156	0.9325	0.5008	0.4690	0.4813	0.5134	0.4478	0.4905
K-means[5]	0.9597	0.9182	0.9304	0.5008	0.4587	0.4890	0.5145	0.4318	0.4932
Hybrid PSO and K-means[6]	0.9598	0.9165	0.9331	0.5048	0.4588	0.4926	0.5157	0.4341	0.4949
DPC[1]	**0.9632**	0.9029	0.9377	0.5022	0.4837	0.4919	0.5632	0.5632	0.5632
PDPC(This study)	**0.9632**	**0.9203**	**0.9490**	**0.6010**	**0.5524**	**0.5813**	**0.6301**	**0.5864**	**0.6098**
	Frogs − MFCCs			*Electrical Grid*			*Pendigits*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-medoids[3]	0.5264	0.4608	0.4806	0.5235	0.4715	0.5046	0.7480	0.6946	0.7263
DPNM_K-medoids[3]	0.5259	0.4598	0.4877	0.6358	0.5371	0.5638	0.7461	0.6903	0.7232
Improved K-means[4]	**0.5736**	0.5078	0.5405	0.5924	0.5015	0.5240	0.7128	0.6634	0.6726
K-means[5]	0.5204	0.4359	0.4836	0.5958	0.5153	0.5375	0.6884	0.6391	0.6582
Hybrid PSO and K-means[6]	0.5286	0.4372	0.4879	0.5975	0.5177	0.5376	0.7136	0.6655	0.6841
DPC[1]	0.5632	0.4041	0.5052	**0.6600**	0.5457	0.5929	0.7274	0.6703	0.7033
PDPC(This study)	0.5730	**0.5197**	**0.5564**	0.6068	**0.5924**	**0.5980**	**0.7668**	**0.7174**	**0.7366**

different, that is, the stopping conditions of the clustering are different [3]. For the Wdbc data set, the PDPC algorithm obtains the highest values of AC and RE on the evaluation index of clustering performance, while DPNM_K-medoids obtains the highest value of PR. The PDPC algorithm performed best on the seven data sets when using the PR index; on the other hand, it is still the best-performing algorithm on the eight data sets when using the RE value, just as it is when using the AC value. For the Waveform data set, the PDPC algorithm showed the best clustering performance on AC, PR and RE. Furthermore, for the Waveform(noise) data set, which had an increase of nineteen attributes with noise data in relation to Waveform, the performance of the PDPC algorithm is still better than that of the other six algorithms. Therefore, the PDPC algorithm is the best method for processing the Waveform(noise) data set, which indicates that the PDPC algorithm is more stable than the other six algorithms.

Table 7 gives the number of data sets in which each of the eight algorithms showed the top clustering performance for the different evaluation indexes when using synthetic data sets and real data sets. For AC, the PDPC algorithm tied for the best clustering performance by achieving the highest value on eight of the nine data sets. PR and RE all showed similar results to those for AC. In all cases, the PDPC algorithm demonstrated the best clustering performance. In each evaluation index, the PDPC algorithm showed the best clustering performance. These results demonstrate that the PDPC algorithm is effective and excellent regardless of the evaluation index chosen.

It can be seen from Table 4-6 that the clustering quality of PDPC algorithm is better than DPC on most of the data sets in Table 2. Further, Figure 5 visually shows that PDPC (red line) is superior to DPC (blue line) on most of the data sets. From the above analysis, combined with the advantages of the PSO algorithm, the PDPC algorithm proposed in this paper solves the disadvantages of DPC. A method for calculating the parameter d_c is proposed to solve the uncertainty and unreliability of DPC selection based on empirical values. For some unevenly distributed data sets, the initial centers found by the DPC algorithm may be located in the same cluster or may not be found. The DPC may consider the non-cluster centers in the dense clusters as the center points of the sparse clusters, causing the cluster centers found to fall into a local optimum. Our algorithm solves this problem well. And PDPC algorithm solves the limitation that traditional DPC cannot automatically determine the cluster centers, avoids the subjectivity of the manual selection process. The experimental results show that our algorithm has a stronger global search ability, higher stability and a better clustering effect.

TABLE 6. The RE tested by seven algorithms on each data set.

