IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 28, 2020, accepted January 4, 2021, date of publication January 12, 2021, date of current version January 21, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051169

A Link Quality Estimation Method Based on Improved Weighted Extreme Learning Machine

LINLAN LIU[1], HUI LV[1], JIANGBO XU[1], AND JIAN SHU[2]
[1]School of Information Engineering, Nanchang Hangkong University, Nanchang 330063, China
[2]School of Software, Nanchang Hangkong University, Nanchang 330063, China

Corresponding authors: Linlan Liu (liulinlan@nchu.edu.cn) and Jian Shu (shujian@nchu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61962037, Grant 62062050, and Grant 61762065; in part by the Natural Science Foundation of Jiangxi Province under Grant 20202BABL202039 and Grant 20181BAB202015; and in part by the Innovation Foundation for Postgraduate Student of Jiangxi Province under Grant YC2020032.

ABSTRACT The link quality of wireless sensor networks is the basis for selecting communication links in routing protocols. Effective link quality estimation is helpful to select high-quality links for communication and to improve network stability. The correlation of link quality parameter and packet reception rate (PRR) is calculated by the Pearson correlation coefficient. According to Pearson coefficient values, the averages of the link quality indication, received signal strength indication, and signal-to-noise are selected as the parameters of the link quality. The link quality grade is taken as a metric of the link quality estimation. Particle Swarm Optimization (PSO) algorithm is used to optimize the parameters of the weighted extreme learning machine (WELM), including the number of hidden nodes, weights, and the normalization factor. A link quality estimator (LQE) based on the improved weighted extreme learning machine (LQE-IWELM) is constructed. In different scenarios, experiment results show that the improved weighted extreme learning machine (IWELM) is more effective than extreme learning machine (ELM) and WELM. Compared with the other three link quality estimation models, LQE-IWELM has better precision and G_mean.

INDEX TERMS Wireless sensor networks, link quality estimation, weighted extreme learning machine, particle swarm optimization algorithm.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) are multi-hop self-organizing networks, which is formed by a large number of cheap micro-sensor nodes deployed in the monitoring area through wireless communication [1]. Sensor nodes in WSNs are small and low-cost, and can simply and efficiently collect environmental information in the monitoring area. The WSNs composed of multi-sensor nodes and sink nodes have the advantages of rapid deployment, high fault-tolerance, and good portability. Due to the advantages mentioned above, it has a broad application prospect in smart homes, urban traffic, surveillance [2], military reconnaissance, e-health [3], medical rescue [4], and other fields [5]–[7].

The WSNs are deployed in complex environments, such as the industrial, natural, and smart grids frequently, the links in WSNs are susceptible to multipath effect, loss, and adjacent channel interference, which result in unreliable links, low channel quality, and frequent topological changes [8]. Previous studies show that compared with the processing module and sensor modules, most of the energy is consumed in the data transmission modules, such as packet retransmission caused by interference [9].

The purpose of the link quality estimator is to accurately evaluate the link quality between nodes and provides a reference for the routing protocol to select a good link for data transmission. Effective link quality estimation can improve the stability of WSNs, reduce the number of data retransmissions, and save energy consumption, thus improving network throughput, and prolonging the network life cycle [10].

On the basis of the link quality estimation method based on machine learning, we combine the extreme learning machine with the class weighting method to get the weighted extreme learning machine (WELM) algorithm. Besides, the improved weighted extreme learning machine (IWELM) algorithm is obtained by optimizing the parameters of WELM with particle swarm optimization.

The associate editor coordinating the review of this manuscript and approving it for publication was Tiankui Zhang.

TABLE 1. A simple review of related work on link quality estimation.

Type	Author (year)	LQE name	Basic metric	Technical details
Software-based	Afzal S R (2017) [9]	xDDR	PDR, ETX	Pulse quality based neighbor selection and classification
	Woo A, Culler D (2003) [10]	WMEWMA	PRR	Use window mean with Exponentially Weighted Moving Average to estimate the PRR in next moment
Hardware-based	Gomes R D (2017) [11]	LQE Node	RSSI, received information	Design and implement the LQE node to estimate the link quality
Composite metric-based	N. Baccour (2015) [8]	F-LQE	SNR, PRR	Fuzzy logic is used to fuse PRR, SNR, and their derived metrics, a link quality evaluation model considering multiple attributes of the link is constructed
	T. Jayasri, M. Hemalatha (2017) [12]	ELQET	SNR, RSSI, PRR	Use Kalman filter and Fuzzy Logic to process basic metric and return quality score, then use the quality score to estimate the link quality
	W. Liu, Y. Xia et al. (2020) [13]	LFI-LQE	SNR, LQI, PRR	Combining LQI and SNR, a new metric WEB is obtained, and construct a mapping model with PRR
Machine learning-based	W. Sun, W. Lu et al. (2020) [14]	WNN-LQE	SNR, PRR	Decomposition SNR series as the input of WNN-LQE, which use to the estimate time-varying nonlinear part and the variance of the non-stationary random part, then estimate the range of PRR by constructing a related function
	Shu J et al. (2017) [15]	LQE-SVM	RSSI, LQI, SNR, PRR	Get labels according to the range of PRR. Then RSSI, LQI, SNR, and label are used as the input of SVM to train the LQE

The main contributions of this paper are as follows:

(1) Considering the imbalance of the samples after dividing the link quality grades, the link quality estimator based on the weighted extreme learning machine (LQE-WELM) is proposed in this paper. The category weight of the LQE-WELM model is determined by the golden section coefficient method, and L2 regularization is added to improve the generalization of the model.

(2) The particle swarm optimization algorithm is used to optimize the parameters of WELM. Experiments show that IWELM has better G_mean than the ones of WELM and ELM.

The rest of this paper is structured as follows: In Section II, we briefly describe related studies, particularly those that existing research in link quality estimation problems and extreme learning machine approaches. And we give the reasons for choosing three link quality parameters and divide the link quality grades. Section III analyzes the Pearson correlation among physical layer parameters, derived parameters, and PRR, select appropriate parameters as the input of the link quality estimation model. In Section IV, we provide an overview of our proposed methodology for link quality estimation including the design of model structure and selection of activation function, optimization of model, time complexity analysis, and evaluation of the model. In Section V, the performance of our approach is evaluated based on the experimental results and compared with related approaches for link quality estimation. Our conclusion and future work are presented in Section VI and Section VII respectively.

II. RELATED WORK

This section mainly introduces the recent research of link quality estimation and extreme learning machine. The link quality estimation methods in WSNs can be divided into software-based methods, hardware-based methods, composite metric-based methods, and machine learning-based methods. The related work about link quality estimation in this paper is reported in Table 1. In addition, Table 2 reported the acronyms for this paper.

TABLE 2. Legend of acronyms in this paper.

Acronym	Description
PRR	Packet Reception Rate
RSSI	Received Signal Strength Indication
LQI	Link Quality Indication
SNR	Signal-to-Noise Ratio
CV	Coefficient of Variation
LQE	Link Quality Estimator
EWMA	Window mean with Exponentially Weighted Moving Average
ELM	Extreme Learning Machine
WELM	Weighted Extreme Learning Machine
PSO	Particle Swarm Optimization
IWELM	Improved Weighted Extreme Learning Machine
SVC	Support Vector Classifier
WNN	Wavelet Neural Network
LFI	Lightweight, Fluctuation Insensitive Multi-Parameter Fusion
MLP	Multi-Layer Perceptron

A. LINK QUALITY ESTIMATION

The software-based link quality estimation method uses the parameters of the software layer in WSNs to estimate link quality. Based on the Expected Transmission Count (ETX), Afzal *et al.* [12] put forward a link quality estimation Indicator named xDDR. Experiments show that xDDR performs well in end-to-end packet transmission estimation in different scenarios and has a higher transmission rate than ETX. Woo *et al.* [13] proposed an window mean with Exponentially Weighted Moving Average (WMEWMA) link quality estimation algorithm, which predicted the packet received rate (PRR) at the next moment through a set of historical PRR, but the lack of physical layer parameters may lead to a decrease in the accuracy.

The hardware-based method evaluates link quality by the parameters of the physical layer in WSNs. Gomes *et al.* [14] proposed a link quality estimator for industrial wireless sensor networks and designed a new type of node, namely Link Quality Estimator (LQE) node, which estimates the link quality in real-time by using received signal strength indicator (RSSI) and received information. The proposed LQE can capture the effects of multipath, interference, and link asymmetry, and has higher accuracy and better responsiveness compared with optimized Fuzzy-LQE (Opt-FLQE) [11].

