IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 25, 2018, accepted October 13, 2018, date of publication October 18, 2018, date of current version November 9, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2876701

Multi-Attribute Missing Data Reconstruction Based on Adaptive Weighted Nuclear Norm Minimization in IoT

XIANG YU, XIA FAN, KAN CHEN, AND SIRUI DUAN

School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

Corresponding author: Xia Fan (fanxia1198@163.com)

This work was supported in part by the National Science and Technology Major Program of China under Grant 2017ZX03001004-004, in part by the Research and Innovation Project of Graduate Student of Chongqing City under Grant CYS18239, and in part by Chongqing Research Program of Basic science and Frontier Technology under Grant cstc2016jcyjA0542.

ABSTRACT In the Internet of Things, data sets gathered from sensor nodes are missing a significant fraction of data, owing to noise, collision, unreliable links, and unexpected damage. This phenomenon is more serious in some scenarios, which limits the applications of sensor data. It is therefore necessary to develop methods to reconstruct these lost data with high accuracy. In this paper, a new multi-attribute missing data reconstruction method based on adaptive weighted nuclear norm minimization is developed, and a K-means clustering analysis is embedded into this forecasting process to improve the prediction accuracy. First, to ensure that sensors in one group have similar patterns of measurement, we use a traditional machine learning algorithm, called K-means clustering algorithm to separate sensors into different groups. Second, considering the correlations among multiple attributes of sensor data and its joint low-rank characteristics, we propose an algorithm based on the matrix rank-minimization method of automatic weighted nuclear norm minimization, which adaptively assigns different weights to each singular value simultaneously. Moreover, we use the alternating direction method of multipliers to obtain its optimal solution. Finally, we evaluate the proposed method by using a real sensor data set from the Intel Berkeley Research Laboratory with two missing patterns, namely, random missing pattern and consecutive missing pattern. The simulation results prove that our algorithm performs well even with a small number of samples, and it can propagate through a structure to fill in large missing regions.

INDEX TERMS Data reconstruction, Internet of Things (IoT), K-means, multi-attribute, tensor completion, weighted nuclear norm minimization (WNNM).

I. INTRODUCTION

Owing to advancements in information technology, the new era of the Internet of Things (IoT) is encompassing computing and communication technologies, spanning every aspect of our lives, and it has emerged as one of central issues in our daily lives. The IoT is an intelligent network, it has been defined as a global network with an infrastructure that has self-configuring capabilities. Specifically, the sensors are embedded in various objects and then integrated with the existing Internet to realize the information exchange between human society and physical systems. IoT is to fully utilize the new generation of IT technology in all walks of life, which is called the third wave of the world information industry after computers and the Internet [1]–[3].

IoT has employed many technologies and the core technologies of it mainly include Radio Frequency Identification (RFID), sensor technology, wireless network technology, artificial intelligence and cloud computing [1]. IoT is widely used in conventional Long Term Evolution (LTE), Wi-Fi, ZigBee, wireless sensor network (WSN), as well as device-to-device (D2D) communication [4], transmit antenna selection (TAS) in cooperative networks [5], the low-power wide area network from the LoRa Alliance (LoRaWAN), narrow band IoT (Nb-IoT), Ethernet and many other communications technologies. Therefore, IoT is rapidly transforming into a highly heterogeneous ecosystem that provides inter operability among different types of devices and communications technologies [6]–[8].

Stylized as IoT, it is the interconnection of everyday ''thing,'' such as vehicles, buildings, and other Internet-connected devices embedded with software, electronics, actuators, and sensors, enabling them to send and receive data. Thus, data is the most crucial and significant asset responsible for the functioning of these smart devices. IoT applications gather huge volumes of multiple attribute data from the physical world through all connected sensors and reconstruct environmental data in the cyber world [9]. For instance, scientists have revealed the plant evolution based on wind speed, air humidity, and air temperature data [10]. Volcanic eruptions have been predicted based on temperature and shake data of volcanoes [11].

IoT applications have strict requirements in terms of data integrity, correctness, and on-time delivery [12]. However, in many IoT applications, massive data loss is common and unavoidable. For example, the Intel Berkeley Research lab dataset [11] is missing roughly 50% of data, the Ocean Sense project is missing 64% of data [13], and the GreenOrbs project is missing 35% of data [14]. This can be ascribed to many reasons. On the one hand, these data have the characteristics of large scale, high dimensions, and complex structures. Data loss or damage may occur during acquisition, transmission, and storage. On the other hand, sensor nodes have limited capabilities, so limitations in terms of energy, storage, communication capabilities, battery depletion, hardware failures, or other environmental and communications issues may cause lead to the capture of incomplete data.

Loss of sensing data hinders various IoT applications and makes it more difficult to process sensor data. Moreover, if the missing data values cannot be filled in accurately, the existing analysis tools cannot be applied; if the missing data are deleted directly, a large amount of raw data would be lost, which would reduce the accuracy and reliability of the analysis results and lead to the wastage of a massive amount of energy. According to a report by IBM, tons of data are produced every day throughout the world. Currently, it is about 2.5 quintillion bytes per day. With IoT, this number will increase considerably. For example, in 2009, there were about 0.9 billion smart objects, and their number is projected to reach 26 billion by 2020. The more widespread the use of this technology, the greater will be the volume of data produced. Recovering missing sensor data effectively to better analyze them for IoT applications is a major challenge. Therefore, it is urgent and important to design effective methods for reconstructing missing values in big sensor data.

Many studies have contributed techniques to predict missing sensor data. Most such techniques are based on temporal methods, spatial methods, or spatio-temporal methods. However, in a few special scenarios, high data loss rates may veil temporal and spatial correlations. Therefore, it is necessary to find new ways to address this problem. We are aware that a sensing node is usually integrated with a multi-function sensor, and these nodes usually gather multiple attributes simultaneously, for example, the data collected by sensor nodes in [15] contains four attributes: temperature, humidity, light, and voltage. Thus, we can assume that there are a few connections between these attributes objectively. For instance, when light illumination is increased, the ambient temperature will increase simultaneously, and air humidity will decrease. An empirical study [16] revealed that temperature, dewpoint temperature, and relative humidity are linearly correlated. That is, the correlations among the attributes can be used as a supplement of the internal correlations to increase estimation accuracy.

In this paper, a new approach to reconstructing missing values in IoT data is proposed. This method implements a K-means clustering algorithm to separate sensors into different groups. The main goal of clustering is to increase the similarity within the same group and the difference between different clusters. Thereafter, we use the potential relationships among multiple data attributes to propose an algorithm based on the matrix rank-minimization method, which is an approach to low rank matrix approximation [17], to reconstruct the multi-attribute sensor data within each cluster. To improve data reconstruction performance, we enhance our algorithm by normalizing the data and using weighted nuclear norm minimization (WNNM) [18]. Our contributions are summarized as follows:

1) We use the K-means clustering algorithm to divide the sensor nodes into different clusters, so as to ensure that sensors within one group have similar patterns of measurement.
2) To the best of our knowledge, this is the first work to apply adaptive weighted nuclear norm minimization algorithm in tensor-based method to sensor data reconstruction problem.
3) We combine the multi-attribute correlation of sensor data and the low-rank minimization technique to propose a tensor-based algorithm, named DR-AWNNM, for reconstruction missing values.

The remainder of this paper is organized as follows. In Section II, we present related work. Section III describes problem formulation. The performance of the proposed method is evaluated in Section IV. Section V provides our concluding remarks and an outline for future work.

II. RELATED WORK

A. EXISTING APPROACHES AND THEIR LIMITATIONS

In the modern IoT paradigm, data integrity is the most important aspect that influences the overall performance of any system. The IoT is used for many critical applications, such as telemetry in hazardous environments, control of industrial processes, e-Health, smart transportation systems, national security, etc. Recently, IoT was used also for the network monitoring to manage the performance of 5G heterogeneous networks under variable conditions The problem of missing sensor data in WSN has been known for a long time [6]. WSN, which is one of the key technologies of IoT, has been researched extensively on the back of rapid development of wireless communication technology, microelectronics technology, and embedded computing technology [9].

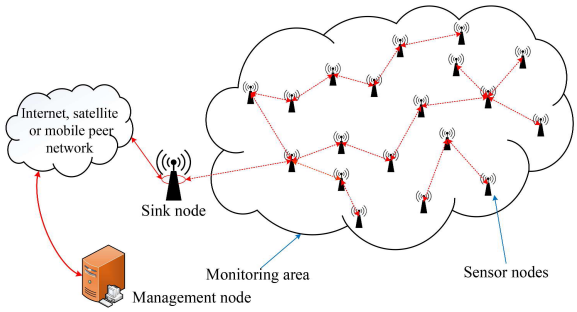

FIGURE 1. Network structure of WSN.

A WSN consists of many sensor nodes distributed in a specific area, each of which has certain computing, storage, and communication capabilities. Figure 1 shows the network structure of a WSN: These devices capture similar data and transmit it to the sink node. Sink node is considered as a store and forward device which retrieves information from the IoT devices, performs data acquisition and finally transmits it to a central entity called cloud under a cloud robotics framework. Many studies in the literature have focused on missing data reconstruction in WSN, and most such techniques are based on temporal methods, spatial methods, or spatio-temporal methods.

