Orchestration-based mechanism for sampling adaptation in sensing-based applications

H. Harb¹² | H. Baalbaki¹ | C. Abou Jaoude² | A. Jaber¹

¹Computer Science Department, Faculty of Sciences, Lebanese University, Beirut, Lebanon
²Faculty of Engineering, TICKET Lab, Antonine University, Baabda, Lebanon

Abstract
Currently, the world witnesses a boom in the sensing-based applications where the number of connected devices is becoming higher than the people. Such small sensing devices are now deployed in billions around the world, collecting data about the surroundings and reporting them to the data analysis centres. This fact allows a better understanding of the world and helps to reduce the effects of potential risks. However, while the benefits of such devices are real and significant, sensing-based applications face two major challenges: big data collection and restricted power of sensor battery. In order to overcome these challenges, data reduction and sampling sensor adaptation techniques have been proposed to reduce data collection and to save the sensor energy. The authors propose an orchestration-based mechanism (OM) for adapting the sampling rate of the sensors in the network. OM is two-fold: first, it proposes a data transmission model at the sensor level, based on the clustering and Spearman coefficient, in order to reduce the amount of data transmitted to the sink; second, it proposes a sampling rate mechanism at the cluster-head level that allows searching the similarity between data collected by the neighbouring sensors, and then to adapt their sensing frequencies accordingly. A set of simulations on real sensor data have been conducted to evaluate the efficiency of OM, in terms of data reduction and energy conservation, compared to other exiting techniques.

1 | INTRODUCTION

Nowadays, sensor devices occupy broad zones in the world and monitor most of the daily life’s activities. Indeed, the number of currently deployed sensors arrives to 50 billion, and it is expected to reach 100 billion in the next 5 years [1,2]. Contemporarily, with such a massive number of sensors, researchers are able to monitor Anything around us from Any place then to deliver, at Any time, the collected data, to the sink node for later analysis and study. These characteristics of sensors, usually referred to as AAA, allows a new type of application known as sensing-based applications in which the internet of things (IoT) and the wireless sensor networks (WSN) form the pillars of such applications. Consequently, sensing applications quickly find their way into various sectors such as industrial, environment, military, agriculture, and security.

1.1 | Problem statement

Indeed, sensing-based applications provide many challenges for both community and researchers. On one hand, sensor’s hardware is limited in resources especially in battery supply which cannot be replaced or recharged in hostile or harsh environments [3,4]. On the other hand, the density deployment of the sensor nodes along with the need for continuous zone monitoring leads to a massive amount of data collection in such networks [5,6]. Consequently, transmitting such big data will quickly deplete the available energy of sensors. Therefore, researchers have focused on data reduction and adapting sensing frequency techniques to improve the energy consumption in sensors and reduce the complexity of data analysis at the sink node [7–10].
Our contribution

Here, an orchestration-based mechanism for energy conservation and minimising transmission in sensing-based applications is proposed. The objective of this mechanism is to adapt the sampling rate of each sensor according to the variation of the monitored condition, the remaining energy of the sensor and the similarity to data collected by neighbouring nodes. The contribution of this study is described as follows:

- At the sensor level, a new version of K-means clustering algorithm called SK-means, for example, Spearman-based K-means that combines the Spearman coefficient to the traditional K-means is proposed. The new version aims to overcome the two main challenges of traditional K-means: the selection of the optimal number of clusters and the convergence function. Subsequently, SK-means allows reducing the periodic data transmission from each sensor to the sink thus avoiding network overload and saving sensor energies.
- At the cluster-head (CH) node, the authors propose a model that, at first, allows searching for the spatial-temporal correlations between nodes. Then, based on the node correlation degree, the model adapts the sensing frequency of each sensor according to the similarity with data collected by its neighbours. This will lead to reduce the data collection size, eliminate the redundancy existing among nodes and enhance the power consumption of the sensor.

Through simulations on real sensor data, the effectiveness of the proposed mechanism has been validated in terms of minimising energy consumption and data transmission compared to other existing techniques.

The remaining article is organised as follows. Section 2 outlines different data reduction and energy-efficient techniques proposed in sensing-based applications. Section 3 depicts the periodic clustering architecture used in the network. Sections 4 and 5 present the data reduction model and the sampling rate model proposed at the sensor and CH levels, respectively. Simulation results are discussed in Section 6. Finally, the conclusion and future work are highlighted in Section 7.

2 RELATED WORKS

In recent years, proposing energy-efficient techniques was the main target almost of all researchers’ works. Their primary focus is to reduce the data routed over the network either at the sensor node or at intermediate nodes, mostly the CHs. Indeed, in most of the proposed techniques, the reduction process can be performed based on data aggregation [11], clustering [12], compression [13] or adapting sampling rate [7,8].

The authors in [9,10,14–17] dedicated their works in reducing the raw data transmission at the level of sensors. In [14], the authors propose a priority-based compressed data aggregation (PCDA) technique to reduce the amount of data transmitted. PCDA uses compressed sensing approach, based on the sensing matrix and convex optimisation, followed by a cryptographic hash algorithm, which uses a key predistribution scheme, at the biosensor level to save information accuracy before sending data for diagnosis. The simulation shows that PCDA ensures a low execution time and communication overhead with a moderate energy consumption. In [15], the authors propose a sequential lossless entropy compression (S-LEC) which organises the alphabet of integer residues obtained from differential predictor into increased size groups. S-LEC codeword consists of two parts: the entropy code specifying the group and the binary code representing the index in the group. The performance of S-LEC is evaluated based on real word dataset from sensor scope and volcanic monitoring and the obtained results show reduced energy consumption for dynamic volcano dataset compared to other existing techniques, particularly LEC and S-LZW. In [9], the authors propose three mechanisms that allow the sensor to adapt its sampling rate to the variation of the monitored environment. The proposed mechanisms are respectively based on similarity functions (Jaccard coefficient), distance functions (Euclidean distance) and analysis variance with statistical tests (ANOVA and Bartlett test). The proposed techniques work on rounds, where each round consists of a set of period time, in which the sensor adapts its sampling frequency at the end of each round. By adapting different scenarios, the proposed techniques realise minimum energy consumption with accurate data collection. Finally, the authors of [10] propose an adapted version of the dual prediction scheme (DPS) algorithm. The new version uses a collection of models for data prediction during the past sequences of the DPS algorithm without updating the history data table classically. Indeed, the new prediction model is computed at the sensors and sent to the sink or vice-versa. The performance of DPS is tested using the data collected from the meteorological station located at Tlemcen (Algeria) while the results show that the data transmission ratio is reduced by more than 90% when accurate predictions are achieved.

