A Distributed Real-Time Data Prediction and Adaptive Sensing Approach for Wireless Sensor Networks

Gaby Bou Tayeh[a], Abdallah Makhoul[a], David Laiymani[a], Jacques Demerjian[b]

[a] *Femto-St Institute, UMR 6174 CNRS, Université de Bourgogne Franche-Comté, France*
[b] *LARIFA-EDST, Faculty of Sciences, Lebanese University, Fanar, Lebanon*

Abstract

Many approaches have been proposed in the literature to reduce energy consumption in Wireless Sensor Networks (WSNs). Influenced by the fact that radio communication and sensing are considered to be the most energy consuming activities in such networks. Most of these approaches focused on either reducing the number of collected data using adaptive sampling techniques or on reducing the number of data transmitted over the network using prediction models. In this article, we propose a novel prediction-based data reduction method. furthermore, we combine it with an adaptive sampling rate technique, allowing us to significantly decrease energy consumption and extend the whole network lifetime. To validate our work, we tested our approach on real sensor data collected at our offices. The final results were promising and confirmed our theoretical claims.

Keywords: Wireless Sensor Networks; Data estimation; Data reduction; Data prediction; Adaptive Sampling; Energy saving.

1. Introduction

A Wireless Sensor Network (WSN) consists of a base station (Sink) and a number of small, wireless electronic devices called sensor nodes. These nodes react to inputs from both the physical or environmental conditions of a monitored area, such as pressure, temperature, humidity, motion, light, etc [1], and they cooperatively pass data through the network to the Sink for further processing. In recent years, efficient design of Wireless Sensor Networks has become a leading area of research for scientists. Several challenges needed to be addressed for a realistic implementation of WSNs, such as localization, deployment, coverage, data integrity, reliability, etc. All of these challenges are subject to many constraints that sometimes might be incompatible. However, the most important constraint is the limitation in energy resources.

Email addresses: gaby.bou_tayeh@edu.univ-fcomte.fr (Gaby Bou Tayeh), abdallah.makhoul@univ-fcomte.fr (Abdallah Makhoul), david.laiymani@univ-fcomte.fr (David Laiymani), jacques.demerjian@ul.edu.lb (Jacques Demerjian)

Generally, wireless sensors are not connected to an electrical circuit that feeds them power. Instead, due to design constraints they rely on small, low powered batteries. Therefore, the optimal use of energy is the key solution for other challenges and to an extended operational lifetime of these networks.

In WSN, sensing and radio communication are considered to be the most expensive activities in term of energy consumption. Therefore, many approaches have been proposed in the literature to reduce the number of sampled data and the transmitted ones.

Due to the nature of WSNs, sensor data tend to change smoothly over time and it contains a significant chunk of redundant information. Therefore, to reduce the number of sampled data some researchers proposed several adaptive sampling techniques [2, 3, 4, 5, 6, 7] that dynamically increase or decrease the sampling rate of a sensor according to the level of variance between collected data over a certain period of time. This approach prevents the sensor from collecting redundant information. Thus, the sensing activity is reduced which in turn leads to a reduction in the transmission activity. Hence, less energy is consumed.

Instead of relying on adapted sampling rate to produce less transmissions, other approaches preferred to tackle the problem differently through a direct reduction of radio communication, since it consumes more energy than sensing. One of the most commonly used technique to limit data transmission is the dual prediction mechanism [8, 9, 10, 11, 12]. An identical prediction model is shared between each node and the Sink. This model is used to forecast future values. Thus, instead of transmitting all the collected data, a sensor transmits only the measurements that deviate from the predicted value by a threshold predefined by the user. Therefore, if the Sink does not receive any measurement at a given time, it acknowledges that the model's prediction is within the error budget.

Merging both adaptive sampling and dual prediction based transmission reduction into a single mechanism, can reduce energy consumption significantly compared with the approaches relying on either one of them.

In this paper we present an approach combining an adaptive sampling and a novel transmission reduction technique into a single energy efficient algorithm. Thus, our main contributions in this work can be summarized as follows:

- Proposing a new data transmission reduction algorithm that reduces the amount of data reported to the sink using a dual prediction model. In contrast to other similar techniques our model is light in term of computational cost and requires a very small memory footprint, yet it is robust and efficient.

- Coupling our novel transmission reduction algorithm with an adaptive sampling technique. Enabling the sensor to collect fewer measurements which in turn increase the efficiency of the transmission reduction algorithm and reduces the amount of energy consumed by the sensing activity.

- Conducting experiments on real sensor data and comparing our proposed method with a recent data reduction approach [13] based on a combination of a prediction model and an adaptive sampling technique. The final results demonstrates that our proposal outperforms the latter in reducing the overall energy consumption.

The rest of this paper is organized as follows. In section 2 the work related to data transmission reduction, adaptive sampling and approximate replication of sensor data is briefly presented. In section 3 the Kruskal-Wallis based algorithm that allows the sensor to adapt its sampling rate is explained. In section 4 our novel dual prediction based transmission reduction method is introduced. Section 5 explains how the adaptive sampling and transmission reduction techniques can be merged together. The obtained experimental results are shown in section 6. Finally, the paper is concluded in section 7.

2. Related Work

The elements of energy dissipation in a wireless sensor node that monitors a specific environment and report the collected information to a central workstation are very different [14, 15]. However, in the monitoring stage, the micro-controller, sensor and radio components dominate the energy consumption and they are the main optimization targets.

In previous studies, authors studied several energy-saving schemes for wireless monitoring operations such as: data aggregation [16, 17], data compression [18, 19], adaptive sampling [2, 3, 4, 5, 6, 7, 20] and data prediction [8, 9, 10, 11, 12, 13].