Temporal methods leverage the temporal correlations among readings recorded by the same sensor node [19], that is to say, the corresponding sensory data collected by the sensing node within the monitoring range usually change slowly within a short time interval, and the sensing data of the same sensory node are often the same or similar before and after the time interval. Salient methods include observed data mean [20], last seen [21], and linear interpolation. However, for long temporal gaps in observations for a given sensor or when the WSN changes sharply and irregularly, the temporal methods cannot perform well, and their effectiveness decreases rapidly as the number of consecutively missing readings increases.

Spatial methods leverage spatial correlations among the sensor data of spatially similar sensor nodes at the same monitoring time; usually, the closer the sensory nodes to each other spatially, the more relevant are the data. The most classical interpolation method is the K-nearest-neighbor method [22], which utilizes the values of the nearest K neighbors to estimate the missing one. The K-nearest neighbor estimation method [23] is applied to describe the spatial correlation among the sensor data of different sensor nodes by using the linear regression model, and it uses the data information of each neighboring node to estimate the missing data. The window association rule mining [24] and freshness association rule mining [25] estimate missing data based on association rules among spatially correlated neighbors. However, such models are suitable only for limited types of signals and environments, which typically require precise three-dimensional distances between sensors.

Compressive Sensing(CS) is a powerful and generic technique for estimating data, it can utilize a small fraction of data to reconstruct the entire dataset. Reference [26] developed a Compressive Sensing based Data Prediction (CS-DP) model to predict data at the gateways by learning the data pattern received from IoT devices, [27] developed an extended sparse adaptive matching tracking algorithm based on aforementioned Greedy algorithm (CAMP) to reconstruct data in WSN. However, CS cannot be directly applied for environment reconstruction because of its special inherent structures. Meanwhile, the missing data must follow the Gaussian or pure random distribution.

All above methods aiming to estimate missing values are based on a single attribute. However, many physical attributes in nature are strongly correlated, such as humidity and temperature. In [28], a method was proposed to analyze inter-attribute correlations based on the perceived data in real environments. The authors selected the temperature and humidity attribute data of GreenOrbs [14], and after smoothing, they found a significant negative correlation between the changes in temperature and humidity according to the scatter plot. However, actual perception data may be positively correlated or non-linearly related, depending on nature-specific attributes. In [29], the characteristics of real sensor data were studied, and a multi-attribute-assistant compressive-sensing-based algorithm was developed to approximate missing data. The simulation results show this algorithm performed well, except in cases where the data were highly complex.

Recently, the intrinsic low-rank property of high dimensional data has been considered. In [30], the tensor completion theory was used to recover missing data from the sink node of a large-scale WSN; the authors proposed a high-accuracy low-rank tensor completion algorithm (HaLRTC) to solve the tensor completion problem without considering noise. Shao *et al.* [31] propose a tensor-based method called ADMAR to reconstruct multi-attribute sensor data. However, this method cannot perform well when the data missing rate is more than 60% with consecutive missing patterns.

B. MATRIX COMPLEMENT BASED ON NUCLEAR NORM RELAXATION

Wu *et al.* [32] analyzed a temperature dataset collated using sensor data and verified that most of the information in the dataset can be described with few principal components, indicating that sensor data have a low-rank characteristic. Therefore, the missing data reconstruction problem can be transformed into matrix rank minimization problem. We first introduce the basic theory of matrix complement based on nuclear norm relaxation.

Given an incomplete low-rank matrix $\mathbf{E} \in \mathbb{R}^{m\times n}$, the goal is to recover all elements based on a few observed elements of the matrix, which can be described by the following radiation rank minimization problem:

$$\min_{\mathbf{X}} \text{rank}(\mathbf{X}), \quad s.t.\ \mathbf{X}_{ij} = \mathbf{E}_{ij}, (i,j) \in \Omega. \tag{1}$$

where $\Omega \subset \{1, 2, \cdots, n\} \times \{1, 2, \cdots, m\}$ is the set of sampled entries of $\mathbf{E}$, and rank($\mathbf{X}$) is the rank of matrix $\mathbf{X}$. The constraints of the optimization problem can be expressed as $\mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{E})$, as well, where $\mathcal{P}_\Omega(\cdot)$ is defined as follows:

$$[\mathcal{P}_\Omega(\mathbf{X}_{ij})] = \begin{cases} \mathbf{X}_{ij}, & \text{if}(i,j) \in \Omega, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The rank function of the matrix is discontinuous and non-convex, and the optimization problem has been evidenced to be NP-hard [17], therefore, it cannot be solved easily in practice. At present, the method mainly converts the convex relaxation process of the problem into the problem of solving Nuclear Norm Minimization (NNM). Thus problem (1) is transformed into the following:

$$\min_{\mathbf{X}} \|\mathbf{X}\|_*, \quad s.t.\ \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{E}). \tag{3}$$

where $\|\mathbf{X}\|_* = \Sigma_i \sigma_i(\mathbf{X})$ denotes the nuclear norm of $\mathbf{X}$, and $\sigma_i(\mathbf{X})$ denotes the ith largest singular value. In practice, the noises in sensory data may lead to over-fitting, so the equality constraints cannot be satisfied strictly. Thus, problem (3) is often placed in the objective function:

$$\min_{\mathbf{X}} \|\mathbf{X}\|_* + \frac{1}{2\lambda} \left\| \mathcal{P}_\Omega(\mathbf{X}) - \mathcal{P}_\Omega(\mathbf{X}) \right\|_F^2, \tag{4}$$

In practice, noises in sensory data may lead to over-fitting if strict satisfaction is required. Therefore, we use the parameter λ ($0 < \lambda < 1$) controls to the fit to constraint $\mathcal{P}_\Omega(\mathcal{X}) = \mathcal{P}_\Omega(\mathcal{T})$. $\|\mathbf{X}\|_F^2 := \sqrt{\Sigma_{i,j} |x_{ij}|^2}$ is the Frobenius norm of $\mathbf{X}$. Cai *et al.* [33] proposed a singular value threshold algorithm for solving NNM problems:

$$\mathbf{D}_\tau(\mathbf{X}) = \arg\min_{\mathbf{X}} \frac{\lambda}{2} \left\| \mathbf{X} - \mathbf{E} \right\|_F^2 + \tau \|\mathbf{X}\|_*, \tag{5}$$

where $\mathbf{D}_\tau(\mathbf{X})$ is the "shrinkage," operator of $\mathbf{X}$, and $\tau(\tau > 0)$ is a constant. If rank($\mathbf{X}$) $= r$, its singular value is decomposed into $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}_\tau\mathbf{V}^T$, where $\mathbf{\Sigma} = diag(\{\sigma_i\}, 1 \le i \le r)$ is a vector, consisting of $\sigma_i(\mathbf{X})$ with descending order, and $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices. Thus, the matrix shrinkage operator $\mathbf{D}_\tau(\mathbf{X})$ is defined as follows:

$$\mathbf{D}_\tau(\mathbf{X}) = \mathbf{U}\text{diag}(\{\max(0, \sigma_i - \tau)\})\mathbf{V}^T. \tag{6}$$

III. PROBLEM FORMULATION

A. GROUPING SENSOR NODES WITH K-MEANS CLUSTERING ALGORITHM

Normally, in a monitored region, many sensor nodes are deployed. In [16], it was proved that data sensed from these nodes have spatial correlations. For example, Fig. 2 shows the temperature observed by three sensor nodes from the Intel Berkeley Research Laboratory dataset over two days. On the one hand, the data sensed by nearby nodes, namely, nodes 32 and 33, have similar curves. Thus, when some of the sensed data of a sensor node are missing, we can estimate them by using the data of the neighboring nodes. On the other hand, the data curve of node 2 is completely different from those of nodes 32 and 33 because node 2 is

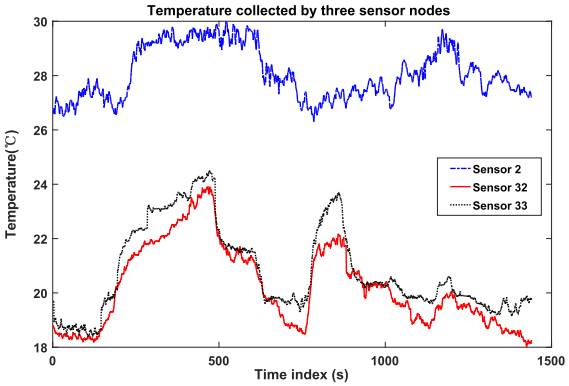

FIGURE 2. Temperature collected by three sensor nodes.

far from nodes 32 and 33. This example shows that not every node in a given is useful for recovering the missing values of a certain node. Therefore, to better use the degree of similarity among measured values of neighboring sensors, we first divide the sensors into different groups to minimize measurement changes within each group.

In this work, we use the K-means clustering algorithm to group sensors. K-means clustering algorithm is a classic unsupervised clustering algorithm that provides good grouping of rectangular and circular regions [34]. This algorithm divides a set of points into K clusters so that points in each cluster tend to be close to each other. We list the main steps of K-means clustering algorithm in the following:

Step 1: Use $\mathcal{X}^* = \{x_1, x_2, \ldots, x_N\}$ to represent the set of coordinates of N sensor nodes in the monitoring area, and randomly select K objects as the initial clustering center, denoted as $\mathcal{C} = \{c_1, c_2, \ldots, c_K\}$, where each object represents a clustering center. Next, place the cluster centers of K clusters uniformly within the target field of sensors.