The authors in [18–23] dedicated their works to reducing the amount of data circulated in the network along the path to the sink, for example, at intermediates nodes. In [19], the authors propose a cluster-based data gathering algorithm for WSN called lifetime-enhancing cooperative data gathering and relaying (LCDGRA). LCDGRA works on three phases: the first phase aims to group the sensor nodes into clusters based on K-means clustering and Huffman coding algorithms. The second phase assigns a set of relay nodes to each CH in order to aggregate data before sending to the sink node. During the last phase, the aggregated data are coded based on random linear coding and then relayed to the base station. The simulations show that LCDGRA can ensure an efficient convergence of K-means (average of 31 iterations), a reduced data latency (up to 18%) and less energy consumption (up to 37%) compared to other techniques. In [20], an online data tracking and estimation (ODTE) is proposed to track poor data
collected at the sink. ODTE is mainly based on two systems: Data prediction system (DPS) and distortion factor (DF). DPS is used at the sensor in order to reduce its transmission using a defined limit while DF estimates an optimal data collected at the sink node. Although ODTE can highly reduce the data transmission and conserve the battery nodes, it is very complex in terms of computation and processing speed. The authors of [18] propose a routing protocol called gateway clustering energy-efficient centroid (GCEEC) for WSN. The objective of GCEEC is to improve the load among the sensor nodes as well as selects and rotates the CH near the energy centroid position of the cluster. The results show that GCEEC can extend the network lifetime highly and reduce the network overload. However, this is limited to many assumptions taken during the tested scenario. Finally, the authors of [21] propose a structure fidelity data collection (SFDC) technique dedicated to the cluster-based periodic applications in WSNs. SFDC searches both spatial and temporal correlation between nodes, using distance functions and similarity metrics, respectively. Then, it exploits the dependencies to reduce the number of nodes required to work for sampling and data transmission and prove that such reduction is bound to save energy.

The authors in [3–6,24,25] dedicated their works to minimizing the data transmission at several levels in the network, for example, sensor and CH levels. In [24], the authors propose a data management framework for data collection and decision making in connected healthcare. The framework relies on three algorithms: first, an emergency detection algorithm aims to send critical records directly to the coordinator; second, an adaptive sampling rate algorithm based on ANOVA and Fisher test in order to allow each sensor to adapt its sampling frequency to the variation of the patient situation; and third, a data fusion and decision making model is proposed at the coordinator and it is based on a decision matrix and the fuzzy set theory. Although it has great advantages for patient monitoring and assessment, the proposed framework suffers from two main disadvantages: (1) in case of low critical patient, none of the data would be archived in the hospital thus, revising patient archive to check patient progress from doctors is not possible and (2) predicting the progress of the patient situation in the subsequent periods of time is not possible by the doctors. The authors of [5] propose a spatial-temporal model to extend the network lifetime based on three similarity metrics: Euclidean distance, cosine similarity and Pearson product-moment coefficient (PPMC). Then, they propose a scheduling algorithm for switching correlated sensor nodes to the sleep mode. By performing real experiments, the authors show that PPMC gives the best results, in terms of conserving network energy, compared to other similarity metrics. However, PPMC has several disadvantages: (1) it does not search the similarity at the sensor node level; (2) it does not take into account the residual energy of the sensors when switching them to the sleep mode; and (3) it assumes that all the correlated sensors have the same degree of correlation. Finally, the authors of [25] propose a two-level node mechanism which is dedicated to periodic sensor applications. First, the authors propose an on-node aggregation method to remove redundant data collected by the sensor. Then, an in-network data reduction called prefix frequency filtering (PFF) is introduced at the CH level. PFF allows CHs to find similarities between data collected by neighboring nodes in the same cluster, using the Jaccard similarity function.

Despite that most of the proposed techniques allow efficient energy saving, they fail to satisfy all aspects of sensing-based applications and lack maturity. In addition, they are very complex and require massive processing. In the proposed work, an energy-efficient data reduction mechanism that is less complicated and suitable for limited resources sensor nodes is presented. Furthermore, the proposed mechanism takes into account several parameters when adapting the sampling rate of the sensor in order to save the integrity of the collected information.

3 CLUSTER-BASED ARCHITECTURE NETWORK

In sensing-based applications, the network architecture represents one of the most important challenges after deploying the sensor nodes. Indeed, some metrics (such as congestion, energy consumption, network overload, and data loss) are highly dependent on network architecture. Here, the proposed mechanism relies on two main concepts of the network: cluster-based architecture and periodic data acquisition.

On one hand, the cluster-based network has been considered as an efficient architecture for sensing applications in terms of energy conservation, high network scalability and data transmission. Typically, cluster-based architecture divides the sensors in the network into clusters and assigns a cluster-head (CH) for each cluster. CH is responsible for managing the data collected by the sensor members in that cluster. Subsequently, CH can perform any type of data processing (like aggregation, compression, scheduling, spatio-temporal correlation etc.) over the sensor data before sending toward the sink node. Figure 1 illustrates a simple network based on the cluster architecture in which the communication between the sensors and their CHs or the CHs and the sink is performed according to a single-hop transmission. But, dividing network into clusters is not an easy task and it faces many challenges. Hence, one can find a lot of works in the literature that are interested in issues related to cluster network like selection of cluster heads [26–28], optimisation of cluster size [29,30], communication between sensors/CHs and CHs/sink [31,32] etc. However, the concern of the authors is to study the variation of data collected by the sensors.
and not the formation of clusters themselves. Therefore, a geographical clustering scheme in which near sensors are already assigned to the same cluster is considered.

On the other hand, sensor nodes are responsible for monitoring the target zone and transmit the collected data toward the CHs, which, in turn, forward them to the sink. Unfortunately, data transmission is a high-cost operation in terms of energy consumption. Thus, considering its limited energy power, the lifetime of the sensor will decrease drastically if all the collected data are sent to the CH. Hence, periodic data acquisition model has been introduced in sensing-based applications with the aim of reducing the amount of collected and transmitted data from the sensors.

