

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Improved Crow Search Algorithm Optimized Extreme Learning Machine Based on Classification Algorithm and Application

Li Cao^{1,3}, Yinggao Yue^{1,2}, Yong Zhang², Yong Cai³

¹ Oujiang College, Wenzhou University, Wenzhou, 325035, P. R. China

² Computer School, Hubei University of Arts and Science, Xiangyang, 441053, P. R. China

³ School of Information Engineering, Southwest University of Science and Technology, Mianyang, 621010, P. R. China

Corresponding author: Yong Zhang (e-mail: yzhangedu@yeah.net).

This work was supported in part by the Natural Science Foundation of Hubei Province under Grant 2020CFB304, the open found of distinctive disciplines in Hubei University of Arts and Science under Grant XK2020047, the educational science planning issue of Hubei Province under Grant 2020GA057, in part by the talent introduction project of Oujiang College of Wenzhou University. The Educational Science Planning Subject of Hubei Province under Grant 2020GA057, the Xiangyang Soft Science Research Project under Grant 2020rkx05.

ABSTRACT In view of the problems of the connection weights and thresholds of the extreme learning machine are randomly generated before training and remain unchanged during the training process, the number of hidden layer nodes is pre-allocated, and the hidden layer parameters are randomly selected. Too many hidden layer nodes not only make the network more complex but also reduce the generalization ability of the algorithm. Aiming at this problem, an improved crow search algorithm is proposed to optimize the extreme learning machine. Based on the analysis of the limitations of the original crow search algorithm, a particle swarm algorithm search strategy is proposed to enhance the global search capability. In the latter part of the algorithm iteration, Gaussian function is added, and the penalty coefficient of the function is used for local disturbance, gradually reducing the amplitude of the search trajectory, and then adaptively adjusting the parameters to avoid being attracted by local extremum. Finally, the improved crow search algorithm is used to optimize the hidden layer neurons and connection weights of the extreme learning machine neural network, so as to obtain accurate prediction results. Through function fitting, regression data set fitting and classification data set for classification experiment verification, the proposed algorithm has higher training speed and efficiency. At the same time, this method is not only significantly higher than the traditional ELM method, but also obtains a more compact network structure, which is an effective neural network optimization algorithm.

INDEX TERMS Extreme learning machine, crow search algorithm, particle swarm algorithm, fitting regression, data classification

I. INTRODUCTION

Artificial neural network is an intelligent computing model of bionics, which can perfectly handle linear and nonlinear problems. Among them, the single hidden layer feedforward neural network has been widely used in many fields because of its good learning ability [1,2]. However, because traditional feedforward neural networks mostly use gradient descent to modify the value of hidden layer nodes, the performance of the machine is tested in the iterative process, the model is slower to form, and it is more sensitive to the adjustment of hyper parameters [3,4]. In recent years, a new type of feedforward neural network—Extreme Learning Machine (ELM) was proposed by Huang

et al. Because of the ELM method can randomly generate constant connection weights between the input layer and the hidden layer and the hidden layer neuron threshold before training, it can overcome some shortcomings of traditional feedforward neural networks [5]. The ELM method has attracted the research and attention of many scholars and experts at home and abroad because of its fast learning speed and excellent generalization performance. The ELM algorithm has a wide range of applicability [6-9]. It is not only suitable for regression and fitting problems, but also for classification, pattern recognition and other fields, so it has been widely used. Owe to the connection weights and thresholds of ELM are randomly generated

before training and remain unchanged during training, the role of some hidden layer nodes is very small. If the data set is biased, it will even cause most of the nodes to be close to zero, so it is generally optimized and improved [10].

Classification problem is one of the most important research contents in the field of pattern recognition, data mining and machine learning, which has a wide range of applications in the real world. In recent years, swarm intelligence optimization algorithm, as an effective evolutionary computing technology, has been valued by many scholars. Swarm intelligence optimization algorithm refers to a type of algorithm inspired by the real environment. Its core idea is to achieve the balance between random behavior and local search in the search process [11-13]. At present, swarm intelligence optimization algorithms show excellent performance in solving most nonlinear and multi-modal real-world optimization problems [14, 15]. All swarm intelligence optimization algorithms use some degree of randomization and local search trade-offs. Compared with traditional search algorithms, swarm intelligence optimization algorithms can find better solutions for complex and difficult optimization problems. The genetic algorithm evolved from the evolutionary laws of the biological world, the particle swarm optimization based on the information transmission of the bird group foraging, the cuckoo search algorithm that simulates the parasitic brooding characteristics of certain species of cuckoos, and so on [16]. Swarm intelligence optimization algorithms are simple and easy to implement, few parameters, short running time, etc. Therefore, in solving many nonlinear and multi-modal real-world optimization problems, swarm intelligence optimization algorithms show excellent operability and optimization capabilities [17].

However, the traditional network model based on gradient descent and support vector network have some shortcomings, such as too long training time, too slow convergence speed, easy to over fit and so on. For the actual complex classification problems, it is difficult to get the ideal results by using the traditional classification learning methods directly [18]. How to design an efficient and generalization ability classification model is still a problem that has not been well solved. In recent years, the research and application of methods based on extreme learning machines have become more and more extensive. It overcomes the above-mentioned shortcomings of traditional BP algorithms. In the extreme learning machine method, the input weight value and the hidden layer bias value of the network are randomly generated, and the output weight can be calculated by the Moore-Penrose generalized inverse matrix. The activation function used only needs a bounded continuous function, which has extremely fast training speed and good generalization ability. However, in some specific applications, it is compared with traditional optimization methods based on gradient descent, the extreme learning machine method may require more hidden layer neurons. Too many or too few hidden layer neurons will also cause

over-fitting or under-fitting problems. During the training process, non-optimal or unnecessary weight values and thresholds may be generated, which reduces the performance of the algorithm and causes unstable results. In addition, this random assignment reduces the response speed of the extreme learning machine to unknown test data. The larger the number of neurons in the hidden layer, the more complex the network structure. An extremely complex network structure not only has a slow response speed, but also easily causes increase in computational complexity and memory consumption. Therefore, the research on the improvement of the extreme learning machine classification method is of great significance.

In this work, a novel data classification method of extreme learning machine based on the crow search algorithm optimized by particle swarm optimization (PSO-CSA-ELM) is proposed. In comparison with the current general selection approaches, the main contributions of our work in this paper can be summarized as follows:

1. Characterize the swarm intelligence optimization algorithm--crow search algorithm, and classify the current data classification methods.
2. Propose a novel data classification method of extreme learning machine based on the crow search algorithm optimized by particle swarm optimization (PSO-CSA-ELM).
3. Provide extensive simulation results to demonstrate the use and efficiency of the proposed data classification method.
4. Evaluate the performance of the proposed algorithms by comparing them with the data classification methods of the ELM, DE-ELM, PSO-KELM and CSA-ELM algorithm.

The remainder of this paper is organized as follows: Section 2 discusses the related work. Section 3 describes the basic principles of Crow Search Algorithm optimized Particle Swarm Optimization. Section 4 Describes the algorithm design idea of Extreme learning machine method based on crow search algorithm optimized by particle swarm optimization. Section 5 provides the parameters and simulation results that validate the performance of the proposed algorithm. Section 6 concludes the paper.

II. RELATED WORK

Generally speaking, the improvement of the extreme learning machine method has attracted a large number of researchers. In view of the above-mentioned shortcomings of ELM, researchers try to use the good global search ability of swarm intelligence optimization algorithm to solve such problems. Xu et al. [19] proposed a method combining particle swarm optimization (PSO) optimization algorithm and ELM. Among them, the PSO algorithm is used to optimize the input weight value and hidden layer threshold in ELM, and some prediction problems have been studied, and relatively ideal prediction results have been obtained. Li et al. [20] proposed the optimization algorithm E-ELM of the evolutionary extreme learning machine. The differential evolution (DE) algorithm is used to optimize

the important parameter values (input weight value and hidden layer threshold) in ELM, instead of the traditional BP parameter optimization algorithm, to obtain a more compact network structure and improve the accuracy of data classification. Alharbi et al. [21] used the genetic algorithm of integer coding and combined with the ELM classifier to study gene selection and cancer classification. GA algorithm is used for feature selection, redundant features are removed, and the most important features are selected as the input of ELM classifier. The results verify that the algorithm has good classification performance and can handle sparse data and data imbalance. Silitonga et al. [22] studied the effect of the random weight value connecting the input layer and the hidden layer in the ELM algorithm on the ELM performance, and the results obtained proved that the randomly set input weight value does have a greater impact on model training. In some practical classification and regression problems, these effects have a negative effect on algorithm performance. Tian et al. [23] proposed using the improved PSO method to optimize the parameters of the neural network model, and obtained better experimental results than the BP algorithm.

