EAI Endorsed Transactions
on Energy Web
Research Article EAI.EU

Research on Wind Power Prediction Based on PSO-GSA-CS Synergistic Optimization of BP Neural Network

Haijun Wang*, Song Guo, Tao Jin, Xiaojiao Zhang

Department of Mathematics and Computer Engineering, Ordos Institute of Technology, Ordos, Inner Mongolia, 017000, China

Abstract

Accurate wind power prediction is essential for enhancing the dispatchability of wind power and the stability of power grid operation. However, traditional single-group intelligent algorithms for optimizing BP neural network modeling often suffer from large result fluctuations and poor stability. To address this, this paper proposes a wind power prediction model based on BP neural network optimized by a particle swarm–neighborhood gravitation–cuckoo collaborative optimization algorithm (PSGC). PSGC integrates the fast global convergence of particle swarm optimization (PSO), the Lévy flight characteristic of cuckoo search (CS), and the neighborhood gravitation mechanism of gravitational search algorithm (GSA) to maximize the advantages of each algorithm and overcome the limitations of single algorithms. This integration is applied to the initial optimization of the BP neural network's weights and thresholds, followed by secondary optimization through the BP network algorithm, ultimately for wind power prediction. Results show that the PSGC-BP model performs well in prediction accuracy and stability. In the testing phase, the model's root mean square error (*RMSE*) is 22.0127, mean absolute error (*MAE*) is 17.1045, and correlation coefficient (*R*) is 0.7903; in the prediction phase, *RMSE* is 45.2569, *MAE* is 27.9380, and *R* is 0.8408, with the smallest operating fluctuation, highest stability, and best comprehensive performance. This model provides a feasible method for wind power prediction, contributing to improved dispatchability of wind power and grid stability.

Keywords: Wind power, Power prediction, Particle Swarm Optimization, Cuckoo Search, Gravitational Search Algorithm, BP neural network

Received on 12 October 2025, accepted on 14 October 2025, published on 09 February 2026

doi: 10.4108/ew.11839

1. Introduction

With the advancement of the dual-carbon strategy, wind power has become one of the key forms of energy in the global energy transition and the process of achieving carbon neutrality goals. However, its inherent randomness, intermittency, and volatility pose severe challenges to real-time grid dispatch and power balance [1]. Research shows that by the end of 2024, the global installed capacity of wind power has exceeded 1136 GW. It is predicted that by 2030, this figure will rise to about 2 TW; by 2050, it is expected to break through 6 TW, and the proportion of wind power in the global total power generation will approach 28% [2]. Therefore, constructing a high-precision and robust wind power prediction model has become one of the most concerned issues for wind farms and dispatch departments. Traditional physical models rely on numerical weather prediction (NWP) and unit power curves, making it difficult to capture the nonlinear mapping between wind speed and power [3]. Statistical methods such as ARMA and its improved algorithms, although computationally efficient, are prone to feature loss when dealing with multi-source heterogeneous meteorological data [4].

Artificial neural networks are widely used due to their excellent nonlinear approximation capabilities. Among them, the BP neural network, with its simple structure and ease of implementation, has become a baseline model for wind power

*Corresponding author. Email:wanghaijun@oit.edu.cn

prediction [5]. However, the BP network initializes weights and thresholds randomly, making it prone to falling into local minima, resulting in large prediction fluctuations and poor stability [6].To overcome these defects, swarm intelligence optimization algorithms have been introduced to improve the quality of the network's initial parameters. While each individual algorithm has its own advantages, they also have issues such as premature convergence or high computational costs: Genetic algorithms have strong global search capabilities but converge slowly in the later stages [7,8].Particle swarm optimization algorithms converge quickly but are prone to falling into local optima [9].Cuckoo search algorithms, with their Lévy flight characteristics, can effectively escape local traps but are sensitive to parameter settings [10].In addition, novel algorithms such as beetle antennae search, sparrow search, and scarab search have also shown good potential in hyperparameter optimization [11-14].

