IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 10, 2021, accepted June 14, 2021, date of publication June 21, 2021, date of current version June 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3091005

Dynamic Modified Chaotic Particle Swarm Optimization for Radar Signal Sorting

XIAOYAN WANG[1], **XIONGJUN FU**[1], **(Member, IEEE), JIAN DONG**[2], **AND JIAHUAN JIANG**[1]
[1]School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China
[2]Shijiazhuang Campus, Army Engineering University, Shijiazhuang 050003, China

Corresponding author: Xiongjun Fu (fuxiongjun@bit.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61571043, in part by the 111 Project of China under Grant B14010, in part by the Research Innovation Funding of Army Engineering University under Grant KYSZJQZL2005, and in part by the General Program of Natural Science Foundation of Hebei Province of China under Grant F2020506036.

ABSTRACT Radar signal sorting is the core part of electronic support measures, which is responsible for deinterleaving the overlapping pulse sequences received by the receiver from the complex environment, separating different radiation source signals, and providing support for radiation source identification. Particle swarm optimization (PSO) is a population-based global optimization algorithm with great advantages in intelligent sorting of complex signals, which can adapt to the electromagnetic environment with complex and variable radiation source signals and high pulse stream density. However, the PSO-based sorting method is prone to premature convergence and cannot adaptively adjust particle swarm parameters and positions. In this paper, a dynamic modified chaotic PSO algorithm (DMCPSO) is proposed. Chaotic search is used to increase the diversity of particle swarm in the later iteration to avoid premature convergence and falling into local optimum. Adaptive adjustment parameters related to the particle fitness value are adopted to balance the ability of global search and local search. A new fitness function is proposed and the particle position is dynamically corrected by clustering analysis to improve the accuracy of particle position optimization and avoid the influence from the distribution of feature parameters. The simulation results show that the DMCPSO algorithm provides stable and fast performance with excellent sorting indexes in complex, variable, and dense signal environment.

INDEX TERMS Radar signal sorting, particle swarm optimization, chaotic search, adaptive adjustment parameters, fitness function, dynamic position correction.

I. INTRODUCTION

As an integral component of electronic warfare, electronic support measures (ESM) are tasked with searching, intercepting, analyzing, and identifying enemy radar signals. With the increasing complexity of the electromagnetic environment, dense and variable radiation source signals enter digital reconnaissance receivers and intertwine into complex pulse stream sequences. According to the characteristic parameters of the intercepted pulse, the arrival time, and other information, the pulse streams are sorted. The signals belonging to the same radiation source are accurately classified, and then the radar models are identified according to the characteristic parameters of different radiation sources. Based on the identification results, the type, properties, and threat level of each radar are obtained [1]. From the above analysis, it can be seen that radiation source sorting is a key link in radar reconnaissance signal processing, which directly affects the performance of radar reconnaissance equipment and is related to subsequent operational decisions. Wrong sorting results will lead to a large number of false alarms and missed alarms, which will seriously affect the effectiveness of the confrontation. Therefore, an efficient and accurate signal sorting algorithm is extremely important.

Signal sorting technology is mainly divided into pulse repetition interval (PRI) analysis and feature clustering [2]–[5]. Among them, the sorting algorithm based on PRI is mainly divided into two types: statistical histogram algorithm (cumulative difference histogram CDIF, sequence difference histogram SDIF) and sequence search method. CDIF and SDIF algorithms analyze and detect potential signal PRI utilizing order-by-order statistical time of arrival (TOA) difference and separate the interleaved sequences step by step. The sequence search method uses the PRI characteristics of

The associate editor coordinating the review of this manuscript and approving it for publication was Pavlos I. Lazaridis.

known radar signals as the basis for sorting and searches for pulse sequences of known signals from the interleaved pulse stream by step-by-step retrieval. Modern interception systems often combine these two algorithms to achieve signal sorting. However, the wide application of PRI agility and low-interception technology makes the TOA of intercepted signal appear large jitter and loss, which destroys the statistical characteristics of TOA difference. These reasons greatly increase the difficulty of sorting based on PRI analysis, making it impossible for such algorithms to effectively estimate the PRI value of the signals and, consequently, to separate the pulse sequences corresponding to the true PRI value from the interleaved pulse streams. Therefore, the clustering algorithm based on radar characteristic parameters came into being and became an indispensable auxiliary task in the deinterleaving process of pulse signals. This type of algorithm dilutes the pulse stream and divides it into several subspaces for pre-deinterleaving [6]. It draws on the idea of clustering technique to divide the set of pulse signal feature parameters according to a specific criterion, aggregating similar features and separating different features, when the signal source is unknown.

Many researchers at home and abroad have done a lot of research on sorting algorithms based on radar characteristic parameters clustering. Literature [7] proposed a data field-based radar sorting algorithm to improve the correct rate of sorting by introducing data field theory. Literature [8] proposed a sorting algorithm based on point symmetry, which used symmetric measurement distance to cluster pulses instead of Euclidean distance to reduce the influence of distance on data points. These algorithms can achieve good sorting results to a certain extent, but the sorting performance is poor when dealing with complex radar feature sets. Therefore, in order to adapt to the complex electromagnetic environment, many intelligent algorithms are gradually applied to the field of radar signal sorting, and the heuristic optimization algorithm is one of them. Compared with traditional sorting methods based on clustering, the heuristic algorithm is a group algorithm, which improves the sorting performance by finding the global optimum. In recent years, many heuristic optimization algorithms have emerged, such as ant colony algorithm [9], artificial bee colony algorithm [10], particle swarm algorithm [11], etc. Particle swarm optimization (PSO) is a widely used algorithm, which is optimized through information sharing mechanism. It has the advantages of simple modeling, easy implementation and fast convergence [12]. However, although the PSO algorithm provides the possibility of global search, it does not guarantee convergence to the global optimum. And it has disadvantages such as premature convergence and insufficient optimizing ability. At present, the main trend to solve these problems is to increase population diversity or integrate other methods. For example, a method based on dynamic particle swarm optimization and K-means is proposed in the literature [13], [14], which enhances the performance of the algorithm by dynamically adjusting the inertia weight and acceleration coefficients. In the literature [15] the clustering centroid is improved by changing the fitness function. In the literature [16], the inertia weight setting and updating strategy are improved to enhance the ability of local search and global search. However, they are not fully adaptive and the clustering results do not well meet the requirements of radar system for signal sorting accuracy and real-time performance.

In summary, to accurately separate various types of radiation source signals from dense, complex, and variable intercepted pulse streams, this paper proposes a dynamic modified chaotic particle swarm optimization (DMCPSO) algorithm to improve the difficulty of correct classification in traditional sorting algorithms based on clustering and the deficiency of PSO optimization, and enhance the sorting performance. The main contributions of this paper are as follows.

(1) The Tent chaotic search is added. And the chaotic mapping is improved for the phenomenon of small cycles and unstable period points that appear in Tent, which is unfavorable to the optimization problem. It improves the defect that the population diversity is lost in the late iteration and easily falls into local optimum;

(2) Adaptive adjustment parameters (inertia weight and acceleration coefficients) related to the particle fitness value are used to perceive population changes in real-time so that the global and local search ability of particles can be better combined;

(3) A new fitness function is proposed to avoid the shortcomings brought by using only internal distances, such as the single form of the fitness function, containing little information and being influenced by the distribution of feature parameters. It improves the accuracy of sorting;

(4) The positions of particles are adjusted dynamically by clustering analysis and then the gravity center index is introduced to correct the final position of the particles that meet the conditions so that it is closer to the real clustering centers and reduces the impact of discrete points and data distribution on the accuracy of sorting.

The simulation results show that the proposed algorithm in this paper has greater advantages over several other improved particle swarm optimization algorithms in terms of several common and new indexes (Clustering Quality, Adjusted Rand Index, Normalized Mutual Information, Centroid Index, Davies Bouldin Index, Silhouette Index). Its convergence speed, stability and robustness also have a better sorting effect.

II. PARTICLE SWARM OPTIMIZATION ALGORITHM

Particle swarm optimization (PSO) algorithm is the simulation of migration and swarms in the process of birds foraging [17]. Its basic idea is to find the optimal solution through cooperation and information sharing between particles. For the pros and cons of the individual, the fitness function is calculated to evaluate. Each particle in the population contains two information of position and velocity. According to the update of the two information, the individuals find the global optimal position and the individual optimal position in the search space to perform the search.

In the standard PSO algorithm, the positions and velocities of the particles are first randomized, and then the position of each particle is calculated according to the objective function of the optimization problem. In each generation, individual updates its position and velocity according to (1).

$$\begin{aligned} V_i(k+1) &= wV_i(k) + c_1 \cdot rand \cdot (X_{pbest,i} - X_i) \\ &\quad + c_2 \cdot rand \cdot (X_{gbest,i} - X_i) \\ X_i(k+1) &= X_i(k) + V_i(k+1) \end{aligned} \tag{1}$$

where, X_i and V_i represent the position and velocity of the current ith particle, respectively; X_{pbest} and X_{gbest} represent the best individual value and the best value in the cluster, respectively; w is the inertia weight; c_1 and c_2 are the acceleration coefficients. From (1), it is clear that the update of the particle is to search the global optimal value in the solution space of the feasible region by the interaction of memory items, self cognition, and group cognition.

As can be seen above, the PSO algorithm involves few parameters, simple structure, strong operability, and easy implementation, and finds the optimal solution through continuous adjustment in the search space. These advantages make it known as the research direction of radar signal sorting, but it still has congenital disadvantages:

(1) The population is prone to premature convergence in the late iteration. In the PSO, the particles always move to the individual optimal position and the global optimal position. This information sharing mechanism reduces the diversity of the population, and the population is prone to premature convergence.

(2) Fixed inertia weigh w and acceleration coefficients c_1 and c_2. The quality of the population is not the same at different stages of the algorithm, and the fixed parameters make it impossible for the particles to adjust adaptively to changes in the population, which may cause the population to skip the global optimum.

(3) Only the internal distance information is used to characterize the fitness function. This calculation method ignores the information outside the cluster, which is prone to misclassification, and is susceptible to data distribution.