2.1. Compression and Aggregation

The authors in [16], present a data reduction method called the Prefix-Frequency Filtering (PFF). In this approach, the first layer of data reduction is done locally on each sensor node, and a second layer of data reduction is done on a central node collecting data from neighboring nodes which is also referred to as “Aggregator”. On the latter PFF uses Jaccard similarity to measure the correlation among reported measures from different nodes in order to merge and send them to the Sink. In [17] the Dynamical Message List Based Data Aggregation (DMLDA) technique is presented. This method is based on the data clustering technique and it provides a real time data aggregation. The backbone of this method is a special data structure named dynamical list. This list is deployed in every filtering node to store history messages before transmission. Thus, instead of using period delays, older messages are used to filter duplicated ones.

In [18] the authors uses raw signal processing and signal reconstruction to develop a reordering algorithm that resorts the sensor nodes at the Sink. This method enhances the sparsity of the signal by reducing the number of measurements needed for its reconstruction, consequently resulting in a low compression sampling rate that in turn scale down irrelevant communication traffic. The authors in [19] proposed a cluster-based quality-aware adaptive data compression scheme, which takes into consideration the applications data quality and it also limits information loss by using adaptive clustering and novel coding algorithm.

2.2. Adaptive Sampling

In [3] the amount of data is reduced by adapting the sampling rate for air pollution monitoring. The suggested work used the Kalman filter to remove noise from the sensed data. The algorithm adapts the period sampling based on the similarity between the current and the previous sensed data.

In [4] the sampling rate of the sensor node is adapted by taking into consideration both the system and the application context levels. For instance, the availability of the energy for harvesting represents the system context. This availability is the criteria used to set the maximum sampling rate for the node. The application context is represented by the user request, where feedback from the system executing specific rules of user or field scientist is used to set the rates of sensor node sampling in optimal way.

In [20] the authors propose three different data collection and adaptive sampling techniques for Industrial Process Monitoring. The first one uses the ANOVA model, while the second one is based on sets similarity functions, and the third one on the distance functions.

Adaptive sampling techniques are remarkably efficient when the temporal correlation among collected data is high and the irregularity and sudden changes are low. In opposite conditions, where correlation is low and irregularity is high, these techniques perform poorly since the sampling rate will be kept at maximum most of the time. Thus, the computing cost will overcome the achieved reduction in sampling and transmission cost.

2.3. Prediction-based Data Reduction

In [12] the authors proposed a Derivative Based Prediction model (DBP) that is computed based on a learning window, containing m data points. The model is linear, computed as the slope d of the segment connecting the average values over the first and last l edge points at the beginning and end of the learning window.

In [13] the authors proposed a technique named Dual Prediction with Cubic adaptive sampling (DPCAS) that combines an exponential time series predictive model with a TCP CUBIC congestion adaptive sampling technique. Enabling the sensor node to reduce its sampling rate based on the produced prediction error. Moreover, measurements are transmitted to the sink only when a significant change in readings occur. The whole data set is then reproduced on the sink by interpolating the received measurements.

The authors in [8] proposed a prediction model that is based on the Kalman filter. The same instance of this model is built by the Sink using historical data reported by the sensor and then is shared with the latter. The same model on both ends simultaneously performs linear predictions for future readings, enabling the sensor to transmit a measurement to the Sink only when the prediction is not accurate.

The dual Kalman filter method requires a priory knowledge and statistical data on the environment being monitored, in order to build the model. Therefore, the authors in [9] proposed a dual prediction mechanism that is based on Least Mean Square (LMS) adaptive filter. LMS lends itself to be compact, light and requires no priory statistical knowledge of the data. Thus, it makes the prediction model more stable and adaptable with changes.

The authors in [10] proposed to combine the LMS and Recursive Least Squares (RLS) adaptive filters in a single prediction model. Since the latter is able to achieve faster convergence and produces a prediction model that is more stable, RLS is used to build the prediction model. Once this model is built, the parameters are then passed to an LMS adaptive filter to perform predictions. The reason for this switch is that LMS has lower complexity, thus it suits the energy constraints of the sensor better than RLS.

In [11] the authors proposed a dual prediction model that is based on the Hierarchical Least Mean Squares (HLMS) adaptive filter. The HLMS is a multi-level LMS filter, that makes a trade-off between increasing the complexity of LMS filter and having a better prediction filter.

The speed and the success of convergence of the adaptive filters is conditioned by some predefined parameters such as the “step size”. A small alteration in these parameters can heavily affect the performance of these filters, and choosing an optimal value is not feasible most of the time, since it requires a training phase.

We propose in this paper to combine our adaptive sampling technique [5] with our novel dual prediction based forecasting model that is free from any parameters limiting its performance and which only requires two measurements to be built and one measurement to be updated. Targeting at the same time the two most energy consuming activities in WSNs, in order to preserve as much energy as possible.

In the following sections, adaptive sampling and data transmission reduction techniques are explained. Then an algorithm merging these two approaches together is presented.

3. Adaptive Sampling

First of all let us begin by explaining the Kruskal-Wallis statistic model. It forms the skeleton of the adaptive sampling algorithm aiming to reduce the number of data sampled by each sensor.

3.1. The Kruskal-Wallis Statistic Model

The Kruskal-Wallis test [21] takes as input a group of data sets to identify whether there is a difference between at least two of these sets. To understand how this test works and how it could help us to reduce the sampling rate of a sensor, we give the following illustrative example that explains its functionality and applicability in WSNs.

3.1.1. Illustrative example

Let us consider that a sensor operates in rounds, where each round consists of p periods. To simplify the example, let us assume that p is equal to two. Table 1 shows a set of measurements collected by a sensor during two consecutive periods.

