On-Node Correlation Based Data Reduction in WSN for Smart Agriculture

Christian Salim, Nathalie Mitton

▶ To cite this version:

HAL Id: hal-03016015

https://hal.archives-ouvertes.fr/hal-03016015v2

Submitted on 24 Nov 2020

On-Node Correlation Based Data Reduction in WSN for Smart Agriculture

Christian Salim
Inria
France
christian.salim@inria.fr

Nathalie Mitton
Inria
France
nathalie.mitton@inria.fr

Abstract—Nowadays, climate change is one of the numerous factors affecting the agricultural sector. Optimizing the usage of natural resources is one of the challenges this sector faces. For this reason, it could be necessary to locally monitor weather data and soil conditions to make faster and better decisions locally adapted to the crop. Wireless sensor networks (WSNs) can serve as a monitoring system for these types of parameters. However, in WSNs, sensor nodes suffer from limited energy resources. The process of sending a large amount of data from the nodes to the sink results in high energy consumption at the sensor node and significant use of network bandwidth, which reduces the lifetime of the overall network. In this paper, for data reduction, a data correlation and prediction technique is proposed both at the sensor node level and at the sink level. The aim of this approach is to reduce the amount of transmitted data to the sink, depending on the degree of correlation between different parameters. In this work we propose the Pearson Data Correlation and Prediction (PDCP) algorithm to detect this correlation. This data reduction maintains the accuracy of the information while reducing the amount of data sent from the nodes to the sink. This approach is validated through simulations on MATLAB using real meteorological data-sets from Weather-Underground sensor network. The results show the validity of our approach by reducing the amount of data by a percentage up to 69% while maintaining the accuracy of the information. The humidity values prediction based on the temperature parameter is accurate and the deviation from the real value does not surpass 7% of humidity.

Index Terms—Smart Agriculture; Data Correlation; Data Reduction; Data Prediction, Pearson coefficient

I. INTRODUCTION

Modern agricultural fields are in need of new and improved methods. Climate change and scarcity of water present some new challenges. Intelligent decision support tool is becoming primordial to solve some challenges and maximise the usage of the natural resources [9]. Wireless sensor networks (WSN) serve as low-cost monitoring systems in different domains (healthcare, industrial, video surveillance, environmental, etc...) [6]. In the agricultural fields, WSN can also be beneficial to survey the climate and soil parameters. In our scenario, we assume a WSN deployed for smart agriculture, periodically gathering environmental data from different sensor nodes and sending this data to a sink for further analysis [8].

WSNs must take account of the medium occupancy of the network, the bandwidth usage and most importantly the limited energy resources of the sensor nodes. This periodic cycle leads to a lot of redundant data sent to the sink, especially if no changes occur in the monitored feature (e.g. if the temperature stays stable). This big amount of periodic data is even more challenging at the sensor node level. To reduce the amount of data transmitted by the sensor nodes, data correlation is one of the solutions that can be adopted and adapted to the situation under study. In this paper, we introduce the Pearson Data Correlation and Prediction algorithm (PDCP). It consists of a data reduction technique based on a machine learning process to predict the data correlation between the same parameter on different neighbour nodes and between different parameters on a single node. To be able to detect whether two parameters are correlated or not, we applied the Pearson Correlation method where a Pearson correlation coefficient is computed based on numerous past values [4]. If a high correlation is detected, the nodes send less data to the sink. In the inter-nodes correlation, one of the nodes sends the correlated data, reducing interference at the same time. In different parameters case, in the intra-node correlation, the node only sends one value and the sink estimates the other one. As such, PDCP reduces the energy consumption of the communication process, since the nodes do not need to send all the captured values to the sink anymore while still ensuring high data accuracy at the sink. However a critical threshold between the estimated data and the real one is always present on the sensor node level to detect any absurd change in the values as explained in section III.

We conducted simulation using Matlab simulator and the results show the validity of our approach, by reducing the amount of sent data to the sink outperforming other approaches. For the inter-nodes correlation technique the temperature prediction is accurate and a maximum difference of 1 degree Celsius exists if compared to the real value. The data reduction is huge and reaches 100% for the two days of prediction. For the intra-node correlation, while predicting the humidity from the temperature parameter, we needed to send only 6 values out of the 96 existing sensed values and having a difference less than 7% for the humidity parameter prediction while comparing it to the real sensed value.