(4) The particle positions are not accurate. Taking the position of particles as the clustering centers is only a rough estimation method, which does not truly reflect the distribution of clusters and is sometimes affected by the distribution of feature parameters.

Therefore, in order to better play the advantages of the PSO algorithm, the DMCPSO algorithm proposed in this paper has made many improvements to the above shortcomings, to meet the accuracy and real-time requirements of signal sorting technology in complex electromagnetic environment.

III. IMPROVED PARTICLE SWARM OPTIMIZATION ALGORITHM

A. CHAOTIC SEARCH

Chaotic search variables have the advantages of randomness, ergodicity, and regularity, which enable it to go through all stages of a certain range without repetition. Using these advantages, it can maintain the diversity of populations during the iteration of the algorithm, effectively avoid premature convergence, and improve the global search capability. The basic idea of the chaotic search proposed in this paper is as follows. Firstly, the chaotic sequence is generated based on the optimal position searched by the current entire particle swarm. Then, the optimal particles in the generated chaotic sequence are used to replace the particle individuals in the current particle swarm whose individual optimal values are unchanged and are not global optimal.

The existing chaotic mappings are Tent mapping, Logistic mapping, etc. Most chaotic optimization algorithms proposed in the literature use Logistic mapping to generate chaotic search sequences. For example, Reference [18] proposed to add Logistic mapping into the PSO algorithm and achieved some results. However, due to the uneven distribution of Logistic chaotic sequence, the long search time cannot meet the real-time requirements of signal sorting technology. Therefore, in this paper, the Tent mapping [19] with uniform distribution characteristics is chosen to shorten search time and improve the algorithm's optimization speed. However, the small period and unstable periodic points that exist in the Tent mapping will cause the mapping to a fixed point [20]. To solve this problem, this paper adds random disturbance factor to improve Tent mapping, and the relationship is shown in (2).

$$z_{n+1} = \begin{cases} 2\left[z_n + 0.1 \times rand(0,1)\right], & 0 \le z_n < 0.5 \\ 2\left[1 - (z_n + 0.1 \times rand(0,1))\right], & 0.5 \le z_n \le 1 \end{cases} \tag{2}$$

The specific chaotic search process is as follows, where the chaotic initial variable is defined as z:

(1) Transform the current global optimal position X_{gbest} from the optimization variable taking the value interval $[X_{\min}, X_{\max}]$ to the chaotic variable taking the value interval [0,1];

$$z = \frac{X_{gbest} - X_{\min}}{X_{\max} - X_{\min}} \tag{3}$$

(2) The global optimal position X_{gbest} is subjected to M chaotic disturbance to generate chaotic sequences. Then, the generated chaotic variable sequence is mapped to the value space of the original variable through (4):

$$z = (X_{\max} - X_{\min})\, z + X_{\min} \tag{4}$$

(3) Generate new particles based on the generated chaotic sequences: $X = X_{gbest} + \beta \cdot z$. In order to make the particle carry out a small range of chaotic disturbance within the feasible solution range and near the optimal solution, here the value range of β is less than 10 %;

(4) In the original solution space, each feasible solution experienced by the chaotic variable is re-clustered to calculate the fitness. And the feasible solution with the best performance is retained, to replace the particle individuals in the original solution space.

In this work, the Tent mapping has the characteristics of uniform probability density and fast optimization speed, which ensures the high diversity of population particles in the later stage of the algorithm and avoids the PSO falling into local optimum. Meanwhile, the Tent chaotic mapping is improved by adding random perturbation factors, so that it does not tend to the fixed points to give full play to its best performance.

B. ADAPTIVE ADJUSTMENT OF INERTIA WEIGHT AND ACCELERATION COEFFICIENT

The PSO algorithm mainly involves three parameters, namely, inertia weight w and acceleration coefficients c_1, c_2. And their values play an important role in the performance of the algorithm. In this paper, the update methods of the three parameters are analyzed and optimized respectively.

The inertia weight keeps the particle moving inertially, giving it a tendency to expand its search space. The acceleration coefficient adjusts the role played by the particle's own experience and social experience in its motion. To balance the local search capability and global search capability, the main idea of existing research direction is to determine the required search capability of a particle based on the individual fitness value during the same iteration of the particle swarm. For example, literature [13], [21] proposed an update method based on dynamically adjusted parameters, but it still needs to specify the range of inertia weights in advance; literature [22] proposed an inertia weight for partial differential equations, but the update method is complicated. Therefore, in this paper, we propose an update method that concisely and adaptively adjusts the weights according to the fitness value, as follows:

$$w = e^{-\frac{F_{pbest}}{F_{gbest}}} \tag{5}$$

where, F_{gbest} is the global optimal fitness value and F_{pbest} is the particle individual optimal fitness value. From (5), it can be seen that the inertia weight of each particle is inversely proportional to its quality in the population. The smaller the fitness value is, the worse the current position of the particle is. And the larger the corresponding inertia weight, which means that the particle needs to go beyond the local search to find the global optimal point. Conversely, the larger the fitness value is, the smaller the corresponding inertia weight is, and the particle focuses more on searching for the optimal solution in the local range. This modification in strategy can well balance the local search and global search, which is very beneficial to the convergence of the algorithm.

For the acceleration coefficient, if the value of self-cognitive factor c_1 is low, it indicates that the particle has weak self-cognitive ability, and the algorithm will ignore the optimal position of self-exploration, leading to premature convergence and easy to fall into local optimum. If the value of social cognitive c_2 is low, it indicates the weak social information sharing ability among the particles and slows down the convergence of the population. Therefore, this paper proposes a new adaptive acceleration coefficient.

$$\begin{aligned} c_1 &= 2 \cdot e^{-\frac{t}{t_{\max}}} \\ c_2 &= e^{\frac{t}{t_{\max}}} \end{aligned} \tag{6}$$

where, t is the number of iterations, and $t_{\max}$ is the maximum number of iterations. From (6), it can be seen that the proposed adaptive acceleration coefficient causes the value of c to change nonlinearly as the number of iterations increases. This approach makes the particle position update slower in the late iteration so that the particles are less likely to skip the global optimum position. At the same time, c_1 decreases gradually and c_2 increases gradually, which makes the particles focus more on the social cognitive search in order to find the optimal solution when more and more solutions are obtained from the search in the late iteration.

In this work, the ability of the parameters of the population update to adjust adaptively according to the iteration time and fitness value is exploited to allow the particles to continuously adjust their range and speed of the optimization search according to the real-time changes in the population state until the population converges to a better result. This improvement enhances the performance of particle swarm optimization search.

C. FITNESS FUNCTION AND DYNAMIC CORRECTION OF PARTICLE POSITION

The fitness function is the guidance of the PSO algorithm to search for the global optimal solution. It is the basis of particle updating, so it largely determines the quality of the iterative results. The basic fitness function is to calculate the distance between the samples and the cluster centers. However, it is far from enough to minimize the intra-class distance [23] to get higher accuracy, so many scholars have done a lot of research on how to design a better fitness function. For example, in the literature [15], it is proposed to calculate the fitness function using the classification results. This method can obtain good clustering centers. However, it requires prior knowledge of the classification label information. An internal index PBM of clustering is proposed in the literature [24] to characterize the quality of clustering, which guarantees the formation of a small number of tight clusters and large intervals between at least two clusters. However, this index ignores the effect of the distribution of feature parameters. For this reason, this paper proposes an improved PBM index (IPBM) through as the fitness function.

The IPBM index consists of three components, as defined in (7)

$$\begin{aligned} IPBM &= \frac{1}{K} \cdot \frac{E_1}{E_K} \cdot \frac{1}{D_K} \\ E_K &= \sum_{k=1}^{K} \sum_{j=1}^{N} \left\| x_j - z_k \right\|_2 \\ D_K &= \max_{i,j=1}^{K} \left\| z_i - z_j \right\|_2 \end{aligned} \tag{7}$$

where K is the number of clusters; N is the total number of data sets; D_K is the maximum inter-cluster separation; E_K is the sum of the total intra-cluster scatter between samples and their cluster centers, and E_1 is the sum of the distances between all data samples contained and the centers in only one cluster. This index reduces the proportion of the maximum inter-cluster separation and avoids the adverse impact of discrete points being incorrectly selected as cluster centers in the clustering process. A higher IPBM value indicates a better clustering scheme.

In the standard PSO algorithm, individuals adjust their trajectory to any neighbor member's best position and their previous best performance position by searching space. However, it is pointed out in the literature [25] that the updating method of the topological structure ignored the influence of the clusters, and drew on the idea of K-means clustering. It proposed that the previous best positions of individuals and neighbors are replaced with the average of the intra-class samples by performing cluster analysis of their previous best positions. In this paper, we will draw on the above ideas to improve the accuracy of clustering and reduce the influence of data distribution by dynamic position correction of the updated particles according to cluster analysis. Although the literature [25] chose the average value of samples after each particle clustering to replace the individual best position, the average value of a cluster does not necessarily reflect the distribution of clusters. Therefore, this paper draws on literature [26] and proposes the gravity center index as the basis to determine the clustering centers. The specific principle is: when the particle has a better fitness value, the gravity center index of each sample (including the position of the particle) obtained by the particle position as the clustering centers is calculated. And the one with the largest value of the gravity center index is selected to replace the individual best position. The Gravity Center Index (GCI) is defined as:

$$\begin{aligned} GCI &= num(z) \\ z &= \{X_i \, \|X_i - X_k\|_2 \le r, i \ne k\}; \\ r &= \frac{\sum_{k=1}^{N}\sum_{i=1}^{N} d_{ki}}{n^{coef}}, \quad coef = 0.3 \\ d_{ki} &= \{d_{ki} \mid d_{ki} = \|X_k - X_i\|_2, \\ &\qquad k = 1, 2, \ldots, N, i = 1, 2, \ldots, N\} \\ n &= (N-1)N; \end{aligned} \tag{8}$$

where, z represents the set of samples and their neighbors that are distant r from each other. The index reflects the cohesiveness between samples and can eliminate the influence of discrete points. The gravity center index can also reflect whether the cluster contains other samples that do not belong to this class. This paper believes that if the distance between the two samples is greater than a certain threshold, and the gravity center indexes of the two samples are relatively large, both can be used as new clustering centers, namely, new particle positions. Among them, the threshold setting is generally an integer multiple of r. Finally, all the searched particle positions are combined to the required number of clusters.

