

## Research Article

# A Robust Data Interpolation Based on a Back Propagation Artificial Neural Network Operator for Incomplete Acquisition in Wireless Sensor Networks

Mingshan Xie <sup>1,2</sup>, Mengxing Huang <sup>1</sup>, Yong Bai <sup>1</sup>, Zhuhua Hu <sup>1</sup> and Yanfang Deng<sup>1</sup>

<sup>1</sup>State Key Laboratory of Marine Resource Utilization in South China Sea, College of Information Science & Technology, Hainan University, Haikou 570228, China

<sup>2</sup>College of Network, Haikou University of Economics, Haikou 571127, China

Correspondence should be addressed to Yong Bai; [bai@hainu.edu.cn](mailto:bai@hainu.edu.cn)

Received 13 July 2018; Revised 7 October 2018; Accepted 16 October 2018; Published 20 December 2018

Guest Editor: Aniello Falco

Copyright © 2018 Mingshan Xie et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The data space collected by a wireless sensor network (WSN) is the basis of data mining and data visualization. In the process of monitoring physical quantities with large time and space correlations, incomplete acquisition strategy with data interpolation can be adopted to reduce the deployment cost. To improve the performance of data interpolation in such a scenario, we proposed a robust data interpolation based on a back propagation artificial neural network operator. In this paper, a neural network learning operator is proposed based on the strong fault tolerance of artificial neural networks. The learning operator is trained by using the historical data of the data acquisition nodes of WSN and is transferred to estimate the value of physical quantities at the locations where sensors are not deployed. The experimental results show that our proposed method yields smaller interpolation error than the traditional inverse-distance-weighted interpolation (IDWI) method.

## 1. Introduction

The purpose of a wireless sensor network (WSN) is to obtain the data field or data space of the physical world as accurate and complete as possible through acquisition technology. It is an important part of forecasting, simulation, and prediction to obtain the spatial-temporal distribution information of the monitored object accurately. However, in some scenarios, WSN can take an incomplete acquisition strategy, due to the development cost of the sensing device and the deployment environment factor, energy limitation, equipment aging, and other factors, or because it is not necessary to collect the data in each corner of the monitoring area. The incomplete acquisition strategy is divided into three cases: (a) spatial incomplete acquisition strategy—the actual collected area is smaller than the interested area or the actual acquisition location set is part of the entire acquisition locations in the monitoring area; (b) temporal incomplete acquisition strategy—the actual collection time

period is less than the time period in which all devices work. The sleeping schedule is a temporal incomplete acquisition strategy. (c) Incomplete acquisition of attributes—the actual physical quantities collected are less than the interested physical quantities.

Because the constraints of interpolation are relatively small, it is more appropriate to use the interpolation algorithm to complete or refine the entire data space in the case of spatial incomplete acquisition. The interpolation completion algorithm takes advantage of the strong correlation between the data in the data space. At present, data interpolation is the main method to complement the data space of the entire region. In [1], Ding and Song used the linear interpolation theory to evaluate the working status of each node and the whole network coverage case. In [2], Alvarez et al. applied interpolation techniques for creating detailed pollution maps.

However, WSN is often affected by many unfavorable factors. For example, it is usually arranged in a harsh

environment, the node failure rate is relatively high, it is very difficult to physically replace the failure sensor, and the wireless communication network is susceptible to interference, attenuation, multipath, blind zone, and other unfavorable factors. Data is prone to errors, security is not guaranteed, etc. Therefore, WSN data interpolation technology must be highly fault tolerant to ensure high credibility and robustness of the completed data space [3].

Data interpolation is used to predict and estimate the information at an unknown location by means of using known information. Transfer learning opens up a new path for data interpolation. The goal of transfer learning is to extract useful knowledge from one or more source domain tasks and apply them to new target tasks. It is essentially the transfer and reuse of knowledge. Transfer learning has gradually received the attention of scholars. In [4], the authors are motivated by the idea of transfer learning. They proposed a novel domain correction and adaptive extreme learning machine (DC-AELM) framework with transferring capability to realize the knowledge transfer for interference suppression. To improve the radar emitter signal recognition, Yang et al. use transfer learning to obtain the robust feature against signal noise rate (SNR) variation in [5]. In [6], the authors discuss the relationship between transfer learning and other related machine learning techniques such as domain adaptation, multitask learning, and sample selection bias, as well as covariate shift.

Artificial neural networks have strong robustness. One of the requirements to ensure the accuracy of transfer learning is the robustness of the learning algorithm. Many scholars have combined the neural artificial network with transfer learning. In [7] Pan et al. propose a cascade convolutional neural network (CCNN) framework based on transfer learning for aircraft detection. It achieves high accuracy and efficient detection with relatively few samples. In [8], Park et al. showed that the transfer learning of the ImageNet pretrained deep convolutional neural networks (DCNN) can be extremely useful when there are only a small number of Doppler radar-based spectrogram data.

The research aim of data interpolation of WSN is to complete the data space of the entire monitoring area by using the limited data of the acquisition node to estimate the data at the locations where sensors are deployed. However, data errors of WSN due to various reasons have great impact on the accuracy of data interpolation. Due to the strong robustness of an artificial neural network, an artificial neural network learning operator is generated by using historical measurement data of limited data acquisition nodes in this paper. At the same time, this paper applies the learning property of the artificial neural network to the inverse-distance-weighted interpolation method, which is conducive to improving precision and accuracy of data interpolation. On the basis of analyzing the demand of network models, this paper proposes a robust data interpolation based on a back propagation artificial neural network operator for incomplete acquisition in wireless sensor networks. The detailed steps of the algorithm are discussed in detail, and the algorithm is analyzed based on the MATLAB tool and the measured data provided by the Intel Berkeley

Research laboratory [9]. The experimental results are good evaluations of the fault-tolerant performance and lower error of our proposed method.

The main contributions of this paper are as follows:

- (1) Aiming at the data loss and disturbance error, we propose a fault-tolerant complementary algorithm based on the robustness of the artificial neural network
- (2) We combine the artificial neural network with the inverse-distance-weighted interpolation algorithm to obtain a novel back propagation artificial neural network operator
- (3) We use the inverse tangent function to reconcile the relationships among multiple prediction values

The rest of the paper is organized as follows. In Section 2, we summarize the related work. Section 3 introduces the interpolation model in the condition of data error. Section 4 presents how to construct the learning operator set of data acquisition nodes. Section 5 elaborates how to generate the interpolation by the method based on the back propagation artificial neural network operator. We will show the experimental results of our proposed methods compared with the inverse-distance-weighted interpolation in Section 6. The conclusions are given in Section 7.

## 2. Related Works

*2.1. The Inverse-Distance-Weighted Interpolation Method.* The inverse-distance-weighted interpolation (IDWI) method is also called “inverse-distance-weighted averaging” or the “Shepard Method.” The interpolation scheme is explicitly expressed as follows:

Given  $n$  locations whose the plane coordinates are  $(x_i, y_i)$  and the values are  $z_i$ , where  $i = 1, 2, \dots, n$ , the interpolation function is

$$f(x, y) = \begin{cases} \frac{\sum_{i=1}^n (d_i/\text{dist}_i^p)}{\sum_{i=1}^n (1/\text{dist}_i^p)} & \text{if } (x, y) \neq (x_i, y_i), \quad i = 1, 2, \dots, n, \\ d_i & \text{if } (x, y) = (x_i, y_i), \quad i = 1, 2, \dots, n, \end{cases} \quad (1)$$

where  $\text{dist}_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$  is the horizontal distance between  $(x, y)$  and  $(x_i, y_i)$ , where  $i = 1, 2, \dots, n$ .  $p$  is a constant greater than 0, called the weighted power exponent.

It can easily be seen that the interpolation  $f(x, y) = \sum_{i=1}^n (d_i/\text{dist}_i^p) / \sum_{i=1}^n (1/\text{dist}_i^p)$  of the location  $(x, y)$  is the weighted mean of  $\text{dist}_1, \text{dist}_2, \dots, \text{dist}_n$ .

The application of inverse-distance-weighted interpolation is more extensive. Because of its simple computation and having less constraints, the interpolation precision is higher. In [10], Kang and Wang use the Shepard family of interpolants to interpolate the density value of any given computational point within a certain circular influence domain of the point. In [11], Hammoudeh et al. use a

Shepard interpolation method to build a continuous map for a new WSN service called the map generation service.

From (1), we can see that the IDWI algorithm is sensitive to the accuracy of the data. However, WSN is usually deployed in a harsh environment, and the probability of data being collected is high. The error tolerance of the interpolation algorithm is required. This paper improves the robustness of interpolation algorithm on the basis of the inverse-distance interpolation algorithm.

**2.2. Artificial Neural Network.** An artificial neural network (ANN) is an information processing paradigm that is inspired from biological nervous systems, such as how the brain processes information. ANNs, like people, have the ability to learn by example. An ANN is configured for a specific application, such as pattern recognition, function approximation, or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist among neurons. This is true for ANNs as well. They are made up of simple processing units which are linked by weighted connections to form structures that are able to learn relationships between sets of variables. This heuristic method can be useful for nonlinear processes that have unknown functional forms. The feed forward neural networks or the multilayer perceptron (MLP) among different networks is most commonly used in engineering. MLP networks are normally arranged in three layers of neurons; the input layer and output layer represent the input and output variables, respectively, of the model; laid between them is one or more hidden layers that hold the network's ability to learn nonlinear relationships [12].

The natural redundancy of neural networks and the form of the activation function (usually a sigmoid) of neuron responses make them somewhat fault tolerant, particularly with respect to perturbation patterns. Most of the published work on this topic demonstrated this robustness by injecting limited (Gaussian) noise on a software model [13]. Velazco et al. proved the robustness of ANN with respect to bit errors in [13]. Venkitaraman et al. proved that neural network architecture exhibits robustness to the input perturbation: the output feature of the neural network exhibits the Lipschitz continuity in terms of the input perturbation in [14]. Artificial neural networks have strong robustness. The robustness of the algorithm is a requirement to ensure the accuracy of artificial neural network operator transferring. We can see from the literatures [7, 8] that operator transferring can combine well with an artificial neural network. The learning operator in this paper also adopts an artificial neural network algorithm.

### 3. Problem Formulations

**3.1. Data Acquisition Nodes.** To assess the entire environmental condition, WSN collects data by deploying a certain number of sensors in the location of the monitoring area; thus, the physical quantity of the monitoring area is discretized and the monitoring physical quantity is digitized.

