Simple Missing Data Estimation Algorithm in WSN Based on Spatial and Temporal Correlation

Mariem Ahmed, Walied Saeed, and Ashraf El-Sisi
Computer Science dept.,
Faculty of computers and Information, Menofia University.
mariemahmed264@gmail.com , walid_mufic@yahoo.com , ashrafelsisim@yahoo.com

Abstract—we have a common problem in wireless sensor networks which is the missing data problem due to the nature of the wireless communication and the limited resources of the sensor nodes. This problem can't be ignored because it has a negative effect on the applications that use the sensor data. Estimating these missing data is important for the applications that concern with the sensor data. However, the traditional estimation techniques failed to be applied with the sensor data and the existing techniques have high computation complexity, high computation time, or low accuracy. So we introduce the simplified Spatial and Temporal Correlation (STC) estimation algorithm which uses the most related surrounding previous data to increase the accuracy of the estimation and reduce incremental error. The proposed algorithm utilizes the time correlation by using the closet data before the time of missing and utilizes the space correlation by using the data of the nearest sensor depending on the missing pattern. The experimental results show that our algorithm can reduce the error in the estimating process compared with the other algorithms in most of the missing patterns.

Keywords—***Wireless sensor network; data mining; data missing; data estimating; spatial and temporal correlations.***

I. INTRODUCTION

In recent years, wireless sensor network (WSN) brings the attention of the world in many applications [1]. Especially for the goal of discovering the physical world we found WSN in all the places that are difficult to access in the forests [2], on an active volcano [3] and under the water [4]. WSN is adopted in the physical environments for gathering environmental information and using these data in the cyber worlds. WSN consists of a large number of sensors that have limited processing and storage resources and wireless communication. They collect the data from the local environment and send it to the sink node (base station). It is used in the applications of military, industry, and health. In most of those applications, we found the problem of the missing data due to the limited resources of the sensor nodes and the nature of the wireless network [5]. The data loss ratio is 23% in our Intel data set that used in this paper [6]. Missing data is considered as dirty data. Estimating the missing data is a preprocessing approach for cleaning the dirty data before using it in useful applications.

This problem can't be ignored because it brings challenges to the applications of WSN during processing or analysis the sensor data and can lead to wrong research results. Without filling in these data, a large amount of the sensor data can be lost reducing the accuracy and reliability of the application. Missing data estimation algorithms are important to solve this problem.

The traditional techniques for handling the missing data such as ignoring the missing data or re-querying the data are not suitable for the nature of WSN [7]. Ignoring the missing data is a bad solution, especially for applications that require high accuracy; Re-querying data takes more time and network bandwidth, and cannot be ensured to produce the original data. Hence, the need for estimating the missing sensor data has become critical.

We present a simple Spatial and Temporal Correlation (STC) algorithm for estimating the missing data in WSN using the most related data to estimate the missing value according to the pattern of missing. Firstly we analyze the real data, confirming the massive data loss and mining the data loss patterns in the WSN. Then depending on the missing pattern and if the missing sensor has previous data at past time intervals we decide to use the temporal correlation or the spatial one to estimate the missing data value.

The rest of this paper is organized as follows: Section 2 presents the related work dealing with missing sensor data. Section 3 presents the proposed algorithm. Section 4 presents the experiments and result analysis. Section 5 concludes this paper.

II. RELATED WORK

For solving the problem of the missing data in the sensor networks some algorithms use the association rule mining, others use the spatial correlations only, and some use both spatial and temporal correlations.

Using data mining in estimating the missing data in WSN is discovering the knowledge from the raw data then using it in estimating the missing data values [8]. The association rules are used to represent the relations between sensor nodes and data. The task of the association rule mining then is to find all the association rules that satisfy the user-specified threshold. Some of the most popular association rule mining algorithms are Window Association Rule Mining (WARM) [9]. Closed Item sets based Association Rule Mining (CARM) [10]. Mining Autonomously Spatio -Temporal Environmental Rules (MASTER) [11].

WARM [9] uses association rule mining to estimate the missing values. It uses the concept of sliding window (*w*) which store and use only the latest (*w*) rounds of data reports for the estimation process. The limitation of WARM is in the choosing of the window size: a small (*w*) includes a risk of losing data trends, while large (w) makes space overhead. An additional limitation in WARM is ignoring the temporal aspects of the data. CARM [10] uses association rule mining for estimating the missing data. It gives better estimation accuracy than WARM because it gets the estimation result from compact and complete information rather than two frequent item sets in the current sliding window. CARM needs less memory than WARM since it needs to store only the closed item sets information. CARM also ignores both spatial and temporal correlations. MASTER [11] algorithm brings a new concept to the missing data problem which is the spatial and temporal correlations. From the environmental rules, the normal data is stable in short time periods. Spatial correlation means that the sensors that are closed to each other sense similar data or related data. Temporal correlation means that data from the same sensor at contiguous short times are the same or changes smoothly. MASTER is an online Spatio-temporal mining algorithm that uses single scan with incremental data. The problem of the MASTER algorithm is when the relations between the sensor data are weak the estimation results are inaccurate. The association rules mining algorithms require the user's knowledge to predefine the threshold of support and confidence which may decrease the performance and the accuracy of the algorithm.

Some of the existing estimation algorithms depend on the spatial correlation only in the sensor network like K-Nearest-Neighbor (KNN) [12], the Grey System Estimation Algorithm GSEA [13] and the Adaptive Multiple Regression AMR [14].

K-Nearest-Neighbor (KNN) [12] is a classical local interpolation method. It estimates the missing value by utilizing the values of the nearest K neighbors. KNN provides low accuracy estimation results.