In this work, the proposed new fitness function is used to reflect both cluster dispersion and tightness. And the location correction according to gravity center index can adapt to different data distributions. These improve the population location search, and in turn improve the accuracy of the algorithm.

D. ALGORITHM FLOW

The flow chart of the algorithm is shown in Fig.1.

(1) Take radar signal features as input and do normalization;

(2) Initialize the population parameters: the population number N, the maximum number of iterations, the maximum velocity of the particles, the position range of the particles, and the individual historical best fitness value. Population position initialization: one data is randomly selected as a cluster center, and then the remaining cluster centers are selected according to the Maxmin distance principle [27]. The operation is repeated N times to produce N particles (each particle is a K × D dimensional vector, where K is the number of cluster clusters and D is the dimensionality of the data set), and the distance is Euclidean distance;

(3) Each particle divides the dataset with its respective position as the cluster center. The fitness value of each particle is calculated according to (7), and the center of each cluster is used as the position of the particle;

(4) If the fitness value of the current particle is greater than that of the previous, the optimal fitness value F_{pbest} and the optimal position X_{pbest} of the individual are updated; The sub-steps are as follows:

① According to (8), the GCI of each intra-class sample (including the particle's position) obtained by taking the particle position as the clustering center is calculated, and the position of the particle with the largest GCI is selected. If the distance between the two samples is greater than $3 * r$, and the gravity center index of the two samples is larger, it can be used as a new position. Merge all the positions that is close to each other until getting the required number of clusters, namely, complete the position correction to obtain the individual optimal position X_{pbest};

② Reclassify clusters and calculate individual fitness value F_{pbest};

(5) The highest individual fitness value is chosen as the global optimal fitness value F_{gbest}. The corresponding particle position is taken as the global optimal position X_{gbest};

(6) According to Section 3.A, chaotic search is used to randomly disturb the particle positions;

(7) Adjust the inertia weight and acceleration coefficients adaptively according to (5)(6). Update the position and velocity of particles and make them fall into feasible region according to (1);

(8) Determine whether the current population fitness variance is lower than a threshold (indicating that the population has converged) or reaches the maximum number of iterations.

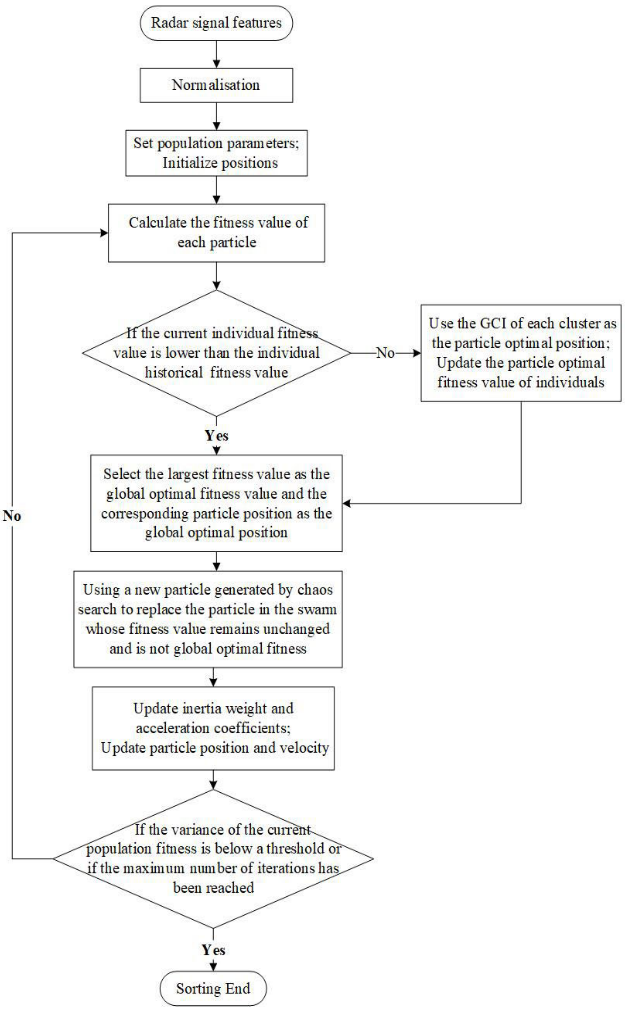


FIGURE 1. DMCPSO algorithm flow.

If yes, the entire algorithm process is completed. Otherwise, turn to (3). The fitness variance is calculated as follows.

$$\text{var} = \frac{1}{n}\sum_{i=1}^{N}\left(f_i - f_{avg}\right)^2 \tag{9}$$

E. TIME COMPLEXITY ANALYSIS

In this subsection, the time complexity of the DMCPSO algorithm is obtained by analyzing the time consumed during the operation of the algorithm. The K-means algorithm, PSO algorithm, DPSOK [14] algorithm, MfPSO [22] algorithm, and IPK-means [18] algorithm are also introduced for comparison and analysis.

The time complexity of K-means algorithm is $O(T_{end} \times N \times K \times D)$; the time complexity of PSO algorithm, DPSOK algorithm, MfPSO algorithm is $O(T_{end} \times N \times K \times D \times P)$; the time complexity of IPK-means algorithm is $O(T_{end} \times N \times K \times D \times P \times t)$; the time complexity of

DMCPSO algorithm is $O(T_{end} \times N \times K \times D \times P \times t)$. Where T_{end} is the number of iterations terminated corresponding to different algorithms; N is the number of data; K is the number of clusters; D is the number of data attributes; P is the number of populations; t is the number of iterations of chaotic search, and t is much smaller than T_{end}.

From the above analysis, it can be seen that the time complexity of the K-means algorithm is the lowest, which also indicates that the K-means algorithm has the feature of simple and fast. The time complexity of the PSO algorithm is higher than that of the K-means algorithm, but it in return has a better optimization effect than the K-means algorithm. Although the DMCPSO algorithm proposed in this paper increases the chaotic search time, the proportion of this time is lower and the iteration termination time is shorter than other latest PSO algorithms. Therefore the proposed algorithm is advanced in terms of timeliness, and subsequent experiments will also be performed to verify it by simulation.

IV. EVALUATION INDICATORS

In order to evaluate the performance of the DMCPSO algorithm, the following clustering evaluation indicators will be used in this paper. For the data set $D = \{x_1, x_2, \ldots, x_m\}$, it is assumed that the cluster division obtained by clustering is $\Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$,and the cluster division given by the reference model is $C = \{c_1, c_2, \ldots, c_K\}$.

(1) Based on homogeneity criterion - Clustering Quality CQ [28]

According to the literature [28], this paper proposes a simplified measure of Clustering Quality (CQ), which is calculated as follows:

$$CQ = \frac{\sum_{i=1}^{K} N_i}{m}, \quad \forall i \in K \tag{10}$$

where, N_i is the number of samples in ω_i that belong to category c_i, which is the homogeneity measure. When ω_i contains only samples from the same category, $CQ = 1$, thus indicating that the larger the clustering quality index, the better the clustering effect, and the higher the correct rate.

(2) Based on the pair counting criterion - Adjusted Rand Index ARI [29]

Let λ, λ^* denote the vector of cluster markers corresponding to Ω, C, respectively. Consider the samples in pairs of two, define:

$$\begin{aligned}
a &= |SS|,\\
SS &= \left\{\left(x_i, x_j\right) \mid \lambda_i = \lambda_j, \lambda_i^* = \lambda_j^*, i < j\right\},\\
b &= |SD|,\\
SD &= \left\{\left(x_i, x_j\right) \mid \lambda_i = \lambda_j, \lambda_i^* \neq \lambda_j^*, i < j\right\},\\
c &= |DS|,\\
DS &= \left\{\left(x_i, x_j\right) \mid \lambda_i \neq \lambda_j, \lambda_i^* = \lambda_j^*, i < j\right\},\\
d &= |DD|,\\
DD &= \left\{\left(x_i, x_j\right) \mid \lambda_i \neq \lambda_j, \lambda_i^* \neq \lambda_j^*, i < j\right\},\\
\textit{Rand}\ \text{Index(RI)} : \text{RI} &= \frac{a+d}{a+b+c+d}\\
ARI &= \frac{RI - E(RI)}{1 - E(RI)}
\end{aligned} \tag{11}$$

Among them, the value range of ARI is [−1,1]. In a broad sense, ARI measures the degree of consistency between the two data distributions. And the larger the value, the more consistent the clustering results are with the real situation.

(3) Based on the mutual information criterion - Normalized Mutual Information NMI [30]

The normalized mutual information NMI is calculated as follows:

$$\begin{aligned} \mathrm{NMI}(\Omega, C) &= \frac{I(\Omega; C)}{(H(\Omega)+H(C))/2} \\ I(\Omega; C) &= \sum_k \sum_j P\left(w_k \cap c_j\right) \log \frac{P\left(w_k \cap c_j\right)}{P\left(w_k\right) P\left(c_j\right)} \\ &= \sum_k \sum_j \frac{\left|w_k \cap c_j\right|}{N} \log \frac{N\left|w_k \cap c_j\right|}{\left|w_k\right|\left|c_j\right|} \\ H(\Omega) &= -\sum_k P\left(w_k\right) \log P\left(w_k\right) \\ &= -\sum_k \frac{\left|w_k\right|}{N} \log \frac{\left|w_k\right|}{N} \end{aligned} \tag{12}$$

where I denotes the increased class information Ω or the decrease of uncertainty under the premise of given class cluster information C. $P\left(w_k\right), P\left(c_j\right), P\left(w_k \cap c_j\right)$ can be regarded as the probability that samples belong to cluster w_k, belongs to category c_j, and belongs to both, respectively. The larger the value, the higher the similarity with the real category information.