*Definition 1. Interested locations:* in the whole monitoring area, they are the central locations of the segment of the monitoring area that we are interested in.

Sensors can be deployed at each interested location to capture data. The data at all interested locations reflects the information status of the entire monitoring area.

We assumed that  $S$  is the set of the interested locations, which is a matrix of  $1 \times n$ .

$$S = \{s_1(x_1, y_1), s_2(x_2, y_2), s_3(x_3, y_3), \dots, s_n(x_n, y_n)\}, \quad (2)$$

where  $s_i(x_i, y_i)$  is the  $i$ th interested location in the monitoring area.  $(x_i, y_i)$  is the coordinates of the  $s_i$  interested location in the monitoring area. Due to the difficulty and limitation of deployment, not all the interested locations can deploy sensors. This paper studies the spatial incomplete collection strategy. We select a subset of  $S$  as the data acquisition node.  $\|S\|$  represents the potential of a set, that is, the number of elements of  $S$ .  $\|S\| = n$ .

*Definition 2. Data acquisition nodes:* they are the interested locations where the sensors are actually deployed.

The sensors are deployed in these interested locations, so that these locations become data acquisition nodes. The all-data acquisition nodes in the monitoring area act as the sensing layer of the WSN, and the information is transmitted to the server through the devices of the transport layer.

In our research, when the sensors are not deployed at the interested location, we use zero as a placeholder to replace the data acquisition node. When the interested location becomes the data acquisition node, we use 1 as a placeholder to replace the data acquisition node. Suppose that  $M$  is the set of data acquisition nodes.

$$M = \{s_i \mid s_i = 1, s_i \in S, i = 1 \dots n\}, \quad (3)$$

where  $s_i = 1$  represents the  $i$ th interested location where the sensors are deployed.  $\|M\|$  indicates the potential of the set  $M$ . It reflects the total number of elements in the set of data acquisition nodes. This paper investigates the case where multiple types of sensors are deployed at a data acquisition node. The data that the data acquisition node  $s_i$  can correctly collect at time  $t_l$  is defined as  $d_{s_i}(t_l) = [d_{s_i,1}(t_l), d_{s_i,2}(t_l), \dots, d_{s_i,k}(t_l)]$ , which is  $k$  dimensional data that is perceived by the  $i$ th data acquisition location  $s_i$  in  $S$ . The physical quantity of temperature, humidity, etc. can be measured at the same time. The data is defined as

$$D_M(t_l) = \left\{ d_{s_i}(t_l) \mid s_i \in M, d_{s_i}(t_l) \in \mathbb{R}^k \right\}. \quad (4)$$

If  $\|M\| < \|S\|$ , then WSN implements incomplete coverage; if  $\|M\| = \|S\|$ , then WSN implements complete coverage. The data of the interested location where the sensors are not deployed can be assessed by the data of the data acquisition node. The interested location where the sensors are not deployed is indicated as non-data acquisition location.

**3.2. Data Acquisition Error.** Because the wireless communication network is susceptible to interference, attenuation, multipath, blind zone, and other unfavorable factors, the data error rate is high. Nodes and links in wireless sensor networks are inherently erroneous and unpredictable. The error data which greatly deviated from the ideal truth value is divided into two types: data loss and data disturbance.

(1) *Data Loss.* These reasons, such as nodes cannot work, links cannot be linked, or data cannot be transmitted, cause the data of the corresponding data acquisition nodes to not reach the sink node.

(2) *Data Disturbance.* Due to the failure, the local function of the WSN is in an incorrect (or ambiguous) system state. This state may cause a deviation between the data measured by the sensors of the corresponding data acquisition node and the true value, or the signal is disturbed during the transmission, and the data received at the sink node is deviant from the true value. The data that corresponds to the data acquisition node is not the desired result. The collected data oscillate in the data area near the true value. In this paper, we assumed that the data disturbance obeys the Gauss distribution.

The main idea of our method is based on the fundamental assumption that the sensing data of WSN are regarded as a vector indexed by the interested locations and recovered from a subset sensing data. As demonstrated in Figure 1, the data acquisition consists of two stages: the data-sensing stage and the data-recovering stage. At the data-sensing stage, instead of deploying sensors and sensing data at all interested locations, a subset of interested locations which are the shaded ones in the second subfigure is selected to sense physical quantity and deliver the sensing data to the sink node at each data collection round. Some locations are drawn by the fork in the second subfigure because their data is lost or disturbed. The fork represented the data errors. When the hardware and software of the network node failures or the communication links of the network are broken, the set of sensors which encounter data errors is only the subset of  $M$ . At the data-recovering stage, the sink node receives these incomplete sensing data over some data collection rounds shown in the third subfigure in which the shaded entries represent the valid sensing data and the white entries are unknown. And then we could use them to recover the complete data by our method.

Here we adopt a mask operator  $A()$  to represent the process of collecting data-encountering errors:

$$A(D_M(t_l)) = B(t_l), \quad (5)$$

where  $B$  is the data set that is actually received for interpolation. For the sake of clarity, the operator  $A()$  can be specified as a vector product as follows:

$$A(D_M(t_l)) = Q \odot D_M(t_l), \quad (6)$$

where  $\odot$  denotes the product of two vectors, i.e.,  $b_{s_i}(t_l) = d_{s_i}(t_l) \times q_{s_i}$ .  $Q$  is a vector of  $1 \times \|M\|$ .  $b_{s_i}$  indicates the data

that is actually received by the  $i$ th data acquisition node for interpolation.

$$q_{s_i} = \begin{cases} 1, & \text{if the data is not error,} \\ 0, & \text{if the data is lost,} \\ \frac{\text{No}(d_{s_i}(t_l), \sigma^2)}{d_{s_i}}, & \text{if the data is disturbed,} \end{cases} \quad (7)$$

where  $\text{No}(d_{s_i}(t_l), \sigma^2)$  represents Gauss distribution with a mean of  $d_{s_i}(t_l)$  and a variance of  $\sigma^2$ .

In this paper, the error rate of the received data is defined as follows:  $\|Q\|_{q_{s_i} \neq 1} / \|M\|$ , where  $\|Q\|_{q_{s_i} \neq 1}$  is the number of non-1 elements.

**3.3. Completion of Data Space with Interpolation.** Due to conditional restrictions, there is no way to deploy sensors in  $S - M$ . The data generated in  $S - M$  can be estimated by interpolation based on the data in  $M$ . The data space of the entire monitoring area is set to

$$D_S(t_l) = D_M(t_l) \cup \hat{D}_{S-M}(t_l), \quad (8)$$

where  $D_M(t_l)$  represents the data set collected from the data acquisition node in  $M$  at epoch  $t_l$ .  $D_{S-M}(t_l)$  is the data set collected from non-data acquisition locations  $S - M$  at the epoch  $t_l$  in the ideal case, if the sensors are deployed in the non-data acquisition location.  $\hat{D}_{S-M}(t_l)$  is the data interpolation set based on the data in  $M$ .

The problem definition is how to process the  $D_M(t_l)$  data so that  $\hat{D}_{S-M}(t_l)$  is as close as possible to  $D_{S-M}(t_l)$ , that is, the problem of minimizing the error between  $\hat{D}_{S-M}(t_l)$  and  $D_{S-M}(t_l)$ . Its mathematical expression is as follows:

$$\begin{aligned} \arg \min \quad & \|\hat{D}_{S-M}(t_l) - D_{S-M}(t_l)\| \\ \text{s.t.} \quad & Q \neq 1, \end{aligned} \quad (9)$$

where  $\|\cdot\|$  is the Euclidean norm form used to evaluate the error between  $\hat{D}_{S-M}(t_l)$  and  $D_{S-M}(t_l)$ ;  $Q \neq 1$  indicates that  $Q$  is not a full 1 matrix.

$$\begin{aligned} \hat{D}_{S-M}(t_l) &= \left\{ \hat{d}_{s_j}(t_l) \mid \hat{d}_{s_j}(t_l) \right. \\ &= E\left(\hat{d}'_{s_j}(t_l), \hat{d}''_{s_j}(t_l), \dots, \hat{d}^n_{s_j}(t_l)\right), s_j \in S - M \left. \right\}, \end{aligned} \quad (10)$$

where  $\hat{d}'_{s_j}(t_l), \hat{d}''_{s_j}(t_l), \dots, \hat{d}^n_{s_j}(t_l)$  represents each value close to  $d_{s_j}(t_l)$ . Suppose  $s_i$  is the closest data acquisition node to  $s_j$ . The value received from the data acquisition node nearest the non-data acquisition location is also close to  $d_{s_j}(t_l)$ .

$$\therefore \text{dist}_{s_i \rightarrow s_j} = \min \left( \text{dist}_{M \rightarrow s_j} \right), \quad s_i \in M, \quad (11)$$

where  $\text{dist}_{M \rightarrow s_j}$  indicates the distance from each data acquisition node in  $M$  to the non-data acquisition location  $s_j$ . The

