Filomat 34:15 (2020), 4937–4952
https://doi.org/10.2298/FIL2015937W

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

Beetle Antennae Search for Neural Network Model With Application to Population Prediction: An Intelligent Optimization Algorithm

Qing Wu[a], Jie Wang[a], Gang Xu[a], Shuai Li[b], Dechao Chen[a]

[a] *School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, P.R. China*
[b] *Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China*

Abstract. Traditional back-propagation (BP) neural networks can implement complex nonlinear mapping relationships, and solve internal mechanism problems. However, as number of samples increases, training BP neural networks may consume a lot of time. For this reason, to improve the efficiency as well as prediction accuracy of the neural network model, in this paper, we propose an intelligent optimization algorithm, by leveraging the beetle antennae search (BAS) strategy to optimize the weights of neural network model, and apply it to the population prediction. A series of experiments demonstrate the improved accuracy of the proposed algorithm over BP neural networks. In particular, the calculation time spent of neural network model via the proposed algorithm is only 20% of the one of BP neural network model. Finally, we present a reasonable trend of population growth in China, and analyze the causes of changes in population trends, which may provide an effective basis for the department to adjust population development strategies.

1. Introduction

In 2010, when Chinese GDP surpassed Japanese, it officially became the second largest economy in the world [1], and the corresponding per capita real GDP growth rate was over 5% per year. China's development is attracting more and more attention, especially for the population [2]. Population has a significant impact on a country, which is embodied in the ecological environment [3, 4], labor employment [5], economic development [6, 7], and so on. However, China faces enormous challenges in the aging of its population. To maintain the country's sustainable development, the government urgently needs to adjust its population development strategy. Hence, population must be accurately predicted to ensure the scientific strategy.

There are many factors that affect population growth, such as economy, culture, and disease [8]. Hence, predicting population is highly challenging [9]. However, many researchers have developed different methods to carry out population predictions, and have made some progress [10–15]. For instance, statistical

2010 *Mathematics Subject Classification*. Primary 78M32; Secondary 92B20, 62M45

Keywords. Population prediction, Neural networks, Beetle antennae search (BAS), Back-propagation (BP), Intelligent optimization algorithm

Received: 06 September 2018; Revised: 19 November 2018; Accepted: 22 March 2019

Communicated by Shuai Li

This work is supported by the National Natural Science Foundation of China (with numbers 61906054, 61401385, 61702146, 61772163 and 61761136010), by Hong Kong Research Grants Council Early Career Scheme (with number 25214015), by Departmental General Research Fund of Hong Kong Polytechnic University (with number G.61.37.UA7L), and also by PolyU Central Research Grant (with number G-YBMU).

Email addresses: wuqing@hdu.edu.cn (Qing Wu), dandyjiew@gmail.com (Jie Wang), gxu@hdu.edu.cn (Gang Xu), shuaili@polyu.edu.hk (Shuai Li), chdchao@hdu.edu.cn (Dechao Chen)

method proposed by Shang *et al.* [10] was a multilevel functional data method, in which mortality and migration were jointly modeled and predicted for females and males. Hui [11] considered the temporal trends of population from the spatial structure of distribution, based on the combination of distance dynamics, occupancy scaling, and spatial autocorrelation domain models, proposed a model that predicts population trends only from their spatial distribution. Raftery *et al.* [12] proposed a novel Bayesian method based on probabilistic population prediction that could serve all countries. By introducing the reproductive health or fertility factor ("Allee's effect") into the Verhulst demographic dynamic logistic model, Miranda *et al.* [13] developed a new analytical procedure as a new method for dealing with a common evolutionary growth process, applied to time evolution and prediction. Zhang *et al.* proposed a forecast method named multiple sine functions decomposition [16], and used it to predict the potential saturation of the world's population [14]. Standard cohort-component [2, 15] was the most classic method of population prediction. United Nations publishes the population prediction every two years in a report called the *World Population Prospects* (WPP) [2], and it is widely used by researchers, international organizations, and governments. In the latest report WPP (2017 version), population forecasts were made for 233 countries and regions in the world, providing a prospect for population trends for countries.

Artificial neural networks (ANNs) [17–20], which is a mathematical model simulating human brain structure, and external stimulus response mechanisms based on network topology knowledge. It is composed of a large number of nodes connected to each other, and the connection between each node represents a weighted value for the signal passing through the connection, which is called weight. Accordingly, ANNs can fully approximate any complex nonlinear relationship, and it has the ability to solve the issues in the real-world [21–26], so it has attracted the attention of many researchers [27–31]. For instance, Xiang *et al.* [32] improved Z-type neural network for computing real-time-dependent matrix pseudoinverse under noisy environments. To compute the Drazin inverse of a real matrix, Stanimirović *et al.* [33, 34] proposed a recurrent neural network (RNN) consisting of n independent parts (subnetworks), it has the computational advantages over the existing algorithms. Li *et al.* [35] proposed two improved neural network models to overcome the two limitations of the existing RNN solutions for manipulator control. Liao *et al.* [36] proposed a new Taylor-type numerical differentiation formula, which is used to discretize continuous-time Zhang neural network. Zhang *et al.* [37] introduced a weights and structure-determination neural network model activated by Chebyshev polynomials of class 1 for data fitting, and apply it to Asian population prediction. The above studies shows that neural networks, through training and testing, can find the approximate correct solution to the problem [38–40]. Training algorithm used by ANNs is conventionally a back-propagation (BP) [41, 42], and the weights between the layers are gradually adjusted by the training error. The algorithm based on this adjustment process is generally the gradient descent [43]. Recently, Fahad *et al.* [44] proposed gradient and divergence estimation methods to achieve a control law for tracking dynamic concentration level curves only by concentration measurement.

The research of meta-heuristic algorithm is an interesting topic in recent years [45], it is generally inspired by the regularity of natural biology, through the modelling of biological and natural intelligence, to solve increasingly complex problems, with the characteristics of easy implementation, high precision, fast convergence and so on. Eberhart *et al.* [46] proposed a particle swarm optimization algorithm, which originated from the enlightenment of bird predation, and then used group intelligence to continuously search for optimal solutions in the solution space. Inspired by the hunting behavior of humpback whales, Mirjalili *et al.* [47] proposed a whale optimization algorithm and applied it to other fields to solve optimization problems in engineering. Rao *et al.* [48] proposed an algorithm called teaching-learning-based optimization, which can obtain the global solution of nonlinear functions with less computational complexity. Different from the previous optimization algorithms, a new optimization algorithm, beetle antennae search (BAS) [49, 50], only needs one individual to solve the optimization problems, so its calculation speed is very fast. Nowadays, the trend of intelligent algorithms and neural networks to solve engineering problems is increasing. Mohamad *et al.* [51] study investigated the application of hybrid neural network model based on particle swarm optimization in the prediction of rock unconfined compressive strength. Momeni *et al.* [52] suggest that ANN model based on genetic algorithm has advantages as a highly-reliable, efficient and practical tool for predicting the pile bearing capacity. Kankal *et al.* [53] studies the performance of an ANN with teaching-learning-based optimization for modeling electric energy demand in Turkey. However, the