(4) Based on matching criterion-centroid Index CI [31]

Centroid Index measures the difference between the two categories by cluster centroids, and the calculation is as follows:

$$\begin{aligned} CI(\Omega, C) &= \sum_{j=1}^{K} orphan(c_j) \\ orphan(c_j) &= \begin{cases} 1, & q_i \neq j \forall i \\ 0, & otherwise \end{cases} \\ q_i &\leftarrow \underset{1 \leq j \leq K}{\arg\min} \|\omega_i - c_j\|^2, \quad \forall i \in [1, K] \\ CI &= \max\{CI(\Omega, C), CI(C, \Omega)\} \end{aligned} \tag{13}$$

where $CI = 0$ denotes that the two clusters have the same structure. The larger the value, the more the number of clusters is allocated, and the worse the clustering effect.

(5) Based on internal evaluation guidelines

① Davies Bouldin Index – DBI [32]

The index takes into account the average proportion of tightness and isolation in all clusters and is calculated as follows:

$$DBI = \frac{1}{k} \sum_{i=1}^{k} \max_{i \neq j} \left(\frac{e_i + e_j}{d_{i,j}} \right) \tag{14}$$

where e_i and e_j are the average Euclidean distances of all samples i and j to their respective centroids, and $d_{i,j}$ is the distance between centroids. The smaller the DBI is, the better the clustering effect is.

② Silhouette Index – SI [33]

The index is defined as:

$$SI = \frac{1}{N} \sum_{i=1}^{N} \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{15}$$

where $a(i)$ is the average Euclidean distance from sample i to the remaining samples in the same cluster; $b(i)$ is the minimum average Euclidean distance between sample i and the samples in the remaining clustering. The higher the SI is, the better the clustering scheme is.

V. SIMULATION EXPERIMENTS ANALYSIS

To verify the effectiveness of the DMCPSO algorithm, the experiments compare the performance of the K-means algorithm, PSO algorithm [34], DPSOK [14] algorithm, MfPSO [22] algorithm, IPK-means [18] algorithm and DMCPSO algorithm on different datasets. The simulation experiment is coded using MATLAB® R2019b, running on a portable computer with AMD Ryzen 7 4800H at 2.9GHz, 8 cores and 16 threads, 8G × 2 RAM and Windows 10 operating system. In the experiments, the parameters involved include the maximum number of iterations $t_{\max}$; population size P; maximum particle velocity $v_{\max}$; inertia weight w; acceleration coefficients c_1 and c_2. To ensure the effectiveness of each algorithm, based on the parameter descriptions mentioned in the literature and simulation experiments, the specific parameter settings involved in each algorithm are shown in Table 1. below.

TABLE 1. Parameter settings.

Method	Parameters
PSO	$t_{\max}$=50;P=10;$v_{\max}$=0.8m/s; $\omega = 0.8$; $c_1 = c_2 = 2.0$;
DPSOK	$t_{\max}$=50;P=10;$v_{\max}$=0.8m/s; $\omega_{\max} = 0.9$; $\omega_{\min} = 0.4$; $c_{\max} = 2.5$; $c_{\min} = 0.5$;
MfPSO	$t_{\max}$=50;P=10;$v_{\max}$=0.8m/s;
IPK-means	$t_{\max}$=50;P=10;$v_{\max}$=0.8m/s
DMCPSO	$t_{\max}$=50;P=10;$v_{\max}$=0.8m/s

A. EXPERIMENT 1

In this experiment, the pulse description word (PDW), which can characterize the signals, will be used for signal sorting to verify the performance of the algorithm in the case of a large overlap of pulse features. In PDW, pulse amplitude (PA) has poor stability and is generally not used as a clustering binning feature. Time of arrival (TOA) does not directly reflect the characteristics of the intercepted signals but is used to extract deeper PRI features to assist in the sorting of the intercepted signals. Therefore, this paper uses three parameters, namely, the direction of arrival (DOA),

TABLE 2. Radar signal characteristic parameters.

Radar emitter	Pulse number	DOA（°） measurement error: ±3	PW（μS） variation range: ±0.05	RF（MHz） variation range: ±100	Modulation mode & PRI（μS）
1	151	59	0.22	1200	Fixed；355
2	85	66	0.29	1050	Fixed；630
3	71	59	0.19	1050	Fixed；753
4	138	60	0.20	1500	Jagged；363/382/395/413
5	60	57	0.29	900	Fixed；887
6	48	64	0.17	900	Fixed；1123
7	354	64	0.22	1500	Jagged；133/145/152/173
8	93	60	0.17	1350	Jitter；550, Jitter ratio:10%

radio frequency (RF), and pulse width (PW), for sorting. In order to effectively evaluate the signal sorting performance of the DMCPSO algorithm in complex electromagnetic environments, the construction of electromagnetic scenarios is particularly important. In this section, sorting is performed using eight sets of mixed sequences generated by the PDW radar signal generator with different degrees of overlapping characteristic parameters and varying pulse numbers. Among them, the DOA is not affected by the radar emitter itself, and the DOA value measured by the receiver is basically constant, but sometimes affected by the measurement error. Therefore, based on the DOA parameters of each radar signals, a random number is added to make the deviation less than 3°. PW, RF these two parameters have a certain range of changes. The radar characteristics parameters are shown in Table 2.

In order to simulate the distribution of characteristic parameters of radar signals intercepted in the complex electromagnetic environment, the radar signal characteristic parameters constructed in this paper have different proportions of overlap, as shown in Fig. 2. This phenomenon is particularly evident in some radiation sources. For example, for similar signal sources 4 and 7 with the same range of RF variation, the ranges of DOA variation are 57.51°-63.31° and 60.83°-67.05°, respectively, and the ranges of PW variation are 0.14μS-0.25μS and 0.17μS-0.28μS, respectively. The overlap ratios of these two characteristic parameters are 41.3% and 98.7%, respectively. At the same time, this paper sets the PRI and modulation mode of the radar emitter to adjust the number of pulses of the corresponding emitter in the intercept time.

From Fig. 2, it is obvious that there are different degrees of overlap between these three characteristic parameters for radar pulse signals, and some data have serious overlap. The number of pulse signals in each class is also different, and the data are widely distributed and poorly aggregated, which may be less effective using traditional clustering methods.

Fig. 3 below shows the graph of the clustering results obtained using the DMCPSO algorithm, demonstrating the effect of the radar signal features on the sorting results under different overlaps. The red dots indicate misclassified data.

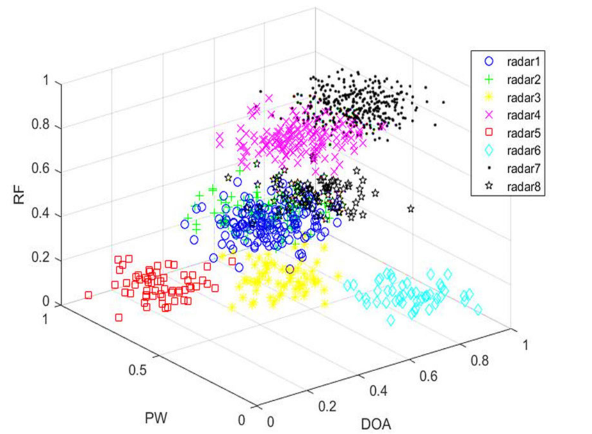


FIGURE 2. Signal characteristics distribution.

TABLE 3. Running time of the algorithm.

	PSO	DPSOK	MfPSO	IPK-Means	DMCPSO
Running Time	0.79	0.45	0.39	1.54	0.33

As can be seen from Fig. 3, most of the pulse data can be correctly clustered, and only in the case of very serious overlap and discrete distribution of few data does the incorrect classification occur, and the algorithm has excellent performance. To further demonstrate the superiority of the algorithms, the convergence times of different algorithm iterations are analyzed in Table 3, and the curve of fitness variance with the number of iterations is plotted in Fig. 4.

As can be seen from Table 3 and Fig. 4, the fitness variance of all algorithms decreases as the number of iterations increases, and the variation of the fitness values gradually decreases towards stability. The algorithms find the optimum by converging continuously during the iterative process. However, there is a large gap between their running time and convergence speed. Among them, the DMCPSO algorithm has the fastest convergence speed and the smallest stable variance, which is significantly better than other algorithms, indicating that the group can quickly find the optimal position in the iterative process, and then converge to a satisfactory

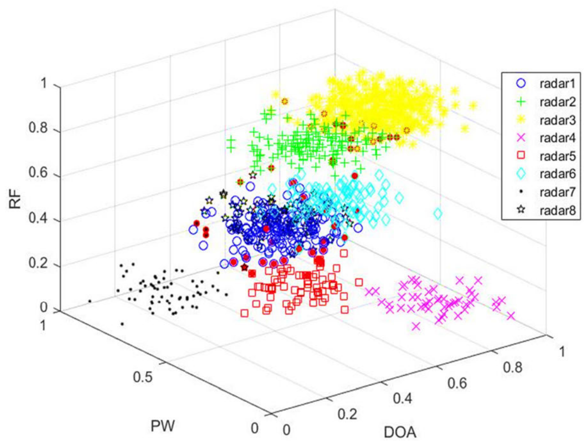


FIGURE 3. DMCPSO clustering result.

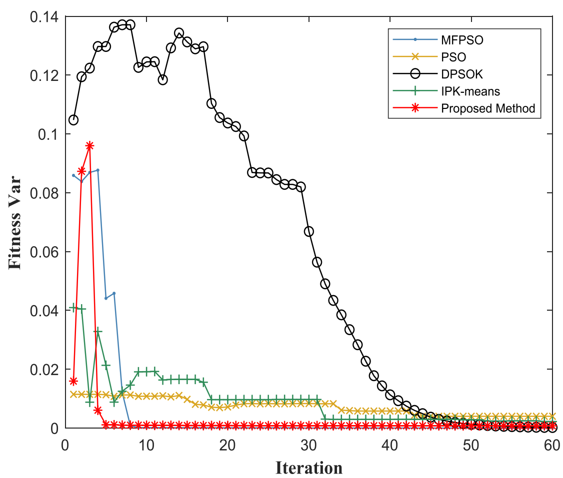


FIGURE 4. Fitness variance curve.