However, most existing studies remain within the single algorithm +BP framework, neglecting the complementary advantages brought by the collaboration of multiple algorithms. Inspired by the idea of multi-agent collaboration, this paper proposes a tri-mechanism collaborative optimization algorithm of Particle Swarm-Neighborhood Gravity-Cuckoo Search,which deeply integrates the rapid global convergence ability of Particle Swarm Optimization (PSO), the Lévy flight characteristic of Cuckoo Search (CS), and the neighborhood gravity mechanism of Gravitational Search Algorithm (GSA). This integration forms an efficient improved PSO, GSA, and CS collaborative optimization algorithm (PSGC), which is applied to the initial optimization selection of weights and thresholds in BP neural networks. Subsequently, BP neural networks are utilized to conduct a secondary optimization of these weights and thresholds. Finally, the optimized BP neural network model is applied to wind power prediction research. The results indicate that compared with the optimization of the BP neural network model by individual algorithms such as PSO, GSA, and CS, the PSGC algorithm proposed in this paper achieves higher prediction accuracy and better stability in wind power prediction.

2. BP Model Analysis and Problem Discussion

The BP neural network is a type of multi-layer feedforward neural network, trained through the back propagation of errors. Its structure includes *m* input nodes, *z* hidden nodes, and *n* output nodes. The optimization process of the BP neural network essentially seeks the solution that minimizes the error for the problem at hand, as mathematically modeled in equation (1) below.

$$\begin{cases} \min E(w,q,\theta,r)=\dfrac{1}{N}\sum_{k=1}^{N}(y_k-t_k)^2 \\ \text{s.t.} w\in\Omega^{m\times z}, q\in\Omega^{z\times n}, \theta\in\Omega^{z}, \gamma\in\Omega^{n} \end{cases} \tag{1}$$

Where y_k is the actual output of the *k*-th sample, t_k is the expected output of the *k*-th sample, N is the number of samples, and Ω denotes the real number space. The final solution obtained from the mathematical model $\min E(w,q,\theta,\gamma)$ corresponds to the optimal parameter combination of w,q,θ,γ for the model.

The choice of initial weights *w*, *q*, and learning rates θ, γ is crucial for the training of a BP neural network. Inappropriate initial values may lead to convergence to local minima or slow down the convergence rate, affecting the model's performance and generalization ability. The BP algorithm uses gradient descent to solve for the weights and thresholds[15], and the update of these weights follows the rule shown in equation (2).

$$\begin{cases} w_{new}=w-\eta\dfrac{\partial E}{\partial w}, q_{new}=q-\eta\dfrac{\partial E}{\partial q} \\ \theta_{new}=\theta-\eta\dfrac{\partial E}{\partial \theta}, \gamma_{new}=\gamma-\eta\dfrac{\partial E}{\partial \gamma} \end{cases} \tag{2}$$

In this context, η represents the learning rate, and $\frac{\partial E}{\partial w},\frac{\partial E}{\partial q},\frac{\partial E}{\partial \theta},\frac{\partial E}{\partial \gamma}$ are the partial derivatives of the error function *E* with respect to the weights *w*, *q*, and the thresholds θ, γ, respectively.

These update rules illustrate that the gradient descent method has limited capability to adjust weights and thresholds during the training process, especially when the initial weights and thresholds are poorly chosen, which may lead the algorithm to become trapped in local minima or slow down the convergence rate. Therefore, selecting appropriate initial values is crucial to ensure the effectiveness of BP neural network training, especially given the limited capacity for adjustment of initial weights and thresholds throughout the execution of the gradient descent algorithm.

3. Introduction of Collaborative Optimization Strategy

To overcome the limitations of initial weight and threshold selection in BP neural networks and enhance their global search ability and convergence speed, this study proposes a collaborative optimization PSGC algorithm, which is a hybrid method of PSO, GSA, and CS. By integrating the advantages of multiple optimization algorithms, this method aims to provide better initial weights and thresholds for the BP neural network wind power prediction model, thereby improving its training performance and generalization ability.

3.1. Particle Swarm Optimization Theory

PSO is a swarm intelligence-based optimization algorithm

that simulates the foraging behavior of bird flocks. Each particle represents a potential solution in the solution space, updating its own position and velocity through individual experience and swarm experience. In the actual algorithm implementation process, the inertia weight w_p, individual learning factor c_1, and social learning factor c_2 are dynamically adjusted to balance global search and local search capabilities. The specific velocity update formula and position update formula of the algorithm are shown in Equations (3) and (4) respectively[16].

$$v_i(t+1) = w_p \cdot v_i(t) + c_1 \cdot r_1 \cdot (pBest_i - x_i(t)) + c_2 \cdot r_2 \cdot (gBest_i - x_i(t)) \tag{3}$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{4}$$

In which, v_i and x_i represent the velocity and position of the particle, respectively, $pBest$ is the optimal individual position of the particle, $gBest$ is the global optimal position, and r_1 and r_2 are random numbers.