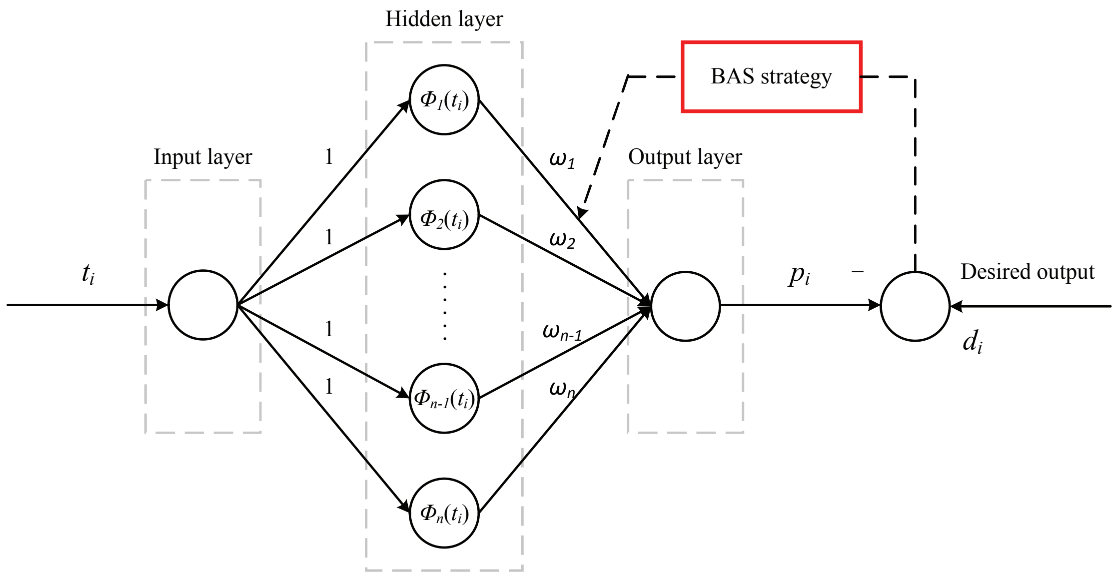


Figure 1: Three-layer neural network model using the proposed intelligent optimization algorithm.

performance impact of the BAS strategy on neural networks has not been reported in the existing literatures. To overcome disadvantages of above mentioned BP neural networks, we propose a novel intelligent optimization algorithm by using the BAS strategy to improve the training performance of the neural network model. The proposed algorithm intelligently and automatically optimizes the weights during neural network training process to improve the prediction accuracy as well as time efficiency.

A summary of our contributions follows.

- A novel intelligent optimization algorithm is proposed by leveraging the BAS strategy to automatically adjust the weights of the neural network model, achieves higher prediction accuracy and computational speed.
- The effectiveness of BAS strategy to find the optimal solution of the objective function are illustrated via two numerical examples.
- The associated neural network model equipped with the proposed algorithm can be applied to the population prediction. Via Chinese population prediction, the associated model is compared with the BP neural network, and the prediction results are analyzed.

The above mentioned advantages are the motivation to use the associated neural network model to predict Chinese population, and provide an effective basis for the population development strategies by the government. The rest of this paper is organized as follows. Section 2 introduces the neural network model used to predict Chinese population. Section 3 presents the proposed intelligent optimization algorithm to optimize the weights of neural network model. In Section 4, the BAS strategy is used to find the optimal solution for two test functions, which shows the effectiveness and versatility of the strategy. In Section 5, the associated neural network model equipped with the proposed algorithm is compared with the conventional BP neural network in performance, and the comprehensive prediction results of Chinese population are presented and discussed. Section 6 discusses the conclusions and future research directions.

2. Preliminaries

Due to the unique nonlinear adaptive information processing capability to overcome the defects of traditional artificial intelligence information processing of unstructured, ANNs have attracted considerable attention [54–57]. The feed-forward neural networks are the representatives. It has a simple structure, can approximate arbitrary continuous functions and square integrable functions with arbitrary precision, and can accurately implement any finite training sample set. It is worth noting that the activation function of the traditional BP neural networks generally adopts a sigmoid function or a linear function. Under certain application scenarios, such an activation function does not apply. Later studies showed that the application of linearly independent and orthogonal activation functions in feed-forward neural networks is better than the sigmoid function [58–61]. Therefore, this paper uses power function as the activation function for the three-layer neural network model.

On the premise of guaranteeing the performance of the neural network, we simplified its structure so as to be able to reduce the computational complexity and facilitate us to do research and analysis. In this paper, a three-layer feed-forward neural network (input layer, hidden layer and output layer) is constructed for the population of China from the year 2 to 2017. The structure is shown in Figure 1. In fact, an ANNs with a hidden layer reduces the complexity of the model and reduces the likelihood of the model being overwritten, and it has been shown that any complex problem can be approximated using a hidden layer ANNs model. In this model, t_i represents the ith input of the neural network, p_i is the ith output of the neural network, and d_i is the expected output corresponding to the ith input data. Obviously, $\{(t_i, d_i)|_{i=1}^{J}\}$ is the data set of the sample pair. We define the input vector $\mathbf{t}$, the output vector $\mathbf{p}$ and the desired output vector $\mathbf{d}$ respectively as follows:

$$\mathbf{t} = [t_1, t_2, \cdots, t_J]^{\mathrm{T}} \in \mathbb{R}^J, \tag{1}$$
$$\mathbf{p} = [p_1, p_2, \cdots, p_J]^{\mathrm{T}} \in \mathbb{R}^J, \tag{2}$$
$$\mathbf{d} = [d_1, d_2, \cdots, d_J]^{\mathrm{T}} \in \mathbb{R}^J, \tag{3}$$

where $\mathbb{R}^J$ represents a collection of real numbers. Activation matrix $\mathbf{H}$ can be expressed as

$$\mathbf{H} = \begin{bmatrix} \Phi_1(t_1) & \Phi_2(t_1) & \cdots & \Phi_K(t_1) \\ \Phi_1(t_2) & \Phi_2(t_2) & \cdots & \Phi_K(t_2) \\ \cdots & \cdots & \ddots & \vdots \\ \Phi_1(t_J) & \Phi_2(t_J) & \cdots & \Phi_K(t_J) \end{bmatrix} \in \mathbb{R}^{J\times K}, \tag{4}$$

where $\Phi_n(t_i)$ indicates that nth neuron in the hidden layer is activated by a set of power functions, which is expressed in the form of

$$\Phi_n(t_i) = t_i^{n-1}, \tag{5}$$

where $n = 1, 2, \ldots, K$, and $i = 1, 2, \ldots, J$. Each input and output layer has only one neuron activated by a linear marker function. It should be noted that in order to simplify the model structure, the weight between the input layer neurons and the hidden layer neurons is fixed to 1, and all offset values in the neural network are set to 0. Then, the relationship between input t_i and output p_i can be established as

$$p_i = \sum_{n=1}^{K} \omega_n \Phi_n(t_i), \tag{6}$$

where ω_n represents the connection weight between the nth hidden layer neuron and the output layer. We define the following root-mean-square error (RMSE) as a performance indicator:

$$\begin{aligned}
\mathrm{RMSE}(\omega) &= \sqrt{\frac{1}{J}\sum_{i=1}^{J}(P_{\mathrm{mea}} - P_{\mathrm{for}})^2} \\
&= \sqrt{\frac{\sum_{i=1}^{J}(d_i - p_i)^2}{J}} \\
&= \sqrt{\frac{\sum_{i=1}^{J}(d_i - \sum_{n=1}^{K}\omega_n \Phi_n(t_i))^2}{J}} \\
&= \frac{\|\mathbf{d} - \mathbf{H}\omega\|_2}{\sqrt{J}},
\end{aligned} \tag{7}$$

where P_{mea} represents the actual observation value and P_{for} represents the output value of the model.