The composite metric-based method evaluates the link quality by synthesizing cross-layer multi-parameters, which can reflect the current link quality from many aspects. Jayasri and Hemalatha [15] designed an enhanced link quality estimation technique (ELQET), which combines Kalman filter and fuzzy logic, and smoothes the quality fraction returned by fuzzy quality with PRR, mean link quality indication (LQI), mean signal-to-noise (SNR), and stability factor by WMEWMA. The result of experiments shows that ELQET has lower energy consumption and root mean square error. Liu *et al.* [16] proposed a link quality estimation model based on the idea of multi-parameter fusion. Firstly, LQI and SNR are preprocessed by an exponential weighted Kalman filter to obtain the stable estimated values. Then, lightweight weighted Euclidean distance is used to fuse the above parameters. Finally, a mapping model is constructed by using the fused parameters and PRR, and the link quality is estimated by using this model. Compared with the method which only uses hardware or software layer parameters to evaluate link quality, it can evaluate link quality more comprehensively and accurately.

The machine learning-based method builds a related mapping model by deeply digging into the correlation between link quality and link quality parameters, including physical layer parameters and software layer parameters, and estimating link quality effectively. The link quality estimation algorithm based on the wavelet neural network (WNN-LQE) is proposed by Sun *et al.* [17]. The time series of SNR is taken as the input of the algorithm, and the SNR and its variance at the future time are obtained through WNN-LQE, then transformed into the estimation limit of PRR through the mapping function between SNR and PRR. Shu *et al.* [18] constructed an estimation model based on the support vector machine (SVM) and decision tree, which transforms the link quality estimation problem into a classification problem, The link quality estimator takes received RSSI and LQI as the input of the model, divides the link quality grade as label according to the PRR, and takes the link quality grade as the output. The experimental results show this model has better estimation accuracy.

The software-based method needs to send a certain number of packets to get software layer parameters, which lacks real-time performance, therefore the software-based method cannot reflect link quality in time and lead to more energy consumption in WSNs. The hardware-based method is fast in the calculation and can reflect the current link quality in time, but it does not consider the information loss caused by packet loss between nodes, therefore the results are not stable. The composite metric-based method can fuse the cross-layer infomation of link quality, and then effectively evaluate the current link quality, but it needs a certain time cost to build an adequate model. The machine learning-based on can also mine the relationship between physical layer information and link quality, which can achieve better evaluation results than the composite-metric-based method, but in general, the time demand for model construction is longer than the former method.

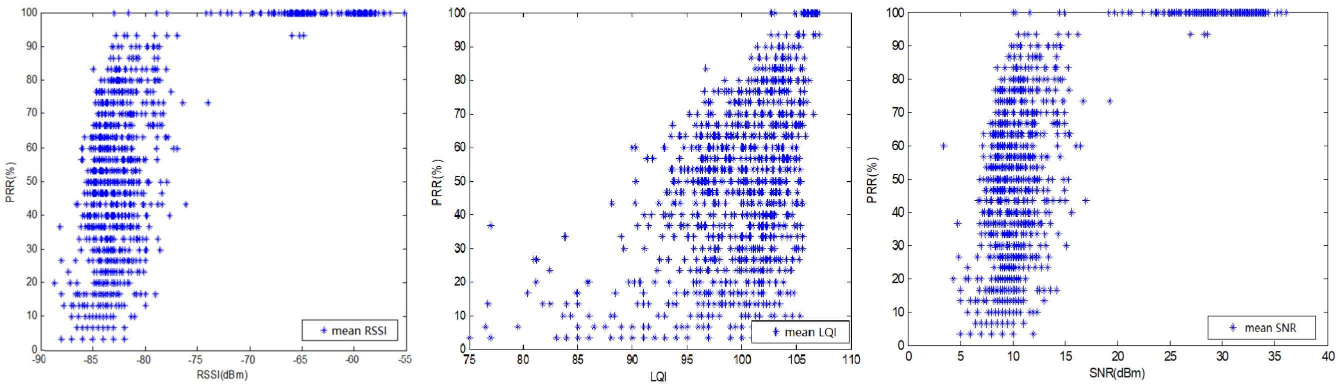
B. EXTREME LEARNING MACHINE

Extreme Learning Machine (ELM) [19] is a feedforward neural network with a single hidden layer. Compared with BP neural network and Support Vector Machine (SVM), the connection weights and hidden layer thresholds of ELM are randomly generated [20]. And the extreme learning machine has the characteristics of fast convergence and strong generalization ability. Moreover, there are a few relevant parameters in ELM, which are easy to determine. ELM can also overcome the local minimum problem [21] and has been widely used in classification or regression problems [22].

In the process of estimating the link quality of WSNs, the samples with different link quality grades are prone to imbalance. However, the shortcoming of extreme learning machines in the multi-classification of unbalanced problems is that the algorithm tends to favor a large number of classes. Zong *et al.* [23] applied cost-sensitive learning to extreme learning machines and proposed weighted extreme learning machines. It solves the problem of imbalance by giving different weights to different grades of samples.

To solve the imbalance problem in the link quality estimation problem, this paper adjusts the weight of different samples based on the original ELM by the golden section coefficient method, thus enhancing the classification ability of the link quality estimator for unbalanced samples. Besides, we use the particle swarm optimization (PSO) algorithm [24] to optimize the parameters of WELM and propose an improved weighted extreme learning machine algorithm based on the PSO algorithm (IWELM).

In this paper, firstly, the Pearson correlation coefficient is used to calculate the correlation degree between link quality hardware parameters and PRR, which is taken as the input of the link quality estimator; Secondly, we divide link quality



(a) The scatter diagram of mean RSSI and PRR. (b) The scatter diagram of mean LQI and PRR (c) The scatter diagram of mean SNR and PRR

FIGURE 1. The relationship among mean RSSI (a), mean LQI (b), mean SNR (c), and PRR.

grades according to the range of PRR and construct the link quality estimation model based on the WELM; Aiming at the optimization of the parameters of WELM, the PSO algorithm is used to optimize the number of hidden layer nodes and class weight, and the LQE-IWELM is proposed. At last, through experiments in different scenes, the validity of the model is verified by the precision and G_mean.

Compared with other link quality estimation methods, The LQE-IWELM proposed in this paper can better estimate the link quality. It reduces the work required for parameter adjustment. However, the proposed method requires a certain amount of data and training time, which makes the construction of the proposed model take more time than other methods.

III. SELECTION OF LINK QUALITY PARAMETERS AND DIVISION OF GRAD

A. SELECTION OF LINK QUALITY PARAMETERS

We collected link quality samples by sent probe packets periodically in different scenarios and used all values in the period to calculate the average parameter and derived parameters, which as alternative sets of link quality parameters, including mean RSSI, mean LQI, mean SNR, Variances of RSSI, LQI, SNR, and CV(Coefficient of variation) of RSSI, LQI, and SNR. Then the Pearson correlation coefficient was used to analyze the correlation between alternative parameters and PRR, and then the appropriate parameters were selected as link quality parameters.

To select suitable parameters, we analyze the Pearson correlation coefficient between the alternative parameter and PRR in three scenarios, as shown in Table 3.

As can be seen from Table 3, $\overline{RSSI}$, $\overline{LQI}$ and $\overline{SNR}$ have a high correlation with PRR in three scenarios. Besides, in the woods scene, the Variance of LQI, CV of LQI also have a high correlation with PRR, but the correlation is not obvious in the other two scenes. In addition, we draw the scatter diagram to analyze the correlation between $\overline{RSSI}$, $\overline{LQI}$ and $\overline{SNR}$ and PRR, as shown in Figure 1. It can also be seen from Fig. 1 that three parameters have high correlations with PRR. This paper select $\overline{RSSI}$, $\overline{LQI}$ and $\overline{SNR}$ as link quality parameters to estimate link quality.

TABLE 3. Correlation between parameters and PRR in different scenarios.

	PRR		
	Corridor	Woods	Square
Mean RSSI	**0.6226**	**0.8853**	**0.4004**
Mean LQI	**0.5341**	**0.9614**	**0.4304**
Mean SNR	**0.6220**	**0.8790**	**0.3357**
Variance of RSSI	0.1301	0.0527	0.2216
Variance of LQI	-0.1235	**-0.4206**	0.2035
Variance of SNR	0.0578	0.0421	0.0199
CV of RSSI	-0.1365	-0.0917	-0.2213
CV of LQI	-0.1327	**-0.4440**	0.1790
CV of SNR	0.0401	-0.1048	0.0314

B. DIVISION OF LINK QUALITY GRADE

Bildea *et al.* [25] divides link quality grades according to the range of PRR. It defines links with PRR values between 80% and 100% as the good link, links between 20% and 80% as the middle link, and links between 0% and 20% as the bad link. In this paper, the link quality is divided according to the standard above, and the link quality grade is defined as the link quality estimation metrics, the division criteria are shown in Table 4.