The first step is to order the measurements in both periods by increasing order of their values and assign a rank denoted r to each one of them, representing its position in the ordered list. However, two or more measurements could have the same value. In this case the mean value of their ranks is calculated and assigned to each one of them. For instance, in Table 1 the value 7.0 is repeated twice, both in period 1 and 2 with ranks 5 and 6 respectively. The mean value of both ranks is 5.5. Thus, the ranks of both measurements holding the value 7.0 are replaced by 5.5.

The second step is to pass the ranked measurements as input to the Kruskal-Wallis test in order to find which one of the following assumptions is correct:

Table 1: Example of collected measures

Raw Measures		Measures Rankings	
Period 1	Period 2	Period 1	Period 2
3.4	4.6	1	2
6.2	5.8	4	3
7.0	7.0	~~5~~ 5.5	~~6~~ 5.5
7.3	7.5	7	8
7.6	8.0	9	10
10.3	10.2	~~12~~ 12.5	11
	10.3		~~13~~ 12.5
Number of Measures		Sum of Rankings	
6	7	39	52

- Assumption 1: the two groups of data (measurements in period 1 and 2) are significantly different.
- Assumption 2: the difference between the two groups of data is not significant.

The test is conducted by calculating the following formula :

$$H = \frac{12}{N \times (N+1)} \sum_{i=1}^{p} \frac{r_i^2}{n_i} - 3 \times (N+1) \tag{1}$$

where:

- N is the total number of measurements in all periods.
- n_i is the number of measurements inside the i^{th} period.
- r_i is the sum of all ranks in the i^{th} period.

Using the data in Table 1 and based on equation (1), H is calculated as follows:

$$H = \frac{12}{13 \times (13+1)} (\frac{39^2}{6} + \frac{52^2}{7}) - 3 \times (13+1) = 0.183$$

Finally, to check which assumption is the correct one, the result of this formula is compared with a “difference value” denoted H_t. H_t varies according to the false rejection probability predefined by the user, denoted α. The relation between α and H_t can be found in the $chi-square$ Table. The risk α is defined in a statistical test as the risk of rejecting the Null hypothesis when in fact it is true, it is also known as Type I error. This risk is stated

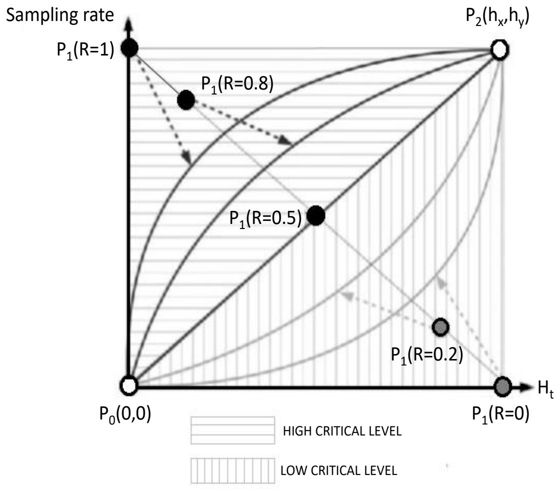

Figure 1: Sampling rate adaptation using the Behavior function.

in terms of probability (such as 0.05 or 5%). It corresponds to the confidence level of a statistical test, so a level of significance $\alpha = 0.05$ corresponds to a 95% confidence level. In our approach it is the probability that a sensor node find a high variance between its collected data while in reality there is no variances and it should adapt the sampling rate. Therefore, when α decreases the value of H_t increases and then the condition $H < H_t$ becomes more difficult to be satisfied. Consequently, the sampling rate increases when α decreases.

Let us assume $\alpha = 0.05$, for this value of α, H_t is equal to 5.991. Comparing the results of the previous equation we notice that $H < H_t$ ($0.183 < 5.991$). Therefore, the first assumption is accepted. Hence, the sampling rate must be adapted.

3.2. The Behavior curve function

Based on the Kruskal-Wallis test, when a node notices high variance differences, it increases its sampling rate in order to prevent missing important measurements and decreases its sampling rate when the variance is less than the threshold H_t. Following the example above low variance was detected, since $H < H_t$.

To compute the sampling rate of the sensor a behavior function (BV) is used, taking as input the risk of the application denoted R, or in other words, how important the quality of data is to the end user. This BV function is expressed by a Bezier curve that passes through three points as shown in Figure 1: (0,0), (H_t, Maximum Sampling Rate), and R.

4. Transmission reduction

To transmit the collected data to the base station, the network involves a large number of radio operations, including listening to the channels, receiving and transmitting data. Since the main part of energy dissipation in wireless sensors is indeed the radio components, the lifetime of the network can be extended significantly if we optimize the radio communication operations.

The data in WSN are defined as time series, since they are sequential data points collected over successive time. Such data is generally affected by three main factors: Trend, Cyclical and Irregular factors. As brief description of these factors, time series in general tend to increase, decrease or stand still over a long period of time, this tendency is usually called "a Trend". For example, the temperature of a room tend to have an increasing trend, from when the sun rises until it starts to set. Cyclic variations can be seen during a longer period of monitoring. For instance, the same cycle of increase and decrease in the temperature takes place every day. Finally, irregular factors are variations caused by unpredictable influences, such as clouds covering the skies during the day and suddenly reducing the temperature or dimming the light.

Therefore, since time series data follows a specific trend and tend to change smoothly over time, one can forecast future measurements by observing past patterns. In the light of this, we propose a dual prediction based reporting mechanism to conserve energy and maximize the lifetime of the network. Let us first begin by explaining the concept of the dual-prediction mechanism.