Basically, in a periodic acquisition model, data are collected periodically where each period \( p \) is partitioned into time slots. At each slot, each sensor node captures a new reading \( r_i \) then it forms, at the end of \( p \), a vector of \( \mathcal{F} \) readings as follows: \( R_i = \{r_1, r_2, \ldots, r_F\} \). After that, the sensor will send its vector of data, for example, \( R_i \), to its appropriate CH.

### 4 | SENSOR DATA REDUCTION MODEL

As mentioned before, data transmission consumes lots of the available sensor energy. Thus, in order to extend its lifetime, the amount of data transmission from the sensor should be reduced. However, data collected by the sensor devices are mostly redundant and contains useless information. Thus, one of the most effective solutions to reduce the data transmission is by eliminating redundancy and filtering non-useful information before sending them to the CH. This section proposes a data reduction model that allows each sensor to locally search the similarity between the data collected periodically, remove the existing redundancy then send toward the CH. The proposed model is based on the K-means algorithm adapted to the Spearman coefficient metric that allows finding similarities among data by grouping them into clusters.

#### 4.1 | Recall of K-means clustering algorithm

Generally, clustering is a data exploratory task that aims to group data into clusters in a way that the similarity among data in the same cluster is high and that among clusters is low. Researchers have proposed many data clustering techniques for various types of data. One of the most popular algorithms in data clustering is K-means [33]; it is flexible, simple, already adapted to a vast number of applications and used with various kinds of data [34–36].

Typically, the K-means is an iterative algorithm in which the process starts by randomly selecting an initial centroid for each cluster. Then, each data point is assigned to the nearest centroid and the first round of cluster formation is performed. After that, the cluster centroids are updated and the process is repeated until the convergence of the criterion function (Algorithm 1).

Subsequently, one of the most criterion functions that have been used in K-means is the sum of the square errors.

#### Algorithm 1 K-means algorithm

**Require:** Set of readings: \( R_i = \{r_1, r_2, \ldots, r_F\} \), Cluster number: \( K \).

**Ensure:** Set of clusters \( C = \{C_1, C_2, \ldots, C_K\} \).

1: for \( j = 1 \) to \( K \) do  
2: randomly choose centroid \( c_j \) among \( R_i \) belongs to \( C_j \)  
3: end for  
4: repeat  
5: for each reading \( r_i \in R_i \) do  
6: Assign \( r_i \) to the cluster \( C_j \) with nearest \( c_j \), \(|r_i, c_j| \leq |r_i, c_k|, j \in \{1, \ldots, K\}\)  
7: end for  
8: for each cluster \( C_j \), where \( j \in \{1, \ldots, K\} \) do  
9: Update the centroid \( c_j \) to be the centroid of all readings currently in \( C_j \), so that \( c_j = \frac{1}{|C_j|}\sum_{r_i \in C_j} r_i \)  
10: end for  
11: until all \( K \) clusters meet the criterion function convergence  
12: return \( C \)

#### 4.2 | Spearman coefficient metric

Spearman correlation is a non-parametric test used to measure the degree of association between two data sets. Spearman correlation is determined by ranking each point of the two data sets and, in case of ties, an average rank is used. Moreover, the Spearman correlation gives a value between +1 and −1; +1 indicates a perfect association of ranks, −1 indicates a perfect negative association of ranks while 0 indicates no association between the ranks of the two data sets. Unlike other tests, especially the Pearson coefficient, the Spearman test does not make any assumption about the distribution, or the linear relationship, of the values in the two data sets.

Mathematically, the Spearman correlation, \( \rho \), between two data sets \( R_s \) and \( R_f \) can be calculated according to the following equation:

\[
\rho(R_s, R_f) = 1 - \frac{6 \times \sum_{k=1}^{\mathcal{F}} d_k^2}{\mathcal{F} \times (\mathcal{F}^2 - 1)}
\]

where

- \( d_k \) is the difference between the ranks of corresponding values in the sets.
- \( \mathcal{F} \) is the number of values in each set.

Therefore, \( R_s \) and \( R_f \) are considered correlated sets with similar values if and only if the Spearman correlation is greater than a threshold \( \rho_s \) which is as follows:

\[
\rho(R_s, R_f) \leq \rho_s
\]
4.3 | K-means adapted to spearman coefficient: SK-means

After collecting the data at each period, for example, $R_i$, the sensor tries to minimise its size before sending it to the CH in order to save its energy. The authors propose to use the K-means algorithm to group similar data in $R_i$ into clusters then the data redundancy, in each cluster, will be eliminated before data transmission. However, the use of the traditional K-means faces two main challenges: the selection of the cluster number ($K$) and the convergence criterion function. On one hand, selecting the number of clusters is a crucial decision as it determines the data transmission ratio from the sensor and affects the accuracy of the information sent to the sink. On the other hand, the number of iterations generated by K-means is highly dependent on the selection of the convergence criterion function; thus, an inappropriate criterion function can lead to an increase in the computation process of K-means and thus affecting the data latency metric. Therefore, in order to overcome these challenges, a new version of K-means, called SK-means, adapting the Spearman correlation to the traditional K-means algorithm is proposed.

The idea behind SK-means is that all readings collected by a sensor during a period are similar and so they are assigned to the same cluster, for example, initial cluster. Then, it recursively divides the initial cluster into small clusters every time a high similarity between the readings inside the new cluster is detected. The criterion function used in SK-means to stop the cluster dividing and obtain the final clusters is the Spearman correlation. Algorithm 2 describes the process of SK-means that is applied over the readings collected at each period, $R_i$. First, all the readings are considered similar and $R_i$ is assigned to a temporary set of clusters, e.g. $L$ (line 1). Then, the Spearman correlation between the readings is calculated by dividing them into two equal subsets using the function $\text{Partition}$ (lines 2–4). Thus, if the correlation exceeds the Spearman threshold, then the readings are considered similar. Consequently, the average of the readings is computed (e.g. $\bar{\tau}$) and added with its weight (e.g. $\text{wg}(\bar{\tau})$) to the final reading set that will be sent to the CH (lines 5–9). Otherwise, for example, if the correlation does not exceed the Spearman threshold, the readings are considered unsimilar and the K-means algorithm is applied in order to divide them into two clusters (lines 10–14). The process is repeated on the new clusters until all readings within each cluster become similar. Therefore, each sensor will send a reduced set of readings $C_i$ in the form $\{ (\bar{\tau}_1, \text{wg}(\bar{\tau}_1)), (\bar{\tau}_2, \text{wg}(\bar{\tau}_2)), \ldots, (\bar{\tau}_k, \text{wg}(\bar{\tau}_k)) \}$ to the CH at the end of each period.