The remainder of this paper is as follows: Section II introduces the state of the art, Section III presents our data correlation technique and the different cases. The PDCP

algorithm is presented in Section IV. In Section V, some experimental results validate the approach. Finally, Section VI concludes the paper.

II. BACKGROUND AND RELATED WORK

Data reduction for WSN has already received much attention in the last years. Different data reduction techniques are present in the literature. In this section we will list and explain some of these approaches while focusing on the data correlation and machine learning techniques.

Data correlation is a type of data reduction, it tests the correlation between several characteristics. In the literature, a lot of approaches based on data correlation have been proposed. In [11], the authors proposed a correlation system based on a Bayesian inference approach in order to avoid transmitting data that can be reconstructed from other data. Machine learning for data prediction is widely used for data reduction as in [18], [13], [15], [14] and [7]. In the dual prediction model [18], the sensor node and the sink both predict the next values of the monitored feature simultaneously. In [18], the authors propose a machine learning technique to predict the next values, while sending all the data in the learning phase to the sink. In this approach, the authors detect a trend directly after a single change, which can cause some problems for the learning process and send more values. Other machine learning based methods have been proposed in the literature for data prediction for data reduction. A lot of approaches were interested in data correlation for this purpose, mainly using the Pearson correlation technique and its derivatives [4], [10], [2], the Auto Regression model [16], [5] and the convolutional long short-term memory (LSTM) networks techniques [17], [1]. The authors in [4] proposed a methodology to analyse data streams, based on spatio-temporal correlations using Pearson correlation. The data is collected via Bluetooth sensors that measure the number of cars in a road, attached to light poles. Nevertheless, the nodes in this method were not energy limited, they were attached to light poles for power and energy reasons. In our approach, the nodes might be in remote areas in a smart agriculture field and the algorithms are executed at the sensor node level for quicker decision making. The authors in [17] proposed a Convolutional LSTM network to enhance the forecasting of the precipitation nowcasting. This approach builds an end-to-end trainable model to solve the issue capturing spatio-temporal correlations by stacking multiple ConvLSTM layers and forming an encoding-forecasting structure. In our approach, we are interested to execute algorithms directly at the sensor node level, however, this type of convolutional layers for training would be time and energy consuming on energy-limited sensor nodes. Some papers investigated data correlation and data similarity between several features as in [11], [12] and [3]. Authors in [11] present a Bayesian Inference Approach to detect data with high spatio-temporal correlation, to avoid transmitting data that can be reconstructed from another data such as temperature and humidity in some cases. They used some complex equations to compute the

correlation and to predict the other values. In this paper, our solution for data reduction at the sensor node is to implement a light data reduction algorithm based on data correlation at the sensor node level to reduce the amount of sent data to the sink. It is based on data correlation using Pearson correlation for the inter-nodes (spatial) and the intra-node data correlations. The data prediction at the sink is done using simple arithmetic equations as explained in Section III.

III. DATA CORRELATION

In our paper, the contribution consists of using the Pearson correlation method to reduce the amount of redundant data sent from the nodes to the sink while maintaining the needed accuracy of the sensed data. We will run two data correlation based mechanisms:

1) Inter-nodes data correlation;
2) Intra-node data correlation.

In the inter-nodes data correlation, Pearson correlation coefficient [2] is used to compute the percentage of data similarity between a same type of data (e.g the similarity between the temperature) sensed by different neighbour nodes. If this similarity is considered high, the values offer redundant information. As a data reduction process, only one of the sensor nodes sends the value to the sink while maintaining the information depending on one or multiple criteria as explained in the next section.

In the intra-node data correlation technique, data correlation is evaluated on the same node but between different types of data. The purpose is to try to extract one parameter value from another one by using effective and simple arithmetic operations taking account of the energy limitations on the sensor node level which prevents us from using of an auto-regression technique like ARMA [16] or LSTM technique [17].

We detail each of these techniques below.

