Hindawi
Complexity
Volume 2019, Article ID 3429816, 15 pages
https://doi.org/10.1155/2019/3429816

Research Article

Prediction Model of Corrosion Current Density Induced by Stray Current Based on QPSO-Driven Neural Network

Chengtao Wang,[1] **Wei Li**,[1] **Gaifang Xin**,[2,3] **Yuqiao Wang**,[1] **and Shaoyi Xu**[1]

[1]*School of Mechatronic Engineering, China University of Mining and Technology, Xuzhou, Jiangsu 221116, China*
[2]*Department of Intelligent Equipment, Changzhou College of Information Technology, Changzhou, Jiangsu 213164, China*
[3]*College of Internet of Things Engineering, Hohai University, Changzhou 213022, China*

Correspondence should be addressed to Wei Li; cmeecumt512@yahoo.com

Received 20 May 2019; Revised 26 August 2019; Accepted 16 September 2019; Published 31 October 2019

Academic Editor: Qingling Wang

The buried pipelines and metallic structures in subway systems are subjected to electrochemical corrosion under the stray current interference. The corrosion current density determines the degree and the speed of stray current corrosion. A method combining electrochemical experiment with the machine learning algorithm was utilized in this research to study the corrosion current density under the coupling action of stray current and chloride ion. In this study, a quantum particle swarm optimization-neural network (QPSO-NN) model was built up to predict the corrosion current density in the process of stray current corrosion. The QPSO algorithm was employed to optimize the updating process of weights and biases in the artificial neural network (ANN). The results show that the accuracy of the proposed QPSO-NN model is better than the model based on backpropagation neural network (BPNN) and particle swarm optimization-neural network (PSO-NN). The accuracy distribution of the QPSO-NN model is more stable than that of the BPNN model and the PSO-NN model. The presented model can be used for the prediction of corrosion current density and provides the possibility to monitor the stray current corrosion in subway system through an intelligent learning algorithm.

1. Introduction

Stray current is defined as electric currents flowing along other elements, which are not components of the purpose-built electric current [1]. Stray current can originate from electrified traction system, offshore structure, marine platforms, cathodic protection system, etc. Stray current leakage and its corrosion in the direct current (DC) mass transit system are shown in Figure 1. Stray current will cause electrochemical corrosion in some of the infrastructure, such as the running rails [2], buried pipelines [3], and concrete structures [4, 5], which overall reduces the safety performance of the DC mass transit system. In particular, it needs to be explained that most pipelines consist of carbon steel, which makes corrosion one of the most common integrity threats of these systems [6]. In some extreme cases, severe structural damage has occurred as a result of stray current leakage [7]. The reasons for stray current formation are diversified and complex and are easily affected by many factors, such as stray current density, structure of buried metal, salt content and composition, soil moisture content, soil texture and tightness, organic content, and temperature.

In the face of rapid development of urban rail transit, it is especially important to understand and monitor the process of stray current corrosion that happens in the DC transit system. The existing monitoring of stray current corrosion is mainly conducted through reference electrode. However, problems existed. For example, the polarization potential of reference electrode sometimes cannot represent the corrosion status of buried pipelines accurately due to the influence of IR drop and internal reactants. The faulty reference electrode is hard to replace in the subway system because it is difficult to excavate the electrode from the concrete structure in order to ensure the overall strength of the system. Besides, because the subway

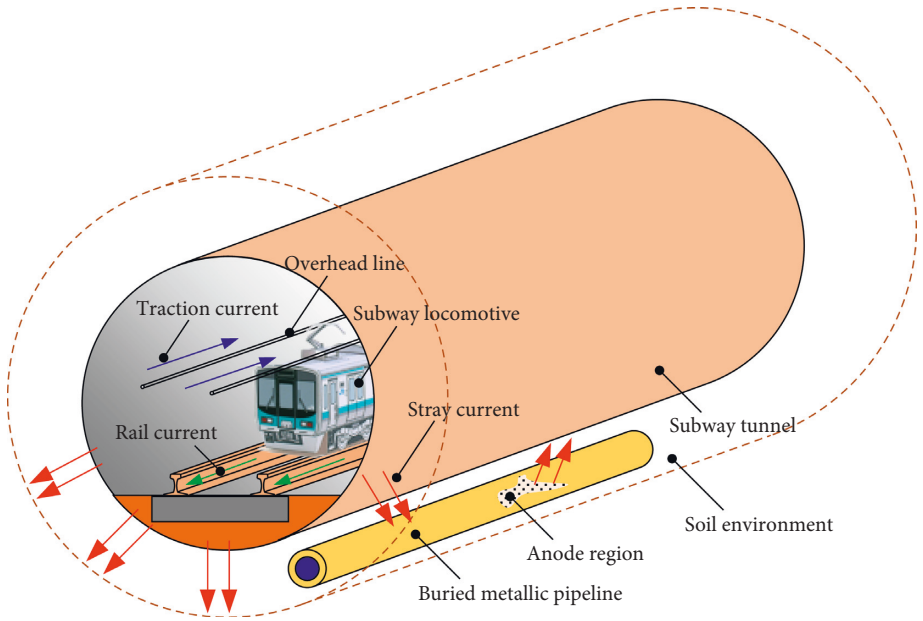

Figure 1: Stray current in the DC mass transit system.

system is difficult to make structural changes after completion, accurate corrosion monitoring without excavation is especially important. Therefore, a method for accurately establishing the mapping relationship between the external environment factors and corrosion status is urgently needed. Based on Faraday's law, corrosion current density i_{corr} is an important parameter in the direct evaluation of stray current corrosion. The corrosion current density not only reflects the degree of stray current corrosion but also the corrosion rate. Corrosion current density is affected by external factors during stray current corrosion. Since the stray current corrosion is a complex process with multiple factors, it is difficult to accurately establish the relationship between external factors and corrosion current density through numerical modelling. The prediction of corrosion current density by a machine learning algorithm is of great practical significance for the evaluation of stray current corrosion state.

As an effective tool to study the corrosion process, many methods based on the machine learning algorithm have been employed, such as the backpropagation neural network (BPNN) model [8, 9], radial basis function (RBF) neural network model [10], support vector machine (SVM) model [11, 12], self-organized map (SOM) model [13], and grey theory model [14]. The intelligence algorithm is often utilized to predict the corrosion current density in steel reinforced concrete structures [13], polarization potential [10], stray current density [8], weight loss [14], corrosion rate [11], corrosion hazard classification of stray current [15], and so on. In the last few years, artificial neural network (ANN) has emerged as powerful devices that can be used in many engineering applications [16–21]. The ANN has been widely employed in the field of corrosion prediction. Some important research results on corrosion prediction have been carried out based on the ANN [22–24]. The main advantage of ANN is its outstanding capabilities in capturing the nonlinear relationships between input and output corrosion data sets. However, some problems also exist in the application of ANN, such as weak global search ability, and poor robustness, and it is easy to fall into local minimum value. Therefore, many algorithms such as the genetic algorithm (GA) [25, 26], gravitational search algorithm (GSA) [27], and particle swarm optimization (PSO) [28, 29] are utilized to optimize the ANN.

In this paper, stray currents existing in the main structure of the subway systems, buried pipelines, and metal structures around the system are studied using the intelligent algorithm-based method combining with an electrochemical experiment. NaCl is an important composition of the extracted soil solution [3]. Thus, the NaCl solution is adopted as the corrosive medium in this study. The quantum particle swarm optimization-neural network (QPSO-NN) model is established to predict the corrosion current density under the coupling excitation of stray current and chloride ion. The quantum particle swarm optimization (QPSO) is utilized to optimize the weight and threshold of each layer. Compared with PSO, QPSO has less parameters, better robustness, and faster convergence rate [10], which has been successfully employed in many optimization problems [30], including the neural network [31]. However, the QPSO algorithm is rarely utilized in the field of corrosion prediction, especially the electrochemical corrosion induced by stray current, in which some new explorations and attempts are needed to improve the performance of stray current corrosion prediction. QPSO can better converge to the global best, shows strong global search ability, and local search ability, which is not easy to fall into the local best [32, 33]. The database for training the QPSO-NN model is collected from the electrochemical experiment of Q235A.

Aiming at stray current corrosion process, the purpose of this paper is to realize the prediction of corrosion current density considering multiple environment factors based on the QPSO-NN algorithm, and then to provide the possibility of accurately monitoring the corrosion status of buried

pipelines without excavation. Thus, the experimental process is firstly conducted to determine the main factors affecting the corrosion current density and then to set up the prediction database. The rest of this paper is organized as follows: The theory of the proposed QPSO-NN algorithm is described in Section 2. Section 3 introduces the experimental system designed for generating data set for the prediction model of corrosion current density and experimental results. Section 4 introduces the corrosion current density prediction framework and the related technique details. In Section 5, the training process is stated and the QPSO-NN method is compared with the BPNN method and the particle swarm-neural network (PSO-NN) method. Further analysis of parameters in the algorithm is also conducted in Section 5. Section 6 draws the conclusions.

2. Training Algorithm of the Proposed Model Based on QPSO

2.1. Brief Review on Artificial Neural Network. In 1949, Donald Hebb's research showed that biological conditioning is generally caused by the nature of neurons, pointing to a learning method of the biological neural unit's learning mechanism, which is now known as neural plasticity [34]. Frank Rosenblatt firstly proposed the perception network model and used it in the computer addition, subtraction, and pattern recognition calculation in 1958 [35]. In 1982, the Hopfield model was proposed to provide nonlinear neural summarization for the information storage and extraction functions of the ANN33, in which the dynamic equations and learning equations were also put forward. This model provides theoretical guidance for the construction and learning of ANN [36]. The ANN has strong nonlinear mapping capabilities, self-learning and self-adaptive capabilities, generalization capabilities, and fault tolerance. The basic structure of the ANN includes information called the processing unit or node of neuron and the connection relationship of each node, which simplifies the imitation and abstraction of the biological neural network. The neurons in the ANN are classified into three types of layers: input layer, hidden layer(s), and output layer. The relationship between various neurons is a singular relationship, and there is no difference. The connection weights can be adaptively adjusted through continuous learning.