Owing to the connection weight and threshold of ELM are randomly generated before training and remain unchanged during the training process. Therefore, the role of some hidden layer nodes is very small. If the data set is biased, it will even cause most of the nodes to be close to zero. Therefore, in literature [24] the author pointed out that in order to achieve the desired accuracy, a large number of hidden layer nodes need to be set up. In order to solve this shortcoming, some researchers have gone to combine the optimization algorithm with the ELM method. In literature [25] the author proposed an artificial bee colony algorithm to optimize the extreme learning machine method. The algorithm uses the artificial bee colony algorithm to optimize the hidden layer node parameters of the ELM, thereby improving the performance of the ELM. According to the inspiration of evolutionary computing, in literature [26] the author obtained an adaptive evolutionary extreme learning machine through combination. The algorithm merged the relevant operators of ELM and evolutionary computing. On the basis of setting fewer parameters, the hidden layer nodes are optimized, which improves the accuracy and stability of ELM in regression and classification problems. In literature [27] the author drew on the memetic evolution mechanism of the leapfrog algorithm and proposes a hybrid intelligent optimization algorithm (SFLA-ELM) for parameter optimization. The ELM algorithm is used to obtain the output weight of the ELM, and good results are obtained. In literature [28] the author used the whale optimization algorithm to simulate and calculate the hyperparameters of the extreme learning machine randomly initialized, and finally obtain an optimal network, which is the extreme learning machine based on particle swarm optimization (WOA-ELM).

Through the above research, it can be seen that in the training process of the algorithm, the input weight, hidden layer threshold, output weight and the number of hidden layer neurons of the ELM algorithm are the essence that affects its generalization performance. There have been some researches on the optimization algorithm of ELM model, but relatively few, although DE algorithm and PSO algorithm can guide the parameter optimization of ELM to a certain extent. However, due to their different search mechanisms, they all have certain shortcomings. For example, although the DE algorithm has better global search capabilities, its convergence speed is very slow. Although the PSO algorithm has a faster convergence speed and a strong local search ability, it is easy to enter the local minimum and cannot find the global optimal solution. The combination of swarm intelligence optimization algorithm and artificial neural network can further improve the performance of ELM algorithm, but due to the complexity of the optimization model itself and the lack of theoretical foundation. The research on this issue has not yet reached a satisfactory level, so further research on this issue is necessary.

Based on the above ideas, in order to overcome the shortcomings of the ELM method, the generalization ability of ELM is further improved. This paper proposes to optimize the ELM algorithm based on the particle swarm optimization crow search algorithm to verify whether the proposed method can get no lower or even higher performance than the existing ELM optimization algorithm. In order to provide a new alternative algorithm and a new way of ELM optimization method, the reason why we use the improved crow search algorithm to optimize the ELM model is that the PSO, DE, and GA algorithms are all random intelligent optimization algorithms. These three algorithms have developed longer than the improved crow search algorithm, and their research and application are more mature. There are relatively few researches on crow search methods, but the algorithm is simple and easy to implement, has low computational overhead, has good global search capabilities, and has fewer parameters to be adjusted in the algorithm. It can not only deal with continuous optimization problems, but also solve combinatorial optimization problems. The performance of dealing with some complex optimization problems is better than these three algorithms.

III. CROW SEARCH ALGORITHM OPTIMIZED PARTICLE SWARM OPTIMIZATION

In 2016, Askarzadeh proposed a new optimization algorithm based on the foraging behavior of crow flocks in nature. Crow search algorithm (CSA) is an intelligent meta-inspired algorithm. Crows are highly intelligent birds that live in groups [29]. After they find food, they usually hide the excess food. The hiding position is called memory. Take it out when needed, and steal food from other crows by tracking other crows. The tracked crows can protect their food with a certain Awareness Probability (AP) to prevent theft. The crow search algorithm has made some

research results in the fields of network optimization and distribution, medical testing and so on [30]. The crow search algorithm has only two parameters (flight length and perception probability), and the crow search algorithm is easy to implement and its convergence speed is fast. Therefore, the crow search algorithm has certain application research value in different fields, and has stronger competitiveness compared with other intelligent optimization algorithms [31].

Set a reasonable number of iterations $iter$, $X^{i,iter} = x_1, x_2, x_3, \dots, x_n$ is the crow coordinate in the n -dimensional space, $Memory_i (m^i = m_1, m_2, m_3, \dots, m_n)$ is the storage point for hidden food, and the storage is used to represent the current best coordinate obtained by the crow individual i . One of the cores of the crow algorithm is the process of chasing each other between individual crows. This process is used to screen the current best points to generate new current best points. The process is mainly divided into two different states. The two different states depend on whether the individual crow perceives other crows chasing itself [32].

If the crow individual i is the chasing individual and j is the chased individual, when the crow individual j is not aware of the existence of the stealer, this situation is attributed to the state A, that is, the crow individual j has not found the tracking of the crow individual i . Then in this state, the crow i will generate new coordinates, which can be expressed as:

$$X^{i,iter+1} = X^{i,iter} + r_i \times fl^{i,iter} \times (m^{j,iter} - X^{i,iter}) \quad (1)$$

The parameter r_i represents a random number between $[0,1]$, and the parameter $fl^{i,iter}$ is the flying distance of the crow i in the $iter$ iteration. The size of the flight distance has a different effect on the search ability of the algorithm. A smaller fl value is helpful for individual crows to perform a local search, while a larger fl value can guide the crows for a global search.

The other state is state B. In this state, the crow individual j finds the tracking of the crow individual i . In order to prevent the crow i from discovering the coordinates of its hidden food, it will mislead the crow i and disturb its audiovisual, so that the crow i will drop to a random coordinate in the search space. Therefore, these two different chasing states can be summarized as:

$$X^{i,iter+1} = \begin{cases} X^{i,iter} + r_i \times fl^{i,iter} \times (m^{j,iter} - X^{i,iter}), & r_i \geq AP^{j,iter} \\ a \text{ random position} & \text{otherwise} \end{cases} \quad (2)$$

The parameter $AP^{j,iter}$ represents the awareness probability (Awareness Probability, AP) of the crow individual j . In the crow search algorithm, the probability of consciousness controls the convergence and diversity of the algorithm. In the crow search algorithm, the probability of consciousness controls the convergence and diversity of the algorithm. Different AP values play different roles. When the AP value is reduced, the algorithm tends to perform a

local search. A smaller AP value can increase the convergence of the algorithm. When the value of AP increases, the crow search algorithm will tend to search globally, and the probability of searching nearby areas to obtain a solution will decrease, but the diversity of the algorithm can be increased [33].

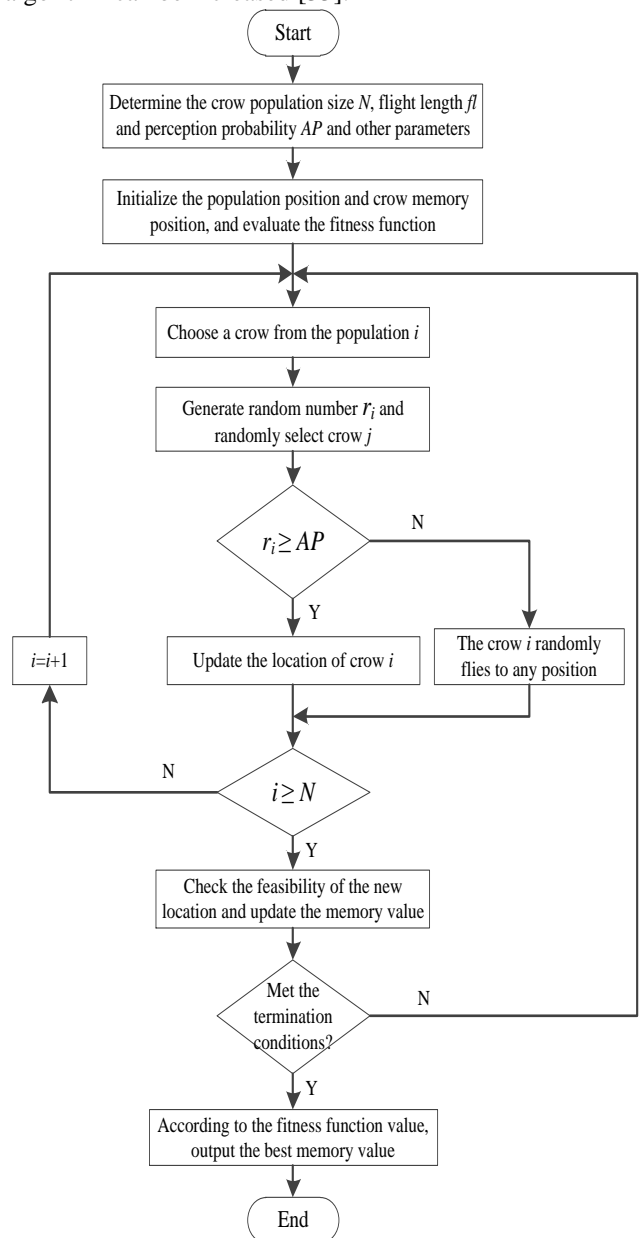


FIGURE 1. Flow chart of crow search algorithm

Since the position generated by the global search of the crow algorithm is completely random, in the crow algorithm, the crow j is found when the crow i is tracking ($r_i < AP$). In order to protect its food from being stolen, crow j will deceive crow i and fly randomly to any position in the solution space, although this mechanism can prevent the crow algorithm from falling into a local optimal solution. But it also makes the global optimization of the crow algorithm too blind, which is not conducive to the

rapid convergence of the algorithm. Secondly, the position of the initial population of the crow algorithm is given randomly, and the position of the initial population is easy to be concentrated, which is easy to fall into the local optimal solution. Finally, the flight length of the crow algorithm is a fixed value. At the later stage of the iteration, a smaller flight length needs to be used for local optimization. The fixed flight length greatly affects the optimization speed. The above shortcomings greatly affect the optimization efficiency of the crow algorithm on the extreme learning machine, so it is necessary to improve the crow algorithm to obtain a better CSA-ELM model. Aiming at the shortcomings of the crow search algorithm, this paper proposes an extreme learning machine method based on the crow search algorithm optimized by particle swarm optimization (PSO-CSA-ELM). In order to enhance the optimization ability of the crow algorithm, according to the structure of the crow algorithm, the crow algorithm is improved from three aspects:

(1) In the population initialization stage, the Tent chaos based on reverse learning is used for initialization.