	Spiral			*Aggregation*			*Wdbc*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-mediods[3]	0.3429	0.3313	0.3358	0.7556	0.7376	0.7406	0.9119	0.8533	0.8947
DPNM_K-mediods[3]	0.3429	0.3355	0.3398	0.7564	0.7397	0.7455	0.9090	0.8409	0.8914
Improved K-means[4]	0.3462	0.3297	0.3401	0.7555	0.7266	0.7387	0.8052	0.7599	0.7843
K-means[5]	0.3462	0.3206	0.3369	0.7847	0.7209	0.7639	0.8052	0.7491	0.7843
Hybrid PSO and K-means[6]	0.3497	0.3270	0.3383	0.7876	0.7209	0.7677	0.8078	0.7542	0.7877
DPC[1]	**0.4662**	**0.3553**	**0.3968**	0.7314	0.3876	0.5635	0.8368	0.5660	0.7488
PDPC(This study)	0.3553	0.3430	0.3472	**0.9414**	**0.7516**	**0.8125**	**0.9152**	**0.8573**	**0.9063**
	Wireless			*Waveform*			*Waveform(noise)*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-mediods[3]	0.9395	0.9086	0.9285	0.5040	0.4697	0.4921	0.5109	0.4598	0.4816
DPNM_K-mediods[3]	0.9395	0.9104	0.9290	0.5042	0.4701	0.4950	0.5115	0.4601	0.4873
Improved K-means[4]	0.9545	0.9167	0.9387	0.5010	0.4615	0.4882	0.5135	0.4397	0.4901
K-means[5]	0.9545	0.9123	0.9345	0.5010	0.4592	0.4813	0.5145	0.4439	0.4921
Hybrid PSO and K-means[6]	0.9561	0.9123	0.9388	0.5055	0.4571	0.4893	0.5167	0.4365	0.4966
DPC[1]	0.9595	0.9031	0.9294	0.5039	**0.5027**	0.5032	0.5728	0.5728	0.5728
PDPC(This study)	**0.9609**	**0.9241**	**0.9420**	**0.6363**	0.5010	**0.6078**	**0.6351**	**0.5745**	**0.6205**
	Frogs − MFCCs			*Electrical Grid*			*Pendigits*		
Algorithms	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*	*Best*	*Worst*	*Average*
DP_K-mediods[3]	0.5925	0.5248	0.4836	0.5254	0.4682	0.5053	0.7429	0.6933	0.7226
DPNM_K-mediods[3]	0.5901	0.5241	0.4821	0.6470	0.5395	0.5612	0.7406	0.6903	0.7204
Improved K-means[4]	0.6054	0.5397	0.5775	0.6000	0.5045	0.5304	0.6682	0.6144	0.6363
K-means[5]	0.5778	0.4939	0.5458	0.6037	0.5015	0.5326	0.6691	0.6153	0.6372
Hybrid PSO and K-means[6]	0.5982	0.4967	0.5464	0.6065	0.5064	0.5364	0.6786	0.6175	0.6380
DPC[1]	0.5328	0.3726	0.4727	**0.6541**	0.5718	0.5963	0.7254	0.6753	0.7053
PDPC(This study)	**0.5946**	**0.5345**	**0.5745**	0.6156	**0.6000**	**0.6061**	**0.7639**	**0.7145**	**0.7337**

TABLE 7. The number of data sets in which each of the seven algorithms showed top clustering performance for the average value of the different evaluation indexes when using synthetic data sets and real data sets.

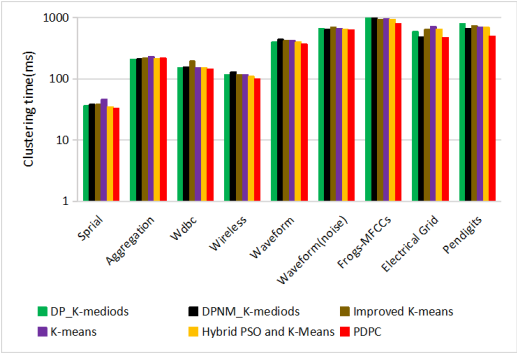
Algorithms	AC	PR	RE
DP_K-medoids[3]	0	0	0
DPNM_K-medoids[3]	0	1	0
Improve K-means[4]	0	0	0
K-means[5]	0	0	0
Hybrid PSO and K-means[6]	0	0	0
DPC[1]	1	1	1
PDPC(This study)	8	7	8

