IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 21 August 2022, accepted 25 August 2022, date of publication 1 September 2022, date of current version 23 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3203695

RESEARCH ARTICLE

Clustering of Typical Wind Power Scenarios Based on K-Means Clustering Algorithm and Improved Artificial Bee Colony Algorithm

GANG YAO, YANG WU, XIAOXU HUANG, QINFENG MA, AND JIANG DU
Power Dispatch Control Center, Guizhou Power Grid Company Ltd., Guiyang 550002, China

Corresponding author: Gang Yao (395139559@qq.com)

This work was supported by the Key Science and Technology Project of China Southern Power Grid Company Ltd. under Grant GZKJXM20210372.

ABSTRACT An improved artificial bee colony algorithm (IABC) was proposed to solve the problems of the k-means clustering (KMC) algorithm, such as poor global search ability, sensitive selection of initial cluster center, randomness of initialization, precocity and slow convergence of the original artificial bee colony (ABC) algorithm. To improve the efficiency of iterative optimization process, a fitness function adapted to KMC algorithm and a position updating formula based on global guidance were constructed. By comparing the improved artificial bee colony algorithm with the original artificial bee colony algorithm and particle swarm optimization algorithm, it is confirmed that IABC algorithm converges speed block and overcomes the shortcoming of the original algorithm which is easy to fall into local optimal solution. IABC algorithm is combined with KMC to get better clustering effect, and the algorithm is used to select typical wind power output scenarios, which plays an important role in actual production.

INDEX TERMS Artificial bee colony (ABC) algorithm, k-means clustering (KMC) algorithm, fitness function, position update, wind power forecast, typical scenario.

I. INTRODUCTION

The safety and reliability of energy supply are the precondition and demand for the smooth operation and development of national economy. As the supply of traditional fossil energy grows increasingly prominent and the natural environment continues to deteriorate, countries all over the world are vigorous developing clean and renewable energy [1]. It has become a significant decision for many countries to increase the proportion of clean energy in the national energy structure and get rid of the dependence on fossil energy. With the improvement if power generation economy, wind power in China will grow rapidly. Newly install wind power capacity of about 70-140 GW/year. China will usher in the large-scale construction peak of wind power in the next decade [2], [3].

Wind energy is the largest clean energy developed and utilized at present. Since it's low space-time energy density, non-enrichment, non-transportability and incapacity of transportation and storage, it must be converted into electric energy [4]. However, due to the intermittence, volatility and randomness of wind power output, there is a large amount of wind abandoning phenomenon at present. The wind energy is seriously wasted, which has a negative impact on the long-term development of wind power. Because the accurate wind power output scenario is crucial to the security and economy of grid, to describe the randomness of wind power putout, it is of vital to select wind power output in typical scenarios [5].

Cluster analysis is an important data analysis technology, which tries to divide physical or abstract sets into similar object classes, so that objects in the same group have a high degree of similarity, and there are large differences between objects in different groups [6].

Swarm intelligence algorithm is an artificial intelligence algorithm that simulates the behavior of biological groups. Common algorithms include ant colony algorithm, particle swarm algorithm and genetic algorithm. Karaboga proposed artificial bee colony algorithm in 2005 [7], [8], which has simple concept, easy implementation and few control parameters.

The associate editor coordinating the review of this manuscript and approving it for publication was Ton Duc Do.

Swarm intelligence algorithm is widely used in the field of clustering because of its powerful global search ability. Reference [9] proposes a new information learning artificial bee colony algorithm (ILABC) for opportunity informatics, which can dynamically adjust the size of subgroups to improve efficiency. Reference [10] proposed an improved KD-ABC algorithm, which changed the way of nectar source generation. Reference [11] combined ABC and K-means to improve the effectiveness of wind farm clustering. In reference [12], artificial bee colony (MOABC) algorithm was used for multi-objective optimization to achieve the highest efficiency and lowest cost of the system. Reference [13] combines HABC with bee life cycle due to dynamic and static problems. In reference [14], in order to realize wind power Patterns clustering, wind power patterns are set as the objective function of clustering and K-means is improved. In reference [15], the co-evolution framework was introduced into the ABC algorithm, and a global optimal and leading artificial bee colony algorithm was designed. The improved strategy was adopted to accelerate the convergence speed of the algorithm and overcome the dimension dependence problem respectively. The algorithm proposed in reference [15] combines the filter to realize the function of noise reduction and avoid the influence of bad data on the whole.

TABLE 1. Comparison of different algorithms.

Algorithm	Convergence speed	Advantage	Disadvantage
Original ABC	Take some time	Excellent global search capability	Poor local search capability
Hybrid ABC [17]	Faster than origin	Strong local search ability and increased population diversity	Efficiency and accuracy are not good enough
Global-Best - ABC [18]	Faster than origin	Good global optimization capability	Possibility of falling into local optimal solutions
MEABC [19]	Faster than origin	High efficiency	No complementarity
CABC [20]	Slow	Good search ability	Maybe reduce convergence speed
Improved ABC	Faster than origin	Excellent global search capability	Excellent local search capability

In view of the respective characteristics of KMC and ABC algorithms, this paper first proposes an Improved ABC (IABC) algorithm, which uses the proposed maximum and minimum distance product method to initialize the bee colony to ensure that the selection of initial points can represent the distribution characteristics of the data set as much as possible. In the iterative process, the new fitness function and position update formula are used to optimize the evolution. Then IABC algorithm is applied to KMC and IABC-K-means algorithm is proposed to improve the clustering performance. It is applied to wind power generation to provide a theoretical basis for practical production.