(2) In terms of flight length, according to the increase in the number of iterations, the size of the flight length is adaptively reduced.

(3) In the global search, the particle search strategy of the particle swarm optimization algorithm is used to search to expand the search range of the crow search algorithm.

IV. EXTREME LEARNING MACHINE METHOD BASED ON CROW SEARCH ALGORITHM OPTIMIZED BY PARTICLE SWARM OPTIMIZATION

The original crow search algorithm used a random initialization method when initializing the population. The randomness of the initial solution would lead to poor distribution and affect the convergence speed of the experiment. The crow search algorithm tries to maintain a balance between diversity and convergence through flight length (fl) and perceived probability (AP). The smaller the value of the flight length fl is, the more likely it is to cause a local search. On the contrary, the larger the value of fl is, the more likely it is to cause a global search. These parameters are selected by the user before the algorithm is executed, and the state of the solution obtained during the execution is not considered when selecting these parameters, which may cause the final solution to easily fall into the local optimum.

The PSO-CSA-ELM model proposed in this paper combines the improved crow algorithm with the ELM model. The input weight value and hidden layer threshold in the ELM are intelligently optimized by the CSA algorithm, and the output weight value is calculated by the Moore-Penrose generalized inverse matrix. Then train and construct the optimized ELM model on the classification data set, and finally use the test set to obtain the classification results of the model. The specific process of the PSO-CSA-ELM algorithm is as follows:

Step1: Define the relevant parameters of the PSO-CSA algorithm and the extreme learning machine

algorithm, randomly generate a d-dimensional crow population, and then initialize the crow population by mapping based on the chaos method. The 5-fold CV method is used to divide the training data set into five subsets, four of which are used as training sets and the remaining one is used as test sets.

Step2: Initialize the population, randomly generate initial solutions and encode them. The dimension of each solution is $L \times (n+1)$, and the first $L \times n$ dimension represents the input weight value. The remaining L dimensions represent hidden layer thresholds, and they are all continuous real numbers.

Step3: Initialize the parameters of the algorithm, including the number of populations N , the upper and lower bounds of the population, and the maximum number of iterations $Genmax$.

Step4: Use the solution obtained in step 2 to decode, obtain the input weight value and the hidden layer threshold, and train the ELM model on the training data set. Note that the solution obtained is actually a vector value, but in the training process it is actually an $L \times (n+1)$ -dimensional matrix, which needs to be converted into a vector form in advance.

Step5: Calculate the fitness value corresponding to each solution.

Step6: Add one iteration. $iter = iter + 1$.

Step7: In the employment bee phase, update each solution.

Step8: Use the input weight value and hidden layer threshold value obtained in step 7 to calculate the fitness value corresponding to each solution.

Step9: A crow j is randomly selected. According to the perception probability AP , if the random number r_j is greater than or equal to the perception probability, then the crow i follows the crow j and flies to the memory position of the crow j . If the random number r_j is less than the perception probability AP , in order to deceive crow i , crow j will fly to another location according to the particle search strategy in the particle swarm algorithm.

Step10: The current crow position is used as the ELM model parameter and the data is predicted, and the prediction result is converted into a fitness function value and compared with the fitness function value of the memory position of the crow. If it is better than the memory position, the memory position is updated to the current position.

Step11: Perform the above steps for all crows, iterate the number of times specified in the above steps, and return the global optimal position as the initial input weight and threshold of the ELM prediction model.

Step12: Use the new solution obtained in step 11 to train the ELM model and calculate its fitness value.

- Step13: Determine whether the algorithm has reached the maximum number of iterations, if it is satisfied, go to step 14; otherwise, return to step 6 to continue running the algorithm.
- Step14: Decoding from the returned optimal solution can obtain the optimal input weight value and the threshold value of the hidden layer.
- Step15 : Use the trained ELM model to perform classification tests and record the final classification results.

The flow chart of the PSO-CSA-ELM algorithm is shown in Figure 2.

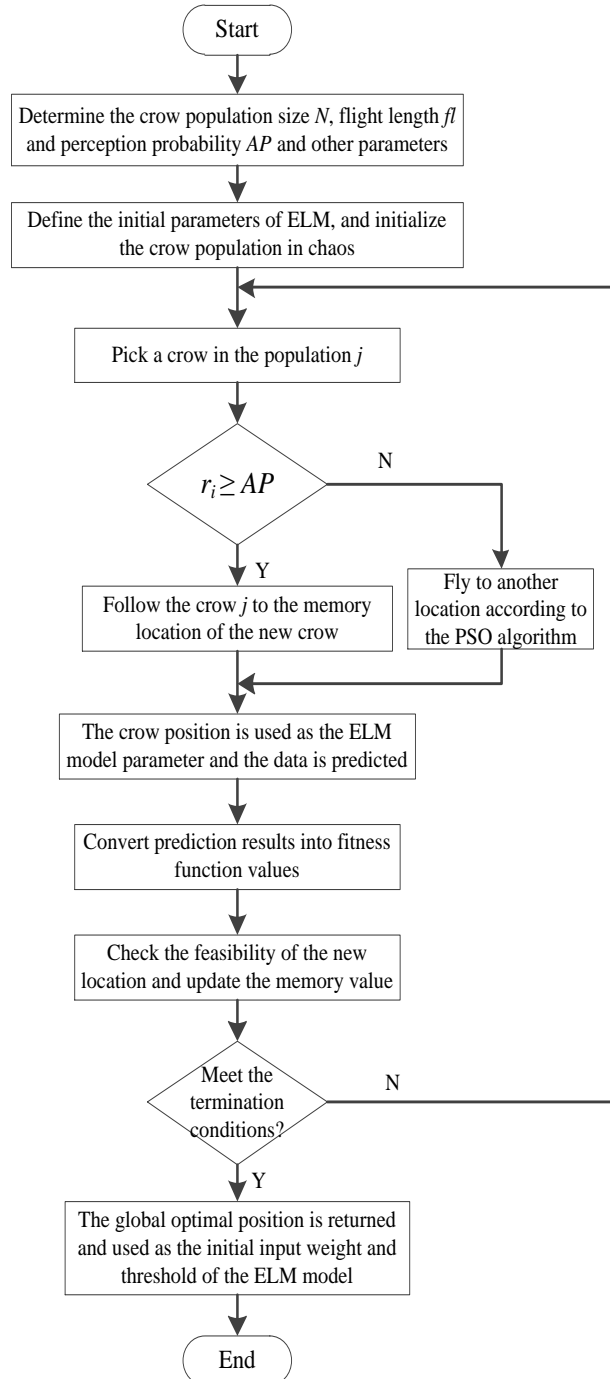


FIGURE 2. Flow chart of PSO-CSA-ELM algorithm

The time complexity indirectly reflects the length of time the algorithm executes. In the CSA-ELM algorithm, it is assumed that the execution time required to initialize the parameters (under the condition that the population size is N and the spatial dimension is n) is x_1 , and the time to generate a uniform distribution is x_2 . The time required to find the fitness value is $f(n)$, then the time complexity of the initial stage of the CSA-ELM algorithm is as follows:

$$O(x_1 + N(nx_2 + f(n))) = O(n + f(n)) \quad (3)$$

Assuming that the execution time required for the iterative update of each dimension of the individual is the same, which is x_3 , the time for comparing the advantages and disadvantages and selecting the best after iteration is x_4 . The calculation time of the flight length of the crow is x_5 , and the time consumption of the Awareness Probability is x_6 , then the time complexity of the algorithm at this stage is:

$$O(N(nx_3 + f(n)) + x_4 + x_5 + x_6) = O(n + f(n)) \quad (4)$$

Therefore, the total time complexity of the CSA-ELM algorithm to solve each generation's optimal is:

$$T(n) = O(n + f(n)) + O(n + f(n)) = O(n + f(n)) \quad (5)$$

In the improved PSO-CSA-ELM algorithm, the time required for the initialization phase of the algorithm is basically the same as the CSA-ELM algorithm. Therefore, the time complexity of the initialization phase of the improved algorithm is the same as equation (12). In the algorithm loop, suppose the calculation time of the weighted center is z_1 , the calculation time of the individual learning position is z_2 , and the calculation time of the comparison and selection process between the learning individual and the initial individual is z_3 . Then the time complexity of the loop part is:

$$O(N(nx_3 + f(n)) + x_4 + x_5 + x_6 + N(z_2 + z_3) + z_1) = O(n + f(n)) \quad (6)$$

Therefore, the total time complexity of the improved PSO-CSA-ELM algorithm to solve the optimal of each generation is:

$$T(n) = O(n + f(n)) + O(n + f(n)) = O(n + f(n)) \quad (7)$$

In summary, the improved strategy of the improved PSO-CSA-ELM algorithm does not increase the time complexity of the algorithm solution compared to the initial CSA-ELM algorithm.