Algorithm 2.1 (BAS strategy for weight updating of neural network model). ***Input:*** *Source* $\mathbf{x}$ *and maximum budget for weight search* K_{max}*;*

Output: *Weight* $\mathbf{U}_{bes}$ *and minimum error value* f_{bes}*;*

Initialize s, μ*;*
Read input data $\mathbf{x}$*;*
Input data normalization;
Beetle initial search direction $\leftarrow$ *Initialize weights* $\mathbf{U}$*;*
$f_{bes} \leftarrow f(\mathbf{U})$ *via equation (7);*
for $k = 1$ *to* K_{max} ***do***
    Calculate the sensing length;
    $\mathbf{U}_{lef}$, $\mathbf{U}_{rig} \leftarrow$ *Search in variable space;*
    Update weights $\mathbf{U} \leftarrow$ *Update the state variable;*
    if $f(\mathbf{U}) < f_{bes}$ ***then***
        $\mathbf{U}_{bes} \leftarrow \mathbf{U}$*;*
        $f_{bes} \leftarrow f(\mathbf{U})$*;*
    end if
    Update s *via equation (13);*
end for
Return $(\mathbf{U}_{bes}, f_{bes})$

3. Methodology

A new intelligent optimization algorithm, BAS, has been developed recently. Different from the previous swarm intelligence algorithms, BAS only needs one individual, so it can greatly reduce the calculation time, which gives us the motivation to update the weight of neural network model with BAS strategy, so as to achieve a more efficient and accurate results than the traditional methods.

3.1. BAS strategy

Similar to the previous intelligent optimization methods, the BAS strategy does not need to know the specific form of the function, and any gradient information. Hence, BAS strategy can realize high speed function optimization. Compared to the swarm intelligence optimization algorithm, BAS only needs one individual, so its computation time will be greatly reduced.

The biological principle of the beetle antennae search strategy is: when the beetle go out for food, it doesn't know the exact location of the food, it will rely on two antennae to judge. When the left antennae has a large concentration of odors, it will fly to the left. When the right antennae receive a large odor concentration, it will fly to the right. We can imagine the smell of the food as a function, each point of the function value is equivalent to the smell of the food concentration. Beetle is collecting the function value of two points near it. If it is to find the minimum value, it will fly to the point where the function value is small; if the maximum, it will fly to the point where the function value is large. By constantly searching for the global maximum and minimum smell, we can find the global optimal point of the function and achieve efficient function optimization.

Considering beetle search behavior, a random search direction of beetle as follows

$$\mathbf{a} = \frac{\mathrm{rnd}(b, 1)}{\|\mathrm{rnd}(b, 1)\|}, \tag{8}$$

where $\mathrm{rnd}(\cdot)$ represents a random function, and b represents the dimension of the search space. The sensing length of the antennae d_t indicates the exploit ability, showing as follows

$$d_t = \frac{s_t}{c}, \tag{9}$$

where s_t denotes the step length, and c denotes the constant. Based on the direction of the antennae, the search behavior of the left and right antennae can be simulated as follows:

$$\begin{aligned} \mathbf{x}_l &= \mathbf{x}_t - d_t\mathbf{a}, \\ \mathbf{x}_r &= \mathbf{x}_t + d_t\mathbf{a}, \end{aligned} \tag{10}$$

where $\mathbf{x}_l$ denotes the position in the left-hand search area at time t, and $\mathbf{x}_r$ denotes that of the right-hand side. Finally, based on the function value of the left and right search area, the position of beetle at time t can be determined as follows:

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \mathbf{a}\mathrm{sign}(f(x_r) - f(x_l))s_t, \tag{11}$$

where $\mathrm{sign}(\cdot)$ is a sign function, $f(\cdot)$ denotes the fitness function, and x_t represents the position of beetle in the search area at time t.

3.2. *Weights updating*

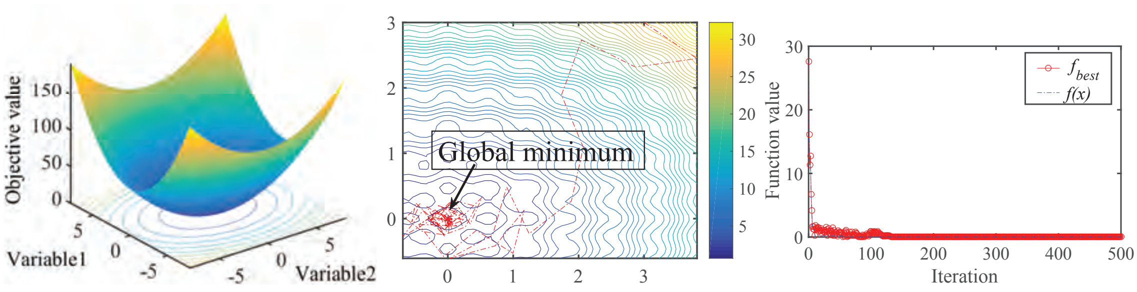
In the beginning, the input data is normalized and specific process can be seen in subsection (5.1). For the initial weight setting, we utilize the following method:

$$\mathbf{U} = \frac{2 * \mathbf{H}^{\mathrm{T}}}{\mathrm{trace}(\mathbf{H}^{\mathrm{T}}\mathbf{H})}, \tag{12}$$

where $\mathbf{H}$ represents the activation matrix equation (4), $\mathrm{trace}(\cdot)$ is a function of calculation sum of diagonal elements, and $\mathbf{U}$ takes the first column in the result.

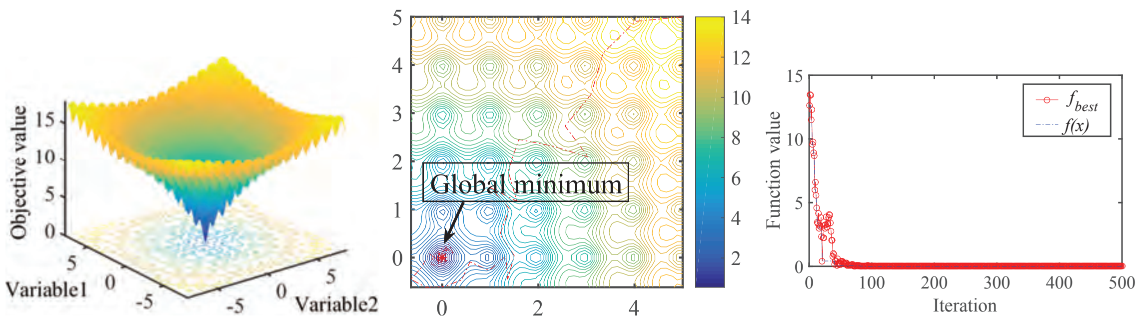
Input the data into the neural network model, process it through the network weights and activation functions, and obtain the output values. Compare the desired output d_i with the resulting output p_i to calculate the error, and save the result in the variable f_{bes}. The dimension of beetle search space is determined by the number of hidden layer neurons (i.e., dimension of weight vector). Determine the search direction of the beetle antennae, and then determine the position in the left and right search areas, $\mathbf{U}_{\mathrm{lef}}$ and $\mathbf{U}_{\mathrm{rig}}$, respectively. Calculate their corresponding values of the fitness function. It's worth noting that the fitness function is equation (7). Update the weight $\mathbf{U}$ and calculate the current error value. Compare local and global best values, if the local best is better than the global best, then the global best alternative is local best. Otherwise, keep the global best unchanged. The step factor can be used to control the ability of beetle area search, and initial step size should be as large as possible, which can cover the current search area and avoid falling into a local minimum. Step update can adopt a linear decrement strategy to ensure the refinement of search, i.e.,

$$s_{t+1} = \mu s_t, \tag{13}$$



(a) Bohachevsky model (b) Optimal path (c) Minimum change

Figure 2: The BAS strategy is used to find the optimal value of Bohachevsky function.



(a) Ackley model (b) Optimal path (c) Minimum change

Figure 3: The BAS strategy is used to find the optimal value of Ackley function.

where μ is the attenuation factor, generally can take the value between $[0, 1]$. Finally, determine whether the fitness function value converges or iterates to the maximum number of times. If the condition is satisfied, the program stop; otherwise, the program continues to run. When the algorithm stops iterating, it returns U_{bes} and f_{bes}, which is the optimal weight and minimum error value of the neural network model. For more details of the algorithm process, refer to Algorithm 2.1.