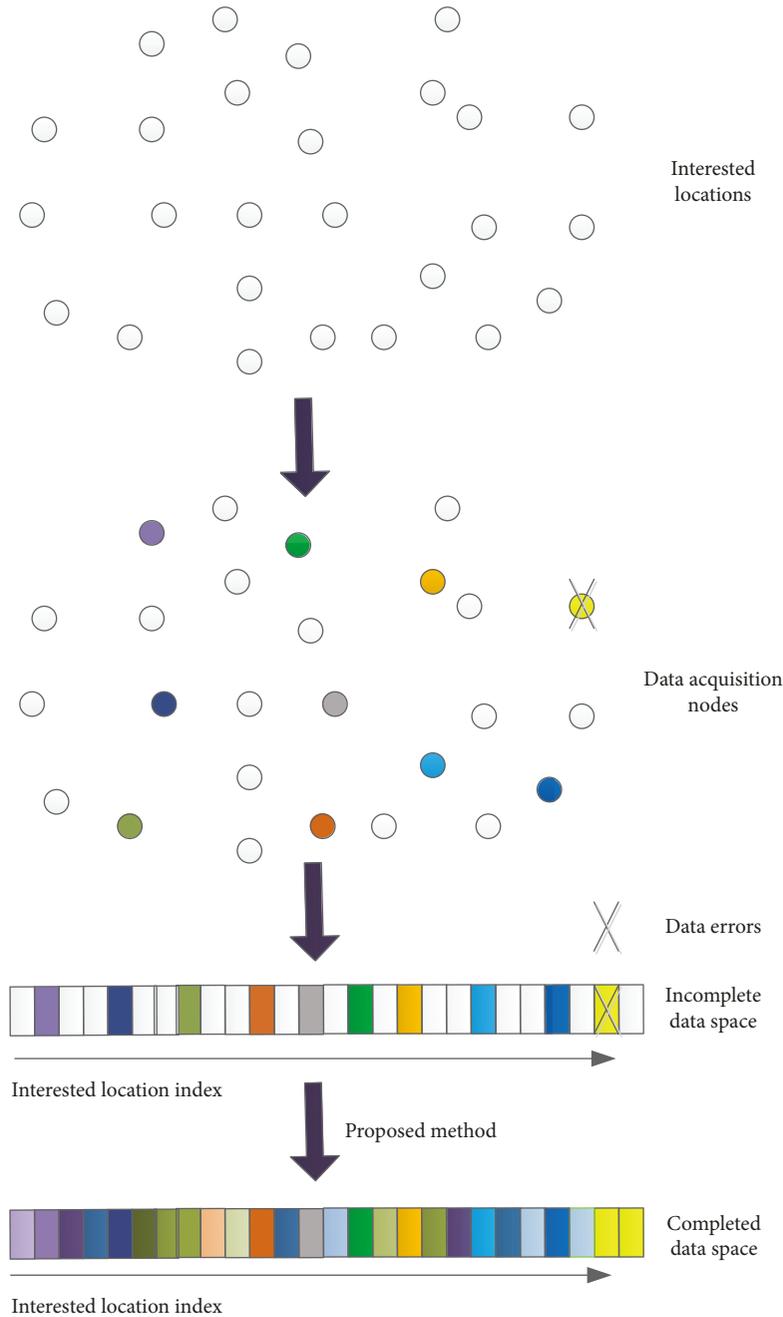


FIGURE 1: Incomplete acquisition process based on our proposed method.

closer the information collected by the node, the greater the correlation [15].

$$\therefore d'_{s_j}(t_l) = d_{s_i}(t_l), \quad (12)$$

where  $d_{s_i}(t_l)$  represents the value of the physical quantity actually collected by the  $s_i$  data acquisition node.

Suppose that  $\hat{d}''_{s_j}(t_l)$  represents the assessed value at the  $j$ th non-data acquisition location, which is obtained by the back propagation artificial neural network operator. In this

paper, we get the  $\hat{d}''_{s_j}(t_l)$  that is close to  $d_{s_j}(t_l)$ . The data set of non-data acquisition locations for interpolation is

$$d''_{s_j}(t_l) = \Upsilon_{s_j}(b_{s_i}(t_l)), \quad s_j \in S - M, s_i \in M, \quad (13)$$

$$\hat{D}_{S-M}(t_l) = \Upsilon_{S-M}(B(t_l)),$$

where  $\Upsilon_{S-M}(\cdot)$  represents the learning operator to assess  $d_{s_j}(t_l)$ .  $B(t_l)$  represents the received data set at the epoch  $t_l$ . We can use the back propagation artificial neural network operator of the data acquisition node closest to the non-

data acquisition location to predict the data of the non-data acquisition location.

#### 4. Learning Operator Set of Data Acquisition Nodes

The mathematical model of the learning operator for the reconstruction is as follows:

$$\begin{aligned} \arg \min \quad & \|\Upsilon(\text{Inp}) - \text{Tar}\| \\ \text{s.t.} \quad & \text{Inp} \neq \text{Tar}. \end{aligned} \quad (14)$$

where Tar represents the learning goals,  $\Upsilon(\cdot)$  is the learning operator of Inp, and Inp can be individuals, variables, and even algorithms or functions, sets, and so on. The input of  $\Upsilon(\cdot)$  is Inp. Inp and Tar are different. If they do not have differences, there is no need to learn for reconstruction. The purpose of learning is to make  $\Upsilon(\text{Inp})$  gradually approach Tar.

Because data of WSN is error-prone, we need fault-tolerant and robust learning operators. The learning operator in this paper uses a back propagation (BP) artificial neural network. We can use data of data acquisition nodes to predict the data of the non-data acquisition location, thus assessing the data space of the entire monitoring area. Because of the strong robustness and adaptability of the artificial neural network, we can use the artificial neural network to interpolate the data of non-data acquisition locations in the case of data error.

The BP artificial neural network is a multilayer (at least 3 levels) feedforward network based on error backpropagation. Because of its characteristics, such as nonlinear mapping, multiple input and multiple output, and self-organizing self-learning, the BP artificial neural network can be more suitable for dealing with the complex problems of nonlinear multiple input and multiple output. The BP artificial neural network model is composed of an input layer, hidden layer (which can be multilayer, but at least one level), and output layer. Each layer is composed of a number of juxtaposed neurons. The neurons in the same layer are not connected to each other, and the neurons in the adjacent layers are connected by means of full interconnection [16].

After constructing the topology of the artificial neural network, it is necessary to learn and train the network in order to make the network intelligent. For the BP artificial neural network, the learning process is accomplished by forward propagation and reverse correction propagation. As each data acquisition node is related to other data acquisition nodes, and each data acquisition node has historical

data, it can be trained through the historical data of data acquisition nodes to generate the artificial neural network learning operator of the data acquisition node.

*4.1. Transform Function of the Input Unit.* The data of the non-data acquisition location  $s_j$  is assessed by using the inverse-distance learning operator of the data acquisition node  $s_i$  closest to  $s_j$ . Because there is spatial correlation between interested locations in space acquisition, in this paper, we adopt the IDWI algorithm combined with the BP artificial neural network algorithm.

At a certain epoch  $t_j$ , the data set of all other data acquisition nodes except the  $s_i$  is as follows:

$$B_{M-s_i}(t_l) = \begin{bmatrix} b_{s_{o,1}}(t_l) & \dots & b_{s_{o,1}}(t_l) \\ \vdots & b_{s_{q,i}}(t_l) & \vdots \\ b_{s_{m,k}}(t_l) & \dots & b_{s_{m,k}}(t_l) \end{bmatrix}, \quad b_s \in M - s_i. \quad (15)$$

In this paper, the inverse-distance weight is used to construct the transform function. The transform function  $h_{s_i}(B_{M-s_i}(t_l))$  of the input unit  $B_{M-s_i}(t_l)$  of the artificial neural network with data acquisition node  $s_i$  is as follows:

$$h_{s_i}(B_{M-s_i}(t_l)) = \frac{(1/\text{dist}_{s_q \rightarrow s_i}^\gamma) \times B_{M-s_i}(t_l)}{\sum_{s_q \in M}^{\|M\|-1} (1/\text{dist}_{s_q \rightarrow s_i}^\gamma)}, \quad s_q \in M - s_i, s_i \in M, \quad (16)$$

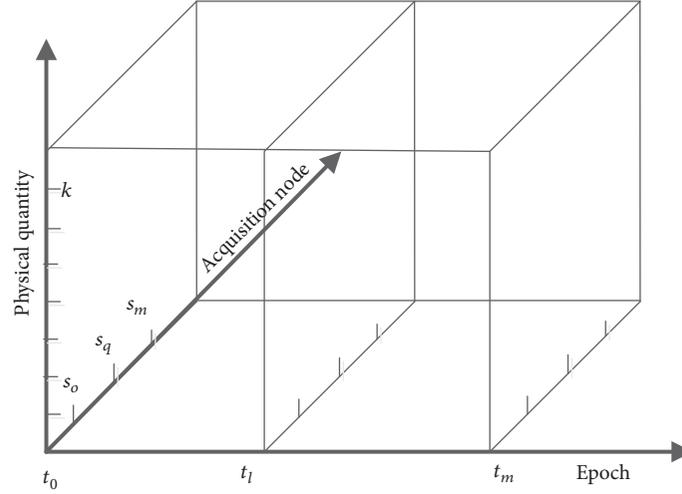
where  $\text{dist}_{s_q \rightarrow s_i}$  represents the distance between the data acquisition node  $s_i$  and the non-data acquisition location  $s_q$ .  $\gamma$  represents the weighted power exponent of the distance reciprocal.  $\sum_{s_q \in M}^{\|M\|-1} (1/\text{dist}_{s_q \rightarrow s_i}^\gamma)$  represents the sum of the weighted reciprocal of the distance from the data acquisition node  $s_i$  to the rest of data acquisition nodes.

The artificial neural network requires a training set. We take the historical data of the period  $T$  of all data acquisition nodes as the training set.  $T = [t_0, t_1, \dots, t_l, \dots, t_m]$ . In practical engineering, it is feasible for us to get the data from data acquisition nodes in a period to learn.

In this paper, data collected from all data acquisition nodes are used as the training set  $B_{M-s_i}(T)$ .  $B_{M-s_i}(T)$  is a three-order tensor, as shown in the following Figure 2.  $B_{M-s_i}(T) \in R^{k \times \|M-s_i\| \times t_m}$ . The elements of the training set are indexed by physical quantities, acquisition node, and epoch.

The  $B_{M-s_i}(T)$  matrix is obtained by

$$B_{M-s_i}(T) = \begin{bmatrix} \left[ \begin{array}{ccc|ccc} b_{s_{o,1}}(t_0) & \dots & b_{s_{o,1}}(t_0) & b_{s_{o,1}}(t_1) & \dots & b_{s_{o,1}}(t_1) \\ \vdots & b_{s_{q,i}}(t_0) & \vdots & \vdots & b_{s_{q,i}}(t_1) & \vdots \\ b_{s_{m,k}}(t_0) & \dots & b_{s_{m,k}}(t_0) & b_{s_{m,k}}(t_1) & \dots & b_{s_{m,k}}(t_1) \end{array} \right] & \dots & \left[ \begin{array}{ccc|ccc} b_{s_{o,1}}(t_l) & \dots & b_{s_{o,1}}(t_l) & b_{s_{o,1}}(t_m) & \dots & b_{s_{o,1}}(t_m) \\ \vdots & b_{s_{q,i}}(t_l) & \vdots & \vdots & b_{s_{q,i}}(t_m) & \vdots \\ b_{s_{m,k}}(t_l) & \dots & b_{s_{m,k}}(t_l) & b_{s_{m,k}}(t_m) & \dots & b_{s_{m,k}}(t_m) \end{array} \right] \end{bmatrix}, \quad b_s \in M - s_i. \quad (17)$$

FIGURE 2: The model of  $B_{M-s_i}(T)$ .