TABLE 4. Clustering evaluation indexs under different algorithms.

Dataset	Method	CQ	ARI	NMI	CI	SI	DBI
Radar feature set	K-means	85.61%	0.7728	0.8627	2.0	0.5562	0.9139
	PSO	88.55%	0.7774	0.8710	1.5	0.5788	0.8700
	DPSOK	88.81%	0.7825	0.8697	1.4	0.5779	0.8859
	MfPSO	89.40%	0.7753	0.8638	1.6	0.5844	0.8758
	IPK-means	87.37%	0.7720	0.8639	1.8	0.6057	0.8318
	DMCPSO	**94.99%**	**0.7994**	**0.8929**	**0.0**	**0.6424**	**0.8116**

fitness value. The experiment also verifies the conclusions of the time complexity analysis in Section 3.E, and the algorithm's timeliness can be well applied in the field of signal sorting.

To further illustrate the effectiveness of the algorithm, this paper will validate the algorithm with several clustering evaluation indexes mentioned in Section 4. The simulation is averaged over several experiments, and the specific results are shown in Table 4.

It can be seen from Table 4 that although the K-means algorithm is simple and converges quickly, its accuracy and matching degree are different from those of the PSO series algorithms, which indicates that the PSO can well deal with complex radar data. Compared with other latest improved PSO algorithms, the DMCPSO algorithm has significant advantages in various indicators: the CQ value of the DMCPSO algorithm reaches 94.99%, indicating that the clustering accuracy is the highest, and the classification data tend to the label data most, which is hardly affected by discrete data; ARI value is the largest, indicating that the clustering results are more consistent with the real situation, and the probability of correct decision is the largest; the NMI value is the largest, which indicates that the uncertainty of the category information is lowest, and the relationship between the categorized data and the labeled data is closest; the CI drops to 0, which indicates that there is a better match between the classification data and the labeled data, with exactly the same structure between the two; SI is the largest, indicating that the cohesion and separation between samples have higher values after clustering; DBI is the smallest, indicating that the intra-class distance is the smallest and the inter-class distance is the largest after clustering.

The above simulation results prove that the DMCPSO algorithm obtains the best sorting results from all angles for radar simulation data with high pulse overlap and small number of some pulses.

B. EXPERIMENT 2

In this experiment, according to the features of the intercepted emitter signals that are dense and the overlapping characteristic parameters, several representative data similar to the distribution of radar feature parameters are selected for verifying the performance of the proposed algorithm in the commonly used clustering data sets [35]. Although these standard datasets have only two-dimensional features and contain less information about the features compared to the PDW streams used for radar sorting, these standard datasets have high complexity and can simulate the distribution of pulse signals in complex electromagnetic space to some extent. The two-dimensional characteristics of the normalized standard data sets in this experiment can be regarded as PW and RF features of the same normalized radar features. The following Table 5 shows the selected distribution features for the different datasets.

TABLE 5. Data sets distribution.

Dataset	Varying	Overlap	Size	Clusters	Per cluster
A(a1-a3)	Number of clusters	20%	3000–7500	20,30,50	150
S(s1-s4)	Overlap	9%,22%,41%, 44%	5000	15	333
Unbalance	Balance	0%	6500	8	100-2000

From the datasets a1 to a3, the number of clusters is gradually increasing, which can be characterized by the increasing density of radar radiation source signals in electromagnetic space. From the datasets s1 to s4, the degree of data overlap is

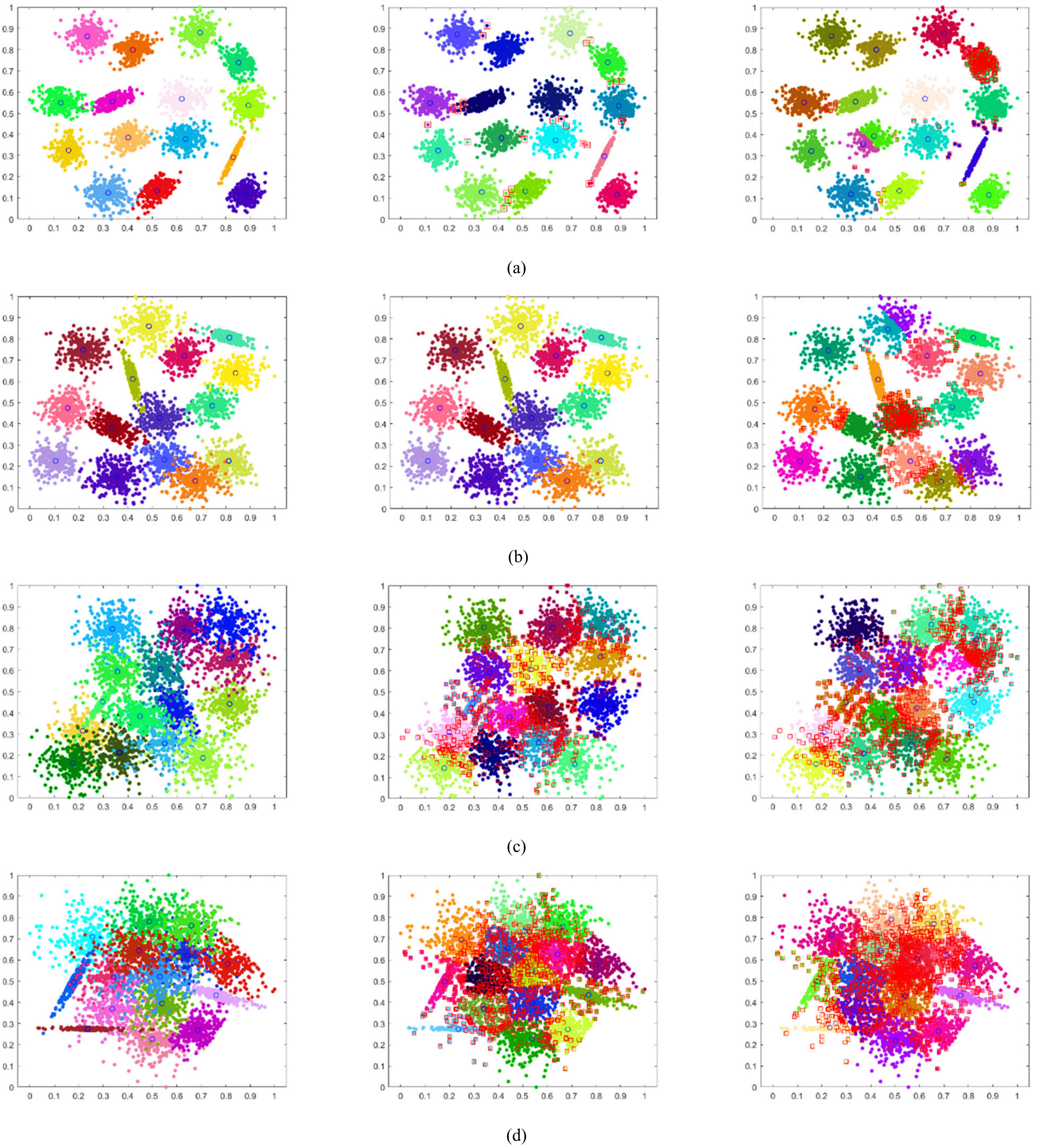


FIGURE 5. (a)-(d) is the clustering results of different algorithms in the s1-s4 datasets. The first column is the distribution of the original data; the second column is the clustering results of the DMCPSO algorithm proposed in this paper; the third column is the clustering results of the DPSOK algorithm.

gradually increasing, which can characterize the overlapping degree of pulse measurement parameters of each radiation source signal. The dataset Unbalance can characterize the uneven distribution of radiation source signals during the interception time, resulting in quantitative imbalance.

This experiment compares the performance of the K-means algorithm, PSO algorithm, DPSOK algorithm, MfPSO algorithm, IPK-means algorithm, and the DMCPSO algorithm on this standard dataset. The following Fig. 5 records the clustering results of the algorithm under the S datasets and shows the clustering effect by taking the DPSOK algorithm and DMCPSO algorithm as examples. Clusters of different colors and shapes represent the set composed of different emitter signal features. The red boxes indicate the data with clustering errors, and the blue circles are the clustering centers.

It can be seen from Fig. 5, as the overlapping degree of datasets from s1 to s4 gradually increases, the difficulty

