Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2015, Article ID 435391, 10 pages
http://dx.doi.org/10.1155/2015/435391

Hindawi

Research Article

A Missing Sensor Data Estimation Algorithm Based on Temporal and Spatial Correlation

Zhipeng Gao, Weijing Cheng, Xuesong Qiu, and Luoming Meng

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Weijing Cheng; chengweijing1990@bupt.edu.cn

Received 24 June 2015; Revised 22 August 2015; Accepted 31 August 2015

Academic Editor: Michelangelo Ceci

In wireless sensor network, data loss is inevitable due to its inherent characteristics. This phenomenon is even serious in some situation which brings a big challenge to the applications of sensor data. However, the traditional data estimation methods can not be directly used in wireless sensor network and existing estimation algorithms fail to provide a satisfactory accuracy or have high complexity. To address this problem, *Temporal and Spatial Correlation Algorithm* (TSCA) is proposed to estimate missing data as accurately as possible in this paper. Firstly, it saves all the data sensed at the same time as a time series, and the most relevant series are selected as the analysis sample, which improves efficiency and accuracy of the algorithm significantly. Secondly, it estimates missing values from temporal and spatial dimensions. Different weights are assigned to these two dimensions. Thirdly, there are two strategies to deal with severe data loss, which improves the applicability of the algorithm. Simulation results on different sensor datasets verify that the proposed approach outperforms existing solutions in terms of estimation accuracy.

1. Introduction

In recent years, with the development of sensing technology, wireless communication, and computing technology, wireless sensor network (WSN) [1] has been a focus of research and attracts strong attention from military, industry, and academia. In many applications of WSN, data loss [2, 3] is common due to limited resources of sensor nodes [4], interference of noise, and influence of environment. Even in some special situation, this phenomenon is very serious [5] which brings a big challenge for a variety of sensor data processing. If these missing values cannot be filled in accurately, the existing analysis tools cannot be applied. If the missing data are directly deleted, a large amount of raw data will be lost which will reduce the accuracy and reliability of analysis results and cause a great waste of energy. Data estimation algorithms can effectively solve this problem, and they provide strong support for query [6], aggregation, transmission, and warning [7]. So missing data estimation is particularly important for various applications of WSN.

However, the traditional data estimation methods [8] cannot be directly used in WSN. Sensor data estimation methods should consider the characteristics of the application system and sensor data. While many studies on sensor data estimation have been conducted and some achievements have been made, there are still some issues unresolved such as underutilization of sensor data's properties, high computational complexity, and low estimation accuracy.

We present a *Temporal and Spatial Correlation Algorithm* (TSCA) to estimate missing data in this paper. There are four main innovations of this algorithm. Firstly, it saves all the data sensed at the same time as a time series, and the most relevant series are selected as the analysis sample, which improves efficiency and accuracy of the algorithm significantly. Secondly, it selects the most data-relevant sensor nodes and gets spatial estimation based on comprehensive instantaneous rate of change. In the time dimension, it differentiates the order of past frames to estimate the missing rate which highlights the timeliness of sensor data. Thirdly, different

weights are assigned to temporal and spatial dimensions to get the final result. Finally, there are two strategies to deal with severe data loss, which improves the applicability of the algorithm.

The rest of this paper is organized as follows. Section 2 presents the classic estimation algorithms of missing sensor data. Section 3 presents the framework of the algorithm proposed in this paper. Section 4 describes specific design of our algorithm and extends to severe loss scenes. Section 5 evaluates the proposed approach through simulation experiment. Section 6 concludes this paper.

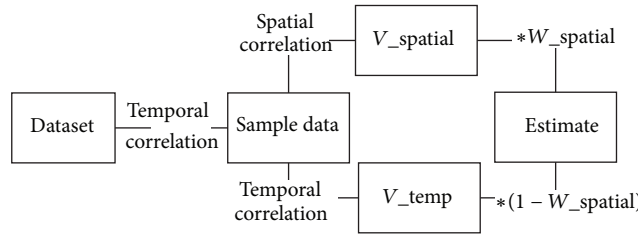


Figure 1: Framework of the algorithm in this paper.

2. Related Work

The estimation algorithms of missing data have been extensively researched in statistics, for example, Mean Substitution, Imputation by Regression, Expectation Maximization, Maximum Likelihood, Multiple Imputations, Bayesian Estimation, and Hot/Cold Deck Imputation [9]. However, none of these algorithms can be used in WSN, because they require the data miss at random and their efficiency is low.

To solve sensor data missing problem, Tiny DB [10] which is a mainstream sensor database system uses the mean of data sensed by other nodes directly as the estimated value. However, when the relationship among the sensor nodes is weak, the estimation result is not precise. MASTER-M algorithm [11] computes the similarity between sensor nodes and sorts them. It selects nodes which have high missing rate as seeds and clusters the whole network into several groups. MARSTER-tree is used to estimate missing data in each cluster. However, the relationship between the sensor nodes is not transitive; for example, $S1$ and $S2$, $S2$ and $S3$ are similar but $S1$ and $S3$ may not be similar. So in an n nodes network, C_n^2 calculations and comparisons need to be conducted in each process of clustering. If the similar relationships between the sensor nodes change rapidly, reclustering is needed constantly which will cause high computational complexity. Adaptive Multiple Regression (AMR) algorithm is proposed in [12]. Sample data and the most relevant sensor nodes are determined heuristically. Missing values are evaluated using linear regression models according to the data of the relevant nodes. The key steps in this algorithm are realized heuristically which will increase the computational complexity. In addition, the location-related nodes are not always data-related; for example, in a place with several heat sources [13], the nodes which are near heat sources but far apart from each other may be more relevant. So location-based association mining is not accurate. Assessment using linear regression models also increases errors. Grey System Estimate Algorithm (GSEA) [14] estimates missing values based on gray model. Minimized Similarity Distortion (MSD) [15] uses linear regression to evaluate the loss. The accuracy of both GSEA and MSD is poor.