Step 2: Associate each sensor with the nearest cluster center by using the criterion of distance:

$$x_i^c = \arg \min_{k=1,\ldots,K} \|x_i - c_k\|^2, \tag{7}$$

where c_i stands for the clustering center of sensor node i. We gather the sensor nodes associated with clustering center k into set $\mathcal{C}_k$. We call set $\mathcal{C}_k$ as a cluster.

Step 3: For each cluster $\mathcal{C}_k$, calculate the average coordinates as follows:

$$c_k' = \frac{\sum_{i \in \mathcal{C}_k} x_i}{|\mathcal{C}_k|}, \tag{8}$$

where $|\mathcal{C}_k|$ denotes the cardinality of set $\mathcal{C}_k$.

Step 4: Update the coordinates of cluster centers:

$$\mathcal{C}' = \left\{c_1', c_2', \ldots, c_K'\right\}. \tag{9}$$

Step 5: Determine the differences between the new cluster center and the previous cluster center:

$$\triangle c_k = \left\| c_k' - c_k \right\|^2. \tag{10}$$

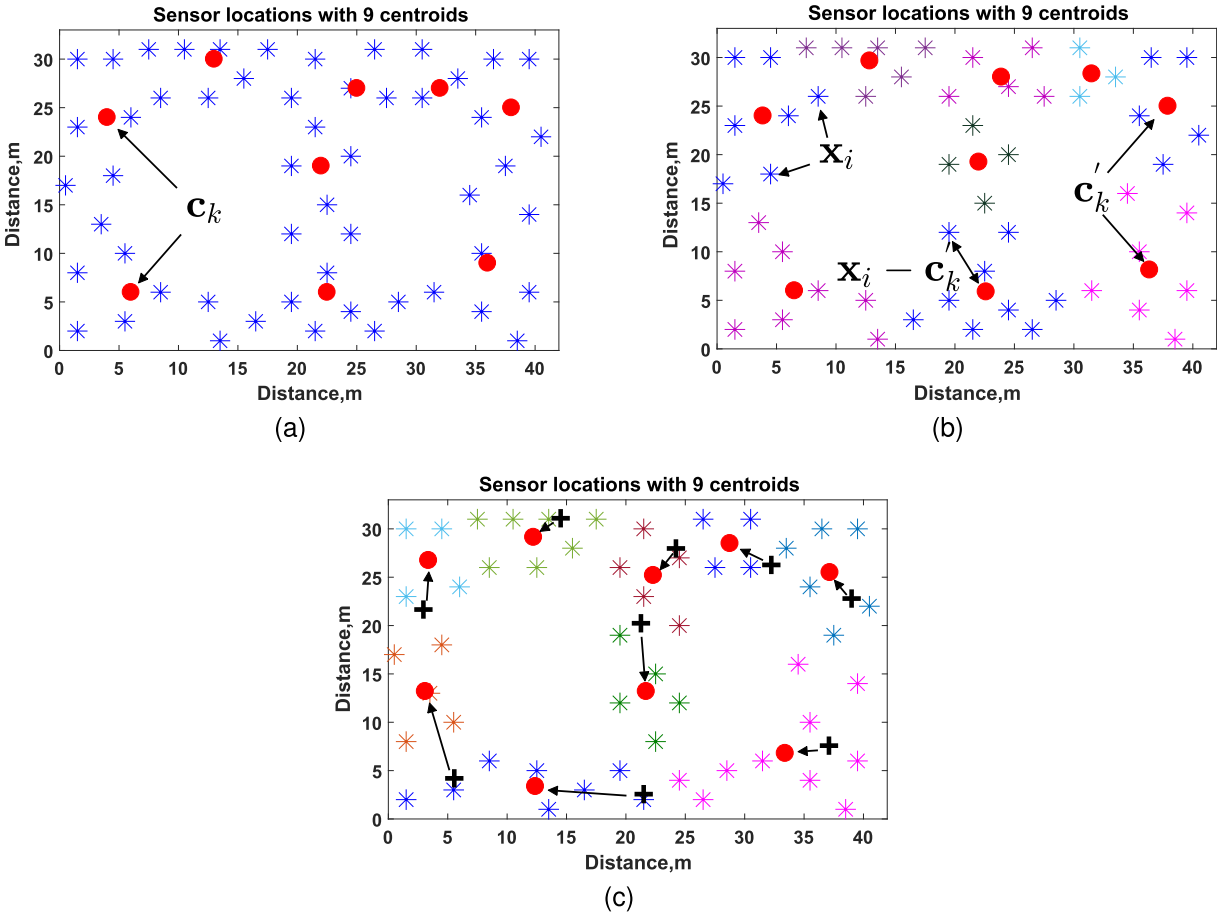

FIGURE 3. Iterations of K-means clustering algorithm ($K = 9$, (a)-Step 1, (b)-step 2 and 3, (c)-step 4 and 5).

Step 6: Iterate steps 2, 3, 4 and 5 until set $\mathcal{C}_k$ no longer changes or the difference between adjacent iterations is smaller than the given threshold:

$$\sum_k \triangle c_k < \eta. \tag{11}$$

where η is a small positive number, here we set $\eta = 1e^{-4}$ [35].

Figure 3 shows the procedure the five-step procedure of one iteration. The red solid points represent the cluster centers, and the stars represent the sensor nodes; the different colors of sensor points indicate different clusters.

B. DATA RECONSTRUCTION WITH ADAPTIVE WEIGHTED NUCLEAR NORM (DR-AWNNM)

Suppose that N nodes are deployed in a monitoring area, each of which is equipped with K^* sensors to monitor different at-tributes simultaneously. The monitoring period consists of Q time slots. The sensor data gathered in one node can be organized in the following format [29], where sensor ID stands for sensor identity number, time stamp represents sampling time, and the attributes include temperature and humidity.

TABLE 1. The data format.

Sensor ID	Time Stamp	Attribute 1	Attribute 2	...

Let $\mathcal{T}$ be a tensor of the sensor data with M attributes collected by N nodes within Q time slots, that is, $\mathcal{T} \in \mathbb{R}^{N\times M\times Q}$; the tensor is a generalization of the matrix concept. We use $t_{n,m,q}$ to represent the entry of $\mathcal{T}$. Owing to data loss in IoT, $\mathcal{T}$ is usually an incomplete tensor. The intact information of $\mathcal{T}$ is a set of entries $t_{n,m,q}$, for $(n, m, q) \in \Omega$, where Ω is the set of sampled entries of $\mathcal{T}$. We use $\mathcal{P}_\Omega(\cdot)$ to indicate the sampling operator, which is defined as follows:

$$[\mathcal{P}_\Omega(\mathcal{T})]_{n,m,q} = \begin{cases} t_{n,m,q}, & \text{if } (n, m, q) \in \Omega, \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

The multi-attribute sensor data reconstruction problem is defined as follows:

Given a subset of $\mathcal{T}$, denoted as $\mathcal{P}_\Omega(\mathcal{T})$, the ultimate goal is to find an optimal solution $\hat{\mathcal{T}}$ that minimizes the error between $\mathcal{T}$ and $\hat{\mathcal{T}}$:

$$\begin{aligned} &\min \ \|\mathcal{T} - \hat{\mathcal{T}}\|_F \\ &\text{s.t. } \mathcal{P}_\Omega(\hat{\mathcal{T}}) = \mathcal{P}_\Omega(\mathcal{T}), \end{aligned} \tag{13}$$

where $\|\mathcal{X}\|_F := \sqrt{\text{ïijĹ}(\Sigma_{i_1,i_2,\ldots,i_n}\left|x_{i_1,i_2,\ldots,i_n}\right|^2)\text{ïijL'}}$is defined as the Frobenius norm of tensor $\mathcal{X}$.

Many studies [19], [32], have mentioned that the sensor data tensor is low-rank, so the missing values in $\mathcal{T}$ can be recovered using the matrix complement based on nuclear norm relaxation:

$$\begin{aligned} &\min \ \|\mathcal{X}\|_* \\ &\text{s.t. } \mathcal{P}_\Omega(\mathcal{X}) = \mathcal{P}_\Omega(\mathcal{T}), \end{aligned} \tag{14}$$

where $\|\mathcal{X}\|_*$ is the nuclear norm of $\mathcal{X}$, which is defined as $\|\mathcal{X}\|_* := \sum_{i=1}^{N} \alpha_i \left\|\mathcal{X}_{(i)}\right\|_*$, where α_i is the weight corresponding to the unfolding of the i-th mode. Here, let $\alpha_1 = \alpha_2 = \alpha_3$ $(N = 3)$, which means each $\mathcal{X}_{(i)}$ is assigned equal importance. However, traditional NNM has certain limitations, for example, all singular values are treated equally and shrunk with the same threshold [18]. In doing so, prior knowledge on singular values of a practical data matrix is ignored. More specifically, larger singular values of an input data matrix quantify the information of its underlying principal directions. In other words, inner the same unfolding $\mathcal{X}_{(i)}$, considering the information included in different singular value is different [33]. Obviously, the traditional NNM model, as well as its corresponding soft-thresholding solvers, are not adequately flexible to handle such issues.