A. Inter-Nodes Data Correlation

In this part, the correlation coefficient between the same parameter from different nodes is computed using the Pearson correlation coefficient method. However computing the Pearson correlation coefficient requires several values of the same data parameters from different sensor nodes (e.g temperature, wind speed, humidity, etc...). Several ways exist to compute this coefficient. The normal and most known formula is presented in the equation below:

$$\rho_{X_1 X_2} = \frac{\sum_{i=1}^{n}(X_{1_i} - \overline{X_1})(X_{2_i} - \overline{X_2})}{\sqrt{\sum_{i=1}^{n}(X_{1_i} - \overline{X_1})^2}\sqrt{\sum_{i=1}^{n}(X_{2_i} - \overline{X_2})^2}} \quad (1)$$

In our work, we used the geometric interpretation of the Pearson correlation. This technique needs several values to detect the percentage of correlation between two parameters of the same type on two different sensor nodes. This method does not take account of the case when the parameters under consideration are proportional or inversely proportional. For this purpose, Equation 2 is multiplied by a sign parameter s

to have the right positive and negative values, it takes two values 1 or −1. As an example we consider two vectors of five values each X_1 and X_2:

X_1=(2, 3, 6, 7, 9) ; X_2= (0.12, 0.13, 0.16, 0.17, 0.19) The correlation coefficient is computed as follows:

$$\rho_{X_1 X_2} = \cos\theta = \frac{\overrightarrow{X_1}.\overrightarrow{X_2}}{\|X_1\| .\|X_2\|} \times s \qquad (2)$$

where $\rho_{X_1 X_2}$ represents the Pearson correlation coefficient. In this example,
$\rho_{X_1 X_2}$=0.99, however the percentage of correlation is the square of the correlation coefficient times 100 as shown in the equation below:

$$\rho_p = \rho_{X_1 X_2}^2 \times 100 \qquad (3)$$

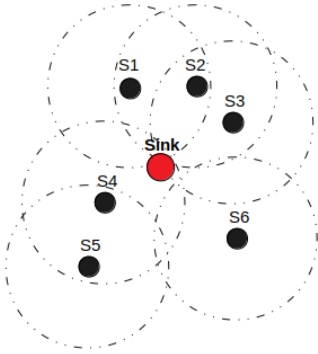In this case the two vectors are 99.8% correlated.

Fig. 1. Network Scenario

This method is applied on every sensor node to compute the correlation coefficient with all its neighbour nodes in its communication range. We assume that the information needed to compute the correlation coefficient together with the remaining energy of each node is piggybacked in the Hello messages used for each node to discover each other. A threshold of correlation is predefined on each node for each type of data. The whole process is executed at the very start of the network. The mean ratio of difference R_m is computed between the two parameters after n consecutive values as written below:

$$R_m = \frac{\sum_{i=0}^{n} \frac{y_i}{x_i}}{n} \qquad (4)$$

Then, the sink computes the "missing" value of y_{n+1} based on the values of ρ, x_{n+1}, x_n, y_n, R_m and the sign value s as follows:

$$y_{n+1} = y_n + (x_{n+1} - x_n) \times R_m \times \rho^2 \times s \qquad (5)$$

Once two nodes detect a high correlation between one or several parameters, they locally decide whether to send or not the message based on the residual energy of each other. Indeed, only the one with the highest energy sends the message. In case of ties, the message will be sent by the node with the smallest identifier. This does not require any additional exchange between nodes, the needed information (Id, battery level) being included in Hello messages.

Let's consider Fig 1 to illustrate our approach. S2 and S3 are neighbour nodes (they can directly communicate with each other) and can apply the Pearson correlation coefficient equation to detect the percentage of correlation between their parameters. Let's assume the correlation coefficient is high, (high correlation), the sensor-node that has more residual energy between S2 and S3 will send the data to the sink. This same sensor sends the ratio R_m at the start. The sink computes the value of the other parameter based on R_m and the real values from the sending sensor node (one real value and one predicted value).

B. Intra-Node Data Correlation

Data correlation can be computed with different types of data on the same node. For example the correlation between humidity and temperature can be high and follows a certain shape, in this case we can extract one value from the other one. This method is applicable on every sensor node trying to find correlations between several types of parameters such as temperature, humidity or wind speed. If any important correlation is found, it will help reduce the amount of data sent from the node to the sink by sending only one of the two correlated parameters. The sink can extract the second parameter value based on the received one using some formulas as explained below.

The Pearson Correlation Coefficient is also applied in this kind of data reduction process. It is used to detect if the correlation between two different parameters is sufficient to start the prediction. The correlation coefficient ρ_{xy} between two different parameters is compared to a predefined threshold of correlation th_{cor}. Two parameters are considered correlated if the equation below is true:

$$\rho_{xy} > th_{cor} \qquad (6)$$

where x and y are two vectors of several values representing two different parameters such as the temperature and the humidity. We compute the Pearson correlation coefficient as mentioned in Equation 2. If Equation 6 holds, we compute the mean of the ratio R_m for the same number of values n that was needed to compute the correlation coefficient. We apply the equations 4 and 5 to compute R_m and the next value y_{n+1} based on the values of ρ, x_{n+1}, x_n, y_n and R_m.