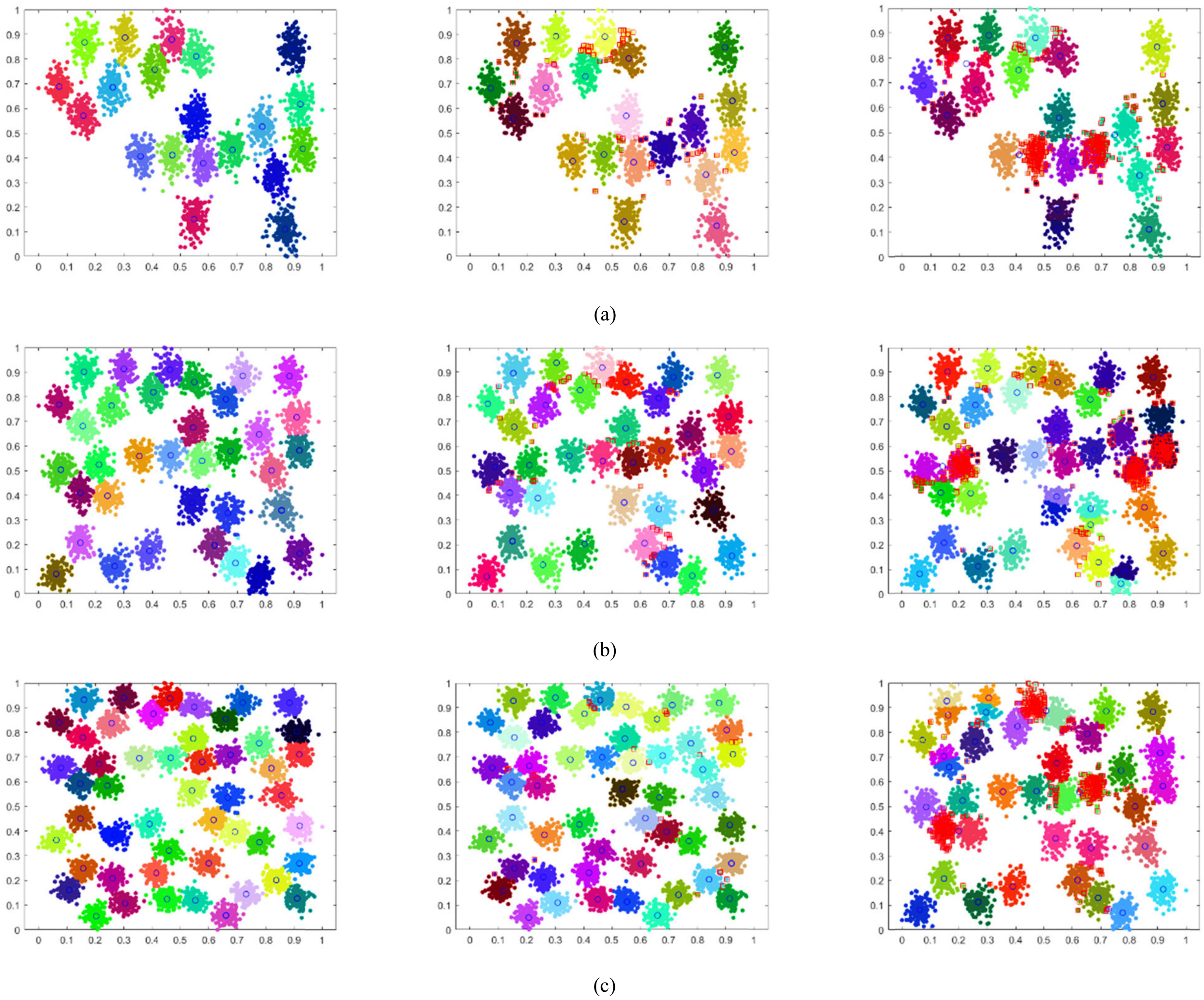


FIGURE 6. (a)-(c) is the clustering results of different algorithms in the a1-a3 datasets. The first column is the distribution of the original data; the second column is the clustering results of the DMCPSO algorithm proposed in this paper; the third column is the clustering results of the DPSOK algorithm.

of clustering increases greatly, and many misclassified data appear. However, unlike DPSOK, the DMCPSO algorithm only fails to distinguish effectively on the discrete points that are far from the center of the true class, and most of the data belonging to one class can be correctly clustered. Some of these discrete points are evenly distributed within the range of other classes, so it is difficult to distinguish them. Outliers can characterize the errors of radar characteristic parameters in measurement, and these errors are allowed for radar signal sorting. However, the DPSOK algorithm cannot even classify accurately on the S datasets, and the increase in CI values causes a significant decrease in the correct rate, which can make the radar system misjudge radiation sources with similar characteristics, leading to the subsequent signal processing process using the wrong operation and making the wrong strategic decisions.

Fig. 6 also shows the effect of algorithmic clustering under A datasets using the DPSOK algorithm and the DMCPSO algorithm as examples.

As can be seen from Fig. 6, with the number of clusters gradually increasing from a1 to a3 datasets, the DMCPSO algorithm has been able to classify accurately, and only a few points are incorrectly clustered, but the proportion is negligible. It shows that the algorithm is insensitive to the number of clusters and can achieve 100% clustering. For the DPSOK algorithm, however, as the number of clusters increases, the more data are clustered incorrectly, the worse the result is, and the similarity to the true category is linearly decreasing with the number of clusters.

Fig. 7 also shows the effect of algorithmic clustering under Unbalance dataset using the DPSOK algorithm and the DMCPSO algorithm as examples.

It can be seen from Fig. 7 that the class of Unbalance dataset is extremely unbalanced and the separation between clusters is large, which undoubtedly increases the difficulty of clustering. Due to the improvement of the DMCPSO algorithm, it always searches for the best clustering until the best centroid is found. So the dataset can always achieve

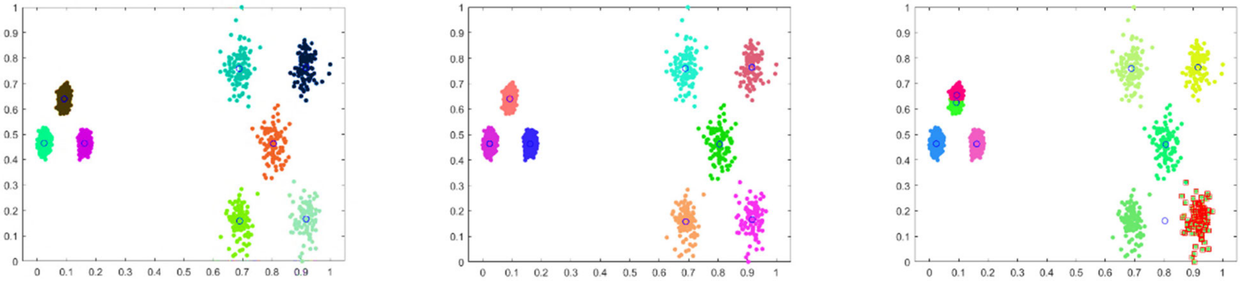


FIGURE 7. Clustering results of different algorithms under Unbalance dataset. The first column is the distribution of the original data; the second column is the clustering results of the DMCPSO algorithm proposed in this paper; the third column is the clustering results of the DPSOK algorithm.

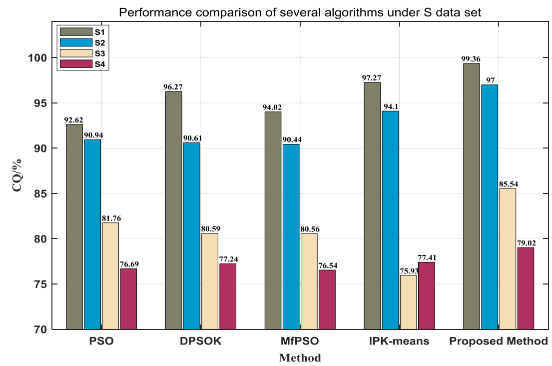
excellent results without error classification. In contrast, for the DPSOK algorithm, if the clustering center is chosen incorrectly at the beginning of the iteration, it will be difficult for the particles to find the correctly classified position by shifting during the later iterations.

Therefore, by visually observing the graphs of the clustering results, it can be seen that the DMCPSO algorithm has excellent performance and is able to solve a wide range of datasets that are difficult to classify. Based on the analysis in the previous subsection 5.A on the radar dataset, this experiment will quantitatively analyze the clustering effect in terms of various clustering indexes, as shown in Table 6 below.

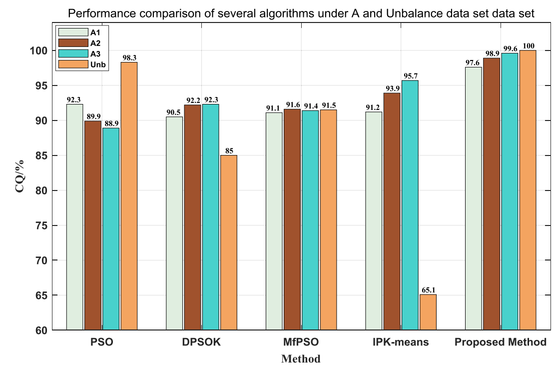
In order to better show the performance of the DMCPSO algorithm, this paper will intuitively illustrate the problem by histogram based on the clustering quality CQ of radar signal sorting technology concerned by the table above.

It can be seen from Table 6 and Fig. 8 above that on these standard data, the DMCPSO algorithm proposed in this paper can achieve good clustering results compared with other particle swarm optimization algorithms. For the S datasets, the performance of the algorithm decreases slightly as the overlap of the data increases, but for the radar processing system, separating the signal with a sufficiently high overlap and the result with a small difference from the true category can already be adapted to the complex electromagnetic environment. For the A datasets, with the increase of the number of clusters, the performance of other algorithms is declining. While the DMCPSO algorithm has always maintained the best effect, and the classification results are basically correct. This undoubtedly solves a major problem of signal sorting for the increasingly dense electromagnetic environment. For the Unbalance dataset, due to the serious imbalance of categories and the separation of each cluster, other algorithms do not locate the center of the class correctly and will be distributed in dense areas. However, the DMCPSO algorithm uses a new fitness function and position correction to accurately classify each cluster.

In summary, from the various clustering index parameters, the DMCPSO algorithm has the highest clustering quality, matching, similarity, and coincidence with the real classes among all algorithms. And the intra-class cohesion and the inter-class separation make the clustering effect closer to the real cluster distribution. The increase in the number of clusters, the overlap in the distribution of feature parameters, the imbalance in the distribution, and the presence of outliers and the data distribution have no significant impact on the algorithm.



(a)



(b)

FIGURE 8. Clustering results of different algorithms under different datasets.

In addition to the excellent clustering effect, we hope that in the case of better clustering indexes, the algorithm can also have a short running time to deal with the electromagnetic environment that has more stringent real-time requirements. Table 7 compares the running time of different algorithms.

It can be seen from Table 7, the running time of the DMCPSO algorithm is relatively short under different datasets with high enough accuracy, which can meet the real-time requirements well.

