

Article

Application of a Hybrid Optimized BP Network Model to Estimate Water Quality Parameters of Beihai Lake in Beijing

Jianzhuo Yan ^{1,2}, Zongbao Xu ^{1,2,*} , Yongchuan Yu ^{1,2}, Hongxia Xu ^{1,2} and Kaili Gao ^{1,2}

¹ Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China; yanjianzhuo@bjut.edu.cn (J.Y.); yuyongchuan@bjut.edu.cn (Y.Y.); xhxcl@bjut.edu.cn (H.X.); gkl@emails.bjut.edu.cn (K.G.)

² Engineering Research Center of Digital Community, Beijing University of Technology, Beijing 100124, China

* Correspondence: zongbao_xu@emails.bjut.edu.cn; Tel.: +86-185-0111-4195

Received: 16 April 2019; Accepted: 29 April 2019; Published: 7 May 2019



Abstract: Nowadays, freshwater resources are facing numerous crises and pressures, resulting from both artificial and natural process, so it is crucial to predict the water quality for the department of water environment protection. This paper proposes a hybrid optimized algorithm involving a particle swarm optimization (PSO) and genetic algorithm (GA) combined BP neural network that can predict the water quality in time series and has good performance in Beihai Lake in Beijing. The data sets consist of six water quality parameters which include Hydrogen Ion Concentration (pH), Chlorophyll-a (CHLA), Hydrogenated Amine (NH₄H), Dissolved Oxygen (DO), Biochemical Oxygen Demand (BOD), and electrical conductivity (EC). The performance of the model was assessed through the absolute percentage error (APE_{max}), the mean absolute percentage error (MAPE), the root mean square error (RMSE), and the coefficient of determination (R^2). Study results show that the model based on PSO and GA to optimize the BP neural network is able to predict the water quality parameters with reasonable accuracy, suggesting that the model is a valuable tool for lake water quality estimation. The results show that the hybrid optimized BP model has a higher prediction capacity and better robustness of water quality parameters compared with the traditional BP neural network, the PSO-optimized BP neural network, and the GA-optimized BP neural network.

Keywords: water quality prediction; particle swarm optimization; genetic algorithm; BP neural network

1. Introduction

Human activities are considered to cause major pollution of water quality, which calls for urgent action around the world. Some of the main emission sources that can significantly affect surface water quality are discharge of hazardous substances from industrial processes and urban waste water, accidental hazardous substances pollution, and diffuse pollution originating from agricultural areas [1].

Numerous water quality parameters are being measured to indicate lake water status and guide decision makers towards implementing optimal and sustainable measures. Dissolved oxygen (DO) content is one of the most important water quality parameters as it directly indicates the status of aquatic ecosystem and its ability to sustain aquatic life. DO is considered to be the most badly affected among all water quality parameters [1]. So far, many methods have been used to predict water quality: [2] use the grey relational method to predict the water quality and perform water quality evaluation of rivers in china; [3] developed a Bayesian approach to river WQM combined with inverse methods to support the practical adaptive water quality management under uncertainty of Hun-Taiz River in northeastern China; [4] used genetic algorithm (GA) and geographic information system

(GIS) methods to calculate BOD and DO from the WQMM model; Approximately 85–90% of water quality prediction work has been completed using neural networks (NNs) [5]. For example, The ANN modeling technique's application for dynamic seawater quality prediction variation [6] was presented in combination with high-efficiency learning methods (general regression neural networks (GRNN) and multilayer perceptron (MLP)); [5] established a water quality prediction system combined principal component analysis (PCA), genetic algorithm (GA) and back propagation neural network (BPNN), a hybrid intelligent algorithm is designed for river water quality; [7] presented a flexible structure Radial Basis Function neural network (FS-RBFNN) for water quality prediction, and it can change its structure dynamically to maintain prediction accuracy. Also, [8] verified that the developed ANFIS model performs quite well for each data combination for BOD prediction with adequate performance. The learning abilities of an ANN and argumental abilities of fuzzy logic are combined to increase ANFIS prediction abilities [9]. Because of its strong self-learning ability, an artificial neural network can be widely used in the field of time series prediction. Nevertheless, ANN application carries some disadvantages, including the slow rate of learning and getting trapped in local minima [10–12]; therefore, many researchers are committed to the optimization of neural network parameters to solve these problems. Some researchers consider combining particle swarm optimization (PSO) with genetic algorithms to optimize neural networks and this has been applied in the field of nonlinear prediction successfully, such as in industrial manufacture [13], energy demand [14,15], and biological engineering [16].

In this work, we practically predict water quality parameters using the methods of BPNN, PSO-BPNN, GA-BPNN, and PSO-GA-BPNN, and analyze the algorithm theoretically. Based on the observed value, the performance of the models in prediction of lake water quality was evaluated to prove that PSO-GA-BPNN has a better prediction accuracy than others.

2. Materials and Methods

2.1. Study Area and Water Quality Data

Beijing (39°28' N–41°05' N, 115°25' E–117°30' E), the capital of China, is located at a central latitude, belonging to the eastern warm temperate monsoon zone with semi-humid continental climate and four distinct seasons. Human activities and climate change have important effects on water quality. The local government faces serious challenges of pollution control and natural resource management of lakes in Beijing, especially for landscape water and drinking water in lakes and reservoirs. As an issue of national concern, numerous studies in lakes have been paid more attention regarding their physical, chemical (e.g., total nitrogen [17], total phosphorus [18], heavy metals [19], etc.), and biological (e.g., phytoplankton [20], aquatic fish [21] and aquatic plant [22] etc.) parameters, as well as the influences of land use [23] and eutrophication [24]. In this research, we focus on water quality parameters of Beihai Lake. Beihai Lake, connecting Qianhai Lake in the north and Zhonghai Lake and Nanhai Lake in the south, is the largest landscape lake around the Forbidden City. The location of Beihai Lake is shown in Figure 1. Water of Beihai Lake comes from Miyun and Guanting Reservoirs, which are very important drinking water areas in Beijing. The water quality parameters of reclaimed landscape water can reach the national water quality standards Class IV at most, which is the minimum standard for industry and recreation [25].