The above algorithms only consider the temporal or spatial correlation and few algorithms take both of them into account. Environmental Space Time Improved Compressive Sensing (ESTI-CS) algorithm [16] is based on compressed sensing. This algorithm uses L1 norm optimization method for solving the reconstructed signal and it requires iteration which causes high complexity. Reference [17] proposes Trend Regression Expanding Cluster Interpolation (TRECI) algorithm which considers the change of sensor data over the time. Sensor nodes are divided into several groups dynamically and time interpolation assessments are conducted within each group. It only analyzes similarity rather than predicting the loss in the spatial dimension. Data Estimation using Statistical Model (DESM) [18] algorithm estimates the missing data based on the propagation characteristics of physical quantities in the time dimension; for example, according to the fact that light intensity is inversely proportional to the square of the distance, the light intensity can be estimated in certain region. In the spatial dimension, it estimates missing data based on the correlation between the estimated node and its surrounding nodes. The disadvantage of this algorithm is that it is only appropriate for attributes which have explicit physical models. Besides, the estimation in the spatial dimension is rough. Reference [19] proposes Mining Autonomously Spatial-Temporal Environmental Rules (MASTER) algorithm. It mines association of sensor data in temporal and spatial dimensions. A big drawback of this algorithm is that when the relationship among sensor data is weak, the prediction is very inaccurate.

3. Framework of Proposed Algorithm

Sensor data collected by a node Si can be seen as a time series $Si = [(V_{i1}, T_1), (V_{i2}, T_2), \ldots, (V_{in}, T_n)]$. V_{ik} is the sensing data at T_k. For any time T_k $(k = 1, 2, \ldots, n)$, if the data V_{ik} is lost, seeking the estimated value V'_{ik} and minimizing $|V'_{ik} - V_{ik}|$ are the missing data estimation problem.

From the comparison of difference between two consecutive intervals and difference between neighbors [16], we can see that most of measured data in real world always change stably; that is, there is little mutation on environmental value between adjacent time slots. In addition, environments are often smooth in a small area; that is, over a period of time, environmental values are similar among some nodes. Thus, we can use spatiotemporal correlations to estimate the missing data.

Considering that the existing missing data estimation algorithms have not made full use of features of sensor data and they have high computational complexity as well as low accuracy, this paper proposes a missing data estimation algorithm based on temporal and spatial correlations as shown in

Figure 1. The evaluation result of this algorithm is Estimate which can be computed by the following formula:

$$\text{Estimate} = \sum_{i=1}^{s\_n} wi * V\_\text{Spatial} + \left(1 - \sum_{i=1}^{s\_n} wi\right) * V\_\text{Temple}, \quad (1)$$

where $V_$Spatial and $V_$Temple are the analysis results of spatial and temporal correlations. wi is the weight of each relevant sensor node. s_n is the number of sensor nodes used to estimate the missing data.

This algorithm consists of three parts:

(i) Firstly, the algorithm needs to determine the sample data used in the process of analysis. Because sensor data is time-sensitive, using a different number of sensor data for analysis will get different results. Relationship between the sensor nodes in different periods is not the same, so selecting appropriate data used for analysis is important. Sensor nodes sense data periodically. The algorithm in this paper saves data sensed by all the nodes at the same time as a series. Continuous period produces continuous time series. For example, sensed data at $t_i, t_{i+1}, t_{i+2}, \ldots$ can be saved as the continuous time series $(V_{S1t_i}, V_{S2t_i}, V_{S3t_i}, \ldots, V_{Smt_i})$, $(V_{S1t_{i+1}}, V_{S2t_{i+1}}, V_{S3t_{i+1}}, \ldots, V_{Smt_{i+1}})$, and $(V_{S1t_{i+2}}, V_{S2t_{i+2}}, V_{S3t_{i+2}}, \ldots, V_{Smt_{i+2}}), \ldots$. The most relevant time series are selected based on the correlation function as the sample. It cannot only ensure that there are no redundant sample data which will reduce the computational complexity but also ensure that the sample data has the strongest correlation with missing data which will improve the accuracy of the analysis.

(ii) Secondly, correlation analyses are conducted in the spatial dimension. The distance between sensor nodes is defined according to the requirement of estimation. The most relevant sensor nodes are selected based on the distance function through analyzing the aforementioned sample data. Those relevant nodes are used to get spatial estimation. The weight of each relevant node wi is determined according to the average correlation coefficient with the estimated node.

(iii) Thirdly, in the time dimension, estimation is based on the sample data sensed by the estimated node. In order to give full play to the timeliness of data, past frames are distinguished chronologically during the process of analysis, so the contribution of newer data is greater. The weight of temporal estimation is $1 - \sum_{i=1}^{S\_n} wi$. Temporal and spatial results are integrated to obtain the final estimation value.

4. Detailed Design of TSCA

4.1. Select Sample Data. The relationship between the sensor nodes in WSN will change over time, so analyzing different sample data will generate different relationship, and we get different assessment values. In addition, the size of sample data will have a great impact on the assessment results. Due to the interference of environmental noise, too little sample data cannot reflect the spatiotemporal correlation of sensor data fully, while excessive sample data reflect the average value over an extended period of time rather than the instantaneous correlation which will reduce the accuracy of the assessment. Therefore, the values and the size of sample data should be determined as accurately as possible.

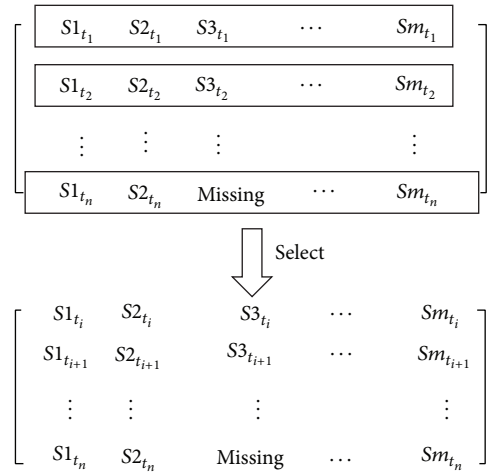


Figure 2: Select sample data.

Considering the fact that the spatiotemporal correlation of sensor data approximately remains constant in a short period of time, when we assess the missing data at t_n, data close to t_n should be selected accurately as the sample.