The Grey System Estimation Algorithm GSEA [13] is based on grey system model GM (1, 1) which calculate the correlations between the sensor of the missing data (the target sensor) and the neighbors based on the distance, choosing the sensor with the highest correlation value as the nearest one to the target sensor then construct the grey system model GM (1, 1) to estimate the missed data. The limitations of this algorithm are when the missing data is in a large area and for the unstable variables such as light, the estimation results is inaccurate because it ignores the temporal correlation. In the Adaptive Multiple Regression AMR algorithm [14] choosing the sample data and the relevant sensors are done heuristically which will increase the computational complexity. The algorithm uses the linear regression models with the data of the relevant nodes to estimate the missing data which increases the errors. In addition estimation algorithms based on the location is not accurate.

Another estimation algorithm uses both spatial and temporal correlation are Temporal and Spatial Correlation Algorithm (TSCA) [15] and Data Reconstruction with Spatial and Temporal Correlation in Wireless Sensor Networks (DRA) [16]. TSCA [15] firstly, it selects the sample data for each missing data that will be used in the estimation analysis. Secondly, it utilizes the spatial correlation by calculating the distance between each sensor node and the missing sensor then selecting the most relevant nodes based on the distance function giving them weights based on the average correlation coefficient with the estimated sensor. Then, it goes on the time dimension by using the sample data of the missing sensor at past time stamps to get the temporal estimation. Finally, it integrates the temporal and spatial estimation to get the estimated value in the next equation:

$$\text{Estimate} = \sum_{i=1}^{n} w_i * v\_spatial + (1 - \sum_{i=1}^{n} w_i) * v\_temple \quad (1)$$

Where *v_spatial* and *v_temple* are the result from the spatial and temporal analysis, w_i is the weight of each relevant sensor node, *n* is the number of sensors used to estimate the missing data.

DRA [16] is a spatial and temporal correlation reconstruction algorithm. It firstly estimates the missing data by the temporal correlation using the linear interpolation function of the data from the closet two times by the next equation:

$$ed\,(i,t) = z\,(i,t_1) + (z(i,t_2) - z(i,t_1)) * (t - t_1) / (t_2 - t_1) \quad (2)$$

where *ed(i, t)* is the estimated data for the sensor *i* at time *t*, t_1, t_2 are the closet two times when the data is not lost, and *z* (*i*, *t*) is the data of sensor *i* at time *t*. Then it uses the curved face reconstruction (NURBS) to reconstruct the data of the missed node by interpolating the missing values with constrains: 1- All the sensors are in the NURBS surface at any time. 2- The difference between the reconstructed data and the estimated data must be less than a given threshold. The details of a NURBS can be found in [17]. DRA considers the estimated data as the original data and iterates to minimize the difference between the estimated data and the reconstructed data. This process is done for each time until the difference is less than the given threshold which takes more time for reconstructing each missing data.

III. PROPOSED ALGORITHM

The wireless sensor network consists of a group of sensors $S = \{s_1, s_2..., s_m\}$ collecting the environmental information in a periodical time *n*, such as the temperature, humidity, and so on, which are changing continuously.

Let *S* (*m***n*) be the sensor data that is correctly received in the form of a matrix of *m* sensors and *n* times, given a sensor S_{misss} missed data V_{Smiss} at time T_{miss}, *ED* the estimated value. We need to minimize the difference between the missed value and the estimated value to give accurate estimation results $| V_{Smiss} - ED |$.

Figure 1 demonstrates the four loss patterns. From [18] we found four patterns of the missing in the sensor data: **element random loss**, **block random loss**, **element frequent loss,** and **element sequence loss**. The **element random loss** presents the case where the missing data is random at any time. The **block random loss** presents a group of sensors in the same block that are losing their data at the same random time. The **element frequent loss** presents the case where any single sensor lost its data at frequent times. The **element sequence loss** represents the case that any single sensor lost its data at continuous time duration. In real-world, data loss always happens in a combination of some of the loss patterns discussed above.

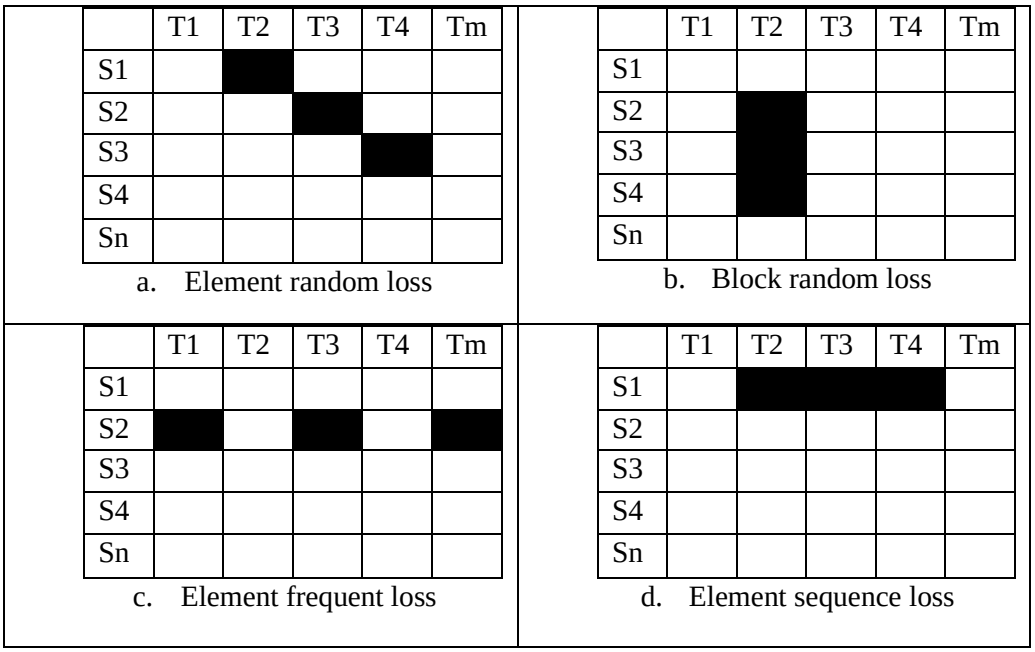

Figure 1: data loss patterns

Since STC utilizes the spatial and temporal correlations, this paper will discuss both of them and how each one is used in the proposed algorithm. The temporal correlations come from the fact that the environmental variables are stable and change smoothly at short time intervals for the same sensor node, so STC chooses the closet two times t_1, t_2 to the time of missing (T_{miss}), evaluates the rate of change for the data at these two times, where

$$\mathrm{Rt(temporal)} = (\mathrm{V}(\mathrm{S_{miss}}, \mathrm{t_1}) - \mathrm{V}(\mathrm{S_{miss}}, \mathrm{t_2})/(\mathrm{t_1} - \mathrm{t_2})) \quad (3)$$