2.2. Quantum Particle Swarm Optimization. The QPSO algorithm based on quantum behavior is generated from the classical PSO algorithm. It is mainly combined with the improvement of quantum theory behavior, which improves the global optimization performance of the algorithm. In quantum mechanics, the state of a particle is described by a wave function $\Psi(X,t)$, which is a complex function of the particle position X (X is a position vector) and time t. The physical meaning of the wave function can be expressed as the square of the wave function modulus represents the probability density at which the particle appears at spatial position X. By integrating the wave function over entire space, the sum of the probability of particles appearing at various points in space is obtained. In n-dimensional space, one-dimensional potential well δ at the centre of p_i for each dimension is defined, in coordination with the attractor $p_i = (p_{i1}, p_{i2}, \ldots, p_{iN})$. Each particle converges to a certain area by an attractor p_i. The position coordinate of the attractor p_i is as follows:

$$p_{i,j}(t) = \varphi_{i,j}(t) P_{i,j}(t) + \left[1 - \varphi_{i,j}(t)\right], \quad 1 \le j \le N, \tag{1}$$

$$\varphi_{i,j}(t) = \frac{c_1 r_{1,j}(t)}{c_1 r_{1,j}(t) + c_2 r_{2,j}(t)}, \tag{2}$$

where $r_{1,j}$, $r_{2,j}$, and $\varphi_{i,j}$ are all independent random positive real numbers between [0, 1]; $P_{i,j}(t)$ represents the local optimal position of attractor p_i; $G_j(t)$ represents the global optimal position of the population; c_1 and c_2 is local cognitive coefficient and overall cognitive coefficient, respectively.

According to the theory of quantum mechanics, the dynamic behavior of particles in quantum space can be described by Schrödinger equation. In the QPSO algorithm, the wave function in the Schrödinger equation is used to express the position of the particles in the quantum particle space. The wave function of a particle with the position of $X_{i,j}(t+1)$ in each dimension for $p_{i,j}$ can be obtained as follows:

$$\psi\left[X_{i,j}(t+1)\right] = \frac{1}{\sqrt{L_{i,j}(t)}} \exp\left[-\frac{\left|X_{i,j}(t+1) - p_{i,j}(t)\right|}{L_{i,j}(t)}\right]. \tag{3}$$

After the probability density function for a particle appearing at a certain point in the quantum space is calculated, the Monte Carlo method is employed to obtain the position equation of the particle. The position equation is given in the following equation:

$$x_{i,j}(t+1) = p_{i,j}(t) \pm \frac{L_{i,j}(t)}{2} \ln\left[\frac{1}{u_{i,j}(t)}\right], \tag{4}$$

$$u_{i,j}(t) \sim U(0,1). \tag{5}$$

The $L_{i,j}(t)$ in equation (4) is the characteristic length of the potential well δ, which is calculated as follows:

$$L_{i,j}(t) = 2\alpha \left| mbest_j(t) - x_{i,j}(t) \right|, \tag{6}$$

where α is the contraction-expansion coefficient, the only parameter in the QPSO algorithm except from population size and iteration times.

The mean best position $mbest(t)$ in equation (6) is the individual optimal mean of all particles, which is defined as follows:

$$\begin{aligned} mbest(t) &= \frac{1}{M} \sum_{i=1}^{M} P_i(t) \\ &= \left[\frac{1}{M} \sum_{i=1}^{M} P_{i1}(t), \frac{1}{M} \sum_{i=1}^{M} P_{i2}(t), \ldots, \frac{1}{M} \sum_{i=1}^{M} P_{iD}(t)\right]. \end{aligned} \tag{7}$$

In equation (7), $P_{ij}(t)$ represents the individual best position of the j^{th} dimension of the i^{th} particle at the t^{th} iteration. For minimization, the smaller the objective function value, the better the fitness it corresponds to. When the current position of the i^{th} particle is $X_i(t) = [X_{i,1}(t), X_{i,2}(t), \ldots, X_{i,D}(t)]$ $(i = 1, 2, \ldots, M)$, the current individual optimal position of the i^{th} particle can be updated through equation (8), where $f(\,)$ is the object function. The current global optimal position $G(t)$ can be obtained through equation (9), where $G(t) = [G_1(t), G_2(t), \ldots, G_D(t)]$:

$$P_i(t) = \begin{cases} X_i(t), & \text{if: } f[X_i(t)] < f[P_i(t-1)], \\ P_i(t-1), & \text{if: } f[P_i(t-1)] \le f[X_i(t)], \end{cases} \tag{8}$$

$$\begin{aligned} G(t) &= P_g(t), \\ g &= \arg \min_{1 \le i \le M} \{f[P_i(t)]\}. \end{aligned} \tag{9}$$

Finally, the evolution formula of the QPSO can be expressed as follows:

$$RP_{ij}(t) = \varphi_j(t) \times P_{ij}(t) + \left(1 - \varphi_j(t)\right) \times G_j(t), \tag{10}$$

$$\begin{aligned} X_{ij}(t+1) = RP_{ij}(t) + \text{Rand}(t) \times a(t) \times \left| mbest(t) - X_{ij}(t) \right| \\ \times \ln\left(\frac{1}{u_{ij}(t)}\right), \end{aligned} \tag{11}$$

where $f_{ij}(t) = \text{rad}(\,)$ and $u_{ij}(t) = \text{rad}(\,)$; the function $\text{rad}\, f(\,)$ produces a random number that is uniformly distributed between [0, 1]; the function $\text{Rand}(t)$ in equation (11) is given as follows:

$$\text{Rand}(t) = \begin{cases} -1, & \text{if: rad } f(\,) < 0.5, \\ +1, & \text{if: rad } f(\,) > 0.5. \end{cases} \tag{12}$$

M is the total number of particles; D is the dimension of the particle; $RP_{ij}(t)$ is the random point between the individual optimal position of the i^{th} particle at the j^{th} dimension $P_{ij}(t)$ and the global optimal position at the j^{th} dimension $G_j(t)$; $\varphi_j(t)$ is the random number between [0, 1]; and $a(t)$ represents the contraction-expansion coefficient of QPSO, which is an important parameter of convergence. The value of $a(t)$ depends on the situation, which can be dynamically changed in a certain way. The variable $a(t)$ can be determined according to the following formula:

$$a(t) = m - (m - n) \times \frac{1}{\text{MaxTimes}}. \tag{13}$$

The m and n in equation (13) usually are set as 1 and 0.5, and the MaxTimes represent the maximum number of iterations.

The flow of the QPSO algorithm is stated in detail as follows:

(1) Randomly initialize the initial position of M particles $X_i(0)$, and set the current best position of each particle as $P_i(0) = X_i(0)$; the initial global optimal value is the minimum value of the local optimal values of all current particles (assuming to find the minimum value of the function as an example), which is $P_g(0) = \min\{X_1(0), X_2(0), \ldots, X_D(0)\}$

(2) Calculate the fitness value of particles according to object function $f[X_i(t)]$

(3) Update current position of each particle $X_i(t)$ based on the calculated current fitness value

(4) Compare the last individual optimal values $P_i(t-1)$ with current position of each particle $X_i(t)$ to update current individual optimal position of each particle $P_i(t)$ through equation (8) and update current global optimal position $G(t)$ through equation (9)

(5) Compare current global optimal position $G(t)$ with last global optimal position $G(t-1)$, if $f[G(t)] < f[G(t-1)]$, $G(t)$ updates as the global optimal position; otherwise, $G(t-1)$ remains the global optimal position (for minimization)

(6) Calculate mean best position $mbest(t)$ according to equation (7)

(7) For each dimension of the particle, calculate random point $RP_{ij}(t)$ between $P_{ij}(t)$ and $G_j(t)$ according to equation (6)

(8) Update new position of each particle $X_i(t+1)$ based on equations (10) and (11)

(9) Repeat Step 2 to Step 8, until iteration MaxTimes is met

3. Experimental System

3.1. Specimen and Solution. Specimens fabricated from the sheet of Q235A pipelines were used, and the chemical composition in wt.% is shown in Table 1. Q235A steel is Class A ordinary carbon structural steel with a yield of strength of 235 MPa and a carbon content of 0.13~0.22%, which corresponds to Cr. D in America, 080A15 in UK, and E235B in ISO [37]. The specimens of 20 mm × 20 mm × 7 mm in size were adopted, and all the surfaces were subsequently grounded with 80, 240, 400, 600, 800, 1000, 1500, 2000, and 5000 sandpaper and cleaned by acetone and anhydrous ethanol. The epoxy resin was utilized to isolate the contact of the other five faces with the NaCl solution, and a working surface of 20 mm × 20 mm was left for the stray current corrosion.

In this paper, the accelerated simulation corrosion was conducted by the experimental system shown in Figure 2. A DC power supply is employed to apply the excitation of 0.0125 A/cm^2, 0.025 A/cm^2, 0.05 A/cm^2, and 0.075 A/cm^2, respectively. The input current is chosen based on the previous field test in a DC transit system with a directly grounded scheme, which is conducted through a non-contact fibre-optic current sensor embedded in the flange joint of the buried pipeline [37]. When the stray current corrosion was conducted for 10 min, the Tafel curve was tested until a total corrosion time of 60 min is reached. In this experimental system, the time of each corrosion interval is realized through the switch S1. A system of three electrodes was used in the corrosion cell. The preparation of NaCl solution was carried out using the

Table 1: Element composition of Q235A specimen.