**Algorithm 2 SK-means algorithm**

Require: A period: $p$. Reading set: $R_i = \{ r_1, r_2, \ldots, r_F \}$, Cluster number: $K$, Spearman threshold: $\rho_s$.

Ensure: Reduced reading set: $C_i$.

1. $L \leftarrow \{ R_i \}$
2. for each set $R_j \in L$ do
3.   // divide readings in $R_j$ into two equal subsets
4.   $(R_{j1}, R_{j2}) = \text{Partition}(R_j)$
5.   if $\rho(R_{j1}, R_{j2}) \leq \rho_s$ then
6.     $\bar{\tau} = \text{average}(R_j)$
7.     $\text{wg}(\bar{\tau}) = |R_j| // \text{the total number of elements in } R_j$
8.     $C_i \leftarrow C_i \cup \{ (\bar{\tau}, \text{wg}(\bar{\tau})) \}$
9.   else
10.     remove $R_j$ from $L$
11.    $(C_1, C_2) = \text{K means}(R_j, 2)$
12.   $L \leftarrow L \cup \{ C_1, C_2 \}$
13.   remove $R_j$ from $L$
14. end if
15. end for
16. return $C_i$

4.4 | Analytical illustration of SK-means

This section shows the process of SK-means algorithm using an analytical example (Figure 2). It assumes a set $R_i$ consisting of 8 readings (i.e. $F = 8$) collected during a period. The first step is to divide $R_i$ into two equal partitions $R_{i1}$ and $R_{i2}$ of size 4, using the $\text{Partition}$ function. Then, the Spearman correlation is calculated between the partitions and it indicates that the partitions are not correlated (e.g. $\rho > \rho_s$). Thus, the K-means algorithm is applied over the set $R_i$ to divide the readings into two clusters $C_1$ and $C_2$. For each cluster, the process of dividing the readings into equal partitions and calculating the correlation between them is repeated. Whilst, the K-means is applied each time, a high correlation is detected between the readings until obtaining the final clusters. Finally, the mean values for all clusters are calculated while assigning their weights (e.g. the number of readings in each cluster). Therefore, the reduced set of readings (e.g. $C_2$) will be sent towards the CH.

**Figure 2** An illustrative example of S-K means
5 | CH SAMPLING RATE MODEL

Mostly, the data collected by the sensors are spatial-temporal correlated; on one hand, the spatial node correlation results from the dense deployment of the sensors along with the random scattering strategy. On the other hand, the temporal node correlation is dependent on the variation of the monitored condition, which can speed up or slowdown, which pushes the neighbouring nodes to collect redundant data. Thus, after receiving the datasets coming from the sensors, OM proposes a sampling rate model that allows the CH to search the spatial-temporal correlations among the sensors in order to adapt their sampling rate for the next period. The objective of the model is to reduce the amount of data collection at the sensors, thus enhancing their energies, and minimising the data correlation among neighbouring nodes before sending the data to the sink. Furthermore, the proposed CH sampling model takes into account two features to calculate the new sensing frequencies of each sensor: the spatial-temporal node correlation and the remaining energy of each sensor.

5.1 | Spatial-based node correlation

In large zone sensing applications, there is need to deploy a massive number of nodes to ensure the full zone coverage and maintain a high-reliability level of the collected data. In addition, in some harsh and hostile zones, the nodes are scattered over the target zone in a random manner. This leads to a specific spatial degree among the deployed nodes in which the distance between more the spatial degree is close, and vice versa. Then, two sensor nodes are considered spatially correlated if the geographical distance between them is less than a defined threshold. Let first define that each node \( N_i \) is representing by the following 4-tuples: \( N_i = \{x_i, y_i, S_i, T_r\} \), where \( x_i \) and \( y_i \) indicate the position of \( N_i \), \( S_i \) is its sensing range and \( T_r \) is its transmission range (Figure 3). Thus, it is considered that two nodes \( N_i \) and \( N_j \) are spatially correlated if the intersection between their sensing ranges is greater than a threshold \( \alpha \):

\[
\cap_{N_i,N_j} \geq \alpha \tag{3}
\]

where:

\[
\cap_{N_i,N_j} = 2 \times S_r \times \arccos \left( \frac{E_d}{2 \times S_r} \right) - E_d \sqrt{ \left( S_r - \frac{E_d^2}{4} \right) }
\]

and \( \alpha \) is the threshold for the sensing range intersection between two nodes that may be in \([0, 2 \times S_r]\), 0 indicates that both sensors monitor the same zone area and 2 \( S_r \) indicates that no spatial correlation among the nodes. Whilst \( E_d \) is the geographical Euclidean distance between \( N_i \) and \( N_j \).

5.2 | Temporal-based node correlation

The temporal correlation among nodes aims to find the similarities between their collected data, even when they are spatially correlated or not. But, the closer the geographical distance between nodes or the low variation of the monitored zone can increase the similarity between the data nodes. Subsequently, one can find several functions that allow searching the similarity among data sets such as Jaccard, Dice and Cosine. Here, the authors focus on the Jaccard similarity as one of the most used and well adapted functions to several domains. Then, in order to calculate the temporal correlation among two data sets \( C_i \) and \( C_j \) sent from two nodes \( N_i \) and \( N_j \) during a period, a score table (ST) is defined at first.

The ST is a customisable guide defined by the end-user or the expert that aims to determine the criticality of readings collected about a condition. Thus, the ST allows early detection of the critical situation and alerting the end-user as fast as possible. Typically, ST defines a normal range of readings, for example, \( |r_i, r_j| \), indicating that the monitored condition is in the normal situation. Readings in the normal range are assigning a score of 0. Then, the more the readings are far from the normal range, the more their criticalities (or scores) are. Table 1 shows ST that determines the criticality of the captured readings. First, ST determines the normal range of readings and then it defines a threshold \( \varepsilon \) in order to calculate the deviation of a reading from the normal range; readings can be more or less than the range normal. \( \varepsilon \) is a user-defined threshold determined according to the application requirements. The score of readings can take a value between 0 and 3 where 3 indicates high critical readings according to the normal range.