Meanwhile the sensor nodes are always sensing the real values of all the parameters. The predicted value must be in a certain range based on the real value to be accepted and to continue the prediction process. This range is defined by and upper and lower thresholds based on the correlation coefficient as follows:

$$th_{up} = \left(\frac{1-\rho_{xy}}{2} + 1\right) \times y_r \qquad (7)$$

$$th_{low} = \left(\frac{1-\rho_{xy}}{2} + \rho_{xy}\right) \times y_r \qquad (8)$$

where R_y is the real value of parameter y. In this case, if $\rho_{xy} = 0.8$, it is considered at 0.2 far from the perfect correlation 1. The upper threshold $th_{up} = 1.1 \times y_r$ and $th_{low} = 0.9 \times y_r$

The node must be able to make a decision by comparing the real sensed value with the predicted one. For this purpose, the dual prediction model is adopted from [15]. In this scenario, the node and the sink predict the new values at the same time by applying the same equations. If the predicted value falls outside this range, the node sends the real value to the sink and the prediction process continues based on the new real value. However, the correlation coefficient is not recomputed unless we have 2 consecutive values falling outside of the range.

IV. SCENARIO AND ALGORITHM

In this section, we discuss the scenario where the Pearson Data Correlation and Prediction algorithm PDCP is implemented. Fig 1 shows several neighboring sensor-nodes where different neighbor couples can be formed trying to find correlation between their parameters (inter-nodes). This is done while implementing the intra-node correlation on each node to detect the degree of correlation between different parameters on each node. The PDCP algorithm is detailed in 1. The intra-node correlation is always applicable on each sensor-node, however, the inter-nodes correlation depends on the neighboring parameters (distance and radius of communication). E.g, in Fig 1, the whole algorithm of inter-nodes and intra-node correlation techniques can be applicable on several neighbor couples such as $\{S1, S2\}.\{S2, S3\}$ and $\{S4, S5\}$.

In Algorithm 1, the number of values needed to compute the correlation depends on the scenario and the studied application. Later on, in our experiments, a full day of values is needed to compute the correlation (48 values are captured in a day by a sensor node for every parameter). In this scenario and algorithm, for the intra-node correlation, we focused on the temperature and humidity correlation specifically which is represented by ρ_{hute} in the algorithm. For the inter-nodes correlation, we computed the correlation for the temperature in Sensors S1 and S2.

Algorithm 1 Pearson Data Correlation and Prediction Algorithm PDCP

    Set $th_{cor_{inter}}, th_{cor_{intra}}, th_{up}, th_{low}, n, i = 1, R_m, \rho_{te}, \rho_{hute},$ RN
2: **for** each sensor-node S_0 **do**
      Compute $\rho_{te_{S_0 S_i}}$ for each neighbor S_i
4:     **if** $\rho_{te} > th_{cor_{inter}}$ **then**
        Check the residual energy
6:       **if** $RN_{S_0} > RN_{S_i}$ **then**
          S_i Stops sending the temperature Values to the sink
8:       **end if**
    end if
10:    Compute ρ_{hute}
    if $\rho_{hute} > th_{cor_{intra}}$ **then**
12:     Compute th_{up}, th_{low} and R_m # Equations 4,7,8
      Compute the next humidity value   # Equation 5
14:    **end if**
   end for

In the inter-nodes correlation, if the temperature parameters of S1 and S2 are correlated, one of the two sensors will not send its temperature values anymore and the sink considers the sent one as the data for both (one real and one predicted). As mentioned before, the residual energy is the parameter taken into account between two neighbour nodes to decide which node will send the data. In the intra-node process, if the humidity is correlated to the temperature, each node will send one of the two values (e.g, temperature) to the sink with all the needed parameters (R_m, ρ,..) to predict the other value. However, the two levels of correlation can be related, while taking into account the predicted temperature from the inter-nodes correlation in the intra-node level. In this case the humidity value can be predicted through the predicted value of the temperature which will reduce furthermore the amount of sent data to the sink.