For the convenience of the reader, we also briefly introduce the BP neural network, which is often used to solve prediction problems. In the process of learning, the data is input to the model, the appropriate output value is obtained through the model processing. Compare the output value with the expected value, propagate the resulting error back to the network, adjust the weights in the network model, and repeat the process until the error reaches the threshold. This whole learning process is the BP algorithm. In mathematical expression, weight update formula is expressed as

$$\omega(k+1) = \omega(k) + \Delta\omega(k), \tag{14}$$

where

$$\Delta\omega(k) = -\eta \frac{\partial E(\omega)}{\partial w}\bigg|_{\omega=\omega(k)} \tag{15}$$

In addition, the weight updates formula of this paper can be changed to

$$\omega(k+1) = \omega(k) - \eta \mathbf{H}^{\mathrm{T}}(\mathbf{H}\omega(k) - \mathbf{d}), \tag{16}$$

where $\eta = 0.5/\mathrm{trace}(\mathbf{H}^{\mathrm{T}}\mathbf{H})$.

Table 1: Training error obtained by training the data before a specified year using different hidden-neuron number.

Hidden-neuron number	30	40	50	53	65	80
2010	0.0518	0.0503	0.0390	0.0358	0.0303	0.0299
2011	0.0532	0.0508	0.0430	0.0383	0.0301	0.0261
2012	0.0547	0.0504	0.0395	0.0382	0.0341	0.0283
2013	0.0559	0.0499	0.0418	0.0420	0.0305	0.0313
2014	0.0557	0.0515	0.0416	0.0379	0.0332	0.0295
2015	0.0570	0.0531	0.0407	0.0337	0.0337	0.0309
Avg	0.0547	0.0510	0.0409	0.0376	0.0319	0.0293

4. Numerical experiments

We briefly describe two classic test functions, i.e., Bohachevsky function, and Ackley function. Then, we try to use BAS strategy to find the optimal solution of these two functions, which indicated that BAS has the ability of optimization, and that is one of the motivations for us to use it to optimize the weight of the neural network model.

4.1. Test via Bohachevsky function

The Bohachevsky functions [62] is a continuous convex body defined in two-dimensional space. Its shape is similar to a bowl, as shown in Figure 2(a), it obviously has a global minimum.

Now, considering the Bohachevsky function:

$$f(x) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x_2) + 0.7, \tag{17}$$

the function is usually evaluted on the square $x_i \in [-100, 100]$, for all $i = 1, 2$. Its global minimum point is located at $x^* = (0, 0)$, and the corresponding minimum value is $f(x^*) = 0$. The BAS strategy looks for the optimal solution of the function as shown in Figure 2(b). The parameter configuration is that the sensing length is $d_t = 4$, and step size s follows rule (13) with initialized value $s = 0.98$, $\mu = 0.97$. We can see that the BAS strategy can find the global minimum point of function from Figure 2(c). Numerically, the final solution point found by BAS is $x_{\text{bes}} = (-1.2023e\text{-}08, -5.1518e\text{-}09)$, and the corresponding function value $f(x_{\text{bes}}) = 2.8866e\text{-}15$ is approximated by $f(x^*)$.

4.2. Test via Ackley function

The Ackley function [62] is widely used to test optimization algorithms. In its two-dimensional form, as shown in the 3(a), it is characterized in that an almost flat region is modulated by a cosine wave to form a hole or a peak, thereby making the surface undulating. This function has the risk that the optimization algorithm is trapped in many of its local minimum values.

We also consider the Ackley function:

$$f(x) = -a\exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right) + a + \exp(1). \tag{18}$$

Recommended variable values are: $a = 20$, $b = 0.2$ and $c = 2\pi$. The function is usually evaluated on the hypercube $x_i \in [-32.768, 32.768]$, for all $i = 1, \ldots, d$, although it may also be restricted to a smaller domain. The minimized value satisfies $f(x^*) = 0$ locating in $x^* = (0, 0)$ in $i = 2$ dimension. Similarly, the number of iteration steps is also from 0 to 500. Here, the sensing length d_t is set to 3, and the step number updating follows (13), $s = 0.99$, $\mu = 0.95$. Figure 3(b) shows the visualization of the optimal path for function (18), and Figure 3(c) shows the trend of the function value changing with the number of iterations in the optimization process. The solution achieved by the strategy in numerical experiments is $x_{\text{bes}} = (3.8045e\text{-}13, -7.3645e\text{-}13)$, the corresponding function value is $f(x_{\text{bes}}) = 2.3457e\text{-}12$, which is sufficiently close to the global minimum point $f(x^*)$.

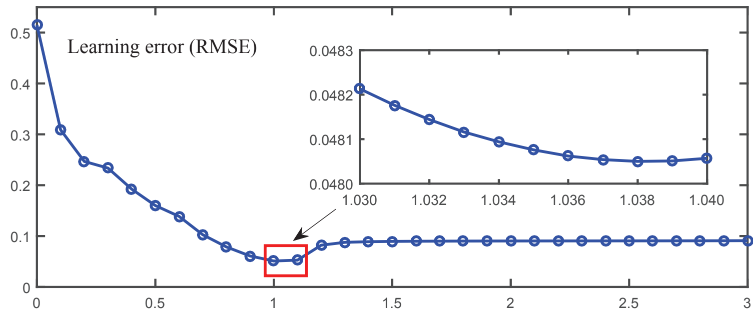


Figure 4: The relationship between projection factor α and learning error.

Table 2: Testing error between the specified year and 2017 using different hidden-neuron number.

Hidden-neuron number	30	40	50	53	65	80
2010	0.1770	0.1360	0.0503	0.0306	0.0320	0.0906
2011	0.1702	0.1273	0.0520	0.0137	0.0313	0.0798
2012	0.1598	0.1128	0.0316	0.0202	0.0239	0.0975
2013	0.1426	0.0995	0.0293	0.0212	0.0289	0.0788
2014	0.1299	0.0969	0.0180	0.0061	0.0179	0.0848
2015	0.1288	0.0890	0.0162	0.0039	0.0238	0.0628
Avg	0.1513	0.1102	0.0329	0.0159	0.0263	0.0823

5. Application to population prediction

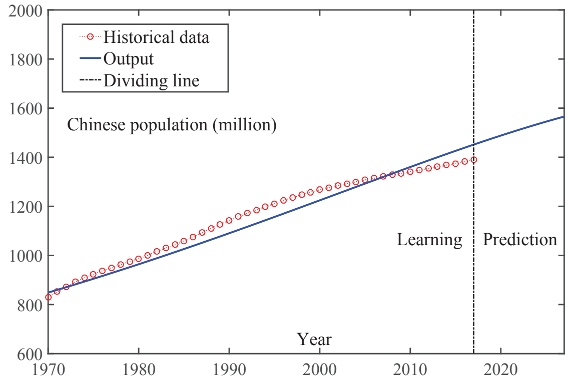
In this section, we apply the associated neural network model equipped with the proposed intelligent optimization algorithm to the Chinese population prediction, and analyze the trends of Chinese future population through numerical results. In addition, comparisons with the conventional BP neural networks shows the feasibility and superiority of the associated neural network model equipped with the proposed algorithm.