In actual engineering,  $B_{M-s_i}(T)$  has a lot of noise. It is necessary to clean it. If it is not cleaned, it will affect the estimation accuracy of the artificial neural network and the precision of learning. Data cleaning is the process of reexamining and verifying data to remove duplicate information, correct existing errors, and provide data consistency. We can use various filter algorithms to get  $\bar{B}_{M-s_i}(T)$ .

**4.2. Artificial Neural Network Learning Operator.** The sensing data of the real network include several physical quantities, such as temperature, humidity, and illumination. Usually, a variety of sensors are deployed at an acquisition node, and multiple physical quantities are collected at the same time. Physical quantities at the same acquisition node have the same temporal and spatial variation trend, but in the process of recovery using the artificial neural network, each quantity has a mutual promotion effect. The distance between data acquisition nodes and non-data acquisition locations is very easy to obtain. Based on the inverse-distance interpolation algorithm and BP artificial neural network algorithm, we propose a multidimensional inverse-distance BP artificial neural network learning operator.

The BP artificial neural network, which is the most widely used one-way propagating multilayer feedforward network, is characterized by continuous adjustment of the network connection weight, so that any nonlinear function can be approximated with arbitrary accuracy. The BP artificial neural network is self-learning and adaptive and has robustness and generalization. The training of the BP artificial neural network is the study of “supervisor supervision.” The training process of the BP artificial neural network is shown in Figure 3.

For the input information, it is first transmitted to the node of the hidden layer through the weighted threshold summation, and then the output information of the hidden node is transmitted to the output node through the weighted threshold summation after the operation of the transfer

function of each element. Finally, the output result is given. The purpose of network learning is to obtain the right weights and thresholds. The training process consists of two parts: forward and backward propagation.

The BP artificial neural network is a multilayer feed-forward network trained by the error backpropagation algorithm. The BP artificial neural network structure is a multilayer network structure, which has not only input nodes and output nodes but also one or more hidden nodes. As demonstrated in Figure 3, according to the prediction error, the  $v$  and  $w$  are continuously adjusted, and finally the BP artificial neural network learning operator for reconstructing the data of data acquisition node  $s_i$  can be determined, as shown in the part  $Y_{s_i}(\cdot)$  of Figure 3.

The input layer of the BP artificial neural network learning operator is the data collected at a certain time at all acquisition nodes except the  $s_i$  acquisition node, which is the vector  $h_{s_i}(\bar{B}_{M-s_i}(t_l))$ . The input of the  $j$ th neuron in the hidden layer is calculated by

$$\Sigma_j = \sum_{q=1}^{\|M\|-1} \sum_{p=1}^k v_{q,j,p} h_{s_i}(\bar{B}_{M-s_i}(t_l)) - \theta_j, \quad (18)$$

where  $v_{q,j,p}$  represents the weight between the  $(q,p)$ th input neurons and the  $j$ th neurons in the hidden layer. The  $(q,p)$ th input neurons are  $h_{s_i}(b_{s_q,k}(t_l))$ , that is, the  $k$ -th dimension data of the  $s_q$  acquisition node at time  $t_l$ .  $\theta_j$  is the threshold of the  $j$ th neuron in the hidden layer. The output of neurons in the hidden layer is calculated by

$$ho_j = \frac{1}{1 + e^{-\Sigma_j}}. \quad (19)$$

Similarly, the output of each neuron in the output layer is set to  $o_p$ .

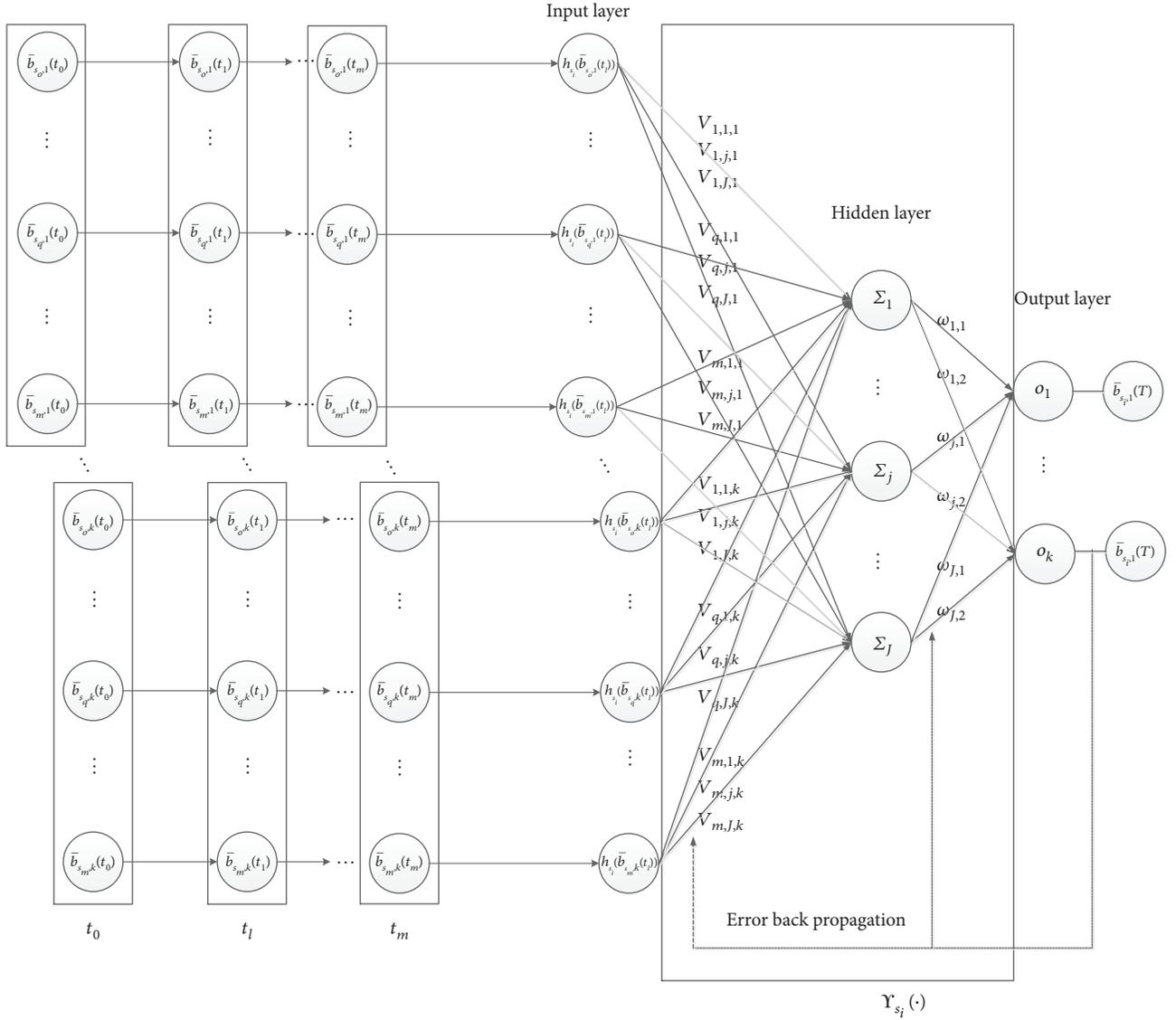


FIGURE 3: The training process of the BP artificial neural network.

The sum of squared errors of all neurons in the output layer is the objective function value optimized by the BP artificial neural network algorithm, which is calculated by

$$E = \frac{1}{(\|M\| - 1) \times k \times k} \sum_{q=1}^{\|M\|-1} \sum_{p=1}^k \sum_{p=1}^k (\bar{b}_{s_i,p} - o_p)^2, \quad (20)$$

where  $\bar{b}_{s_i,p}$  represents the expected output value of neuron  $o_p$  in the output layer, corresponding to the value of the  $p$ th physical quantity sensed by the data acquisition node  $s_i$ . According to the gradient descent method, the error of each neuron in the output layer is obtained by

$$E_p = o_p \times (1 - o_p) \times (\bar{b}_{s_i,p} - o_p). \quad (21)$$

The weights and thresholds of the output layer can be adjusted by

$$\Delta\omega = \partial \times E_p \times ho_j, \quad (22)$$

$$\Delta\theta_p = \partial \times E_p, \quad (23)$$

where  $\partial \in [0, 1]$  is the learning rate, which reflects the speed of training and learning. Similarly, the weight and threshold of the hidden layer can be obtained. If the desired results cannot be obtained from the output layer, it needs to constantly adjust the weights and thresholds, gradually reducing the error. The BP neural network has strong self-learning ability and can quickly obtain the optimal solution.

The mathematical model of Figure 3 is as follows:

$$f_{o_p} \left( \sum_{j=1}^J \sum_{p=1}^k \omega_j f_{\Sigma_j} \left( \sum_{q=1}^{\|M\|-1} \sum_{p=1}^k v_{q,j,p} h_{s_i} (\bar{B}_{M-s_i}(T)) \right) \right) \longrightarrow \bar{b}_{s_i,p}(T), \quad (24)$$

where  $\bar{B}_{s_i}(T)$  is the training data, that is, the historical data of acquisition nodes except  $s_i$  in the period  $T$ .  $h_{s_i}(\cdot)$  is the transform function of the input layer.  $v_{q,j}$  is the weight of the input layer.  $f_{\Sigma_j}(\cdot)$  is the transfer function of the input layer to the hidden layer.  $w$  is the weight of the hidden layer to the output layer.  $f_{o_p}(\cdot)$  is the transfer function of the hidden layer to the input layer.  $\bar{b}_{s_i}(T)$  is the ‘‘supervisor supervision’’ that is the historical data of acquisition node  $s_i$  in the period  $T$ .

From (14) and (24), the following formula can be obtained:

$$\Upsilon_{s_i}(\cdot) = f_{o_p} \left( \sum_{j=1}^J \omega_j f_{\Sigma_j} \left( \sum_{q=1}^{\|M\|-1} \sum_{p=1}^k v_{q,j,p}(\cdot) \right) \right) \bar{b}_{s_i}(T), \quad (25)$$

where  $\Upsilon_{s_i}(\cdot)$  represents the inverse-distance BP artificial neural network learning operator of acquisition node  $s_i$ .  $\bar{b}_{s_i}(T)$  represents that this neural network operator  $\Upsilon_{s_i}(\cdot)$  is a learning operator trained by the data of  $s_i$  as tutor information.