IV. LINK QUALITY ESTIMATION MODEL

In this paper, WELM is used to build the link quality estimation model, $\overline{RSSI}$, $\overline{LQI}$ and $\overline{SNR}$ are used as the input of the model, and the link quality grade is the output of the WELM. The trained model is used to estimate the link quality. Before establishing the model, this paper uses the Kalman filter to denoise the data firstly, and the weight distribution

TABLE 4. Definition of link quality grade.

Link Quality Grade (LQG)	Description	The range of PRR
1	Good link	$80\% \leq PRR \leq 100\%$
2	Middle link	$20\% \leq PRR < 80\%$
3	Bad link	$0 \leq PRR < 20\%$

mechanism based on the golden section coefficient [23] is used to determine the category weight of samples. The number of hidden nodes in WELM is set randomly. Aiming at the shortcoming of the setting of hidden layer nodes manually and the uncertain effect of the weighting method, we use the PSO algorithm to set the above parameters, and finally, LQE-IWELM is obtained.

A. DATA PREPROCESSING

The data will be affected by noise and multipath effects in the process of collection and needs to be preprocessed before it can be used as the input of the model. The different ranges and dimensions of link quality parameters will also affect the effectiveness of the model, so it is necessary to normalize the data. In this paper, data preprocessing can be divided into the following steps:

1) DENOISING OF DATA

The raw data collected directly by WSNs-LQT (Wireless Sensor Networks Link Quality Testbed) is noisy and cannot reflect the real link quality. Kalman filter [26] is a state-space model of the linear stochastic system composed of observation equations and state equations, which can divide into two stages. In the prediction stage, the current state prediction is estimated by using the estimated value of the previous time. In the correction stage, the currently observed value is used to correct the predicted value in the last stage, and then obtain the best-estimated value at the current time. Therefore, this paper uses the Kalman filter to denoise the data.

2) REMOVING ABNORMAL DATA

Due to the interference in the environment, all data packets may be lost in the detection period. In this paper, such samples are regarded as outlier samples. In the process of data statistics, the outlier samples consisting of zeros are eliminated.

3) NORMALIZATION

Different hardware parameters have different ranges and dimensions, taking them directly as the input of the model will affect the learning effect of WELM. After data normalization, the calculation can be simplified and the solving speed can be increased. In this paper, the max-min normalization method is used to normalize the parameters. The normalization formula is as shown in Eq. 1.

$$X_{ij} = \frac{x_{ij} - \min\left\{x_{ij}, \cdots, x_{Mj}\right\}}{\max\left\{x_{ij}, \cdots, x_{Mj}\right\} - \min\left\{x_{ij}, \cdots, x_{Mj}\right\}} \tag{1}$$

where x_{ij} is the value corresponding to the $i-th$ parameter of the $j-th$ sample, and X_{ij} is the normalized sample value.

B. LINK QUALITY ESTIMATION MODEL BASED ON WELM

ELM is one of the single hidden layer feedforward neural networks [27]. The number of nodes in the input layer and the output layer is determined by the dimension of input data and the expected output dimension. The number of hidden layer nodes directly affects its classification performance [27]. However, the number of hidden layer nodes is the hyperparameter of WELM, which is set manually. In this paper, the parameter obtained by PSO is taken as the number of hidden layer nodes of WELM.

In this paper, preprocessed $\overline{RSSI}$, $\overline{LQI}$ and $\overline{SNR}$ are combined into a triple, which is used as input. The input layer has three nodes. The link quality of wireless sensor networks is divided into three grades: good link, medium link, and bad link, and there are three nodes in the output layer. The ELM structure is shown in Fig. 2.

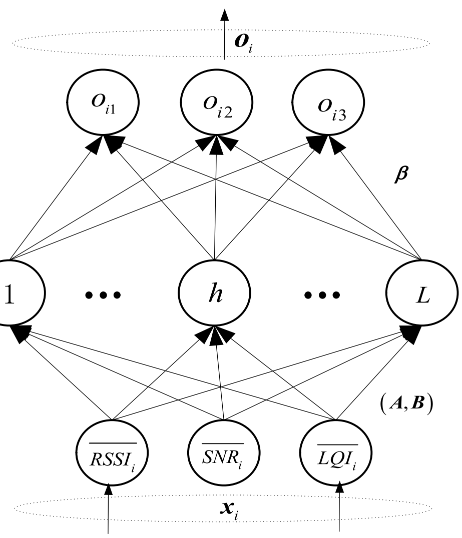


FIGURE 2. Link quality estimation model structure based on WELM.

Where $\boldsymbol{x}_i = \left\{\overline{RSSI}_i, \overline{SNR}_i, \overline{LQI}_i\right\}$ is the input of the $i-th$ sample, A and B are the connection weight matrix and the offset matrix respectively. L is the number of neurons in the hidden layer. β is the connection weight matrix between the hidden layer and the output layer. $o_i = (o_{i1}, o_{i2}, o_{i3})$ is the one-hot encoding of the link quality grades of the $i-th$ sample. The WELM is obtained by weighting the different categories of ELM input, which can solve the problem that a few samples are difficult to identify due to the imbalance of sample categories. The category weight of samples is obtained based on the weight distribution mechanism of the golden section coefficient.

The WELM sets the connection weight matrix $\boldsymbol{A}$ and the bias weight matrix $\boldsymbol{B}$ to be randomly generated and fixed. After obtaining the output matrix of hidden layer nodes, $\hat{\boldsymbol{\beta}}$ is obtained by solving the minimum norm least square solution [28]. The essence of build a link quality estimation model based on WELM is the process of optimizing the parameters.

We suppose the data set processed by Kalman filter as $\Gamma = \{\sigma_i\}$ where $\sigma_i = (x_i, y_i)\,, i = 1, 2, \cdots, M, M$ is the total number of samples; $x_i = \left\{\overline{RSSI_i}, \overline{SNR_i}, \overline{LQI_i}\right\}$ is the input of ELM, and $y_i = (y_{i1}, y_{i2}, y_{i3})$ is the link quality grade by one-hot encoded of the $i-th$ sample.

Based on the model structure, for an ELM network with hidden layer nodes, the network output is shown in Eq. 2 [19].

$$o_i = \sum_{h=1}^{L} \beta_h g\left(a_h X_i + b_h\right) \tag{2}$$

where $\beta_h = (\beta_{h1}, \beta_{h2}, \beta_{h3})$ is the weight vector from the $h-th$ hidden layer node to the output node; b_h is the offset of the $h-th$ hidden layer node; $g()$ is the activation function; $a_h = (a_{h1}, a_{h2}, a_{h3})$ is the weight vector from the $h-th$ hidden layer node to the input node. ELM algorithm aims to minimize the error between expected output and actual output, and the objective function is shown in Eq. 3 [19].

$$\sum_{i=1}^{N} \|o_i - y_i\| = 0 \tag{3}$$

The existence of β_h, a_h and b_h makes the Eq. 4 holds [19].

$$y_i = \sum_{h=1}^{L} \beta_h g\left(a_h X_i + b_h\right), \quad i = 1, 2, \cdots M \tag{4}$$

Expand to the whole network by Eq. 5 [16], we can calculate H.

$$\boldsymbol{H\beta} = \boldsymbol{Y} \tag{5}$$

where H is the output matrix of hidden layer nodes and as shown in Eq. 6 [19].

$$H = \begin{bmatrix} g\left(\boldsymbol{a}_1\boldsymbol{X}_1 + b_1\right) & \cdots & g\left(\boldsymbol{a}_L\boldsymbol{X}_1 + b_L\right) \\ \vdots & & \vdots \\ g\left(\boldsymbol{a}_1\boldsymbol{X}_N + b_1\right) & \cdots & g\left(\boldsymbol{a}_L\boldsymbol{X}_N + b_L\right) \end{bmatrix} \tag{6}$$

After the input layer weight and hidden layer node basis are randomly determined, the least square solution with the minimum norm is obtained, as shown in the equation above [19].

$$\boldsymbol{\beta}\hat{} = \boldsymbol{H}^{+}\boldsymbol{Y} \tag{7}$$

Among them, $\boldsymbol{H}^{+}$ is Moore-Penrose generalized inverse of $\boldsymbol{H}$, $\hat{\boldsymbol{\beta}}$ is the weight matrix from the hidden layer node to the output layer, which can be calculated by orthogonal projection method [19], as shown in Eq. 8:

$$\boldsymbol{\beta}\hat{} = \begin{cases} \boldsymbol{H}^{T}\left(\boldsymbol{HH}^{T}\right)^{-1}\boldsymbol{Y}, & N \leq L \\ \left(\boldsymbol{H}^{T}\boldsymbol{H}\right)^{-1}\boldsymbol{H}^{T}\boldsymbol{Y}, & N > L \end{cases} \tag{8}$$

To solve the problem of unbalanced samples after link quality grade division, this paper uses the sample class weighting method to maximize the boundary distance and minimizes the accumulated error of all training samples, thus improving the classification ability of the ELM.