The spatial correlations mean that data at the same time interval from neighbor sensors are mostly similar or related to each other. Choosing a different neighbor sensor will produce a different estimating result, so STC specifies the nearest neighbor sensor (z) to the missing sensor (S_{miss}) and calculates the rate of change for sensor z, where

$$\mathrm{Rt(spatial)} = (\mathrm{V}(\mathrm{z}, \mathrm{T_{miss}}) - \mathrm{V}(\mathrm{z}, \mathrm{T_{miss-1}}))/(\mathrm{T_{miss}} - (\mathrm{T_{miss-1}})) \quad (4)$$

STC algorithm consists mainly from 3 parts:

1. Firstly, we define the parameters of the missing used in the estimation process which are T_x and Rs. We save the closet two times t_1, t_2 in T_x vector, where S_{miss} record data before the time of missing when data is not lost to be used in the temporal estimation process. We save all the sensors that record data at the time of missing calling them the related sensors list in Rs to be used in the spatial estimation process. Depending on the data missing pattern we go on using the spatial correlations or the temporal one. If the sensor missed data at a long period time (sequence random loss) or the sensor has no data at previous times (T_x is empty) we use the spatial correlation estimation. If the sensor missed data for a short time period (element random loss), (frequent random loss) or (block random loss) we use the temporal correlation estimation.

2. The spatial estimation process uses the related sensors Rs to choose which sensor is the nearest one (z) to the missing sensor S_{miss} by using the Euclidian distance equation. Then use the data of z at the same time of missing in the estimation.

3. The temporal estimation process uses the data at the time vector T_x which contains t_1 and t_2 the nearest two times of the missing sensor S_{miss} when the data is not missing to get the estimated value.

Firstly, we define the parameters that will be used in the estimation process which are T_x and *Rs*. In line 2, 3 we save the closet previous two times in T_x vector, which are the first two times before the time of missing where S_{miss} record data and the data is not lost to be used in the temporal estimation process. In line 7, 8 we save all the sensors that record data at the time of missing calling them the related sensors list in *Rs* to be used in the spatial estimation process if T_x vector is empty, as shown in **Algorithm 1**.

Algorithm 1: Define the parameters of the missing

Input: $S(m*n)$ → matrix of sensor data
Consider: S_{miss} → sensor of missing data
T_{miss} → time of missing
Output: *Rs* → related sensors
T_x → the vector of the closet two times to T_{miss}
1- For $j = T_{miss}$ to 1:
2- If ($S(S_{miss}, j)$!= " Nan ")& (T_x. size()<2)
3- $T_x \leftarrow T_j$
4- End for
5- If T_x is empty
6- For i =1 to n:
7- If ($S(i, T_{miss})$!="Nan")
8- $Rs \leftarrow s_i$
9- End for

After identifying the parameters we move either in the spatial estimation or the temporal one. If the missing sensor has data before the time of missing we use the data of T_x for the temporal estimation. We use the spatial estimation if the sensor doesn't have data before the time of missing using the data from the group of related sensors *Rs*.

The spatial estimation uses the data from the nearest sensor in *Rs* at the time of missing, as shown in **Algorithm 2**. From line 10 to 13 we choose the nearest sensor (z) using the Euclidian distance between two sensors (the missing sensor and each sensor from the related sensors) based on their X and Y coordinates,

$$\text{Euclidian distance} = \sqrt{(X_s - X_i)^2 + (Y_s - Y_i)^2} \qquad (5)$$

Where (X_s, Y_s) are X and Y coordinates of the missing sensor (S_{miss}) and (X_i, Y_i) are X and Y coordinates of each sensor i in the related sensors group *Rs*.
In line 13, we calculate the rate of change for the most related sensor z where:

$$Rt(spatial) = (S(z, T_{miss}) - S(z, T_{miss-1}))/(T_{miss} - (T_{miss-1})) \qquad (6)$$

In line 15, the estimated data is the data of the most related sensor z at the time of missing plus the rate of change,

$$ED = S(z, T_{miss}) + Rt(spatial) \qquad (7)$$

Algorithm 2: Spatial Estimation Algorithm

Inputs: Rs → list of the related sensors
Xi, Yi → X and Y coordinates of sensor i
Output: ED → the estimated value using spatial correlation
Consider Xs, Ys → X and Y coordinates of the missing sensor
10- For each Si in Rs:
11- calculate Euclidian Distance (Si, Smiss) , where
Euclidian distance = $\sqrt{(X_s - X_i)^2 + (Y_s - Y_i)^2}$.
12- end for
13- Consider z → the sensor with the least Euclidian Distance
14- $Rt = (S(z, T_{miss}) - S(z, T_{miss-1}))/(T_{miss} - (T_{miss-1}))$
15- $ED = S(z, T_{miss}) + Rt$.

In the temporal estimation as shown in **Algorithm 3**, in line 16 we calculate the rate of change of the missing sensor (S_{miss}) for the first time before the missing time (t_1) and the previous time (t_2),

$$Rt(temporal) = (S(S_{miss}, t_1) - S(S_{miss}, t_2)/(t_1 - t_2)) \quad (8)$$

The estimated value is the data of the missing sensor (S_{miss}) at the first time before missing (t_1) plus the rate of change as shown in line 17,

$$ED = S(S_{miss}, t_1) + Rt. \quad (9)$$

Algorithm 3: Temporal Estimation Algorithm

Inputs:
T_x → the closet two times to missing sensor
Output:
ED → estimated data using the temporal correlation
16- Calculate the rate of change:
where $Rt = (S(S_{miss}, t_1) - S(S_{miss}, t_2)/(t_1 - t_2))$
17- $ED = S(S_{miss}, t_1) + Rt$

Complexity analysis of STC:

We will divide the computational complexity of STC into three parts. The first one is that of computing the parameters of the missing. The second is that of computing the spatial correlations. The third is the temporal correlations. The cost of computing the parameters of the missing is O(m+n) and the cost of computing the spatial or temporal correlations is O(m log m) or O(n) respectively. Where, m represents the number of sensor nodes, n represents the number of time slots.