The data collected by each data acquisition node in the monitoring area is related to each other. The data of a data acquisition node can be learned by inputting the data of other data acquisition nodes into the learning operator. The learning operator set of data acquisition nodes in the whole monitoring area is as follows:

$$\Upsilon(\cdot) = \{ \Upsilon_{s_i}(\cdot) \mid s_i \in M \}. \quad (26)$$

## 5. Interpolation at Non-Data Acquisition Locations

We transfer the BP artificial neural network operators from data acquisition nodes to non-data acquisition locations. We can use the learning operator of the data acquisition node closest to the non-data acquisition location to estimate the data of the non-data acquisition location. There are four ways to implement the learning operator transferring, including sample transferring, feature transferring, model transferring, and relationship transferring. In this paper, model transferring (also called parameter transferring) is used to better combine with the BP artificial neural network; that is, we can use the pretrained BP artificial neural network to interpolate. The BP artificial neural network learning operator has strong robustness and adaptability. In the interested area, if the physical quantity collected is highly correlated, and there is no situation in which the physical quantity changes

drastically between the various interested locations, the learning operator can be transferred.

Since the construction of the artificial neural network requires historical data for training, and there is no historical data at the non-data acquisition location deployed without sensors, it is very difficult to construct the artificial neural networks of the non-data acquisition location. However, we can predict the physical quantities of non-data acquisition locations by using the learning operator from the nearest data acquisition node.

*5.1. Transform Function Corresponding to Nonacquisition Location  $s_j$ .* First, we use the IDWI method to construct the transform function. The transform function of the artificial neural network input unit  $s_q$  of the acquisition node  $s_i$  is defined as  $h_{s_i}(B_{M-s_i}(t_l))$ :

$$h_{s_i}(B_{M-s_i}(t_l)) = \frac{B_{M-s_i}(t_l) / \text{dist}_{s_q \rightarrow s_j}^\gamma}{\sum_{s_q \in M-s_i}^{\|M\|-1} (1 / \text{dist}_{s_q \rightarrow s_j}^\gamma)}, \quad (27)$$

where  $\text{dist}_{s_q \rightarrow s_j}$  represents the distance between  $s_q$  and  $s_j$ .  $\gamma$  represents the weighted exponentiation of the distance reciprocal.  $\sum_{s_q \in M-s_i}^{\|M\|-1} (1 / \text{dist}_{s_q \rightarrow s_j}^\gamma)$  represents the sum of the weighted reciprocal of the distance from the non-data acquisition location  $s_j$  to the rest of the acquisition nodes. The physical data collected from the remaining  $\|M\| - 1$  data acquisition nodes except  $s_i$  are as follows:

$$B_{M-s_i} = \{ b_{s_q}(t_l) \mid s_q \in M - s_i \}. \quad (28)$$

*5.2. Learning Operator Transferring.* Since the data of the data acquisition node closest to the non-data acquisition location is important to the non-data acquisition location, and its data is most correlated with the data of the non-data acquisition location, we estimate the data of the non-data acquisition location with data from the nearest data acquisition node and its learning operator.

Since the data we are targeting is spatially correlated, the smaller the distance between the two interested locations is, the smaller the difference of the collected data is. Conversely, the greater the difference of the collected data is. We can use the data of the data acquisition node close to the non-data acquisition location to assess the data at the non-data acquisition location.

Because the BP artificial neural network learning operator has strong robustness and adaptability, we can transfer the inverse BP artificial neural network learning operator of  $s_i$  to  $s_j$  in order to estimate the data at  $s_j$  in this paper. The transform function  $h(\cdot)$  does not change with time, and it is a function of the distance between the sampling locations.  $h_{s_i}(B_{M-s_i}(t_l)) \longrightarrow h_{s_j}(B_{M-s_i}(t_l))$ , the number of input parameters is constant, while the input parameters vary with time. So the change of the transform function will not affect the

trend of input parameters. The change of the transform function will not affect the accuracy of prediction, and the operator transferring can be implemented.

$$\begin{aligned}\widehat{d}_{s_j}''(t_l) &= \Upsilon_{s_i} \left( h_{s_j} (B_{M-s_i}(t_l)) \right) \\ &= f_{s_i} \left( \sum_{j=1}^J \omega_j f_{z_j} \left( \sum_{q=1}^{\|M\|} \nu_{qj} h_{s_j} (B_{M-s_i}(t_l)) \right) \right).\end{aligned}\quad (29)$$

The assessment network based on the BP artificial neural network operator is shown in Figure 4.

In this paper, our proposed method is improved on the basis of the BP artificial neural network. According to spatial correlation, the physical quantities of the monitored and interested location close to the data acquisition nodes  $s_q$  can be approximated by learning the operator of  $s_q$ .

Suppose  $d_{s_j}$  represents the estimated value of the physical quantity of the interested location  $s_j$ . Due to conditional restrictions, no sensor is deployed at the interested location  $s_j$ . We choose the physical quantity of the data acquisition node  $s_i$  nearest to the non-data acquisition location  $s_j$  for estimation. We can use Algorithm 1 to achieve the determination of  $s_i$ .

**5.3. Assessment at the Non-Data Acquisition Location.**  $s_i$  is the nearest data acquisition node to  $s_j$ . We can use the learning operator of the data acquisition node  $s_i$  to estimate  $d_{s_j}$ . This paper is an improvement on the inverse-distance interpolation method. Because  $s_i$  is closest to  $s_j$ , the correlation of their data is the largest. The data collected actually by  $s_i$  have the greatest impact on the predictive value at  $s_j$ .

$$\begin{aligned}d_{s_j} &= E \left( \widehat{d}'_{s_j}(t_l), \widehat{d}''_{s_j}(t_l) \right) \\ &= \alpha \times \widehat{d}'_{s_j}(t_l) + \beta \times \widehat{d}''_{s_j}(t_l), \quad s_i \in M, s_j \in S - M,\end{aligned}\quad (30)$$

where  $\alpha$  and  $\beta$  denote the weight of  $\widehat{d}'_{s_j}(t_l)$  and  $\widehat{d}''_{s_j}(t_l)$ , respectively, to estimate physical quantities at  $s_j$ .  $\alpha + \beta = 1$ .

$\widehat{d}'_{s_j}(t_l)$  is the value of the actual measurement, so its credibility is higher. The closer  $s_i$  is to  $s_j$ , the greater the correlation between physical quantity at  $s_i$  and the data collected by  $s_i$  based on spatial correlation.

We assume that  $\text{dist}_{s_i \rightarrow s_j}$  represents the distance between  $s_i$  and  $s_j$ . The influence weight of data collected by  $s_i$  on the assessment of  $s_j$  decreases with increasing  $\text{dist}_{s_i \rightarrow s_j}$ . Conversely, the greater the distance  $\text{dist}_{s_i \rightarrow s_j}$ , the smaller the impact. The change field of  $\text{dist}_{s_i \rightarrow s_j}$  is  $[0, +\infty)$ . We find that the inverse tangent function is an increasing function on  $[0, +\infty)$ . The function curve is shown in Figure 5.

We use the boundedness and monotonically increasing characteristics of the arctangent function curve and are

inspired by the idea of IDWI. The formula for calculating  $\beta$  is as follows:

$$\beta = \frac{\arctan \left( \frac{\text{dist}_{s_i \rightarrow s_j}}{\pi/2} \right)}{\pi/2}, \quad \text{dist}_{s_i \rightarrow s_j} \in [0, +\infty). \quad (31)$$

We limit the value of  $\beta$  to the interval  $[0, 1]$ . When  $\text{dist}_{s_i \rightarrow s_j}$  is close to 0,  $\beta$  is close to 0, and  $1 - \beta$  is close to 1, then the data measured by the data acquisition point is closer to the value at the non-data acquisition location. When  $\text{dist}_{s_i \rightarrow s_j} = 0$ , it means that the sensors have deployed in  $s_j$ , and we do not need other values calculated by the prediction algorithm and directly use the actual measured data.

$$\begin{aligned}\widehat{d}_{s_j} &= \frac{\arctan \left( \frac{\text{dist}_{s_i \rightarrow s_j}}{\pi/2} \right)}{\pi/2} \times \widehat{d}''_{s_j}(t_l) \\ &+ \left( 1 - \frac{\arctan \left( \frac{\text{dist}_{s_i \rightarrow s_j}}{\pi/2} \right)}{\pi/2} \right) \times \widehat{d}'_{s_j}(t_l), \quad s_i \in M, s_j \in S - M.\end{aligned}\quad (32)$$

If the interested location of the interpolation is still far from the nearest data acquisition node, then this algorithm will cause a large error. Since we are using the sensor placement based on the iterative dividing four subregions, the sensors of the data acquisition node are omnidirectional throughout the space, not only concentrated in a certain domain. The error is not too large.

## 6. Experiments and Evaluation

**6.1. Parameter Setup.** The data set we used is the measured data provided by the Intel Berkeley Research lab [9]. The data is collected from 54 data acquisition nodes in the Intel Berkeley Research lab between February 28th and April 5th, 2004. In this case, the epoch is a monotonically increasing sequence number from each mote. Two readings from the same epoch number were produced from different motes at the same time. There are some missing epochs in this data set. Mote IDs range from 1–54. In this experiment, we selected the data of these 9 motes as a set of interested locations, because these 9 points have the same epoch.

$$\begin{aligned}S = \{ &s_7(22.5, 8), s_{18}(5.5, 10), s_{19}(3.5, 13), s_{21}(4.5, 18), \\ &s_{22}(1.5, 23), s_{31}(15.5, 28), s_{46}(34.5, 16), s_{47}(39.5, 14), \\ &s_{48}(35.5, 10) \}.\end{aligned}\quad (33)$$

When the environment of the monitoring area is not very complicated, the physical quantity of acquisition is a very spatial correlation, and it is feasible to use the placement based on the iterative dividing four subregions to deploy sensors. In this experiment, we use the method of the iterative dividing four subregions to select the interested locations from  $S$  as data acquisition nodes. The closest  $s_i(x_i, y_i)$  ( $s_i(x_i, y_i) \in S$ ) to the deployment location