5.1. Input data normalization

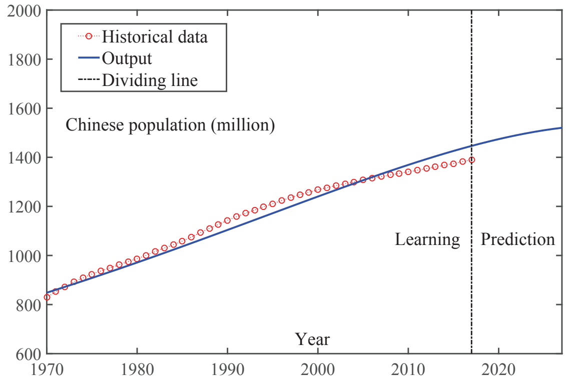
The normalization processing of input data can not only accelerate the calculation speed [63], but also help to find the global optimal solution when evaluating the loss function. Therefore, we project the time interval [2, 2017] onto the projection interval (which can be understood as the "normalized interval") [0, α], where α is the projection factor [37]. In order to further analyze the influence of α on experimental results, corresponding numerical experiments were performed. The results are shown in Figure 4, we can intuitively see the relationship between the projection factor α and the learning error (i.e., RMSE). Different α effects on experimental results exist. In addition, we can see from the map observation that there is a global minimum point, i.e., $\alpha = 1.038$, corresponding to the minimum value of RMSE = 0.0480 (i.e., average relative error of all Chinese population data before 2017). Therefore, all subsequent experiments will use this value to do normalization.

5.2. Population prediction

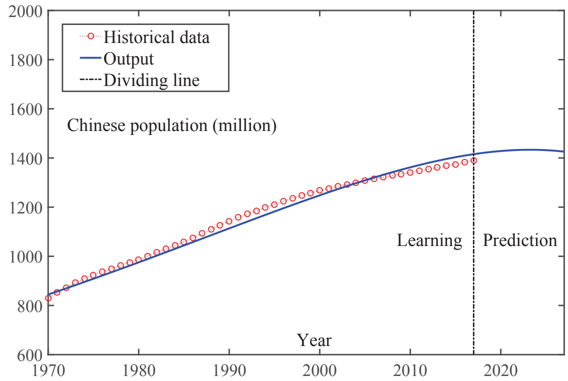
Just like the BP neural networks, the degree of convergence of the neural network model equipped with the proposed algorithm calculation results is also affected by the iteration index k. In general, the iteration index k is set according to the space size of the feasible solution, and the standard of selecting k is to obtain a better convergence effect in a short time. The k chosen in this paper is 1×10^5, 2×10^5, 3×10^5, 4×10^5, 5×10^5, 6×10^5, 7×10^5, 8×10^5, 9×10^5, respectively. In order to select a representative k, we use the proposed neural network model to train all the data before 2017. Because the BAS strategy has randomness (random direction of the antennae), it runs 10 times on the basis of each iteration, such as Figure 6 which shows the trend of the maximum, minimum and average values of the training error in 10 runs. As the iterative index k increases, the training error gradually decreases. When k reaches 9×10^5, the training error converges. In



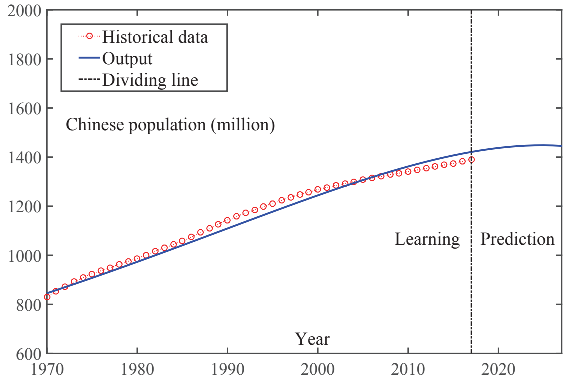
(a) With $k = 1 \times 10^5$



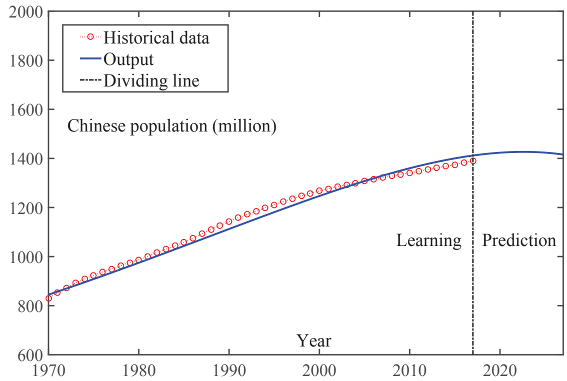
(b) With $k = 2 \times 10^5$



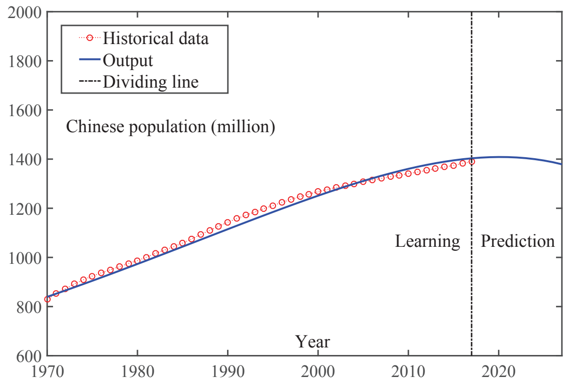
(c) With $k = 4 \times 10^5$



(d) With $k = 5 \times 10^5$



(e) With $k = 8 \times 10^5$



(f) With $k = 9 \times 10^5$

Figure 5: Learning and prediction performance of the propose neural network model for Chinese population prediction with different iteration.

order to better understand the relationship between k value and RMSE, the fitting of the training results can be seen from Figure 5. The number of hidden layer neurons used in the experiment is 53, which will be analyzed later.

We use different numbers of hidden layer neurons in the proposed neural network model to make test errors. The iterative index k used is 1×10^6, run 10 times respectively, and the minimum value of the test error results is taken. Table 1 shows the training errors obtained for training different neurons for a given year. Take the first value 0.0518 as an example: it is the training error for 2010 (i.e., input data to 2010, with

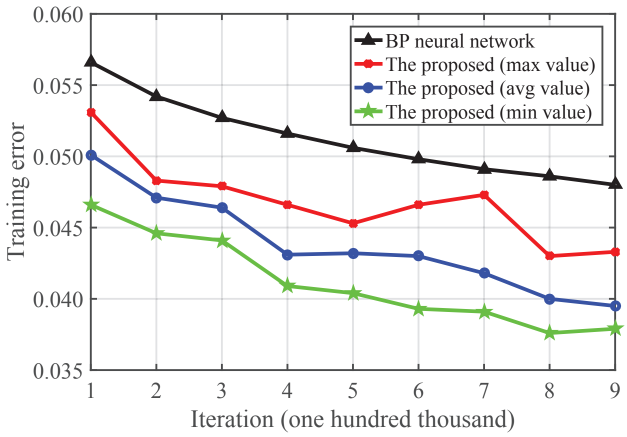


Figure 6: Comparison of training errors by the proposed neural network model and BP neural network.

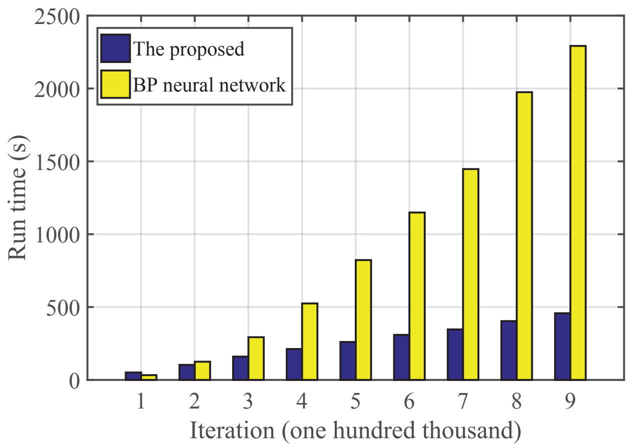


Figure 7: Comparison of run time by the proposed neural network model and BP neural network.

the remaining data as a test). Table 2 shows the test error between the specified year and 2017. Take the first value 0.1770 as an example: it represents the difference between the data from 2011 to 2017 obtained from the model training and the test data. It can be seen from the table that when the number of neurons is 80, although it has the smallest training error, its generalization ability is poor, and the test error is large. When the number of neurons is 30, both training and testing errors are large. When the number of neurons is 53, it has the best effect, not only the training error is small, but also the test error. Therefore, the number of hidden layer neurons in the model was determined to be 53, and all experiments used this value. From the test error results, we can infer that the neural network model equipped with the proposed algorithm has the ability to accurately predict the population.