E. EVALUATE OF CLUSTERING TIME AND NUMBER OF ITERATIONS

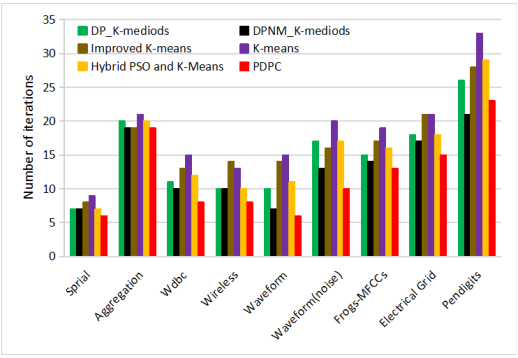
In Section IV.E, we analyze theoretically the complexity of the DP_K-medoids [3], DPNM_K-medoids [3], Improved K-means [4], K-means [5], Hybrid PSO and K-means [6], DPC [1] and PDPC algorithms. Table 1 gives the detailed theoretical results. In this subsection, we compare the actual clustering time and the number of iterations of the six algorithms other than DPC, measured by the average clustering time and the number of iterations of 20 repeated clustering processes. The DPC algorithm does not perform the iterative clustering process, which distributes the remaining data points directly to the nearest cluster centers, so it is not compared with this method.

Figure 6(a) shows the average clustering time of the six clustering algorithms in milliseconds on the nine data sets. As shown, the difference in clustering time between the six methods is not large. However, compared with the other five algorithms, the clustering time of the proposed PDPC algorithm is relatively low, although the time complexity is not greatly improved. We can see that the DP_K-medoids algorithm clustering time was close to that of DPNM_K-medoids. Although the time required to manually select the centers was excluded, the DP_K-medoids and DPNM_K-medoids algorithms must generate a decision diagram, which is time consuming. This was one reason why their computational efficiency was lower. We can also see that the K-means algorithm has a longer clustering time because it has more iterations than the other algorithms on most data sets, as shown in Figure 6(b). Figure 6(b) shows the average number of iterations of the six clustering algorithms on the nine data sets. Overall, the number of iterations of PDPC is less than that of the other algorithms.

This paper introduces the PSO optimization algorithm; because of its simple concept, strong global search capability and high stability, it can find the optimal solution in relatively few iterations. The above analysis shows that the PDPC algorithm runs faster than the other algorithms.



(a) Clustering time.



(b) Number of iterations.

FIGURE 6. The six algorithms evaluate the clustering time and number of iterations on different data sets.

Therefore, the PDPC algorithm reduces the number of iterations and the clustering time and improves the efficiency of the DPC algorithm.

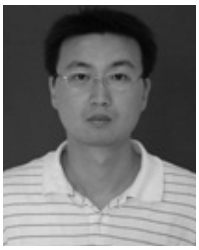
VI. SUMMARY

To overcome the disadvantages in the DPC algorithm, a novel clustering algorithm based on DPC & PSO (PDPC) is proposed. Particle swarm optimization (PSO) is introduced because of its simple concept and strong global search ability, which can find the optimal solution in relatively few iterations. Furthermore, to address the influence of the selection parameter cut-off distance d_c value on the clustering results, a method for calculating the parameter d_c is proposed. Finally, the PDPC and six typical algorithms are tested on classical synthetic data sets and real data sets, and the experiments verified that the clustering results, the clustering time and the number of iterations of the PDPC algorithm are better than those of other algorithms. The PDPC algorithm achieves the purpose of automatically selecting cluster centers and overcomes the effects of the parameter d_c. Compared with the other six algorithms, the PDPC algorithm has a stronger global search ability, higher stability and a better clustering effect.