Based on the ST, the CH calculates the temporal correlation, according to Jaccard similarity, between two data sets \( C_i \)

![Figure 3](image_url)
and $C_i$ collected by nodes $N_i$ and $N_j$, respectively, according to the following steps:

- For each reading set, the CH calculates its score set. For instance, the score set $O_i$ of $C_i$ is as follows: $O_i = \{(o_{i1}, \text{wgt}(o_{i1})), (o_{i2}, \text{wgt}(o_{i2})), ..., (o_{in}, \text{wgt}(o_{in}))\}$ where $o_i$ is the score of the mean reading $\bar{r}_i$, $\text{wgt}(o_i) = \text{wgt}(\bar{r}_i)$ and $o_i \in [0, 3]$. 
- The Jaccard similarity is calculated based on the reading scores between both sets:
  \[
  J_{\text{score}}(C_i, C_j) = \frac{\sum_{t=0}^{3} \min(|\text{wgt}(o_i)|, |\text{wgt}(o_j)|)}{2 \times F - \sum_{t=0}^{3} \min(|\text{wgt}(o_i)|, |\text{wgt}(o_j)|)}
  \]
  where $|\text{wgt}(o_t)|$ is the total weight of readings having score $t$ in $O_i$.
- $N_i$ and $N_j$ are temporally correlated if the Jaccard similarity between their score sets is greater than a threshold $t_f$:
  \[
  J_{\text{score}}(C_i, C_j) \geq t_f
  \]
  where $t_f$ takes a value in $[0, 1]$. 

### 5.3 Degree of node correlation

In order to adapt the sampling rate of a sensor, the CH searches its correlation degree with other nodes. The correlation degree, indicated as $D_i$, of a node $N_i$ represents the set of neighbour nodes that are spatial-temporally correlated to $N_i$. Subsequently, based on Equations 3 and 4, two nodes $N_i$ and $N_j$ are considered spatial-temporal correlated if they are geographically close and their generated data sets $C_i$ and $C_j$ are similar as follows:

\[
\cap_{N_i, N_j} \geq \alpha \quad \text{and} \quad J_{\text{score}}(C_i, C_j) \geq t_f
\]

Therefore, the correlation degree of the node $N_i$ can be defined as: $D_i = \{N_j \text{ where } j \neq i, \text{ such that } \cap_{N_i, N_j} \geq \alpha \quad \text{and} \quad J_{\text{score}}(C_i, C_j) \geq t_f\}$. It is also assumed that $|D_i|$ is the number of nodes in $D_i$.

### 5.4 Sampling rate algorithm

In addition to the correlation degree of each node, the CH takes into account the remaining energy of the node in order to adapt its sampling rate. The intuition is that if the node has a high correlation degree and its battery level is at a low level compared to its correlated nodes, then its sampling rate should be decreased, and vice versa. Consequently, the energy of the sensor will be conserved and the redundancy among the data collected by neighbouring nodes will be minimised.

Algorithm 3 shows the sampling rate model applied at the CH after receiving data sets from sensors at each period. The algorithm takes the initial sampling rate of a sensor (e.g., the period size, $F$), its initial energy (e.g., $E_i$) and its correlation degree as input. Then, the algorithm calculates, as output, the new sampling rate of the sensor, for example, $S_i$ for the next period. Furthermore, OM defines a threshold known as the minimum sampling rate, for example, $S_{\text{min}}$ that allows taking into account the criticality of the monitored application; $S_{\text{min}}$ takes a value between 0% and 100% of the period size in which a value of $S_{\text{min}}$ is close to 0% indicating a less critical application while a value of $S_{\text{min}}$ is close to 100% indicating a more critical application. Moreover, an energy sampling threshold $\beta\%$ has been defined indicating the percentage of sampling rate that the sensor must add or reduce from its current sampling rate depending on the energy level of its correlated nodes; $\beta\%$ takes a value in [0%, 100%]. The process of Algorithm 5.4 starts by initialising the sampling rate of the sensor to its maximum, for example, the period size (line 1). After that, if the node has a spatial-temporal correlation with other nodes, then its sampling rate is reduced according to its correlation degree (lines 2–3). In addition, for each correlated node $N_j \in D_i$, if the remaining energy of $N_i$ is less (respectively more) than that of $N_j$ the sampling rate of $N_j$ must be further reduced (increased) by $\beta\%$ (lines 4–11). Finally, if the new sampling rate of $N_i$ is less than the minimum sampling rate determined for the application, then the new sampling rate of the sensor is set to $S_{\text{min}}$ (lines 12–14).

**Algorithm 3 Sampling rate algorithm**

**Require:** A node: $N_i$, A period size: $F$, Initial energy: $E_i$, Set of correlated nodes: $D_i$, Minimum sampling rate: $S_{\text{min}}$, Energy sampling threshold: $\beta$.

**Ensure:** New sensor sampling rate: $S_i$.

1: $S_i = F$
2: if $|D_i| > 0$ then
3: $S_i = F/(|D_i| + 1)$
4: for each node $N_j \in D_i$ do
5: if $E_i < E_j$ then
6: $S_i = S_i + \beta/100 \times F$
7: else
8: $S_i = S_i - \beta/100 \times F$
9: end if
10: end for
11: end if
12: if $S_i < S_{\text{min}}/100 \times F$ then
13: $S_i = S_{\text{min}}/100 \times F$
14: end if
15: return $S_i$
6 | SIMULATION RESULTS

In order to evaluate the performance of OM, multiple series of simulations using real sensor data collected from Intel Berkeley Research Laboratory [37] is performed. This data contains readings for 46 Mica2Dot sensors with weather boards collect humidity, temperature, light and voltage values. For the sake of simplicity, the simulation only focuses on the temperature field. For every 31 s, each Mica2Dot sensor collects new reading for each feature and it sends towards the sink for archive purpose.