C	Si	Mn	P	S	As	Alt	V	Nb	Ti	Cu	Ni	Cr	Mo	B	N	Ceq
10^{-2}	10^{-2}	10^{-2}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-2}	10^{-2}	10^{-2}	10^{-4}	10^{-4}	10^{-4}	10^{-2}
13	19	49	28	19	10	1	2	2	1	1	1	3	1	1	37	22

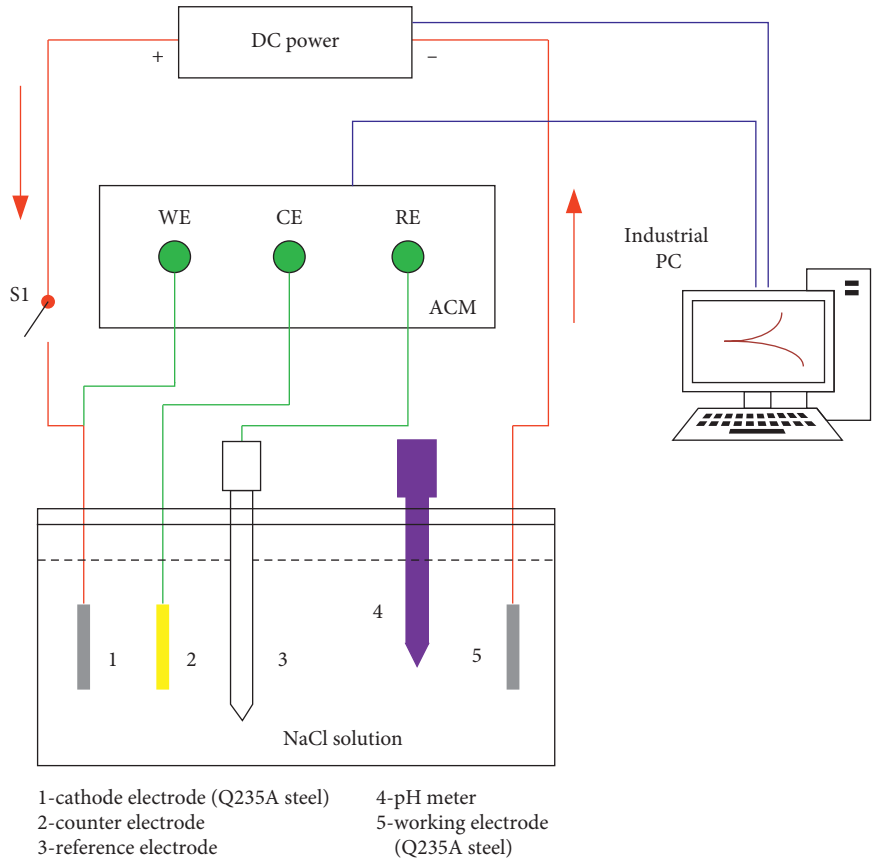

Figure 2: Experimental system of electrochemical corrosion under the coupling excitation of stray current and Cl^- ion.

chemically analytical grade reagents, the pH of which is 7.1. The concentration of NaCl solution is 0.05 mol/L, 0.1 mol/L, 0.2 mol/L, and 0.3 mol/L, respectively. The redox characteristic was changed by adding $FeCl_3$ to the solution.

The counter electrode used in this experiment is a 4 cm^2 Ti electrode, and the reference electrode is a saturated calomel electrode (SCE). The potential values shown here and in the next are of the relative SCE. There is a maintained gap of 15 mm between the counterelectrode and counterelectrode during each measurement. A CHI604e electrochemical workstation was employed as the electrochemical measurement instrument. During electrochemical testing, the test loop is independent of the applied DC loop. Each time before the testing of Tafel polarization curves, the open-circuit potential (OCP) needs to be measured to ensure scanning range. The potential scanning direction in the Tafel method is from negative to positive. The potential range of each scan is based on the testing result of OCP. The scanning rate utilized in the Tafel method was set to 2 mV/s. All experiments were performed in an indoor environment at 20°C. The solution remained undisturbed during polarization curve measurement.

3.2. Experimental Results and Database. In the case of excitation amplitude of 0.025 A/cm^2, the polarization curve under stray current corrosion in the initial oxidation-reduction potential of 230 mV and 0.2 mol/L NaCl is shown in Figure 3. The complete cathode and anode processes are exhibited on the polarization curve in Figure 3 during the stray current corrosion of different electrolyte concentrations. Taking the curve of 10 min as an example, there is an exponential increase of i_{corr} due to the oxidation reaction of steel electrode in the anodic polarization zone, which can be explained by Bulter–Volmer equation. The Bulter–Volmer equation is given in equation (14), where I is the polarization current, I_a is the anodic current, I_c is the cathodic current, I_{corr} is the corrosion current, E represents the applied potential, E_{corr} represents the corrosion potential, β_a is the anode Tafel slope, and β_c is the cathode Tafel slope. Under this circumstance, the corrosion current density measured at different corrosion time is illustrated in Figure 4. The

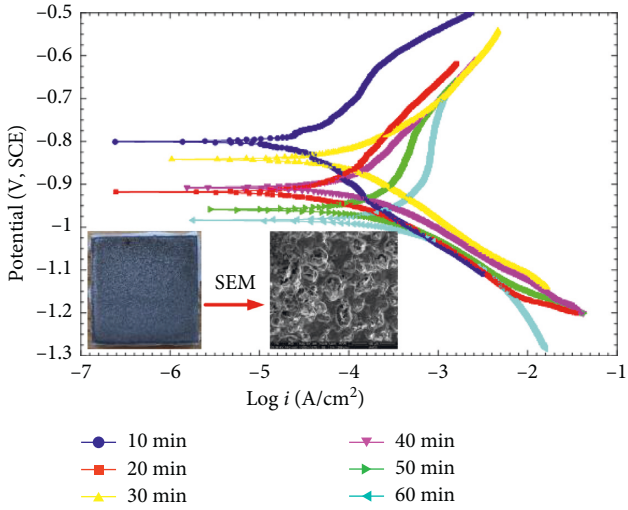

Figure 3: Experimental results of stray current corrosion on Q235A steel test samples.

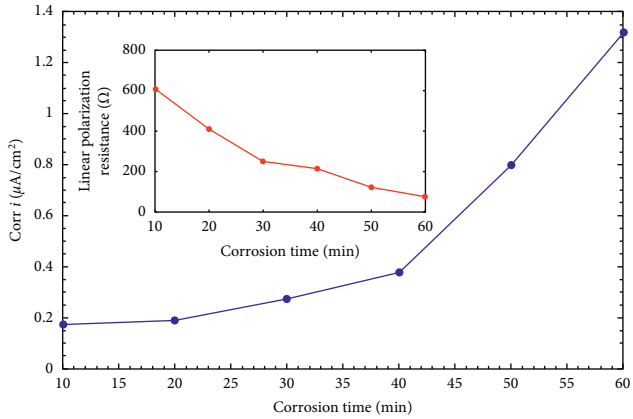

Figure 4: Electrochemical parameters during stray current corrosion on Q235A steel test samples.

corrosion current density i_{corr} increases nonlinearly with the increasing corrosion time, while the linear polarization resistance decreases:

$$I = I_a - I_c = I_{corr}\left[\exp\left(\frac{E - E_{corr}}{\beta_a}\right) - \exp\left(\frac{E - E_{corr}}{\beta_c}\right)\right]. \tag{14}$$

The experimental database is obtained from the electrochemical corrosion experiment stated above. The stray current density I_s, corrosion time t_{corr}, chloride concentration Cl%, and oxidation-reduction potential ORP were set as input valuables. The corrosion current density i_{corr} was set as output valuable. The correlation coefficient values between input variables and i_{corr} were calculated to demonstrate that there is a strong correlation between these parameters and i_{corr}, as shown in Figure 5. The correlations were investigated using Pearson's, Spearman's, and Kendall's correlation coefficients. The absolute correlation coefficients of four input variables range from 0.6071 to 0.8680,

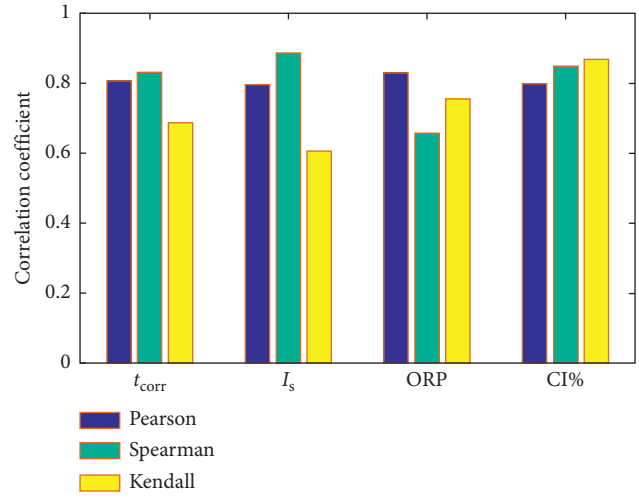

Figure 5: Correlation coefficient between input variables and i_{corr}.

which indicates that these parameters are of great significance in predicting the i_{corr} in the QPSO-NN model.