3.2. Gravitational Search Algorithm Theory

The GSA simulates the gravitational interactions between celestial bodies, where each solution is considered a celestial body with a mass proportional to its fitness. The algorithm dynamically adjusts the positions of particles through gravitational interactions. In the GSA algorithm, the gravitational constant G gradually decreases as the number of iterations increases, simulating the attenuation of gravity and enhancing the algorithm's local search capability. The specific calculations involved in the mass computation, gravitational force computation, and acceleration computation during the operation are shown in Equations (5), (6), and (7) respectively[17].

$$m_i = \frac{fitness_i - worst}{best - worst} \tag{5}$$

$$F_{ij} = G \cdot \frac{m_i \cdot m_j}{d_{ij} + \varepsilon} \tag{6}$$

$$a_{ij} = \frac{\sum_{j=1}^{n} F_{ij}}{m_i + \varepsilon} \tag{7}$$

In which, m_i is the mass of the particle, d_{ij} is the distance between particles, G is the gravitational constant, and ε is a small constant to prevent division by zero.

3.3. Cuckoo Search Theory

The CS algorithm utilizes the characteristics of Lévy flights to enhance global search capabilities through long-distance random jumps. Lévy flights are a type of random walk with a long-tailed distribution, effectively avoiding local optima. The step size of Lévy flights is shown in Equation (8).

$$L = \frac{\mu}{|v|^{1/\beta}} \tag{8}$$

In which, μ and v are random variables following a normal distribution, β is the parameter of the Lévy distribution, typically ranging between (0,2), used to control the distribution characteristics of the step size, and L is the generated Lévy step size.

3.4. Collaborative Optimization Algorithm Process

The PSGC algorithm achieves a dynamic balance between global exploration and local exploitation by integrating three optimization mechanisms in each generation of iteration. The process of optimizing the BP neural network with the PSGC algorithm is divided into three stages, each containing several steps, as shown in Figure 1.

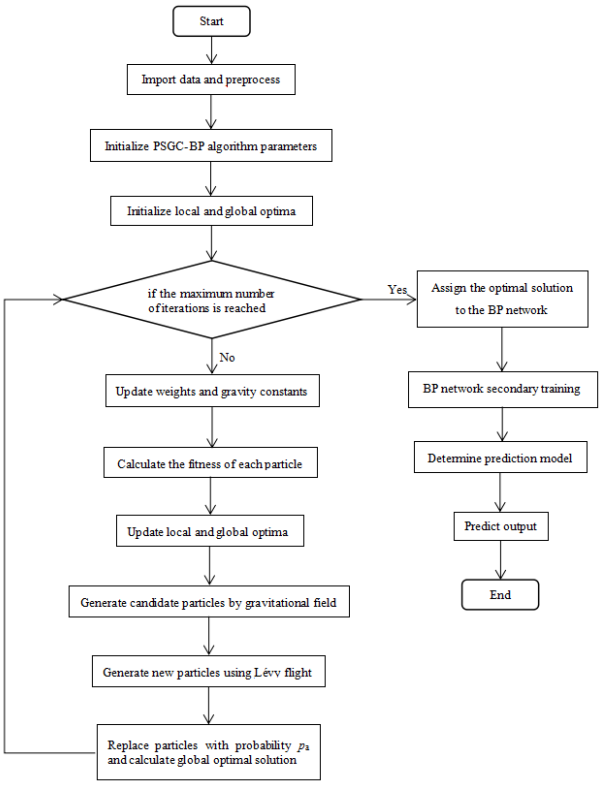

Figure 1. Operation Flow of the PSGC-BP Algorithm

The specific steps of the PSGC algorithm for optimizing the BP neural network are as follows.