II. PROBLEM FORMULATION AND PRELIMINARIES

A. K-MEANS CLUSTERING ALGORITHM

The k-means clustering algorithm divides the data into a predetermined class number k on the basis of minimizing the error [21], adopts distance as similarity assessment, and uses the center E_j of cluster $E_j (j = 1, 2, \ldots, k)$ to represent the cluster. $\mathcal{D}\left(x_i, x_j\right)$ is used to represent the Euclidean distance between two data objects x_i and x_j, and its calculation formula is as follows:

$$\mathcal{D}\left(x_i, x_j\right) = \sqrt{(x_{i1} - x_{i2})^2 + \cdots (x_{iL} - x_{iL})} \tag{1}$$

where L is the number of data object attributes.

Error square and SSE are used as objective functions to measure the clustering quality and represent the tightness of the samples around the center of the cluster. The smaller SSE is, the higher the sample similarity is. The calculation formula of SSE is as follows:

$$SSE = \sum_{j=1}^{k} \sum_{x \in E_j} \mathcal{D}\left(x, e_j\right) \tag{2}$$

$$e_j = \frac{1}{n_j} \sum_{x \in E_j} x \tag{3}$$

where, n_j is the number of sample data in the j^{th} cluster E_j.

B. PARTICLE SWARM OPTIMIZATION ALGORITHM

Particle swarm optimization (PSO) algorithm is a typical swarm intelligence optimization algorithm. Just like the majority of intelligence optimization algorithm, PSO usually generate a set of particles as an initial solution [22]. Then, the particles are updated iteratively, to get the better fitness for entire population. Finally, the optimal solution is expected to be found within a limited time of iterative steps.

A search space dimension N problem is solved by using the PSO algorithm containing M particles. Then, in the n^{th} iteration of PSO, the current position and current velocity of the $i^{\text{th}} (1 \leq i \leq M)$ particle can be expressed as $X_{i,n}\left(X_{i,n} = X_{i,n}^1, X_{i,n}^2, \ldots, X_{i,n}^N\right)$ and $V_{i,n}\left(V_{i,n} = V_{i,n}^1, V_{i,n}^2, \ldots, V_{i,n}^N\right)$ respectively. PSO updates the velocity and position of each particle on each dimensional component with the following formula at each iteration:

$$V_{i,n+1}^j = V_{i,n}^j + c_1 r_{i,n}^j \left(P_{i,n}^j - X_{i,n}^j\right) + c_2 R_{i,n}^j \left(G_{i,n}^j - X_{i,n}^j\right) \tag{4}$$

$$X_{i,n+1}^j = X_{i,n}^j + V_{i,n+1}^j \tag{5}$$

The above two equations are the updated iterative formula of PSO algorithm. Where $i = 1, 2, \ldots, M, j = 1, 2, \ldots, N$.

c_1, c_2 is the acceleration factor, which is used to adjust the convergence speed of the algorithm, usually with the value of 2. Where $P_{i,n}$ is the location of the best fitness found by the particle from initialization to the first iteration, also known as the individual optimal location, $P_{i,n} = P^1_{i,n}, P^2_{i,n}, \ldots, P^N_{i,n}$. After each iteration, the p_{best} position of each particle should be updated according to the following rules:

$$P_{i,n+1} = \begin{cases} P_{i,n} & f\left(P_{i,n}\right) < f\left(X_{i,n+1}\right) \\ X_{i,n+1} & f\left(P_{i,n}\right) \geq f\left(X_{i,n+1}\right) \end{cases} \tag{6}$$

where $f(\bullet)$ is the objective function for finding the corresponding position adaptation. Vector G_n is the position with the best fitness among all the p_{best} positions in the first iteration, in other words, the best position found by all particles up to the n^{th} iteration, also known as the global optimal position. $r^j_{i,n}$ and $R^j_{i,n}$ are usually random numbers evenly distributed between 0 and 1, that is, $r^j_{i,n}, R^j_{i,n} \in U(0, 1)$. For most problems, the j^{th} dimension of particle velocity should be limited within a certain interval $\left[-V^j_{\max}, V^j_{\max}\right]$ according to the actual situation when solving with PSO. However, because original PSO's search behavior tends to be global to solve some problems, it leads to slow convergence of the algorithm. To solve this problem, the PSO algorithm with inertia weight is proposed, that is, the inertia weight w is added into the above equation:

$$V^j_{i,n+1} = wV^j_{i,n} + c_1 r^j_{i,n}\left(P^j_{i,n} - X^j_{i,n}\right) + c_2 R^j_{i,n}\left(G^j_{i,n} - X^j_{i,n}\right) \tag{7}$$

A commonly used value is to decrease linearly with the increase of the number of iterations from 0.9 to 0.4. This version of PSO algorithm has been verified to have better performance than other versions in many experiments and applications, so it is called the standard PSO algorithm.

In the process of studying the motion trajectory of the standard PSO algorithm, the condition that the whole population can converge stably is that every particle in the population tends to point $p_{i,n}$, where $p_{i,n}$ can be expressed by the following formula:

$$p^j_{i,n} = \frac{c_1 r^j_{i,n} P^j_{i,n} + c_2 R^j_{i,n} G^j_n}{c_1 r^j_{i,n} + c_2 R^j_{i,n}} \tag{8}$$

The above equation can also be written as:

$$p^j_{i,n} = \eta^j_{i,n} P^j_{i,n} + \left(1 - \eta^j_{i,n}\right) G^j_n \tag{9}$$

where $\eta^j_{i,n}$ is a random number that satisfies (0,1) uniform distribution.

C. ARTIFICIAL BEE COLONY ALGORITHM

Artificial bee colony algorithm is a swarm intelligence algorithm that imitates the foraging behavior of bees. The swarm in the algorithm can be divided into three parts: leader, follower and scouter. Leader correspond to a specific food source and carry specific information about the food source; The follower bees waited for the leader bees to share information about the food source in the dance area of the hive, and then selected a food source to further explore around it. Scouters are responsible for randomly searching for new food sources.