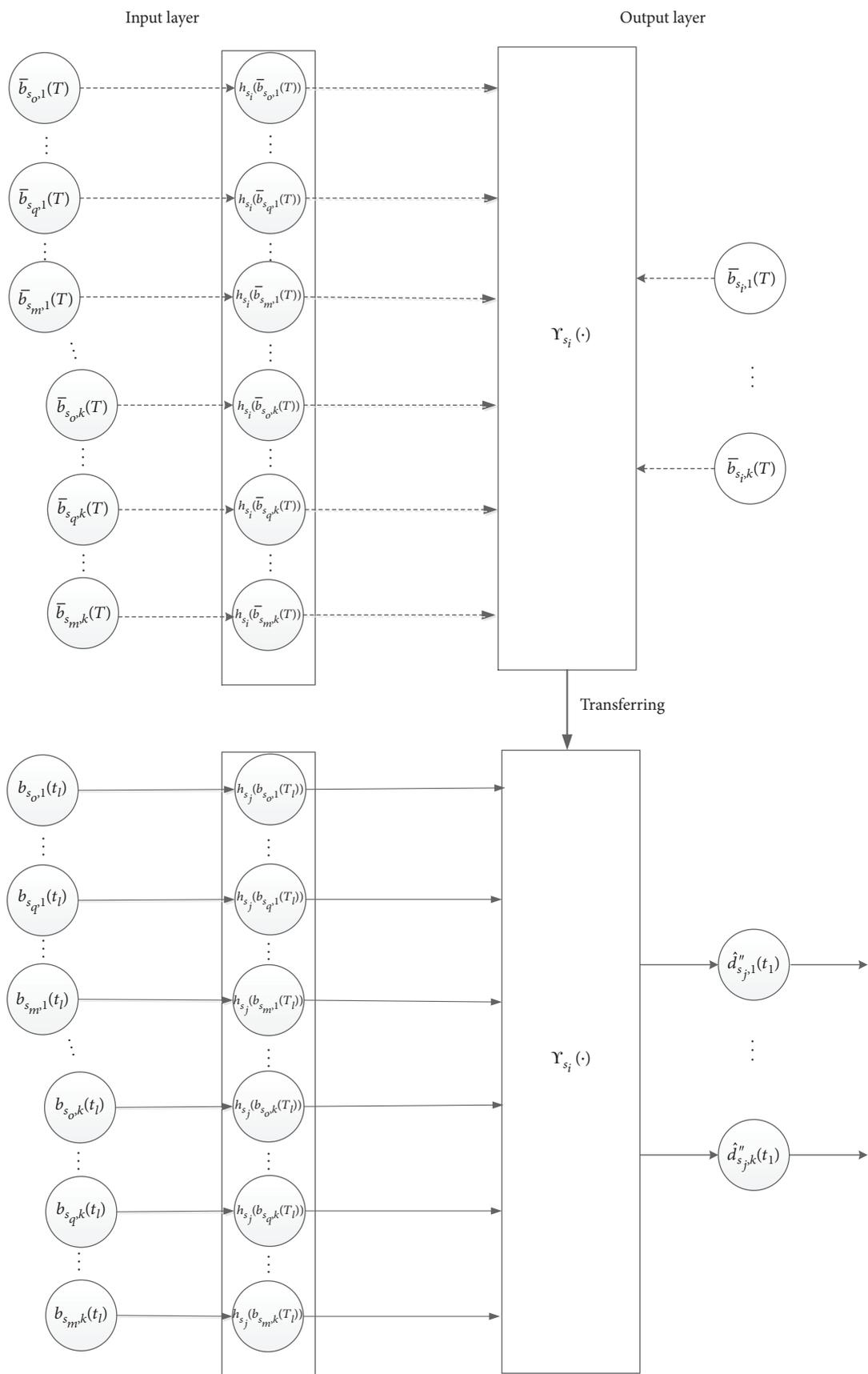


FIGURE 4: The assessment network based on the BP artificial neural network operator.

generated by the method of the iterative dividing four subregions is selected in  $M$ .

In this experiment, we set the dimension of the collected data to 2, that is,  $k=2$ . We take two physical quantities: temperature and humidity. Data acquisition nodes need at least two, i.e.,  $\|M\| \geq 2$ . Because one data acquisition node should be the closest node from the interested location for interpolation, and  $s_i \notin \emptyset$ , the data of other acquisition nodes should be used as the training set of the artificial neural network.  $M - s_i \neq \emptyset$ ,  $\|M - s_i\| \geq 1$ .

The epochs we chose are:  $T = \{33, 35, 42, 44, 51, 56, 61, 62, 70, 73, 86, 88, 89, 91, 95, 99, 102, 107, 112, 113, 123, 124, 127\}$ . These data whose epoch is an element in  $T$  are used as a training set epoch. Unfortunately, the actual sampling sensing data are always corrupted and some values go missing. We need real clean sensing data to train the BP artificial neural networks to improve the interpolation precision.

In order to not lose the trend of data change in the process of data filtering, we simply use the mean processing with a filter window on the temporal direction of the measurements as the clean sensing data. If  $|d_{s_i}(t-1) - d_{s_i}(t)| \gg \delta \times |\bar{d}_{M-s_i}(t-1) - \bar{d}_{M-s_i}(t)|$ , then  $d_{s_i}(t)$  can be replaced by the value calculated

$$d_{s_i}(t) = \frac{d_{s_i}(t-1) + d_{s_i}(t+1)}{2}, \quad (34)$$

where  $\delta$  is the adjusting coefficient. In this experiment, we take  $\delta = 10$ .

The filtered result of the temperature measurements is shown in Figure 6(a), and the actual sensing data is shown in Figure 6(b). The filtered result of the humidity measurements is shown in Figure 7(a), and the actual sensing data is shown in Figure 7(b).

When comparing the test results, we need that the epoch of the data supplied by each mote is the same. Because in the actual application, the data space of the monitoring area we built is the data space at a certain moment. We assumed that  $t_l = 138$ . In this experiment, the time we selected was the 138th epoch for interpolation. We compare the actual collected value at the 138th epoch with the interpolation calculated by algorithms.

To evaluate the accuracy of the reconstructed measurement, we choose the mean relative error (MRE). It reflects the precision of the estimated data relative to the measured data. The formula for the calculation of MRE is as follows [17]:

$$\text{MRE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{d(x_i) - \hat{d}(x_i)}{d(x_i)} \right|, \quad (35)$$

where  $d(x_i)$  is the actual acquisition value of the  $i$ th sensor. Correspondingly,  $\hat{d}(x_i)$  is the assessed value.  $n$  is the total number of data acquisition nodes.

**6.2. Results of the Experiment.** This experiment is in a small range, the physical quantity of the collected data is highly correlated, and there is no situation in which the physical

```

Input:  $M, s_j(x_j, y_j)$ 
Output:  $s_i$ 
Initialize mindist
For each  $s_i$  in  $M$ 
     $\text{dist}_{s_j \rightarrow s_q} = \sqrt{(x_j - x_q)^2 + (y_j - y_q)^2}$ 
    if  $\text{mindist} > \text{dist}_{s_j \rightarrow s_q}$ 
        then  $i = q, \text{mindist} = \text{dist}_{s_j \rightarrow s_q}$ 
    end if
End for

```

ALGORITHM 1: Culling  $s_i$  from  $M$ .

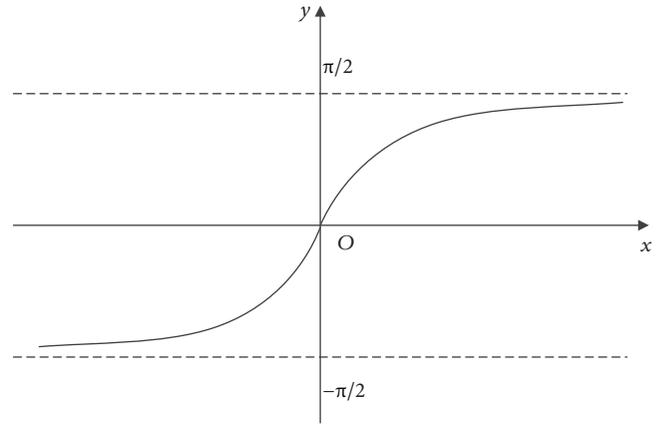


FIGURE 5: Inverse tangent function curve.

quantity changes drastically between the various interested locations, so the learning operator can be transferred.

In the case of data loss, we compare our method with the inverse-distance-weighted interpolation algorithm in terms of interpolation accuracy. Then  $q_{s_i} = 0$ . Since the data acquisition nodes for data loss are random, we conducted 20 tests for a more accurate comparison. Obviously, it is necessary that the number of acquisition node points for data loss be less than or equal to the total number of acquisition nodes. The results are shown in Figures 8 and 9.

As can be seen from Figures 8 and 9, as the proportion of lost data in all collected data decreases, the error of interpolation is gradually reduced. The curve of the proposed algorithm is relatively flat, indicating that it is less affected by data loss.

In the case of data loss, especially when the number of data acquisition nodes is relatively small, the interpolation error of our algorithm is much smaller than that of the inverse-distance-weighted interpolation.

In the case of data disturbance, we compare our method with the inverse-distance-weighted interpolation algorithm in terms of interpolation accuracy. Then  $q_{s_i} = \text{No}(d_{s_i}(t_l), \sigma^2)/d_{s_i}$ . For the interpolation of temperature, we use the mean temperature of all acquisition nodes as the mean value of the Gauss distribution. We set the parameters

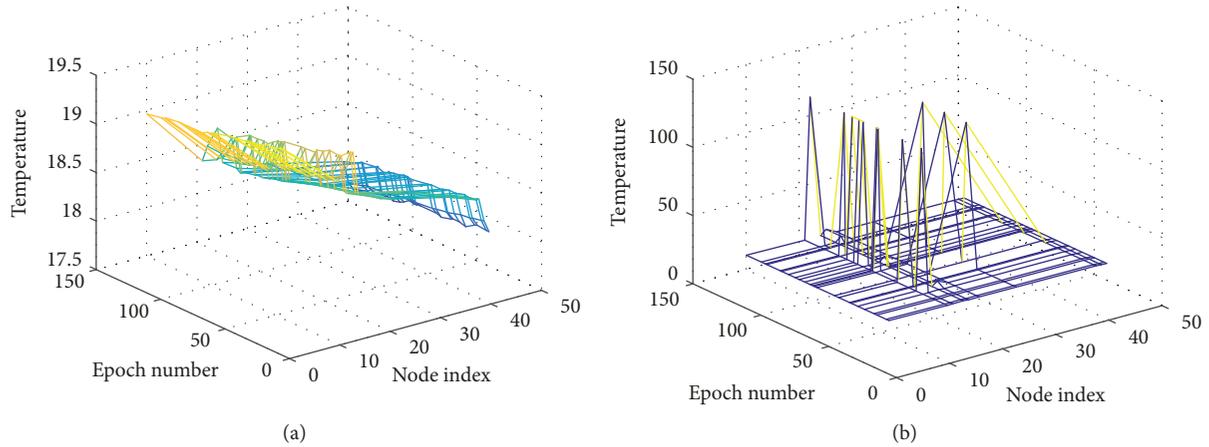


FIGURE 6: The temperature sensing data from the Intel Berkeley Research lab: (a) the filtered sensing data; (b) the actual sensing data.