Stage1.All parameters of the operation process are initialized, where the weights and thresholds are initialized using Equations (9) and (10) for the particle velocities. Based on this initialization, the individual best position $pBest$ and the global best position $gBest$ are calculated.

$$curr_{\text{pos}} = rand(num, \dim) \times (up - low) + low \tag{9}$$

$$v_{el} = 0.3 \times randn(num, \dim) \tag{10}$$

Stage2.Determine whether the maximum number of iterations has been reached. If so, proceed to **Stage3** for decoding and modeling to conduct secondary optimization. Otherwise, execute the main loop—repeatedly performing **Step1** to **Step6** until the maximum iteration count $iter_{\max}$ is achieved.

Step1.Update the inertia weight and learning factors using Equations (11) and (12).

$$w_p = w_{\max} - (w_{\max} - w_{\min}) \cdot \frac{iter}{iter_{\max}} \tag{11}$$

$$\begin{cases} c_1 = 1.5 - \dfrac{iter}{iter_{\max}} \\ c_2 = 1 + \dfrac{iter}{iter_{\max}} \end{cases} \tag{12}$$

Step2.Calculate the fitness value of each particle. In the optimization of BP neural networks, fitness is usually related to the prediction error of the network. Considering that a smaller fitness value in this study indicates a smaller prediction error and higher solution quality of the particle, solving for the minimum fitness value essentially involves finding the optimal solution of Equation (1). Therefore, the fitness function is represented as shown in Equation (13).

$$fitness = E(w, q, \theta, \gamma) = \frac{1}{N} \sum_{k=1}^{N} (y_k - t_k)^2 \tag{13}$$

Step3.Update the individual best position $pBest$ and the global best position $gBest$ for each particle based on Equation (13). For each particle i, if $curr\_fitness(i) < pBestScore(i)$, then $pBestScore(i) = curr\_fitness(i)$, and $pBest(i) = curr_{\text{pos}}(i)$. For each particle i, if $curr\_fitness(i) < gBestScore(i)$, then $gBestScore(i) = curr\_fitness(i)$, and $gBest(i) = curr_{\text{pos}}(i)$.

Step4.Update the velocity and position using Equations (3) and (4) to generate new v_{el} and $curr_{pos}$.

Step5. According to formulas (5), (6), and (7) in the GSA algorithm, calculate the new acceleration a_i to generate candidate particles, and update the velocity and position of the candidate particles as shown in equations (14) and (15).

$$v_{el}(i) = v_{el}(i) + a_i \tag{14}$$

$$curr_{pos}(i) = curr_{pos}(i) + v_{el}(i) \tag{15}$$

Step6. For the candidate particle positions generated in **Step5**, use the new positions new_{pos} generated by the Lévy flight to update the particle positions, as shown in equation (16). According to the random substitution mechanism of the cuckoo search algorithm, replace the positions of some particles with a certain probability p_a to introduce new solutions and enhance global search capability.

$$new_{pos} = curr_{pos} + \alpha \cdot r_3 \cdot L \tag{16}$$

In which, α is the step size factor, and r_3 is a random number from a standard normal distribution.

Stage3. After the completion of **Stage2**, the obtained weights w, q, and thresholds θ, γ are combined to assign initial values to the BP neural network. Then, the BP algorithm is used to perform a secondary optimization on this initial solution until the end to construct the corresponding wind power prediction model.

4. Experiment

4.1. Data Acquisition

In this study, we used the wind power dataset from the Alibaba Cloud Tianchi public dataset as the source of experimental data. This dataset contains a total of 3,648 records, each with 12 attributes, including wind speed and direction at four different heights (10m, 30m, 50m, and 70m), as well as temperature, air pressure, humidity, and wind power information. These rich multidimensional data provide comprehensive and detailed support for the study. Figure 2 shows the curve of wind power in the dataset as it changes with sample points, from which the volatility of the data can be observed, reflecting the instability of wind power and possible periodic changes. Given the BP neural network's capability in handling nonlinear relationships and capturing complex patterns in data, it is well-suited for predicting the trend of wind power data.