Given the specificity of the application scenario of the radar signal sorting algorithm, we hope that the excellent

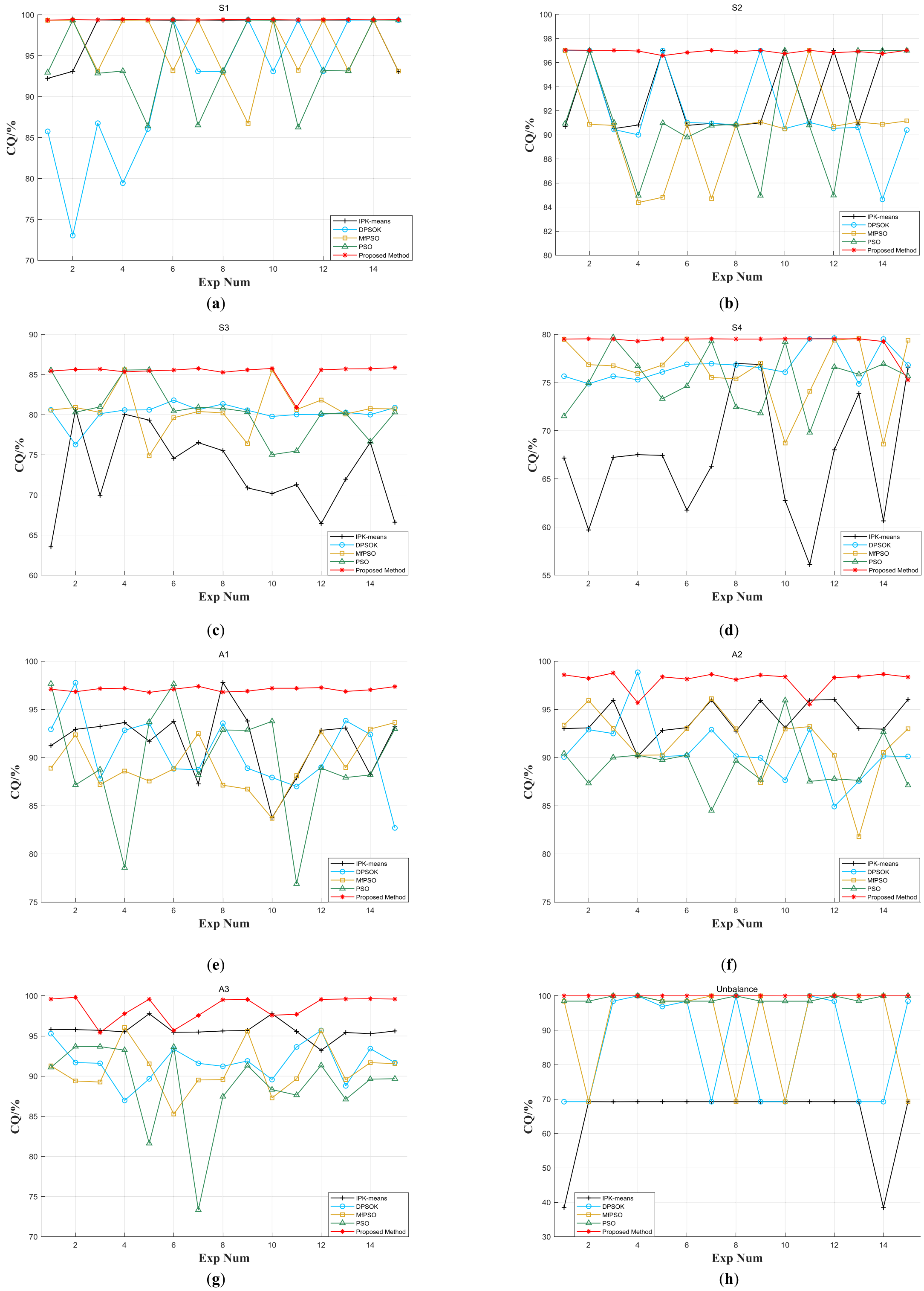


FIGURE 9. Comparison of the stability of different algorithms under different datasets. (a)-(d) are the results under s1-s4 datasets; (e)-(g) are the results under a1-a3 datasets; (h) are the results under unbalance dataset.

performance of the algorithm is not a coincidental phenomenon. A single sorting error is likely to bring about an inestimable strategic loss, so the stability and robustness of the sorting algorithm are important factors that must be considered in practical applications. Fig. 9 below records the clustering quality CQ curve of the algorithm after 15 runs of the program.

As shown in Fig. 9, under different datasets, the DMCPSO algorithm shows strong stability and robustness while maintaining the accuracy advantage compared to the substantial

TABLE 6. Clustering evaluation indexs under different algorithms.

Dataset	Method	CQ	ARI	NMI	CI	SI	DBI
s1	PSO	94.25%	0.9305	0.9663	1.4	0.8304	0.4738
	DPSOK	96.27%	0.9334	0.9778	1.0	0.8532	0.4358
	MfPSO	94.02%	0.9291	0.9684	1.1	0.8175	0.5161
	IPK-means	97.27%	0.9347	0.9790	0.4	0.8555	0.4250
	DMCPSO	**99.36%**	**0.9374**	**0.9863**	**0.0**	**0.8805**	**0.3655**
s2	PSO	91.67%	0.9264	0.9280	1.7	0.7561	0.5693
	DPSOK	90.61%	0.9249	0.9202	1.5	0.7284	0.6223
	MfPSO	90.44%	0.9248	0.9222	2.0	0.7471	0.5898
	IPK-means	94.10%	0.9296	0.9356	1.0	0.7684	0.5371
	DMCPSO	**97.00%**	**0.9336**	**0.9465**	**0.0**	**0.8009**	**0.4654**
s3	PSO	81.76%	0.9140	0.7841	1.6	0.6441	0.7126
	DPSOK	80.59%	0.9115	0.7787	1.8	0.6310	0.7379
	MfPSO	80.56%	0.9195	0.7756	1.8	0.6252	0.7345
	IPK-means	75.93%	0.8991	0.7560	3.3	0.5982	0.7765
	DMCPSO	**85.54%**	**0.9193**	**0.7945**	**0.0**	**0.6672**	**0.6432**
s4	PSO	75.24%	0.9062	0.7082	1.5	0.6161	0.7190
	DPSOK	77.24%	0.9185	0.7142	1.0	0.6338	0.6758
	MfPSO	76.54%	0.9104	0.7194	1.3	0.6346	0.6953
	IPK-means	77.41%	0.8928	0.6829	3.1	0.5771	0.7723
	DMCPSO	**79.02%**	**0.9132**	**0.7174**	**0.2**	**0.6339**	**0.6705**
a1	PSO	92.25%	0.9449	0.9458	2.2	0.7127	0.6313
	DPSOK	90.50%	0.9432	0.9471	3.0	0.7223	0.6236
	MfPSO	91.08%	0.9440	0.9438	2.6	0.7166	0.6248
	IPK-means	91.24%	0.9461	0.9567	1.0	0.7543	0.5859
	DMCPSO	**97.60%**	**0.9503**	**0.9627**	**0.0**	**0.7563**	**0.5726**
a2	PSO	89.92%	0.9666	0.9546	6.1	0.7078	0.6503
	DPSOK	92.17%	0.9680	0.9624	4.5	0.7343	0.6080
	MfPSO	91.61%	0.9669	0.9574	4.8	0.7136	0.6410
	IPK-means	93.94%	0.9702	0.9734	4.4	0.7546	0.5757
	DMCPSO	**98.90%**	**0.9718**	**0.9838**	**0.0**	**0.7754**	**0.5432**
a3	PSO	88.85%	0.9760	0.9617	9.0	0.6936	0.6771
	DPSOK	92.32%	0.9775	0.9720	7.0	0.7293	0.6343
	MfPSO	91.44%	0.9772	0.9686	7.5	0.7223	0.6529
	IPK-means	95.71%	0.9793	0.9883	3.8	0.7669	0.5653
	DMCPSO	**99.55%**	**0.9803**	**0.9939**	**0.0**	**0.7939**	**0.5186**
unb	PSO	98.26%	0.8089	0.9903	2.0	0.8574	0.5127
	DPSOK	85.02%	0.8091	0.9917	1.5	0.8528	0.5079
	MfPSO	91.48%	0.7589	0.9508	0.8	0.9022	0.4290
	IPK-means	65.12%	0.5570	0.7704	2.0	0.8209	0.5221
	DMCPSO	**100.00%**	**0.8095**	**1.0000**	**0.0**	**0.9585**	**0.3578**

TABLE 7. Running time of different algorithms with different datasets.

	Method	S1	S2	S3	S4	A1	A2	A3	Unb
Running Time(s)	PSO	26.71	26.46	24.81	25.84	15.29	71.07	190.99	23.11
	DPSOK	18.18	18.01	15.32	15.43	10.73	47.05	130.91	15.65
	MfPSO	6.78	11.49	3.61	2.77	5.06	41.76	135.35	9.44
	IPK-means	1.05	2.28	8.63	10.58	0.75	2.41	5.14	2.25
	DMCPSO	**0.86**	**0.91**	**1.14**	**3.16**	**0.64**	**1.40**	**16.02**	**3.67**

fluctuations in the performance of other algorithms, which is of great significance for the practical application of radar signal sorting algorithms.

VI. CONCLUSION

The main difficulty of signal sorting technology is that the intercepted pulse signals have a large density, complex form, and variable parameter distribution. The traditional sorting methods based on clustering cannot get good results. Therefore, this paper applies the PSO algorithm to the field of radar signal sorting and proposes a DMCPSO sorting algorithm in view of the shortcomings of the PSO algorithm, such as premature convergence and insufficient optimization ability. In the iterative process, chaotic disturbance is added to replace the individuals whose performance remains unchanged in the particle swarm, which increases the diversity of the population and improves the ability of the algorithm to jump out of local solution. The inertia weight and acceleration coefficients are adaptively adjusted to update the position and velocity of the particles, allowing the particles to perceive the change of the population in real-time and find a balance in global search and local search. Besides, on this basis, the algorithm adopts a new fitness function. In the construction of the fitness function, the influence of sample distribution and the distance between the sample and its corresponding center are considered, which reduces the possibility of wrong clustering because of the sample distribution. At the same time, the dynamic correction of particle position is carried out, so that the individual can continuously find the optimal clustering center through clustering analysis, and avoid the interference caused by the distribution of feature parameters to obtain better performance.