To improve the flexibility of NNM, weak shrinkage strength should be enforced on the large singular value, while a large weight should be assigned to the small singular value. Based on this idea, we use the weighted nuclear norm minimization (WNNM) algorithm, and define the weighted nuclear norm of $\mathcal{X}$ as follows:

$$\left\|\mathcal{X}_{(i)}\right\|_{\mathbf{w},*} = \sum_{j=1}^{3} w_i \sigma_i(\mathcal{X}_{(i)}), \tag{15}$$

where $\mathbf{w} = [w_1, w_2, \ldots, w_n]^T$ and $w_i \geq 0$, which is a non-negative weight assigned to $\sigma_i(\mathcal{X})$. The weight vector enhances the representation capability of the original nuclear norm.

Therefore, the proposed model is revised as follows:

$$\begin{aligned} &\min \sum_{i=1}^{3} \left\|\mathcal{X}_{(i)}\right\|_{w,*} \\ &\text{s.t. } \mathcal{P}_\Omega(\mathcal{X}) = \mathcal{P}_\Omega(\mathcal{T}), \end{aligned} \tag{16}$$

For convenience, we define a sampling tensor $\mathcal{H}$, where

$$h_{i,j,\ldots,n} = \begin{cases} 1, & \text{if}(i,j,\ldots,n) \in \Omega \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

Clearly, $\mathcal{H}$ is a binary tensor that indicates whether any data in $\mathcal{T}$ are missing.

Now, we can define the multi-attribute sensor-data reconstruction problem. Let $\mathcal{H}$ denote a binary sampling tensor, and $\mathcal{T}$ denote the incomplete tensor of multi-attribute sensor data. Then, the missing values in Eq.(16) can be estimated effectively by solving the following convex optimization problem,

$$\underset{\mathcal{X}}{minimize} \sum_{i=1}^{3} \left\|\mathcal{X}_{i,(i)}\right\|_{w,*} + \frac{1}{2\lambda}\left\|\mathcal{H} \cdot \sum_{i=1}^{3} \mathcal{X}_i - \mathcal{H} \cdot \mathcal{T}\right\|_F^2. \tag{18}$$

where $(\cdot)$ denotes element-wise production of the tensor, and λ is an adjustable parameter introduced to prevent over-fitting.

C. ALTERNATE DIRECTION MULTIPLIER METHOD FOR DR-AWNNM

The alternate direction multiplier method (ADMM) algorithm [36], a convex optimization algorithm, is widely used to solve convex optimization problems by breaking them into smaller pieces, each of which can then be handled easily. It is an extension of the augmented Lagrange multiplier method algorithm . Therefore, in the present study, we attempted to use the ADMM algorithm to solve the formulated problem.

To apply the ADMM method to problem (18), we need to transform it into the ADMM form. Thus, we first perform variable splitting. We introduce $\mathcal{N}$ new tensor-valued variables, $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_N$. Let $\mathcal{X}_i = \mathcal{M}_i$ $(i \in 1, 2, \ldots, N)$; by using these new variables $\mathcal{M}_i$, Eq. (18) can be rewritten as follows:

$$\begin{aligned} &\underset{\mathcal{X}_i, \mathcal{M}_i}{minimize} \sum_{i=1}^{3} \left\|\mathcal{X}_{i,(i)}\right\|_{\mathbf{w},*} + \frac{1}{2\lambda}\left\|\mathcal{H} \cdot \sum_{i=1}^{3} \mathcal{M}_i - \mathcal{H} \cdot \mathcal{T}\right\|_F^2 \\ &\text{s.t. } \mathcal{X}_i = \mathcal{M}_i, \quad i = 1, 2, \ldots, N \end{aligned} \tag{19}$$

The augmented Lagrangian of (19) becomes

$$\begin{aligned} \mathcal{L}(\mathcal{X}_i, \mathcal{M}_i, \mathcal{Y}_i) = &\langle \mathcal{Y}_i, \mathcal{X}_i - \mathcal{M}_i \rangle + \frac{\rho}{2}\|\mathcal{X}_i - \mathcal{M}_i\|_F^2 \\ &+ \frac{1}{2\lambda}\left\|\mathcal{H} \cdot \sum_{i=1}^{3} \mathcal{M}_i - \mathcal{H} \cdot \mathcal{T}\right\|_F^2 \\ &+ \sum_{i=1}^{3} \left\|\mathcal{X}_{i,(i)}\right\|_{w,*}, \end{aligned} \tag{20}$$

For convenience, by combining the linear and quadratic terms in the augmented Lagrangian and scaling the dual variable, Eq.(20) can be simplified as follows

$$\begin{aligned} \mathcal{L}(\mathcal{X}_i, \mathcal{M}_i, \mathcal{Y}_i) = &\sum_{i=1}^{3} \left\|\mathcal{X}_{i,(i)}\right\|_{w,*} + \frac{\rho}{2}\|\mathcal{X}_i - \mathcal{M}_i + \mathcal{U}_i\|_F^2 \\ &+ \frac{1}{2\lambda}\left\|\mathcal{H} \cdot \sum_{i=1}^{3} \mathcal{M}_i - \mathcal{H} \cdot \mathcal{T}\right\|_F^2 + const, \end{aligned} \tag{21}$$

where *const* represents a constant term, $\mathcal{U}$ is the scaled dual variable [36], and $\mathcal{U}_i = \rho^{-1}\mathcal{Y}_i$. Now, we can obtain the

iterations of ADMM as follows.

$$\begin{aligned}\mathcal{X}_i^{k+1} &= \underset{\mathcal{X}_i}{\operatorname{argmin}}\,(\left\|\mathcal{X}_{i,(i)}\right\|_{w,*} + \frac{\rho}{2}\left\|\mathcal{X}_i + \mathcal{U}_i^k - \mathcal{M}_i^k\right\|_F^2),\\ \mathcal{M}_i^{k+1} &= \underset{\mathcal{M}_i}{\operatorname{argmin}}\,(\frac{1}{2\lambda}\left\|\mathcal{H}\cdot\sum_{i=1}^{N}\mathcal{M}_i - \mathcal{H}\cdot\mathcal{T}\right\|_F^2\\ &\quad + \frac{\rho}{2}\left\|\mathcal{M}_i - \mathcal{X}_i^{k+1} - \mathcal{U}_i^k\right\|_F^2),\\ \mathcal{U}_i^{k+1} &= \mathcal{U}_i^k + \mathcal{X}_i^{k+1} - \mathcal{M}_i^{k+1}. \end{aligned} \tag{22}$$

1) UPDATE $\mathcal{X}$

Before the update step for $\mathcal{X}$, we need to introduce the following definition and theorem.

Definition 1: Given a matrix $\mathbf{Y}$, the WNNM problem [18] aims to find a matrix $\mathbf{X}$, which is as close to $\mathbf{Y}$ as possible under certain data fidelity functions, and the WNNM problem is described as follows:

$$\min_{\mathbf{X}} \|\mathbf{Y}-\mathbf{X}\|_F^2 + \|\mathbf{X}\|_{\boldsymbol{w},*} \tag{23}$$

here, we let

$$w_i = \frac{C}{\sigma_i(\mathcal{X}_{(i)}) + \xi}, \tag{24}$$

where ξ is a small constant, and C is a compromising constant, both constants were set empirically in our experiment.

Theorem 1: $\forall \mathbf{Y} \in \Re^{m\times n}$, denote by $\mathbf{Y} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ the SVD of it, where $\Sigma = \begin{pmatrix} diag(\sigma_1(\mathbf{Y}), \sigma_2(\mathbf{Y}), \ldots, \sigma_n(\mathbf{Y})) \\ 0, \end{pmatrix}$ and $\sigma_i(\mathbf{Y})$ denotes the i-th singular value of $\mathbf{Y}$. If ε and C satisfy the inequality $\varepsilon < \left(\sqrt{(C)}, \frac{C}{\sigma_i(Y)}\right)$, and the weights satisfy $w_1 \geq \cdots \geq w_n \geq 0$ simultaneously, the WNNM problem in Eq.(23) has the closed-form solution: $\mathbf{X}^* = \mathbf{U}\tilde{\Sigma}\mathbf{V}^T$, where $\tilde{\Sigma} = \begin{pmatrix} diag(\sigma_1(\mathbf{X}^*), \sigma_2(\mathbf{X}^*), \ldots, \sigma_n(\mathbf{X}^*)) \\ 0, \end{pmatrix}$, and

$$\sigma_i(\mathbf{X}^*) = \begin{cases} 0 & \text{if } c_2 < 0 \\ \dfrac{c_1 + \sqrt{c_2}}{2} & \text{if } c_2 \geq 0 \end{cases} \tag{25}$$

where
$c_1 = \sigma_1(\mathbf{Y}) - \varepsilon$, $c_2 = (\sigma_1(\mathbf{Y}) + \varepsilon)^2 - 4C$.

Therefore, the variable $\mathcal{X}_i$ can be solved independently by using the matrix shrinkage operator introduced above. Hence, the update of $\mathcal{X}_i$ can be given as follows:

$$\mathcal{X}_i^{k+1} = fold_i(\mathcal{D}_{1/\rho}(\mathcal{M}_i^k - \mathcal{U}_i^k)_{(i)}). \tag{26}$$

where the "unfold" operation along the i-th mode on a tensor $\mathcal{X}$ is defined as $\text{unfold}_i(\mathcal{X}) := X_{(i)} \in \mathbb{R}^{K_i\times(K_1\ldots K_{i-1}K_{i+1}\ldots K_n)}$. The opposite operation "fold" is defined as $\text{fold}_i(X_{(i)}) := \mathcal{X}$. It is clear that $\|\mathcal{X}\|_F = \left\|X_{(i)}\right\|_2$ is for any $1 \leq i \leq n$.