We used continuous time series water quality monitoring data from the Beijing Water-affair Authority for about 120 h in August 2013. Water quality parameters measured include PH, CHLA (mg/L), NH₄H (mg/L), DO (mg/L), BOD (mg/L), and EC (μs/cm). Basic statistics of the measured water quality variables of Beihai Lake are shown in Table 1 as follows.



Figure 1. Location map of Beihai Lake in Beijing.

Table 1. Basic statistics of the measured water quality variables of Beihai Lake.

Variable	Unit	Minimum	Maximum	Mean	SD	CDO
PH	-	7.2000	8.2000	7.6735	0.2538	0.8818
CHLA	mg/L	0.0176	0.0283	0.0230	0.0021	−0.0839
NH4H	mg/L	0.2900	0.3900	0.3387	0.0154	−0.0266
DO	mg/L	2.6000	6.5000	4.1800	0.8654	1
BOD	mg/L	2.6	5.4	3.8	0.509	0.675
EC	µs/cm	457	474	465.1440	3.6654	−0.1370

SD: standard deviation; CDO: correlation with DO.

2.2. Data Preparation and Input Selection

Because the water quality parameters of the test come from different Internet of Things (IOT) collection devices and different time data intervals, and a large amount of data was input manually as well, the original data is not satisfactory. In addition, water parameters usually have different dimensions and orders of magnitude, and the data can be converted to the same order of magnitude without dimension by a normalized processing method (Equation (2)). At the beginning, the standard deviation was calculated using Equation (1). After the models have been successfully executed, the outputs of the models in the form of normalized values are converted to original values by inverse transformation using Equation (3). The priority is to preprocess the original data. We eliminate invalid and discrete data; the remaining data were normalized and grouped into training samples and test samples. These procedures were done with SPSS.18.

$$SD = \sqrt{\frac{\sum(X_i - \bar{X}_i)^2}{N - 1}} \tag{1}$$

$$X' = \frac{X_{max} - X}{X_{max} - X_{min}} \tag{2}$$

$$X = X' * (X'_{max} - X'_{min}) - X'_{min} \tag{3}$$

where SD = standard deviation of X_i ; X_i = input data; \bar{X} = arithmetic average of X ; X' = values after normalized method and also as input to models; X_{max} = maximum value of X and X_{min} = minimum value of X . After normalization of linear functions, the experimental data values will be mapped to [0,1].

2.3. Back Propagation Neural Network (BPNN)

BP is a multilayer forward feedback neural network, and it is also named error reverse propagation neural network according to the method of error inverse propagation. According to incomplete statistics, 80–90% of the neural network models used by people adopt BP network or some form of change. The core process consists of four parts, forward calculation, feedback calculation of local gradient, weight correction between neurons, and calculation of the total mean squared error (Equation (4)). A sigmoidal (Equation (5)) function is taken as the transfer function

$$E_{AV} = \frac{1}{2N} \sum_{j=1}^N \sum_{j \in c} e_j^2(n) \tag{4}$$

$$f(x) = \frac{1}{1 + e^{-ex}} \tag{5}$$

where, N = sample number, c = collection of all output units.

2.4. Optimizing BPNN Using PSO

Particle swarm optimization (PSO) is inspired by the behavior of bird and fish swarms. It was developed by Eberhart and Kennedy in 1995 [26]. Each member of the population is called a particle, each particle represents a potential feasible solution, and the position of the particle is considered to be the global optimal solution. The population searches for the global optimal solution in the d -dimensional space. Meanwhile, each particle decides its own flight direction through the value of the adaptive function and the velocity, and gradually moves to the better region to finally search for the global optimal solution. Particles in a particle swarm are described by position vectors and velocity vectors, and the possible solutions of particle position vectors correspond to the weight values in the network. The velocity vector and position vector are updated with Equations (6) and (7) respectively. Using PSO to optimize the BP neural network accelerates the convergence speed and reduces the possibility of falling into local extremum.

$$v_{id}(t + 1) = \omega v_{id}(t) + c_1 rand_1^d (gBest_{id}(t) - x_{id}(t)) + c_2 rand_2^d (zBest_d(t) - x_{id}(t)) \tag{6}$$

$$x_{id}(t + 1) = x_{id}(t) + v_{id}(t + 1) \tag{7}$$

where, $v_{id}(t)$ is the d -dimensional flying speed component of particle i when it evolves to generation t ; $x_{id}(t)$ is the d -dimensional position component of particle i when it evolves to generation t ; $gBest_{id}(t)$ is the optimal position $gBest_i$ component of d -dimensional individual in the evolution of particle i to generation t ; $zBest_d(t)$ is the d -dimensional component of $zBest_t$, the optimal position of the whole particle swarm in the evolution to generation t ; ω = inertia weight, $c_{1,2}$ = accelerated constant, $rand_{1,2}^d$ is a random number of [0,1].