4.1. Dual-Prediction Mechanism (DPM)

The Dual-Prediction Mechanism is a model that analyzes the history of previously collected information, extract the moving trend of data in order to approximately estimate future readings. In the DPM, the same prediction model is deployed at both the sensor nodes and the base station. Using the same historical data, sensor nodes and the base station regularly make the same prediction of any future observation. This technique allows the sensor nodes to avoid transmitting its sensed data to the base station, as long as the predictions match the readings.

Meanwhile, the base station always presume that its prediction reflects the real observation, unless it receives the corrections from the sensor node (since the sensor can compare the prediction with the real sensed measurement). Figure 2. illustrates in a simplified way how this mechanism work.

4.2. Transmission reduction method based on DPM

As mentioned previously, the prediction model is built using historical measurements collected by the concerned sensor. The number of these measurements depends on the type of the chosen prediction model. Different types of prediction models has been proposed in the literature [8, 9, 10, 11, 12], where the number of data needed to train the model can vary from tens to hundred even thousands of reading depending on its learning capacity. In our method, only two measurements are sufficient to build the prediction model, and a single measurement to correct it.

At the beginning, the sensor collects and sends to the Sink the two first measurements $x[0]$ and $x[1]$ at time t_0 and t_1. Each time the sink receives a new measurements it stores in its memory the value of this measurement and the time when it was received. Let us denote the

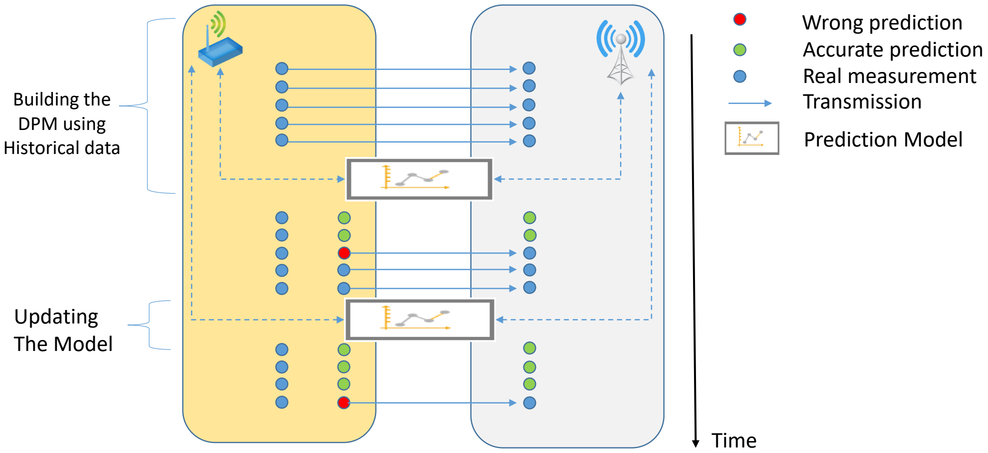

Figure 2: The concept of DPM

newly received value as NR and the time when it is received as t_{nr}. Subsequently, both the sensor and the sink calculate simultaneously the difference between these two measurements denoted d, as shown in the equation (2).

$$d[0] = x[1] - x[0] \tag{2}$$

Once the difference is calculated, the sensor and the Sink switches to the prediction phase, where they both assume that the difference between any two successive measurements will always be equal to $d[0]$. Based on the fact that time series data tend to change smoothly over time and the values of measurements at neighboring time ticks, are usually very close to each other. Therefore, as shown in equation (3), to predict $\widehat{x}$[k], the value of the measurement $x[k]$ at time t_k, both the sensor and the sink adds $d[0]$ to $\widehat{x}$[k-1] which is the predicted measurement at time $t_{(k-1)}$.

$$\widehat{x}[k] = \widehat{x}[k-1] + d[0] \tag{3}$$

Afterward, the sensor compares the predicted value $\widehat{x}[k]$ to the real sensed measurement $x[k]$. If the difference between them does not exceed an error threshold $emax$ predefined by the user, the real reading is discarded and not transmitted to the Sink. Meanwhile, when the Sink does not receive anything, it assumes that its prediction is within the error threshold.

Oppositely, if the prediction does not respect the error budget, the sensor discards it and transmits to the Sink the real reading, which we will refer to as "correction packet". Once the Sink receives the packet, they both update the value of d, by subtracting NR from the correction value $x[k]$ and dividing the result by the time difference between the reception of these tow measurements, as shown in the equation (4). And once again to keep track of the last received value for potential future update, the old value of NR is replaced in the memory by the newly received measurement $x[k]$.

$$d[k] = \frac{x[k] - NR}{t_k - t_{nr}} \tag{4}$$

In other words, d represents an estimated linear change rate of future readings, for a certain period of time. Moreover, as mentioned earlier, time series data usually have an increase, stand still and decrease cycle, especially when measuring environmental features, such as temperature, humidity, light etc. Thus, to harmonize the prediction line with the real data curve, d is multiplied by a rectification value denoted $\beta \in [0, 1]$.

Accordingly, the predicted value is calculated using equation (5) instead of equation (3).

$$\widehat{x}[k] = \widehat{x}[k-1] + d[k] \times \beta \tag{5}$$

The algorithm 1 illustrates the functioning of our method.