2) UPDATE $\mathcal{M}$

Relative to the update of $\mathcal{X}$, the iteration of $\mathcal{M}$ is more complicated. In the following part, we use the method introduced in [31] to solve this problem.

According to the method in [31], the update of $\mathcal{M}$ is equal to the iteration of $\bar{\mathcal{M}}$:

$$\begin{aligned}\bar{\mathcal{M}}^{k+1} = \underset{\bar{\mathcal{M}}_i}{\operatorname{argmin}}\,(&\frac{1}{2\lambda}\left\|\mathcal{H}\cdot N\bar{\mathcal{M}} - \mathcal{H}\cdot\mathcal{T}\right\|_F^2\\ &+ \frac{\rho}{2}\left\|\bar{\mathcal{Z}} - \bar{\mathcal{X}}^{k+1} - \bar{\mathcal{U}}^k\right\|_F^2),\end{aligned} \tag{27}$$

where

$$\begin{aligned}\bar{\mathcal{M}} &= \frac{1}{N}\sum_{i=1}^{N}\mathcal{M}_i,\\ \bar{\mathcal{X}}^{k+1} &= \frac{1}{N}\sum_{i=1}^{N}\mathcal{X}_i^{k+1},\\ \bar{\mathcal{U}}^k &= \frac{1}{N}\sum_{i=1}^{N}\mathcal{U}_i^k. \end{aligned} \tag{28}$$

and then we get the $\bar{\mathcal{M}}$-update solution :

$$\bar{\mathcal{M}}^{k+1} = \mathcal{M}_+ \cdot / \mathcal{M}_-, \tag{29}$$

where

$$\begin{aligned}\mathcal{M}_- &= (1/\lambda)(\mathcal{H} + \rho\mathcal{I}),\\ \mathcal{M}_+ &= \rho(\bar{\mathcal{X}}^{k+1} + \bar{\mathcal{U}}^k + (1/(N\lambda))\mathcal{H}\cdot\mathcal{T}).\end{aligned} \tag{30}$$

3) THE DR-AWNNM ALGORITHM

After discussing the appearing subproblem, we present the complete DR-AWNNM algorithm for multi-attribute sensor-data reconstruction, as in Alg.1.

DR-AWNNM algorithm can be mainly divided into four steps: *Step 1:* The updating of parameter $\mathcal{X}_i^{k+1}$; *Step 2:* Calculating $\bar{\mathcal{X}}^{k+1}, \bar{\mathcal{U}}^k$ and $\bar{\mathcal{M}}$ to update $\bar{\mathcal{M}}^{k+1}$; *Step 3:* Using $\mathcal{X}_i^{k+1}$ and $\bar{\mathcal{M}}^{k+1}$ to update $\mathcal{U}_i^{k+1}$; *Step 4:* Iterate steps 1, 2 and 3 until the difference between adjacent iterations is smaller than the given threshold ρ.

The algorithm uses the sampling binary index tensor $\mathcal{H}$, incomplete sensor data tensor $\mathcal{T}$, and the parameters λ,ρ,c_λ,λ^* as inputs. It minimizes Eq.(18) iteratively by decreasing A toward convergence. λ^* is set as the lower bound of λ.

Figure 4 shows the flowchart of the proposed missing data reconstruction algorithm. Our algorithm is divided into two main steps: first, to fully use the spatial correlations among the data, we use the k-means clustering algorithm to group sensor nodes for obtaining the best classification. Second, the complete dataset $\mathcal{T}$ is processed using two patterns to represent missing data, namely, random missing pattern and consecutive missing pattern. Finally, the DR-AWNNM algorithm is used in each cluster according to the clustering result, and then the ADMM algorithm is used to iterate parameters until the global optimal solution is output by the algorithm.

IV. EXPERIMENTAL RESULTS

In this section, we evaluate the performance of the proposed algorithm and compare it with exiting algorithms for missing data estimation in sensor data reconstruction.

TABLE 2. Data samples obtained in the Intel Berkeley Research Laboratory.

Date	Time	Sensor ID	Temperature($^\circ C$)	Humidity(%)	Light(Lux)	Voltage(V)
3/11/2004	6:20:33 AM	20	24.2272	38.1454	115.43	2.65332
3/11/2004	6:21:25 AM	21	21.0636	50.4818	116.33	2.63753
3/11/2004	6:22:27 AM	22	21.0755	48.6375	115.42	2.64331
3/11/2004	6:25:23 AM	23	20.0778	47.3857	114.35	2.64332
3/11/2004	6:27:30 AM	24	27.4872	35.1636	121.44	2.67532
...	...	...	...	...	...	...

Algorithm 1 DR-AWNNM Algorithm for Multi-Attribute Sensor-Data Reconstruction

Input: $\mathcal{T}$,$\mathcal{H}$, ε, C, λ, ρ, c_λ, λ^*;
Initialize $k=0$, $\bar{\mathcal{M}}^0=\mathcal{U}^0=\mathcal{X}_i^0=0, i=1,2,\ldots,N$;
for $k=0,1,\ldots$ **do**
  for $i=1:N$ **do**
    // Lines 6 solve $\mathcal{X}_i^{k+1}=\underset{\mathcal{X}_i}{\operatorname{argmin}}\,\mathcal{L}(\mathcal{X}_i,\mathcal{M}_i^k,\mathcal{U}_i^k)$
    $\mathcal{X}_i^{k+1}=fold_i(\mathcal{D}_{1/\rho}(\mathcal{M}_i^k-\mathcal{U}_i^k)_{(i)})$.
  end for
  //Line 9-12 solve$\mathcal{M}_i^{k+1}=\underset{\mathcal{M}_i}{\operatorname{argmin}}\,\mathcal{L}(\mathcal{X}_i^{k+1},\mathcal{M}_i,\mathcal{U}_i^k)$
  Calculate $\bar{\mathcal{X}}^{k+1}=\frac{1}{N}\sum_{i=1}^{N}\mathcal{X}_i^{k+1}$;
  $\bar{\mathcal{U}}^k=\frac{1}{N}\sum_{i=1}^{N}\mathcal{U}_i^k$;
  $\bar{\mathcal{M}}=\frac{1}{N}\sum_{i=1}^{N}\mathcal{M}_i$;
  $\mathcal{M}_-=(1/\lambda)(\mathcal{H}+\rho\mathcal{I})$;
  $\mathcal{M}_+=\rho(\bar{\mathcal{X}}^{k+1}+\bar{\mathcal{U}}^k+(1/(N\lambda))\mathcal{H}\cdot\mathcal{T})$;
  $\bar{\mathcal{M}}^{k+1}=\mathcal{M}_+\cdot/\mathcal{M}_-$;
  for $i=1:N$ **do**
    $\mathcal{U}_i^{k+1}=\mathcal{U}_i^k+\mathcal{X}_i^{k+1}-\mathcal{M}^{k+1}$.
  end for
  $\lambda^{k+1}=\max(c_\lambda\lambda^k,\lambda^*)$;
end for
Output: $\sum_{i=1}^{N}\mathcal{X}_i^{k+1}$.

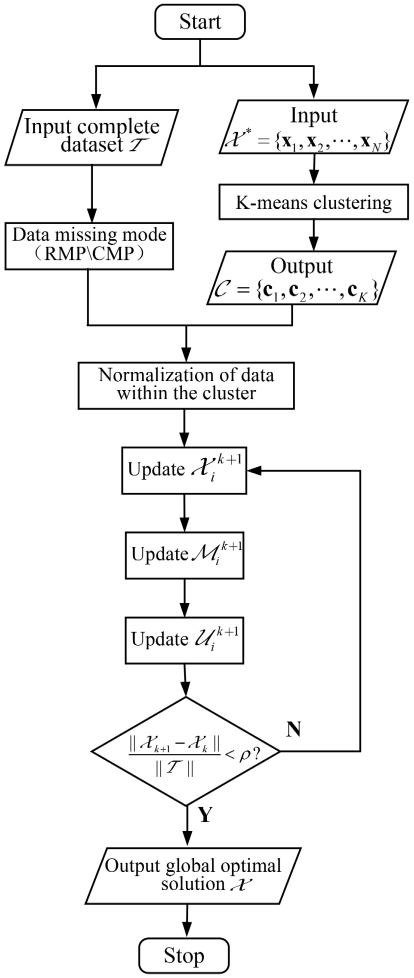

FIGURE 4. A flowchart of the proposed missing data reconstruction algorithm.

A. EXPLANATION OF THE DATASET USED FOR SIMULATION

1) INTEL BERKELEY DATASET

For our experimental simulation, we used data collected from the Intel Berkeley Research Laboratory between February 28 and April 5, 2004 [37]. The laboratory has different rooms, and in each room, Mica2Dot sensors collect times tamped topology information, along with humidity, temperature, light, and voltage values once every 31 s. Data were collected using the TinyDB in-network query processing system implemented on the TinyOS platform. The data collected from all sensors were merged into one big dataset containing 2.3 million readings (∼150 MB). Samples from this dataset are given in Table 2.

2) MISSING DATA

To verify the validity of the algorithm accurately and objectively, we adopted two data processing patterns, namely, random missing pattern and consecutive missing pattern. We first obtained complete raw sensor data and then produced artificial missing data based on the two patterns.