4. Corrosion Current Density Prediction Framework

4.1. Overall Framework and Algorithm Description. The whole prediction algorithm can be divided into preparation phase, optimization phase, training phase, and prediction phase. The backpropagation (BP) learning is to generate a final prediction model. The QPSO algorithm is employed to search for the suitable connection weights and biases for the initial structure of the neural network. The framework of the QPSO-NN method for predicting corrosion current density is shown in Figure 6. In the QPSO-NN algorithm, the NN training process is embedded in the fitness calculation function for processing. The NN training phase is continuously cycled throughout the prediction algorithm. The algorithm flow is given specifically as follows:

(1) Prepare the training and validation data sets and set the training ratio, validation ratio, and testing ratio in data normalization process

(2) Initialize the input weights, output weights, and hidden biases of NN structure, establish the initial NN model, and initialize the vector containing optimization object

(3) Update particles according to the QPSO evolution equations (10) and (11), which has been stated in Section 2

(4) Update the parameters for each particle according to the global optimal solution *gbest* in the QPSO algorithm

(5) When the weights and the hidden biases are updated, the NN model will be correspondingly updated with new parameters

(6) Calculate the fitness according to the prediction results with the training data sets. (the fitness

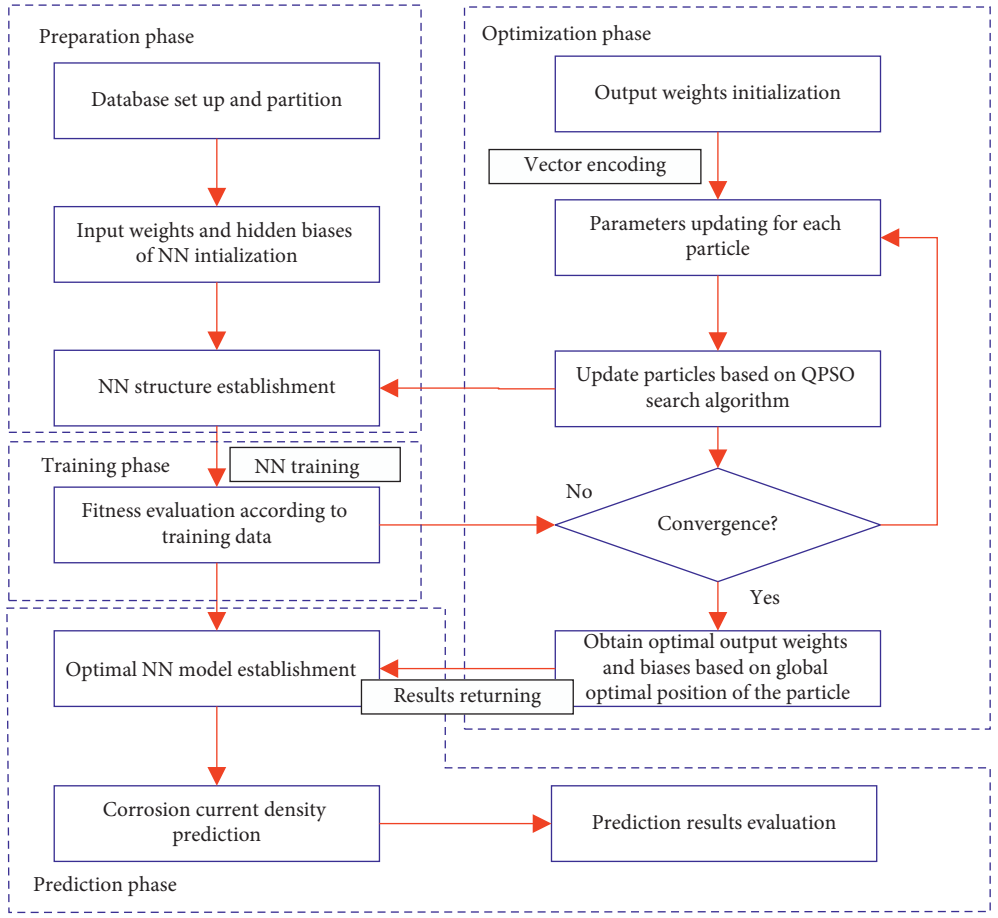

Figure 6: Structure of the ANN and encoding details in QPSO.

calculation method will be explained detailedly in equation (16))

(7) Judge whether meeting end of criterion, if yes, obtain the optimal input weights, output weights, and the hidden biases; if not, jump to step 3

(8) Establish the optimal NN model based on the optimization results returning from QPSO (optimized weights and biases)

(9) Predict corrosion current density based on the built QPSO-NN model

(10) Evaluate prediction results of corrosion current density

4.2. Technical Details. The ratio of training set, validation set, and testing set were preliminarily set as 70%, 15%, and 15%, respectively. The error minimization tolerance was set to 1%. The ANN training target was set to MSE of 0.01. The establishment, optimization, training, and testing of the QPSO-NN were carried out in the environment of MATLAB R2015b. The parameters of the QPSO-NN are given in Table 2.

The input weights, hidden biases, output weights, and output biases are the optimization target of the QPSO algorithm. The details of the QPSO-NN algorithm are shown in Figure 7. The number of hidden nodes m remains to be determined in the next section.

Table 2: Parameters of the QPSO-ANN algorithm.

QPSO	
Population size	80
Max iterations	300
Contraction-expansion coefficient	0.5
ANN	
Number of input neurons	4
Number of output neurons	1
Transfer function in hidden layer	*Tansig*
Transfer function in output layer	*Purelin*
Learning rate	0.01
Training times	300

(1) Encoding for particle in QPSO and pretreatment for data set in the NN: the structure of the ANN model is shown in Figure 5. A position vector of particle p_j is built up to store the parameter information of the ANN. The vector consists of four parts: input weights between input layer and hidden layer: $\{\omega_{ih(1,1)}, \ldots, \omega_{ih(4,1)}, \ldots, \omega_{ih(1,m)}, \ldots, \omega_{ih(4,m)}\}$, hidden biases on hidden layer: $\{\theta_{1,1}, \theta_{1,2}, \ldots, \theta_{1,m}\}$, output weights between hidden layer and output layer:

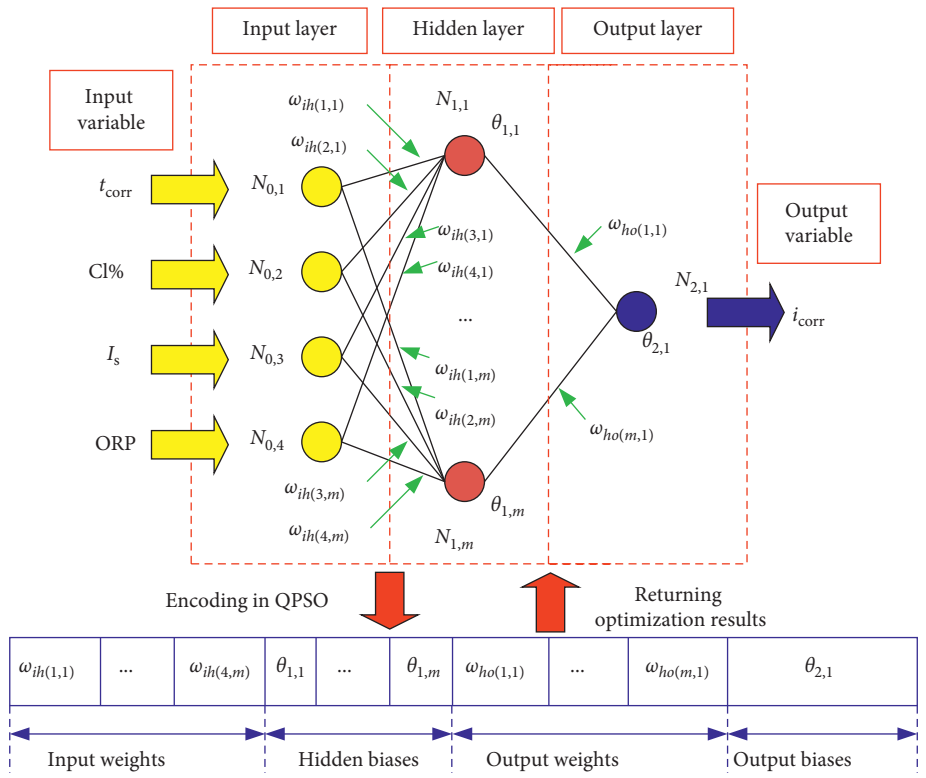

Figure 7: Structure of the ANN and encoding details in QPSO.

$\{\omega_{ho(1,1)}, \omega_{ho(2,1)}, \ldots, \omega_{ho(m,1)}\}$, and output bias on output layer: $\theta_{2,1}$.

(2) Data normalization: before the training and optimization process in the QPSO-NN model, the data set needs to be normalized to induct statistical distribution of unified samples and eliminate the dimensional impact between input indicators. In this model, linear min-max normalization was employed. Taking the input parameter stray current density I_s as an example, the I_s^i is mapped to I_s^{i*} through equation the following equation:

$$I_s^{i*} = \frac{I_s^i - I_{\min}}{I_{\max} - I_{\min}}, \tag{15}$$

where $I_{\min}$ and $I_{\max}$ is the minimum and maximum of stray current density I_s in the input samples.

(3) Optimization process in QPSO: in general, two termination conditions can be chosen: the max generation $G_{\max}$ is met or the deviation between adjacent generations is within a specified range [38]. In this model, the max generation $G_{\max}$ is utilized as the termination condition. The particle p_m $(1 \le m \le G_{\max})$ is updated for each searching process until the $G_{\max}$ is reached, as shown in Figure 6.

(4) Fitness function in the QPSO-NN model: when a search process is finished and a particle is updated with the determined weights and biases, the fitness of this particle p_j needs to be calculated to evaluate the performance of these structure parameters of the ANN. In this model, the fitness function is defined as equation (16). The calculation results of fitness will be discussed in Section 5 to determine the best topological structure of the ANN:

$$\text{fitness}(p_j) = \sum_{i=1}^{\text{batch}} \text{abs}(I_i - I_i'), \tag{16}$$

where I_i is the real value of i_{corr}, I_i' is the predicted value of i_{corr}, and *batch* is the number of training set during one training session.