In the simulation, a file that includes a log of about 50,000 readings for each sensor is used. It is assumed that each sensor reads the data from its corresponding file for a period of time, and then it sends them toward a CH placed at the centre of the lab after applying the proposed mechanism. Figure 4 shows the geographical distribution of Mica2Dot sensors in the Intel lab where each sensor takes an Id from 1 to 54 (yellow sign indicates some failure sensors). The algorithms used in OM are implemented based on the Java simulator and the obtained results are compared to those obtained in the PFF [23] and SLEC [15].

Table 2 summarises the parameters used in the simulation with their tested values.

Furthermore, the $\varepsilon$ threshold used in the score is set to 1 and thus the customizable score table adapted to the temperature readings is shown in Table 3:

---

**FIGURE 4** Distribution map of the sensors in the Intel lab

| Parameter                     | Symbol | Values                  |
|-------------------------------|--------|-------------------------|
| Period size                   | $F$    | 50, 100, 250 readings   |
| Simulation duration           | $\pi$  | $15 \times F \times 31$ s |
| Spearman threshold            | $\rho_s$| 0.4, 0.5, 0.6, 0.7     |
| Spatial correlation threshold | $A$    | $\frac{1}{2} \times S$, $S$, $\frac{3}{2} \times S$ |
| Jaccard similarity threshold  | $t_j$  | 0.6, 0.7, 0.8           |
| Sensor energy                 | $E_i$  | 5 mJ                    |
| Minimum sampling rate         | $S_{min}$| 20%, 30%, 50%          |
| Energy sampling threshold     | $\beta$| 5%, 10%                 |

**TABLE 2** Simulation environment

---

**6.1 | Reading score study**

Figure 5 shows the reading values collected by five sensors randomly selected (Figure 5a) with their calculated scores (Figure 5b) according to Table 3. The obtained results show the following observations: (1) the temperature condition in the Intel lab changes very slowly due to the high redundancy existing among the data collected by the sensors; (2) The spatial-temporal correlation among neighbouring nodes is high; for instance, sensors 1 and 2 or sensors 3 and 4; (3) The spatial correlation between sensors does not always lead to a temporal correlation among the collected data; For instance, the sensors 1 and 2 are spatial correlated and they generate similar data until the period number 500 then, after this period, the collected data are becoming unsimilar; (4) Sensors that are not spatially correlated can sometimes have a temporal correlation; For instance, the sensors 2 and 5 are not geographically close but, starting from the period number 700, they are collecting redundant data; and (5) The correlation among the nodes can also be verified through the same scores calculated for the data collected by the correlated nodes; the readings scores of sensors 1 and 2 are mostly varying between 0 and 2 while those of sensors 3 and 4 are between 2 and 3. Thus, the

---

**FIGURE 5** A portion of readings collected by a sensor with their calculated scores
$\rho_s = 0.6$, $\alpha = S_r$, $t_J = 0.7$, $S_{min} = 30\%$, $\beta = 5\%$

$\mathcal{F} = 100$, $\alpha = S_r$, $t_J = 0.7$, $S_{min} = 30\%$, $\beta = 5\%$

$\mathcal{F} = 100$, $\rho_s = 0.6$, $t_J = 0.7$, $S_{min} = 30\%$, $\beta = 5\%$

$\mathcal{F} = 100$, $\rho_s = 0.6$, $\alpha = S_r$, $S_{min} = 30\%$, $\beta = 5\%$

$\mathcal{F} = 100$, $\rho_s = 0.6$, $\alpha = S_r$, $t_J = 0.7$, $S_{min} = 30\%$, $\beta = 5\%$

$\mathcal{F} = 100$, $\rho_s = 0.6$, $\alpha = S_r$, $t_J = 0.7$, $S_{min} = 30\%$
criticality of the temperature condition in the Intel can change from one place (e.g. sensor) to another.

6.2 Data transmission ratio at sensor

Figure 6 shows the percentage of readings sent from each sensor to the sink after applying the proposed mechanism (e.g. OM), PFF and S-LEC techniques. The results are dependent on $F$ (Figure 6a), $\rho_t$ (Figure 6b), $\alpha$ (Figure 6c), $t_f$ (Figure 6d), $S_{\text{min}}$ (Figure 6e) and $\beta$ (Figure 6f). It is observed that OM can reduce the data transmission to the CH more than the PFF and S-LEC in all cases. Subsequently, it allows each sensor to send up to 65% of data less than PFF and up to 78% of data less than S-LEC. Furthermore, the obtained results show that:

- By decreasing the period size, OM allows each sensor to reduce its data transmission to the sink (Figure 6a). This is because when the period size increases, the redundancy among the collected data decreases. Thus, the sensor must increase its data transmission in order to save the information integrity.
- By increasing the spatial correlation threshold or decreasing the Jaccard similarity threshold, the periodic data transmission from each sensor will decrease (Figure 6c,d). This is due to the spatial-temporal correlation which will increase when $\alpha$ increases or $t_f$ decreases. Thus, the CH will reduce the sampling rate of neighbouring nodes in order to reduce the data redundancy among their collected data.
- By decreasing the minimum sampling rate, OM gives better results in terms of data reduction at the sensor (Figure 6e). This is because the criticality of the monitored application will increase when $S_{\text{min}}$ increases and thus, the CH must increase the sensor sampling rate in order to increase the reliability of the decision making.

6.3 Sensor sampling rate study

This section shows the performance of the sampling rate model proposed at CH level in OM in terms of adapting the sensing frequency of a sensor based on the spatial-temporal correlation with its neighbouring nodes. The performance is studied based on the variation of the energy sampling threshold (Figure 7a) and the minimum sampling frequency (Figure 7b) for a set of 15 periods. The obtained results show that the sampling rate of the sensor is dynamically adapted after each period depending on the spatio-temporal correlation between nodes. Although the spatial correlation is fixed respecting to the nodes deployment, the temporal correlation can differ from period to another. Furthermore, the following observations are eminent:

- By varying the energy sampling threshold (Figure 7a), it is shown that the sampling rate of the sensor is more reduced with the increasing value of $\beta$. Therefore, with $\beta = 10\%$, the sensor will reduce its sampling rate to the minimum sampling threshold more than with $\beta = 5\%$. This confirms the behaviour of OM by reducing its sampling rate in order to save further its energy.
- By varying the minimum sampling threshold (Figure 7b), it is shown that the sensor reduces more its sampling rate when the value of $S_{\text{min}}$. For instance, the sampling rate of the sensor varies mostly between 20 and 70 when $S_{\text{min}} = 20\%$, between 30 and 80 when $S_{\text{min}} = 30\%$ and 40 and 90 when $S_{\text{min}} = 40\%$. This also confirms the behaviour of OM by increasing (respectively decreasing) the sampling rate when the criticality of the application is high (respectively low).