In WSN, sensor nodes are deployed in the given area. All the sensor nodes can be listed as $(S1, S2, S3, \ldots, Sm)$. These sensor nodes report sensing data at a certain time interval. At time t_i, all the reported data constitute a time series $S(t_i) = (S1_{t_i}, S2_{t_i}, S3_{t_i}, \ldots, Sm_{t_i})$. Data sensed at many contiguous moments form a random process $S(t)$, as shown in Figure 2. Assuming that certain sensor data loses at t_n, we analyze its average correlation with the former time series to determine the optimal sample data:

$$R = \frac{1}{n - t\_k} \sum_{j=n-1}^{t\_k} Rss\left(S_{t_n}, S_{t_j}\right) \quad (2)$$

$$\text{objective:} \quad \min k$$

$$\text{subject to:} \quad R = \max(R).$$

As validated by practical data, the correlation of time series is basically stable in a short period of time and then follows a decreasing trend. So we can get the most relevant sample data $t\_k \sim (n-1)$ based on formula (2). t_k is determined heuristically which is initially set to $n-1$. Correlation between t_n and t_{n-1} is calculated firstly; then, t_k moves forward and the average correlation values are calculated until the average correlation function is maximized. In Figure 2, we can see that $t\_k = i$, so the data between $t_i \sim t_{n-1}$ are the

Input:
$S_{m\times t}$: matrix of sensor data
Output:
$S_{m\times(t-i+1)}$: a collection of sample data
Main Steps:
(1) $Z_t \leftarrow \text{normalize}(S_t)$
(2) $R \leftarrow 0$
(3) for $i = t - 1$ to 1 do
(4) $\quad Z_i \leftarrow \text{normalize}(S_i)$
(5) $\quad Rss(S_t, S_i) \leftarrow Z_t Z_i^T$
(6) $\quad R\_\text{last} \leftarrow R$
(7) $\quad R \leftarrow (R + Rss(S_t, S_i))/(t - i)$
(8) $\quad$ if $R < R\_\text{last}$
(9) $\quad\quad$ return i;
(10) end for

Algorithm 1: Procedure SelectSampleData.

sample data. *Rss* which is the value of correlation between two time series can be computed as in the following formula:

$$Rss\left(S_{t_i}, S_{t_{i-1}}\right) = Z_{t_i} Z_{t_{i-1}}^T, \tag{3}$$

where Z_{t_i} is the standardized result of vector $S(t_i)$:

$$Z_{t_i} = \text{normalize}\left(S_{t_i}\right) = \left(\frac{S1_{t_i}}{\sqrt{S1_{t_i}^2 + S2_{t_i}^2 + \cdots + Sn_{t_i}^2}}, \right.$$
$$\frac{S2_{t_i}}{\sqrt{S1_{t_i}^2 + S2_{t_i}^2 + \cdots + Sn_{t_i}^2}}, \ldots, \tag{4}$$
$$\left. \frac{Sn_{t_i}}{\sqrt{S1_{t_i}^2 + S2_{t_i}^2 + \cdots + Sn_{t_i}^2}} \right).$$

The pseudocode of selecting process is described as in Algorithm 1.

4.2. Spatial Correlation

Definition 1. If the sample datasets (data sensed between $t_i \sim t_{n-1}$) reported by sensor nodes i, j are Si and Sj, data dissimilarity of these two nodes is $d\_\text{diff}(Si_{t_n}, Sj_{t_n}) = |Si - Sj|$, the collections of lost data are Si_miss and Sj_miss, the frequency of data loss at the same time is $d\_\text{miss}(Si_{t_n}, Sj_{t_n}) = |Si\_\text{miss} \cap Sj\_\text{miss}|$, and the size of sample data is $\text{sample\_size} = |Si| = |Sj|$.

Definition 2. The distance between sensor nodes Si and Sj is $d(Si_t, Sj_t)$ at t:

$$d\left(Si_t, Sj_t\right) = \frac{\sqrt{d\_\text{diff}\left(Si_t, Sj_t\right)^2 + d\_\text{miss}\left(Si_t, Sj_t\right)^2}}{\text{sample\_size}}, \tag{5}$$
$$d\left(Si_t, Si_t\right) = 1.$$

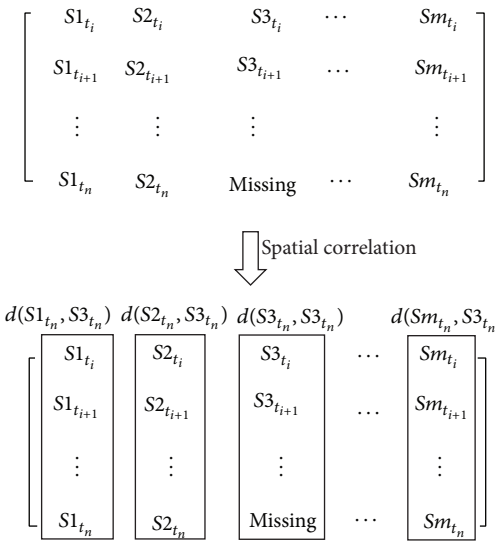


Figure 3: Spatial correlation.

If Sj loses data with the estimated node Si at the same time t, then $d(Si_t, Sj_t) = 1$. For example, in Figure 3, sensor node 3 will be estimated at t_n. If there are missing data of a node i $(i = 1, 2, 4, \ldots, n)$ at t_n, then $d(Si_{t_n}, S3_{t_n}) = 1$.

As shown in Figure 3, in order to estimate missing data of sensor node $S3$, distance between $S3$ and all the other nodes $S1, S2, S4, \ldots, Sm$ will be computed to get an array $d(S3_{t_n}) = [d(S1_{t_n}, S3_{t_n}), d(S2_{t_n}, S3_{t_n}), \ldots, d(Sm_{t_n}, S3_{t_n})]$. Select the nodes whose distance from $S3$ is smaller than the threshold value (the default is 0.2 in this paper) according to $d(S3_{t_n})$. These selected sensor nodes which have strong spatial correlation with node $S3$ compose the collection $S\_\text{Correlate}$.