5. Corrosion Current Density Prediction and Results Analysis

5.1. Training Process. In this section, the effectiveness of the QPSO-NN is validated through experiments on the experimental data. The work in this paper is mainly to improve the performance of the traditional NN for the i_{corr} prediction problem in the process of stray current corrosion. Hence, the specific contrast between the QPSO-NN algorithm and the basic PSO-NN algorithm and the proposed QPSO-NN algorithm is indispensable in the next section. In the experiments, the QPSO-NN and the PSO-NN were implemented on MATLAB software, and the prediction process of the BPNN was realized by using the toolbox in the MATLAB. There is one more important parameter that needs to be determined, which is the number of hidden nodes in the NN. The number of hidden nodes in the NN is selected according to the calculation results. Figure 8 gives the value of fitness

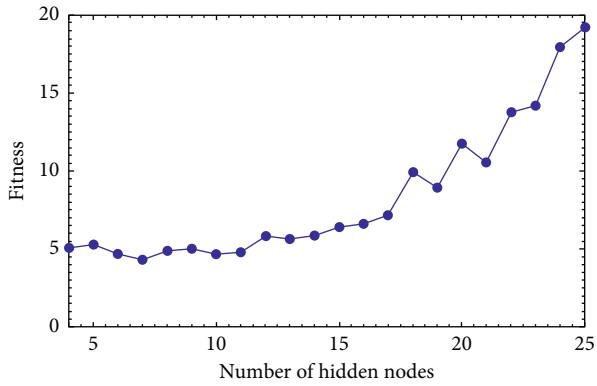

Figure 8: Fitness versus number of hidden nodes in the ANN.

function corresponding to the number of hidden nodes. It can be observed from Figure 6 that the lowest validation is achieved, when the number of hidden nodes is 7. Thus, the basic topological structure of the QPSO-NN is determined as 4-7-1.

The experiments were performed on a computer running 64 bits Windows 10 system. In order to perform quantitative comparison, the following indexes are used to measure the performance of each prediction model, such as the precision, mean absolute error (MAE), mean square error (MSE), root mean square error (RMSE), proportional error (PERR), and coefficient of determination:

$$\begin{aligned}
\text{precision} &= \frac{\left|I_i' - I_i\right|}{I_i} \times 100\%, \\
\text{MAE} &= \frac{\sum_{i=1}^{n}\left|I_i' - I_i\right|}{n}, \\
\text{MSE} &= \frac{\sum_{i=1}^{n}\left(I_i' - I_i\right)^2}{n}, \\
\text{RMSE} &= \sqrt{\frac{\sum_{i=1}^{n}\left(I_i' - I_i\right)^2}{n}}, \\
\text{PERR} &= \frac{\sum_{i=1}^{n}\left(I_i' - I_i\right)^2}{\sum_{i=1}^{n} I_i^2}, \\
R^2 &= 1 - \frac{\sum_{i=1}^{n}\left(I_i' - I_i\right)^2}{\sum_{i=1}^{n}\left(I_i - \overline{I}\right)^2},
\end{aligned} \tag{17}$$

where n is the number of samples; $i \in [1, n]$; I_i' is the predicted value; I_i is the measured value; and $\overline{I}$ is the mean value of I_i.

5.2. QPSO-NN vs. BPNN and PSO-NN. Comparative study needs to be conducted first to determine the proposed model for corrosion current density is effective. While comparing the QPSO-NN with the BPNN and PSO-NN, 100 runs are repeated to get the average value of prediction results. Based on the prediction results in Table 3, a higher prediction accuracy of the QPSO-NN is always observed regardless of the proportion of training set, compared to the BPNN and PSO-NN. There is about 7 to 9 percent difference in prediction accuracy between BPNN and QPSO-NN, while the difference between the PSO-NN and the QPSO-NN is about 5 to 6 percent. The biggest difference in accuracy between the BPNN and the QPSO-NN is 9.39%, while the biggest difference between the PSO-NN and the QPSO-NN is 6.64%. To quantitatively verify the advantage of the QPSO-NN over traditional algorithms, an analysis of variance (ANOVA) test is employed for evaluating the difference among the BPNN, PSO-NN, and QPSO-NN. As can be seen from the 6$^{\text{th}}$ and 8$^{\text{th}}$ column in Table 4, the P values of the data column among the BPNN, PSO-NN, and QPSO-NN in all cases of the proportion of all training set are far less than 1×10^{-4}. Thus, a conclusion can be drawn from Table 4 that the proposed QPSO-NN model is obviously more applicable than the PSO-NN and BPNN for this prediction problem.

Although the precision of the QPSO-NN model is proved to be better than the PSO-NN and the BPNN model through the general statistics analysis like mean value, the comparison conducted above is rough. Furthermore, a deeper analysis is indispensable to be carried out. In this analysis, the accuracy distribution of each algorithm is conducted after running 100 times. The corresponding results of the BPNN, PSO-NN, and QPSO-NN with six different proportions of training set are illustrated in Figure 9. With the ratio of the training set increasing from 40% to 80% in Figures 9(a)–9(e), the distribution range of QPSO-NN's accuracy in 100 trials is smaller than that of QPSO-NN's and BPNN's. The QPSO-NN exhibits a better performance and higher stability than the PSO-NN and BPNN in the prediction problem of corrosion current density induced by stray current. Thus, the strategy of using QPSO instead of PSO to optimize neural networks is effective, which improves the accuracy of the prediction. In the case of ratio = 90%, the highest and lowest prediction is 98.32% and 81.22%, and the mean precision is 91.95%.

Next, the effect on prediction precision due to different proportions of training set is analysed in this section. The effect of ratio on the precision results is shown in Figure 10. No matter the population size is 60 or 80, the prediction precision of the QPSO-NN, PSO-NN, and BPNN gradually increases with the increasing proportion of the training set. With population size of 60, the precision increases from 88.60% to 90.77%. The precision increases from 89.51% to 91.95% when the population size is 80. Thus, the precision is improved by 2.17% and 2.44%, which indicates that the ratio of training set has little influence on the prediction accuracy.

When the ratio of training set is 80% and parameters are used as given in Table 2, the prediction results of the QPSO-NN, PSO-NN, and BPNN are intuitively illustrated in Figure 11. The testing set contains 50 samples. The precision of three algorithms is 82.86%, 85.38%, and 90.32%, respectively. Compared with the BPNN and PSO-NN, the predicted results of the PSO-NN are more consistent with the measured results. The evaluation indexes MAE, MSE, and RMSE are shown in Table 4. The MAE, MSE, RMSE, and PERR of prediction results in the QPSO-NN model are smaller than those in the BPNN model and PSO-NN model, which also indicates the prediction accuracy of the QPSO-

Table 3: Comparison on prediction among the QPSO-NN, PSO-NN, and BPNN.

Ratio of training set (%)	Prediction precision and ANOVA test						
	BPNN (x) (%)	PSO-NN (y) (%)	QPSO-NN (z) (%)	z–y (%)	P value	z–x (%)	P value
40	80.12	82.87	89.51	6.64	8.65×10^{-7}	9.39	4.13×10^{-13}
50	80.23	84.03	89.65	5.62	1.12×10^{-7}	9.42	1.39×10^{-9}
60	81.26	84.26	90.11	5.85	1.03×10^{-7}	8.85	1.68×10^{-11}
70	82.75	85.10	90.23	5.13	5.99×10^{-8}	7.48	1.22×10^{-11}
80	83.33	85.80	91.34	5.54	3.49×10^{-5}	8.01	1.26×10^{-11}
90	83.41	86.02	91.95	5.93	1.67×10^{-6}	8.54	6.29×10^{-18}

Table 4: Evaluation indexes on prediction results.

Error	BPNN	PSO-NN	QPSO-NN
MAE	0.0837	0.0720	0.0454
MSE	0.0110	0.0097	0.0039
RMSE	0.1047	0.0983	0.0624
PERR	0.5478	0.4830	0.1949

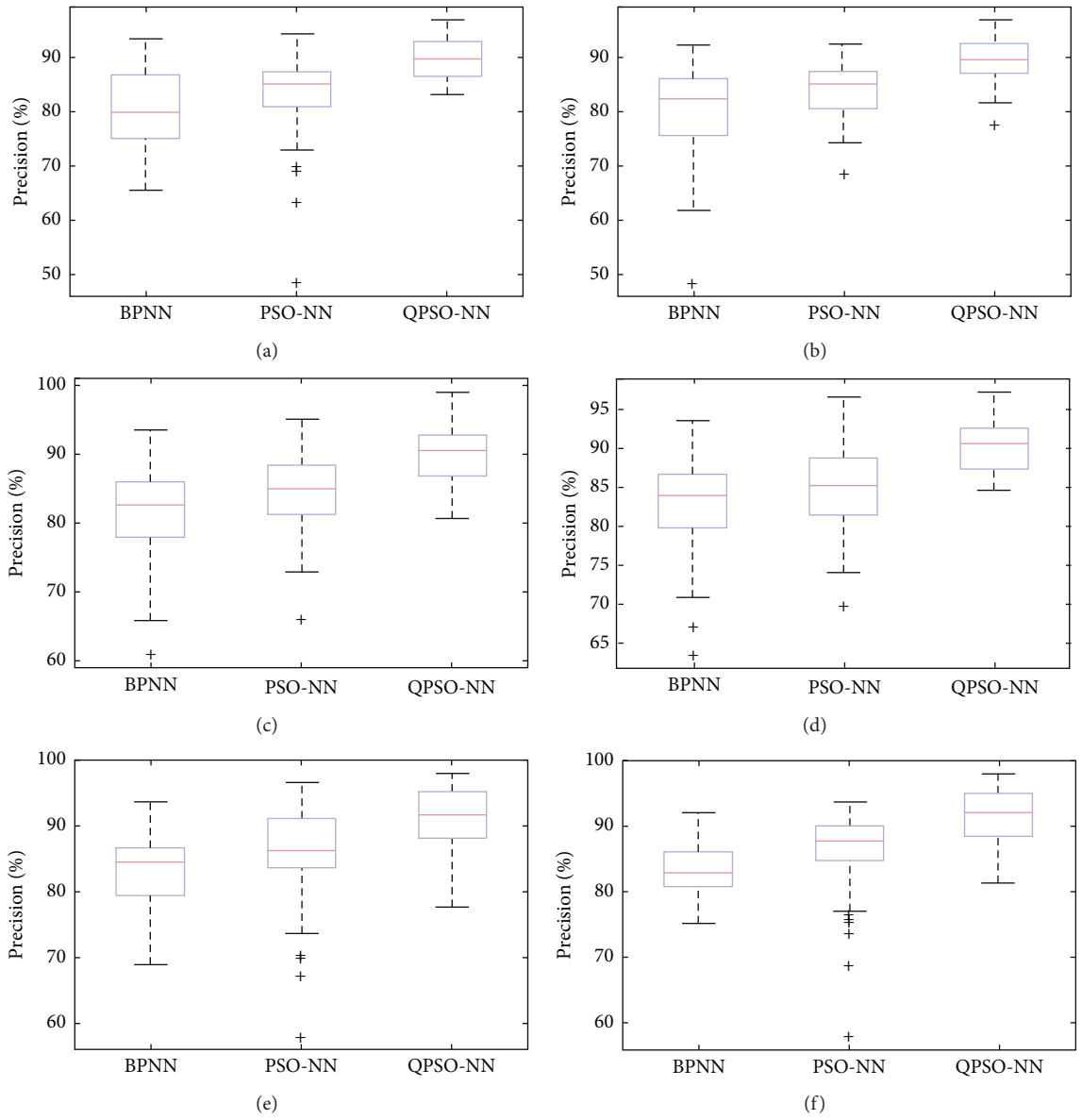

Figure 9: Precision distributions of three algorithms in different training set ratios: (a) 40%; (b) 50%; (c) 60%; (d) 70%; (e) 80%; (f) 90%.