The category weight of WELM is obtained by the weight allocation method based on the golden section coefficient [23], as shown in Eq. 9.

$$\boldsymbol{w} = \begin{cases} w_{ii} = \dfrac{1}{\vartheta_{im}}, & \vartheta_{im} > \Psi \\ w_{ii} = \dfrac{0.618}{\vartheta_{im}}, & \vartheta_{im} \leq \Psi \end{cases} \tag{9}$$

where ϑ_{im} is the $i-th$ sample, which corresponds to the number of samples of category m, and Ψ is the average number of samples of different category.

Furthermore, we introducing a regularization term to improve the generalization ability of the model. The corresponding optimization objective is as follows [22]:

$$minimize: \frac{1}{2}\|\beta\|^2 + C\frac{W}{2}\sum_{i=1}^{N}\|\varepsilon_i\|^2$$
$$Subject\ to: \boldsymbol{H\beta} = \boldsymbol{Y} - \boldsymbol{\varepsilon} \tag{10}$$

Among them, the second section is L2-norm. Where C is the regularization factor; $W = \left[w_{good}, w_{middle}, w_{bad}\right]$ is the weight vector of three link quality grades. The weighted ELM objective function is shown in Eq. 11 [23].

$$\hat{\boldsymbol{\beta}} = \begin{cases} \boldsymbol{H}^{\mathrm{T}}\left(\dfrac{1}{C} + \boldsymbol{WHH}^{\mathrm{T}}\right)^{-1}\boldsymbol{WY}, & N \leq L \\ \left(\dfrac{1}{C} + \boldsymbol{H}^{\mathrm{T}}\boldsymbol{WH}\right)^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{WY}, & N > L \end{cases} \tag{11}$$

The link quality estimation model based on the weighted extreme learning machine is shown in Eq. 12.

$$LQG = LQE - WELM\left(\mathbf{W}, C, G, L, X\right) \tag{12}$$

where LQG is the link quality grade output by the WELM. The WELM training and testing algorithm is described as Algorithm 1.

C. SELECTION OF THE ACTIVATION FUNCTION

The selection of activation function has a great influence on the performance of the ELM. Selecting the appropriate activation function can make the model have strong fitting ability. The activation functions commonly used by ELM are sigmoid function, Sine function, and Hardlim function [29].

Therefore, in this paper, the above three activation functions are taken as the candidate and choose the appropriate function based on the experimental results. The formula for each activation function is as follows.

$$g(x) = sigmoid(x) = \frac{1}{1 + e^{-x}} \tag{13}$$

$$g(x) = \sin(x) \tag{14}$$

$$g(x) = hardlim(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{15}$$

Algorithm 1 WELM Training and Testing Algorithm

Input: link quality data set X, number of hidden layer nodes L, activation function g, weight coefficient W, regularization factor C;
Output: G_mean, link quality grade LQG;
1: constructing a sample inter-class weight matrix $\boldsymbol{W}$ by Eq. 9;
2: randomly generating a connection weight matrix $\boldsymbol{A}$ with values at [-1,1];
3: randomly generating a bias weight matrix $\boldsymbol{B}$ with values at [0,1];
4: using the training set as the input of WELM, compute output matrix $\boldsymbol{H}$ according to Eq. 6;
5: obtaining a connection weight matrix $\hat{\boldsymbol{\beta}}$ according to Eq. 11;
6: calculate the G_mean on the testing set;
7: return LQG, G_mean

The experimental results of three different scenarios show that, when Sigmoid is used as the activation function of the model, the model performs better than others in both the training set and the test set. The influence of the activation function on the training algorithm is described in the section of analysis of experimental results.

D. OPTIMIZING THE PARAMETERS OF WELM BY PSO ALGORITHM

WELM algorithm calculates the weights of each category by the weight distribution mechanism of the golden section coefficient, and sets the number of hidden layer nodes manually, which makes it difficult to obtain the optimal model.

In this paper, we use the PSO algorithm to optimize the parameters of WELM, including the number of hidden layer nodes L, weight coefficient W, and regularization factor C. PSO algorithm is an intelligent optimization algorithm, which is easy to implement and robust. The algorithm will generate a random solution to iterate until the fitness value meets the requirements or reaches the maximum number of iterations. The quality of the solution is evaluated by fitness value, and the higher the fitness value, the greater the quality of the solution.

Considering the computational complexity of the model, the search range of W is set to [0, 1], the search range of L and C are set to [20, 500] and [0.001, 50] respectively. The position of particles is evaluated by the G_mean.

This paper encodes the parameters to be optimized as the positions of particles in particle swarm optimization. $q_r = [W, L, C]$ is the encoding vector of the parameter, which indicates the position of the $r-th$ particle in the particle swarm.

PSO initializes particle swarm $Q = \{q_1 \ldots q_r \ldots q_N\}$ randomly, where N is the number of particles, particle velocity is randomly initialized to v_{5*1}. Then, the PSO algorithm updates the position and velocity of particles by using Eq. 16 and Eq. 17 [30] respectively.

$$\begin{aligned} v_r^{t+1} &= \varpi v_r^t + c_1 \times rand() \times \left(h_r^p - q_r^t\right) \\ &\quad + c_2 \times rand() \times \left(h_r^g - q_r^t\right) \end{aligned} \tag{16}$$

$$q_r^{t+1} = q_r^t + v_r^{t+1} \tag{17}$$

where v_r^{t+1} indicates the speed of the $r-th$ particle in the $r+1-th$ iteration. q_r^t and q_r^{t+1} are the position of the $r-th$ particle in the $r-th$ iteration and the $r+1-th$ iteration respectively. h_r^p is the individual extremum of the $r-th$ particle, and h_r^g is the optimal solution of the whole particle swarm. c_1 and c_2 are acceleration factors, and set to 1.5 and 1.7 empirically. ϖ is the inertia weight, which is used to control the influence of historical speed on the current speed, and set as 1 empirically.

The link quality estimation model LQE-IWELM is shown in Eq. 18.

$$LQG = LQE\text{-}IWELM\ (\mathrm{g}, N, T, X) \tag{18}$$

The training algorithm of LQE-IWELM is shown in Algorithm 2.

Algorithm 2 The Training Algorithm of LQE-IWELM

Input: link quality data set X, maximum iteration T, particle population number N, activation function g;
Output: LQE-IWELM;
1: Initializing parameters of PSO algorithm; Randomly initialize the initial position and velocity of all particles;
2: Get the G_mean of the particle through **Algorithm 1**;
3: Calculate the fitness of all particles, and record as $\hbar_r^p$;
4: $h_r^g = max(\hbar_r^p)$;
4: for $t < T$
5: increment t $(t++)$;
6: for $i < N$
7: increment i $(i++)$;
8: update v_r^t and q_r^t according to Eq. 16 and Eq. 17;
9: check whether to exceed the boundary value, If exceed the boundary-value, adjust the legitimacy;
10: $G\_mean = WELM(q_r^t, g, X)$
11: if $\hbar_r^p < G\_mean$
$\hbar_r^p = G\_mean$;
12: $temp = max(\hbar_r^p)$
if $h_r^g < temp$
$h_r^g = temp$;
13: return: LQE-IWELM according to particle position h_r^g

E. TIME COMPLEXITY ANALYSIS

In this paper, the time complexity of the proposed IWELM consists of two parts [31], [32]. The first part is the time complexity of the WELM training and prediction process. And the second part is the time complexity of updating particle velocity and position in the PSO algorithm.

The training process of WELM is as follows: First, calculate the hidden layer output matrix H, which has time complexity equal to $O(M^*L^*D)$, where L is the number of hidden nodes, M is the number of training samples, and D is the dimension of input data; Then, Calculate the network output weight matrix $\widehat{\beta}$ through Eq. 11, which has the time complexity equal to $O(L^3 + L^{2*}M + L^*M^*C)$, and C is the number of output class. The time complexity of the WELM training process is $O(L^3 + L^{2*}N + (C + D)^*L^*M)$. And the time complexity of the WELM prediction process is $O((C + D)^*L^*M)$.

The time complexity of updating particle velocity and position in PSO is $O(dim^*N)$, where dim is the dimension of the particle and N is the number of particles.