With the training of proposed neural network, we predict the number of Chinese population in the next 10 years, as shown in Figure 5. From Figure 5(a) and Figure 5(f) shows, the larger k value leads to the smaller training error, which usually means better prediction performance. It can be seen that in the next few years, the Chinese population will continue to grow at a relatively gradual pace because of the large population base in China and the historical trend of continuous growth. However, after 2021, the Chinese population began to slowly decline. In fact, in 1982, family planning was defined as a basic national policy. It mainly advocates late marriage and late childbirth. This policy effectively controlled population growth. The downward trend shown in Figure 5(f) may be affected by this policy.

Table 3: Training error results of the two methods (in %, the lower the values the better). Boldface indicates the best result among these two approaches, performance improvement ratios is shown in parentheses.

Iteration	RMSE		Run time (s)	
	BP	The proposed	BP	The proposed
1×10^5	0.0566	**0.0501 (12.97)**	**32.86**	51.00
2×10^5	0.0542	**0.0471 (15.07)**	125.06	**103.25 (21.12)**
3×10^5	0.0527	**0.0464 (13.57)**	293.18	**160.00 (83.23)**
4×10^5	0.0516	**0.0431 (19.72)**	524.90	**212.10 (147.47)**
5×10^5	0.0506	**0.0432 (17.12)**	822.39	**260.34 (215.89)**
6×10^5	0.0498	**0.0430 (15.81)**	1148.77	**309.14 (271.60)**
7×10^5	0.0491	**0.0418 (17.46)**	1446.89	**346.48 (317.59)**
8×10^5	0.0486	**0.0400 (21.50)**	1974.75	**403.64 (389.23)**
9×10^5	0.0480	**0.0395 (21.51)**	2292.01	**457.76 (400.70)**

Table 4: Comparison of testing error by the proposed neural network model and the BP neural network model.

Neural network	2010	2011	2012	2013	2014	2015
BP neural network	0.1097	0.0979	0.0871	0.0772	0.0682	0.0598
The proposed	**0.0306**	**0.0137**	**0.0202**	**0.0212**	**0.0061**	**0.0039**

5.3. Comprehensive Comparison

BP neural network takes the same parameter settings as the experiment performed by the proposed one. Table 3 shows the comparison of the training error and the run time results of the two models. Here, the RMSE acquisition of the proposed neural network model is an average value. It is not difficult to see that the proposed neural network model has stronger search capability than BP neural network. With the increase in the iteration index, its computation time is much shorter than that of BP neural network, which can be seen from Figure 7. As shown in Figure 6, even if the RMSE of the proposed takes the maximum value, the error rate is less than BP neural network. When the k is 9×10^5, the time spent of proposed neural network model is 20% of the BP neural network model, which benefits from the characteristics of the BAS strategy, the fast convergence speed, the strong optimization ability, and integrate with neural networks to give full play to its advantages. Table 4 shows the comparison of the test error values of the two models. It's obvious that compared with the BP neural network, the proposed neural network model may have a better ability to predict the population and the test error is significantly smaller. As can be seen from Figure 8, the proposed neural network model is similar to WPP, and BP neural network has a higher population prediction result. The numerical results show that the prediction of the neural network model equipped with the proposed intelligent optimization algorithm is reasonable to some extent.

Further, we compare the performance of proposed algorithm with other neural network models, and results are shown in Table 5. The proposed algorithm does not need to find the inverse of the matrix, which reduces computational complexity, and because it is an individual search, it is faster than swarm search. Hence, the proposed algorithm has some advantages, and it is a natural fit to optimize the neural network model.

6. Conclusion and future work

In this paper, a neural network model has combined with meta-heuristic strategy (i.e., the BAS) to construct a novel intelligent optimization algorithm. Through the practical application of predicting the future population of China, the proposed neural network model and the BP neural network have conducted a large number of comparative experiments. It has been found that the proposed neural network model equipped with the novel intelligent optimization algorithm has overcome the shortcomings of BP neural

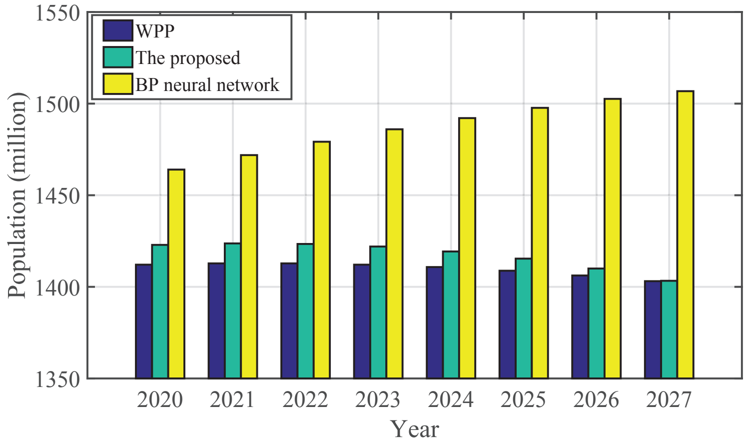


Figure 8: Population prediction comparison of the results via the WPP, the proposed neural network model and BP neural network model.

Table 5: The associated neural network model equipped with the proposed algorithm to compare with other neural network models in performance.

Method/Reference	Performance			
	Speed	Gradient	Inversion	Swarm
The Proposed	**Fast**	**No**	**No**	**No**
[37]	Fast	No	Yes	No
[51]	Slow	No	No	Yes
[52]	Slow	No	No	Yes
[53]	Slow	No	No	Yes
[64]	Fast	No	Yes	No
[65]	Slow	Yes	No	No

networks that slow learning speed and often falling into the local minimum value. Experimental results show that the error rate of the former is smaller than the latter, and that the proposed neural network model can save 80% computation time. This paper also has analyzed the trend of Chinese population in the next 10 years. The results obtained from the the proposed neural network model are that the population will continue to grow slowly over the next few years, but after 2021, there will be a slow decline. Similar to the results of WPP, it is reasonable to use the proposed neural network model to predict. In addition, we also have provided high-quality population data that can support sociological researchers to further explore. It is worth noting that the neural network model equipped with the proposed intelligent optimization algorithm can be easily extended to predictive analysis of population in other countries and regions. As a final remark of this paper, to the best of author's knowledge, this is the first work in the field of neural networks that is able to elegantly address the weight optimization by leveraging the BAS strategy, and effectively apply to population prediction.

Future directions are listed as follows: (1) optimize the structure of neural network model and adaptively find the appropriate number of hidden layer neurons; (2) the proposed intelligent optimization algorithm is made in parallel and runs on a distributed platform; (3) population prediction for other countries or regions.

Appendix A

This paper collects rough data from the past 2016-year (i.e., from 2 to 2017). In this appendix, for convenience of further analysis by researchers and practitioners, these data are presented in Table 6.

Table 6: Chinese population data used in this paper.