It can be seen that inertia weight ω determines the global optimization and local optimization of PSO in Equation (8). The larger the ω value is, the stronger the global optimization ability is. On the contrary, the smaller the ω value is, the stronger the local optimization ability is. Linear decrement is used to adjust the weight for this part in Equation (8).

$$\omega = \omega_{max} - \frac{(\omega_{max} - \omega_{min}) \times t}{G} \tag{8}$$

where, ω_{max} = maximum inertia weight; ω_{min} = minimum inertia weight; t = current iteration number; G_{max} = maximum evolutionary algebra.

The particle swarm optimization algorithm procedures are usually demonstrated as follows.

Step 1: Initialize swarm

According to the problem to be optimized, the particle swarm's velocity v_i , position x_i , population size N , individual extreme value $gBest_i$ and global extreme value $zBest_i$ are initialized.

Step 2: Calculate the particle fitness value

The mean square error (MSE) Equation (9) is selected as the objective function to calculate the fitness value of the initial particle swarm.

Step 3: Update individual extremum $gBest_i$

The individual fitness value calculated by step 2 is compared with the fitness value of the individual extreme value $gBest_i$. If the individual fitness value is good, the current position of the individual will be regarded as the historic optimal position of the individual; that is, the individual extreme value of $gBest_i$. Otherwise, the current individual extreme value $gBest_i$ will be maintained until a better individual appears.

Step 4: Update global extremum $zBest_i$

The fitness values of $gBest_i$ and $zBest_i$ are compared. If the fitness value of $gBest_i$ is better, the optimal position of an individual will be taken as the historical optimal position of the group, namely the global extreme value. Otherwise, the current global extreme value will be maintained until a better individual extreme value appears.

Step 5: Update the particle's speed and position

Update particle speed v_i and position x_i according to particle swarm optimization speed and position (Equations (6) and (7)).

Step 6: End particle swarm optimization algorithm

The particle swarm optimization algorithm is judged by the end condition. According to the set end condition of the algorithm (maximum iteration number or target fitness value), if the condition is not met, then jump to step 2, otherwise, output the global optimal solution $zBest_i$.

2.5. Optimizing BPNN Using GA

The GA is an optimization algorithm that simulates Darwinian evolutionary mechanisms to find the optimal solution. It features adaptability, randomness, and high parallelism. GA obeys the principle of survival of the fittest, repeats the operation of selection, crossover, and mutation with individual fitness as the evaluation standard, eliminates chromosomes with poor adaptability, retains the fitness of individuals, and forms a new population. This algorithm's procedures are usually demonstrated as follows.

Step 1: Population initialization

Within a certain range, a random initial population with the number of N is generated, and each individual in the population becomes a chromosome.

Step 2: Code the population

The initial population is coded according to binary rules, which are made up of Numbers 0 and 1.

Step 3: Calculate fitness

Mean square error (MSE) is selected as the objective function. According to the objective function, the fitness value of each individual in the population is calculated.

$$f_i = \frac{1}{X} \sum_{i=1}^X (Y_i - \bar{Y}_i)^2 \quad (9)$$

where f_i represents individual fitness; Y_i = actual output value of the sample; \bar{Y}_i = expected output value of sample; X = number of samples.

Step 4: Select operator

According to the fitness value of each individual, individuals with high fitness are selected to carry out to the next iteration operation, while those with low fitness are less likely to enter the next iteration operation or may even be eliminated. The probability of an individual being selected is proportional to its fitness by using the wheel type probability selection method. The probability selection is as follows.

$$P_k = \frac{f_k}{\sum_{i=1}^N f_i} \quad (10)$$

where, N = initial population number, f_k is the fitness value of individual k and P_k is the probability that individual k is selected.

Step 5: Crossover operator

The selected individuals pair with each other according to the principle of arithmetic crossover, exchange some genes, and form new individuals, which will have the characteristics of their parents. The arithmetic crossover operator is as follows [27].

$$x'_1 = ax_1 + (1-a)x_2, \quad a \in (0, 1) \quad (11)$$

$$x'_2 = ax_2 + (1-a)x_1, \quad a \in (0, 1) \quad (12)$$

where, $x_{1,2}$ represent two parent individuals, while $x'_{1,2}$ represent two offspring individuals, a is a random number between 0 and 1.

Step 6: Mutation operator

By replacing certain alleles on individual chromosomes with a certain probability of mutation, new individuals different from their parents can be created to expand the population size.

Repeat Step 3 to Step 6. The algorithm converges by iteration. When the iteration number reaches the maximum iteration number T , the individuals with the maximum fitness obtained in the evolutionary process are taken as the output of the optimal solution.

2.6. The Combined Model of PSO, GA, and BPNN

Both PSO and GA are optimization algorithms that try to simulate the adaptability of individual populations on the basis of natural characteristics. Both of them use certain transformation rules to solve the problem through searching space. Besides, they both have parallel search features, thus reducing the possibility of falling into local minimum in BPNN. Both PSO and GA algorithms have good optimization performance but also have disadvantages and limitations. Both in PSO and GA, parameters are determined by experience, which will cause premature convergence, slow convergence speed, and finally affect the optimization performance.

PSO and GA optimize BPNN by optimizing the connection weight and threshold of BPNN. This PSO and GA hybrid algorithm is based on the PSO algorithm, and the genetic algorithm is added in the process of the PSO algorithm. It combines the advantages of the two algorithms and

has the advantages of less computation, fast convergence, and good global convergence performance. The steps of the PSO-GA-BP are as follows.

Step 1. BP neural network initialization. According to the input and output dimensions of the model, the hierarchical structure of the neural network and the number of nodes in the hidden layer are determined.