Each node in $S\_\text{Correlate}$ estimates the missing data based on its instantaneous rate of change at t_n. Different weights are distributed to them according to the spatial correlation. The spatial correlation estimation is computed by the following:

$$V\_\text{Spatial} = \sum_{Si} wi * Vsj\left(t_{n-1}\right) * \frac{dV\left(Si_{t_n}\right)}{dt_n} \tag{6}$$
$$Si \in S\_\text{Correlate},$$

where Si is the sensor node in $S\_\text{Correlate}$. $VSj(t_{n-1})$ is the value of node Sj at the first moment before t_n. $dV(Si_{t_n})/dt_n$ is the instantaneous change rate of the relevant node Si at t_n which can be approximated as the change rate between t_n and t_{n-1}; that is, $dV(Si_{t_n})/dt_n = (V(Si_{t_n}) - V(Si_{t_{n-1}}))/(t_n - t_{n-1})$.

wi is the weight corresponding to Si, which is determined by the average correlation coefficient between the sensor nodes. The way to calculate wi is shown in the following:

$$wi = \frac{\psi\left(Si, Sj\right)}{\left|S_{\text{Correlate}}\right|} = \frac{\text{cov}\left(Si, Sj\right)}{\sigma si * \sigma sj * \left|S_{\text{Correlate}}\right|}$$
$$= \frac{E\left[\left(Si - E\left(Si\right)\right) * \left(Sj - E\left(Sj\right)\right)\right]}{\sigma si * \sigma sj * |S\_\text{Correlate}|}. \tag{7}$$

Input:
$S_{m\times(t-i+1)}$: sample data
S_{miss}: estimated sensor node
V: threshold of distance
Output:
V_Spatial: estimation value in spatial dimension
Main Steps:
(1) V_Spatial $\leftarrow 0$
(2) for $k = t$ to $t - i + 1$ do
(3) $\quad d\_S3[t-k+1] \leftarrow d(S_k, S_{\text{miss}})$
(4) $\quad$ if $d\_S3[t-k+1] <= V$
(5) $\quad\quad S_k \in S$_Correlate
(6) $\quad$ end if
(7) end for
(8) for each $S_k \in S$_Correlate
(9) $\quad w_k \longleftarrow \dfrac{\psi(S_k, S_{\text{miss}})}{|S_{\text{Correlate}}|}$
(10) $\quad r_k \longleftarrow \dfrac{dV(Sk_t)}{dt}$
(11) $\quad V$_Spatial $\leftarrow V$_Spatial $+ w_k * r_k * Vs_{\text{miss}}(t_{n-1})$
(12) end for

Algorithm 2: Procedure AnalysisInSpace.

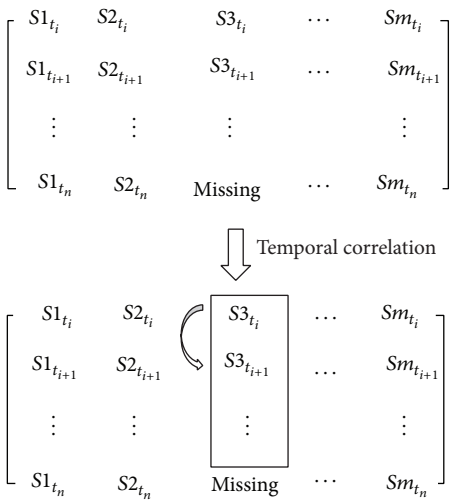


Figure 4: Temporal correlation.

The pseudocode of analysis in spatial correlation is described as in Algorithm 2.

4.3. Temporal Correlation. As shown in Figure 4, we estimate the missing data based on historical sample data of the estimated node. Evaluated result is obtained by a comprehensive measure on the variation of sample data. Change rate of data is defined as r_{t_n}:

$$r_{t_n} = \frac{\nabla V t_n}{t_n - t_{n-1}} = \frac{V t_n - V t_{n-1}}{t_n - t_{n-1}}, \tag{8}$$

where V_{t_n} is the sensing data of the estimated node at t_n.

Table 1: Weighted rate of change.

Time t_{n-1}	t_{n-2}	t_{n-3}	t_{n-4}	$\cdots$	t_i
Data Vt_{n-1}	Vt_{n-2}	Vt_{n-3}	Vt_{n-4}	$\cdots$	Vt_i
r_t	$\dfrac{\nabla Vt_{n-1}}{t_{n-1}-t_{n-2}}$	$\dfrac{\nabla Vt_{n-2}}{t_{n-2}-t_{n-3}}$	$\dfrac{\nabla Vt_{n-3}}{t_{n-3}-t_{n-4}}$	$\cdots$	$\dfrac{\nabla Vt_{i+1}}{t_{i+1}-t_i}$
w_t	$\dfrac{n-i-1}{\sum_{k=1}^{n-i-1} k}$	$\dfrac{n-i-2}{\sum_{k=1}^{n-i-1} k}$	$\dfrac{n-i-3}{\sum_{k=1}^{n-i-1} k}$	$\cdots$	$\dfrac{1}{\sum_{k=1}^{n-i-1} k}$

Change rates of all sample data are computed, and different weights w_{t_n} are given to them; then, we can get the weighted rate of change r_w:

$$r_w = \sum r_t * w_t. \tag{9}$$

Based on the weighted rate of change and value of the estimated node at t_{n-1}, we can get the temporal estimation V_Temple:

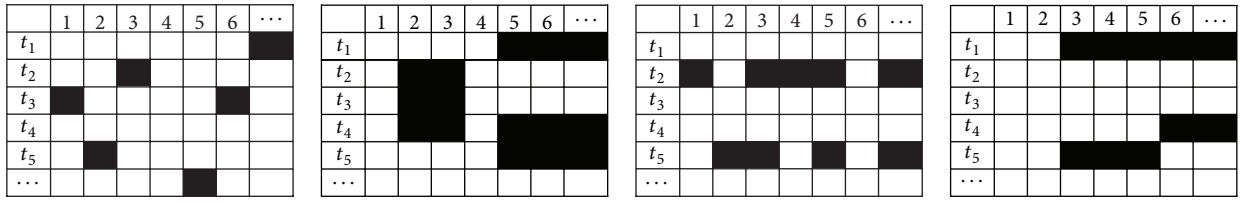
$$V\_\text{Temple} = V_{t_{n-1}} + r_w * (t_n - t_{n-1}). \tag{10}$$

The way to calculate the weighted rate of change is listed in Table 1, where r_t is the rate of change. w_t based on the sequence is assigned to each r_t. The pseudocode of analysis in temporal correlation is described in Algorithm 3.