The basic artificial bee colony algorithm can be divided into the following four stages.

1) INITIALIZATION PHASE

N food sources are randomly generated in the feasible solution space, and each food source represents a feasible solution. The specific formula is as follows:

$$x_{i,j} = x^{\min}_j + random\,(0, 1) \times \left(x^{\max}_j - x^{\min}_j\right) \tag{10}$$

where $i = 1, 2, \ldots, N; j = 1, 2, \ldots, D$, D is the dimension of the feasible solution. x^{max}_j and x^{max}_i represent the upper and lower limits of the j^{th} parameter. Also, set a counter for each food source and set its value to 0.

2) LEADER SEARCH PHASE

The leader conducted neighborhood search near the corresponding food source to find a new food source v_i, and the search formula was as follows:

$$v_{i,j} = x_{i,j} + (-1 + 2 \times random) \times \left(x_{i,j} - x_{k,j}\right) \tag{11}$$

where k represents randomly selected food resource different from i, $k \neq i$. For the new and old food sources x_i and v_i, the "greedy selection" algorithm is adopted, that is, the quality of the new food source and the old food source is compared. If the quality of the new food source is better, the new food source is retained and its counter is set to 0. Otherwise, keep the old food source and add one to its counter.

3) FOLLOWER SEARCH PHASE

Follower bees play a roulette wheel to select a food source from the current food source. The probability of each food source being selected is as follows:

$$P_i = \frac{fit_i}{\sum_{j=1}^{N} fit_j} \tag{12}$$

where fit_i represents quality of the food source and is calculated by the following formula:

$$fit_i = \begin{cases} \frac{1}{1+f_i} & f_i \geq 0 \\ 1 + |f_i| & otherwise \end{cases} \tag{13}$$

where f_i represents the value of objective function.

After the follower selected the food source, they searched the field according to phase 2, and then carried out greedy selection related operations

4) SCOUTER SEARCH PHASE

To avoid the loss of population diversity in the process of evolution, a special scout bee search mode was added to the bee colony algorithm. When the value in the counter corresponding to a food source is greater than a pre-set threshold limit, the current food source can be considered as exhausted, the food source is abandoned, the corresponding lead bee becomes the scout bee, and a new food source is randomly generated in the feasible solution space using phase 1 method.

D. SELECTION OF TYPICAL WIND POWER OUTPUT SCENARIOS

Because there is great randomness in renewable energy scheduling, whether from the power generation side or the user side, it is necessary to predict the wind power output, conduct day-ahead scheduling, and then conduct real-time scheduling according to the fluctuation of the day.

The number of original wind power scenarios is too large to be representative. To obtain typical wind power scenarios, n original scenarios in the merger cycle need to be reduced. The improved K-means clustering algorithm was used to process the data of different scenarios, and the original n scenarios were reduced and merged into a few k typical wind power scenarios, which could be represented by $k \times t$ matrix. In the process of scene reduction and merging, the number of scenes is reduced, and the data of t moments in the scene keeps the original time sequence. The typical scenario generation process of wind power.

The traditional k-means clustering method has the problems of poor global search ability and low clustering accuracy. If the initial clustering center is randomly selected, it may fall into the local optimal solution or even no solution. Thus, it is difficult to obtain the optimal typical wind power output scenario.

III. IMPROVED ARTIFICIAL BEE COLONY ALGORITHM

The basic artificial bee colony algorithm has the following two shortcomings: 1) the randomness of initialization results in low efficiency; 2) one-dimensional domain search leads to slow speed of convergence. In this paper, the maximum and minimum distance product method is used to initialize the sealed group to overcome its randomness. The new fitness function and the position change formula of the global guide factor are used for iterative optimization.

A. INITIALIZATION BASED ON MAXIMUM AND MINIMUM DISTANCE PRODUCT

Population initialization is very important in evolutionary algorithms because it affects the global convergence rate and the quality of solution.

The maximum and minimum distance method is adopted to search for the optimal initial cluster center [24], which reduces the sensitivity to the initial cluster center and greatly improves the convergence speed and accuracy. However, due to its adherence to the idea of minimum distance, the selection of initial cluster centers may be too dense and cluster conflicts may occur. The maximum distance product method was proposed to search for the initial cluster centers, which made the selection of initial points more consistent with the characteristics of data distribution and effectively reduced the number of iterations. However, the maximum distance product method also has defects. For example, two distance products are equal while the point densities in their regions differ greatly. Some parameters need to be input by users themselves, and the selected initial points tend to be inclined to the periphery of the point set, which cannot accurately reflect the actual data distribution. Based on the literature mentioned above, the maximum and minimum distance product method is proposed and used to initialize the population. This method not only overcomes the randomness of colony initialization, but also reduces the sensitivity of k-means algorithm to the initial point.

Aiming at the shortcomings of existing maximum and minimum distance method and maximum distance product method, a maximum and minimum distance product method is proposed. Where, D is the set containing all data; N is the number of initial colonies; k is the initial number of points to be selected; Z is the set of k initial points to be added, which is empty before the algorithm starts. *Temp* is an array that stores the product of the elements from Z to D. The algorithm flow is shown in the figure below.