Step 2. Particle swarm initialization. According to the network structure, the particle parameters and the number of particles are determined. The velocity and position of the particles are encoded by binary code. The mean square error (MSE) is selected to calculate the fitness value.

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y - \hat{y})^2 \quad (13)$$

Step 3. Calculate the fitness value. Calculate the fitness value of each particle and determine whether the target conditions are met. If the target condition is met, the output result is obtained; If the target condition is not satisfied, the individual optimum and global optimum of the particles are updated.

Step 4. PSO added Crossover operator. Particle swarm optimization adds the selection crossover step of genetic algorithm, selects the particles with better fitness by the wheel bet method, crosses the position and speed of the particle group according to the probability $P_a = 0.4$, and selects the particles with better fitness after crossing into the particle group for the next iteration.

Step 5. PSO added Mutation operator. Particle swarm optimization adds the mutation step of the genetic algorithm. According to the probability $P_b = 0.01$, the mutation operation is carried out on the position and velocity of the particles with poor fitness in the particle swarm, and the particles after the mutation operation are put into the particle swarm.

Step 6. Calculate the fitness value by fitness function and update the $gBest$ and $zBest$ particle swarm optimization.

Step 7. Determine whether the target value of the set particle swarm is met or has reached the maximum evolutionary algebra. If the condition is met, the optimal solution $zBest$ is output; If it is not met, then jump to step (3) and continue to complete the iteration.

Step 8. Decode the optimal solution after the iteration is completed and substitute initial weights and thresholds into the preset BP neural network. Further, the PSO-GA-BP neural network model was obtained.

Flow chart of PSO-GA-BP neural network prediction model is shown in Figure 2 as follows.

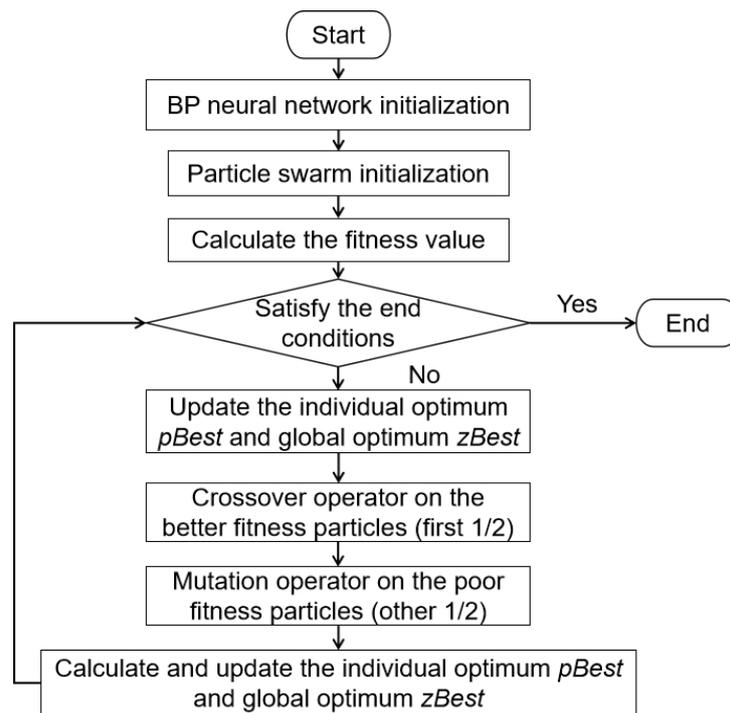


Figure 2. Flow chart of PSO-GA-BP neural network prediction model.

2.7. Evaluation of Performance

In this study, a time series data set was divided into two subsets for training and testing the models; the first 70% of the data set was used to train and the remaining (30%) was used to test the models. To evaluate the performance of the PSO-GA-BPNN model and other prediction models, some comparison standards were employed as follows.

$$APE = \frac{|y - \hat{y}|}{y} \times 100\% \tag{14}$$

$$MAPE = \frac{1}{n} \sum_{t=1}^n \frac{|y - \hat{y}|}{y} \times 100\% \tag{15}$$

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (y - \hat{y})^2}{n}} \tag{16}$$

$$R^2 = \frac{(\sum_{t=1}^n (y - \bar{y})(\hat{y} - \bar{\hat{y}}))^2}{\sum_{t=1}^n (y - \bar{y})^2 \sum_{t=1}^n (\hat{y} - \bar{\hat{y}})^2} \tag{17}$$

where APE is the absolute percentage error, MAPE is the mean absolute percentage error, RMSE is the root mean square error. In Equations (14)–(17), y is the measured quality parameter in period t , while \hat{y} is the predicted quality parameter, n is the total number of periods. R^2 is the coefficient of determination, \bar{y} is the average of measured quality parameter in period t , $\bar{\hat{y}}$ is the average of the predicted quality parameter in these equations.

The APE is able to show how far each predicted value deviated from the measured value even when the value of error is very small. APE_{max} also shows the point at which the worst prediction effect occurs. Average prediction accuracy can be seen in MAPE, which helps to show the performance of the prediction model. RMSE is recognized as one of the most important indicators for evaluating the performance of prediction models; the smaller the value of RMSE, the better the prediction of the

model. While the value of R^2 (between 0 and 1) represents whether the measured value is related to the predicted value, it represents the degree of fitting between the measured value and the predicted value. The closer the value of R^2 is to 1, then the better the fitting result is. Normally, MAPE and RMSE are always used to evaluate the accuracy, while APE and R^2 are more suitable for evaluating the robustness of the model.