4.4. Discussion. Unlike traditional missing data [20], sensor data have five typical patterns of missing [16] which are *Element Random Loss*, *Block Random Loss*, *Element Frequent Loss in Row*, *Successive Elements Loss in Row*, and *Combinational Loss*, as shown in Figure 5. The algorithm in this paper uses *Combinational Loss* mode, that is, any combination of the first four modes. In order to improve the applicability of our algorithm, we take a certain strategy to make the algorithm

Input:
$S_{\text{miss}\times(t-i+1)}$: sample data of estimated sensor node
Output:
V_Temple: estimation value in temporal dimension
Main Steps:
(1) for $j = t - 1$ to i do
(2) $r_{t_j} \longleftarrow \dfrac{\nabla V t_j}{t_j - t_{j-1}}$
(3) $w_{t_j} \longleftarrow \dfrac{j + 1 - i}{\sum_{k=1}^{n-i} k}$
(4) end for
(5) $r_w \leftarrow \sum r_{t_j} * w_{t_j}$
(6) V_Temple $\leftarrow V_{t_{n-1}} + r_w * (t_n - t_{n-1})$

Algorithm 3: Procedure AnalysisInTime.



(a) Element Random Loss (b) Block Random Loss (c) Element Frequent Loss in Row (d) Successive Elements Loss in Row

Figure 5: Data loss patterns in WSN (the black cells represent missing data).

suitable for some serious loss situations. This algorithm estimates missing values from the spatiotemporal aspects, so severe loss mainly shows up as rows or columns missing continuously.

As for severe data loss in time series, if data missing rate of a time series exceeds a certain threshold (the default is 40% in this paper), this time series will be ignored and we move forward to select sample data. As shown in Figure 6(a), the loss of time series at t_{n-2} is serious, so this moment will be ignored in the selection process of the sample data.

If data missing rate of the estimated sensor node does not exceed the threshold (the default is 50% in this paper), missing data will be ignored and the algorithm described before will be used to estimate the missing data directly. If the missing rate of sample data exceeds the threshold, the algorithm will obtain the final result through iteration. As shown in Figure 6(b), node 6 has a serious lack of sample data. Compute the values of node 6 at t_{n-6}, t_{n-5}, t_{n-3}, and t_{n-2} in turn until the missing rate is less than the threshold. Every iteration is conducted based on the results of last estimation. So if the result of previous estimation is not accurate enough, the estimation error in the next time will increase. However, the algorithm in this paper avoids the iteration in spatial correlation analysis by calculating the distance. Iteration occurs only in temporal correlation analysis. From the simulation results in the fifth section, it can be seen that the iterative error of our algorithm is small.

5. Performance Evaluation

The algorithm proposed in this paper is evaluated over real-world data, namely, Intel-lab dataset [21]. This dataset is a trace of readings from 54 sensor nodes deployed in the Intel Research Berkeley Lab. These sensor nodes collected light, humidity, temperature, and other information once every 30 s from February 28 to April 5, 2004.

Since the original dataset contains missing values, in order to evaluate the performance of the algorithm, we select the relatively complete part of the test data through deleting sensor nodes which contain serious data loss. For example, when the sampling interval is set to five minutes, there is a serious lack of sensor data in nodes 5 and 15 (with 90% of data lost). So data of these two sensor nodes will not be selected as sample. In this paper, we use the accuracy of the estimation as the evaluation criteria. Specifically, we use Root Mean Square Error (RMSE):

$$\text{RMSE} = \sqrt{\text{average}\left(Vsj\left(t_i\right) - V'sj\left(t_i\right)\right)^2}, \tag{11}$$

where $Vsj(t_i)$ is the known value which is assumed as missing data. $V'sj(t_i)$ is the estimated value of $Vsj(t_i)$.

To verify the effectiveness of the algorithm proposed in this paper, we compare it against other algorithms—AMR [12], TRECI [17], DESM [18], and MASTER [19].

	1	2	3	4	5	6	7	8	⋯
t_n	V_{1t_n}	V_{2t_n}	V_{3t_n}	V_{4t_n}		Missing	V_{7t_n}	V_{8t_n}	
t_{n-1}	$V_{1t_{n-1}}$	$V_{2t_{n-1}}$	$V_{3t_{n-1}}$	$V_{4t_{n-1}}$	$V_{5t_{n-1}}$	$V_{6t_{n-1}}$	$V_{7t_{n-1}}$	$V_{8t_{n-1}}$	
t_{n-2}				$V_{4t_{n-2}}$	$V_{5t_{n-2}}$		$V_{7t_{n-2}}$		
t_{n-3}	$V_{1t_{n-3}}$	$V_{2t_{n-3}}$	$V_{3t_{n-3}}$	$V_{4t_{n-3}}$		$V_{6t_{n-3}}$	$V_{7t_{n-3}}$	$V_{8t_{n-3}}$	
t_{n-4}	$V_{1t_{n-4}}$	$V_{2t_{n-4}}$	$V_{3t_{n-4}}$	$V_{4t_{n-4}}$	$V_{5t_{n-4}}$	$V_{6t_{n-4}}$	$V_{7t_{n-4}}$	$V_{8t_{n-4}}$	
t_{n-5}	$V_{1t_{n-5}}$		$V_{3t_{n-5}}$	$V_{4t_{n-5}}$	$V_{5t_{n-5}}$	$V_{6t_{n-5}}$	$V_{7t_{n-5}}$	$V_{8t_{n-5}}$	
t_{n-6}	$V_{1t_{n-6}}$	$V_{2t_{n-6}}$	$V_{3t_{n-6}}$	$V_{4t_{n-6}}$	$V_{5t_{n-6}}$	$V_{6t_{n-6}}$	$V_{7t_{n-6}}$	$V_{8t_{n-6}}$	
⋯									