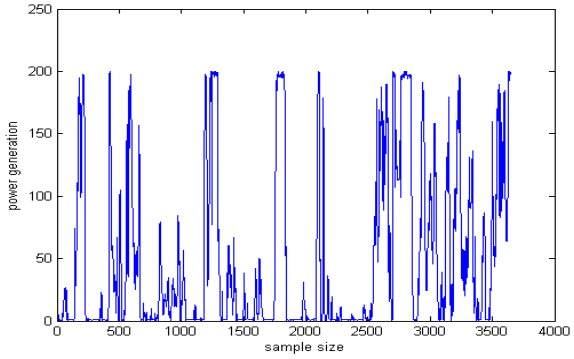

Figure 2. Original Wind Power Trend Curve

4.2. Data Analysis

To accurately select the correlation between data and eliminate redundant features, this study employs Pearson correlation coefficients for correlation analysis. The core principle of Pearson correlation analysis is to calculate the degree of correlation between two different variables to assess the strength of their linear relationship. The calculation of the Pearson correlation coefficient is shown in Equation (17).

$$R = \frac{\sum_{i=1}^{l}(u_i - \bar{u})(v_i - \bar{v})}{\sqrt{\sum_{i=1}^{l}(u_i - \bar{u})^2}\sqrt{\sum_{i=1}^{l}(v_i - \bar{v})^2}} \tag{17}$$

In the formula, u_i and v_i are the i-th observed values of variables u and v, respectively. $\bar{u}$ and v_i are the mean values of variables u and v, respectively. The coefficient R ranges between [-1, +1], where values closer to 0 indicate weaker correlation, and values closer to -1 or +1 indicate stronger correlation. Factors with high correlation are selected as model inputs. The relationship between the absolute value of the Pearson correlation coefficient and the degree of correlation is shown in Table 1[18].

Table 1. Relationship Between the Absolute Value of the Correlation Coefficient and the Degree of Correlation

Absolute Value Range of $\|R\|$	(0.8,1]	(0.6,0.8]	(0.4,0.6]	(0.2,0.4]	(0,0.2]
R	Very Strong	Strong	Moderate	Weak	Very Weak

Using Pearson's formula, we calculated the correlation between wind speed (at 10m, 30m, 50m, and 70m), wind direction (at 10m, 30m, 50m, and 70m), air pressure(AP), temperature(Temp), relative humidity(RH), and wind power. In this experiment, the 3,648 samples were divided into 3,000 training samples, 48 testing samples, and 600 prediction samples. The correlation coefficient R was calculated using the training samples, and the results are shown in Table 2. From the table, it can be observed that humidity, air pressure, and temperature have a very weak correlation with wind power and can be neglected. Therefore, the final selection of input variables for the BP neural network includes the 8 attributes of wind speed (at 10m, 30m, 50m, and 70m) and wind direction (at 10m, 30m, 50m, and 70m).

Table 2. Correlation Coefficients of Various Factors with Wind Power

Prop	Wind Speed				Wind Direction				Temp	AP	RH
	10m	30m	50m	70m	10m	30m	50m	70m			
R	0.8313	0.8720	0.8988	0.9123	-0.3776	-0.3908	-0.4055	-0.4164	-0.0004	0.0109	0.0749

4.3. Determination of BP Network Structure

Existing research indicates that determining the optimal number of nodes in the hidden layer is a core issue in the design of the network structure, given the a priori dimensions of the input and output layers. In this paper, we employ the golden section search strategy[19]to determine the number of nodes z in the hidden layer, with the feasible domain for the number of nodes in the hidden layer defined by Equation (18). Here, m and n correspond to the number of input and output neurons in Equation (1), respectively, with 8 and 1 being the counts for input and output neurons. a and b represent the left and right boundaries of the search interval during the iteration process. The Pearson correlation coefficient R between the test samples and the actual samples is used as the objective function. We sequentially evaluate the generalization performance at the 0.618 and 0.382 quantiles, iteratively narrowing the search interval until the final number of nodes in the hidden layer is determined. Table 3 shows the correlation values for each number of nodes in the feasible domain. Therefore, the final number of nodes selected for the hidden layer is 15, and the structure of the BP model is as shown in Figure 3.

$$\begin{matrix} a \\ b \end{matrix} = \frac{n+m}{2} \le z \le (n+m)+10 = \tag{18}$$