- *Random Missing Pattern (RMP)*: This pattern repeatedly chooses a random time and random sensor to be missing and hence the data is missing.
- *Consecutive Missing Pattern (CMP)*: This pattern reflects that a few nodes miss all data after a certain

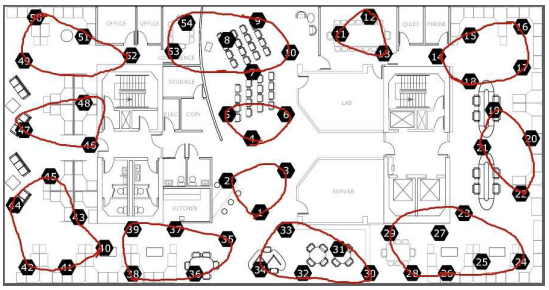

FIGURE 5. Sensor locations and clustering in the Intel Berkeley Research Laboratory.

sampling time point owing to damage or running out of energy. Therefore, we randomly selected 10% of nodes as objective nodes that suffer from consecutive data missing and then let each objective node lose the last x% of all its data.

3) PARAMETER SETTINGS

We employed error formulation to measure the differences between the predicated values and the actual values. Performance was measured in terms of RSE, a metric for measuring reconstruction error, and we defined it as follows:

$$RES = \frac{\|\mathcal{X} - \mathcal{T}\|_F^2}{\|\mathcal{T}\|_F^2}, \tag{31}$$

The sampling ratio ε refers to the observation ratio of sensor data, which is defined in [31]

$$\varepsilon = \frac{\sum_{(i,j)\in\Omega} 1}{\sum_{(i,j)\in(\Omega \bigcup \tilde{\Omega})} 1}. \tag{32}$$

In our experiment, we set some parameters empirically [31], specifically, $\lambda = 1$, $c = \frac{1}{4}$, $\lambda^* = 1e^{-6}$. ρ is a positive number, and its value affects the speed of convergence of the ADMM algorithm. Here, we set $\rho = 1.05$ [30]. The parameter C is associated with the allocation of weights to singular values, and it is a very important parameter. Therefore, we discussed its value separately in the following separate.

All simulations were run in the MATLAB environment on a desktop computer equipped with a 3.60-GHz Intel i7-4790 CPU and 8 GB RAM.

B. SIMULATION PARAMETERS AND RESULTS

Figure 5 shows the locations of sensor nodes and their clusters in the Intel Berkeley Research Laboratory. Here, the number of clusters is 12. The corresponding sensor nodes of each cluster are shown within the solid lines. Our DR-AWNNM algorithm for missing data reconstruction was applied within each group.

The main reason for clustering the sensors was to find similarities between their measurements. Figure 6 shows temperature measurements from the sensors in two clusters. Here, sensors 22 and 20 are in the first cluster, and

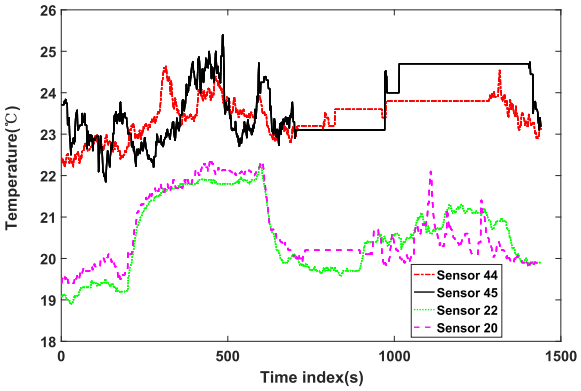

FIGURE 6. Comparison of the measurements within and among clusters.

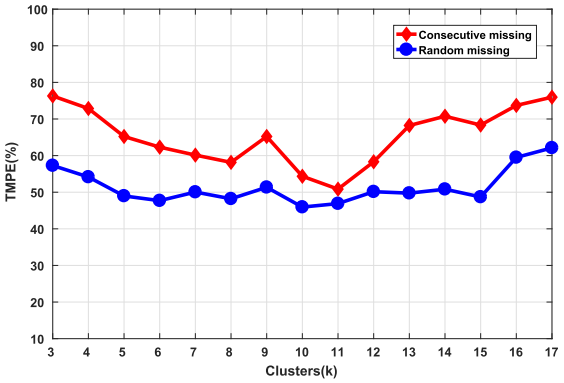

FIGURE 7. Total maximum prediction error for different numbers of clusters.

sensors 44 and 45 are in the second cluster. As can be seen in Fig. 6, measurements recorded by the sensors within one cluster follow very similar patterns, but the difference between clusters is relatively large.

Therefore, when there are missing values in a cluster, we can estimate them from neighboring nodes within the same cluster. We first used the spatial relationship between sensor nodes to divide them into different clusters and then applied our algorithm within each cluster. To enhance the reliability of the proposed algorithm, we changed K from 3 to 17 and then calculated the difference between the predicted and original values within each cluster to obtain the maximum error between them.

Theoretically, as the cluster size decreases, the spatial relationships within the same cluster become stronger; thus, the final calculation result is more accurate. However, as the number of clusters increases, there will be fewer nodes in the same cluster.

Figure 7 shows the simulation results of the total maximum prediction error (TMPE) for different numbers of clusters. The red and the blue curves, respectively, represent the total maximum error of all classes corresponding to each K in the case of consecutive missing and random missing data. According to the obtained results, the TMPE value decreases upon increasing the number of clusters, which confirms the

TABLE 3. Partial experimental results with RMP.

C \ ε	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$1.3m$	0.1539	0.2266	0.2887	0.3505	0.4192	0.4926	0.5754	0.6802	0.8650
$2m^2$	0.1809	0.0450	0.0250	0.0188	0.0166	0.0153	0.0135	0.0121	0.0094
$4m^2$	0.2430	0.1045	0.0817	0.0614	0.0584	0.0506	0.0395	0.0298	0.0197
$5m^2$	0.2501	0.1071	0.0873	0.0639	0.0559	0.0525	0.0393	0.0309	0.0217
$8m^2$	0.2408	0.1002	0.0786	0.0639	0.0536	0.0437	0.0394	0.0319	0.0233

TABLE 4. Partial Experimental Results With CMP.

C \ ϵ	10%	20%	30%	40%	50%	60%	70%	80%	90%
$0.5m$	0.0103	0.0171	0.0241	0.0307	0.0393	0.0496	0.0691	0.1813	0.3526
$1.3m$	0.0114	0.0187	0.0259	0.0331	0.0406	0.0526	0.0665	0.1095	0.2034
$3m$	0.0126	0.0204	0.0279	0.0353	0.0452	0.0557	0.0673	0.1224	0.2333
$5m$	0.0133	0.0222	0.0298	0.0368	0.0466	0.0570	0.0724	0.1336	0.2729
$8m$	0.0143	0.0230	0.0310	0.0403	0.0507	0.0592	0.0743	0.1494	0.3039

theoretical expectations. However, when the number of clusters exceeds a certain threshold, the curve shows an upward trend. Thus, we selected the minimum TMPE corresponding to the best K as the basis for subsequent research.

Next, we evaluated the performance of the proposed DR-AWNNM algorithm in reconstructing multi-attribute sensor data. In this step, we first used multi-attribute sensor data to constitute a third-order tensor, where the three modes represent sensor time stamp, sensor node ID, and attributes (such as temperature and humidity). Then, we obtained the tensor of multi-attribute sensor data.

In our experiment, we compared the output of the recovered sensor data with that of existing algorithms, namely, HaLRTC [30], ADMAR [31], CAMP [27] and EM-based Tucker decomposition algorithm [38]. The Tucker rank is approximately rank-[2,2,2]. By contrast, we used the correct rank (rank-[2,2,2]) and a higher rank (here we obtained rank-[5,5,2]) to execute Tucker decomposition. We applied Tucker decomposition to the Intel Berkeley dataset under two patterns: random missing and consecutive missing.

We know that the parameter C is very important, and it can even directly affect the effectiveness of our algorithm. Therefore, we discuss it separately. In [18], the parameter C was set empirically as the square root of the tensor size; we changed its value on this basis, and its final value was as follows:

- RMP : $C = 2m^2$,
- CMP : $C = 1.3m$.

where m is the row of $\mathcal{T}$. We obtained this result by conducting multiple experiments, and some of these experimental results are presented in Tables 3 and 4; ε represents the sampling ratio, and ϵ represents the data missing rate.

Figure 8 shows the results of multi-attribute sensor data reconstruction by using data from the Intel Berkeley Research

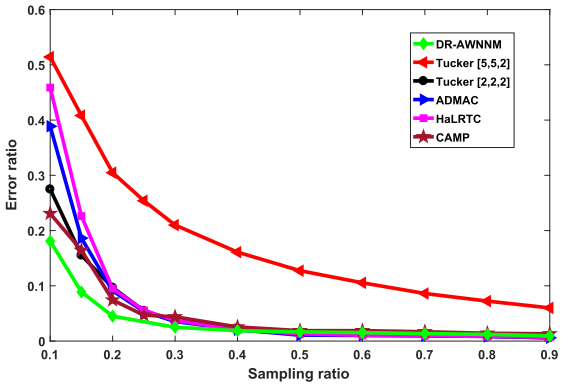

FIGURE 8. Tensor-based multi-attribute sensor-data reconstruction, with random missing pattern.