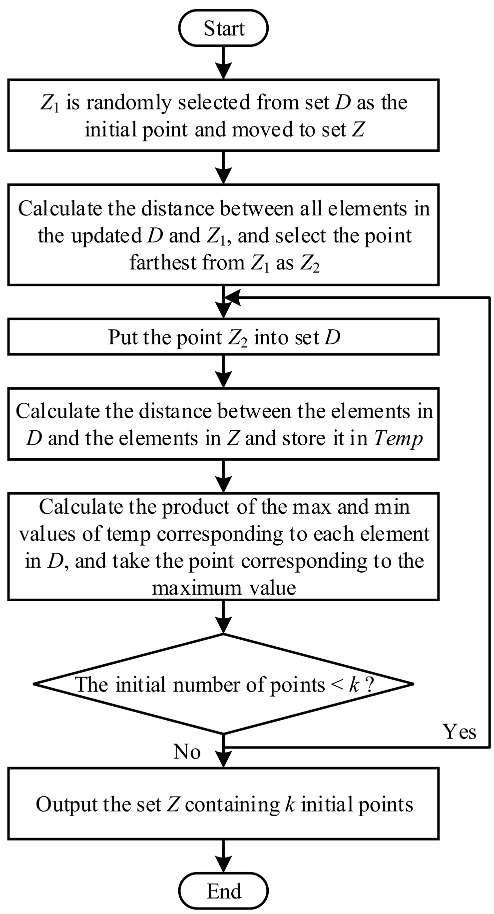


FIGURE 1. Max-min distance product algorithm flow chart.

It can be seen from the idea and steps of this method that the algorithm requires fewer parameters, and the product of ($Temp_{\max} \times Temp_{\min}$) can select points with higher point density, and the distribution of initial points is sparse. By this process, not only can we avoid the situation where two distance products are equal while the density of points in their regions varies greatly, but we can also use the products to amplify the differences between points, making the selection process more discriminative.

B. FITNESS FUNCTION

Fitness function will guide the direction of population evolution and directly determine the evolutionary behavior of the population, the number of iterations and the quality of the solution. Different fitness function will get different solutions. Therefore, a new fitness function is proposed based on the iterative search process of artificial bee colony and the idea of k-means algorithm, as shown in the following formula:

$$fitness_i = \frac{CM_i}{Dist_i} \quad i = 1, 2, \ldots, N \tag{14}$$

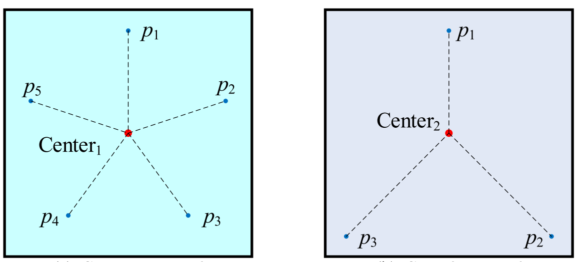
where CM_i is the number if points belonging to class i; $Dist_i$ is the sum of distances between all objects in class i and center

$$C_i, \quad Dist_i = \sum_{x_j \in C_i} d\left(x_j, C_k\right), Dist = \sum_{j=1}^{k} \sum_{x_i \in C_j} d\left(x_i, C_j\right).$$

If the fitness function is only used as points or in-class distance, there will be shortcomings as follows:

1) SAME DIST$_i$, DIFFERENT CM$_i$

If only the inner distance of class is used as fitness function, the accuracy will be lost in the process of selection. According to the above formula, there is obviously $fitness_a > fitness_b$, so iteration will evolve towards the trend of figure (a), reducing the number of iterations while improving the accuracy.



(a) CM_a=5, $Dist_a$=6 (b) CM_b=3 $Dist_b$=6

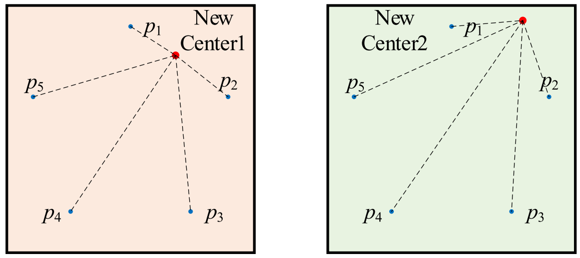
FIGURE 2. Same ***CM***, different ***Dist***.

2) SAME CM$_i$, DIFFERENT DIST$_i$

When only points are used as fitness function, the adaptability will decrease under the following circumstances. $fitness_a > fitness_b$ can be obtained from the above formula to make the iterative process more accurate.

C. POSITION UPDATING FORMULA

Position updating formula determines whether bees can find new nectar sources quickly and accurately. The original position updating formula has strong search ability, but its



(a) CM_a=5, $Dist_a^{NP}$=4.22 (b) CM_b=5 $Dist_b^{NP}$=5.38

FIGURE 3. Same ***Dist***, different ***CM***.

exploration ability is insufficient. It is easy to fall into local optimal solution and its update speed is slow in the process of domain search. To solve this problem, this paper proposes a position updating formula which introduces global factors:

$$V_{ij} = x_{ij} + r_{ij}\left(x_{mj} - x_{kj}\right) + \mu\left(x_{best,j} - x_{ij}\right) \tag{15}$$

where v_{ij} is a new position generated near x_{ij}; k, m, and j are random numbers generated by random formulas, $k, m \in \{1, 2, \ldots, N\}$, k and m are mutually exclusive and neither is equal to i; $r_{ij} \in [-1, 1]$; $\mu \in [0, 1]$ is a random number; $x_{best,j}$ is the most abundant source of honey.

The original method only iterated toward the vector direction of $random\,(0, 1) \times \left(x_i^{max} - x_i^{min}\right)$ in the field search, without considering the comparison of position advantages and disadvantages before and after the iteration. During the whole search process, each leader could only obtain the information of historical optimum and current position, lacking the consideration of global optimum for the whole colony. From the evolutionary perspective of swarm intelligence, individual in a group can benefit from the experiences of all other individuals in the group. Therefore, on the basis of the original formula, the global guiding factor ($x_{best,i}$ - x_{ij}) is added to make the bee search have a strong direction and purpose, and the influence factor μ is added in front of the global factor to constrain the amplitude of the search. As can be seen from the factor composition, if the gap between the current position and the optimal position is large, the updated step size will increase dynamically. Otherwise, it slowly approaches.