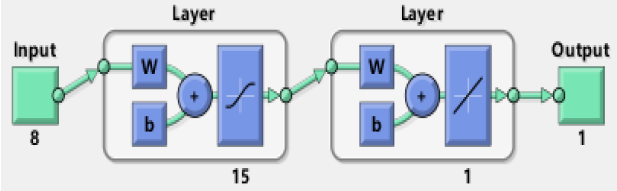

Figure 3. Structure of the BP Model

Table 3 Calculation Results of Correlation Coefficients for Each Hidden Layer Node in the Feasible Domain

z	13	14	15	16	17
R	0.5525	0.5939	0.6614	0.6129	0.4931

4.4. Experiment and Result Analysis

Wind power prediction models were established and tested using GSA-BP, PSO-BP, CS-BP, and PSGC-BP algorithms. Considering that both BP algorithms and PSO, GSA, and CS optimization algorithms involve a large number of random parameters, the results of the algorithmic models are highly stochastic. Therefore, in this experiment, the comparison was made based on the output results of each model run 10 times to reduce the impact of random fluctuations in the output results on the analysis. Figure 4 and Figure 5 show the comparison of output results for the testing and prediction segments, respectively. It can be intuitively seen from the figures that, in the testing segment, except for the GSA-BP model which has some output points that clearly do not follow the actual value trend, the outputs of the other models are in line with the actual value trend. In the prediction segment, the PSGC-BP model performs the best, with its prediction results being closer to the original values at most sample points, demonstrating higher prediction accuracy.

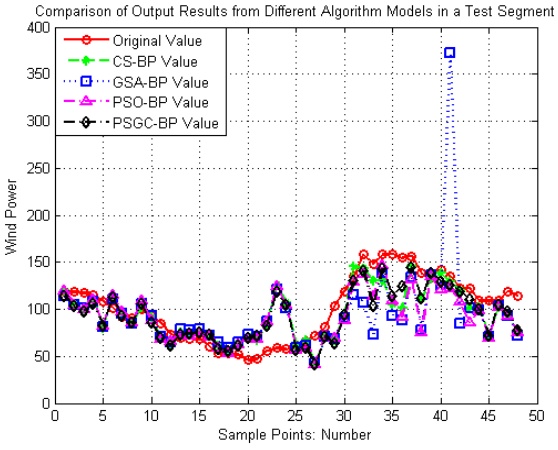

Figure 4. Comparison of Output Results from Various Algorithmic Models in the Testing Segment

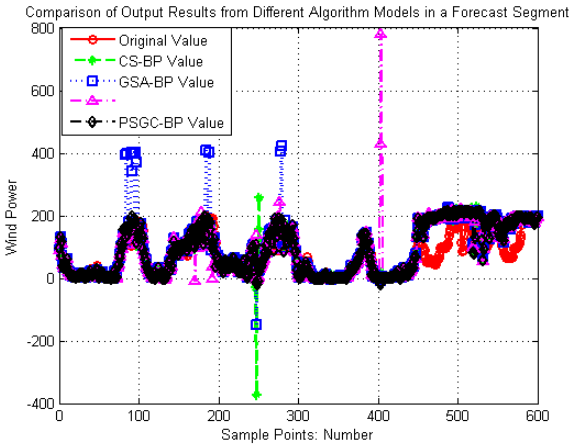

Figure 5. Comparison of Output Results from Various Algorithmic Models in the Prediction Segment

To further compare the algorithmic models, the Root Mean Square Error (*RMSE*), Mean Absolute Error (*MAE*), and Pearson correlation coefficient *R* were used to evaluate the model results. Table 4 presents the mean comparison of the results from 10 runs of different algorithmic models. From Table 4, it can be seen that the PSGC-BP algorithmic model demonstrates good performance in most key performance indicators. During the testing phase, the *RMSE* of the PSGC-BP algorithmic model's output results is 22.0127, the *MAE* is 17.1045, and the correlation coefficient *R* is 0.7903, all of which show its advantage in predictive accuracy. Although during the prediction phase, the *MAE* of the PSGC-BP algorithmic model is 27.9380, slightly higher than that of CS-BP's 27.9189, its *RMSE* is 45.2569, and the correlation coefficient *R* is 0.8408, still significantly outperforming other algorithms, demonstrating the excellent comprehensive predictive performance of PSGC-BP.