IV. EXPERIMENTS AND RESULT ANALYSIS

Sensor Data Set:

We test our algorithm efficiency on the data set [6] from Intel Berkeley Research Lab. The arrangement of the sensors in the lab is shown in **figure 2**. The data were collected from the 54 Mica2Dot sensors with weatherboards deployed in the lab for 36 days. The sensors collect humidity, temperature, light, and voltage values once every 30 seconds.

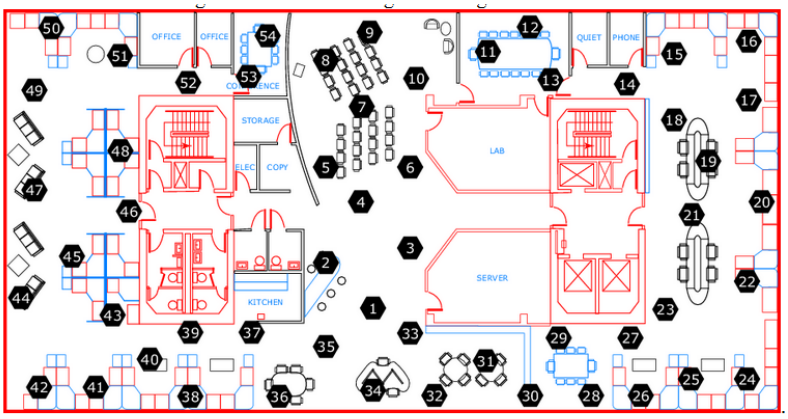

Figure 2: The diagram of sensor arrangement

The dataset model is shown in **figure 3**. The dataset is sorted sequentially. Each record consists of the next entries:

time: real	moteid: int	temperature: real	humidity: real	light: real	voltage: real

Figure 3: The schema of data record

Moteids are integer numbers from 1 to 54; motes' data may be lost. The temperature is in Celsius. Humidity is ranging from 0-100% and it is relative to temperature. Light is in Lux (1 Lux equals moonlight, 400 Lux equals a bright office, and 100,000 Lux equals full sunlight.) Voltage measures in volts, ranging from 2 to 3; the batteries were lithium-ion cells which keep a reasonably constant voltage over their lifetime; voltage is highly correlated with temperature.

Experimental result analysis:

There are many missing data in this data set. To evaluate our algorithm, we choose the relative complete part of the data and delete some readings to compare the estimated value with the real existing value. In this paper, we use the estimation accuracy to evaluate our algorithm. Specifically, we use the Root Mean Square Error (RMSE):

$$RMSE = \sqrt{average\ (V(s_j, t_i) - ED(s_j, t_i))^2} \quad (10)$$

Where $V(s_j, t_i)$ is the real data which is assumed as the missing data, $ED\ (s_j, t_i)$ is the estimated value of the missing data.

Firstly we compare STC against DRA and KNN on different loss patterns:

Estimation on Element Random Loss: As shown in figure 4, 5 we show the performance of the STC algorithm in element random loss pattern compared with DRA. The number of sensors that missed data ranges from 5 to 35, the X-axis represents the number of sensors with missing data, and the Y-axis represents the RMSE value, which reflects the estimation accuracy.

Estimation on Temperature, in figure 4 the estimation error in temperature in the STC algorithm varies from .03 to 0.17, DRA varies from 0.21 to 0.28 and KNN from 0.47 to 0.72. When the number of nodes with loss is 5, STC RMSE is 0.03, DRA .21, and KNN is 0.47, when the number of nodes with loss is 15, RMSE value of STC is 0.1, DRA 0.23, and KNN is 0.52, when the number of nodes with loss is 35, RMSE value of STC is 0.17, DRA 0.28, and KNN is 0.72.

Estimation on Humidity, in figure 5 the estimation error in humidity in STC algorithm varies from 0.1 to 0.37, DRA varies from 0.27 to 0.46, and KNN from 1 to 1.9, when the number of nodes with loss is 5, STC RMSE is 0.1 DRA 0.27, and KNN is 1, when the number of nodes with loss is 15, RMSE value of STC is 0.25, DRA 0.31, and KNN is 1.6, when the number of nodes with loss is 35, RMSE value of STC is 0.37, DRA 0.46 and KNN is 1.9.

Estimation results in element random loss on both temperature and humidity in the STC algorithm are better than DRA and KNN, since STC gives the play role priority to the temporal correlations and the temporal correlation in the element random loss is much stronger than the spatial one, KNN is a spatial estimation algorithm and DRA uses both the spatial and the temporal correlations in the reconstruction process. Estimation accuracy in temperature is better than in humidity, because of the temporal and spatial correlations in the humidity are weaker than in temperature.