IV. KMC ALGORITHM BASED ON IMPROVED ARTIFICIAL BEE COLONY ALGORITHM

Based on the previous chapter, IABC-Kmeans algorithm is proposed. The basic idea of this algorithm is: The IABC algorithm was used for an iteration, and the new location obtained by the iteration was used as the initial point of KMC for clustering, and then the cluster center was used to update the colony. In this way, the IABC algorithm and KMC algorithm were alternately executed until the conditions were met to end the iteration.

The basic steps are as follows:

Step1. Set the number of leaders, followers and scouters (usually number of leaders = number of followers); Maximum iteration CM_{max} and control parameter T_{limit}; Current iteration $C_{current}$, initial value is 1; Number of clustering categories k; The bees were initialized with the product of maximum and minimum distance to generate $\{Z_1, Z_2, \ldots, Z_N\}$ initial colonies.

Step2. The initial colony was clustered and divided. The fitness of each bee was calculated and ranked, with the former as the latter and the last half as the follower.

Step3. Lead bees to search for a new position. According to the greedy selection principle, if the fitness of the new position is larger than the original position, the position will be updated to the new. Otherwise, it stays the same. When all the leaders have completed the search, the probability P_i is calculated.

Step4. Followers select the leaders based on the roulette principle according to the probability P_i obtained. In principle, the higher the P_i value is, the larger the fitness of the leader bee i is, and the higher the probability of being selected by the follower. After followers completing the selection of the leaders, the field search was conducted, and the position with large fitness was also selected according to the greedy principle.

Step5. After all the followers completed the search, the obtained position was used as the cluster center, and the data set was clustered by k-means iteration. According to the cluster division, the colonies were updated with the new cluster center of each class.

Step6. If the result of a leader does not change after iteration T_{limit}, the leader will be transformed into a scouter and a new position will be randomly generated to update the original position.

Step7. If the current iteration number is greater than the maximum iteration number CM_{max}, the iteration ends and the algorithm ends; otherwise, go to Step 2, $C_{current} = C_{current} + 1$

V. CASE STUDY

This chapter verifies the performance of the proposed algorithm, which mainly includes two parts: 1) the performance of IABC algorithm to find the optimal solution; 2) clustering performance of k-means algorithm based on IABC algorithm.

A. OPTIMIZATION PERFORMANCE ANALYSIS OF IABC ALGORITHM

The IABC algorithm proposed in this paper is compared with the basic ABC algorithm and PSO algorithm, and a series of benchmark functions are used to test the performance of the algorithm. The standard test data are shown in the formula below, which are Rastrigin, Rosenbrock, Griewank and Ackley, respectively.

$$f_1(x) = \sum_{i=1}^{D} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right] \tag{16}$$

$$f_2(x) = \sum_{i=1}^{D-1} \left[100\left(x_i^2 - x_{i+1}\right)^2 - (x_i - 1)^2 \right] \tag{17}$$

$$f_3(x) = \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{N} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \tag{18}$$

$$f_4 = -c_1 \cdot \exp\left[-0.2\sqrt{\frac{1}{n}\sum_{j=1}^{n} x_j^2} - \exp\left(\frac{1}{n}\sum_{j=1}^{n}\cos(2\pi x_j)\right) \right] + c_1 + e \tag{19}$$

When optimizing the function, the colony size was set to 20, that is, the number of leaders and followers was set to 10. The CM_{max} value was 50, i.e., the individuals with more than 50 iterations in the same food source changed from leaders to scouters. The maximum number of iterations is 1000. The following figure shows the fitness variation trend of the three algorithms under different test functions

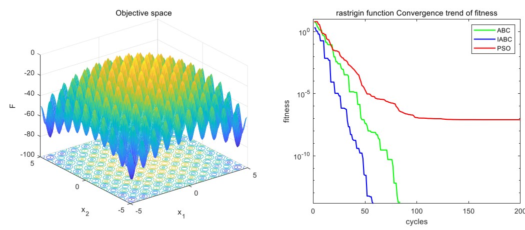


FIGURE 4. The variation trend of fitness value of different algorithms in Rastrigin function.

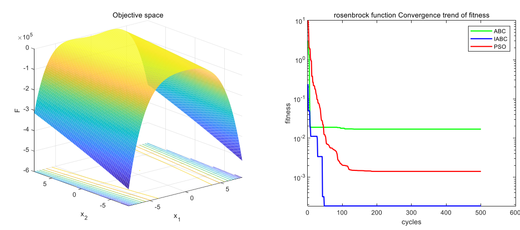


FIGURE 5. The variation trend of fitness value of different algorithms in Rosenbrock function.

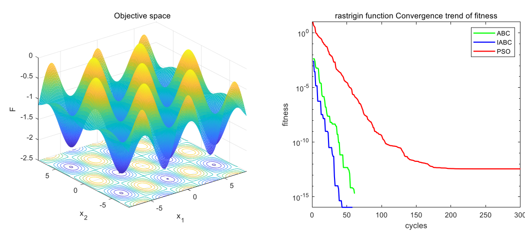


FIGURE 6. The variation trend of fitness value of different algorithms in Griewank function.