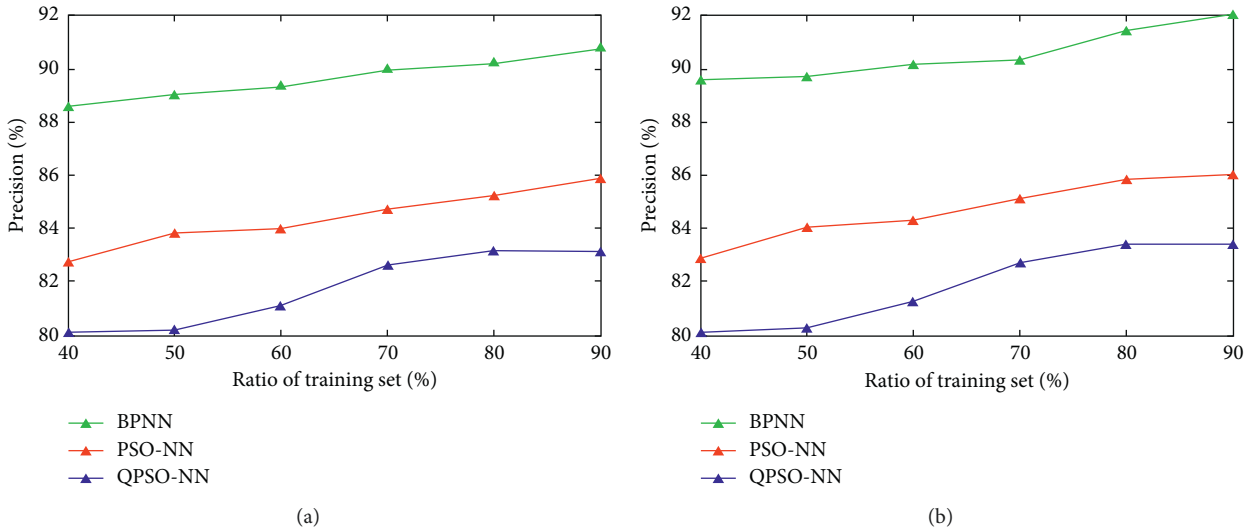

Figure 10: The impact on precision caused by ratio of training set. (a) Population size = 60; (b) population size = 80.

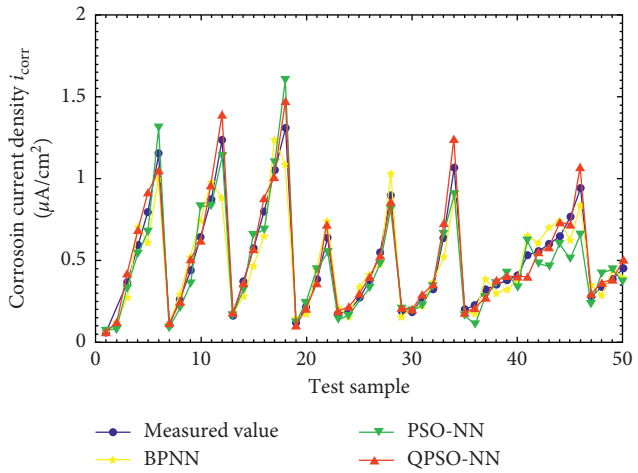

Figure 11: Prediction results of the QPSO-NN model, PSO-NN model, and BPNN model.

NN is better than the BPNN and PSO-NN from another point of view.

The regression lines between tested and predicted values of three algorithms are displayed in Figure 12. The correlation coefficient of determination R^2 is calculated so as to test how well the model fits the sample observations. The model is considered to be more successful when the R^2 is close to 1. The correlation coefficient of determination for the BPNN, PSO-NN, and QPSO-NN is 0.9440, 0.9552, and 0.9867, respectively. The regression lines are given in equations (18)–(20). The results of correlation coefficient also demonstrate that the QPSO-NN model shows better accuracy than the PSO-NN model and BPNN model:

$$\text{Regression Line 1}: y = 0.8826x + 0.0425, \quad (18)$$

$$\text{Regression Line 2}: y = 0.9158x + 0.0575, \quad (19)$$

$$\text{Regression Line 3}: y = 1.0760x - 0.0222. \quad (20)$$

Based on the comparison analysis from multiple aspects in this section, a higher precision is achieved by the QPSO-NN model regardless of the ratio of the training set. In terms of computational efficiency, the QPSO-NN model costs a relatively longer time than the BPNN due to the optimization process of the QPSO algorithm. Since the corrosion is a long-term process, real-time performance is not a major factor in corrosion current density prediction, which means the offline analysis can be performed based on the collected data. Therefore, the model based on the QPSO-NN algorithm is feasible and more resultful for this prediction problem.

5.3. Analysis of Training Times of NN. The parameter of training times represents the times of iterations for optimizing the configuration of a neural network during the training process [39], which affects the performance of the neural network. To investigate its effect on the prediction accuracy, the corrosion current density prediction under two different proportions of training set (60% and 80%) is conducted with varied training times of the NN. The results are illustrated in Figure 13.

Based on the results in Figure 13, with the ratio of training set = 60%, the average precision is 87.40% when the training times of the NN is 100, and the average precision is 90.44% when the training times of the NN is 400. When the training times of the NN is 100 and 400, the average precision is 88.13% and 91.62%, with the ratio of training set = 80%, respectively. In both cases, the average precision is improved by 3.04% and 3.49%. Thus, the training times of the NN have a little impact on the precision. However, there is no further increase in accuracy when the training times of the NN is bigger than 300, as can been seen from Figures 13(a) and 13(b). When the training times of the NN is bigger than 300, the increase of training times will not significantly improve the prediction accuracy but increase the algorithm runtime.

In terms of distribution of precision, the mean precision is looser when the training times of the NN is small. With the

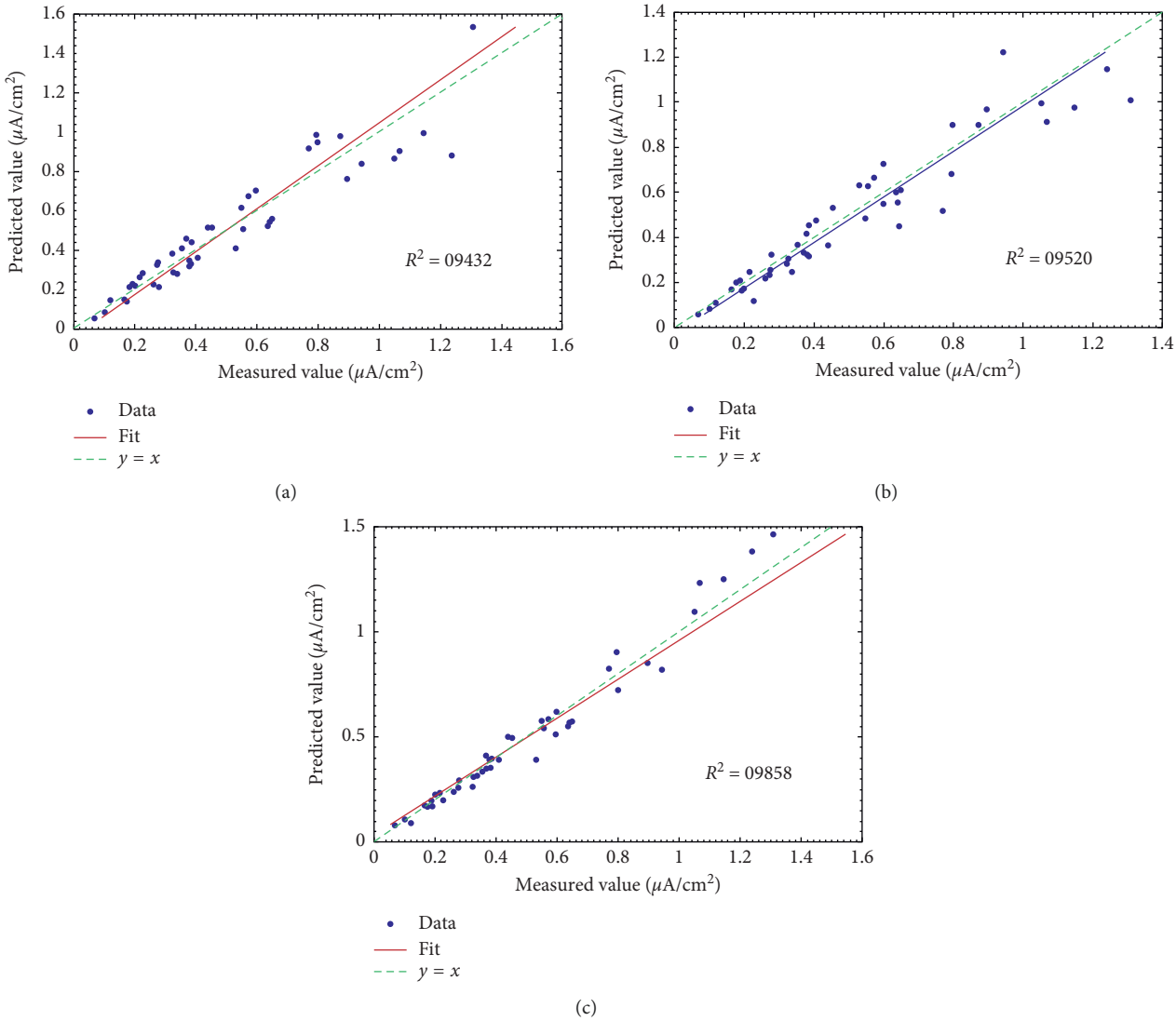

Figure 12: Regression lines between measured values and predicted values of three algorithms. (a) BPNN model; (b) PSO-NN model; (c) QPSO-NN model.