V. EXPERIMENTAL RESULTS

In this section, we compare PDCP algorithm to a Bayesian approach in [11]. This comparison is based on the prediction accuracy and the amount of sent data. We used a MATLAB simulator with a meteorological dataset for the 8th, 9th and the 10th of April 2020 from two sensor nodes deployed in Lille city, France (Lille airport and Lille city centre) from Weather_Underground website which gathers data from a sensor network of different weather stations deployed around the globe[1]. In the remaining of the simulations we consider S1 as the airport node and S2 as the city centre node. For the inter-nodes correlation the temperature parameter was taken into account as the studied parameter between both sensors. For the intra-node correlation the temperature and humidity parameters in each sensor were selected to evaluate their correlation and test our approach.

The sampling rate of the sensor node is set to 1 value each 30 minutes (by default).

After conducting several simulations, the number of needed values to compute the correlation coefficient between two parameters is set to 48 values gathered in a whole day. The 8th of April is used for learning the values and thresholds. These parameters are used in the testing phase in the next two days.

The temperature in those 3 days varied from 9 degrees Celsius as a minimum to 26 degrees Celsius as a maximum. The humidity varied from 30% to 100%.

A. Inter-Nodes

In this part, for the inter-nodes correlation, the temperature parameters from the two neighbour sensor nodes S1 and S2 is studied.

The threshold of correlation th_{cor} is set to 0.9, since in this part we need a very high correlation to decide to stop sending one of the two values.

---

[1]https://www.wunderground.com

In those 3 days, 144 values from each parameter are sensed at each sensor-node. The main purpose is to reduce the amount of sent data to the sink.

The simulations show that the correlation coefficient for the whole first day (April 8 2020) is equal to 0.91, greater than the predefined threshold of correlation (0.9) for the inter-nodes correlation, the mean ratio R_m is equal to 1.09.
This high correlation coefficient leads to send only one of the two temperature values to the sink (Value of S2) alongside the R_m value at the very start of the network. The sink computes the other value (Value of S1) as shown in equations 4 and 5. Fig 2 shows the difference between the real and the predicted values for the next 2 days (April 9 and 10). This difference does not surpass 1 degree Celsius.
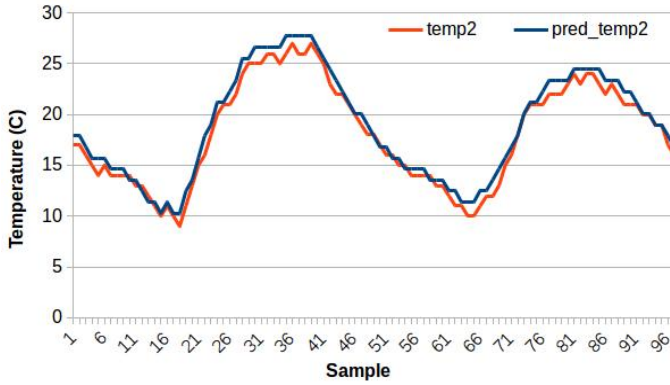

Fig. 2.   Temperature prediction on the sink

While applying the inter-nodes correlation part of PDCP algorithm, the number of sent values is decreased from 192 values for the two nodes in two days to 96 values as shown in Table I (S1 is not sending the temperature values).

TABLE I
AMOUNT OF TRANSMITTED TEMPERATURE VALUES PER DAY BY S1 AND S2

Day	All data	Inter-Nodes
9th	96	48
10th	96	48
Total	192	96
Data Reduction	0%	50%

B. Intra-Node

Different parameters are sensed by each sensor node. After conducting several simulations, we decided to study the correlation between the temperature and the humidity parameters on S1 (Airport Node) while applying the intra-node technique in the PDCP algorithm based on the Pearson Correlation coefficient method.

The threshold of correlation th_{cor} for the intra-node part is equal to 0.75, since we are comparing different types of parameters.