To sum up, we can get the time complexity of IWEM is $O(T^*(dim^*N + cof^*N))$, where T is the maximum iteration, and cof is the time complexity of fitness function, that is, the time complexity of WELM training and prediction process. And the worst time complexity of the proposed method is:

$$\begin{aligned} T_w &= O(T^*(dim^*N + (L^3 + L^{2*}N + 2 \\ &\quad {}^*(C + D)^*L^*M)^*N)) \\ &= O(T^*(5^*N + (L^3 + L^{2*}N + 2^*(3 + 3) \\ &\quad {}^*L^*M)^*N)) \\ &= O(T^*N^*(L^3 + L^{2*}N + 12^*L^*M + 5)) \\ &= O(T^*N^*(L^3 + L^{2*}N + 12^*L^*M)) \end{aligned}$$

F. EVALUATION OF THE MODEL

The evaluation of the proposed model is essentially the evaluation of the classifier. Considering the unbalanced characteristics of the dataset, confusion matrix, precision, and G_mean [30] are used to evaluate the model. The confusion matrix is shown in Table 5.

TABLE 5. The confusion matrix.

		Forecast category	
		Positive (P)	Negative (N)
True category	True (T)	TP	TN
	False (F)	FP	FN

In order to adapt the multi-level link quality estimation problem, The arithmetic mean of G_mean is used as the metric of LQE. Based on the confusion matrix, and the calculation formula of G_mean is as follows:

$$G\_mean = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}} \tag{19}$$

Precision is a common metric to evaluate unbalanced multi-classification, which is the fraction of relevant instances among the retrieved instances, and the calculation formula is as follows:

$$Precision = \frac{TP}{TP + FP} \tag{20}$$

The link quality is divided into three grades in this paper, and the confusion matrix is defined as a matrix with three rows and three columns. Among them, the horizontal coordinates and vertical coordinates are prediction category and actual category, and the confusion matrix is normalized by rows. The corresponding normalized confusion matrix is shown in Table 6.

TABLE 6. The confusion matrix normalized by rows.

		Forecast category		
		Good Link	Middle Link	Bad Link
Actual category	Good Link	0.8	0.1	0.1
	Middle Link	0.1	0.8	0.1
	Bad Link	0.1	0.1	0.8

V. EXPERIMENTAL SCENARIOS AND ANALYSIS

In order to verify the effectiveness of the model, the link quality data are acquired from multiple application scenarios. We use the TelosB node that is created by CrossBow to send and receive packets, and use the WSNs-LQT that is developed by the lab to collect the link quality parameters, including the RSSI, LQI, and SNR. The WSNs-LQT platform, which is shown in Fig. 3.

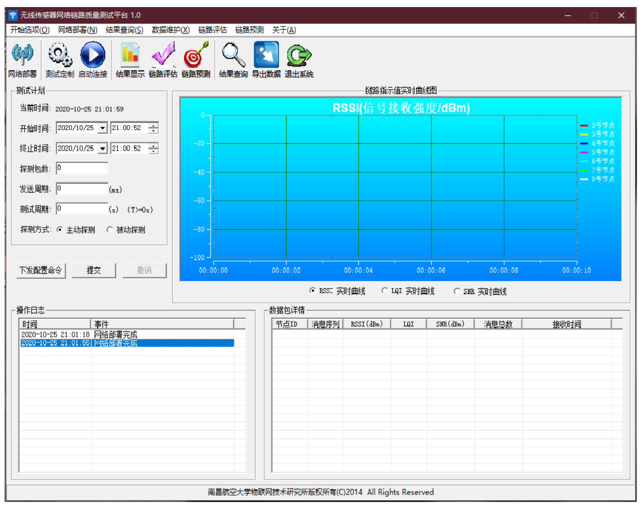


FIGURE 3. WSNs link quality testbed (WSNs-LQT).

The WSNs-LQT platform dynamically displays the link quality information from nodes. The RSSI, LQI, and SNR corresponding to each parameter are obtained after platform statistical processing. The link quality test parameters are set as shown in Table 7.

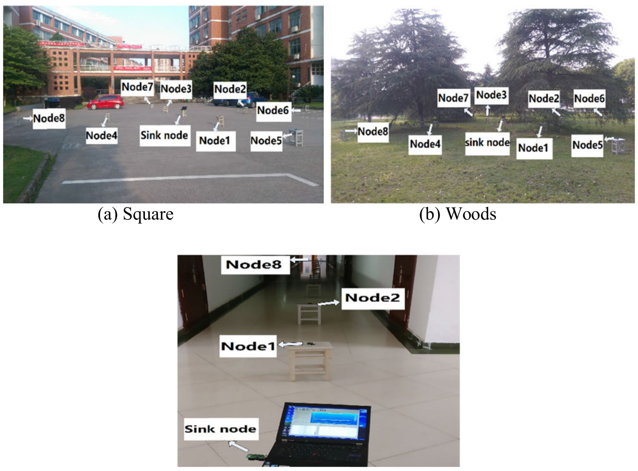
A. EXPERIMENTAL SCENARIO AND PARAMETER SETTING

Experimental scenarios have different environmental interference factors, such as environmental noise, multipath propagation, and channel interference, which have different impacts on the link quality of WSNs. To ensure the influence

TABLE 7. The setting of link quality testbed parameters.

Parameter attribute	value
Transmission power	-24~0 dBm
Channel	26
Number of probe packets	30
Packet send rate	10pcs/s
Test cycle	7s
Sending interval	200ms

of different interference sources on the experiment, a star network is deployed in the square, woods, and corridor to collect link quality data, as shown in Fig. 4.



(a) Square (b) Woods

(c) Corridor

FIGURE 4. Experimental scenarios: (a) Square, (b) Woods, (c) Corridor.

In the square scene, there are a large number of moving pedestrians after class, which are moving disturbances. And the interference caused by corresponding electronic devices, such as mobile phones and computers, will also increase. In this scenario, there are nine nodes in total, 1 sink node is connected to the laptop, and 8 sensor nodes are distributed at a distance of 5 meters around the receiving node. See Fig. 4 (a) for the specific distribution.

In the woods scene, the factors affecting the link quality mainly come from obstacles. As obstacles will counteract the propagation of wireless signals, resulting in reflection and refraction, the specific distribution of nodes in the woods scene is shown in Fig. 4 (b).

The corridor scene is used to simulate the communication between sensor nodes under mutual interference. Besides, there is interference from nearby laboratory handheld terminal devices (mobile phones, computers, Wi-Fi, etc.). In this scenario, the nodes are placed every 5 meters in the form of a straight line. And the specific distribution of nodes is shown in Fig. 4 (c).

B. ANALYSIS OF EXPERIMENTAL RESULTS

1) SELECTION OF ACTIVATION FUNCTION

The selection of activation function has a great influence on the performance of the LQE-IWELM. Hardlim, Sigmoid, and Sine function are selected for comparative experiments in this paper. The same data sets are used for training and testing in the three scenarios above. By comparing the experimental results of the LQE-IWELM model with different activation functions in three scenarios, the optimal LQE-IWELM in each scenario is determined. Then, we use the test set to analyze the performance of activation functions in different scenarios, and finally, the most suitable activation function is selected. In this experiment, the training set and test set were divided according to the ratio of 1:1. The number of particles and the maximum iteration number is 30 and 50 respectively.

In the square scene, 811 samples were collected from 1172 detection cycle samples after data preprocessing, including 56 good link samples, 384 medium link samples, and 370 poor link samples.

The convergence diagram of the IWELM algorithm with different activation functions on the training set is shown in Fig. 5.The confusion matrix of the IWELM model with different activation functions on the test set is shown in Table 8.

It can be seen from Fig. 5 that when Hardlim is used as the activation function of IWELM, the algorithm can achieve a better convergence effect on the training set of the square scene, and the corresponding G_mean is higher than the other two activation functions, which can reach 0.92.

However, it can be seen from Table 8 that when Hardlim is used as the activation function, although it performs well in the training set, it does not perform well in the test set, especially in the medium and poor links.

It can be seen that the generalization performance of the model using Hardlim as the activation function is poor in the square scene, while the model using Sigmoid has better classification ability in both the training set and the test set.

The LQE-IWELM parameters corresponding to different activation functions in the square scene are shown in Table 9.

It can be seen from Table 9 that IWELM with Sigmoid activation function can obtain a better G_mean, therefore, the sigmoid activation function is taken as the activation function of the LQE-IWELM model in the square scene.

In the Woods scene, 736 samples were collected from 1054 detection cycle samples after data preprocessing, including 355 good link samples, 304 medium link samples, and 77 poor link samples.The convergence diagram of the IWELM algorithm with different activation functions on the training set is shown in Fig. 6.