It can be seen from the figure above that the original ABC algorithm has different degrees of slow convergence speed and easy to fall into the local optimal solution on the multi-peak function. Compared with the original algorithm, PSO algorithm has higher convergence speed and fewer iterations, but it is weak in global optimization ability. The IABC algorithm in this paper adopts a new fitness function and position

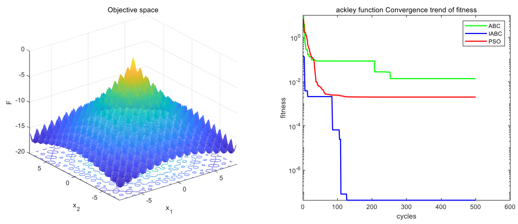


FIGURE 7. **The variation trend of fitness value of different algorithms in Ackley function.**

update formula to avoid the randomness of the neighborhood of food source location update. The algorithm enables bees to quickly move to the region where the optimal food source is located through the global guiding factor. Therefore, IABC algorithm has a great improvement in both iteration speed and global optimization ability.

B. CLUSTERING PERFORMANCE ANALYSIS OF IABC ALGORITHM

To test the effectiveness of the proposed algorithm, manual and real data sets are used to verify the algorithm to find the effect of the optimal cluster number.

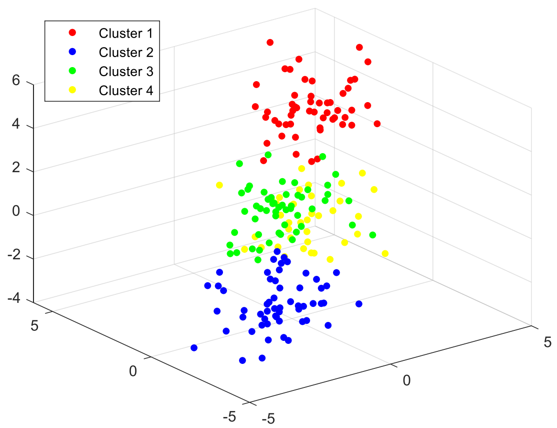


FIGURE 8. **Artificial datasets subject to Gaussian distribution S1.**

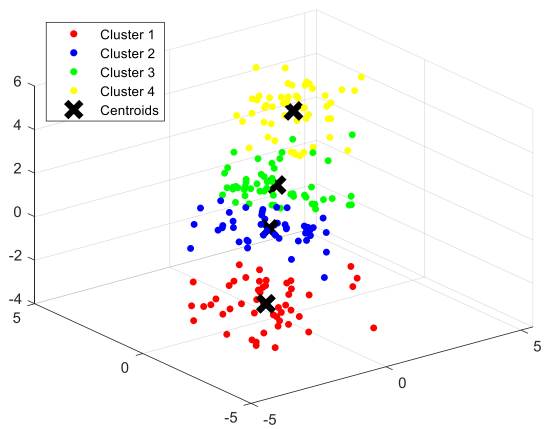


FIGURE 9. **Clustering result of dataset S1.**

TABLE 2. **Test dataset feature description.**

Dataset	Size	Category	Dimension
S1	200	4	3
S2	1800	6	3
Iris	150	3	4
Balance scale	625	4	3
Glass	214	6	9

To test the clustering accuracy of the proposed algorithm, the correct clustering number of each data set is given in advance. When testing the performance of the proposed algorithm to obtain the optimal cluster number, the algorithm was repeatedly run for 10 times, and The Times of the correct optimal cluster number obtained by the algorithm were recorded.

TABLE 3. **Algorithm running results.**

Dataset	Clustering accuracy	k-means correct times
S1	100	10
S2	100	10
Iris	90	9
Balance scale	40	4
Glass	60	6

In this paper, IABC algorithm is combined with KMC, and the initialization process, fitness formula and global guide factor are added to enhance the global search ability of the algorithm, which can jump out of the local optimal solution, with fewer iterations and better convergence accuracy.

C. TYPICAL WIND POWER SCENARIOS SELECTION

In this paper, four typical wind power scenarios are used to generate 90 artificial wind power output datasets that obey Gaussian distribution respectively, and cluster analysis is carried out on the datasets. The clustering results are shown in the figure below. The ordinate in the figure is the unit value, and its base value is 20MW.

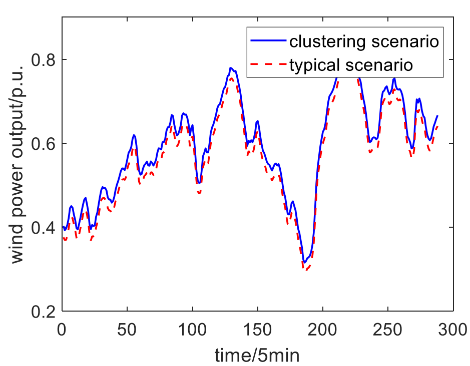


FIGURE 10. **Typical wind power output scenario 1.**

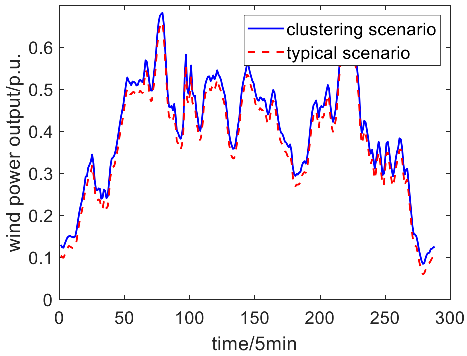


FIGURE 11. Typical wind power output scenario 2.

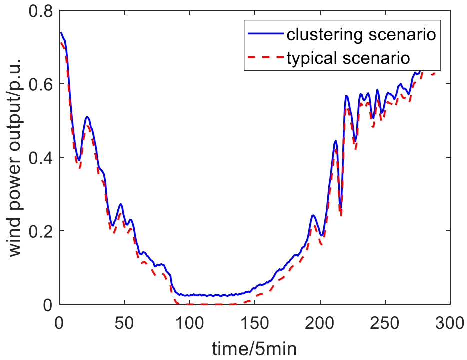


FIGURE 12. Typical wind power output scenario 3.