Algorithm 1 Transmission reduction.

```
Read $x_0$ and $x_1$
Transmit $x_0$ and $x_1$ to Sink
$d[0] \leftarrow x_1 - x_0$
$NR \leftarrow x_1$
$t_{nr} = t_1$
while $Energy \neq 0$ do
    Read $x_t$ at time $t_x$
    $\widehat{x_t} \leftarrow \widehat{x}_{t-1} + d \times \beta$
    if $|x_t - \widehat{x_t}| \geq emax$ then
        Send $x_t$ to Sink
        $d[t] \leftarrow \frac{x_t - NR}{t_x - t_{nr}}$
        $NR \leftarrow x_t$
        $t_{nr} \leftarrow t_x$
        $\widehat{x_t} \leftarrow x_t$
    end if
end while
```

5. Merging Adaptive Sampling and DPM based Transmission reduction

The Adaptive Sampling algorithm (AS) reduces the sampling rate of a sensor when the difference between collected measurements is not significant. Thus, enabling the sensor node to avoid collecting redundant and superfluous information. The Transmission Reduction algorithm (TR) reduces the number of data transmitted to the Sink, using a prediction model that can forecast future measurements within a narrow error range. The efficiency of the prediction model is at peak when data is smoothly changing with low variance between measurements.

Thus, one can notice that these two techniques are compatible. First their work does not overlap and they do not affect each other's results. Secondly, the prediction model is capable of filling the gap of "non collected data", since as mentioned before these measurements are mostly redundant or roughly similar to closely collected ones, and the prediction model efficiency is at maximum when the change in values is smooth and slow. Therefore, on one hand the sampling rate is reduced and on the other hand, the end user will still have access to the complete set of data. Finally the complexity of the transmission reduction algorithm is extremely low. Thus, when combined with the adaptive sampling algorithm the overall complexity will remain unchanged. Therefore, we propose to combine these two techniques into a single algorithm, enabling us to achieve lower energy consumption compared to each one of them when implemented solely.

Let us begin by explaining how to combine these two techniques together, and how the algorithm works as a whole. As mentioned earlier, the operations conducted by AS and TR does not overlap, and they do not affect the results of each others. Therefore, since they are totally compatible they can be implemented as they are originally without any changes in the way of working.

The only difference is, instead of having a single sampling rate, the sensor has two: a real one and a hypothetical one. The real sampling rate is the rate defined by the AS algorithm after each round. The hypothetical one is a fixed rate that is always equal to the maximum sampling rate. The sensor collects measurements at the real rate speed returned by AS. However, it uses the hypothetical rate while applying the TR algorithm. In other words, let us suppose for a round ro, the real rate is RR measures/period and the hypothetical rate is HR measures/period. In this case, during this period, on one hand the sensor collects RR measurements, on the other hand it predicts HR measurements. Note that RR is always less than HR, as HR is equal to the maximum sampling rate allowed for the sensor. Thus, the sensor is able to predict the "non collected" measurements caused by a slowed down sampling rate forced by AS.

Figure 3 gives an illustrative example on how this algorithm behave.

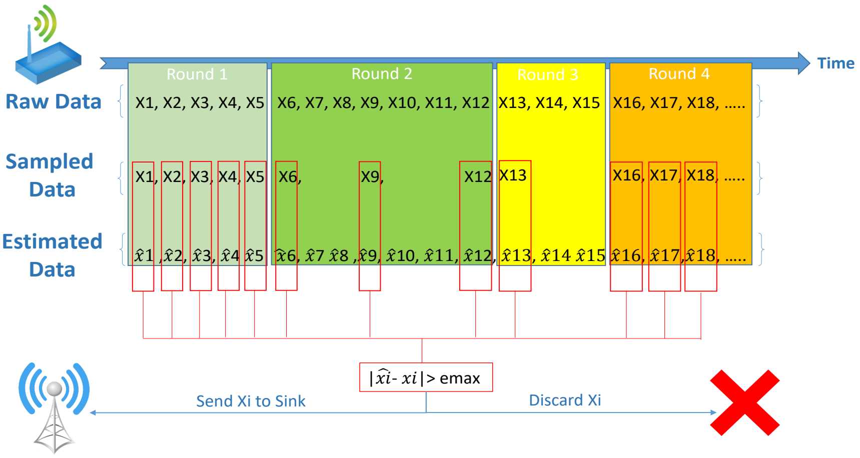

Figure 3: Illustrative example of our method (AS+TR)

We have explained in section 4.2 that every time a sensor predicts a new measurement it must compare it with the real sensed value in order to decide whether it should send it to the Sink or not. However, when adding AS to the equation, if RR is $\leq HR$ some predictions might not have a matching sensed measurements to validate its accuracy. Therefore, these predictions are considered to be within the error budget automatically. Since the "non collected" measurements are assumed to be redundant or already similar to collected neighboring measures. This assumption should not affect the accuracy of replicated data. However, we discuss this issue in the Experimental Results section below.

6. Experimental Results

In this section, we present the experimentation we have conducted on real wireless sensor readings, collected by twenty Crossbow TesloB nodes deployed in our laboratory as shown in Figure 4. The walls separating the rooms of the lab are of a thickness of 8 cm. Each one of the twenty nodes collects five environmental features: temperature, humidity, light, infrared and voltage. The collected measurements are directly transmitted to a central node (Sink) called SG1000 [22], connected to a laptop machine, through a star topology.

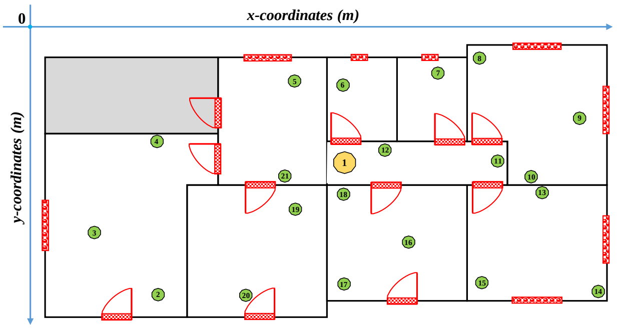

Figure 4: Deployment of the sensor network

We have chosen to test our approach on temperature, humidity, and infrared data since the variation in measured values for each one of them is different. For instance, the variation in temperature data is low, medium for humidity, and high for infrared. Thus, we can assess the efficiency of our approach on different scenarios. The simulation was conducted on twenty sets of 300,000 readings each (100,000 temperature, 100,000 humidity, and 100,000 infrared readings), which is equivalent to approximately 35 days of non-stop data collection. Since the effectiveness of these algorithms depends on the variation of the data being collected and can greatly differ from one node to another. The setup parameters for this experimentation are shown in table 2.