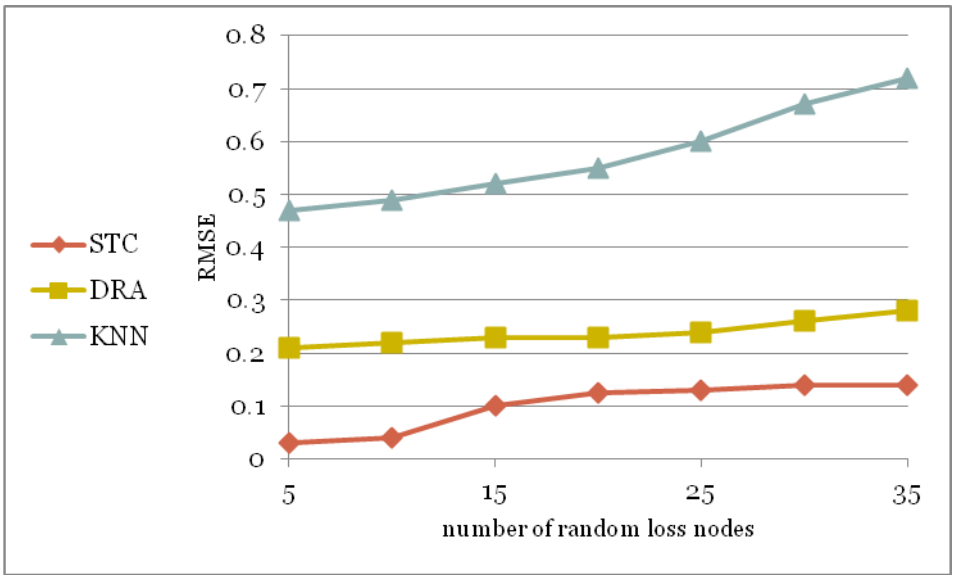

Figure 4: element random loss (temperature)

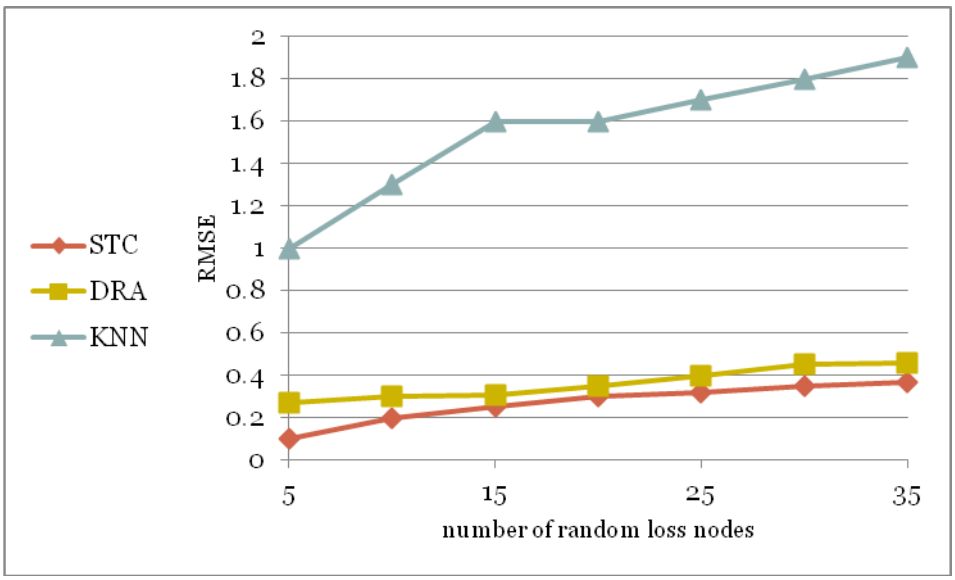

Figure 5: element random loss (humidity)

Estimation on Block Random Loss: Here, we show the performance of STC algorithm in block random loss pattern compared with DRA, and KNN. The X-axis represents the number of sensors in the block, and the Y-axis is the value of RMSE. The size of the blocks starts from 1 to 15.

Estimation on Temperature, as we can see in figure 6 RMSE in STC algorithm ranges from 0.07 to 0.1 where DRA ranges from 0.18 to 0.25 and KNN from 0.35 to 0.58, when the number of nodes with loss is 1, RMSE value of STC is 0.07, DRA 0.18, and KNN 0.35, when the number of nodes with loss is 5, RMSE value of STC is 0.08, DRA 0.2, and KNN 0.43, when the number of nodes with loss is 10, RMSE value of STC is 0.1, DRA 0.23, and KNN 0.51, when the number of nodes with loss is 15, RMSE value of STC is 0.08, DRA 0.25, and KNN 0.58.

Estimation on Humidity, as we can see in figure 7 RMSE in STC algorithm ranges from 0.15 to 0.25 where DRA ranges from 0.2 to 0.38 and KNN from 0.75 to 1.6, when the number of nodes with loss is 1, RMSE value of STC is 0.15, DRA 0.2, and KNN 0.75, when the number of nodes with loss is 5, RMSE value of STC is 0.2, DRA 0.23, and KNN 1.23, when the number of nodes with loss is 10, RMSE value of STC is 0.25, DRA 0.3, and KNN 1.56, when the number of nodes with loss is 15, RMSE value of STC is 0.25, DRA is 0.38 and KNN is 1.6.

Estimation results in the STC on block random loss outperform DRA and KNN. KNN has the maximum error rate because it utilizes the spatial correlations only. Since spatial and temporal correlations in temperature are stronger than humidity, the estimation accuracy in temperature is better than humidity. When the block size increases the DRA error has an upward trend, the estimation accuracy f STC isn't completely dependent on the block size. STC firstly looks for the data at previous time slots of the missing sensor. The block size affects the estimation results only if the spatial correlations are used (if the nearest neighbor sensor loses its data). When the block size is 5, most of the neighbor sensors still having data that enable the selection of the nearest sensor node and giving good estimation accuracy. When the block size is 10 or 15, this means that most of the sensor neighbors lost their data (if the spatial estimation is used) which makes the selection of the most related sensor is bad and gives us relatively high estimation error.

