**Peer-to-Peer Networking and Applications manuscript No.**
(will be inserted by the editor)

# Energy-Efficient Scheduling Strategies for Minimizing Big Data Collection in Cluster-based Sensor Networks

**Hassan Harb**[*] · **Abdallah Makhoul**
[*] *Corresponding author*

Received: date / Accepted: date

**Abstract** Today, wireless sensor networks (WSNs) have been widely used in monitoring various applications, such as environment, military and healthcare, etc. The explosive growth of the data volume generated in these applications has led to one of the most challenging research issues of the big data era. To deal with such amounts of data, exploring data correlation and scheduling strategies have received great attention in WSNs. In this paper, we propose an efficient mechanism based on the Euclidean distance for searching the spatial-temporal correlation between sensor nodes in periodic applications. Based on this correlation, we propose two sleep/active strategies for scheduling sensors in the network. The first one searches the minimum number of active sensors based on the set covering problem while the second one takes advantages from the correlation degree and the residual energy of the sensors for scheduling them in the network. Our mechanism with the proposed strategies were successfully tested on real sensor data. Compared to other existing techniques, the simulation results show that our mechanism significantly extends the lifetime of the network while conserving the quality of the collected data and the coverage of the monitored area.

**Keywords** Periodic sensor networks (PSNs) · Big-data sensing · Spatial-temporal data correlation · Scheduling sensors · Clustering topology · Real data readings

H. HARB
Department of Computer Science, Faculty of Sciences, American University of Culture and Education (AUCE), Lebanon
Tel.: +961-03-399 252
E-mail: hassanharb@auce.edu.lb

A. MAKHOUL
FEMTO-ST Institute/CNRS, the DISC department, Univ. Bourgogne Franche-Comté, Belfort, France
E-mail: abdallah.makhoul@univ-fcomte.fr

## 1 Introduction

Nowadays, wireless sensor networks (WSNs) can be considered as one of the most important source of big data era. In some applications, such as healthcare services and atmospheric conditions monitoring [21] and commercial flights [22], the volume of data generated by sensors nodes reaches the order of petabytes every day. Moreover, the sensor nodes have a limited energy supply. Therefore, the problems of energy constraint and data redundancy emerge inevitably at the core of WSNs challenges.

To deal with big data generated in WSNs, recent studies [23–25] pay a great attention to inter-nodes data correlation techniques and scheduling nodes strategies. The aim of such approaches is to schedule sensors that generate high spatial-temporal correlation into sleep/active mode thus, enhancing the network lifetime. In this paper, we focus on periodic real-time data collection model in WSNs based on clustering architecture. In one hand, periodic model has been used over a wide range of areas [26, 27] where each sensor periodically collects local readings of interest then transmitting them toward the sink. On the other hand, recent works have mainly focused on the clustering architecture as an efficient topology of WSNs that organizes data traffic and improves scalability of the network [28, 29]. Hence, our main goal in this paper is to minimize the huge amount of data generated in clustering-based periodic sensor networks, by searching the spatial and temporal correlation between neighboring nodes. When correlated nodes are detected, we propose two scheduling strategies in order to switch sensors in each cluster into sleep/active modes. The first strategy is based on the set cover problem while the second strategy takes into account the correlation degree and the residual energy of the sensors when scheduling nodes in the cluster.

The remainder of this paper is organized as follows. Section 2 presents the related work on data correlation and scheduling techniques in sensor networks. Section 3 describes our mechanism, based on the Euclidean distance, for searching spatially-temporally correlated nodes. In Section 4, we propose two strategies for scheduling sensors in the network. Simulation results based on real data readings are exposed in Section 5. Finally, we conclude our paper and we provide our directions for future work in Section 6.