V. ALGORITHM SIMULATION COMPARISON AND ANALYSIS

A. SIMULATION ENVIRONMENT SETTINGS

To verify the performance of the proposed algorithm, this article conducts experiments on eight data sets, which are Computer Hardware, QSAR Aquatic Toxicity, Real Estate Valuation, Servo, Bupa Liver, Cleveland Heart, Breast Cancer and iris data sets. The first 4 are used for regression and the last 4 are used for classification. These experimental data sets are all from well-known open source

databases-the University of California Irvine provides a database for machine learning. Before the experiment, the data needs to be preprocessed first. Because the Australian, Breast Cancer and Cleveland Heart data sets have missing features, in order to ensure the integrity of the sample data, this experiment has performed an average processing method on these records. At the same time, in order to reduce the difference between the eigenvalues and prevent the larger eigenvalues from overly affecting the smaller eigenvalues, we normalize each eigenvalue to the interval [-1,1].

In order to verify that the PSO-CSA-ELM algorithm has better performance in terms of convergence and optimization speed, the proposed PSO-CSA-ELM algorithm is compared with the CSA-ELM algorithm, PSO-ELM algorithm and DE-ELM algorithm on function test, regression and classification data sets. The maximum number of population evolution in all experiments is set to 50, and the population size of the algorithm is 30. All experiments were run 50 times and the root mean square error or the average and standard deviation of the classification accuracy were taken as the experimental results. In the crow search algorithm (CSA), the crow flight length fl and the awareness probability are 2 and 0.1 respectively. The parameter settings of the PSO algorithm: learning factor c_1 is 2, c_2 is 2, inertia weight factor ω_1 is 0.9, ω_2 is 0.4, and the maximum number of iterations T_{max} is 50. Evolutionary algorithm (DE), the parameter scaling factor F and crossover probability CR are 1 and 0.8, respectively.

B. TEST OBJECTIVE FUNCTION OPTIMIZATION

(1) Sinc function simulation experiment comparison

The four algorithms are compared by fitting the Sinc function. The expression of the Sinc function is as follows:

$$f(x) = \begin{cases} \frac{\sin(x)}{x}, & x \neq 0 \\ 0, & x = 0 \end{cases} \quad (8)$$

We set to generate 1000 [-10,10] uniformly distributed data sets x , and calculate 1000 data sets $\{x_i, f(x_i)\}, i = 1, 2, 3, \dots, 1000$, Then generate 1000 [-0.2,0.2] uniformly distributed noises ε . Let the training set be $\{x_i, f(x_i) + \varepsilon_i\}, i = 1, 2, 3, \dots, 1000$, and then generate another set of 1000 data sets $\{y_i, f(y_i)\}, i = 1, 2, 3, \dots, 1000$ as the test set. Gradually increase the number of iterations of the five algorithms to fit the function.

Root Mean Square Error (RMSE) and Standard Deviation (Std. Dev) indicators are used as evaluation indicators for error analysis. The calculation formulas of the two indicators are as follows:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y(i) - y'(i))^2} \quad (9)$$

$$Std.Dev = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (y'(i) - \bar{y}'(i))^2} \quad (10)$$

Among them, the smaller the index values of RMSE and Std. Dev, the lower the forecast error. The Sinc function fitting results are shown in Table 1.

TABLE 1. Comparison of Sinc function fitting results

Number of hidden layer nodes	Performance	PSO-CSA-ELM	CSA-ELM	DE-ELM	PSO-ELM	ELM
1	RMSE	0.3236	0.3433	0.3529	0.3563	0.3607
	Std.Dev.	0.0175	1.6784e-06	0.0028	1.9267e-04	0.0038
2	RMSE	0.1097	0.1232	0.1257	0.2122	0.3502
	Std.Dev.	0.0020	0.0057	0.0069	0.0030	0.0093
3	RMSE	0.0873	0.0905	0.0996	0.1072	0.2746
	Std.Dev.	0.0090	0.0115	0.0059	0.0094	0.0583
4	RMSE	0.0685	0.0711	0.0713	0.0721	0.1903
	Std.Dev.	0.0034	0.0020	0.0028	0.0057	0.0684
5	RMSE	0.0507	0.0566	0.0603	0.0619	0.1623
	Std.Dev.	0.0092	0.0046	0.0022	0.0078	0.0722
6	RMSE	0.0227	0.0278	0.0340	0.0335	0.1286
	Std.Dev.	0.0067	0.0014	0.0068	0.0088	0.0520
7	RMSE	0.0155	0.0167	0.0170	0.0173	0.0817
	Std.Dev.	0.0011	0.0043	0.0021	0.0022	0.0085

8	RMSE	0.0088	0.0097	0.0114	0.0137	0.0704
	Std.Dev.	0.0012	0.0012	0.0017	0.0012	0.0108
9	RMSE	0.0045	0.0057	0.0063	0.0098	0.0545
	Std.Dev.	1.0839-04	2.1032e-04	9.1469e-04	7.3011e-04	0.0259
10	RMSE	0.0043	0.0047	0.0049	0.0050	0.0334
	Std.Dev.	1.9953-04	1.3988-04	4.3264-04	2.8901-04	0.0164
11	RMSE	0.0036	0.0040	0.0045	0.0046	0.0182
	Std.Dev.	2.6949e-05	8.7440e-05	1.3726e-04	4.4726e-05	0.0104

It can be seen from Table 1 that the RMSE and Std. Dev index values of the basic ELM method are the largest and the performance is the worst. Calculated by the PSO-ELM algorithm, the index values of RMSE and Std. Dev are large, and the performance of the test results is poor. The index values of RMSE and Std. Dev of the DE-ELM algorithm are large, and the test results have poor performance. The RMSE and Std. Dev index values of the CSA-ELM algorithm are smaller, and the test results have better performance. The RMSE and Std. Dev index values of the PSO-CSA-ELM algorithm are the smallest, and the test results have the best performance. It shows that the error of the PSO-CSA-ELM algorithm model is relatively smaller, and the prediction accuracy is better than the ELM, PSO-ELM, DE-ELM and CSA-ELM algorithms. At the same time, it can be seen from Table 1 that as the number of hidden layer nodes increases, the average test error and standard deviation gradually decrease. When there are too many hidden layer nodes, over-fitting will occur. Because the CSA-ELM algorithm is easy to fall into the local optimal solution and other shortcomings, the effect is still poor when the number of nodes is high. In most cases, when the number of hidden layer nodes is the same, the PSO-CSA-ELM algorithm has a smaller average test error and standard deviation.

(2) Comparison of classification data set

In this paper, we compare the performance of the four algorithms using four real regression data sets in the machine learning library of the University of California,

Irvine. The names of the data sets are: Breast Cancer, Bupa Liver, Cleveland Heart and Iris. In the experiment, the data in the data set is randomly divided into training set and test set, 70% of which are used as training set and the remaining 30% are used as test set. In order to reduce the influence of the large difference of each variable, we normalize the data before the algorithm runs, that is, the input variable is normalized to $[-1,1]$, and the output variable is normalized to $[0,1]$. In all experiments, the algorithm iterates 50 times, and calculates the average of 50 experimental results.

In order to evaluate the effectiveness of the PSO-CSA-ELM algorithm, we conduct a series of experiments on four classification data sets. The range of the number of hidden layer neurons in the algorithm is set to $[5, 25]$, and the step size is 5. The reason why this section of the experiment chooses this range is because for ELM, too large number of hidden layer neurons will cause the ELM algorithm to produce over-fitting problems as the number of nodes increases. In addition, the ABC method can find better parameters, so that the algorithm only needs fewer hidden layer neurons and achieves more stable results. Tables 2, 3, 4 and 5 show the comparison of the results of the 50% CV on the four data sets of Breast Cancer, Bupa Liver, Cleveland Heart, and Iris, respectively. Performance evaluation criteria include training classification accuracy, test classification accuracy, standard square deviation, the number of hidden layer neurons required, output weight norm, and training time. In addition, we add a large data set for simulation test comparison, the electrical grid stability simulated data set. The electrical grid stability simulated data set consists of 10000 data sets, each of which has 14 eigenvalues. Comparison of electrical grid stability simulated data set fitting results is shown in Tables 6.