3. Results and Discussion

The algorithm was realized by the mathematical software Matlab18a which provides support for algorithm development, data visualization, data analysis, and numerical calculation and has the advantages of simple operation, convenient operation, and fast calculation speed, in which the structure of the BP neural network was 5–11–1. The maximum number of iterations was 2000, the threshold of error precision ε was 0.0001, and the learning rate η was 0.005. The population size was set at 50 and the number of evolutions was 200 in the PSO algorithm and GA algorithm. The learning factor $c_1 = c_2 = 1.49$. PH, CHLA, NH₄H, BOD, and EC were used as inputs; the subsequent DO predictive values were used as the output in models. According to the steps of the neural network training algorithm model, after the prediction model is properly trained, prediction of the dissolved oxygen concentration of the Beihai Lake was possible. In order to test the algorithm's performance, the researchers compared the PSO-GA-BP neural network algorithm with the common BP neural network model, the PSO-BP neural network model, and the GA-BP neural network model. The prediction performance of the four models are shown in Figures 3 and 4, while the water quality prediction results are listed in Tables 2 and 3 as follows.

Figure 3 shows the water quality prediction performance of these models intuitively, and it is easy to identify the most different time point between the predicted value and the measured value (real value). Table 2 shows the specific values of the predicted results and the observed results in time series. It can be seen in Figure 3 that the BP neural network prediction model optimized with the genetic algorithm and particle swarm optimization is much better than the traditional BP neural network prediction model. It can even be said that the BP neural network model without optimization shows poor prediction, while the curve fitting result of PSO-GA-BPNN model is the closest to that of the curve of real value.

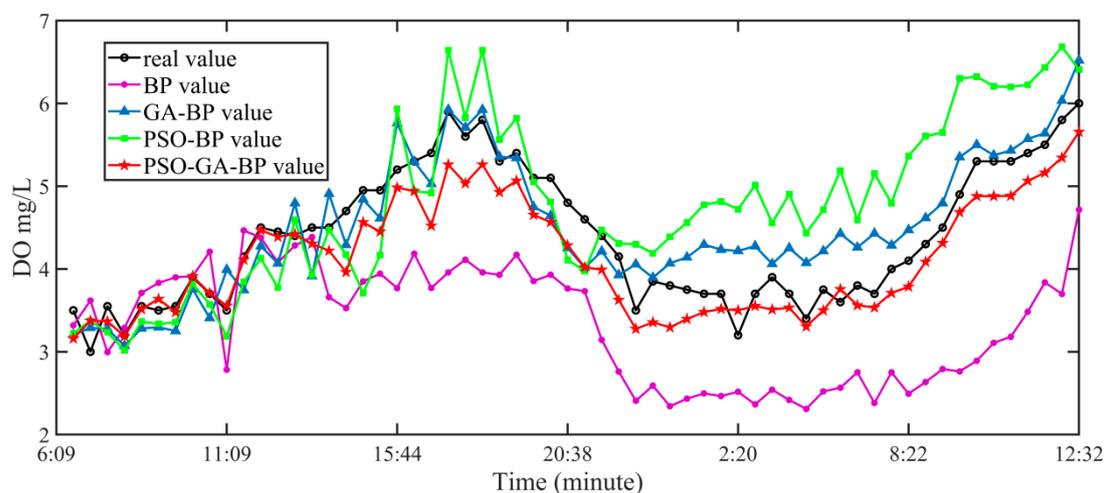


Figure 3. Comparison of predictions by the BPNN, PSO-BPNN, GA-BPNN, and PSO-GA-BPNN models.

Table 2. Comparison of the measured value and the predicted value in time series.