Table 4. Comparison of Final Prediction Results of Different Algorithm Models

Algorithm	Data Segment	*RMSE*	*MAE*	*R*
GSA-BP	Test Segment	45.1321	27.1321	0.4041
	Prediction Segment	57.1042	30.6134	0.7918
PSO-BP	Test Segment	26.0500	19.7881	0.7001
	Prediction Segment	58.6479	29.8317	0.7366
CS-BP	Test Segment	23.4126	17.3471	0.7491
	Prediction Segment	47.6685	27.9189	0.8193
PSGC-BP	Test Segment	22.0127	17.1045	0.7903
	Prediction Segment	45.2569	27.9380	0.8408

Further analysis of the fluctuation of the 10 output results in Table 5 reveals the following: during the testing phase, the PSGC-BP algorithm model exhibited fluctuations of 10.0549 in *RMSE*, 8.4257 in *MAE*, and 0.1847 in *R*. During the prediction phase, the fluctuations were 10.2563 in *RMSE*, 3.8517 in *MAE*, and 0.0494 in *R*. It can be observed that the PSGC-BP algorithm model has the smallest fluctuation values, indicating that its predictive results are more stable compared to other algorithms.

Table 5. Comparison of Output Fluctuations Over 10 Runs for Different Algorithm Models

Algorithm	Data Segment	*RMSE* Fluctuation Value	*MAE* Fluctuation Value	*R* Fluctuation Value
GSA-BP	Test Segment	297.5196	50.0854	0.6134
	Prediction Segment	274.4028	48.6420	0.5138
PSO-BP	Test Segment	32.7039	11.3395	0.5039
	Prediction Segment	381.3890	39.9263	0.7638
CS-BP	Test Segment	26.6069	11.7220	0.4938
	Prediction Segment	157.9730	15.3385	0.5411
PSGC-BP	Test Segment	10.0549	8.4257	0.1847
	Prediction Segment	10.2563	3.8517	0.0494

Table 6 presents the average running time of 10 optimization trials for various algorithms in the process of optimizing the weights and thresholds of the BP model. As shown in the table, the running time of the PSGC algorithm is only slightly longer than that of the PSO algorithm, but significantly shorter than that of the GSA and CS algorithms. Although the running time of the PSGC algorithm is slightly longer, it achieves a significant improvement in prediction accuracy and stability. Taking into account both running time and prediction performance, the PSGC algorithm still performs excellently among all algorithms and has a clear advantage. In addition, considering that the state of the computing equipment may affect the running time, we conducted multiple experiments for verification. The results show that the conclusions are basically consistent with the results of this experiment, further confirming the superiority of the PSGC algorithm.

Table 6. Comparison of Running Times for Different Algorithm Optimization Phases (Unit: s)

Algorithm	GSA	PSO	CS	PSGC
Running Time	83.6336	65.3351	87.8780	74.2273

5. Conclusion

This paper proposes a wind power prediction model optimized by a BP neural network based on the Particle Swarm-Neighborhood Gravity-Cuckoo collaborative optimization algorithm. The model integrates the particle swarm optimization, gravitational search algorithm, and cuckoo search mechanisms, fully leveraging the advantages of each algorithm and overcoming the limitations of individual algorithms. Experiments using the wind power dataset from Alibaba Cloud's Tianchi Lab demonstrate that the PSGC-BP model exhibits excellent prediction accuracy and stability in both the testing and prediction phases. This proves the necessity and effectiveness of the integrated optimization strategy. Although the running time of the PSGC algorithm is slightly longer than that of the PSO algorithm, its comprehensive performance remains the best, especially in application scenarios where high requirements for prediction accuracy and result stability are prioritized.

With the advancement of the dual carbon strategy, the proportion of wind power in the energy structure is continuously increasing, and the accuracy and stability of its power prediction are crucial for real-time scheduling and power balance in the power grid. The model proposed in this paper provides a new and effective method for wind power prediction, which helps to improve the dispatchability of wind power and the stability of power grid operations. Future work will further combine more meteorological data and optimization algorithms to enhance the model's performance and generalization capabilities.

Acknowledgements.

This paper is supported by Key Project of Natural Science of Ordos Institute of Technology (ZRZD2024005).Ordos High-Level Talent Research Startup Fund Project-Research on Prediction and Classification Based on Swarm Intelligence and Machine Learning Technology. Thanks are hereby extended to all.