(a) Severe data loss in time series

	1	2	3	4	5	6	7	8	⋯
t_n	V_{1t_n}	V_{2t_n}	V_{3t_n}	V_{4t_n}		Missing	V_{7t_n}	V_{8t_n}	
t_{n-1}	$V_{1t_{n-1}}$	$V_{2t_{n-1}}$	$V_{3t_{n-1}}$	$V_{4t_{n-1}}$	$V_{5t_{n-1}}$	$V_{6t_{n-1}}$	$V_{7t_{n-1}}$	$V_{8t_{n-1}}$	
t_{n-2}	$V_{1t_{n-2}}$	$V_{2t_{n-2}}$	$V_{3t_{n-2}}$	$V_{4t_{n-2}}$	$V_{5t_{n-2}}$		$V_{7t_{n-2}}$		
t_{n-3}	$V_{1t_{n-3}}$	$V_{2t_{n-3}}$	$V_{3t_{n-3}}$	$V_{4t_{n-3}}$	$V_{5t_{n-3}}$		$V_{7t_{n-3}}$	$V_{8t_{n-3}}$	
t_{n-4}	$V_{1t_{n-4}}$	$V_{2t_{n-4}}$	$V_{3t_{n-4}}$	$V_{4t_{n-4}}$	$V_{5t_{n-4}}$	$V_{6t_{n-4}}$	$V_{7t_{n-4}}$	$V_{8t_{n-4}}$	
t_{n-5}	$V_{1t_{n-5}}$		$V_{3t_{n-5}}$	$V_{4t_{n-5}}$	$V_{5t_{n-5}}$		$V_{7t_{n-5}}$	$V_{8t_{n-5}}$	
t_{n-6}	$V_{1t_{n-6}}$	$V_{2t_{n-6}}$	$V_{3t_{n-6}}$	$V_{4t_{n-6}}$	$V_{5t_{n-6}}$		$V_{7t_{n-6}}$	$V_{8t_{n-6}}$	
⋯									

(b) Severe data loss in sample data

Figure 6: Severe data loss patterns in this paper (the black cells represent missing data).

5.1. Convergence. Loss rate of raw data is about 5%. We verify the validity of the first step in this algorithm on the original temperature dataset where the sampling interval is set to 5 min. By calculating the average correlation, it can be known that the size of the sample data is 13. We choose a different number of data, and accuracy comparison of results is shown in Figure 7. It shows that a small or too large amount of data will cause an increase in the error rate. So we choose the smallest advisable size to ensure the accuracy while reducing the complexity of the algorithm. In Figure 8, we compare different algorithms against the size of required sample converging to the optimal solution. It can be seen that TSCA converges fast and has the best performance.

5.2. Estimation on Temperature. Error rate is compared among different algorithms on the original dataset where different sampling intervals are set, as shown in Figure 9. The spatiotemporal correlation of temperature is strong, so MASTER can obtain accurate relationships based on mining correlation rules. But a few of sensor nodes which are not associated with others will increase the estimation error, so its error is slightly larger than TSCA. As the sampling interval increases, temporal correlation of the sensor data weakens. TRECI and DESM use temporal correlation, so estimation error increases. However, the increase of DESM is slight because it also considers spatial correlation. The spatial correlation of the indoor sensor node in a short period of time remains substantially constant, so the sampling interval has little effect on AMR which only considers the spatial correlation. Particularly, TSCA takes the temporal and spatial correlation into account and assigns different weights according to the time series of data which makes newer data playing a more important role in the evaluation, so the size of the sampling interval has less effect on the results.

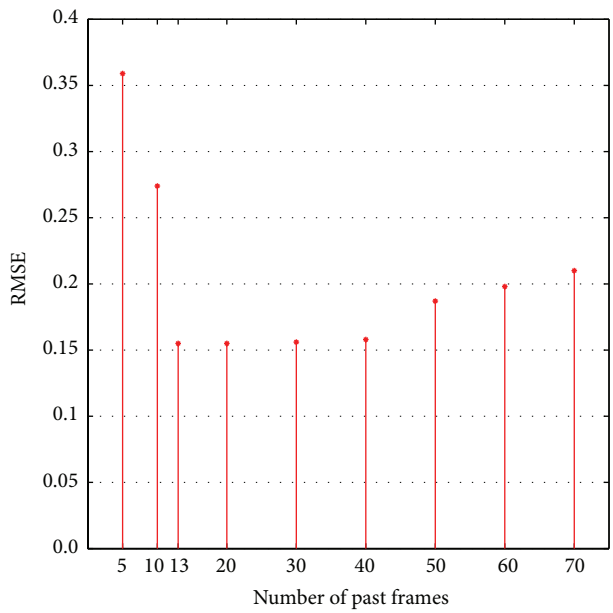


Figure 7: RMSE versus the size of sample data.

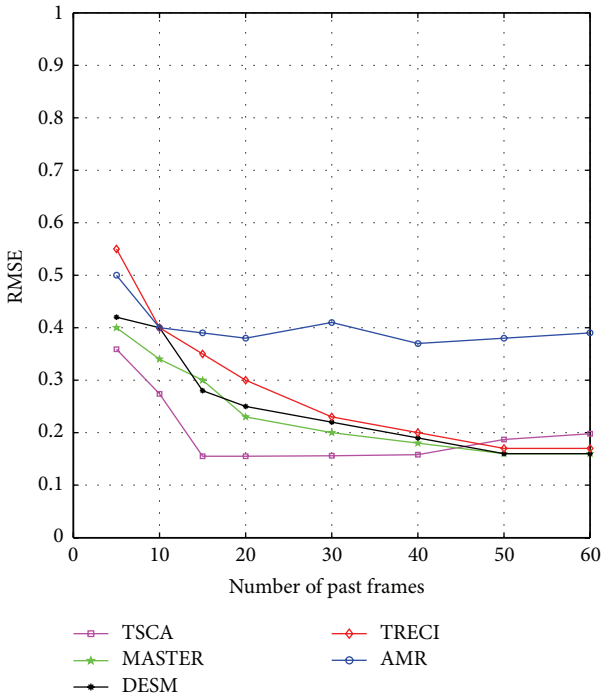


Figure 8: Convergence.