Laboratory with the random missing pattern. In general, the performance of our algorithm is better than that of the existing algorithms considered for comparison. The proposed DR-AWNNM algorithm performs as well as the other algorithms when sampling ratio is higher than 40%. Moreover, when the sampling ratio is lower than 40%, our error is close to 0.2, which is considerably lower than that of the other three algorithms. We can also know that the CAMP algorithm also has a good performance, because when the data is missing, the algorithm can develop and gradually develop the sparsity adaptive matching tracking framework based on these observation nodes to reconstruct the missing data more accurately.

Figure 9 shows the results of the consecutive missing pattern. From this figure, it can be seen that the overall error ratio under this pattern is lower than that under the random missing pattern. The Tucker decomposition of correct rank-[2,2,2] leads to the best performance, and the DR-AWNNM algorithm performs as well as the best one

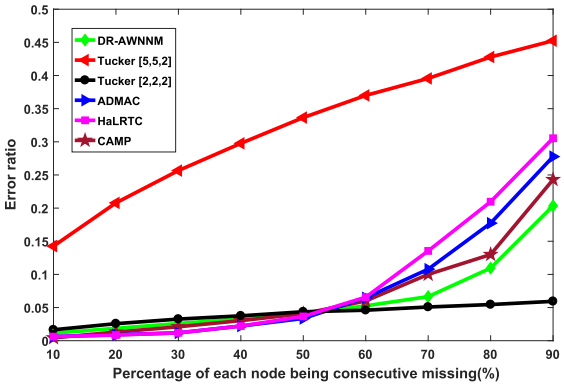

FIGURE 9. Tensor-based multi-attribute sensor data reconstruction, with consecutive missing pattern.

when the data missing rate of each node is lower than 50%. However, with a slightly higher rank-[5,5,2], Tucker decomposition achieves poor performance. This means that to use the Tucker decomposition-based algorithm for accurately reconstructing sensor data, we should first obtain the correct n-rank. However, this is usually difficult in practice, especially when the tensor is incomplete. In addition, the error rate of the CAMP algorithm is also lower than other algorithms. When the sparsity becomes better and the compression ratio gets higher, the CAMP algorithm can effectively reduce the error and improve the accuracy. Notably, when the data missing rate exceeds 50%, the error rate of the proposed algorithm is considerably lower than that of the other three algorithms, and even when the data missing rate is as high as 90%, the error rate of the proposed algorithm is lower than 20%. The reason is that we combine the various attributes of the data to enhance the intrinsic relationship between the data, and our algorithm uses the K-means cluster algorithm to group sensors at the beginning, which greatly enhances the external connection of the data and improves the accuracy of the algorithm.

In addition, we present in Figure 10 the results obtained by using only the DR-AWNNM algorithm (DR-AWNNM1) to reconstruct the missing data under the RMP and the CMP models, here all parameters have the same value as DR-AWNNM algorithm. It can be seen that when the data loss rate is high, the DR-AWNNM1 algorithm can reconstruct the missing data well. Moreover, the DR-AWNNM algorithm is superior to the other algorithms, including DR-AWNNM1.

The solution of DR-AWNNM algorithm, ADMAC algorithm, and HaLRTC algorithm essentially involves finding the augmented Lagrange function of the target function, and then the ADMM algorithm is used to solve it. Hence, Fig.11 shows the convergence curves of the three algorithms in the continuous missing data mode, in which the abscissa represents the iteration number of algorithm convergence, and the ordinate represents the tolerance of the relative difference of outputs of two neighboring iterations. The red curve represents the proposed algorithm. It can be seen that our algorithm is convergent, and it is slightly superior to other algorithms.

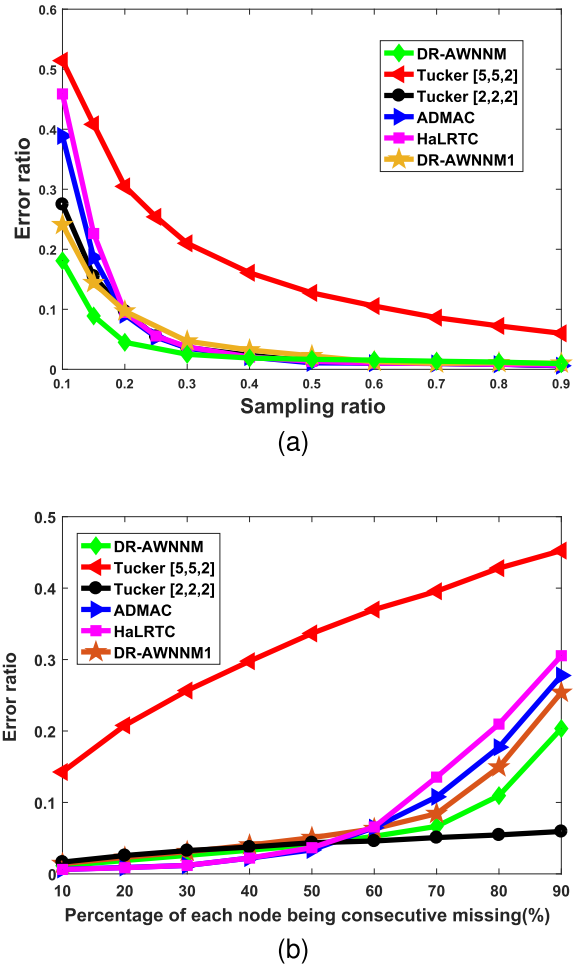

FIGURE 10. (a) DR-AWNNM algorithm and DR-AWNNM1 algorithm under RMP model. (b) DR-AWNNM algorithm and DR-AWNNM1 algorithm under CMP model.

The computational complexity of DR-AWNNM consists mainly of two parts. One is the complexity of the K-means clustering algorithm, and the other is the cost of matrix completion calculation. When we use K-means clustering algorithm to separate sensor nodes, we first set k cluster centers are randomly, and then calculate the distance from n points to k centers, the distance between the single image and the image is calculated as d, and last repeat the above process for t times, so the complexity of K-means is $O = k \times n \times d \times t$, namely $O(kndt)$. Usually k, d and t can be considered as constants, therefore, the computational cost of K-means can be simplified to $O(n)$, which is linear. From Alg.1 we know that the matrix completion algorithm mainly involves the iteration of $\mathcal{X}_i^{k+1}$, $\mathcal{M}_i^{k+1}$ and $\mathcal{U}_i^{k+1}$, the parameters iterated M times each time for N times, so the computational complexity of the matrix completion algorithm is $O(n^2)$, that is to say the complexity of the ADMAC algorithm and the HaLRTC algorithm is $O(n^2)$. Based on Eq.(24)-(26) we can learn that the calculation cost of the weighted nuclear norm is O(1). Therefore, the computational complexity of DR-AWNNM in this paper is also $O(n^2)$. In addition, Fig.8 and Fig.9 show

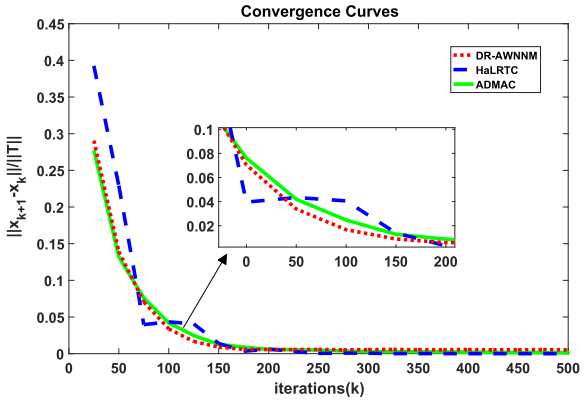

FIGURE 11. Convergence Curves with consecutive missing pattern.

that the DR-AWNNM algorithm has significant effects and far-reaching significance.

To summarize, the proposed DR-AWNNM algorithm outperforms the other tested algorithms on the dataset obtained from Intel Berkeley Research Laboratory and considering the two patterns of data missing. Specifically, the proposed algorithm provides better results under the random missing pattern than the consecutive missing pattern.

V. CONCLUSION

Missing data causes many difficulties in various IoT applications. Although this is inevitable due to the inherent characteristic of IoT, to solve the problem, it is imperative to estimate the missing data as accurately as possible.

In this paper, a missing data reconstruction method based on automatic WNNM was developed, and K-means clustering analysis was embedded into this forecasting process to improve the prediction accuracy. First, to apply spatial correlation between sensor nodes, we used the K-means clustering algorithm to separate sensor nodes into different groups.

Second, we considered sensor networks with multiple types of sensors in each node. Accounting for possible correlations among multiple-attribute sensor data, we provided a tensor-based method to estimate missing data and proposed an algorithm based on matrix rank-minimization method, namely, DR-AWNNM. We considered the weights between singular values and adaptively assign different weights to each singular value. Especially, when the weights are sorted in a non-descending order, the optimal solution can be easily obtained in closed-form.

Third, to demonstrate the feasibility of proposed method, a traditional EM-based Tucker decomposition algorithm and other excellent algorithms, namely ADMAC, HaLRTC and CAMP were introduced and compared. Our experiment suggested that the proposed method, on the one hand, when the data missing rate is low or the sampling rate is high, have outstanding performance as well as the other algorithms. On the other hand, when the data missing rate is high or the sampling rate is low, it performs much better than the other algorithms.

Finally, by comparing the convergence of different algorithms, we showed that the proposed algorithm is better, easier to converge, and less complex. In addition, we learned that the proposed algorithm is more suitable for missing data reconstruction under the random missing pattern than under the consecutive missing pattern. In future research, we will provide more insights into the performance of the DR-AWNNM algorithm in different scenarios.