REFERENCES

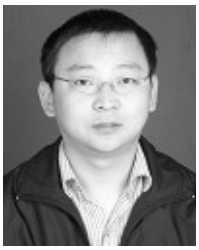
[1] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014.
[2] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Human Sci. (MHS)*, 1995, pp. 39–43.
[3] X. Juanying and Y. Qu, "K-medoids clustering algorithms with optimized initial seeds by density peaks," *J. Frontiers Comput. Sci. Technol.*, vol. 10, no. 2, pp. 230–247, 2016.
[4] E. Zhu and R. Ma, "An effective partitional clustering algorithm based on new clustering validity index," *Appl. Soft Comput.*, vol. 71, pp. 608–621, Oct. 2018.
[5] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, 1967, vol. 1, no. 14, pp. 281–297.
[6] D. W. van der Merwe and A. P. Engelbrecht, "Data clustering using particle swarm optimization," in *Proc. Congr. Evol. Comput. (CEC)*, vol. 1, 2003, pp. 215–220.
[7] Y. Si, P. Liu, P. Li, and T. P. Brutnell, "Model-based clustering for RNA-seq data," *Bioinformatics*, vol. 30, no. 2, pp. 197–205, Jan. 2014.
[8] L. H. Son and T. M. Tuan, "A cooperative semi-supervised fuzzy clustering framework for dental X-ray image segmentation," *Expert Syst. Appl.*, vol. 46, pp. 380–393, Mar. 2016.
[9] A. Mehta and O. Dikshit, "Comparative study on projected clustering methods for hyperspectral imagery classification," *Geocarto Int.*, vol. 31, no. 3, pp. 296–307, Mar. 2016.
[10] Y. Yang, "TAD: A trajectory clustering algorithm based on spatial-temporal density analysis," *Expert Syst. Appl.*, vol. 139, Jan. 2020, Art. no. 112846, doi: 10.1016/j.eswa.2019.112846.
[11] C. Jiang-Hui, "Spectral analysis of sky light based on trajectory clustering," *Spectrosc. Spectral Anal.*, vol. 39, no. 4, pp. 1301–1306, 2019.
[12] C. Qu, H. Yang, J. Cai, J. Zhang, and Y. Zhou, "DoPS: A double-peaked profiles search method based on the RS and SVM," *IEEE Access*, vol. 7, pp. 106139–106154, 2019, doi: 10.1109/ACCESS.2019.2927251.
[13] Q. Cai-Xia, Y. Hai-Feng, C. Jiang-Hui, and X. Ya-Ling, "P-Cygni profile analysis of the spectrum: LAMOST J152238.11+333136.1," *Spectrosc. Spectral Anal.*, vol. 40, no. 4, pp. 1304–1308, 2020.
[14] H. Yang, C. Qu, J. Cai, S. Zhang, and X. Zhao, "SVM-Lattice: A recognition & evaluation frame for double-peaked profiles," *IEEE Access*, early access, Apr. 27, 2020, doi: 10.1109/ACCESS.2020.2990801.
[15] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques* (Series in Data Management Systems), 3rd ed. San Mateo, CA, USA: Morgan Kaufmann, 2011, pp. 83–124.
[16] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, vol. 344. Hoboken, NJ, USA: Wiley, 2009.
[17] Y. Li, J. Cai, H. Yang, J. Zhang, and X. Zhao, "A novel algorithm for initial cluster center selection," *IEEE Access*, vol. 7, pp. 74683–74693, 2019, doi: 10.1109/ACCESS.2019.2921320.
[18] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: An overview," *WIREs Data Mining Knowl. Discovery*, vol. 2, no. 1, pp. 86–97, Jan. 2012.
[19] J. D. Banfield and A. E. Raftery, "Model-based Gaussian and non-Gaussian clustering," *Biometrics*, vol. 49, no. 3, pp. 803–821, Sep. 1993.
[20] M. Ester, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. Kdd*, 1996, vol. 96, no. 34, pp. 226–231.
[21] A. Hinneburg and D. A. Keim, "An efficient approach to clustering in large multimedia databases with noise," in *Proc. 4th. Int. Conf. Knowl. Discovery. Data Mining*, vol. 98, Aug. 1998, pp. 58–65.
[22] D. McParland and I. C. Gormley, "Model based clustering for mixed data: ClustMD," *Adv. Data Anal. Classification*, vol. 10, no. 2, pp. 155–169, Jun. 2016.
[23] A. Rodríguez, E. Cuevas, D. Zaldivar, and L. Castañeda, "Clustering with biological visual models," *Phys. A, Stat. Mech. Appl.*, vol. 528, Aug. 2019, Art. no. 121505.
[24] L. Rokach, "A survey of clustering algorithms," *Data Mining and Knowledge Discovery Handbook*. Boston, MA, USA: Springer, 2009, pp. 269–298.
[25] Y.-J. Zheng, H.-F. Ling, S.-Y. Chen, and J.-Y. Xue, "A hybrid neuro-fuzzy network based on differential biogeography-based optimization for online population classification in earthquakes," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 4, pp. 1070–1083, Aug. 2015.
[26] Y.-J. Zheng and H.-F. Ling, "Emergency transportation planning in disaster relief supply chain management: A cooperative fuzzy optimization approach," *Soft Comput.*, vol. 17, no. 7, pp. 1301–1314, Jul. 2013.
[27] B. Jiang and N. Wang, "Cooperative bare-bone particle swarm optimization for data clustering," *Soft Comput.*, vol. 18, no. 6, pp. 1079–1091, Jun. 2014.