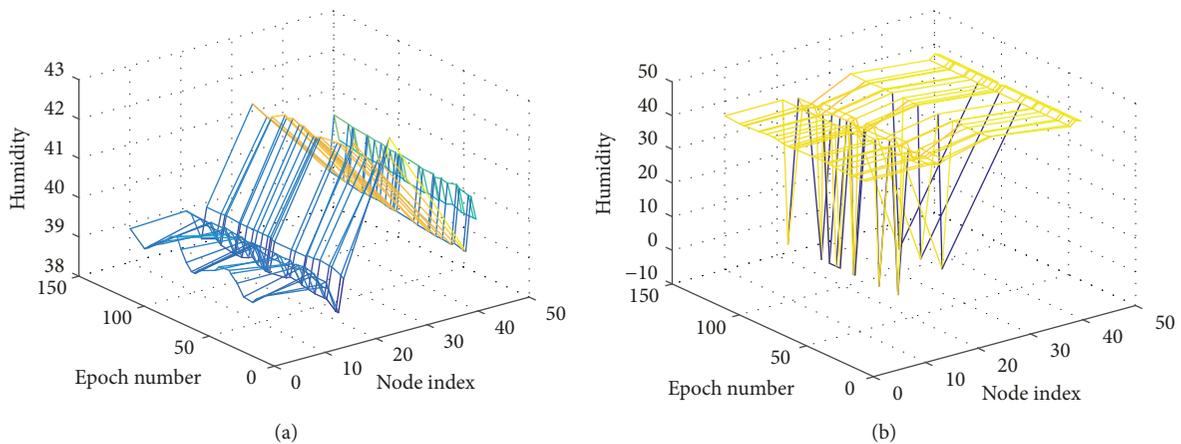


FIGURE 7: The humidity sensing data from the Intel Berkeley Research lab: (a) the filtered sensing data; (b) the actual sensing data.

as  $d_{s_i}(t_1) = 18.47895$  and  $\sigma^2 = 18$ ; for the interpolation of humidity, we use the mean humidity of all acquisition nodes as the mean value of the Gauss distribution. We set the parameters as  $d_{s_i}(t_1) = 39.44691$  and  $\sigma^2 = 39$ . Again, we did 20 tests. The results are shown in Figures 10 and 11.

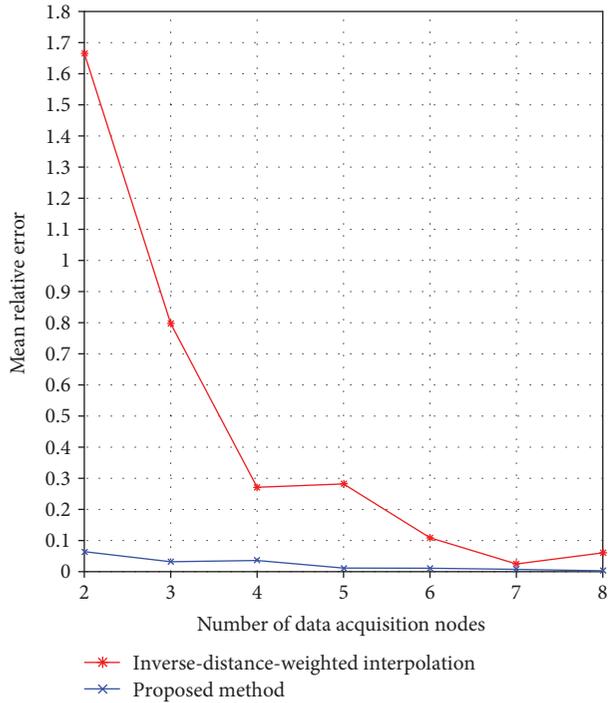
It can be seen from Figures 10 and 11 that the variation curve of the proposed algorithm is relatively flat, while the curve of the inverse-distance-weighted interpolation algorithm fluctuates greatly. The interpolation error of the IDWI algorithm is not only affected by data disturbance but also affected by the deployment location of the data acquisition nodes. When 7 acquisition nodes are deployed, the error of the IDWI algorithm is the smallest. Because sensor placement based on the iterative dividing four subregions is near uniform deployment when 7 acquisition nodes are deployed, the interpolation error is small.

In the case where there are not many acquisition nodes, the density of the acquisition nodes where data disturbance occurs is large, so the error of interpolation is more prominent. The number of sensors that can be deployed increases, and the error of interpolation is also reduced.

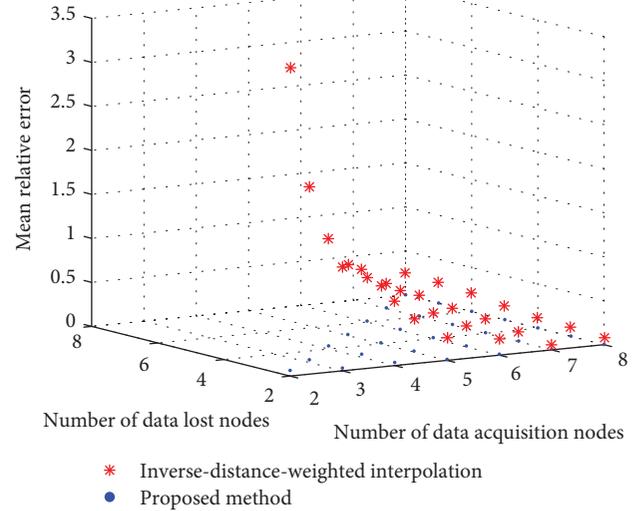
As we can see from Figures 8–11, our algorithm is insensitive to errors and strong in robustness when the data is wrong, while the inverse-distance interpolation method has a great influence on the interpolation accuracy of the error data. In particular, when the error rate is high, the relative error of our algorithm is much lower than that of the inverse-distance interpolation algorithm. When the error rate is 50%, our algorithm has a relative error of 0.1 and the relative error of the inverse-distance interpolation algorithm is 3.5, as shown in Figure 9(a).

## 7. Conclusions

In this paper, we proposed a robust data interpolation based on a back propagation artificial neural network operator for incomplete acquisition in a wireless sensor network. Under the incomplete collection strategy of WSN, the effect of complete acquisition can be approximately obtained by interpolation. In the case of limited data acquisition nodes, the data of the acquisition nodes are used to train to obtain the learning operator. Then, the learning operator of the acquisition node

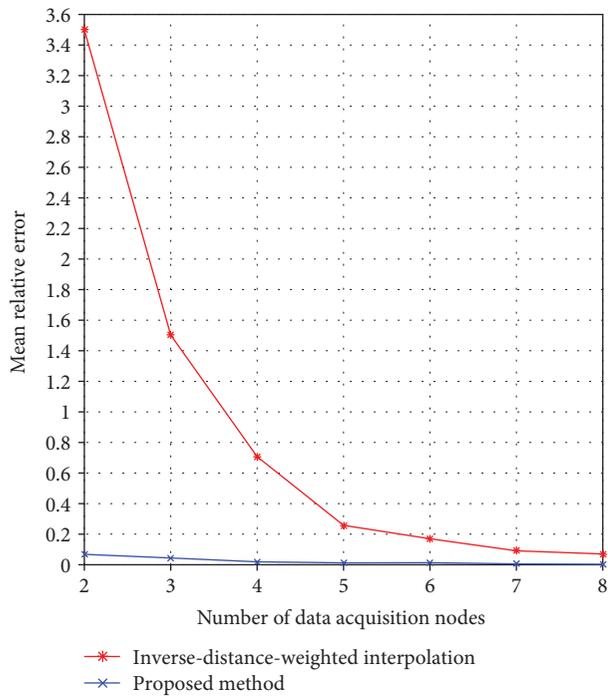


(a)

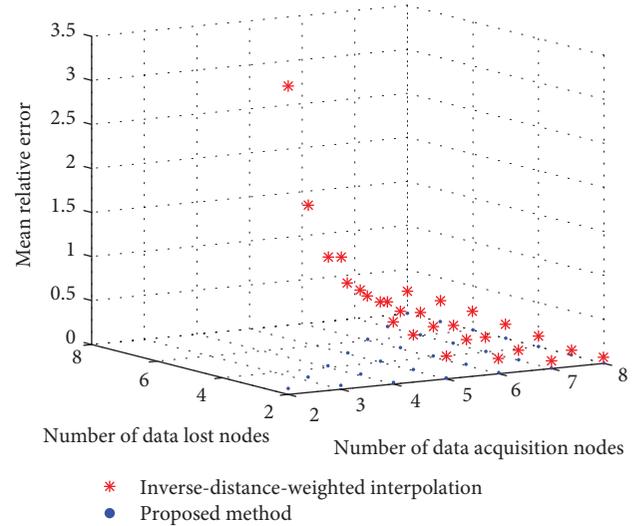


(b)

FIGURE 8: The temperature sensing data from the Intel Berkeley Research lab: (a) data loss occurred at 1 data acquisition node; (b) data loss occurred at data acquisition nodes increased from 2 to 8.



(a)



(b)

FIGURE 9: The humidity sensing data from the Intel Berkeley Research lab: (a) data loss occurred at 1 data acquisition node; (b) data loss occurred at data acquisition nodes increased from 2 to 8.