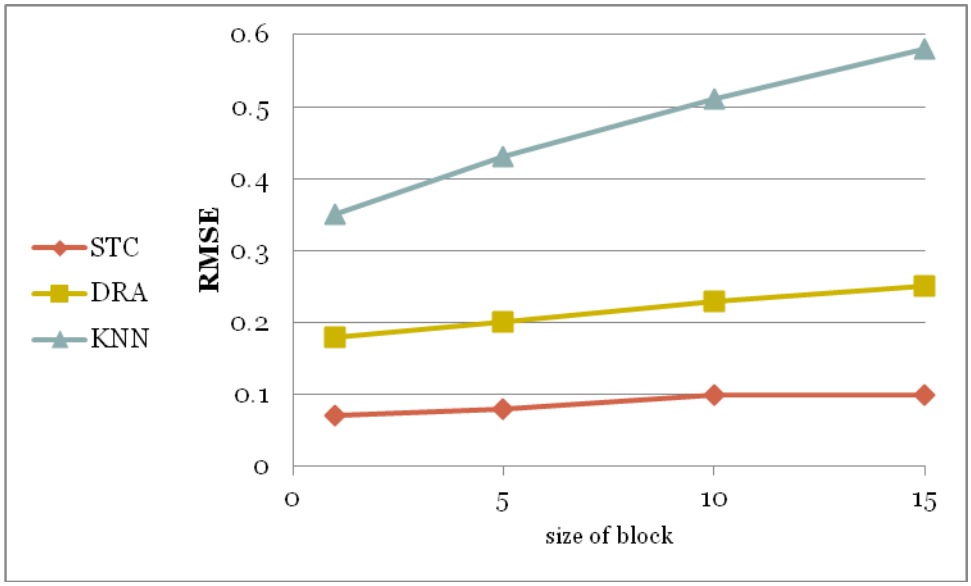

Figure 6: block random loss (temperature)

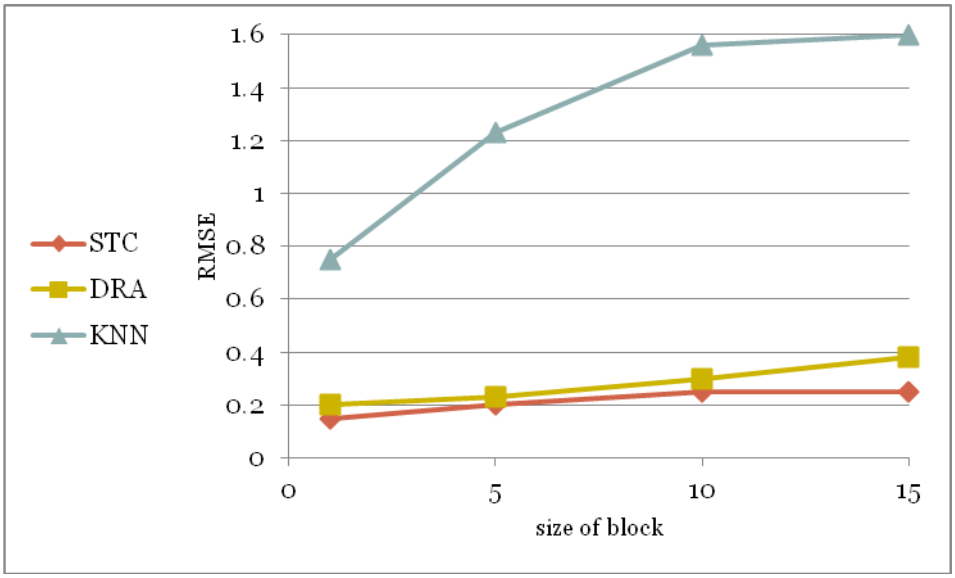

Figure7: block random loss (humidity)

Estimation on Element Sequence Loss: Here, we evaluate our algorithm, DRA and KNN in element sequence loss pattern. The number of time slots starts from 5 to 35.

Estimation on Temperature, as we can see from Figure 8, the RMSE value of STC ranges from 0.1 to 0.33, DRA ranges from 0.11 to 0.29 and KNN from 0.49 to .058. As the number of time slots increases, the RMSE in STC and

DRA shows an upward trend since the temporal correlation becomes weaker with the increasing loss for the same sensor in a sequence of time slots.

Estimation on Humidity, as we can see from Figure 9, the RMSE value of STC ranges from 0.3 to 0.57, the DRA ranges from 0.25 to 0.46 and KNN from 0.65 to 0.85.

The number of slots represents the number of time slots when the sensor missed data sequentially. As the number of time slots increases, the previous periods are diverging, this makes the temporal correlation becomes weaker or disappears which means that STC mostly will depend on the spatial correlation to get the estimated data giving high RMSE. Spatial correlations in humidity are weaker than temperature so the value of RMSE in STC is greater. DRA uses both spatial and temporal correlations so it has better accuracy. KNN uses the spatial correlations so its error changes up and down depending on the strong of the spatial correlation of the nearest sensor.
Estimation accuracy in element sequence loss on the temperature in the DRA is a little better, and STC estimation accuracy results on humidity are less than DRA. KNN has a maximum error rate.

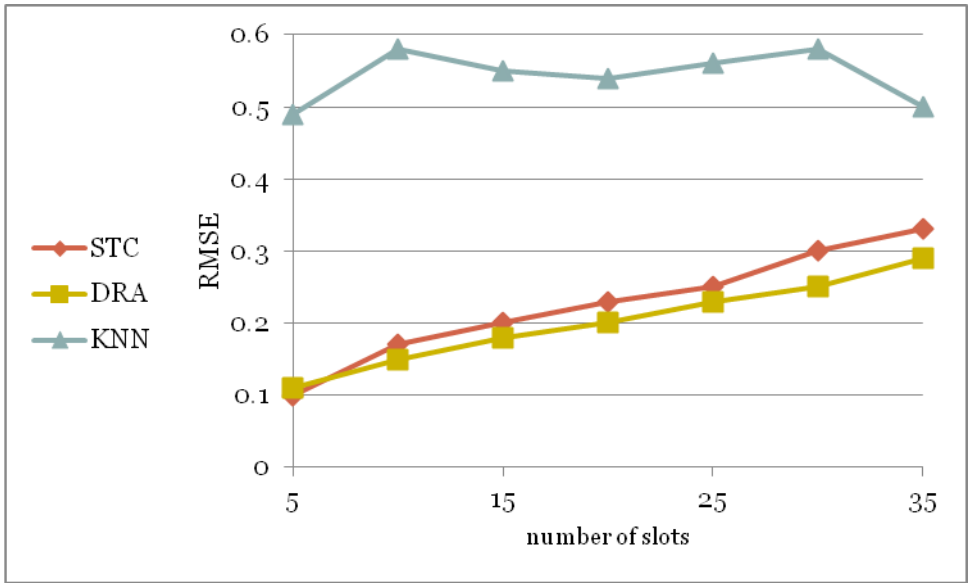

Figure 8: element sequence loss (temperature)