The convergence diagram of the IWELM algorithm corresponding to the three activation functions is shown in Fig. 6, which has a low convergence effect. It is mainly related to the connection weights between the input layer and the hidden layer generate randomly. The G_mean obtained by IWELM with Hardlim as activation function is equivalent to

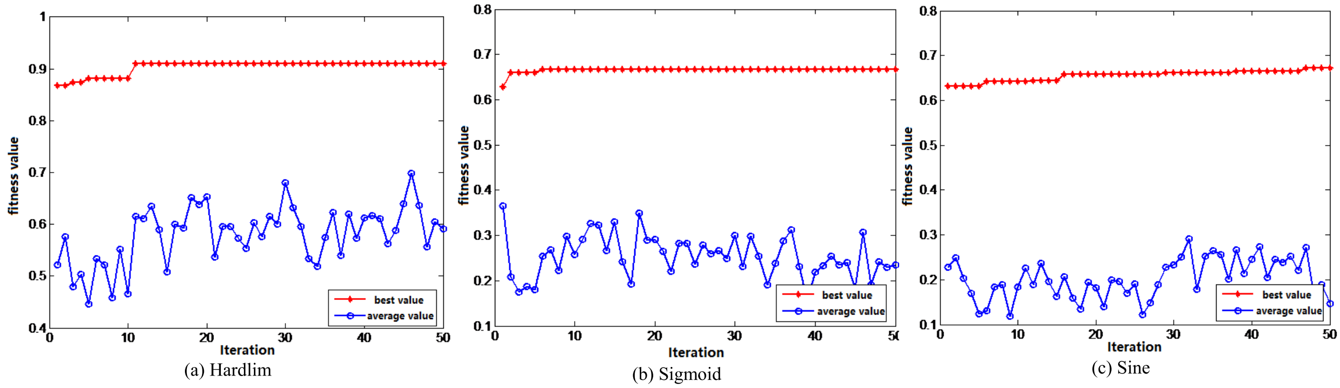


FIGURE 5. Convergence diagram of LQE-IWELM algorithm in the square scene.

TABLE 8. The confusion matrix of the lqe-iwelm with different activation functions in the square scene.

		Forecast category with Hardlim			Forecast category with Sigmoid			Forecast category with Sine		
		Good Link	Middle Link	Bad Link	Good Link	Middle Link	Bad Link	Good Link	Middle Link	Bad Link
Actual category	Good link	0.57	0.25	0.18	0.85	0.15	0.00	0.79	0.21	0.00
	Middle link	0.09	0.52	0.39	0.01	0.64	0.35	0.13	0.63	0.24
	Bad link	0.04	0.39	0.57	0.04	0.24	0.72	0.08	0.27	0.65

TABLE 9. Parameter statistics of lqe-iwelm corresponding to different activation functions in the square scene.

	Hardlim	Sigmoid	Sine
Regularization factor C	10	5.2265	3.9318
W_{good}	1	0.9257	1
W_{middle}	0.3365	0.0863	0.1037
W_{bad}	0.3099	0.0795	0.1152
Number of hidden layer nodes L	400	320	400
G_mean	0.5528	**0.7317**	0.6865

the G_mean obtained by IWELM with Sine, and the corresponding value is about 0.85.

Table 10 shows that the LQE-IWELM model with the Hardlim activation function has the same performance as the test set in the square, and its classification ability for bad links is poor. The parameter statistics of LQE-IWELM corresponding to different activation functions in the current scene are shown in Table 11.

It can be seen from Table 11 that LQE-IWELM based on the sigmoid activation function has better G_mean than others, and the corresponding value is 0.8276, and the Sigmoid is taken as the activation function in the woods scene.

In the corridor scene, 1800 samples were collected, and 1752 samples were obtained after data preprocessing, including 728 good link samples, 972 middle link samples, and 52 bad link samples. The convergence diagram of the IWELM algorithm with different activation functions on the training set is shown in Fig. 7.

The confusion matrix of the LQE-IWELM model with different activation functions on the test set is shown in Table 12. It can be seen from Fig. 7 that the IWELM algorithm using Hardlim as the activation function has the same effect as the other scenes in the training set, and the corresponding G_mean reaches 0.98.

As can be seen from Table 12, compared to the previous two scenarios, the LQE-IWELM model with Hardlim as the activation function has the same performance in the test set, the corresponding model has poor generalization. The parameter statistics of LQE-IWELM corresponding to different activation functions in the current scene are shown in Table 13.

It can be seen from Table 13 that LQE-IWELM based on the Sigmoid activation function has better G_mean, while LQE-IWELM based on the Hardlim activation function has poor overall evaluation performance, The Sigmoid is taken as the activation function for LQE-IWELM in corridor scene.

According to the experimental results above, the performance of the LQE-IWELM model with a certain activation function in three scenarios shows that the estimation effect of the model is the worst in the square scene, with an estimation accuracy of 85% for good links and 64% for medium links.

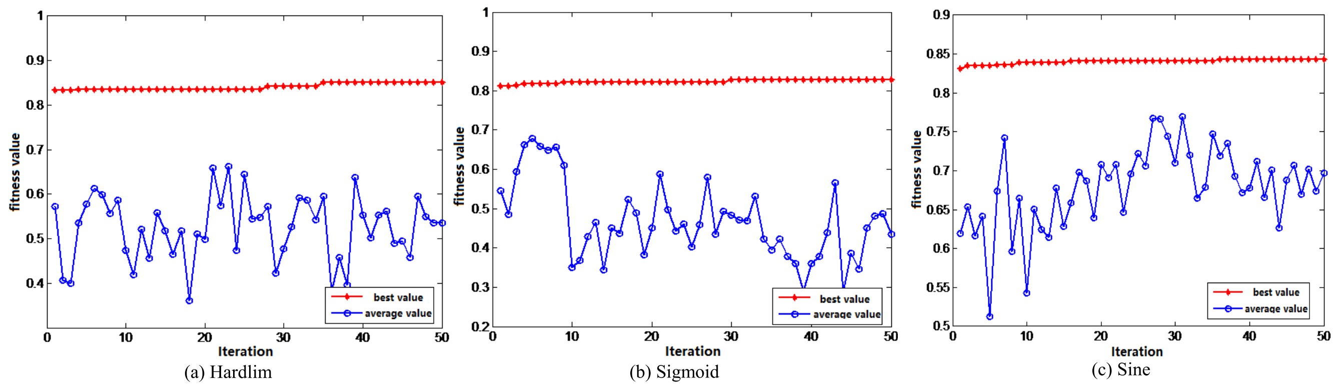


FIGURE 6. Convergence diagram of LQE-IWELM algorithm in the woods scene.

TABLE 10. The confusion matrix of the lqe-iwelm with different activation functions in the woods scene.

		Forecast category with Hardlim			Forecast category with Sigmoid			Forecast category with Sine		
		Good Link	Middle Link	Bad Link	Good Link	Middle Link	Bad Link	Good Link	Middle Link	Bad Link
Actual category	Good link	0.81	0.19	0.00	0.79	0.21	0.00	0.82	0.18	0.00
	Middle link	0.17	0.77	0.06	0.05	0.83	0.12	0.09	0.78	0.13
	Bad link	0.00	0.44	0.56	0.00	0.15	0.85	0.00	0.18	0.82

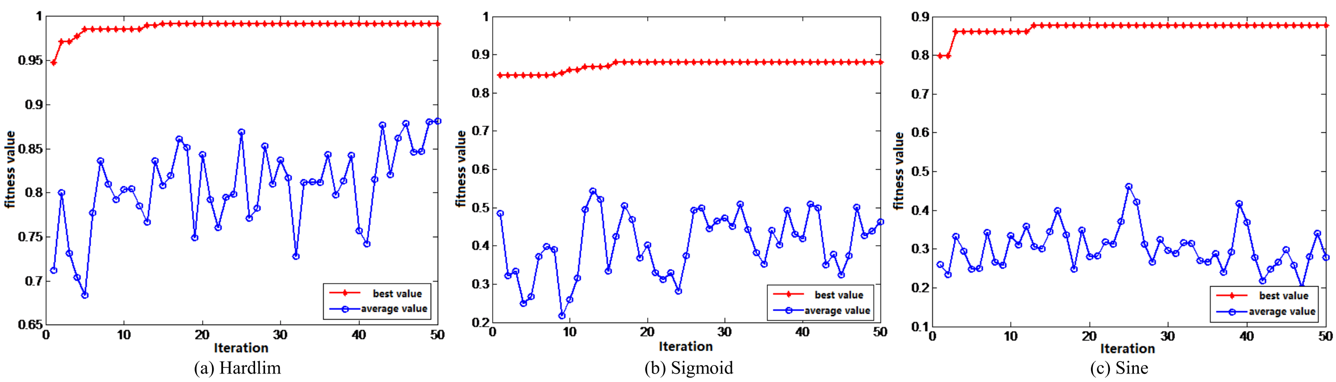


FIGURE 7. Convergence diagram of LQE-IWELM algorithm in the corridor scene.