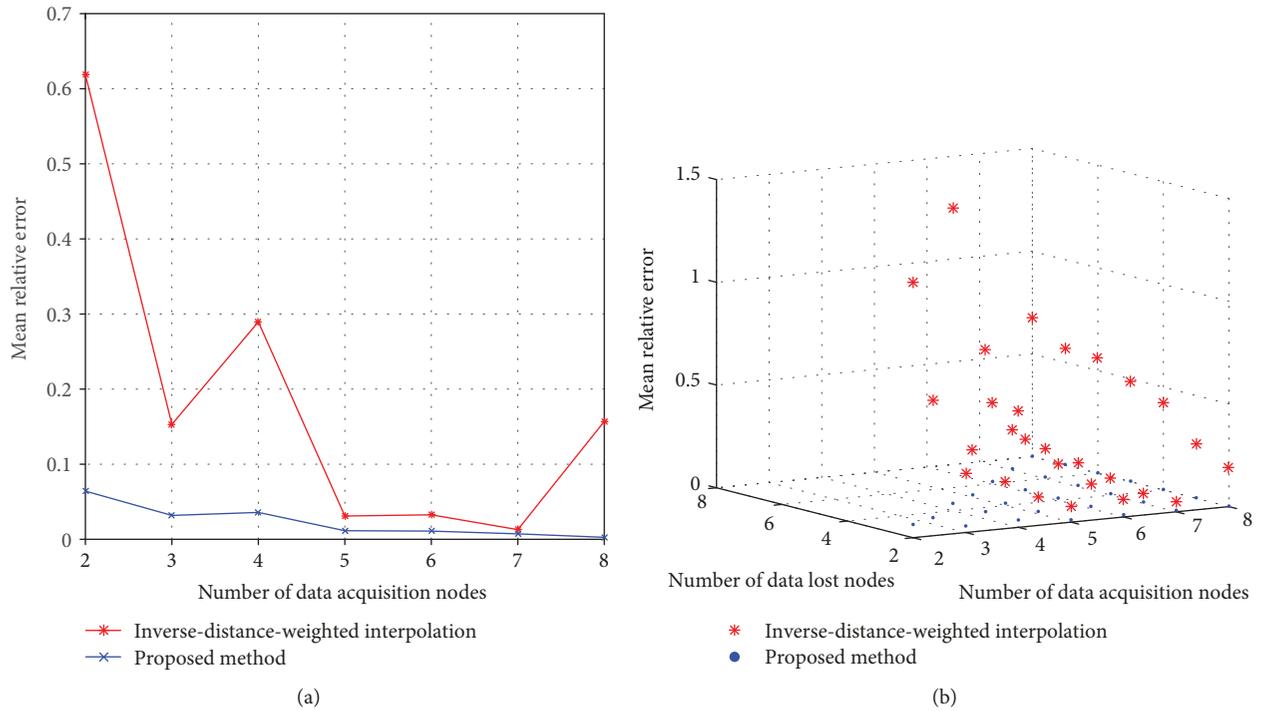


FIGURE 10: The temperature sensing data from the Intel Berkeley Research lab: (a) data disturbance occurred at 1 data acquisition node; (b) data disturbance occurred at data acquisition nodes increased from 2 to 8.

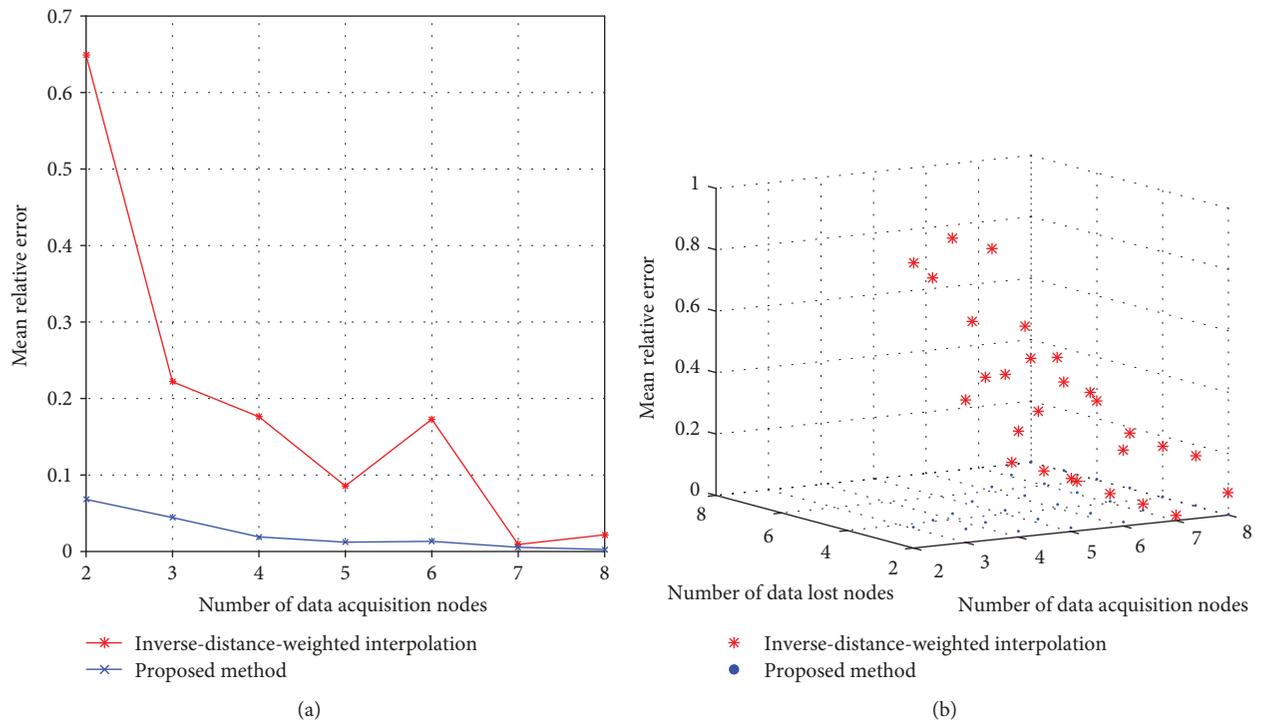


FIGURE 11: The humidity sensing data from the Intel Berkeley Research lab: (a) data disturbance occurred at 1 data acquisition node; (b) data disturbance occurred at data acquisition nodes increased from 2 to 8.

closest to the non-data acquisition location is transferred to the non-data acquisition location for interpolation. Considering that the data collected by WSN is prone to error, we

analyzed the reasons for the error. In order to improve the fault tolerance and robustness of the interpolation algorithm, we proposed a BP artificial neural network learning

operator. From the experiments, we demonstrated that our algorithm has strong robustness, and it has a lower error in the case of data errors collected by the WSN. This method has strong potential for practical data visualization, data analysis, WSN monitoring, etc.

### Data Availability

The measured data that support the findings of this study are supplied by Intel Berkeley Research lab. They are available from <http://db.csail.mit.edu/labdata/labdata.html>.

### Conflicts of Interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgments

This work was financially supported by the National Natural Science Foundation of China (Grant Nos. 61561017 and 61462022), the Innovative Research Projects for Graduate Students of Hainan Higher Education Institutions (Grant No. Hyb2017-06), the Hainan Province Major Science & Technology Project (Grant No. ZDKJ2016015), the project of the Natural Science Foundation of Hainan Province in China (Grant No.617033), the open project of the State Key Laboratory of Marine Resource Utilization in South China Sea (Grant No. 2016013B), and the Key R&D Project of Hainan Province (No. ZDYF2018015).

### References

- [1] F. Ding and A. Song, "Development and coverage evaluation of ZigBee-based wireless network applications," *Journal of Sensors*, vol. 2016, Article ID 2943974, 9 pages, 2016.
- [2] O. Alvear, W. Zamora, C. Calafate, J. C. Cano, and P. Manzoni, "An architecture offering mobile pollution sensing with high spatial resolution," *Journal of Sensors*, vol. 2016, Article ID 1458147, 13 pages, 2016.
- [3] W. Xiao, *The Key Technology of Adaptive Data Fault Tolerance in Wireless Sensor Networks*, National Defense University of Science and Technology, Hunan, China, 2010.
- [4] Z. Liang, F. Tian, C. Zhang, H. Sun, A. Song, and T. Liu, "Improving the robustness of prediction model by transfer learning for interference suppression of electronic nose," *IEEE Sensors Journal*, vol. 18, no. 3, pp. 1111–1121, 2017.
- [5] Z. Yang, W. Qiu, H. Sun, and A. Nallanathan, "Robust radar emitter recognition based on the three-dimensional distribution feature and transfer learning," *Sensors*, vol. 16, no. 3, pp. 289–303, 2016.
- [6] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [7] B. Pan, J. Tai, Q. Zheng, and S. Zhao, "Cascade convolutional neural network based on transfer-learning for aircraft detection on high-resolution remote sensing images," *Journal of Sensors*, vol. 2017, Article ID 1796728, 14 pages, 2017.
- [8] J. Park, R. Javier, T. Moon, and Y. Kim, "Micro-Doppler based classification of human aquatic activities via transfer learning of convolutional neural networks," *Sensors*, vol. 16, no. 12, pp. 1990–2000, 2016.
- [9] <http://db.csail.mit.edu/labdata/labdata.html>.
- [10] Z. Kang and Y. Wang, "Structural topology optimization based on non-local Shepard interpolation of density field," *Computer Methods in Applied Mechanics and Engineering*, vol. 200, no. 49-52, pp. 3515–3525, 2011.
- [11] M. Hammoudeh, R. Newman, C. Dennett, and S. Mount, "Interpolation techniques for building a continuous map from discrete wireless sensor network data," *Wireless Communications and Mobile Computing*, vol. 13, no. 9, pp. 809–827, 2013.
- [12] M. Saberi, A. Azadeh, A. Nourmohammadzadeh, and P. Pazhoheshfar, "Comparing performance and robustness of SVM and ANN for fault diagnosis in a centrifugal pump," in *19th International Congress on Modelling and Simulation*, pp. 433–439, Perth, Australia, 2011.
- [13] R. Velasco, P. Cheynet, J. D. Muller, R. Ecoffet, and S. Buchner, "Artificial neural network robustness for on-board satellite image processing: results of upset simulations and ground tests," *IEEE Transactions on Nuclear Science*, vol. 44, no. 6, pp. 2337–2344, 1997.
- [14] A. Venkitaraman, A. M. Javid, and S. Chatterjee, "R3Net: random weights, rectifier linear units and robustness for artificial neural network," 2018, <https://arxiv.org/abs/1803.04186>.
- [15] C. Zhang, X. Zhang, O. Li et al., "WSN data collection algorithm based on compressed sensing under unreliable links," *Journal of Communication*, vol. 37, no. 9, pp. 131–141, 2016.
- [16] H. L. Chen and W. Peng, "Improved BP artificial neural network in traffic accident prediction," *Journal of East China Normal University (Natural Science)*, vol. 2017, no. 2, pp. 61–68, 2017.
- [17] G. Y. Lu and D. W. Wong, "An adaptive inverse-distance weighting spatial interpolation technique," *Computers & Geosciences*, vol. 34, no. 9, pp. 1044–1055, 2008.