6.4 Data loss study

Figure 8 shows the percentage of data loss after applying OM and the PFF technique, with respect to several parameters. But the study of data loss is an essential metric in sensing-applications that can affect the decision made at the end-user. In the simulation, a reading is considered as a lost reading if a sensor collects it but it is not sent either by their correlated neighbouring to the sink. Thus, the percentage of data loss is calculated by dividing the lost readings by all the sensors over the entire raw data. The obtained results show that OM can
\( \rho_s = 0.6, \ \alpha = S_r, \ t_J = 0.7, \)

\( S_{\min} = 30\%, \ \beta = 5\% \)

\( \mathcal{F} = 100, \ \alpha = S_r, \ t_J = 0.7, \)

\( S_{\min} = 30\%, \ \beta = 5\% \)

\( \mathcal{F} = 100, \ \rho_s = 0.6, \ \alpha = S_r, \)

\( S_{\min} = 30\%, \ \beta = 5\% \)

**Figure 8** Data loss

outperform PFF in terms of maintaining the accuracy of data compared to PFF in all cases. Subsequently, the percentage of data loss using OM does not reach 4\%, in the worst case, while it exceeds 6\% using PFF. This is because PFF reduces the data transmission from sensors based on the temporal correlation only while OM uses both spatial and temporal correlation, which increases the accuracy of the collected data. Furthermore, it can be observed that percentage of data loss usually decreases with the increase of the amount of data transmitted (see Figure 6). Therefore, the data accuracy of OM increases with the increasing values of \( \rho_s \) and \( t_J \), or the decreasing values of \( \mathcal{F} \) and \( \alpha \).

**Figure 9** Distribution of the sensor readings over clusters after applying SK-means

6.5 | Reading weight study

Figure 9 shows the number of obtained clusters after applying the SK-means algorithm over the periodic data collected by a sensor, for a set of 10 periods. In addition, the figure shows the number of readings assigned to each cluster, for example,
cluster weight. It is observed that the number of clusters is dynamically changing among the periods depending on the redundancy among the collected data; more the collected data are redundant then less the number of clusters is, and vice versa. Subsequently, the number of clusters varies between 4, in the best case, and 10, in the worst case. On the other hand, it is observed that the cluster weights are unequally distributed because of the random selection of the initial cluster centroids in SK-means; the cluster weights vary between 4 and 26 readings. However, most of the clusters weights are close to 4 than to 26, which increase the integrity of the merged information.

6.6 Further discussion

This section gives further consideration to the proposed technique by studying the feasibility of applying it under certain conditions and circumstances of the application.

From the data collection point of view, OM outperforms compression (e.g. S-LEC) and aggregation (e.g. PFF) techniques in terms of reducing the sensor sampling frequency, under different parameter values. Thus, for applications where a dense network is deployed and a high level of redundancy is existing, OM can be efficiently adapted.

From the data accuracy point of view, OM can adjust the values of \( F, \rho_s, \alpha \) and \( t_f \) to preserve the accurate level of collected data. Therefore, for the critical applications where a high level of accuracy is needed, like in military and healthcare applications, the values of \( F \) and \( \alpha \) can be decreased or the values of \( \rho_s \) and \( t_f \) can be increased. Otherwise, for low critical applications like weather monitoring, \( F \) and \( \alpha \) must be increased or \( \rho_s \) and \( t_f \) must be decreased.

From the energy consumption point of view, OM can extend the lifetime of the network highly and save the battery power of sensors by adjusting several parameters. For instance, for remote and harsh environments where the sensor battery cannot be recharged or replaced, the experts should decrease the values of \( F, \rho_s, t_f \) and \( S_{\text{min}} \) or increase the values of \( \alpha \) and \( \beta \). Otherwise, for applications where the sensor power is not a priority, the reverse process should be adapted.

7 CONCLUSIONS AND FUTURE WORK

Studying data correlation along with the adaptation of sampling rate strategies for the sensor is one of the most effective solutions to save energy and reducing data collection in sensing-based applications. The authors proposed an OM for adapting the sampling rate of the sensors in the network. OM is based on two folds and it works at the sensor and CH levels. At the first level, it reduces the periodic data transmission of the sensor based on a new version of data clustering called SK-means. At the second level, a sampling rate mechanism that allows the CH to search the similarity between data collected by the neighbouring sensors and then to make them adapt accordingly, their sensing frequencies are proposed accordingly. Through simulations on real sensor data, the efficiency of the proposed mechanism, in terms of data reduction and energy conservation is evaluated, and compared to other exiting techniques.

In the future works, there are two main directions to enhance OM. First, a scheduling strategy can be added at the CH to switch correlated nodes into sleep/active modes. Second, the sampling rate model proposed at the CH will be enhanced by studying the correlation among data collected by the same sensor at successive periods. Thus, the data collection at the sensor node level will be further reduced.