Year	2	57	75	88	105	125	140	144
Data (million)	59.59	21.01	34.13	43.36	53.26	48.69	49.15	49.73
Year	145	146	156	280	609	705	726	732
Data (million)	49.52	47.57	56.48	16.16	46.02	37.14	41.42	45.43
Year	740	742	754	755	760	764	820	1006
Data (million)	48.14	48.91	52.88	52.92	16.99	16.90	15.76	16.28
Year	1053	1066	1083	1122	1162	1187	1207	1281
Data (million)	22.29	29.09	24.97	46.73	33.11	44.71	45.82	58.83
Year	1291	1381	1393	1403	1502	1504	1578	1620
Data (million)	59.85	59.87	60.54	66.60	50.91	60.11	60.69	51.66
Year	1724	1741	1751	1764	1776	1790	1834	1844
Data (million)	126.11	143.41	181.81	205.50	208.10	301.49	401.01	419.44
Year	1887	1901	1911	1913	1928	1931	1935	1947
Data (million)	377.64	426.45	341.42	432.00	474.78	421.07	462.15	461.00
Year	1949	1950	1951	1955	1960	1965	1970	1971
Data (million)	541.67	551.96	563.00	614.65	662.07	725.38	829.92	852.29
Year	1972	1973	1911	1913	1928	1931	1935	1947
Data (million)	377.64	426.45	341.42	432.00	474.78	421.07	462.15	461.00
Year	1949	1950	1951	1955	1960	1965	1970	1971
Data (million)	541.67	551.96	563.00	614.65	662.07	725.38	829.92	852.29
Year	1972	1973	1974	1975	1976	1977	1978	1979
Data (million)	871.77	892.11	908.59	924.20	937.17	949.74	962.59	975.42
Year	1980	1981	1982	1983	1984	1985	1986	1987
Data (million)	987.05	1000.72	1016.54	1030.08	1043.57	1058.51	1075.07	1093.00
Year	1988	1989	1990	1991	1992	1993	1994	1995
Data (million)	1110.26	1127.04	1143.33	1158.23	1171.71	1185.17	1198.50	1211.21
Year	1996	1997	1998	1999	2000	2001	2002	2003
Data (million)	1223.89	1236.26	1247.61	1257.86	1267.43	1276.27	1284.53	1292.27
Year	2004	2005	2006	2007	2008	2009	2010	2011
Data (million)	1299.88	1307.56	1314.48	1321.29	1328.02	1334.50	1340.91	1347.35
Year	2012	2013	2014	2015	2016	2017		
Data (million)	1354.04	1360.72	1367.82	1374.62	1382.71	1390.08		

References

[1] X. Jin, D.D. Li, S. Wu, How will China shape the world economy? China Economic Review 40 (2016) 272–280.

[2] United Nations, On-line Database, In: World Population Prospects: The 2017 Revision, Highlights and Advance Tables, United Nations, New York, 2018.

[3] W. Wu, S. Niu, Evolutional analysis of coupling between population and resource-environment in China, Procedia Environmental Sciences 12 (2012) 793–801.

[4] G. Chi, H.C. Ho, Population stress: A spatiotemporal analysis of population change and land development at the county level in the contiguous United States, 2001-2011, Land Use Policy 70 (2018) 128–137.

[5] H. Dadashpoor, M. Alidadi, Towards decentralization: Spatial changes of employment and population in Tehran Metropolitan Region, Iran, Applied Geography 85 (2017) 51–61.

[6] J. Chen, Y. Wang, J. Wen, F. Fang, M. Song, The influences of aging population and economic growth on Chinese rural poverty, Journal of Rural Studies 47 (2016) 665–676.

[7] J. Golley, W. Zheng, Population dynamics and economic growth in China, China Economic Review 35 (2015) 15–32.

[8] J. Lassila, T. Valkonen, J.M. Alho, Demographic forecasts and fiscal policy rules, International Journal of Forecasting 30 (2014) 1098–1109.

[9] J.M. Alho, Forecasting demographic forecasts, International Journal of Forecasting 30 (2014) 1128–1135.

[10] H.L. Shang, P.W.F. Smith, J. Bijak, A. Wiśniowski, A multilevel functional data method for forecasting population, with an application to the United Kingdom, International Journal of Forecasting 32 (2016) 629–649.

[11] C. Hui, Forecasting population trend from the scaling pattern of occupancy, Ecological Modelling 222 (2011) 442–446.

[12] A.E. Raftery, N. Li, H. Sevcikova, P. Gerland, G.K. Heilig, Bayesian probabilistic population projections for all countries, Proceedings of the National Academy of Sciences of the United States of America 109 (2012) 13915–13921.