TABLE 11. Parameter statistics of lqe-iwelm corresponding to different activation functions in the woods scene.

	Hardlim	Sigmoid	Sine
Regularization factor C	6.5788	9.5944	3.9318
w_{good}	0.8511	0.1964	0.1521
w_{middle}	0.7307	0.1726	0.1060
w_{bad}	1	1	0.2325
Number of hidden layer nodes L	400	400	314
G_mean	0.7042	**0.8276**	0.8064

The main reason is that there are many pedestrians in the square, the handheld devices may cause more interference, and affects the estimation.

According to the experimental results in the above three scenarios, the LQE-IWELM model has the best performance when Sigmoid is used as the activation function. And Sigmoid is chosen as the activation function.

2) COMPARISON BETWEEN ELM, WELM, AND IWELM

To verify the effectiveness of the IWELM algorithm proposed in this paper, the number of hidden layer nodes, the activation function, and the regularization factor are fixed, which are obtained by the PSO algorithm.

The link quality estimation models constructed by the ELM, WELM, and IWELM algorithm in different scenarios are compared. Table 14-16 show the performances and

TABLE 12. The confusion matrix of the lqe-iwelm with different activation functions in the corridor scene.

		Forecast category with Hardlim			Forecast category with Sigmoid			Forecast category with Sine		
		Good Link	Middle Link	Bad Link	Good Link	Middle Link	Bad Link	Good Link	Middle Link	Bad Link
Actual category	Good link	0.96	0.04	0.00	0.96	0.04	0.00	0.97	0.03	0.00
	Middle link	0.02	0.94	0.03	0.00	0.90	0.10	0.00	0.87	0.13
	Bad link	0.00	0.62	0.38	0.00	0.27	0.73	0.00	0.27	0.73

TABLE 13. Parameter statistics of lqe-iwelm corresponding to different activation functions in the corridor scene.

	Hardlim	Sigmoid	Sine
Regularization factor C	6.1658	10	10
w_{good}	0.4202	0.4449	0.2207
w_{middle}	0.2490	0.0552	0.0690
w_{bad}	1	1	1
Number of hidden layer nodes L	318	400	176
G_mean	0.6999	**0.8631**	0.8514

TABLE 14. Comparison of experimental results in the square scenes.

	ELM	WELM	IWELM
Number of hidden layer nodes	320	320	320
Regularization factor	5.2265	5.2265	5.2265
w_{good}	/	0.0179	0.9257
w_{middle}	/	0.0026	0.0863
w_{bad}	/	0.0027	0.0795
G_mean	0.3781	0.5605	0.7317

TABLE 15. Comparison of experimental results in the woods scene.

	ELM	WELM	IWELM
Number of hidden layer nodes	400	400	400
Regularization factor	9.5944	9.5944	9.5944
w_{good}	/	0.0028	0.1964
w_{middle}	/	0.0033	0.1726
w_{bad}	/	0.0130	1
G_mean	0.6814	0.7531	**0.8276**

algorithm parameters of each model on the test set in different scenarios.

It can be seen from Table 14-16 that due to the unbalanced data distribution, the estimation model obtained by

TABLE 16. Comparison of experimental results in the corridor scene.

	ELM	WELM	IWELM
Number of hidden layer nodes	400	400	400
Regularization factor	10.000	10.000	10.000
w_{good}	/	0.0014	0.4449
w_{middle}	/	0.0010	0.0552
w_{bad}	/	0.0192	1
G_mean	0.5621	0.7809	**0.8631**

the ELM algorithm has poor performance. WELM weighting different sample categories according to the number of training samples, which can improve the performance of unbalanced multi-classification. Compared with the ELM algorithm, the G_mean in the three scenarios is increased by about 0.2. IWELM can achieve the best G_mean in various experimental scenarios. Compared with the WELM algorithm, the G_mean of IWELM is improved by 0.17 at the maximum. Therefore, the IWELM proposed in this paper is effective and has better evaluation performance.

3) PERFORMANCE ANALYSIS AND COMPARISON OF DIFFERENT ESTIMATION MODELS

The method proposed in this paper belongs to machine learning-based LQE, and LFI-LQE [16] and WNN-LQE [17] are used as comparative methods to verify the effectiveness of the proposed method. Moreover the method proposed is a weighted method based on the neural network, therefore the weighted LQE-SVC [18] is chosen as a comparison method. Besides, the MLP(Multi-Layer Perceptron) is chosen as the baseline.

The experimental results of the LQE-IWELM model, the MLP, the LQE-WSVC model, the LFI-LQE model, and the WNN-LQE model in three scenarios are counted, and the estimation precision with different grades and G_mean are calculated. The results are shown in Fig. 8 and Table 17.

It can be seen from Fig. 8 (a), except LFI-LQE and Baseline, the other three LQEs have better precision when evaluating good link samples, because LFI-LQE uses a single fusion matrix as the input after processing, which limits the link characterization of links; Fig. 8 (b) shows the WNN-LQE and

(a)Good Link
(b)Middle Link
(c) Bad Link

FIGURE 8. Precision comparison of different link quality estimation models.

the MLP with the great effect of identifying the middle grades samples, because of the above two LQEs identify many bad links and good link as middle links, and The ability of two LQEs to distinguish different grade samples are not good. Other estimation models have similar effects in the process of identifying medium links; Fig. 8 (c) shows that except for the WNN-LQE and the MLP misclassification mentioned above, the other LQEs are also not far behind. The best one is LQE-WSVC, and LQE-IWELM is followed. The unidirectional link quality parameters are selected, causing the limited for LQE-IWELM to characterize the link quality of medium links and poor links.

Compare the estimation effects of five LQEs in three scenarios. Except in the woods scene, the performance of the MLP is not good in the other two scenes, and the MLP has serious misclassification in all scenes. In the square scene, LQE-IWELM has the best estimation precision for good links and bad links, while LQE-WSVC and LFI-LQE have similar precision, but the LQE-WSVC is 10% higher than the LFI-LQE, while WNN-LQE can hardly identify bad links. In the woods scene, LQE-WSVC, LFI-LQE, and LQE-IWELM have little difference in the estimated precision of the three links. Overall, LQE-WSVC is slightly better than other models. In the corridor scene, LQE-IWELM and LQE-WSVC have the same precision in evaluating good links, and LQE-IWELM is 11% higher than the latter in identifying medium links, but the former is 6% higher than the latter in identifying bad links. In different scenarios, the LQE-IWELM has a good recognition ability for links of different grades.

TABLE 17. The g_mean of different link quality estimation models.

	G_mean				
	MLP	WNN-LQE	LFI-LQE	LQE-WSVC	LQE-IWELM
square	0.5381	0.6268	0.6822	0.5989	**0.7317**
woods	0.7111	0.7761	0.8087	0.8245	**0.8276**
corridor	0.5968	0.6888	0.7089	0.8308	**0.8631**

It can be seen from Table 17 that, The MLP does not perform well in the square and the corridor scene, and only performs well in the woods scene, but not as well as other LQEs. The LQE-IWELM has higher G_mean than the others, especially in the square scene and corridor scene. In the square scene, the G_mean of the LQE-IWELM model is 6% higher than LFI-LQE and about 13% higher than LQE-WSVC. In the corridor scene, the G_mean of the LQE-IWELM model is about 3% higher than that of LQE-WSVC and 15% higher than that of LFI-LQE. The result shows that the LQE-IWELM model has a better classification ability for unbalanced samples, and its performance is better than the other three models.

VI. CONCLUSION

Based on analyzing the existing link quality estimation methods, this paper selects the hardware parameter $\overline{RSSI}$, $\overline{LQI}$, and $\overline{SNR}$ as the link quality parameters, and divides the link quality grades according to the range of PRR. The PSO algorithm is used to set the parameter of WELM. Compared with ELM and WELM, the effectiveness of the IWELM algorithm is verified. And the link quality estimation model LQE-IWELM is constructed. Through experiments in the three scenarios, it is confirmed that LQE-IWELM with Sigmoid as the activation function has better performance. The link quality estimation model LQE-IWELM compare with MLP, LFI-LQE, LQE-WSVC, and WNN-LQE in the three scenarios, LQE-IWELM has better precision and G_mean.

The existing link quality estimation models based on hardware parameters and machine learning methods are generally trained offline and need a large number of link samples for establishing the LQE, therefore the practical application of LQE is limited. This paper considers the establishment of a link quality estimation model based on unbalanced link samples, which can be realized based on a small amount of data, but still relies on offline data collection.