6.1. Transmission reduction

To test the effectiveness of our transmission reduction method compared to DPCAS. We have simulated both methods on the same sets of data using Matlab. The error threshold "emax" was set to ±0.1 for temperature and humidity, and ±1 for infrared.

Table 2: setup parameters

DPCAS		**AS+TR**	
Smin	310 sec	Smin	310 sec
Smax	31 sec	Smax	31 sec
smoothing coefficient α & multiplicative reduction factor β	0.2	Risk R	0.6
cubic parameter C	0.4	Rejection Probability α	0.05

Figures 5, 6 and 7 shows a comparison between the amount of temperature, humidity, and infrared data that have been transmitted to the sink when both DPCAS and AS+TR were implemented. For instance, On average, only 193 temperature readings or 0.193% were transmitted when our algorithm was implemented. When DPCAS is used, the number of data transmitted increased slightly to 714 (0.714%).

For humidity and infrared, the results were identical, our method outperformed DPCAS in reducing the number transmissions. The average amount of transmitted humidity and infrared data for AS+TR is 6148 and 5528 respectively. As for DPCAS the numbers increase to 13964 and 15848 respectively. Hence, these results show that our transmission reduction method is better at reducing radio communication, which enables the node to preserve more of its energy resources.

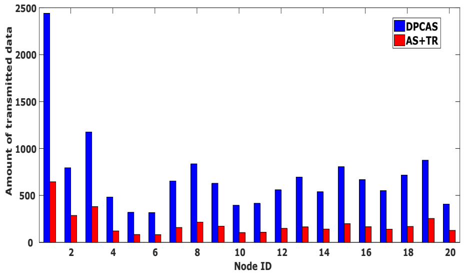

Figure 5: Amount of temperature data transmitted to the Sink

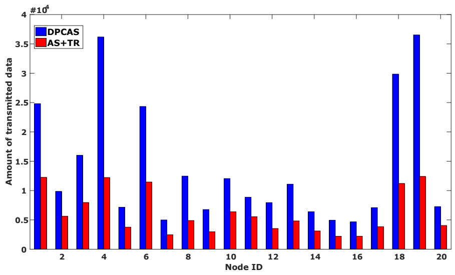

Figure 6: Amount of humidity data transmitted to the Sink

6.2. Adaptive Sampling

By looking at figure 8, 9 and 10, it is clear that DPCAS has the upper hand when it comes to reducing the sensing activity. Since the latter decreases gradually the sampling rate after each sample. However, AS decreases it only at the end of each period (after 50 samples in this experiment). The average amount of temperature, humidity and infrared data sampled by DPCAS are 16688, 26643 and 15847 respectively. As for AS+TR the numbers increase to 77731, 83873 and 46247 respectively. Hence, DPCAS outperforms our adaptive sampling algorithm in reducing the amount of sampled data and the energy consumed by the sensor board.

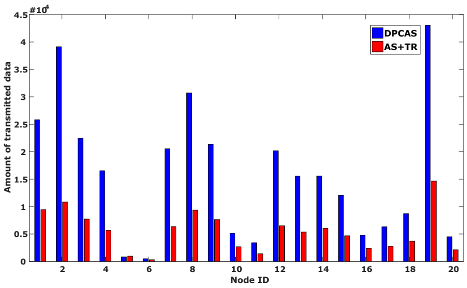

Figure 7: Amount of infrared data transmitted to the Sink

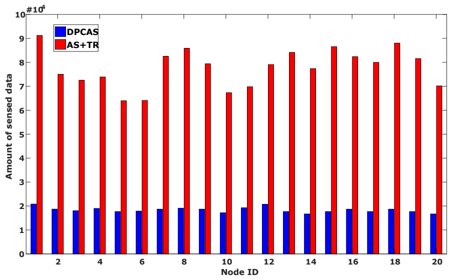

Figure 8: Amount of sampled temperature data

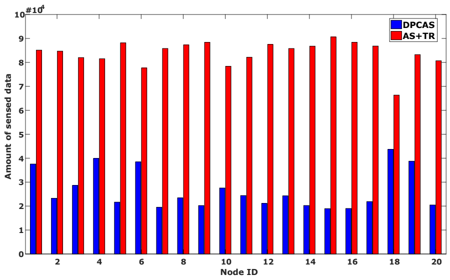

Figure 9: Amount of sampled humidity data

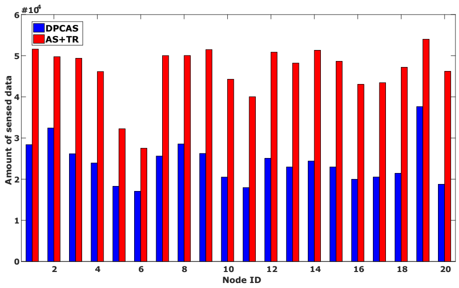

Figure 10: Amount of sampled infrared data

6.3. Energy consumption

It was shown that about 3000 instructions could be executed for the same energy cost as sending a bit for 100 meters by radio [23]. Therefore, to compare the amount of energy consumed by each method, we neglected the cost of computing operations by considering that both algorithms have the same computation complexity. We focused only on the cost of radio transmission and data sensing. In order to calculate the energy consumed by a sensor node we used the energy consumption model in [24]. The previous results demonstrated that on one hand, our method transmits less data to the sink. Therefore, the energy consumed by the radio component is less when compared to DPCAS. On the other hand, the latter samples fewer data. Therefore, the energy consumed by the sensor board is lower when compared to ours.