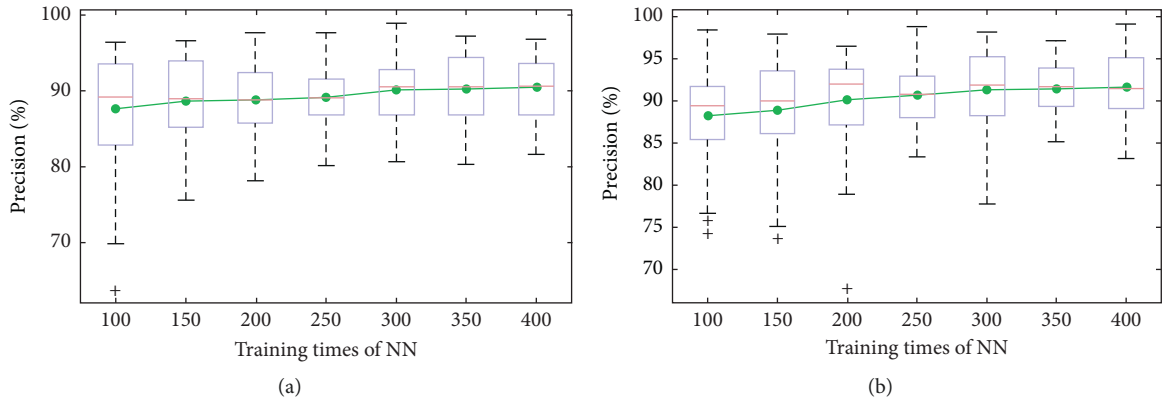

Figure 13: Prediction precision versus training times of NN. (a) Ratio of training set = 60%; (b) ratio of training set = 80%.

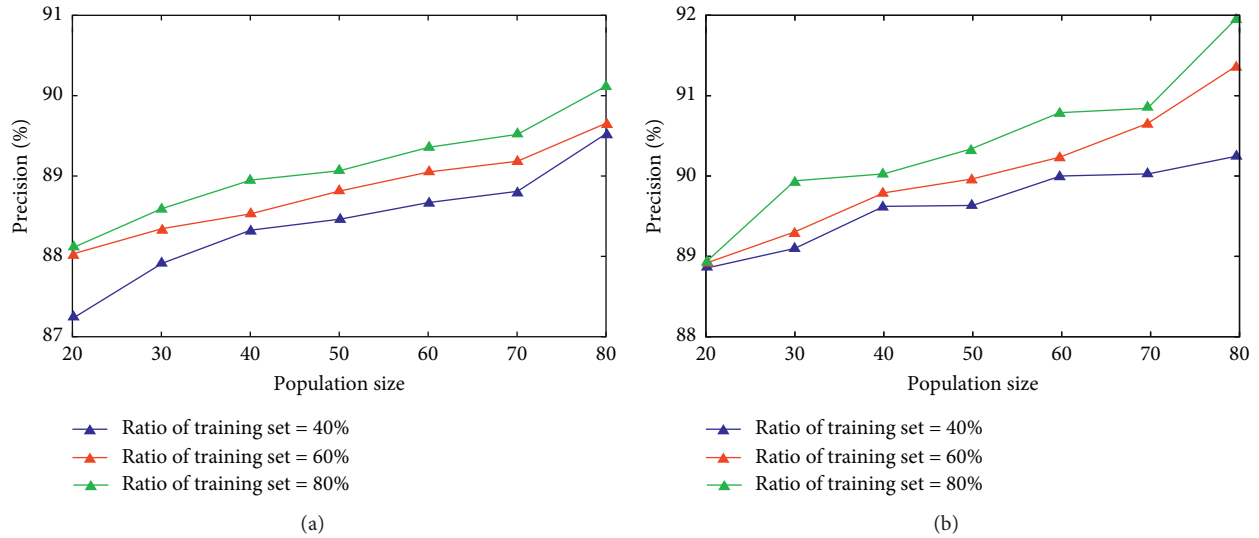

Figure 14: Prediction precision versus population size of QPSO. (a) Ratio of training set = 40%, 60%, and 80%; (b) ratio of training set = 30%, 50%, and 70%.

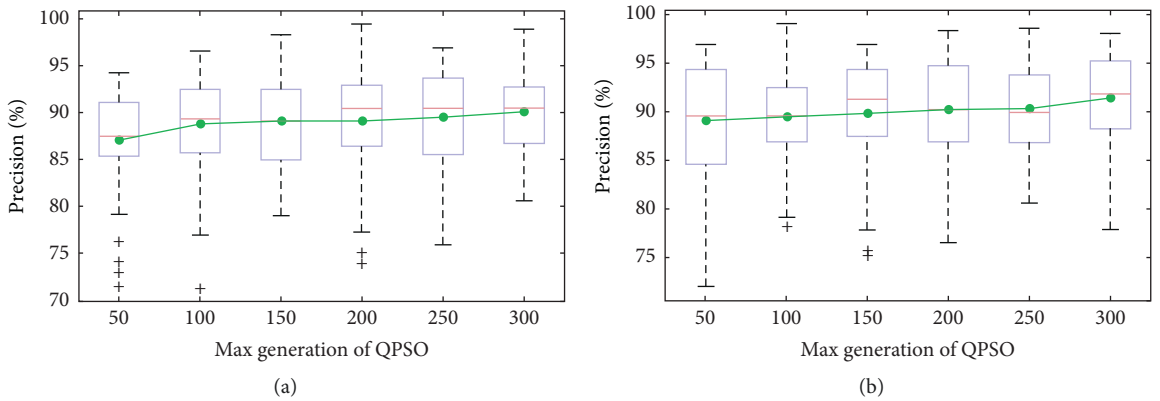

Figure 15: Prediction precision versus max generation of QPSO. (a) Ratio of training set = 60%; (b) ratio of training set = 80%.

increase in training times, the distribution of mean prediction is more concentrated, which indicates that the algorithm is more stable.

5.4. Analysis of Population Size. Generally speaking, the population size of whole particle is a significant parameter that has an impact on the convergence speed, accuracy, and stability of QPSO. In order to evaluate its impact on the QPSO-NN, the population size varies from 20 to 80, and then the results corresponding to each parameter are compared for difference.

As can be seen from Figure 14, the prediction accuracy of the QPSO-NN increases rapidly with the increasing population size no matter how much the proportion of the training set is. Furthermore, the mean precision is compared when the ratio of training set is 80% and 90%. In Figure 14(a), the average precision is 88.92% when the population size is 20 and the ratio of training set is 80%, while the average precision is 91.34% when the population size is 80 and the ratio of training set is 90%. In Figure 14(b), the average accuracy is 88.91% when the population size is 20, and the average precision is 91.95% when the population size is 80. When the population size varies from 20 to 80 in both cases, the average precision is increased by 2.42% and 3.04%.

5.5. Analysis of Max Generation QPSO. The max generation of QPSO affects the optimization process to a large extent, which determines the effectiveness and efficiency of the search process. Therefore, it is essential to study the impact of the maximum generation G_{max} on the QPSO-NN. In this section, the max generation of QPSO was examined from 50 to 300, in the step of 50, with the ratio of training set = 60% and 80%, respectively. The prediction results are shown in Figure 15.

It can be seen from Figure 14 that QPSO-NN's mean precision increases slowly with the increasing max generation QPSO. In Figure 15(a), the average precision is 87.31% when $G_{max} = 50$, while the average precision is 90.10% when

G_{max} = 300. Under this circumstance, the prediction accuracy is improved by 2.79%. In Figure 15(b), the average precision is 89.16% when G_{max} = 50, while the average precision is 91.34% when G_{max} = 300. The prediction accuracy is improved by 2.18% on this occasion. Thus, the increasing max generation of QPSO makes little contribution to the improvement of precision.

In addition, the mean precision is relatively loose-distributed in the low G_{max}. When the G_{max} increases, the distribution of mean precision becomes narrower, which means the prediction performance is more stable than the lower G_{max}. However, the distribution of mean precision does not stay monotonically increasing over the G_{max} but the trend is increasing.

6. Conclusions

In this paper, a QPSO-NN model was proposed to predict the corrosion current density i_{corr} under the coupling effect of stray current and Cl^- ion, in which the corrosion time t_{corr}, oxidation-reduction potential ORP, stray current density $I_{s,}$ and chloride ion concentration Cl% were set as input variables. The database was generated from a stray current electrochemical experiment. The QPSO-driven neural network was compared with the BPNN model and PSO-NN model and proved to be with better accuracy. Compared to the BPNN model and PSO-NN model, the prediction accuracy of the QPSO-NN model is more stable. The best average prediction accuracy of the QPSO-NN during 100 trials is 91.95%.

The proposed model indicates that the QPSO-NN model exhibits a theoretical value in the prediction of corrosion current density i_{corr} under the coupling effect of stray current and Cl^- ion. The max generation of QPSO, training times of the NN, ratio of training set, and population size of QPSO show little impact on the precision of prediction accuracy. The model built up provided the possibility to monitor the corrosion status of buried metallic pipeline based on the information measured without excavation.