References

[1] Luo Dan,Zhang Ruobing,Yu Juan,et al.Short-Term Wind Power Prediction Model Based on Improved BBO and Optimized BP Neural Network[J].Journal of Green Science and Technology, 2024(12): 263-269.

[2] LONG Xiao, ZHANG Jinbin, CHEN Lingte. Prospects for Future Energy Technologies[J]. Power Generation Technology, 2025,46(04):651-693.

[3] Xuewei Duan,Ruiqi Wang,Zhaoxin Wang,et al.Review of Wind Speed and Wind Power Prediction Methods[J]. Shandong Electric Power,2015，42(7)：26-32.

[4] Lei Dong,Lijie Wang,Ying Hao,et al.Prediction of Wind Power Generation Based on Autoregressive Moving Average Model[J]Acta Energiae Solaris Sinica,2011,32(5):617-622.
[5] XIANG Hang,WANG Xin-ju,LI Qiang,et al.Wind Generation Power Forecast and Model Optimization Method Based on Neuron Netowork[J]. Electric Switchgear, 2022, 60(03):66-70+92.
[6] Cao Jinbi,Huang Yanan,Liu Yaotong.Wind Power Prediction Model Based on Neural Network of PSO-BP[J].Modern Industrial Economy and Informationization, 2024(8): 164-168.
[7] DING Jun,GU Yuchuan,HUANG Xia,et al.Research on Prediction Accuracy of Flow Stress of 304 Stainless Steel Based on Artificial Neural Network Optimized by Improved Genetic Algorithm[J]. Journal of Mechanical Engineering, 2022, 58(10):78-86.
[8] Das P , Nayak P K , Kesavan R K .Artificial neural networks (ANN)-genetic algorithm (GA) optimization on thermosonicated achocha juice:kinetic and thermodynamic perspectives of retained phytocompounds[J].Preparative Biochemistry & Biotechnology, 2025, 55(2):171-186.
[9] Liu C , Liu H , Qiu Y .Fault diagnosis of new energy vehicles based on PSO-IBP neural network[J]. International Journal of Low Carbon Technologies, 2025, 20:1104 – 1111.
[10] Vijayakumar K , Vinothkumar C , Esakkiappan C .RETRACTED: Cuckoo Search Optimization based PI Controller Tuning for Hopper Tank System:[J].Concurrent Engineering, 2022, 30(3):300-308.
[11] Ma J , Zhang W , Guo D ,et al.A Path Planning Method Based on Improved Beetle Antennae Search Algorithm for Intelligent Agent[C].2024 5th International Conference on Artificial Intelligence and Electromechanical Automation (AIEA), 2024:116-121
[12] Singh M K , Verma Y K .Refraction reverse learning based hybrid Namib Antenna Beetle Optimization for resource allocation in NB-IoT platform[J].Multimedia Tools & Applications, 2024, 83:53687 – 53713.
[13] Zhiyang L,Ping L,Dinghui J,et al.Short-term Wind Power Forecasting Based on ISSA-BP Neural Network[J].Industrial Control Computer,2023,36 (02):95-97.
[14] Weiguang G,Fang W.Short-term wind power prediction based on BP optimized by an improved dung beetle optimization algorithm[J].Journal of Shanghai Dianji University,2024,27 (03):161-166+180.
[15] LI Xu,ZHANG Fan,TANG Chao,et al.A multi-objective optimization algorithm based on neural network gradient descent[J].Intelligent Computer and Applications, 2024, 14(06):224-229.
[16] TAO Kang-hui,LI Xiang.An Improved Dual-Population Multi-Objective Particle Swarm Optimization Algorithm[J]. Computer Simulation,2025,42(08):317-324.
[17] Kang sheng F,Guang yong Y,Dafei W,et al.Improved gravitational search algorithm with multi-strategy fusion[J]. Application Research of Computers, 2023,40(12):3592-3598.
[18] Yi meng L,Hui qiang W,Ding Xiao ming,et al. Optimization of nutrient solution temperature prediction model of LSTM based on improved genetic algorithm[J].Journal of Chinese Agricultural Mechanization,2025,46(6):91-97.
[19] Kewen X,Changbiao L,Junyi S.An Optimization Algorithm on the Number of Hidden Layer Nodes in Feed-forward Neural Network[J]Computer Science,2005,(10):143-145.