Time(min)	Measured DO (mg/L)	BPNN Estimated DO (mg/L)	GA- BPNN Estimated DO (mg/L)	PSO- BPNN Estimated DO (mg/L)	PSO-GA- BPNN Estimated DO (mg/L)
06:38	3.5	3.319104	3.204458	3.221109	3.159507
07:12	3	3.61863	3.288944	3.374401	3.374958
07:50	3.55	2.99579	3.276824	3.236666	3.367245
08:14	3.2	3.28741	3.072583	3.016254	3.185695
09:00	3.55	3.711411	3.28187	3.363945	3.515026
09:24	3.5	3.832495	3.295884	3.338031	3.637039
10:06	3.55	3.898983	3.251454	3.355381	3.478035
10:32	3.9	3.913548	3.756746	3.819875	3.912443
10:53	3.7	4.208063	3.407529	3.57347	3.708232
11:09	3.5	2.783473	3.991442	3.187187	3.556292
11:38	4.15	4.465491	3.744656	3.843019	4.111433
12:15	4.5	4.377572	4.275918	4.126045	4.467993
12:41	4.45	4.077742	4.066996	3.774387	4.388143
13:22	4.4	4.284307	4.795432	4.594932	4.419245
13:58	4.5	4.38479	3.913068	3.923559	4.304517
14:25	4.5	3.658515	4.908481	4.47059	4.220097
14:46	4.7	3.527018	4.290607	4.174959	3.962504
15:03	4.95	3.850285	4.840639	3.708508	4.56419
15:25	4.95	3.942745	4.61324	4.167664	4.449615
15:44	5.2	3.769086	5.761613	5.931281	4.983137
16:09	5.3	4.184881	5.296669	4.937933	4.939002
16:42	5.4	3.770225	5.027915	4.918132	4.521631
17:5	5.9	3.957918	5.923698	6.641184	5.259978
16:05	5.6	4.112009	5.708972	5.828493	5.033455
17:32	5.8	3.957918	5.923698	6.641184	5.259978
17:59	5.3	3.92944	5.355163	5.565494	4.926943
18:26	5.4	4.17127	5.340072	5.828493	5.065938
18:51	5.1	3.853098	4.74945	5.053349	4.653622
19:56	5.1	3.929797	4.642732	4.807976	4.565552
20:38	4.8	3.764067	4.254866	4.110626	4.285194
21:04	4.6	3.730952	4.015172	3.973494	4.020465
21:47	4.4	3.143282	4.21379	4.47169	3.99098
22:10	4.15	2.760956	3.927626	4.309683	3.625617
22:54	3.5	2.408149	4.057317	4.298296	3.276418
23:21	3.85	2.590122	3.894758	4.18824	3.355598
00:02	3.8	2.34191	4.06850	4.387963	3.294731
00:33	3.75	2.433357	4.140601	4.560365	3.395101
01:13	3.7	2.497519	4.293738	4.774741	3.479532
01:37	3.7	2.464745	4.233978	4.814534	3.516759
02:20	3.2	2.516551	4.217563	4.721049	3.499973
02:48	3.7	2.36452	4.27645	5.011103	3.552755
03:26	3.9	2.542249	4.06065	4.555414	3.510115
03:52	3.7	2.417853	4.251827	4.903804	3.532915
04:37	3.4	2.309732	4.075028	4.432614	3.303942
05:00	3.75	2.520505	4.216435	4.714122	3.49873
05:48	3.6	2.565653	4.426973	5.187666	3.756473
06:08	3.8	2.751876	4.260384	4.593129	3.561234
06:54	3.7	2.381106	4.428611	5.15049	3.535047
07:13	4	2.752043	4.286189	4.795322	3.709041
08:22	4.1	2.493497	4.470807	5.364273	3.786323
08:48	4.3	2.634323	4.616357	5.603613	4.0856
09:07	4.5	2.790682	4.795054	5.648655	4.31199
09:26	4.9	2.763163	5.350451	6.299513	4.685816
10:10	5.3	2.890539	5.499668	6.322683	4.8796
10:35	5.3	3.107112	5.372916	6.206093	4.879736
11:03	5.3	3.18107	5.430898	6.199451	4.882933
11:32	5.4	3.484189	5.571332	6.223342	5.064208
11:57	5.5	3.837739	5.638552	6.432722	5.162566
12:10	5.8	3.698872	6.033317	6.682627	5.342825
12:32	6	4.71609	6.518356	6.408937	5.655351

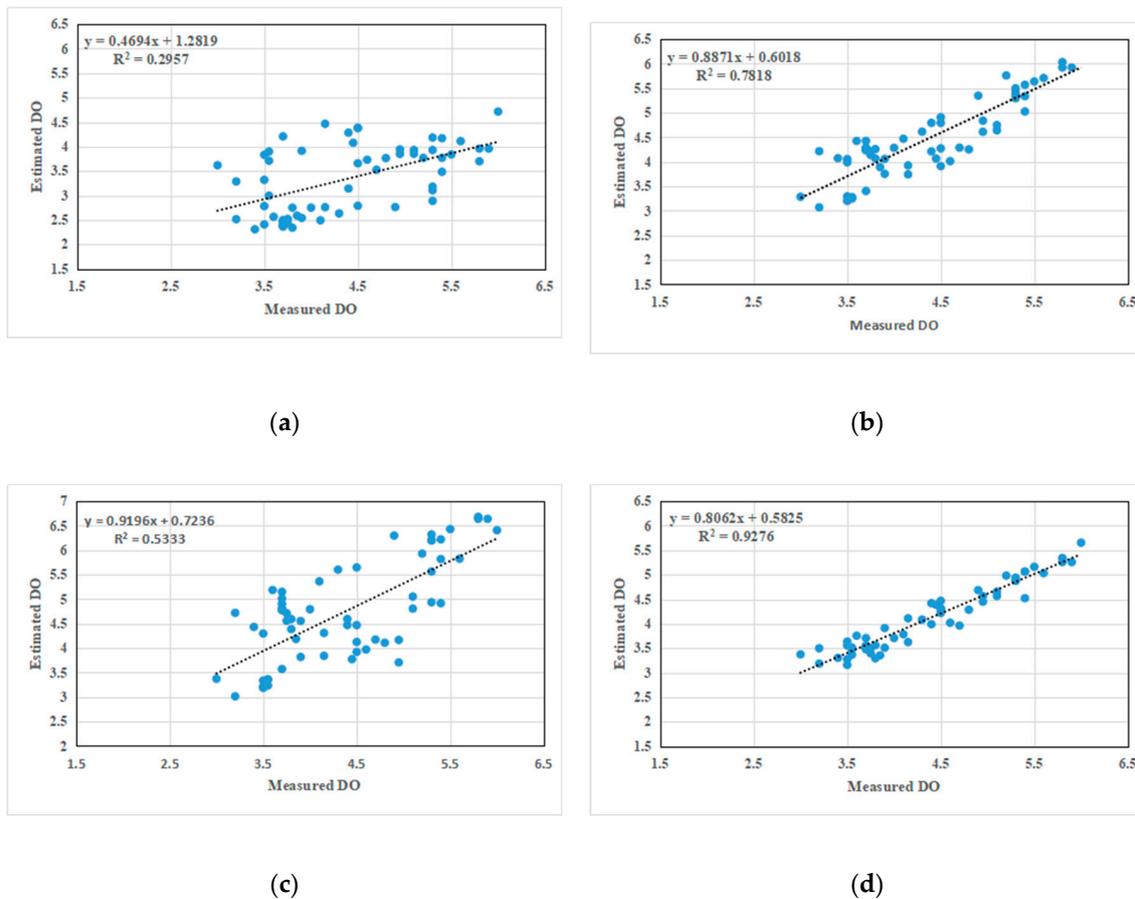


Figure 4. R^2 of the (a) BPNN, (b) PSO-BPNN, (c) GA-BPNN, and (d) PSO-GA-BPNN models.