The proposed model is designed for the prediction of corrosion current density for Q235A steel. However, buried pipelines contain a variety of metals, stray current corrosion characteristics of which are different. Future work will concentrate on predicting the corrosion current density under the coupling action of stray current and multiple ions regardless of the type of buried pipeline metal. To establish the prediction model dealing with different pipeline metals, new database has to be created through extra electrochemical experiments.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors acknowledge the project funded by the National Natural Science Foundation of China (NSFC) (51607178); special project supported by the China Postdoctoral Science Foundation (2018T110570); and project funded by the China Postdoctoral Science Foundation (2019M652005), the Natural Science Research Project of Jiangsu Higher Education Institutions (18KJB460003), and Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) for the financial support to this research.

References

[1] W. Li and X. Yan, "Research on integrated monitoring and prevention system for stray current in metro," *International Journal of Mining Science and Technology*, vol. 11, no. 2, pp. 221–228, 2001.

[2] F. C. R. Hernández, G. Plascencia, and K. Koch, "Rail base corrosion problem for North American transit systems," *Engineering Failure Analysis*, vol. 16, no. 1, pp. 281–294, 2009.

[3] Z. Chen, C. Qin, J. Tang, and Y. Zhou, "Experiment research of dynamic stray current interference on buried gas pipeline from urban rail transit," *Journal of Natural Gas Science and Engineering*, vol. 15, pp. 76–81, 2013.

[4] L. Bertolini, M. Carsana, and P. Pedeferri, "Corrosion behaviour of steel in concrete in the presence of stray current," *Corrosion Science*, vol. 49, no. 3, pp. 1056–1068, 2007.

[5] A. Susanto, D. A. Koleva, O. Copuroglu, K. van Beek, and K. van Breugel, "Mechanical, electrical and microstructural properties of cement-based materials in conditions of stray current flow," *Journal of Advanced Concrete Technology*, vol. 11, no. 3, pp. 119–134, 2013.

[6] M. R. Dann and M. A. Maes, "Stochastic corrosion growth modeling for pipelines using mass inspection data," *Reliability Engineering & System Safety*, vol. 180, pp. 245–254, 2018.

[7] S. Y. Xu, W. Li, and Y. Q. Wang, "Effects of vehicle running mode on rail potential and stray current in DC mass transit systems," *IEEE Trans. Veh. Technol.*vol. 62, no. 8, pp. 3569–3580, 2013.

[8] A. L. Cao, Q. F. Zhu, S. T. Zhang, and B. R. Hou, "BP neural network predictive model for stray current density of a buried metallic pipeline," *Anti-Corrosion Methods and Materials*, vol. 57, no. 5, pp. 234–237, 2010.

[9] N. Narimani, B. Zarei, H. Pouraliakbar, and G. Khalaj, "Predictions of corrosion current density and potential by using chemical composition and corrosion cell characteristics in microalloyed pipeline steels," *Measurement*, vol. 62, pp. 97–107, 2015.

[10] Y. Q. Wang, W. Li, S. Y. Xu, and X. F. Yang, "Prediction for corrosion status of the metro metal materials in the stray current interference," *International Journal of Electrochemical Science*, vol. 8, no. 4, pp. 5314–5329, 2013.

[11] S.-Y. Xu, W. Li, F.-F. Xing, and Y.-Q. Wang, "Novel predictive model for metallic structure corrosion status in presence of stray current in DC mass transit systems," *Journal of Central South University*, vol. 21, no. 3, pp. 956–962, 2014.

[12] Y. F. Wen, C. Z. Cai, X. H. Liu, J. F. Pei, X. J. Zhu, and T. T. Xiao, "Corrosion rate prediction of 3C steel under different seawater environment by using support vector regression," *Corrosion Science*, vol. 51, no. 2, pp. 349–355, 2009.

[13] M. Nikoo, Ł. Sadowski, F. Khademi, and M. Nikoo, "Determination of damage in reinforced concrete frames with

shear walls using self-organizing feature map," *Applied Computational Intelligence and Soft Computing*, vol. 2017, Article ID 3508189, 10 pages, 2017.

[14] F. Y. Ma and W. H. Wang, "Prediction of pitting corrosion behavior for stainless SUS 630 based on grey system theory," *Materials Letters*, vol. 61, no. 4-5, pp. 998–1001, 2007.

[15] W. Li and Y. Q. Wang, "Prediction of danger degree of corrosion induced by stray current for metro by using BP artificial neural network," *Corrosion Science and Protection Technology*, vol. 17, pp. 438–441, 2005.

[16] J. Arriagada, P. Olausson, and A. Selimovic, "Artificial neural network simulator for SOFC performance prediction," *Journal of Power Sources*, vol. 112, no. 1, pp. 54–60, 2002.

[17] M. Laguna and R. Martí, "Neural network prediction in a system for optimizing simulations," *IIE Transactions*, vol. 34, no. 3, pp. 273–282, 2002.

[18] M. Hosoz, H. M. Ertunc, and H. Bulgurcu, "Performance prediction of a cooling tower using artificial neural network," *Energy Conversion and Management*, vol. 48, no. 4, pp. 1349–1359, 2007.

[19] K. Ghorbanian and M. Gholamrezaei, "An artificial neural network approach to compressor performance prediction," *Applied Energy*, vol. 86, no. 7-8, pp. 1210–1221, 2009.

[20] M. Saadatseresht and M. Varshosaz, "Visibility prediction based on artificial neural networks used in automatic network design," *The Photogrammetric Record*, vol. 22, no. 120, pp. 336–355, 2007.

[21] G. Cammarata, S. Cavalieri, and A. Fichera, "A neural network architecture for noise prediction," *Neural Networks*, vol. 8, no. 6, pp. 963–973, 1995.

[22] M. Kamrunnahar and M. Urquidi-Macdonald, "Prediction of corrosion behavior using neural network as a data mining tool," *Corrosion Science*, vol. 52, no. 3, pp. 669–677, 2010.

[23] L. Sadowski, "Non-destructive investigation of corrosion current density in steel reinforced concrete by artificial neural networks," *Archives of Civil and Mechanical Engineering*, vol. 13, no. 1, pp. 104–111, 2013.

[24] X. H. Wu, Y. Ye, and H. Ren, "Prediction of the laws of carbon steel erosion corrosion in sour water system based on decision tree and two kinds of artificial neural network model," in *Proceedings of the 2016 Chinese Control and Decision Conference (CCDC)*, IEEE, Yinchuan, China, May 2016.

[25] D. F. Cook, C. T. Ragsdale, and R. L. Major, "Combining a neural network with a genetic algorithm for process parameter optimization," *Engineering Applications of Artificial Intelligence*, vol. 13, no. 4, pp. 391–396, 2000.

[26] F. H. F. Leung, H. K. Lam, S. H. Ling, and P. K. S. Tam, "Tuning of the structure and parameters of a neural network using an improved genetic algorithm," *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 79–88, 2003.

[27] S. C. Tan and C. P. Lim, "Evolving an adaptive artificial neural network with a gravitational search algorithm," *Intelligent Decision Technologies*, Springer International Publishing, Berlin Germany, 2015.

[28] Z. Lin, G. Chen, W. Guo, and Y. Liu, "PSO-BPNN_based prediction of network security situation," in *Proceedings of the International Conference on Innovative Computing Information & Control*, IEEE Computer Society, Dalian, China, June 2008.

[29] S. S. Gilan, H. B. Jovein, and A. A. Ramezanianpour, "Hybrid support vetor regression—particle swarm optimization for prediction of compressive strength and rcpt of concretes containing metakaolin," *Construction and Building Materials*, vol. 34, pp. 321–329, 2012.

[30] X. He, X. Gao, Y. Zhang, and Z. H. Zhou, "Fuzzy C-means based on cooperative QPSO with learning behavior," in *Proceedings of the 5th International Conference on Intelligence Science and Big Data Engineering*, vol. 9243, pp. 343–351, Suzhou, China, June 2015.

[31] S. Y. Li, R. G. Wang, W. W. Hu, and J. Q. Sun, "A new QPSO based BP neural network for face detection," in *Proceedings of the 2nd International Conference on Fuzzy Information and Engineering*, vol. 40, pp. 355–363, Guangzhou, China, May 2007.

[32] S. N. Omkar, R. Khandelwal, T. V. S. Ananth, G. Narayana Naik, and S. Gopalakrishnan, "Quantum behaved particle swarm optimization (QPSO) for multi-objective design optimization of composite structures," *Expert Systems with Applications*, vol. 36, no. 8, pp. 11312–11322, 2009.

[33] C. T. Cheng, W. J. Niu, J. J. Shen, and K. W. Chau, "Daily reservoir runoff forecasting method using artificial neural network based on quantum-behaved particle swarm optimization," *Water*, vol. 7, no. 8, pp. 4232–4246, 2015.

[34] D. O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*, John Wiley, Chapman & Hall, London, UK, 2013.

[35] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958.

[36] G. E. Hinton and T. J. Sejnowski, *Learning and Relearning in Boltzmann Machines in Explorations in the Microstructure of Cognition*, Vol. 19, The MIT Press, Cambridge, MA, USA, 1986.

[37] C. T. Wang, W. Li, Y. Q. Wang, S. Y. Xu, and X. F. Yang, "Chloride-induced stray current corrosion of Q235A steel and prediction model," *Construction and Building Materials*, vol. 219, pp. 164–175, 2019.

[38] C. T. Wang and W. Li, "Improved PSO-NN prediction model for corrosion current density under the stray current excitation based on data mining technique," in *Proceedings of the 2019 2nd International Conference on Data Science and Information Technology—DSIT 2019*, pp. 77–84, Seoul, Republic of Korea, July 2019.

[39] C. Mao, R. Lin, C. Xu, and Q. He, "Towards a trust prediction framework for cloud services based on PSO-driven neural network," *IEEE Access*, vol. 5, pp. 2187–2199, 2017.