REFERENCES

[1] IEEE Internet Initiative. (2015). *Towards a Definition of the Internet of Things (IoT)*. [Online]. Available: http://iot.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Revision1_27MAY15.pdf
[2] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.
[3] K. Rose, S. Eldridge, and L. Chapin, "The Internet of Things: An overview," Internet Soc., Reston, VA, USA, Tech. Rep., 2015, pp. 1–50.
[4] L. Xu, J. Wang, H. Zhang, and T. A. Gulliver, "Performance analysis of IAF relaying mobile D2D cooperative networks," *J. Franklin Inst.*, vol. 354, no. 2, pp. 902–916, Jan. 2017.
[5] L. Xu, J. Wang, Y. Liu, J. Yang, W. Shi, and T. A. Gulliver, "Outage performance for IDF relaying mobile cooperative networks," in *Proc. Int. Conf. 5G Future Wireless Netw.*. Springer, 2017, pp. 395–402.
[6] S. Chen, H. Xu, D. Liu, B. Hu, and H. Wang, "A vision of IoT: Applications, challenges, and opportunities with china perspective," *IEEE Internet Things J.*, vol. 1, no. 4, pp. 349–359, Aug. 2014.
[7] D. Bonino *et al.*, "ALMANAC: Internet of Things for smart cities," in *Proc. 3rd Int. Conf. Future Internet Things Cloud*, Rome, Italy, Aug. 2015, pp. 309–316.
[8] C. Perera, C. H. Liu, S. Jayawardena, and M. Chen, "A survey on Internet of Things from industrial market perspective," *IEEE Access*, vol. 2, pp. 1660–1679, 2014.
[9] O. Salman, I. Elhajj, A. Kayssi, and A. Chehab, "Edge computing enabling the Internet of Things," in *Proc. IEEE 2nd World Forum Internet Things (WF-IoT)*, Milan, Italy, Dec. 2015, pp. 603–608.
[10] M. Heil and R. Karban, "Explaining evolution of plant communication by airborne signals," *Trends Ecol. Evol.*, vol. 25, no. 3, pp. 137–144, 2010.
[11] L. Kong *et al.*, "Surface coverage in sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 234–243, Jan. 2014.
[12] B. Nathani and R. Vijayvergia, "The Internet of intelligent things: An overview," in *Proc. Int. Conf. Intell. Commun. Comput. Techn. (ICCT)*, Jaipur, India, Dec. 2017, pp. 119–122.
[13] Z. Yang, M. Li, and Y. Liu, "Sea depth measurement with restricted floating sensors," in *Proc. 28th IEEE Int. Real-Time Syst. Symp. (RTSS)*, Tucson, AZ, USA, Dec. 2007, pp. 469–478.
[14] Y. Liu, Y. He, M. Li, J. Wang, K. Liu, and X. Li, "Does wireless sensor network scale? A measurement study on GreenOrbs," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 10, pp. 1983–1993, Oct. 2013.
[15] X.-Y. Liu, K.-L. Wu, Y. Zhu, L. Kong, and M.-Y. Wu, "Mobility increases the surface coverage of distributed sensor networks," *Comput. Netw.*, vol. 57, no. 11, pp. 2348–2363, 2013.
[16] M. G. Lawrence, "The relationship between relative humidity and the dewpoint temperature in moist air: A simple conversion and applications," *Bull. Amer. Meteorol. Soc.*, vol. 86, no. 2, pp. 225–233, 2005.
[17] E. J. Candès and B. Recht, "Exact low-rank matrix completion via convex optimization," in *Proc. Found. Comput. Math.*, 2009, vol. 9, no. 6, pp. 806–812.
[18] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, "Weighted nuclear norm minimization and its applications to low level vision," *Int. J. Comput. Vis.*, vol. 121, no. 2, pp. 183–208, Jan. 2017.
[19] L. Kong, M. Xia, X.-Y. Liu, M.-Y. Wu, and X. Liu, "Data loss and reconstruction in sensor networks," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 1654–1662.
[20] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TinyDB: An acquisitional query processing system for sensor networks," *ACM Trans. Database Syst.*, vol. 30, no. 1, pp. 122–173, 2005.

[21] E. Granger, M. A. Rubin, S. Grossberg, and P. Lavoie, "Classification of incomplete data using the fuzzy ARTMAP neural network," in *Proc. IEEE-INNS-ENNS Int. Joint Conf. Neural Netw. (IJCNN), Neural Comput., New Challenges Perspect. New Millennium*, Como, Italy, vol. 6, Jul. 2000, pp. 35–40.
[22] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967.
[23] L. Pan and J. Li, "K-nearest neighbor based missing data estimation algorithm in wireless sensor networks," *Wireless Sensor Netw.*, vol. 2, no. 2, pp. 115–122, 2010.
[24] M. H. Le Gruenwald, "Estimating missing values in related sensor data streams," in *Proc. COMAD*, 2005, pp. 83–94.
[25] L. Gruenwald, H. Chok, and M. Aboukhamis, "Using data mining to estimate missing sensor data," in *Proc. 7th IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Omaha, NE, USA, Oct. 2007, pp. 207–212.
[26] J. Karjee, H. K. Rath, and A. Pal, "Efficient data prediction, reconstruction and estimation in indoor IoT networks," in *Proc. IEEE 6th Int. Conf. Future Internet Things Cloud (FiCloud)*, Aug. 2018, pp. 236–243.
[27] H. Wu, M. Suo, J. Wang, P. Mohapatra, and J. Cao, "A holistic approach to reconstruct data in ocean sensor network using compression sensing," *IEEE Access*, vol. 6, pp. 280–286, 2018.
[28] J. Chen, "The research of missing data recovery method based on feature analysis in wireless sensor networks," in *Proc. China Nat. Knowl. Infrastruct. (CNKI)*, 2016.
[29] G. Chen *et al.*, "Multiple attributes-based data recovery in wireless sensor networks," in *Proc. IEEE GLOBECOM*, Atlanta, GA, USA, Dec. 2013, pp. 103–108.
[30] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 208–220, Jan. 2013.
[31] Y. Shao, Z. Chen, F. Li, and C. Fu, "Reconstruction of big sensor data," in *Proc. 2nd IEEE Int. Conf. Comput. Commun.(ICCC)*, Chengdu, China, Oct. 2016, pp. 1–6.
[32] X. Wu, C.-L. Chuang, and J.-A. Jiang, "Temperature map recovery based on compressive sensing for large-scale wireless sensor networks," in *Proc. IEEE Int. Conf. Green Comput. Commun. IEEE Internet Things IEEE Cyber, Phys. Social Comput.*, Beijing, China, Aug. 2013, pp. 1202–1206.
[33] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010.
[34] D. Raval, G. Raval, and S. Valiveti, "Optimization of clustering process for WSN with hybrid harmony search and K-means algorithm," in *Proc. Int. Conf. Recent Trends Inf. Technol.(ICRTIT)*, Chennai, India, Apr. 2016, pp. 1–6.
[35] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Appl. Statist.*, vol. 28, no. 1, pp. 100–108, 1979.
[36] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Now Foundations and Trends, 2011. [Online]. Available: https://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=8186925
[37] [Online]. Available: http://db.csail.mit.edu/labdata/labdata.html/
[38] A. Smoliński, B. Walczak, and J. W. Einax, "Exploratory analysis of data sets with missing elements and outliers," *Chemosphere*, vol. 49, no. 3, pp. 233–245, 2002.

XIANG YU received the Ph.D. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 1995. From 1999 to 2002, he was Post-Doctoral/Associate Professor of postdoctoral mobile station in computer science and technology with the University of Electronic Science and Technology of China. He first had an industrial career in R&D, mostly in the field of radio communications.

From 1991 to 2008, he was the Chief Representative with Beijing Taige Electronics Co., Ltd., Chengdu Branch, and a Deputy Chief Engineer with the Seventh Research Institute, China Electronics Technology Group Corporation. He was the expert of major projects of national science and technology. He is currently a Professor with the Chongqing University of Posts and Telecommunications, Chongqing, China. His main research interests lie in the areas of digital communications and radio signal processing. He was a member of the China delegation for (ITU-R) WP5A, 5B, and 5C Conference.

XIA FAN received the B.S. degree in communication engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2016, where she is currently pursuing the master's degree in communication and electronic information engineering. Since 2016, she has been studying at the Wireless Communication Technology Innovation Laboratory, Broad Band Equipment Mobilization Center. Her research interests include mobile cloud computing, artificial intelligence, big data analysis in wireless communications, and the Internet of Things.

KAN CHEN received the B.S. degree in communications engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2017, where he is currently pursuing the M.S. degree in information and communication engineering. Since 2016, he has been studying at the Wireless Communication Technology Innovation Laboratory, Broadband Equipment Mobilization Center. His research interest mainly focuses on signal processing for wireless communications, especially the multicarrier transmissions for next generation wireless communication system.

SIRUI DUAN received the Ph.D. degree from the Beijing University of Posts and telecommunications, Beijing, China, in 2014. He is currently a Lecturer with the Chongqing University of Posts and telecommunications, Chongqing, China. His research interests mainly focus on signal processing for wireless communications, with an emphasis on multicarrier transmissions for next generation wireless communication system.