On April 8, 2020 the correlation coefficient was equal to -0.8, which is greater than 0.75, so the prediction process

can take place based on the last value and the computed mean ratio R_m. In this case, $R_m = 4$. The humidity values are then predicted through the temperature values as explained above.
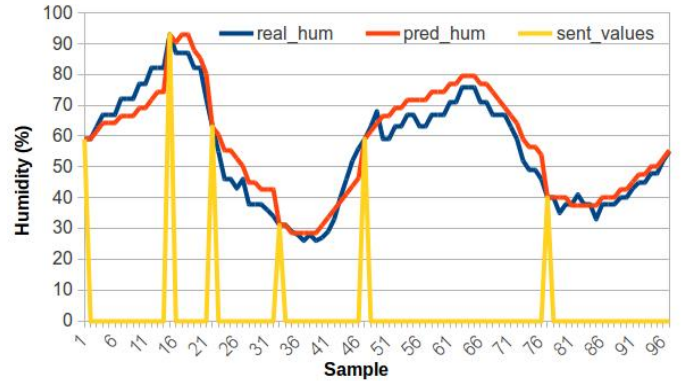

Fig. 3.   Humidity prediction at the sink while applying the intra-node technique

Fig 3 shows the prediction of the humidity for the same sensor node on the sink for April 9 and 10, 2020. In those 2 days, 96 humidity values were captured, however, the node sent 6 humidity values to the sink (surpassing the thresholds) to enhance the prediction at the sink. Their were no consecutive sent values to recompute the correlation coefficient ρ and the mean ratio R_m. The difference between the real and the predicted values did not surpass 7% of humidity which affirms the integrity and the feasibility of the predictions in our approach.

TABLE II
AMOUNT OF TRANSMITTED HUMIDITY VALUES PER DAY BY S1

Day	All data	Intra-Node
9th	48	4
10th	48	2
Total	96	0
Data Reduction	0%	93%

Tables I and II show that the intra-node part in PDCP algorithm when applied reduces the amount of sent data to the sink while maintaining the integrity and the accuracy of the data as shown in the figures above. A comparison with another method is drawn in the section below.

C. Intra-Node and Inter-Nodes Combination

In the inter-nodes part, the temperature value of S1 is predicted on the sink. In the intra-node part, the humidity of S1 is predicted based on the temperature on the same node. However, combining both parts of PDCP algorithm increases the data reduction. This combination helps to predict at the sink the humidity values for S1 based on the already predicted temperature values for the same node.

Fig 4 shows the humidity prediction for S1, 7 values were sent (the threshold was surpassed). The maximum difference between the real humidity value and the prediction value did

not exceed 7% of humidity, thus staying in the same humidity category.

While applying the intra-node correlation on S2, the node sent only 7 humidity values to the sink in two days of predictions as shown in Table III.
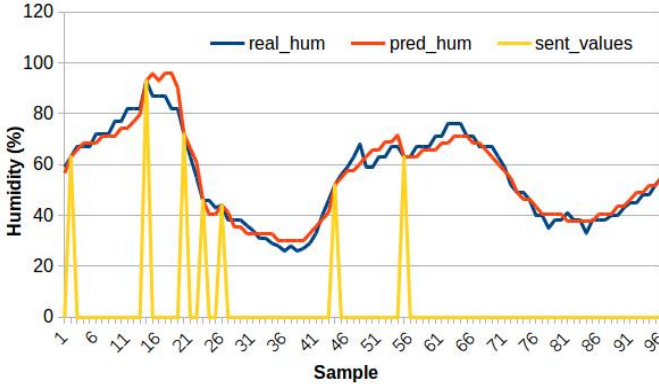

Fig. 4. Humidity prediction at the sink while applying PDCP algorithm

Table III draws the differences between our approach and a Bayesian inference approach from the literature [11] for the same scenario and parameters.

As noticed from the numbers, they are neglecting any change in the humidity values which helps them to improve data reduction to 50% for the intra-node correlation but they lost some accuracy with a humidity standard deviation H_{SD} up to 10%. However, the inter-nodes correlation applied in PDCP gives us the edge to improve the percentage of data reduction to reach 70% with a better accuracy and a standard deviation H_{SD} of 7%.

TABLE III
AMOUNT OF TRANSMITTED VALUES PER DAY BY S1 AND S2

Day	All data	PDCP	Bayesian [11]
$S1_T$	96	0	96
$S1_H$	96	7	0
$S2_T$	96	96	96
$S2_H$	96	5	0
Total	384	108	192
H_{SD}	0%	7%	10%
Data Reduction	0%	70%	50%

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed a data reduction technique based on a data correlation technique by applying the Pearson correlation coefficient functions and equations in WSN implemented for agriculture to detect any abnormal situation in the meteorological data which can harm the agriculture.

Our simulations show a reduction of more than 70% of the overall data which surpasses other approaches from the literature by more than 20%.