TABLE 2. Comparison of Breast Cancer data set fitting results

Data set	Algorithms	Accuracy (%)			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
Breast Cancer	PSO-CSA-ELM	95.3529	98.6892	1.2187	20	31.7435	44.7188
	CSA-ELM	96.8884	98.3908	0.9525	20	17.1606	46.6375
	PSO-ELM	96.4511	98.6860	1.2261	20	6.4844	46.3750
	DE-ELM	96.6324	98.3908	0.9525	20	8.9041	48.0125
	ELM	96.4863	95.4591	2.1461	20	7.5776	0.0219

TABLE 3. Comparison of Bupa Liver data set fitting results

Data set	Algorithms	Accuracy (%)			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
Bupa Liver	PSO-CSA-ELM	73.4699	81.9642	1.2187	20	99.4231	25.0438
	CSA-ELM	74.6399	78.7292	0.9525	20	437.2482	24.8688
	PSO-ELM	75.5726	77.8541	1.2261	20	92.1378	24.6063
	DE-ELM	74.9201	78.1618	0.9525	20	148.4902	26.3375
	ELM	75.8606	61.9063	2.1461	20	96.4493	0.0067

TABLE 4. Comparison of Cleveland Heart data set fitting results

Data set	Algorithms	Accuracy (%)			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
Cleveland Heart	PSO-CSA-ELM	81.6463	93.9492	1.8798	10	2.8641	21.2438
	CSA-ELM	84.5094	93.2712	2.0480	10	5.7329	21.3063
	PSO-ELM	83.5872	92.9322	2.1936	10	3.4483	21.2000
	DE-ELM	83.6702	92.2599	2.2436	10	3.3499	22.9188
	ELM	81.6520	82.8305	4.3716	10	3.6049	0.0056

TABLE 5. Comparison of Iris data set fitting results

Data set	Algorithms	Accuracy (%)			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
Iris	PSO-CSA-ELM	96.5079	100.0000	0.0000	10	80.4465	11.3344
	CSA-ELM	96.8315	100.0000	0.0000	10	1305.1750	12.0563
	PSO-ELM	96.9967	100.0000	0.0000	10	341.1708	11.2156
	DE-ELM	97.8317	100.0000	0.0000	10	231.2270	13.5375
	ELM	96.9995	97.4353	4.0894	10	239.5609	0.0036

TABLE 6. Comparison of electrical grid stability simulated data set fitting results

Data set	Algorithms	Accuracy (%)			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
Electrical Grid Stability Simulated	PSO-CSA-ELM	99.5470	98.6187	0.1723	30	11.9979	1473.0469
	CSA-ELM	99.1638	98.5000	0.1132	30	74.2956	1469.7656
	PSO-ELM	99.0592	98.3303	0.1458	30	16.9637	1451.6875
	DE-ELM	98.9286	98.2187	0.0972	30	18.9183	1442.7813
	ELM	98.9170	97.4167	0.1458	30	17.6571	0.2813

As can be seen from Tables 2, 3, 4, and 5, PSO-CSA-ELM achieved the best results in these four classification data sets, and the number of neurons used to achieve the best results was also the least. This shows that the proposed PSO-CSA algorithm can effectively optimize the parameters and obtain a more compact network structure. It also shows that the PSO-CSA algorithm is used to optimize the ELM model, which can achieve better classification performance and generalization ability. It can also be seen

from Tables 2, 3, 4 and 5 that the standard square deviation obtained by the PSO-CSA-ELM algorithm is also the smallest, which also shows that the algorithm has good stability. From the comparison of training time in Tables 2, 3, 4 and 5, it can be seen that since ELM random generation parameters do not need to be adjusted, the training speed is very fast, but the classification effect is not good. Compared with the ELM, DE-ELM, PSO-ELM and CSA-ELM algorithms, the PSO-CSA-ELM algorithm has little

difference in training time, and does not show the advantage of efficiency. However, due to the ELM model constructed by the PSO-CSA optimization algorithm, our algorithm has achieved better classification accuracy in the classification results, so the calculation efficiency is acceptable. It can be seen from the large number of electronic grid stability simulated data sets in Table 6 that the proposed algorithm also has better classification accuracy, better effect and minimum error than the other four algorithms.

In this experiment, we analyzed the performance of the algorithm with the increasing number of hidden layer neurons, as shown in Figure 3.

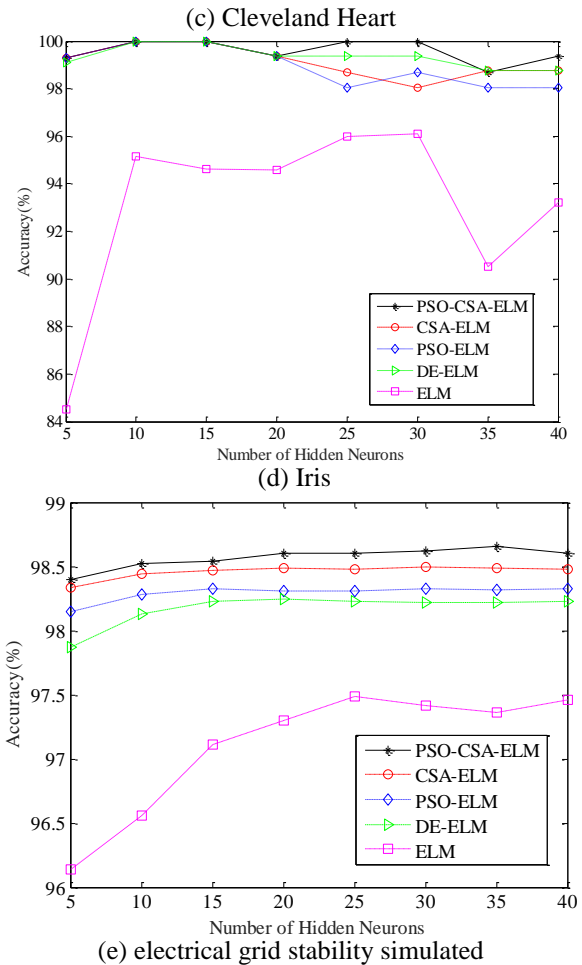
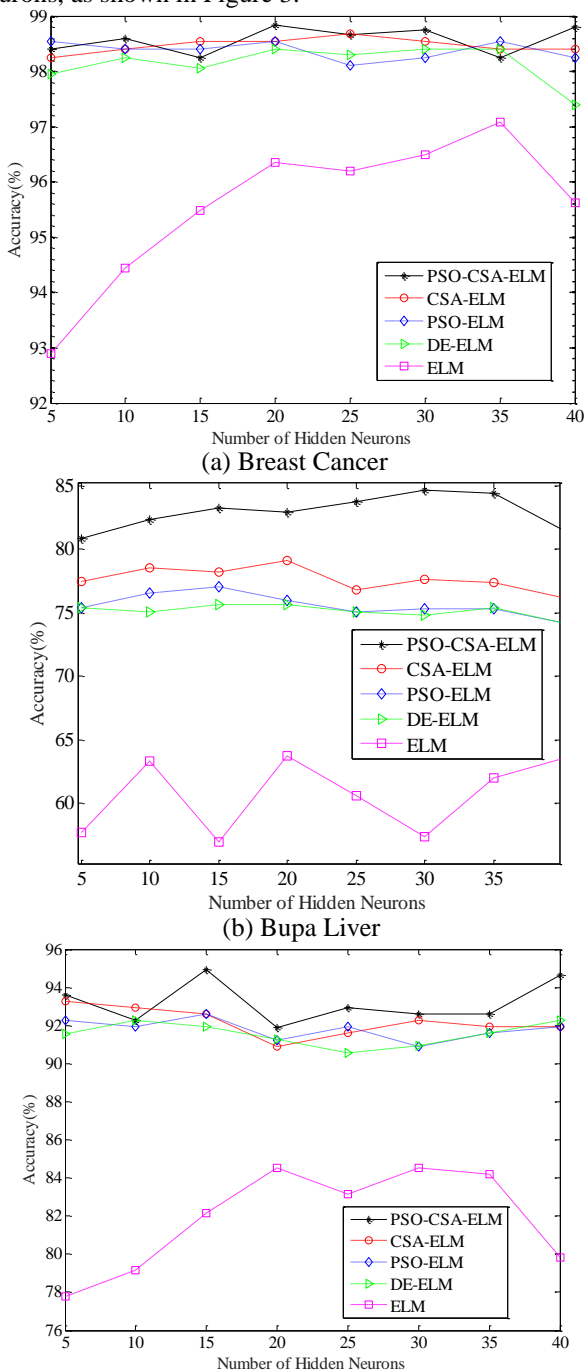


FIGURE 3. The classification accuracy of the five algorithms on different data sets varies with the number of hidden layer nodes

Figure 3 shows the comparison of their changes on different classification data sets. Figure 3 (a) shows the changes of the five algorithms on Breast Cancer. It can be seen from Figure 3(a) that the curve of PSO-CSA-ELM changes relatively smoothly, and the highest classification accuracy is obtained when the number of neurons is equal to 20. Even when the number of neurons is small, the result is higher than 70%. In contrast, the original ELM and DE-ELM have poor results when the number of neurons is less than 15, which may cause underfitting problems. Figure 3(b) shows the changes of the algorithm on the Bupa Liver data set. From Figure 3(b), it can be seen that PSO-CSA-ELM has the highest classification accuracy. Although PSO-CSA-ELM reached the highest value when the number of neurons was 30, PSO-CSA-ELM achieved 95.37% classification accuracy compared to other algorithms with the same number of neurons. For the Cleveland Heart data set, as shown in Figure 3(c), it can be seen from the figure that our proposed method is relatively close to other improved ELM methods, but only requires 15 neurons and the smallest variance value is obtained. For the Iris data set, it can be seen from Figure 3(d) that our proposed method is superior to several other methods. In addition, as the

number of neurons increases, the classification accuracy also increases. However, when the number of neurons is greater than 30, the results of ELM and PSO-ELM drop significantly, which may cause overfitting problems. However, the PSO-CSA-ELM algorithm proposed in this paper has small fluctuations, which shows that the algorithm has good stability. From the data classification effect of the big data set in Figure 3(e), the algorithm proposed in this paper has the highest classification accuracy and the best effect.