Table 3. The performance of the four models.

Model	APE(%)	MAPE(%)	RMSE	R^2
BPNN	45.4614	25.1506	1.2733	0.2957
GA-BPNN	31.7989	8.4506	0.4019	0.7818
PSO-BPNN	47.5328	15.7102	0.7873	0.5333
PSO-GA-BPNN	16.2661	6.7219	0.3596	0.9276

For the four models in our testing, the results of APE_{max} , MAPE, RMSE, and R^2 have been respectively shown in Table 3. One can see exactly how the model performed and the degree to which the four models differ. Figure 4 shows R^2 , the linear formula, and the linear curve. PSO-GA-BPNN shows the best prediction effect in terms of both accuracy and robustness. As can be seen from Table 3, the root mean square error mildly dropped from 1.2733, 0.7873, and 0.4019 to 0.3596; the max absolute percentage error dropped from 45.4614%, 47.5328%, and 31.7989% to 16.2661%; the mean absolute percentage error dropped from 25.1506%, 15.7102%, and 8.4506% to 6.7219%; and the coefficient of determination rose from 0.2957, 0.5333, and 0.7818 to 0.9276. PSO-GA-BP achieved the best model performance in all evaluation indices. Specifically, MAPE went up 14.4% and RMSE shrunk to a quarter of the worst result in terms of accuracy; APE_{max} went up 29.2% and R^2 had increased by 63.2% compared with the common BP neural network model. We conclude that the model based on combined particle swarm optimization and genetic algorithm can better fit the complex dynamic nonlinear relationship between the water ecological environment factors and dissolved oxygen. Furthermore, the improved prediction results of the PSO-GA-BPNN algorithm correspond to the real observations of Beihai Lake.

4. Conclusions

Water quality prediction plays an important role in the control, management, and planning of water quality. The common BP neural network prediction model has many weaknesses, so we combined the genetic optimization algorithm and particle swarm optimization algorithm with BP neural network, and established the PSO-GA-BPNN prediction model in this study. The improved model integrated the function of self-learning, bionic, and nonlinear approximation technology. This network learning model realized fast convergence speed, avoidance of local minima, stronger stability, and suitable results. Due to current hardware, the training time of PSO-GA-BPNN model is relatively long. The PSO-GA-BPNN had improved prediction accuracy and robustness.

There is an urgent requirement for new methods to deal with abnormal data. In complex aquatic ecosystems, the proposed model can meet the management requirements of water quality monitoring and early warning. It is strongly recommended to establish different predictive models according to diversified weather conditions and complex surface environments, and to combine those prediction models to improve the prediction accuracy because water quality is heavily affected by hydrological, meteorological, and surficial factors.

Author Contributions: Conceptualization, J.Y. and K.G.; Investigation, H.X.; Methodology, Z.X.; Project administration, Y.Y.; Resources, J.Y.; Software, Z.X. and K.G.; Supervision, Y.Y. and H.X.; Writing—original draft, Z.X.

Funding: This research was funded by [Water Pollution Control and Treatment Science and Technology Major Project] grant number [2018ZX07111005] and The APC was funded by [Water Pollution Control and Treatment Science and Technology Major Project].

Acknowledgments: The authors are grateful to the Beijing Water-affair Authority for making available the water quality data of the Beihai Lake. The authors thank Jianzhao Yan and Yongchuan Yu for their thoughtful advice and suggestions on research design and implementation.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Li, K.; Wang, L.; Li, Z.; Xie, Y.; Wang, X.; Fang, Q. Exploring the Spatial-Seasonal Dynamics of Water Quality, Submerged Aquatic Plants and Their Influencing Factors in Different Areas of a Lake. *Water* **2017**, *9*, 707. [[CrossRef](#)]
- Carbajal-Hernández, J.J.; Sánchez-Fernández, L.P.; Villa-Vargas, L.A.; Carrasco-Ochoa, J.A.; Martínez-Trinidad, J.F. Water quality assessment in shrimp culture using an analytical hierarchical process. *Ecol. Indic.* **2013**, *29*, 148–158. [[CrossRef](#)]
- Ip, W.C.; Hu, B.Q.; Wong, H.; Xia, J. Applications of grey relational method to river environment quality evaluation in China. *J. Hydrol.* **2009**, *379*, 284–290. [[CrossRef](#)]
- Liu, Y.; Yang, P.; Hu, C.; Guo, H. Water quality modeling for load reduction under uncertainty: A Bayesian approach. *Water Res.* **2008**, *42*, 3305–3314. [[CrossRef](#)]
- Cho, J.H.; Seok, S.K.; Ryong, H.S. A river water quality management model for optimising regional wastewater treatment using a genetic algorithm. *J. Environ. Manag.* **2004**, *73*, 229–242. [[CrossRef](#)] [[PubMed](#)]
- Ding, Y.R.; Cai, Y.J.; Sun, P.D.; Chen, B. The Use of Combined Neural Networks and Genetic Algorithms for Prediction of River Water Quality. *J. Appl. Res. Technol.* **2014**, *12*, 493–499. [[CrossRef](#)]
- Palani, S.; Liong, S.Y.; Tkalic, P. An ANN application for water quality forecasting. *Mar. Pollut. Bull.* **2008**, *56*, 1586–1597. [[CrossRef](#)] [[PubMed](#)]
- Han, H.G.; Chen, Q.L.; Qiao, J.F. An efficient self-organizing RBF neural network for water quality prediction. *Neural Netw.* **2011**, *24*, 717–725. [[CrossRef](#)] [[PubMed](#)]
- Ahmed, A.M.; Shah, S.M.A. Application of adaptive neuro-fuzzy inference system (ANFIS) to estimate the biochemical oxygen demand (BOD) of Surma River. *J. King Saud Univ. Eng. Sci.* **2015**, *12*, 237–243. [[CrossRef](#)]
- Mahmoodabadi, M.; Arshad, R.R. Long-term evaluation of water quality parameters of the Karoun River using a regression approach and the adaptive neuro-fuzzy inference system. *Mar. Pollut. Bull.* **2018**, *126*, 372–380. [[CrossRef](#)]