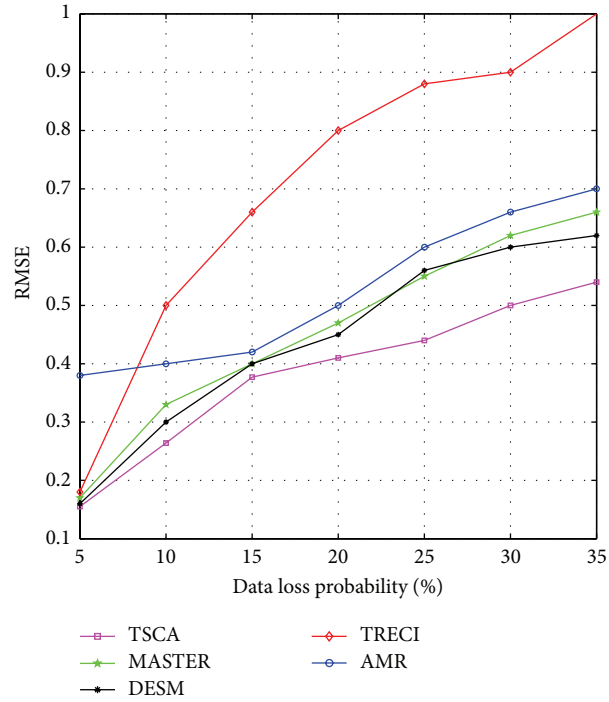


Figure 10: RMSE versus data loss on temperature.

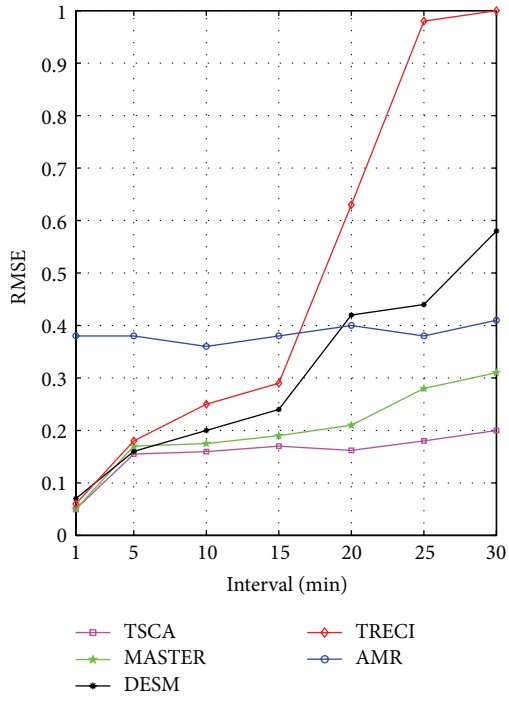


Figure 9: RMSE versus sampling interval on temperature.

According to [16], 23% of data are lost among 84,600 time slots (one month) of Intel Indoor dataset. Therefore, we conduct the error comparison among different algorithms where data missing rate is set as 5%–35% and the sampling interval is set as 5 min. Figure 10 shows that error of all the algorithms increases with the missing rate. This is because spatiotemporal correlation of sensor data will become weaker as missing rate increases. However, TSCA takes corresponding strategies based on the patterns of data loss as described in Section 4 which reduces errors greatly.

5.3. Estimation on Humidity. Error of humidity estimation is compared among different algorithms on the original dataset where the sampling intervals are set as 1–30 min, as shown in Figure 11. Compared with temperature, spatiotemporal correlations of humidity are weaker, and the spatial correlation is much weaker than the temporal one. AMR is only based on the spatial correlation, so its error is maximal. Like temperature, temporal correlation of the sensor data weakens with the sampling interval increasing. TRECI is mainly based on temporal correlation so its error increases remarkably. When the sampling interval reaches 30 min, error of TRECI exceeds AMR algorithm's. Results of the other three algorithms are similar, but the error of TSCA is still the smallest.

Figure 12 shows error rate in the situation of different data loss probability. When loss rate is more than 20%, spatial and temporal correlations of humidity are severely affected and error rate of DESM, TRECI, and AMR surges. Loss rate has a greater impact on the temporal correlation, so error rate of TRECI increases more significantly. TSCA is mainly based on the latest data and the missing data in the sample have been processed, so its performance remains relatively stable.

6. Conclusion

Considering the deficiencies of the existing algorithms for missing data assessment, TSCA is proposed in this paper based on spatiotemporal correlation of sensor data. This

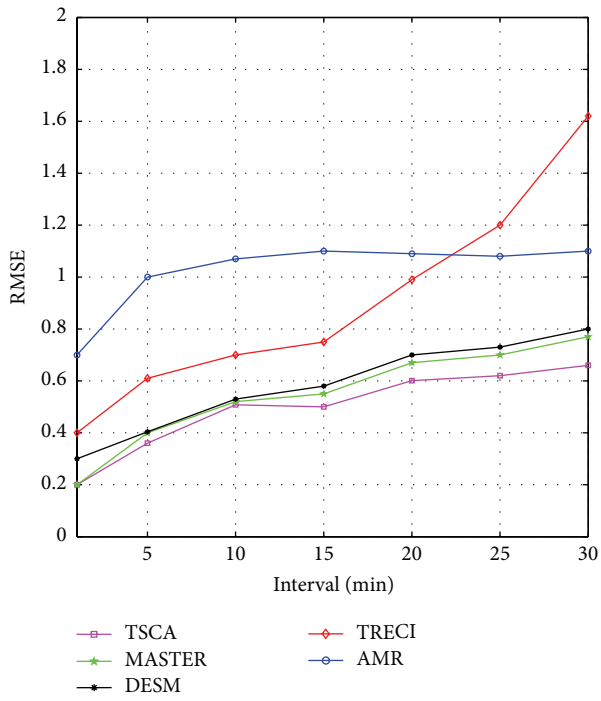


Figure 11: RMSE versus sampling interval on humidity.