(3) Regression test data set simulation experiment comparison

Similarly, we tested the four regression data sets of Computer Hardware, QSAR Aquatic Toxicity, Real Estate Valuation and Servo. The test results are shown in Tables 7, 8, 9 and 10. The classification accuracy of the five algorithms varies with the number of hidden layer nodes as shown in Figure 4.

TABLE 7. Computer Hardware data set classification results

Data set	Algorithms	RMSE			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
Computer Hardware	PSO-CSA-ELM	0.0081	0.0066	0.0018	20	5.1228	55.2188
	CSA-ELM	0.0059	0.0149	0.0027	20	11.6038	54.4688
	PSO-ELM	0.0068	0.0172	0.0044	20	4.0699	53.6875
	DE-ELM	0.0070	0.0178	0.0031	20	3.7342	55.2875
	ELM	0.0078	0.0736	0.0368	20	4.8922	0.0188

TABLE 8. QSAR Aquatic Toxicity data set classification results

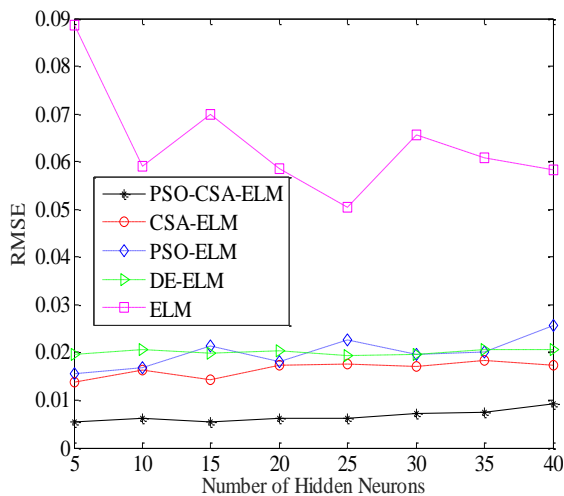
Data set	Algorithms	RMSE			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
QSAR Aquatic Toxicity	PSO-CSA-ELM	0.1128	0.1085	0.0174	20	25.0869	61.9313
	CSA-ELM	0.1162	0.1484	0.0247	20	36.5336	62.1063
	PSO-ELM	0.1190	0.1525	0.0236	20	8.2727	60.9313
	DE-ELM	0.1168	0.1578	0.0263	20	15.0644	62.4813
	ELM	0.1182	0.2195	0.0467	20	10.1408	0.0879

TABLE 9. Real Estate Valuation data set classification results

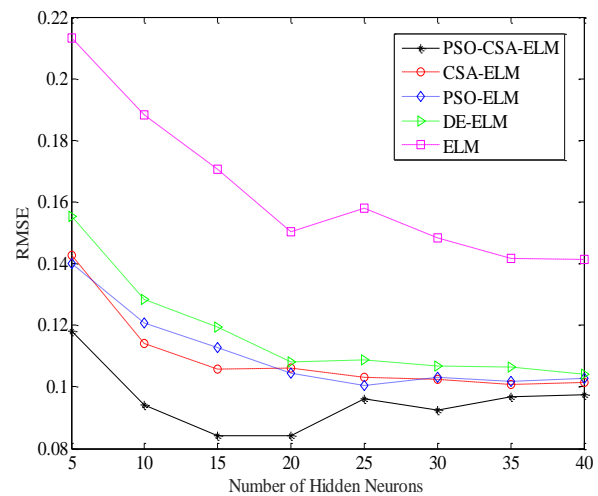
Data set	Algorithms	RMSE			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
Real Estate Valuation	PSO-CSA-ELM	0.0896	0.1349	0.0098	10	13.2982	52.1250
	CSA-ELM	0.0814	0.1623	0.0091	10	32.9739	51.4344
	PSO-ELM	0.0828	0.1625	0.0065	10	3.3832	50.6375
	DE-ELM	0.0832	0.1726	0.0127	10	2.3665	52.0094
	ELM	0.0875	0.1988	0.0238	10	2.0844	0.078

TABLE 10. Servo data set classification results

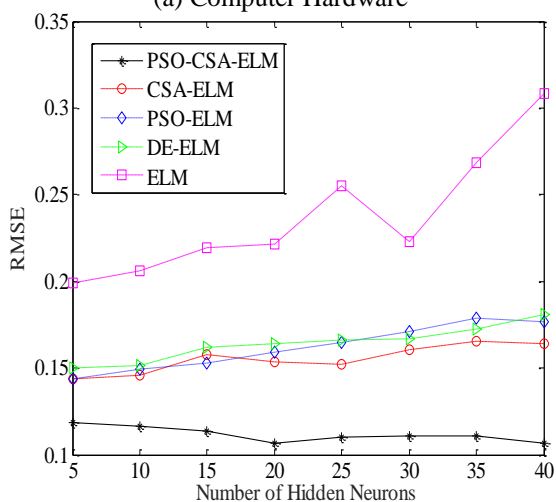
Data set	Algorithms	RMSE			Hidden nodes	Output weight norm	Time (s)
		Training set	Test set	Std.Dev.			
Servo	PSO-CSA-ELM	0.1155	0.0877	0.0146	10	1.2445	60.4688
	CSA-ELM	0.1013	0.1066	0.0201	10	75.3555	61.0813
	PSO-ELM	0.1091	0.1108	0.0193	10	15.8373	59.8156
	DE-ELM	0.1034	0.1230	0.0202	10	9.6533	61.1938
	ELM	0.1317	0.1902	0.0256	10	7.8216	0.0062



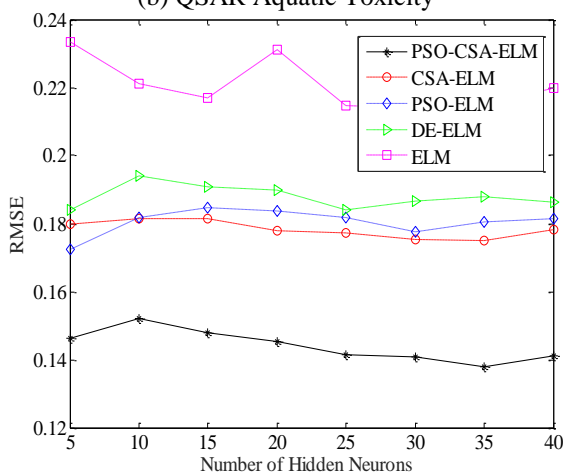
(a) Computer Hardware



(d) Servo



(b) QSAR Aquatic Toxicity



(c) Real Estate Valuation

FIGURE 4. The classification accuracy of five algorithms varies with the number of hidden nodes

It can be seen from Tables 6, 7, 8 and 9, that the ELM method has the largest root mean square error, the largest variation range, and poor performance. The root mean square error of the DE-ELM method is larger, the root mean square error of the PSO-ELM method is also larger, and the root mean square error of the CSA-ELM method is smaller. The PSO-CSA-ELM algorithm proposed in this paper has the smallest root mean square error and the best classification result. The root mean square error obtained by the proposed PSO-CSA-ELM algorithm is the smallest, which also shows that the algorithm has good stability. From the comparison of training time in Tables 6, 7, 8 and 9, it can be seen that since ELM randomly generates parameters without adjustment, its training speed is very fast, but the classification effect is not good. Compared with the ELM, DE-ELM, PSO-ELM and CSA-ELM algorithms, the PSO-CSA-ELM algorithm has little difference in training time, does not reflect the advantage of efficiency, and has little difference in training time. From the four regression classification results shown in Figure 4, whether it is Computer Hardware, QSAR Aquatic Toxicity, or Real Estate Valuation, Servo data sets, the ELM classification method has the worst performance. The PSO-CSA-ELM algorithm proposed in this paper has the highest classification accuracy and the best performance.

(4) Simulation experiment comparison of speech signal classification data set

Speech feature signal recognition and classification is an important aspect in the field of speech recognition research, which is generally solved by the principle of pattern matching. This paper selects four different types of music, namely famous songs, guzheng, rock and pop, and uses the PSO-CSA-ELM classification method proposed in this paper to effectively classify these four types of music. Each piece of music uses the cepstrum coefficient method to extract 500 groups of 24-dimensional speech feature signals. The speech feature signal classification algorithm modeling based on the PSO-CSA-ELM method includes

three steps: PSO-CSA-ELM neural network construction, PSO-CSA-ELM neural network training and PSO-CSA-ELM neural network classification. The comparison of the speech feature signals of the five algorithms is shown in Figure 5. Figure 5(a) is the prediction result of the ELM algorithm, Figure 5(b) is the prediction result of the DE-ELM algorithm, and Figure 5(c) is the prediction result of the PSO-ELM algorithm. Figure 5(d) is the prediction result of the CSA-ELM algorithm, and Figure 5(e) is the prediction result of the PSO-CSA-ELM algorithm. Figure 6 is a graph showing changes in the prediction accuracy of five algorithms with the number of hidden layer nodes. Figure 7 is a comparison of the prediction accuracy of the five algorithms.