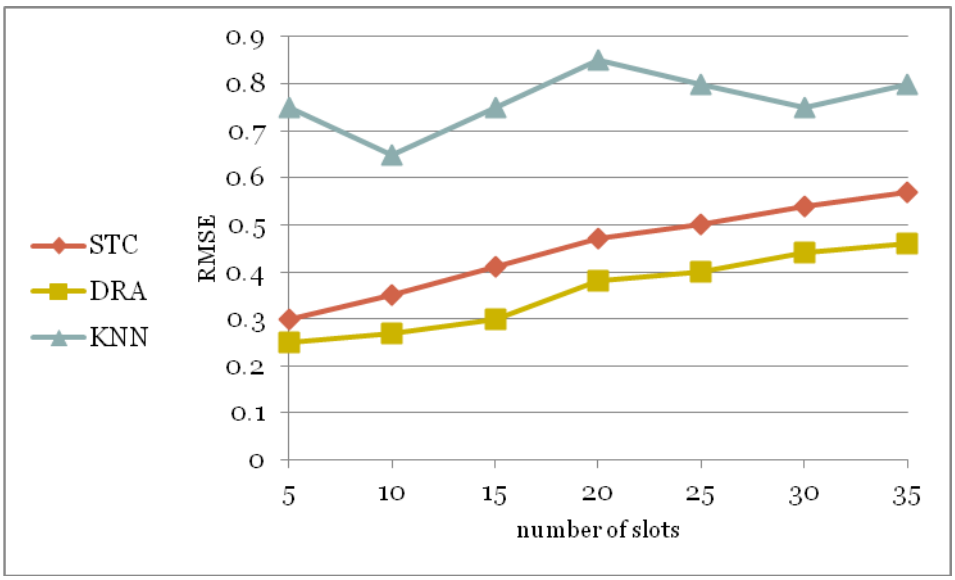

Figure 9: element sequence loss (humidity)

STC gives better estimation accuracy in the element random loss and the block random loss. DRA outperforms STC in element sequence loss.

Estimation on combinational Loss: We compare our algorithm against MASTER [11], AMR [13], and TSCA [14] to verify its effectiveness on the case of combinational loss.

The X axis represents the loss probability in the data starts from 5% to 35%. The Y axis represents the RMSE of the estimation.

Estimation on temperature: As the loss rate increases, the error rate increases. STC has the least error ranges from 0.12 to 0.4, TSCA ranges from 0.16 to 0.53, MASTER ranges from 0.17 to 0.65 and AMR has the highest error ranges from 0.39 to 0.7.

Spatial and temporal correlation in temperature is strong, so we find all the spatial and temporal correlation algorithms perform good estimation accuracy (STC, TSCA, and MASTER). STC outperforms all the spatiotemporal algorithms, since it selects the most related data in the estimation process and avoids the incremental error coming from the estimation iterations. TSCA integrates both spatial and temporal correlations to get the estimated result using selected sample data. When the sample data has increasing missing rate, the algorithm gets the estimated data results through iterations making the estimation error increase (in case of the temporal correlation analysis only) when the previous estimation result wasn't accurate enough. The association rules become weak when the missing rate increases, so MASTER has higher error rate than TSCA and STC. AMR depends on the spatial correlation only. Spatial correlations become weak when the missing rate increases, so DRA is the worst which is clear in figure 10.

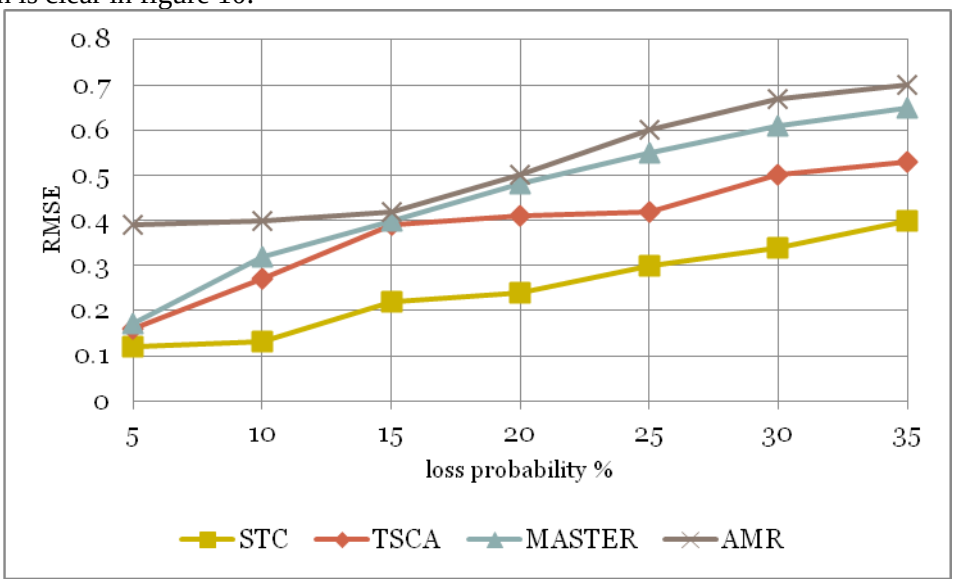

Figure 10: Estimation on Temperature

Estimation on Humidity: As the loss rate increases, the error rate increases. STC has the least error ranges from 0.33 to 0.64, TSCA ranges from 0.5 to 0.85, MASTER ranges from 0.75 to 1.4 and AMR has the highest error ranges from 1.7 to 2.75 as shown in figure 11.