[13] L.C.M. Miranda, C.A.S. Lima, On the logistic modeling and forecasting of evolutionary processes: Application to human population dynamics, Technological Forecasting and Social Change 77 (2010) 699–711.
[14] Y. Zhang, N. Wang, L. He, W. Li, H. Tan, A Potential Saturation Value of World Population is Near? 2016 IEEE 13th International Conference on Signal Processing (2016) 1781–1786.
[15] P. Gerland, A.E. Raftery, H. Sevcikova, N. Li, D. Gu, T. Spoorenberg, L. Alkema, B.K. Fosdick, J. Chunn, N. Lalic, G. Bay, T. Buettner, G.K. Heilig, J. Wilmoth, World population stabilization unlikely this century, Science 346 (2014) 234–237.
[16] Y. Zhang, J. Ma, B. Qiu, S. Ding, Z. Yang, Warming Substantiated by Multiple Sine Functions Decomposition of Multiple Cities' Temperature Data, 2015 Chinese Automation Congress (2015) 70–74.
[17] X.Z. Wang, H. Ma, P.S. Stanimirović, Nonlinearly Activated Recurrent Neural Network for Computing the Drazin Inverse, Neural Processing Letters 46 (2017) 195–217.
[18] I.S. Živković, P.S. Stanimirović, Y. Wei, Recurrent Neural Network for Computing Outer Inverse, Neural Computation 28 (2016) 970–998.
[19] S. Li, J. He, Y. Li, M.U. Rafique, Distributed Recurrent Neural Networks for Cooperative Control of Manipulators: A Game-Theoretic Perspective, IEEE Transactions on Neural Networks and Learning Systems 28 (2017) 415–426.
[20] Y. Zhang, S. Li, X. Liu, Neural Network-Based Model-Free Adaptive Near-Optimal Tracking Control for a Class of Nonlinear Systems, IEEE Transactions on Neural Networks and Learning Systems (2018).
[21] D. Chen, Y. Zhang, Robust Zeroing Neural-Dynamics and Its Time-Varying Disturbances Suppression Model Applied to Mobile Robot Manipulators, IEEE Transactions on Neural Networks and Learning Systems 29 (9) (2018) 4385–4397.
[22] S. Li, S. Chen, B. Liu, Y. Li, Y. Liang, Decentralized kinematic control of a class of collaborative redundant manipulators via recurrent neural networks, Neurocomputing 91 (2012) 1–10.
[23] L. Jin, S. Li, B. Hu, C. Y, Dynamic neural networks aided distributed cooperative control of manipulators capable of different performance indices, Neurocomputing 291 (2018) 50–58.
[24] L. Jin, S. Li, H. Wang, Z. Zhang, Nonconvex projection activated zeroing neurodynamic models for time-varying matrix pseudoinversion with accelerated finite-time convergence, Applied Soft Computing 62 (2018) 840–850.
[25] D. Chen, S. Li, Q. Wu, A Novel Supertwisting Zeroing Neural Network With Application to Mobile Robot Manipulators, IEEE Transactions on Neural Networks and Learning Systems, to be published, doi: 10.1109/TNNLS.2020.2991088.
[26] M. Fahad, N. Saul, Y. Guo, B. Bingham, Robotic simulation of dynamic plume tracking by Unmanned Surface Vessels, 2015 IEEE International Conference on Robotics and Automation (2015) 2654–2659.
[27] B. Liao, Y. Zhang, Different Complex ZFs Leading to Different Complex ZNN Models for Time-Varying Complex Generalized Inverse Matrices, IEEE Transactions on Neural Networks and Learning Systems 25 (2014) 1621–1631.
[28] D. Chen, Y. Zhang, S. Li, Tracking Control of Robot Manipulators with Unknown Models: A Jacobian-Matrix-Adaption Method, IEEE Transactions on Industrial Informatics 14 (2018) 3044–3053.
[29] D. Chen, Y. Zhang, S. Li, Zeroing neural-dynamics approach and its robust and rapid solution for parallel robot manipulators against superposition of multiple disturbances, Neurocomputing 275 (2018) 845–858.
[30] S. Li, B. Liu, Y. Li, Selective Positive-Negative Feedback Produces the Winner-Take-All Competition in Recurrent Neural Networks, IEEE Transactions on Neural Networks and Learning Systems 24 (2013) 301–309.
[31] D. Chen, S. Li, Q. Wu, Rejecting Chaotic Disturbances Using a Super-Exponential-Zeroing Neurodynamic Approach for Synchronization of Chaotic Sensor Systems, Sensors 19 (1) (2019) 74.
[32] Q. Xiang, B. Liao, L. Xiao, L. Jin, A noise-tolerant Z-type neural network for time-dependent pseudoinverse matrices, Optik 165 (2018) 16–28.
[33] P.S. Stanimirović, I.S. Živković, Y. Wei, Recurrent Neural Network for Computing the Drazin Inverse, IEEE Transactions on Neural Networks and Learning Systems 26 (2015) 2830–2843.
[34] P.S. Stanimirović, I.S. Živković, Y. Wei, Recurrent Neural Network Approach Based on the Integral Representation of the Drazin Inverse, Neural Computation 27 (2015) 2107–2131.
[35] S. Li, Y. Zhang, L. Jin, Kinematic Control of Redundant Manipulators Using Neural Networks, IEEE Transactions on Neural Networks and Learning Systems 28 (2017) 2243–2254.
[36] B. Liao, Y. Zhang, L. Jin, Taylor O(h^3) Discretization of ZNN Models for Dynamic Equality-Constrained Quadratic Programming With Application to Manipulators, IEEE Transactions on Neural Networks and Learning Systems, 27 (2016) 225–237.
[37] Y. Zhang, D. Guo, Z. Luo, K. Zhai, H. Tan, CP-activated WASD neuronet approach to Asian population prediction with abundant experimental verification, Neurocomputing 198 (2016) 48–57.
[38] D. Guo, Y. Zhang, Zhang neural network, Getz-Marsden dynamic system, and discrete-time algorithms for time-varying matrix inversion with application to robots' kinematic control, Neurocomputing 97 (2012) 22–32.
[39] B. Liao, Y. Zhang, From different ZFs to different ZNN models accelerated via Li activation functions to finite-time convergence for time-varying matrix pseudoinversion, Neurocomputing 133 (2014) 512–522.
[40] Q. Xiang, B. Liao, L. Xiao, L. Lin, S. Li, Discrete-time noise-tolerant Zhang neural network for dynamic matrix pseudoinversion, Soft Computing (2018) 1–12.
[41] L. Xu, X. Yan, Z. Li, Development of BP-based seismic behavior optimization of RC and steel frame structures, Engineering Structures 164 (2018) 214–229.
[42] D. Guo, Y. Zhang, Z. Xiao, M. Mao, J. Liu, Common nature of learning between BP-type and Hopfield-type neural networks, Neurocomputing 167 (2015) 578–586.
[43] S. Ruder, An overview of gradient descent optimization algorithms, arXiv:1609.04747 (2016).
[44] M. Fahad, Y. Guo, B. Bingham, K. Krasnosky, L. Fitzpatrick, F.A. Sanabria, Ocean Plume Tracking with Unmanned Surface Vessels: Algorithms and Experiments, arXiv:1804.08669 (2018).
[45] Q. Wu, X. Shen, Y. Jin, Z. Chen, S. Li, A.H. Khan, D. Chen, Intelligent beetle antennae search for UAVs sensing and avoidance of obstacles, Sensors 2019 to be published.

[46] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, Proceedings of the Sixth International Symposium on Micro Machine and Human Science (1995) 39–43.
[47] S. Mirjalili, A. Lewis, The Whale Optimization Algorithm, Advances in Engineering Software 95 (2016) 51–67.
[48] R.V. Rao, V.J. Savsani, D.P. Vakharia, Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems, Computer-Aided Design 43 (2011) 303–315.
[49] X. Jiang, S. Li, BAS: Beetle Antennae Search Algorithm for Optimization Problems, International Journal of Robotics and Control 1 (2018) 1–2.
[50] X. Jiang, S. Li, Beetle Antennae Search without Parameter Tuning (BAS-WPT) for Multi-objective Optimization, arXiv:1711.02395v1.
[51] E.T. Mohamad, D.J. Armaghani, E. Momeni, S.V.A.N.K. Abad, Prediction of the unconfined compressive strength of soft rocks: a PSO-based ANN approach, Bulletin of Engineering Geology and the Environment 74 (2015) 745–757.
[52] E. Momeni, R. Nazir, D.J. Armaghani, H. Maizir, Prediction of pile bearing capacity using a hybrid genetic algorithm-based ANN, Measurement 57 (2014) 122–131.
[53] M. Kankal, E. Uzlu, Neural network approach with teaching-learning-based optimization for modeling and forecasting long-term electric energy demand in Turkey, Neural Computing and Applications 28 (2017) 737–747.
[54] L. Xiao, S. Li, J. Yang, Z. Zhang, A new recurrent neural network with noise-tolerance and finite-time convergence for dynamic quadratic minimization, Neurocomputing 285 (2018) 125–132.
[55] D. Guo, Z. Nie, L. Yan, Theoretical analysis, numerical verification and geometrical representation of new three-step DTZD algorithm for time-varying nonlinear equations solving, Neurocomputing 214 (2016) 516–526.
[56] L. Xiao, B. Liao, A convergence-accelerated Zhang neural network and its solution application to Lyapunov equation, Neurocomputing 193 (2016) 213–218.
[57] L. Jin, S. Li, B. Liao, Z. Zhang, Zeroing neural networks: A survey, Neurocomputing 267 (2017) 597–604.
[58] Y. Zhang, Y. Chen, X. Jiang, Q. Zeng, A. Zou, Weights-directly-determined and Structure-adaptively-tuned Neural Network Based on Chebyshev Basis Functions, Computer science 36 (2009) 210-213.
[59] L. Xiao, A nonlinearly activated neural dynamics and its finite-time solution to time-varying nonlinear equation, Neurocomputing 173 (2016) 1983–1988.
[60] L. Xiao, R. Lu, Finite-time solution to nonlinear equation using recurrent neural dynamics with a specially-constructed activation function, Neurocomputing 151 (2015) 246–251.
[61] L. Jin, S. Li, Nonconvex function activated zeroing neural network models for dynamic quadratic programming subject to equality and inequality constraints, Neurocomputing 267 (2017) 107–113.
[62] M. Jamil, X. Yang, A Literature Survey of Benchmark Functions For Global Optimization Problems, Journal of Mathematical Modelling and Numerical Optimisation 4 (2013) 150–194.
[63] J. Sola, J. Sevilla, Importance of Input Data Normalization for the Application of Neural Networks to Complex Industrial Problems, IEEE Transactions on Nuclear Science 44 (1997) 1464–1468.
[64] Y. Zhang, S. Ding, J. Wen, Y. Ding, M. Mao, Combining WASP and ASF Algorithms to Forecast a Japan Earthquake with Mj 7.2 or Above, 2016 12th World Congress on Intelligent Control and Automation (2016) 2407–2413.
[65] T. Ren, S. Liu, G. Yan, H. Mu, Temperature prediction of the molten salt collector tube using BP neural network, IET Renewable Power Generation 10 (2016) 212–220.