VII. FUTURE WORK

It can be seen from Fig. 5, Fig. 6, and Fig. 7 that the PSO algorithm can find the global optimum for IWELM, which proves the effectiveness of the proposed method. However, due to the limitation of the traditional PSO algorithm and uncertainty of link quality sample weight, there is a redundant time in the optimization process. During these times, the fitness value of the LQE-IWELM has not been improved. In the future, we will improve the efficiency of the PSO algorithm to solve this problem and reduce the computational complexity of LQE-IWELM.

REFERENCES

[1] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2224–2287, 3rd Quart., 2019.

[2] X. Cheng, D. Ciuonzo, and P. S. Rossi, "Multibit decentralized detection through fusing smart and dumb sensors based on rao test," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 2, pp. 1391–1405, Apr. 2020.

[3] R. Rucco, A. Sorriso, M. Liparoti, G. Ferraioli, P. Sorrentino, M. Ambrosanio, and F. Baselice, "Type and location of wearable sensors for monitoring falls during static and dynamic tasks in healthy elderly: A review," *Sensors*, vol. 18, no. 5, p. 1613, May 2018, doi: 10.3390/s18051613.

[4] D. Ciuonzo, P. S. Rossi, and P. Willett, "Generalized Rao test for decentralized detection of an uncooperative target," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 8, no. 1, pp. 84–97, Jan. 2004.

[5] Y. Zhang, Y. Mu, Y. Liu, T. Zhang, and Y. Qian, "Deep learning-based beamspace channel estimation in mmWave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 9, no. 12, pp. 2212–2215, Dec. 2020, doi: 10.1109/LWC.2020.3019321.

[6] M. Liu, G. Liao, N. Zhao, H. Song, and F. Gong, "Data-driven deep learning for signal classification in industrial cognitive radio networks," *IEEE Trans. Ind. Informat.*, early access, Apr. 10, 2020, doi: 10.1109/TII.2020.2985715.

[7] M. Liu, K. Yang, N. Zhao, Y. Chen, H. Song, and F. Gong, "Intelligent signal classification in industrial distributed wireless sensor networks-based IIoT," *IEEE Trans. Ind. Informat.*, early access, Aug. 17, 2020, doi: 10.1109/TII.2020.3016958.

[8] A. Tripathi, H. P. Gupta, T. Dutta, R. Mishra, K. K. Shukla, and S. Jit, "Coverage and connectivity in WSNs: A survey, research issues and challenges," *IEEE Access*, vol. 6, pp. 26971–26992, 2018.

[9] J. M. Reason and J. M. Rabaey, "A study of energy consumption and reliability in a multi-hop sensor network," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 6, pp. 26971–26992, Jun. 2018.

[10] K. S. Adu-Manu, N. Adam, C. Tapparello, H. Ayatollahi, and W. Heinzelman, "Energy-harvesting wireless sensor networks (EH-WSNs): A review," *ACM Trans. Sensor Netw.*, vol. 14, no. 2, pp. 1–50, Jul. 2018.

[11] N. Baccour, A. Koubâa, H. Youssef, and M. Alves, "Reliable link quality estimation in low-power wireless networks and its impact on tree-routing," *Ad Hoc Netw.*, vol. 27, pp. 1–25, Apr. 2015.

[12] S. R. Afzal, S. Stuijk, M. Nabi, and T. Basten, "Effective link quality estimation as a means to improved end-to-end packet delivery in high traffic mobile ad hoc networks," *Digit. Commun. Netw.*, vol. 3, no. 3, pp. 150–163, Aug. 2017.

[13] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," in *Proc. 1st Int. Conf. Embedded Netw. Sensor Syst. (SenSys)*, Los Angeles, CA, USA, Nov. 2003, pp. 14–27.

[14] R. D. Gomes, D. V. Queiroz, A. C. L. Filho, I. E. Fonseca, and M. S. Alencar, "Real-time link quality estimation for industrial wireless sensor networks using dedicated nodes," *Ad Hoc Netw.*, vol. 59, pp. 116–133, May 2017.

[15] T. Jayasri and M. Hemalatha, "Link quality estimation for adaptive data streaming in WSN," *Wireless Pers. Commun.*, vol. 94, no. 3, pp. 1543–1562, Jun. 2017.

[16] W. Liu, Y. Xia, R. Luo, and S. Hu, "Lightweight, fluctuation insensitive multi-parameter fusion link quality estimation for wireless sensor networks," *IEEE Access*, vol. 8, pp. 28496–28511, 2020.

[17] W. Sun, W. Lu, Q. Li, L. Chen, D. Mu, and X. Yuan, "WNN-LQE: Wavelet-neural-network-based link quality estimation for smart grid WSNs," *IEEE Access*, vol. 5, pp. 12788–12797, 2017.

[18] J. Shu, S. Liu, L. Liu, L. Zhan, and G. Hu, "Research on link quality estimation mechanism for wireless sensor networks based on support vector machine," *Chin. J. Electron.*, vol. 26, no. 2, pp. 377–384, Mar. 2017.

[19] G.-B. Huang, "What are extreme learning machines? Filling the gap between frank Rosenblatt's dream and John von Neumann's puzzle," *Cognit. Comput.*, vol. 7, no. 3, pp. 263–278, Jun. 2015.

[20] Y. Zhang, Y. Yuan, F. Guo, Y. Wang, and G. Wang, "Improving the multimodal probabilistic semantic model by ELM classifiers," *J. Franklin Inst.*, vol. 355, no. 4, pp. 1967–1990, Mar. 2018.

[21] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?" *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3133–3181, Oct. 2014.

[22] K. Chen, Q. Lv, Y. Lu, and Y. Dou, "Robust regularized extreme learning machine for regression using iteratively reweighted least squares," *Neurocomputing*, vol. 230, pp. 345–358, Mar. 2017.

[23] W. Zong, G.-B. Huang, and Y. Chen, "Weighted extreme learning machine for imbalance learning," *Neurocomputing*, vol. 101, pp. 229–242, Feb. 2013.

[24] M. S. Nobile, P. Cazzaniga, D. Besozzi, R. Colombo, G. Mauri, and G. Pasi, "Fuzzy self-tuning PSO: A settings-free algorithm for global optimization," *Swarm Evol. Comput.*, vol. 39, pp. 70–85, Apr. 2018.

[25] A. Bildea, O. Alphand, F. Rousseau, and A. Duda, "Link quality estimation with the Gilbert-Elliot model for wireless sensor networks," in *Proc. IEEE 26th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Hong Kong, Aug. 2015, pp. 2049–2054.

[26] R. J. Meinhold and N. D. Singpurwalla, "Understanding the Kalman filter," *Amer. Statistician*, vol. 37, no. 2, pp. 123–127, Oct. 1981.

[27] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, Dec. 2006.

[28] J. Cao, K. Zhang, M. Luo, C. Yin, and X. Lai, "Extreme learning machine and adaptive sparse representation for image classification," *Neural Netw.*, vol. 81, pp. 91–102, Sep. 2016.

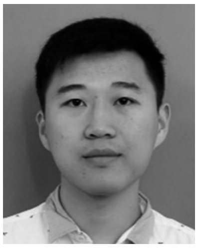
[29] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012.

[30] R. Anderson, Y. S. Koh, and G. Dobbie, "Classifying imbalanced road accident data using recurring concept drift," in *Proc. Australas. Conf. Data Mining*. Singapore: Springer, Nov. 2019, pp. 143–155.

[31] P. S. Oliveto, J. He, and X. Yao, "Time complexity of evolutionary algorithms for combinatorial optimization: A decade of results," *Int. J. Autom. Comput.*, vol. 4, no. 3, pp. 281–293, Jul. 2007.

[32] M. Bucurica, R. Dogaru, and I. Dogaru, "A comparison of extreme learning machine and support vector machine classifiers," in *Proc. IEEE Int. Conf. Intell. Comput. Commun. Process. (ICCP)*, Cluj-Napoca, Romania, Sep. 2015, pp. 471–474.

LINLAN LIU was born in 1968. She received the B.Sc. degree in computer application from the National University of Defense Technology. She is currently a Professor with Nanchang Hangkong University. Her research interests include wireless sensor networks, software engineering, and distributed systems. She is also a member of CCF.



HUI LV was born in Nanchang, Jiangxi, China, in 1996. He is currently pursuing the master's degree with Nanchang Hangkong University. His research interest includes wireless sensor networks.

JIANGBO XU was born in Tongcheng, Anhui, China, in 1991. He received the master's degree in wireless sensor networks from Nanchang Hangkong University, in 2018. His research interest includes wireless sensor networks.

JIAN SHU was born in 1964. He received the M.Sc. degree in computer networks from Northwestern Polytechnical University. He is currently a Professor with Nanchang Hangkong University. His research interests include wireless sensor networks, embedded systems, and software engineering. He is also a Senior Member of CCF.