The amount of energy consumed by the sensor board is significantly smaller than the one consumed by the radio component. Therefore, the greater is the difference between the amount of data transmitted by each algorithm, the harder is for DPCAS to compensate the energy consumed by transmission with less sampling energy.

Figure 11, 12 and, 13 are a great proof for this assumption. The difference between the amount of temperature data transmitted by DPCAS and AS+TR is negligible, and large between the amount of sensed data. Consequently, the energy consumed by the sensor board will impact greatly the overall energy consumption calculation, which gave DPCAS the lead. However, when the gap between the number of transmissions grows bigger, as it is the case for humidity and infrared data. The energy consumed by the sensing activity will have a minor impact on the overall energy consumption calculation. Therefore, the total energy consumed by AS+TR for humidity and infrared data is smaller for the majority of the nodes.

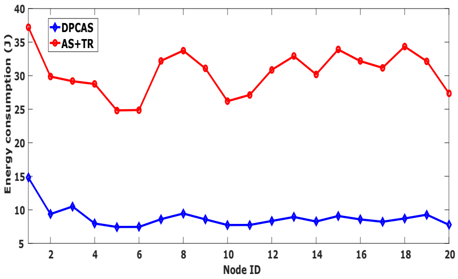

Figure 11: Energy consumption comparison for temperature data

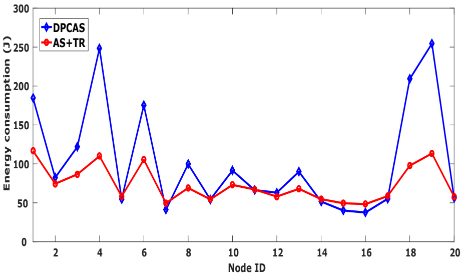

Figure 12: Energy consumption comparison for humidity data

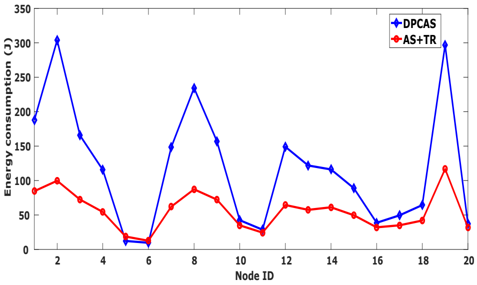

Figure 13: Energy consumption comparison for infrared data

6.4. Quality of replicated data

Data quality is a very important factor in WSNs, since the end user depends on it to make appropriate decisions. Accuracy, precision, completeness, and consistency are the attributes that measure the quality of data.

When we reduce the sampling rate within a certain period, we risk missing sudden variations in measurements. Thus, the estimation of these irregular non sampled data may exceed the desired error threshold. To study the impact of the adaptive sampling algorithms on the integrity of the replicated data we compare the estimated measurements with its corresponding raw data collected by the sensor node, and we calculate the values of 4 quality metrics: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Square Error (MSE), Root Mean Square Error (RMSE). The lower the values of these metrics the better are the results.

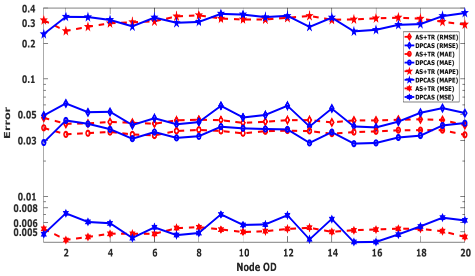

Figure 14: Quality metrics comparison for replicated temperature data

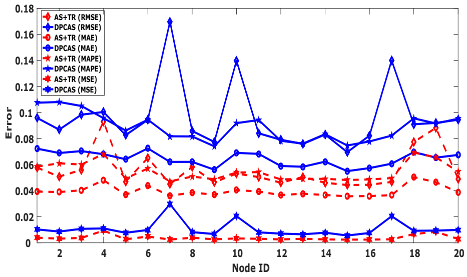

Figure 15: Quality metrics comparison for replicated humidity data

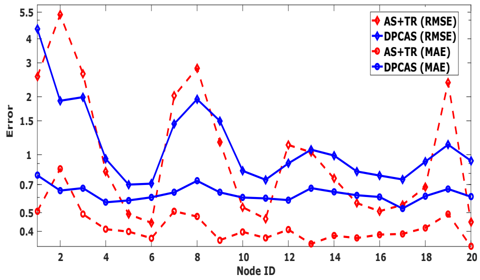

Figure 16: quality metrics comparison for replicated Infrared data

Figure 14, 15, and 16 shows a comparison between the quality metrics for each set of data of the three environmental features. For temperature data, both algorithms are neck to neck. However, AS+TR has a clear superiority for humidity and Infrared. Since Infrared contains 0 elements the value of MPAE cannot be computed. As for MSE in order to keep figure 16 simple an comprehensible, instead of plotting the curve we provide the average values which are 2.49 and 2.72 for AS+TR and DPCAS respectively.

Adaptive sampling makes a trade-off between data quality and the amount of sampled measurements, to deliver a minimum amount of readings while satisfying quality requirements of the application. Thus, the integrity of data depends on how tolerant is the end user to the error in replications. The obtained results demonstrated that our method was able to reproduce the whole data set with less error and better quality compared with DPCAS.