In the near future, this approach can be enhanced by applying correlation on more than two nodes at the same time. Other ideas can be taken into account such as studying the

routing protocol for multi-hop scenarios. To further increase the data reduction sampling rate adaptation is to be considered.

REFERENCES

[1] J. Abdullah, M. K. Hussien, N. A. M. Alduais, M. I. Husni, and A. Jamil. Data reduction algorithms based on computational intelligence for wireless sensor networks applications. In *2019 IEEE 9th Symposium on Computer Applications Industrial Electronics (ISCAIE)*, pages 166–171, April 2019.

[2] F. R. Almeida, A. Brayner, J. J. P. C. Rodrigues, and J. E. B. Maia. Improving multidimensional wireless sensor network lifetime using pearson correlation and fractal clustering. *Sensors*, 17(6), 2017.

[3] M. Azaza, C. Tanougast, E. Fabrizio, and A. Mami. Smart greenhouse fuzzy logic based control system enhanced with wireless data monitoring. *ISA Transactions*, 61:297 – 307, 2016.

[4] M. Bermudez-Edo, P. Barnaghi, and K. Moessner. Analysing real world data streams with spatio-temporal correlations: Entropy vs. pearson correlation. *Automation in Construction*, 88:87 – 100, 2018.

[5] F. Ding, D. Meng, J. Dai, Q. Li, A. Alsaedi, and T. Hayat. Least squares based iterative parameter estimation algorithm for stochastic dynamical systems with arma noise using the model equivalence. *International Journal of Control, Automation and Systems*, 16(2):630–639, 2018.

[6] S. E. Díaz, J. C. Pérez, A. C. Mateos, M.-C. Marinescu, and B. B. Guerra. A novel methodology for the monitoring of the agricultural production process based on wireless sensor networks. *Computers and Electronics in Agriculture*, 76(2):252 – 265, 2011.

[7] C. Habib, A. Makhoul, R. Darazi, and C. Salim. Self-adaptive data collection and fusion for health monitoring based on body sensor networks. *IEEE Transactions on Industrial Informatics*, 12(6):2342–2352, Dec 2016.

[8] K. P. Musaazi, T. Bulega, and S. M. Lubega. Energy efficient data caching in wireless sensor networks: A case of precision agriculture. In A. Nungu, B. Pehrson, and J. Sansa-Otim, editors, *e-Infrastructure and e-Services for Developing Countries*, 2015.

[9] T. Ojha, S. Misra, and N. S. Raghuwanshi. Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges. *Computers and Electronics in Agriculture*, 118:66 – 84, 2015.

[10] G. Rajesh and A. Chaturvedi. Correlation analysis and statistical characterization of heterogeneous sensor data in environmental sensor networks. *Computer Networks*, 164:106902, 2019.

[11] C. Razafimandimby, V. Loscri, A. Maria Vegni, D. Aourir, and A. Neri. A Bayesian approach for an efficient data reduction in IoT. In *Int. Conf. on Interoperability in IoT (InterIoT)*, Nov. 2017.

[12] C. Razafimandimby, V. Loscri, A. M. Vegni, and A. Neri. Efficient bayesian communication approach for smart agriculture applications. In *IEEE Vehicular Technology Conf. (VTC-Fall)*, Sep. 2017.

[13] C. Salim, A. Makhoul, R. Darazi, and R. Couturier. Similarity based image selection with frame rate adaptation and local event detection in wireless video sensor networks. *Multimedia Tools and Applications*, 78(5):5941–5967, Mar 2019.

[14] C. Salim and N. Mitton. K-predictions based data reduction approach in wsn for smart agriculture. *Computing*, pages 1–24, 2020.

[15] C. Salim and N. Mitton. Machine learning based data reduction in wsn for smart agriculture. In *Advanced Information Networking and Applications*, pages 127–138, Cham, 2020. Springer International Publishing.

[16] Y. Shen, J. Guo, X. Liu, Q. Kong, L. Guo, and W. Li. Long-term prediction of polar motion using a combined ssa and arma model. *Journal of Geodesy*, 92(3):333–343, 2018.

[17] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 802–810, Cambridge, MA, USA, 2015. MIT Press.

[18] G. B. Tayeh, A. Makhoul, D. Laiymani, and J. Demerjian. A distributed real-time data prediction and adaptive sensing approach for wireless sensor networks. *Pervasive and Mobile Computing*, 49:62 – 75, 2018.