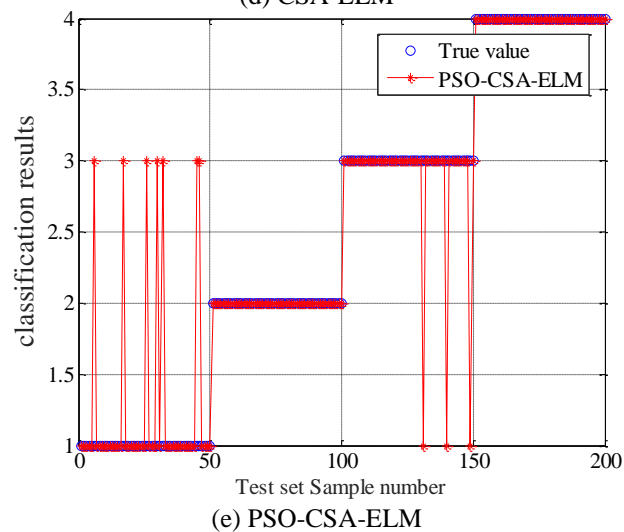
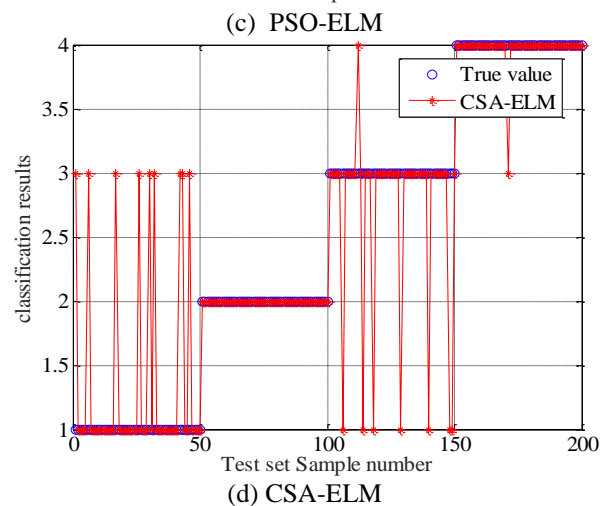
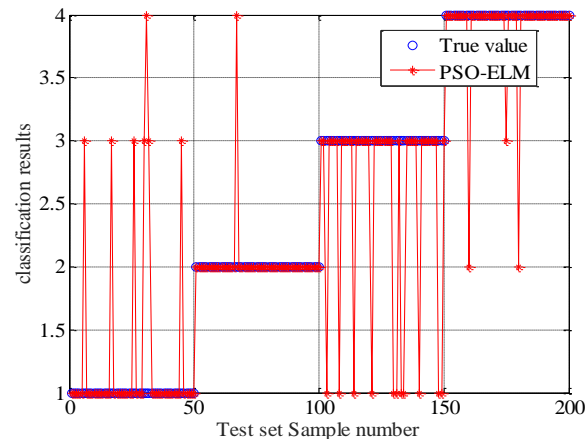
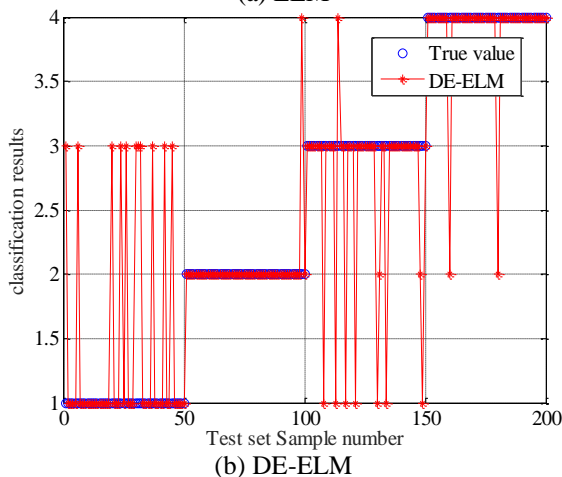
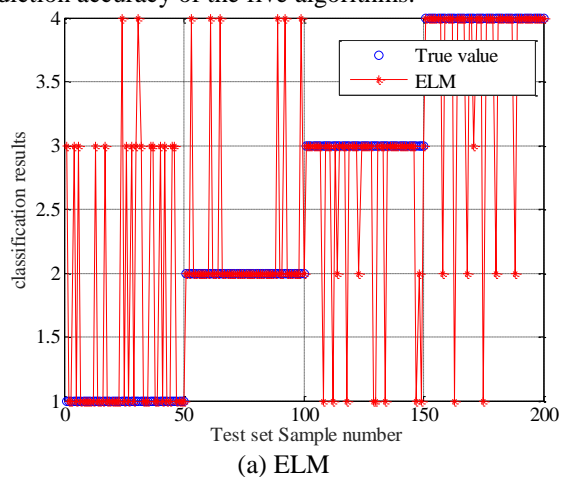


FIGURE 5. Comparison of speech feature signal classification results of five algorithms

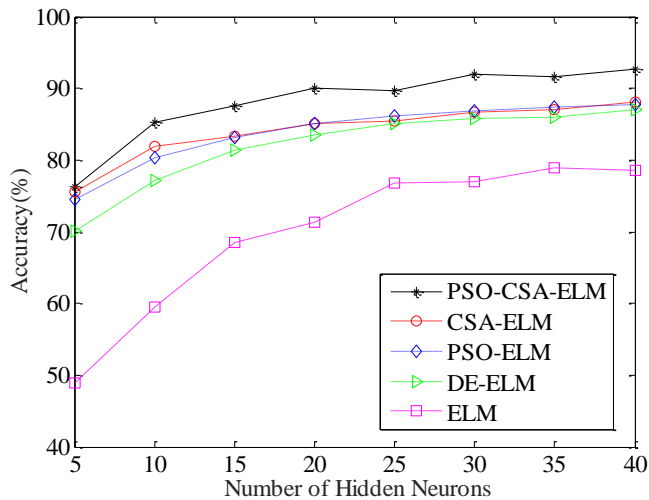


FIGURE 6. The prediction accuracy of five algorithms varies with the number of hidden nodes

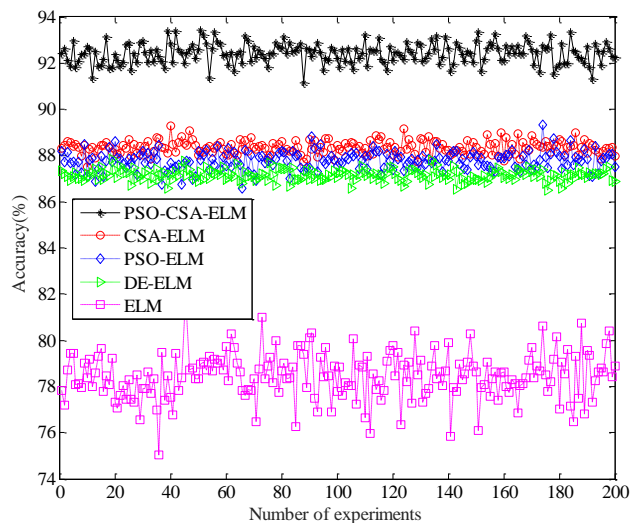


FIGURE 7. Comparison of prediction accuracy of five algorithms

It can be seen from Figure 5 that the prediction result of the ELM algorithm in Figure 5(a) has the worst performance, with an accuracy of 78.5%. The prediction result of DE-ELM algorithm in Figure 5(b) is poor, with an accuracy of 87%. The prediction result of the PSO-ELM algorithm in Figure 5(c) is average, with an accuracy of 89.5%. The prediction result of CSA-ELM algorithm in Figure 5(d) is better, with an accuracy of 92.5%. Figure 5(e) The prediction result of the PSO-CSA-ELM algorithm proposed in this paper has the best performance, with an accuracy of 96.5%. It can be seen that the algorithm proposed in this paper has the best classification accuracy of speech feature signals. Figure 6 shows the comparison of the classification accuracy of the five algorithms in the case of different numbers of hidden layer neurons. As the

number of hidden layer neurons increases, the classification results of the five algorithms are gradually increasing. However, the ELM classification effect is the worst, and the performance of the other four algorithms has increased significantly. At the same time, the classification accuracy of the PSO-CSA-ELM algorithm proposed in this paper has the largest increase. Figure 7 shows the comparison of the classification accuracy of the five algorithms under different experimental times. It can be seen that the algorithm proposed in this paper has the best classification accuracy of speech feature signals.