[28] Y.-J. Zheng, H.-F. Ling, J.-Y. Xue, and S.-Y. Chen, "Population classification in fire evacuation: A multiobjective particle swarm optimization approach," *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 70–81, Feb. 2014.
[29] J. Ding, Z. Chen, X. He, and Y. Zhan, "Clustering by finding density peaks based on Chebyshev's inequality," in *Proc. 35th Chin. Control Conf. (CCC)*, Jul. 2016, pp. 7169–7172.
[30] J. Ding, X. He, J. Yuan, and B. Jiang, "Automatic clustering based on density peak detection using generalized extreme value distribution," *Soft Comput.*, vol. 22, no. 9, pp. 2777–2796, May 2018.
[31] M. Du, S. Ding, and H. Jia, "Study on density peaks clustering based on k-nearest neighbors and principal component analysis," *Knowl.-Based Syst.*, vol. 99, pp. 135–145, May 2016.
[32] J. Xie, H. Gao, W. Xie, X. Liu, and P. W. Grant, "Robust clustering by detecting density peaks and assigning points based on fuzzy weighted K-nearest neighbors," *Inf. Sci.*, vol. 354, pp. 19–40, Aug. 2016.
[33] S. Jia, G. Tang, J. Zhu, and Q. Li, "A novel ranking-based clustering approach for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 88–102, Jan. 2016.
[34] J. Xu, G. Wang, and W. Deng, "DenPEHC: Density peak based efficient hierarchical clustering," *Inf. Sci.*, vol. 373, pp. 200–218, Dec. 2016.
[35] A. A. A. Esmin, D. L. Pereira, and F. P. A. de Araujo, "Study of different approach to clustering data by using the particle swarm optimization algorithm," in *Proc. IEEE Congr. Evol. Comput. (IEEE World Congr. Comput. Intelligence)*, Jun. 2008, pp. 1817–1822.
[36] I. W. Kao, C. Y. Tsai, and Y. C. Wang, "An effective particle swarm optimization method for data clustering.," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage.*, Dec. 2007, pp. 548–552.
[37] R. Chouhan and A. Purohit, "An approach for document clustering using PSO and K-means algorithm," in *Proc. 2nd Int. Conf. Inventive Syst. Control (ICISC)*, Jan. 2018, pp. 1380–1384.
[38] A. Khatami, "A new PSO-based approach to fire flame detection using K-medoids clustering," *Expert Syst. Appl.*, vol. 68, pp. 69–80, Feb. 2017.
[39] Y. Jiang, C. Liu, C. Huang, and X. Wu, "Improved particle swarm algorithm for hydrological parameter optimization," *Appl. Math. Comput.*, vol. 217, no. 7, pp. 3207–3215, Dec. 2010.
[40] A. O'Hagan, T. B. Murphy, I. C. Gormley, P. D. McNicholas, and D. Karlis, "Clustering with the multivariate normal inverse Gaussian distribution," *Comput. Statist. Data Anal.*, vol. 93, pp. 18–30, Jan. 2016.
[41] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proc. IEEE Int. Conf. Evol. Comput., IEEE World Congr. Comput. Intell.*, May 1998, pp. 69–73.
[42] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 240–255, Jun. 2004.
[43] H. Chang and D.-Y. Yeung, "Robust path-based spectral clustering," *Pattern Recognit.*, vol. 41, no. 1, pp. 191–203, Jan. 2008.
[44] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering aggregation," *ACM Trans. Knowl. Discovery Data (TKDD)*, vol. 1, no. 1, p. 4, 2007.
[45] C. M. Stein, "Estimation of the mean of a multivariate normal distribution," *Ann. Statist.*, vol. 9, no. 6, pp. 1135–1151, Nov. 1981.

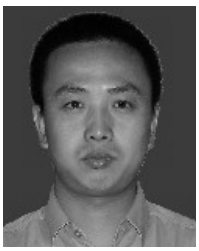


JIANGHUI CAI is a Chief Professor of computer application technology with the Taiyuan University of Science and Technology, Taiyuan, China. He is a long-term member of the Institute for Intelligent Information and Data Mining. His research interests concern the data mining and machine learning methods in specific backgrounds of astronomical informatics, seismology, and mechanical engineering. He is a Senior Member of the China Computer Federation (CCF).

HUILING WEI was born in Shanxi, China, in 1993. She is currently pursuing the M.S. degree with the Department of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan, China. Her current research interests include data mining and artificial intelligence.



HAIFENG YANG is a Professor of computer application technology with the Taiyuan University of Science and Technology, Taiyuan, China. He is a long-term member of the Institute for Intelligent Information and Data Mining. His research interests concern the data mining and machine learning methods in specific backgrounds, especially on astronomical big data. He is a member of the China Computer Federation (CCF) and the Chinese Astronomical Society (CAS).



XUJUN ZHAO received the M.S. degree in computer science and technology from the Taiyuan University of Technology, China. He is currently pursuing the Ph.D. degree with the Taiyuan University of Science and Technology. His research interests include data mining and parallel computing.