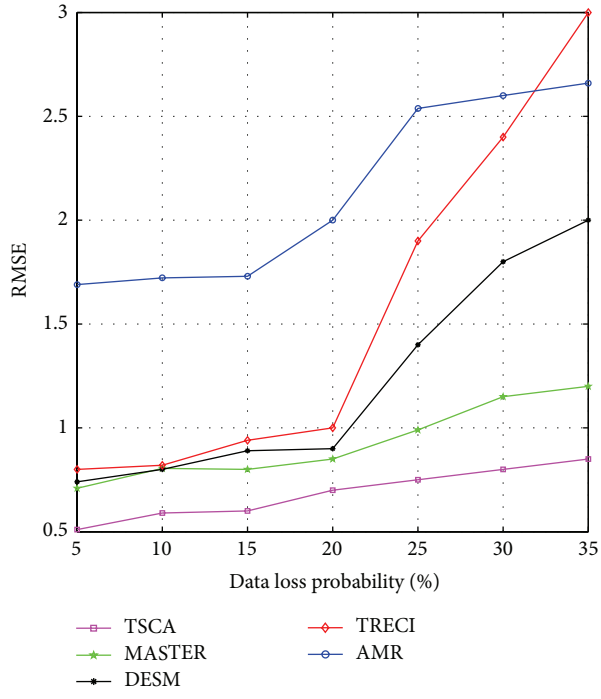


Figure 12: RMSE versus data loss on humidity.

algorithm selects the most relevant data as the analysis sample which ensures that there are no redundant sample data and the sample has the strongest correlation with the missing data. Thus, the efficiency and accuracy of this algorithm are significantly improved. What is more, a comprehensive analysis of the time and space is conducted to get estimation for missing data. Experimental results show that, no matter what the cases, TSCA always performs the best compared with other algorithms.

In the future, we can exploit the correlations between different attributes to further improve the accuracy of estimation; for example, light has an impact on temperature in many scenarios.

Conflict of Interests

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

Acknowledgments

This paper is partly supported by the National Natural Science Foundation of China (61272515, 61121061), Beijing Higher Education Young Elite Teacher Project (YETP0474), and National Science & Technology Pillar Program (2015BAH03F02).

References

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[2] C. Alippi, G. Boracchi, and M. Roveri, "On-line reconstruction of missing data in sensor/actuator networks by exploiting temporal and spatial redundancy," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2012.

[3] M. Moayedi, Y. K. Foo, and Y. C. Soh, "Adaptive Kalman filtering in networked systems with random sensor delays, multiple packet dropouts and missing measurements," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1577–1588, 2010.

[4] A. K. Fletcher, S. Rangan, and V. K. Goyal, "Estimation from lossy sensor data: jump linear modeling and Kalman filtering," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN '04)*, pp. 251–258, ACM, April 2004.

[5] L. Mo, Y. He, Y. Liu et al., "Canopy closure estimates with GreenOrbs: sustainable sensing in the forest," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys '09)*, pp. 99–112, ACM, November 2009.

[6] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "The design of an acquisitional query processor for sensor networks," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '03)*, pp. 491–502, ACM, San Diego, Calif, USA, June 2003.

[7] H. Zhu, Y. Zhu, M. Li, and L. M. Ni, "SEER: metropolitan-scale traffic perception based on lossy sensory data," in *Proceedings of the 28th Conference on Computer Communications (IEEE INFOCOM '09)*, pp. 217–225, April 2009.

[8] J. S. Bendat and A. G. Piersol, *Random Data: Analysis and Measurement Procedures*, John Wiley & Sons, New York, NY, USA, 2011.

[9] L. Gruenwald, H. Chok, and M. Aboukhamis, "Using data mining to estimate missing sensor data," in *Proceedings of the 17th IEEE International Conference on Data Mining Workshops (ICDM '07)*, pp. 207–212, October 2007.

[10] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TinyDB: an acquisitional query processing system for sensor

networks," *ACM Transactions on Database Systems*, vol. 30, no. 1, pp. 122–173, 2005.

[11] L. Gruenwald, H. Yang, M. S. Sadik et al., "Using data mining to handle missing data in multi-hop sensor network applications," in *Proceedings of the 9th ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pp. 9–16, ACM, 2010.

[12] L. Pan, H. Gao, H. Gao, and Y. Liu, "A spatial correlation based adaptive missing data estimation algorithm in wireless sensor networks," *International Journal of Wireless Information Networks*, vol. 21, no. 4, pp. 280–289, 2014.

[13] A. Silberstein, R. Braynard, and J. Yang, "Constraint chaining: on energy-efficient continuous monitoring in sensor networks," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 157–168, ACM, June 2006.

[14] F. Liu, Z. You, W. Shan, and J. Liu, "A grey system based missing sensor data estimation algorithm," in *Proceedings of the 2nd International Conference on Computer Science and Network Technology (ICCSNT '12)*, pp. 482–486, IEEE, December 2012.

[15] K. Niu, F. Zhao, and X. Qiao, "A missing data imputation algorithm in wireless sensor network based on minimized similarity distortion," in *Proceedings of the 6th International Symposium on Computational Intelligence and Design (ISCID '13)*, vol. 2, pp. 235–238, October 2013.

[16] L. Kong, M. Xia, X.-Y. Liu, M.-Y. Wu, and X. Liu, "Data loss and reconstruction in sensor networks," in *Proceedings of the 32nd IEEE Conference on Computer Communications (INFOCOM '13)*, pp. 1654–1662, Turin, Italy, April 2013.

[17] A. Appice, A. Ciampi, D. Malerba, and P. Guccione, "Using trend clusters for spatiotemporal interpolation of missing data in a sensor network," *Journal of Spatial Information Science*, no. 6, pp. 119–153, 2013.

[18] Y. Li, C. Ai, W. P. Deshmukh, and Y. Wu, "Data estimation in sensor networks using physical and statistical methodologies," in *Proceedings of the 28th International Conference on Distributed Computing Systems (ICDCS '08)*, pp. 538–545, IEEE, July 2008.

[19] H. Chok and L. Gruenwald, "Spatio-temporal association rule mining framework for real-time sensor network applications," in *Proceedings of the ACM 18th International Conference on Information and Knowledge Management (CIKM '09)*, pp. 1761–1764, ACM, November 2009.

[20] Z. Li, Y. Zhu, H. Zhu, and M. Li, "Compressive sensing approach to urban traffic sensing," in *Proceedings of the 31st International Conference on Distributed Computing Systems (ICDCS '11)*, pp. 889–898, IEEE, July 2011.

[21] S. Madden, Intel Berkeley research lab data, http://www.select.cs.cmu.edu/data/labapp3/index.html.