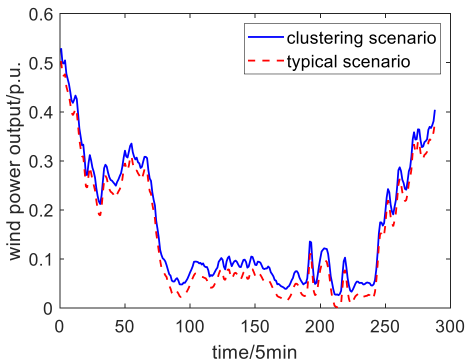


FIGURE 13. Typical wind power output scenario 4.

As can be seen from the figure above, the green dotted line is four typical wind power output scenarios, which are a 288-dimensional column vector. The 360 wind power output scenarios generated from typical output scenarios can achieve high accuracy through clustering.

According to the wind power output curve of a place 365 days, several typical wind power scenarios are aggregated. According to the typical output scenario, effective wind power planning can be carried out locally to achieve the highest economic benefits.

After clustering 365 days' wind power output scenarios, the output of local thermal power units can be adjusted as day-ahead dispatching data, which effectively improves the consumption of new energy and reduces pollution.

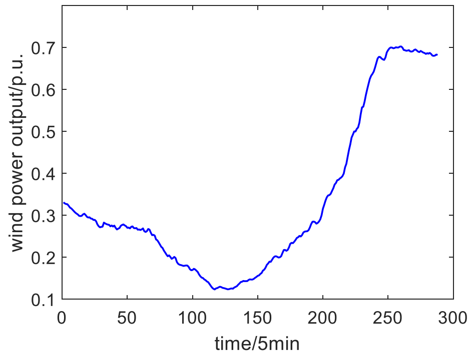


FIGURE 14. Typical wind power output scenario 1.

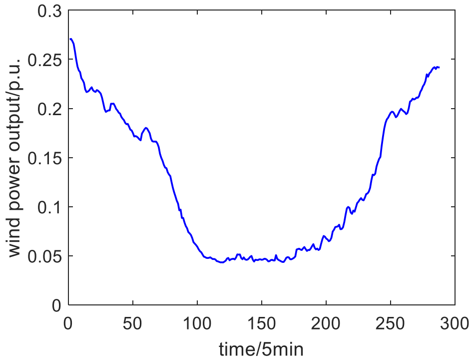


FIGURE 15. Typical wind power output scenario 2.

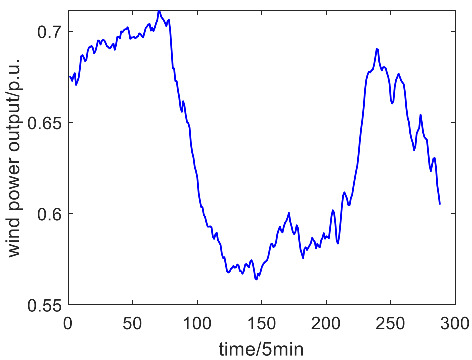


FIGURE 16. Typical wind power output scenario 3.

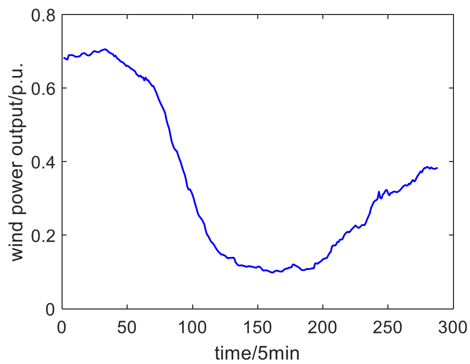


FIGURE 17. Typical wind power output scenario 4.

VI. CONCLUSION

In this paper, an improved artificial bee colony algorithm is proposed, which is improved from three aspects of colony initialization, fitness function and position update formula respectively. Also, it overcomes the randomness of initial algorithm and easy to fall into local optimal solution. The improved artificial bee colony algorithm is combined with KMC algorithm to solve the problem of poor global search ability of KMC algorithm. Experimental results show the effectiveness of the proposed algorithm, and the optimization efficiency and performance are greatly improved. And the method proposed in this paper is used to select typical wind power output scenarios, which can provide certain use value in the process of new energy consumption and power system operation.

REFERENCES

[1] C. Wu, X.-P. Zhang, and M. Sterling, "Wind power generation variations and aggregations," *CSEE J. Power Energy Syst.*, vol. 8, no. 1, pp. 17–38, Jan. 2022.

[2] L. Ye, C. Zhang, Y. Tang, W. Zhong, Y. Zhao, P. Lu, B. Zhai, H. Lan, and Z. Li, "Hierarchical model predictive control strategy based on dynamic active power dispatch for wind power cluster integration," *IEEE Trans. Power Syst.*, vol. 34, no. 6, pp. 4617–4629, Nov. 2019.

[3] J. Ouyang, M. Li, Z. Zhang, and T. Tang, "Multi-timescale active and reactive power-coordinated control of large-scale wind integrated power system for severe wind speed fluctuation," *IEEE Access*, vol. 7, pp. 51201–51210, 2019.

[4] L. Xue, T. Niu, Z. Li, and F. Li, "Fast VAR margin evaluation of high wind penetration power systems: An objective-oriented dimensionality reduction approximation approach," *IEEE Trans. Sustain. Energy*, vol. 13, no. 3, pp. 1722–1733, Jul. 2022.

[5] W. Teng, X. Wang, Y. Meng, and W. Shi, "An improved support vector clustering approach to dynamic aggregation of large wind farms," *CSEE J. Power Energy Syst.*, vol. 5, no. 2, pp. 215–223, Jun. 2019.