To evaluate the performance of the proposed algorithm, the mixed sequence with different degrees of overlapping characteristic parameters and varying number of pulses is used for sorting. Compared with several other improved PSO algorithms, it has great advantages in the accuracy and efficiency of sorting. In addition, several datasets with a similar distribution of characteristic parameters of radar pulse signals are also selected from the benchmark datasets for simulation validation in this paper. The same comparison with other algorithms in several common and new clustering performance indexes is performed to verify the convergence speed and accuracy of the algorithm and to analyze the stability and robustness of the algorithm. All these simulation results show that the DMCPSO algorithm has faster convergence speed, higher sorting accuracy, and stronger stability, which is suitable for complex electromagnetic environment perception and has high application value.

REFERENCES

[1] S. Q. Wang, G. P. Hu, Q. L. Zhang, C. Y. Gao, and T. Cai, "The background and significance of radar signal sorting research in modern warfare," *Procedia Comput. Sci.*, vol. 154, pp. 519–523, Jan. 2019.

[2] W. Jiang and X. Fu, "Improved algorithm for de-interleaving radar signals with overlapping features in the dynamically varying electromagnetic environment," *IET Radar, Sonar Navigat.*, vol. 14, no. 9, pp. 1328–1337, Sep. 2020.

[3] Y. Liu and Q. Zhang, "Improved method for deinterleaving radar signals and estimating PRI values," *IET Radar, Sonar Navigat.*, vol. 12, no. 5, pp. 506–514, May 2018.

[4] M. Bagheri and M. H. Sedaaghi, "A new method for detecting jittered PRI in histogram-based methods," *Turkish J. Elect. Eng. Comput.*, vol. 26, no. 3, pp. 1214–1224, 2018.

[5] Z. Ge, X. Sun, W. Ren, W. Chen, and G. Xu, "Improved algorithm of radar pulse repetition interval deinterleaving based on pulse correlation," *IEEE Access*, vol. 7, pp. 30126–30134, 2019.

[6] A. W. Ata'a and S. N. Abdullah, "Deinterleaving of radar signals and PRF identification algorithms," *IET Radar, Sonar Navigat.*, vol. 1, no. 5, pp. 340–347, 2007.
[7] M. Li, M. He, J. Han, and Y. Tang, "A new clustering and sorting algorithm for radar emitter signals," *J. Phys., Conf. Ser.*, vol. 1617, Aug. 2020, Art. no. 012009.
[8] M. G. S. Ahmed and B. Tang, "Sorting radar signal from symmetry clustering perspective," *J. Syst. Eng. Electron.*, vol. 28, no. 4, pp. 690–696, 2017.
[9] T. N. Reddy and K. P. Supreethi, "Optimization of K-means algorithm: Ant colony optimization," in *Proc. Int. Conf. Comput. Methodologies Commun. (ICCMC)*, Erode, India, Jul. 2017, pp. 530–535.
[10] Y. Xue, J. Jiang, B. Zhao, and T. Ma, "A self-adaptive artificial bee colony algorithm based on global best for global optimization," *Soft Comput.*, vol. 22, no. 9, pp. 2935–2952, May 2018.
[11] T. T. Ngo, A. Sadollah, and J. H. Kim, "A cooperative particle swarm optimizer with stochastic movements for computationally expensive numerical optimization problems," *J. Comput. Sci.*, vol. 13, pp. 68–82, Mar. 2016.
[12] X. Gong, L. Liu, S. Fong, Q. Xu, T. Wen, and Z. Liu, "Comparative research of swarm intelligence clustering algorithms for analyzing medical data," *IEEE Access*, vol. 7, pp. 137560–137569, 2019, doi: 10.1109/ACCESS.2018.2881020.
[13] H. Li, H. He, and Y. Wen, "Dynamic particle swarm optimization and K-means clustering algorithm for image segmentation," *Optik*, vol. 126, no. 24, pp. 4817–4822, Dec. 2015, doi: 10.1016/j.ijleo.2015.09.127.
[14] W. Xiaoqiong and Y. E. Zhang, "Image segmentation algorithm based on dynamic particle swarm optimization and K-means clustering," *Int. J. Comput. Appl.*, vol. 42, no. 7, pp. 649–654, Oct. 2020, doi: 10.1080/1206212X.2018.1521090.
[15] C. Ratanavilisagul, "Modification fitness function of particle swarm optimization to improve the cluster centroid," in *Proc. Int. Conf. Eng. Ind. Technol. (ICEIT)*, Pattaya, Thailand, vol. 965, Sep. 2020, pp. 11–13.
[16] W. Yan, M.-J. Li, Y.-C. Zhong, C.-Y. Qu, and G.-X. Li, "A novel k-MPSO clustering algorithm for the construction of typical driving cycles," *IEEE Access*, vol. 8, pp. 64028–64036, 2020, doi: 10.1109/ACCESS.2020.2985207.
[17] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: An overview," *Soft Comput.*, vol. 22, no. 2, pp. 387–408, Jan. 2018.
[18] T. Shenwei and J. Ruiyu, "K-means cluster algorithm based on improved PSO," *Comput. Eng. Appl.*, vol. 55, no. 18, pp. 140–145, 2019.
[19] Y. Wei, Y. Dai, Y. Zhang, J. Chen, and J. Ding, "Adaptive chaotic embedded particle swarm optimization algorithm based on tent mapping," *Comput. Eng. Appl.*, vol. 49, no. 10, pp. 45–49, 2013.
[20] L. Shan, H. Qiang, J. Li, and Z. Q. Wang, "Chaotic optimization algorithm based on tent map," *Control Decis.*, vol. 20, no. 2, pp. 179–182, 2005.
[21] Z.-H. Zhan, J. Zhang, Y. Li, and H. S.-H. Chung, "Adaptive particle swarm optimization," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 39, no. 6, pp. 1362–1381, Dec. 2009, doi: 10.1109/TSMCB.2009.2015956.
[22] S. Paul, S. De, and S. Dey, "A novel approach of data clustering using an improved particle swarm optimization based K—Means clustering algorithm," in *Proc. IEEE Int. Conf. Electron., Comput. Commun. Technol. (CONECCT)*, Bengaluru, India, Jul. 2020, pp. 1–6, doi: 10.1109/CONECCT50063.2020.9198685.
[23] 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, doi: 10.1109/CEC.2003.1299577.
[24] M. K. Pakhira, S. Bandyopadhyay, and U. Maulik, "Validity index for crisp and fuzzy clusters," *Pattern Recognit.*, vol. 37, no. 3, pp. 487–501, Mar. 2004.
[25] J. Kennedy, "Stereotyping: Improving particle swarm performance with cluster analysis," in *Proc. Congr. Evol. Comput. (CEC)*, La Jolla, CA, USA, vol. 2, 2000, pp. 1507–1512.
[26] A. Amini, H. Saboohi, and T. Y. Wah, "A multi density-based clustering algorithm for data stream with noise," in *Proc. IEEE 13th Int. Conf. Data Mining Workshops*, Dec. 2013, pp. 1105–1112, doi: 10.1109/ICDMW.2013.170.
[27] Z. Yang and K. Luo, "Improved clustering algorithm based on particle swarm optimization," *Appl. Res. Comput.*, vol. 31, no. 9, pp. 2597–2599, 2014.
[28] K. Gençol, A. Kara, and N. At, "Improvements on deinterleaving of radar pulses in dynamically varying signal environments," *Digit. Signal Process.*, vol. 69, pp. 86–93, Oct. 2017.
[29] L. Hubert and P. Arabie, "Comparing partitions," *J. Classification*, vol. 2, no. 1, pp. 193–218, Dec. 1985, doi: 10.1007/BF01908075.
[30] T. O. Kvalseth, "Entropy and correlation: Some comments," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, no. 3, pp. 517–519, May 1987, doi: 10.1109/TSMC.1987.4309069.
[31] P. Fränti, M. Rezaei, and Q. Zhao, "Centroid index: Cluster level similarity measure," *Pattern Recognit.*, vol. 47, no. 9, pp. 3034–3045, Sep. 2014.
[32] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979, doi: 10.1109/TPAMI.1979.4766909.
[33] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, Nov. 1987.
[34] M. Zhao, H. Tang, J. Guo, and Y. Sun, "Data clustering using particle swarm optimization," in *Future Information Technology* (Lecture Notes in Electrical Engineering), vol. 309. Berlin, Germany: Springer, 2014, doi: 10.1007/978-3-642-55038-6_95.
[35] P. Fränti. (2015). *Clustering Datasets*. Accessed: May 4, 2017. [Online]. Available: http://cs.uef.fi/sipu/datasets/

XIAOYAN WANG was born in 1997. She received the B.Eng. degree in electronic science and technology from Zhengzhou University, China, in 2019. She is currently pursuing the master's degree with the School of Information and Electronics, Beijing Institute of Technology, Beijing, China.

Her research interests include radar signal processing, anti-jamming technology, and radar signal sorting.

XIONGJUN FU (Member, IEEE) was born in 1978. He received the B.Eng. and Ph.D. degrees from the Beijing Institute of Technology (BIT), Beijing, China, in 2000 and 2005, respectively.

He is currently the Vice Dean of the School of Information and Electronics, BIT. He is a Professor and a Ph.D. Supervisor with BIT. His current research interests include radar systems, radar signal processing, waveform design, and automatic target recognition.

JIAN DONG received the B.Eng. degree in signal and information processing and the Ph.D. degree in navigation, guidance, and control from the Beijing Institute of Technology (BIT), Beijing, China, in 2006 and 2009, respectively.

He is currently a Lecturer and a Master Candidate Supervisor with Shijiazhuang Campus, Army Engineering University. His research interests include radar signal processing, inverse synthetic aperture radar imaging, waveform design, automatic target recognition, and micro-Doppler feature extraction.

JIAHUAN JIANG was born in 1997. He received the B.Eng. degree in communication engineering from Qingdao University, China, in 2019. He is currently pursuing the master's degree with the School of Information and Electronics, Beijing Institute of Technology, Beijing, China.

His research interests include anti-jamming technology, SAR image detection, and deep learning.