7. Conclusion

In this research work, we proposed an energy-efficient data reduction method for Wireless Sensor Networks based on a combination of the adaptive sampling and dual prediction mechanism techniques. The former, allows the sensor to adapt its sampling rate according to the variance in data. Thus, the sensor samples relevant data only and avoids the sampling of redundant and insignificant information. The latter enables the Sink to estimate the collected data through a prediction mechanism that is shared with the sensor node. Thus, Instead of transmitting all the readings, the sensor report to the Sink a measurement only when the estimation exceeds a predefined error threshold. By merging these two techniques together, we were able to reduce radio communication and data sensing at the same time. Since these two activities are considered to consume most of the energy resources, we were able to preserve a great amount of energy and extend the lifetime of the network significantly compared with another similar technique.

For future work, we plan on improving our adaptive sampling technique in order to reduce the risk of losing important information and increase the quality of the replicated data.

8. Acknowledgement

This work is partially funded by the Labex ACTION program (contract ANR-11-LABX-01-01), the France-Suisse Interreg RESponSE project and Lebanese University research program (Number: 4/6132).

References

[1] J. Yick, B. Mukherjee, D. Ghosal, Wireless sensor network survey, Computer networks 52 (12) (2008) 2292–2330.

[2] J. M. C. Silva, K. A. Bispo, P. Carvalho, S. R. Lima, Litesense: An adaptive sensing scheme for wsns, in: 2017 IEEE Symposium on Computers and Communications (ISCC), 2017, pp. 1209–1212.

[3] Y. Jon, Adaptive sampling in wireless sensor networks for air monitoring system, Master's thesis, Uppsala University, Department of Information Technology (2016).

[4] J. Yang, S. Tilak, T. S. Rosing, An interactive context-aware power management technique for optimizing sensor network lifetime, in: SENSORNETS, 2016, pp. 69–76.

[5] A. Makhoul, H. Harb, Data reduction in sensor networks: Performance evaluation in a real environment, IEEE Embedded Systems Letters 9 (4) (2017) 101–104.

[6] A. Makhoul, H. Harb, D. Laiymani, Residual energy-based adaptive data collection approach for periodic sensor networks, Ad Hoc Networks 35 (2015) 149–160.

[7] D. Laiymani, A. Makhoul, Adaptive data collection approach for periodic sensor networks, in: Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International, IEEE, 2013, pp. 1448–1453.

[8] A. Jain, E. Y. Chang, Y.-F. Wang, Adaptive stream resource management using kalman filters, in: Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data, SIGMOD '04, ACM, 2004, pp. 11–22.

[9] S. Santini, K. Römer, An adaptive strategy for quality-based data reduction in wireless sensor networks, in: Proceedings of the 3rd International Conference on Networked Sensing Systems (INSS 2006), 2006, pp. 29–36.

[10] B. Q. Ali, N. Pissinou, K. Makki, Approximate replication of data using adaptive filters in wireless sensor networks, in: 2008 3rd International Symposium on Wireless Pervasive Computing, 2008, pp. 365–369.

[11] L. Tan, M. Wu, Data reduction in wireless sensor networks: A hierarchical lms prediction approach, IEEE Sensors Journal 16 (6) (2016) 1708–1715.

[12] U. Raza, A. Camerra, A. L. Murphy, T. Palpanas, G. P. Picco, Practical data prediction for real-world wireless sensor networks, IEEE Transactions on Knowledge and Data Engineering 27 (8) (2015) 2231–2244.

[13] L. C. Monteiro, F. C. Delicato, L. Pirmez, P. F. Pires, C. Miceli, Dpcas: Data prediction with cubic adaptive sampling for wireless sensor networks, in: M. H. A. Au, A. Castiglione, K.-K. R. Choo, F. Palmieri, K.-C. Li (Eds.), Green, Pervasive, and Cloud Computing, Springer International Publishing, Cham, 2017, pp. 353–368.

[14] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, K. Pister, System architecture directions for networked sensors, ACM SIGOPS operating systems review 34 (5) (2000) 93–104.

[15] V. Raghunathan, C. Schurgers, S. Park, M. B. Srivastava, Energy-aware wireless microsensor networks, IEEE Signal Processing Magazine 19 (2) (2002) 40–50.

[16] J. Bahi, A. Makhoul, M. MEDLEJ, A two tiers data aggregation scheme for periodic sensor networks, Ad Hoc & Sensor Wireless Networks (2012) –.

[17] T. Du, Z. Qu, Q. Guo, S. Qu, A high efficient and real time data aggregation scheme for wsns, International Journal of Distributed Sensor Networks 11 (6) (2015) 261381.

[18] H. Wu, J. Wang, M. Suo, P. Mohapatra, A holistic approach to reconstruct data in ocean sensor network using compression sensing, IEEE Access PP (99) (2017) 1–1.

[19] A. Basheer, K. Sha, Cluster-based quality-aware adaptive data compression for streaming data, J. Data and Information Quality 9 (1) (2017) 2:1–2:33.

[20] H. Harb, A. Makhoul, Energy-efficient sensor data collection approach for industrial process monitoring, IEEE Trans. Industrial Informatics 14 (2) (2018) 661–672.

[21] P. E. McKight, J. Najab, Kruskal-wallis test, Corsini Encyclopedia of Psychology.

[22] advanticsys, Online data, http://www.advanticsys.com/wiki/index.php?title =SG1000.

[23] G. J. Pottie, W. J. Kaiser, Wireless integrated network sensors, Communications of the ACM 43 (5) (2000) 51–58.

[24] Y. Liang, W. Peng, Minimizing energy consumptions in wireless sensor networks via two-modal transmission, ACM SIGCOMM Computer Communication Review 40 (1) (2010) 12–18.