11. Lee, Y.; Oh, S.H.; Kim, M.W. The effect of initial weights on premature saturation in back-propagation learning. In Proceedings of the International Joint Conference on Neural Networks, Singapore, 18–21 November 1991; pp. 765–770.
12. Jadav, K.; Panchal, M. Optimizing weights of artificial neural networks using genetic algorithms. *Int. J. Adv. Res. Comput. Sci. Electron. Eng. (IJARCSEE)* **2012**, *1*, 47–51.
13. Momeni, E.; Nazir, R.; Armaghani, D.J.; Maizir, H. Prediction of pile bearing capacity using a hybrid genetic algorithm-based Ann. *Measurement* **2014**, *57*, 122–131. [[CrossRef](#)]
14. Hu, Y.; Li, J.; Hong, M.; Ren, J.; Lin, R.; Liu, Y.; Liu, M.; Man, Y. Short term electric load forecasting model and its verification for process industrial enterprises based on hybrid GA-PSO-BPNN algorithm—A case study of papermaking process. *Energy* **2019**, *170*, 1215–1227. [[CrossRef](#)]
15. Yu, S.; Wei, Y.; Wang, K. A PSO-GA optimal model to estimate primary energy demand of China. *Energy Policy* **2012**, *42*, 329–340. [[CrossRef](#)]
16. Yu, S.; Zhu, K.; Zhang, X. Energy demand projection of China using a path-coefficient analysis and PSO-GA approach. *Energy Convers. Manag.* **2012**, *53*, 142–153. [[CrossRef](#)]
17. Jian, Y.; Xinying, L.; Man, Z.; Han, L. Photosynthetic Rate Prediction of Tomato Plant Population Based on PSO and GA. *IFAC-Paper OnLine* **2018**, *51*, 61–66. [[CrossRef](#)]
18. Mentzafou, A.; Dimitriou, E. Nitrogen loading and natural pressures on the water quality of a shallow Mediterranean lake. *Sci. Total Environ.* **2019**, *646*, 134–143. [[CrossRef](#)]
19. Pu, X.; Cheng, H.; Tysklind, M.; Xie, J.; Lu, L.; Yang, S. Occurrence of water phosphorus at the water-sediment interface of a freshwater shallow lake: Indications of lake chemistry. *Ecol. Indic.* **2017**, *81*, 443–452. [[CrossRef](#)]
20. Bian, B.; Zhou, Y.; Fang, B.B. Distribution of heavy metals and benthic macroinvertebrates: Impacts from typical inflow river sediments in the Taihu Basin, China. *Ecol. Indic.* **2016**, *69*, 348–359. [[CrossRef](#)]
21. Chen, S.; Carey, C.C.; Little, J.C.; Lofton, M.E.; McClure, R.P.; Lei, C. Effectiveness of a bubble-plume mixing system for managing phytoplankton in lakes and reservoirs. *Ecol. Eng.* **2018**, *113*, 43–51. [[CrossRef](#)]
22. He, H.; Jin, H.; Jeppesen, E.; Li, K.; Liu, Z.; Zhang, Y. Fish-mediated plankton responses to increased temperature in subtropical aquatic mesocosm ecosystems: Implications for lake management. *Water Res.* **2018**, *144*, 304–311. [[CrossRef](#)]
23. Wang, S.; Gao, Y.; Li, Q.; Gao, J.; Zhai, S.; Zhou, Y.; Cheng, Y. Long-term and inter-monthly dynamics of aquatic vegetation and its relation with environmental factors in Taihu Lake, China. *Sci. Total Environ.* **2018**, *651*, 367–380. [[CrossRef](#)] [[PubMed](#)]
24. Xu, H.; Brown, D.G.; Moore, M.R.; Currie, W.S. Optimizing spatial land management to balance water quality and economic returns in a Lake Erie watershed. *Ecol. Econ.* **2018**, *145*, 104–114. [[CrossRef](#)]
25. Wu, Q.; Xia, X.; Mou, X.; Zhu, B.; Zhao, P.; Dong, H. Effects of seasonal climatic variability on several toxic contaminants in urban lakes: Implications for the impacts of climate change. *J. Environ. Sci.* **2014**, *26*, 2369–2378. [[CrossRef](#)] [[PubMed](#)]
26. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
27. Fister, I.; Tepeh, A.; Fister, I., Jr. Epistatic arithmetic crossover based on Cartesian graph product in ensemble differential evolution. *Appl. Math. Comput.* **2016**, *283*, 181–194.