[6] M. Yesilbudak, "Clustering analysis of multidimensional wind speed data using K-means approach," in *Proc. IEEE Int. Conf. Renew. Energy Res. Appl. (ICRERA)*, Nov. 2016, pp. 961–965.

[7] H. Aoyang, Z. Shengqi, J. Xuehui, and Z. Zhisheng, "Short-term load forecasting model based on RBF neural network optimized by artificial bee colony algorithm," in *Proc. IEEE 2nd Int. Conf. Big Data, Artif. Intell. Internet Things Eng. (ICBAIE)*, Mar. 2021, pp. 486–489.

[8] S. Fairee, C. Khompatraporn, B. Sirinaovakul, and S. Prom-On, "Trim loss optimization in paper production using reinforcement artificial bee colony," *IEEE Access*, vol. 8, pp. 130647–130660, 2020.

[9] W.-F. Gao, L.-L. Huang, S.-Y. Liu, and C. Dai, "Artificial bee colony algorithm based on information learning," *IEEE Trans. Cybern.*, vol. 45, no. 12, pp. 2827–2839, Dec. 2015.

[10] X. Pan, Q. Zhang, and H. Pan, "Improved artificial bee colony algorithm and its application to fundus retinal blood vessel image binarization," *IEEE Access*, vol. 8, pp. 123726–123734, 2020.

[11] X. Wang, H. Yu, Y. Lin, Z. Zhang, and X. Gong, "Dynamic equivalent modeling for wind farms with DFIGs using the artificial bee colony with K-means algorithm," *IEEE Access*, vol. 8, pp. 173723–173731, 2020.

[12] R. Salem, M. A. Salam, H. Abdelkader, and A. A. Mohamed, "An artificial bee colony algorithm for data replication optimization in cloud environments," *IEEE Access*, vol. 8, pp. 51841–51852, 2020.

[13] H. Chen, L. Ma, M. He, X. Wang, X. Liang, L. Sun, and M. Huang, "Artificial bee colony optimizer based on bee life-cycle for stationary and dynamic optimization," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 2, pp. 327–346, Feb. 2017.

[14] A. A. Munshi, "Clustering of wind power patterns based on partitional and swarm algorithms," *IEEE Access*, vol. 8, pp. 111913–111930, 2020.

[15] F. Xu, H. Li, C.-M. Pun, H. Hu, Y. Li, Y. Song, and H. Gao, "A new global best guided artificial bee colony algorithm with application in robot path planning," *Appl. Soft Comput.*, vol. 88, Mar. 2020, Art. no. 106037.

[16] H. Li, F. Xu, C.-M. Pun, and L. Pan, "Hybrid feature selection algorithm based on discrete artificial bee colony for Parkinson diagnosis," *ACM Trans. Internet Technol.*, vol. 21, no. 3, pp. 1–22, 2020.

[17] H. Tang, H. Zhang, R. Liu, and Y. Du, "Integrating multi-index materials classification and inventory control in discrete manufacturing industry: Using a hybrid ABC-chaos algorithm," *IEEE Trans. Eng. Manag.*, vol. 69, no. 4, pp. 1276–1293, Aug. 2022.

[18] A. Toktas and D. Ustun, "A triple-objective optimization scheme using butterfly-integrated ABC algorithm for design of multilayer RAM," *IEEE Trans. Antennas Propag.*, vol. 68, no. 7, pp. 5602–5612, Jul. 2020.

[19] H. Wang, Z. Wu, S. Rahnamayan, H. Sun, Y. Liu, and J.-S. Pan, "Multi-strategy ensemble artificial bee colony algorithm," *Inf. Sci.*, vol. 279, pp. 587–603, Sep. 2014.

[20] O. Rehman, N. Javaid, B. Manzoor, A. Hafeez, A. Iqbal, and M. Ishfaq, "Energy consumption rate based stable election protocol (ECRSEP) for WSNs," *Proc. Comput. Sci.*, vol. 19, no. 2, pp. 932–937, 2013.

[21] O. Chapelle, B. Schölkopf, and A. Zien, "Probabilistic semi-supervised clustering with constraints," in *Semi-Supervised Learning*. Cambridge, MA, USA: MIT Press, 2006, pp. 73–102.

[22] S. M. Mikki and A. A. Kishk, *Particle Swarm Optimizaton: A Physics-Based Approach*. San Rafael, CA, USA: Morgan & Claypool, 2008.

[23] O. Holland, H. Bogucka, and A. Medeisis, "A machine-learning approach based on bio-inspired intelligence," in *Opportunistic Spectrum Sharing and White Space Access: The Practical Reality*. Hoboken, NJ, USA: Wiley, 2015, pp. 167–190.

[24] H. Borchani, C. Bielza, and P. Larrahaga, "Learning CB-decomposable multi-dimensional Bayesian network classifiers," ResearchGate, 2010, pp. 75–103.

GANG YAO is currently a Senior Engineer with the Power Dispatch Control Center, Guizhou Power Grid Company Ltd. His current research interest includes power system dispatching operation and control.

YANG WU is currently an Assistant Engineer with the Power Dispatch Control Center, Guizhou Power Grid Company Ltd. His current research interest includes power system dispatching operation and control.

XIAOXU HUANG is currently a Senior Engineer with the Power Dispatch Control Center, Guizhou Power Grid Company Ltd. His current research interest includes power system dispatching operation and control.

QINFENG MA is currently a Senior Engineer with the Power Dispatch Control Center, Guizhou Power Grid Company Ltd. His current research interest includes power system dispatching operation and control.

JIANG DU is currently an Engineer with the Power Dispatch Control Center, Guizhou Power Grid Company Ltd. His current research interest includes power system dispatching operation and control.