VII. CONCLUSION

In this paper, we thoroughly analyze the performance and related defects of the extreme learning machine, and use an improved crow search algorithm to solve them. By introducing the behavior of following the current optimal solution in the particle swarm optimization algorithm, the shortcomings of the crow search algorithm are slow and easy to fall into the local optimal solution. In this way, the weight and threshold of the ELM can be quickly calculated, and the ELM is optimized to have better calculation speed and accuracy. A search strategy of particle swarm algorithm is proposed to enhance the global search ability, and Gaussian function is added in the later stage of algorithm iteration. Use the penalty coefficient of the function to perform local disturbance, gradually reduce the amplitude of the search trajectory, and then adjust the parameters adaptively to avoid being attracted by the local extremum, and further improve its generalization ability. Finally, the improved crow algorithm is used to optimize the hidden layer neurons and connection weights of the extreme learning machine neural network, so as to obtain accurate prediction results. The proposed PSO-CSA-ELM algorithm has been verified on classification, regression data set and speech signal recognition, and the results show that the accuracy of the improved algorithm has been correspondingly improved.

The algorithm proposed in this paper loses a certain amount of group diversity because the group tends to move in the direction of the local optimal solution every time the algorithm is updated, and the stability of the algorithm is not improved accordingly. In the future research process, we should try to preserve the diversity of the group in the iterative process and make it the focus of research work.

REFERENCES AND FOOTNOTES

- [1]I. Ahmad, M. Basher, M. J. Iqbal and A. Rahim, "Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection," *IEEE Access*, vol. 6, pp. 33789-33795, 2018.
- [2]Y. Yue, P. He, "A comprehensive survey on the reliability of mobile wireless sensor networks: Taxonomy, challenges, and future directions," *Information Fusion*, vol. 44, no.11, pp. 188-204, 2018.

- [3] Y. Zhang , Y. Wang , G. Zhou, "Multi-kernel extreme learning machine for EEG classification in brain-computer interfaces," *Expert Systems with Applications*, vol. 96, no.4, pp. 302-310, 2018.
- [4] W. Feifan , Z. Baihai, C. Senchun, "An Extreme Learning Machine-Based Community Detection Algorithm in Complex Networks," *Complexity*, vol. 2018, no.8, pp. 1-10, 2018.
- [5] Z. Zhou, R. Zhang, J. Zhang, "Fabric wrinkle level classification via online sequential extreme learning machine based on improved sine cosine algorithm," *Textile Research Journal*, vol. 90, no.2, pp. 2007-2021, 2020.
- [6] Z. Zhou, C. Wang, J. Zhang, "Color difference classification of solid color printing and dyeing products based on optimization of the extreme learning machine of the improved whale optimization algorithm," *Textile Research Journal*, vol. 20, no.2, pp. 135-155, 2018.
- [7] J. Li, W. Shi, D. Yang, "Color difference classification of dyed fabrics via a kernel extreme learning machine based on an improved grasshopper optimization algorithm," *Color Research & Application*, vol. 10, no.2, pp. 1-14, 2020.
- [8] J. Li, W. Shi, D. Yang, "Fabric wrinkle evaluation model with regularized extreme learning machine based on improved Harris Hawks optimization," *The Journal of The Textile Institute*, 2021,1(1): 1-16. vol. 1, no.1, pp. 1-16, 2021.
- [9] W. Cao, X. Wang, Z. Ming, "A review on neural networks with random weights," *Neurocomputing*, vol. 275, no.1, pp. 278-287, 2018.
- [10] W. Cao, M. Patwary, P. Yang, "An initial study on the relationship between meta features of dataset and the initialization of nnrw," 2019 International Joint Conference on Neural Networks (IJCNN). IEEE, 2019: 1-8.
- [11] R. M. Adnan , Z. Liang , S. Trajkovic, "Daily streamflow prediction using optimally pruned extreme learning machine," *Journal of Hydrology*, vol. 577, no.10, pp. 1-15, 2019.
- [12] L. Yao and Z. Ge, "Deep Learning of Semisupervised Process Data With Hierarchical Extreme Learning Machine and Soft Sensor Application," in *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1490-1498, Feb. 2018.
- [13] X. Luo , "Short-Term Wind Speed Forecasting via Stacked Extreme Learning Machine With Generalized Correntropy," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4963-4971, Nov. 2018.
- [14] M. A. Mosa , A. S. Anwar , A. Hamouda, "A survey of multiple types of text summarization with their satellite contents based on swarm intelligence optimization algorithms," *Knowledge Based Systems*, vol. 163, no. 1, pp. 518-532, Nov. 2019.
- [15] E. Chen, J. Chen, A. W. Mohamed, B. Wang, Z. Wang and Y. Chen, "Swarm Intelligence Application to UAV Aided IoT Data Acquisition Deployment Optimization," *IEEE Access*, vol. 8, pp. 175660-175668, 2020.
- [16] T. A. Jumani, M. W. Mustafa, A. S. Alghamdi, M. M. Rasid, A. Alamgir and A. B. Awan, "Swarm Intelligence-Based Optimization Techniques for Dynamic Response and Power Quality Enhancement of AC Microgrids: A Comprehensive Review," *IEEE Access*, vol. 8, pp. 75986-76001, 2020.
- [17] A. Slowik and H. Kwasnicka, "Nature Inspired Methods and Their Industry Applications — Swarm Intelligence Algorithms," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 3, pp. 1004-1015, 2018.
- [18] L. Cao, Y. Cai and Y. Yue, "Swarm Intelligence-Based Performance Optimization for Mobile Wireless Sensor Networks: Survey, Challenges, and Future Directions," *IEEE Access*, vol. 7, pp. 161524-161553, 2019.
- [19] Y. Xu , M. Zhang, L. Ye, "A novel prediction intervals method integrating an error & self-feedback extreme learning machine with particle swarm optimization for energy consumption robust prediction," *Energy*, vol. 164, no. 1, pp. 137-146, 2018.
- [20] Y. Li , Y. Ren, H. Zhang, "Selection ELM Parameters Based on Evolution of Modified Differential," *International Journal of Grid and Distributed Computing*, vol. 10, no. 1, pp. 233-244, 2017.
- [21] A. Alharbi, M. Alghahtani, "Using Genetic Algorithm and ELM Neural Networks for Feature Extraction and Classification of Type 2-Diabetes Mellitus," *Applied Artificial Intelligence*, vol. 33, no. 4, pp. 311-328, 2018.
- [22] A. S. Silitonga , A. H. Shamsuddin , T. M. I. Mahlia, " Biodiesel synthesis from Ceiba pentandra oil by microwave irradiation-assisted transesterification: ELM modeling and optimization," *Renewable Energy*, vol. 146, no. 2, pp. 1278-1291, 2020.
- [23] Z. Tian , Y. Ren , G. Wang, "Short-term wind speed prediction based on improved PSO algorithm optimized EM-ELM," *Energy Sources*, vol. 41, no. 6, pp. 26-46, 2019.
- [24] A. Abdoos, "A new intelligent method based on combination of VMD and ELM for short term wind power forecasting," *Neurocomputing*, vol. 203, no. 8, pp. 111-120, 2016.
- [25] X. Tang , L. Chen, "Artificial bee colony optimization-based weighted extreme learning machine for imbalanced data learning," *Cluster Computing*, vol. 310, no. 22, pp. 6937-6952, 2019.
- [26] C. Xiao, Z. Dong, Y. Xu, "Rational and self-adaptive evolutionary extreme learning machine for electricity price forecast," *Memetic Computing*, vol. 471, no. 8, pp. 223-233, 2016.
- [27] X. Jingwen , Z. Yunxuan , Y. Ziqi, "Retraction: Early bearing fault diagnosis based on improved SFLA and ELM network," *Transactions of the Canadian Society for Mechanical Engineering*, vol. 42, no. 2, pp. 187-193, 2018.
- [28] L. L. Li , J. Sun , M. L. Tseng, "Extreme learning machine optimized by whale optimization algorithm using insulated gate bipolar transistor module aging degree evaluation," *Expert Systems with Application*, vol. 127, no. 8, pp. 58-67, 2019.
- [29] C. Qu and Y. Fu, "Crow Search Algorithm Based on Neighborhood Search of Non-Inferior Solution Set," *IEEE Access*, vol. 7, pp. 52871-52895, 2019.
- [30] M. Ahmad, M. Abdullah, H. Moon, S. J. Yoo and D. Han, "Image Classification Based on Automatic Neural Architecture Search Using Binary Crow Search Algorithm," *IEEE Access*, vol. 8, pp. 189891-189912, 2020.
- [31] X. Han, Q. Xu, L. Yue, Y. Dong, G. Xie and X. Xu, "An Improved Crow Search Algorithm Based on Spiral Search Mechanism for Solving Numerical and Engineering Optimization Problems," *IEEE Access*, vol. 8, pp. 92363-92382, 2020.
- [32] A. Omar, H. M. Hasanien, M. A. Elgendy and M. A. L. Badr, "Identification of the photovoltaic model parameters using the crow search algorithm," *The Journal of Engineering*, vol. 2017, no. 13, pp. 1570-1575, 2017.
- [33] R. Sindhuja and A. R. Shankar, "Effective PAPR reduction in SCFDM-based massive MIMO system using binary crow search algorithm for visible light communication towards 5G networks," *IET Communications*, vol. 14, no. 16, pp. 2780-2785, 2020.