Spatial and temporal correlations in humidity are weaker than temperature, so we find the error rate in the humidity estimation is larger than temperature. The spatial correlation is much weaker than the temporal, so AMR has the maximum error. MASTER error is higher than STC and TSCA since it depends on the association rules. Results of STC and TSCA algorithms are close, but the error of STC is still the smallest since it uses the nearest data in both spatial and temporal dimensions.

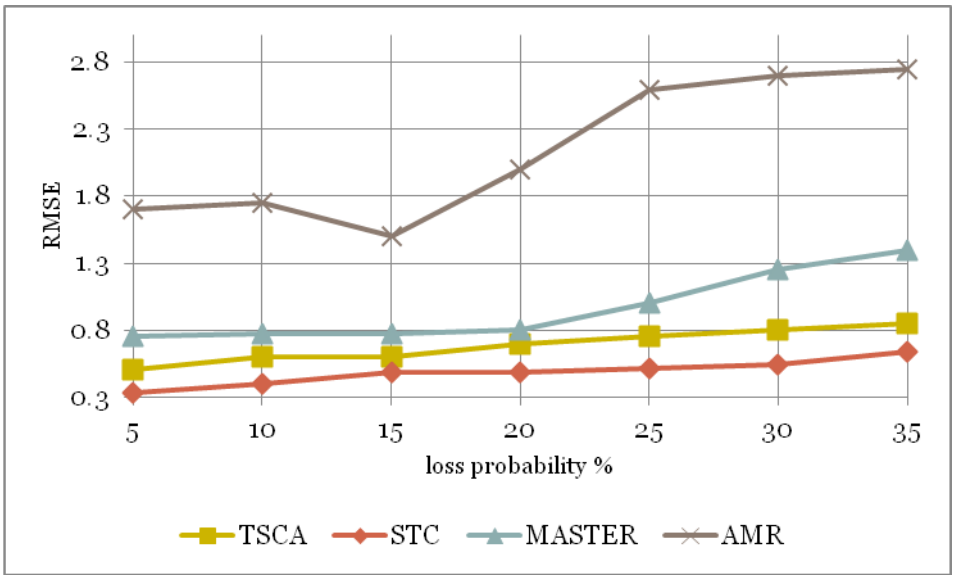

Figure 11: Estimation on Humidity

V. CONCLUSION

To solve the problem of the missing data in the WSN, STC is proposed for the data estimation that utilizes the spatial and temporal correlation among the data. Spatial and temporal correlation algorithms give better estimation accuracy than other algorithms. In short time periods of missing there is no need to use both the spatial estimation and the temporal one in each missing problem. Depending on the type of the missing we decide which correlation we will use thus gives our algorithm more simplicity. When the spatial correlations become weaker such in the humidity data, algorithms that use the spatial correlation only become inaccurate. The proposed algorithm tries to use the temporal correlation in the estimation process in most of the missing cases except those cases when the sensor doesn't have previous data. In this paper, we analyze five patterns of the data loss, i.e., block random loss, element random loss, element sequence loss element, frequent loss, and combinational loss. Then we introduce the system model details and problem definition. We also evaluate the efficiency of the proposed algorithm using the real data from the Intel Indoor project, compare it with other algorithms. The experimental results show that our data estimation algorithm provides more accurate estimation results than other algorithms

References

[1] J. Yick, B. Mukherjee, and D. Ghosal , “Wireless sensor network survey,” Computer Networks, vol. 52,no. 12,pp. 2292–2330, 2008.
[2] L. Mo, Y. He, Y. Liu, J. Zhao, S. Tang, X. Li and G. Dai, “Canopy Closure Estimates with Green Orbs: Sustainable Sensing in the Forest,” In Proc. of ACM SENSYS, Berkeley, CA, USA, 2009.
[3]G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees and M. Welsh, “Fidelity and Yield in a Volcano Monitoring Sensor Network”, In Proc of USENIX OSDI, Seattle, WA, USA, 2006.
[4] Z. Yang, M. Li, and Y. Liu, “Sea Depth Measurement with Restricted Floating Sensors,” In Proc. of IEEE RTSS, Tucson, AR, USA, 2007.
[5] 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.
[6] http://db.csail.mit.edu/labdata/labdata.html.
[7] J. S. Bendat and A. G. Piersol, “Random Data: Analysis and Measurement Procedures”, John Wiley & Sons, New York, NY, USA, 2011.
[8]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.
[9]M. Halatchev, and L. Gruenwald, “Estimating missing values in related sensor data stream”, In: COMAD, pp. 83–94 (2005).

[10] N. Jiang and L. Gruenwald, “Estimating missing data in data streams”, In: DASFAA, pp. 981–987 (2007).
[11] H. Chok, and L. Gruenwald, “Spatio-Temporal Association Rule Mining Framework for Real-time Sensor Network Applications”, In: ACM, *(2009)*
[12] T. Cover and P. Hart, Nearest Neighbor Pattern Classification, IEEE Trans. on Information Theory, vol. 13(1), pp. 21–27, January 1967.
[13] F. Liu1, Z. You1, W. Shan and J. Liu, “A Grey System Based Missing Sensor Data Estimation Algorithm”, in 2nd International Conference on Computer Science and Network Technology, 2012.
[14] 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.
[15] Z. Gao, W. Cheng, X. Qiu, and L. Meng, “A Missing Sensor Data Estimation Algorithm Based on Temporal and Spatial Correlation”, In: International Journal of Distributed Sensor Networks, (2015).
[16] Y. Zhang, H. Cheng, and D. Chen, “Data Reconstruction with Spatial and Temporal Correlation in Wireless Sensor Networks”, In: ACM 2016.
[17] L. Piegl and W. Tiller, the NURBS Book. Springer-Verlag, New York, 1997.
[18] L. Kong, M. Xia, X. Liu, M. Wu, and X. Liu, “Data Loss and Reconstruction in Sensor Networks”, In Proc. IEEE INFOCOM, 2013, pp. 1654–1662.