ORCID

H. Harb https://orcid.org/0000-0003-3066-9189

REFERENCES

1. Zeinab, K.A.M., Elmustafa, S.A.A.: Internet of things applications, challenges and related future technologies, World Sci. News. 2, 126–148 (2017)
2. Al-Fuqaha, A., et al.: Internet of things: A survey on enabling technologies, protocols, and applications, IEEE Commun. Surv. Tutor. 17, 2347–2376 (2015)
3. Ibrahim, M., et al.: ON-IN: An on-node and in-node based mechanism for big data collection in large-scale sensor networks. In:2019 27th European Signal Processing Conference (EUSIPCO), pp. 1–5 (2019)
4. Ibrahim, M., et al.: Adaptive Strategy and Decision Making Model for Sensing-Based Network Applications. In: 2019 19th International Symposium on Communications and Information Technologies (ISCIT), pp. 96–101 (2019)
5. Sunil, D., Kalpana, S.: Energy conservation in wireless sensor networks by exploiting inter-node data similarity metrics, Intl. J. Energy Inform. Commun. 6, 23–32 (2015)
6. Harb, H., Makhoul, A., Couturier, R.: An enhanced K-means and ANOVA-based clustering approach for similarity aggregation in underwater wireless sensor networks, IEEE Sensor. J. 15, 5483–5493 (2015)
7. Abdul-Wahab, S.A., et al.: Prediction of optimum sampling rates of air quality monitoring stations using hierarchical fuzzy logic control system, Atmos. Pollut. Res. 10, 1931–1943 (2019)
8. Rao, Y., et al.: Adaptive data acquisition with energy efficiency and critical-sensing guarantee for wireless sensor networks, Sensors. 19, 2654 (2019)
9. Harb, H., Makhoul, A.: Energy-efficient sensor data collection approach for industrial process monitoring. IEEE Trans. Indus. Inform. 14, 661–672 (2017)
10. Liu, Z., Lebsaine, M., Lazzid, A.: An improved adaptive dual prediction scheme for reducing data transmission in wireless sensor networks. Wirel. Netw. 25, 3545–3555 (2019)
11. Pushpalatha, S., Shivarapakasha, K.S.: Energy-efficient communication using data aggregation and data compression techniques in wireless sensor networks: a survey. In: Advances in Communication, Signal Processing, VLSI, and Embedded Systems, pp. 161–179 (2020)
12. Kuila, P., Jana, P.K.: Evolutionary computing approaches for clustering and routing in wireless sensor networks. Sens. Technol. Concept Methodol. Tool Appl. pp. 125–146 (2020)
13. Tuama, A.Y., et al.: Recent advances of data compression in wireless sensor network, J. Eng. Appl. Sci. 13, 9002–9015 (2018)
14. Soufiane, B.O., et al.: Lightweight and confidential data aggregation in healthcare wireless sensor networks, Trans. Emerg. Telecommun. Technol. 27, 576–588 (2016)
15. Liang, Y., Li, Y.: An efficient and robust data compression algorithm in wireless sensor networks, IEEE Commun. Lett. 18, 439–442 (2014)
16. Russo, A., Verdier, E., Miramond, B.: Energy saving in a wireless sensor network by data prediction by using self-organized maps, Procedia Comput. Sci. 130, pp. 1090–1095 (2018)
17. Xu, Q., et al.: Cluster-based arithmetic coding for data provenance compression in wireless sensor networks, Wirel. Commun. Mobile Comput. (2018).
18. Qureshi, K.N., et al.: Optimized cluster-based dynamic energy-aware routing protocol for wireless sensor networks in agriculture precision, J. Sensors. (2020).
19. Agbula, G.P., Kumar, G.J.R., Juliet, A.V.: A lifetime-enhancing cooperative data gathering and relaying algorithm for cluster-based wireless sensor networks, Intl. J Distribut. Sensor Netw., 16, 1550147719900111 (2020).
20. Karjee, J., Kleinsteuber, M.: Data estimation with predictive switching mechanism in wireless sensor networks, Intl. J. Sensor Netw. 25, 184–197 (2017).
21. Wu, M. Tan, L., Xiong, N.: A structure fidelity approach for big data collection in wireless sensor networks, Sensors. 15, 248–273 (2015).
22. Al-Tabbakh, S.M.: Novel technique for data aggregation in wireless sensor networks. 2017 International conference on internet of things, embedded systems and communications (IINTEC), pp. 1–8 (2017).
23. Alghamdi, Y., Abdullah, M.: Improving K-means algorithm by grid-density clustering for distributed WSN data stream, Intl. J. Adv. Comput. Sci. Appl. 9, pp. 583–588 (2018).
24. Halib, C., et al.: Self-adaptive data collection and fusion for health monitoring based on body sensor networks, IEEE Trans. Indus. Inform. 12, 2342–2352 (2016).
25. Bahi, J.M., Makhoul, A., Medlej, M.: A two tiers data aggregation scheme for periodic sensor networks, Ad Hoc Sensor Wirel. Netw. 21, 77–100 (2014).
26. Baradaran, A., Navi, K.: HQCA-WSN: high-quality clustering algorithm and optimal cluster head selection using fuzzy logic in wireless sensor networks, Fuzzy Set. Syst. 389, 114–144 (2020).
27. Ren, Q., Yao, G.: An energy-efficient cluster head selection scheme for energy-harvesting wireless sensor networks, Sensors. 20, 187 (2020).
28. Ngangbam, R., Hossain, A., Shukla, A.: Improved low energy adaptive clustering hierarchy and its optimum cluster head selection, Intl. J. Electron. 107, 390–402 (2020).
29. Younif, Y., et al.: An energy efficient and load balancing clustering scheme for wireless sensor network (WSN) based on distributed approach, J. Phys. Conf. Ser. 1019, 012007 (2018).
30. Kang, S.: Energy optimization in cluster-based routing protocols for large-area wireless sensor networks, Symmetry. 11, 37 (2019).
31. Gupta, G.: Improved cuckoo search-based clustering protocol for wireless sensor networks, Procedia Comput. Sci. 125, 234–240 (2018).
32. Rais, A., Bouragba, K., Ouzzif, M.: Routing and clustering of sensor nodes in the honeycomb architecture, J. Comput. Netw. Commun. (2019).
33. MacQueen, J.: Some methods for classification and analysis of multivariate observations. Proc. Fifth Berkeley Symp. Math. Stat. Probab. 1, 281–297 (1967).
34. Govender, P., Sivakumar, V.: Application of k-means and hierarchical clustering techniques for analysis of air pollution: A review (1980–2019), Atmospher. Pollut. Res. 11, 40–56 (2020).
35. Lavanya, K., et al.: An enhanced K-means MSOINN based clustering over Neo4j with an application to weather analysis, Intl. Conf. Intel. Comput. Smart Commun. 2020, 451–461 (2019).
36. Zhang, G., Li, Y., Deng, X.: K-means clustering-based electrical equipment identification for smart building application. Information. 11, 27 (2020).
37. Madden, S.: Intel lab data. http://db.csail.mit.edu/labdata/labdata.html (2004).

How to cite this article: Harb H, Baalbaki H, Jaoude CA, Jaber A. Orchestration-based mechanism for sampling adaptation in sensing-based applications. IET Smart Cities. 2021;1–13. https://doi.org/10.1049/smc2.12002