## 2 Related Work

In WSNs, a lot of techniques have been proposed in order to explore the spatial-temporal data correlation between sensors [12–14]. The main objective of such techniques is to reduce the energy consumption in data collection and minimize the redundancy existing in the network. Recently, the authors in [10, 11] present a comprehensive overview about different spatial-temporal data correlation techniques and sleep scheduling methods proposed in the literature for WSNs.

Some works, such as [15–18], study the spatial-temporal correlation between sensor readings as the aggregation process of similar data. In [15], the authors propose a spatial-temporal correlation based fault-tolerant event detection scheme, called STFTED, which leverages a two-stage decision fusion and exploiting spatial-temporal correlation of sensor nodes. In [17], the authors propose an architecture for dynamic and distributed data-aware clustering, and the Dynamic Data-aware Firefly-based Clustering (DDFC) protocol to handle spatial similarity between node readings. The DDFC operation takes into account the biological principles of fireflies to ensure distributed synchronization of the clusters similar readings aggregations.

In other works, such as [5, 19, 2, 20], the spatial-temporal correlation between sensors have been studied in order to schedule sensors in the network. In [5], the authors propose an Efficient Data Collection Aware of Spatial-Temporal Correlation (EAST) for energy-aware data forwarding in WSNs. In EAST, nodes that detected the same event are dynamically grouped in correlated regions and a representative node is selected at each correlation region for observing the phenomenon. In [2], the authors propose a centralized algorithm design and an optimizing protocol for scheduling the sensors during a specified network lifetime. The objective is to maximize the spatial-temporal coverage by scheduling sensors activity after they have been deployed.

Recently, the authors in [9] 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. 3) it assumes that all the correlated sensors have the same degree of correlation. Hence, aiming to overcome these disadvantages, we propose, in this paper, a new mechanism based on the Euclidean distance for searching inter-node data correlation. Once high correlation between inter-node is noticed, we propose two sleep/active strategies for scheduling sensors in the network. Through simulation, we will show that our mechanism, with the two proposed strategies, can significantly outperform PPMC in terms of saving the sensors energies and extending the network lifetime.

## 3 Spatial-Temporal Correlation Mechanism

In WSN, sensors are deployed densely in order to monitor some phenomenon which leads to have high spatial-temporal correlation between sensed data. In the section, we propose a new mechanism, based on the Euclidean distance, in order to exploit spatial-temporal correlation bewteen sensed data in WSN.

### 3.1 Local Temporal Correlation

In periodic applications, each sensor node collects a vector of readings in each period then it send it to the CH at the end of the period. Mostly, consecutive readings collected from the sensor, in each period, are temporally correlated depending from how the monitored condition varies. We call this correlation a *local temporal correlation.* Let us consider a vector of readings $R_i$ collected by the sensor $S_i$ during period $p$ as follows: $R_i = [r_1, r_2, \ldots, r_{\mathcal{T}-1}, r_{\mathcal{T}}]$ where $\mathcal{T}$ is the total number of readings captured during $p$. Thus, our objective is to explore temporal correlation between readings in $R_i$ in order to reduce the amount of data readings that need to be transmitted and thus to save the energy in $S_i$. Hence, we propose "*LocTmp*" function to identify if two consecutive readings $r_t$ and $r_{t+1}$, captured by the sensor $S_i$ during a period $p$, are similar or not. *LocTmp* function is defined as follows:

**Definition 1 (*LocTmp* function)** We define the *LocTmp* function between two consecutive readings $r_t$ and $r_{t+1}$ as:

$$LocTmp(r_t, r_{t+1}) = \begin{cases} 1 & \text{if} \quad |r_t - r_{t+1}| \le \delta, \\ 0 & \text{otherwise.} \end{cases}$$

where the threshold value $\delta$ is a user defined and it depends on the application. Two consecutive readings captured by a sensor are considered similar if and only if their *LocTmp* function is equal to 1.

In order to maintain a desired data accuracy for the transmitted data, we define the weight of a reading as follows:

**Definition 2 (Reading's weight, $wgt(r_t)$)** The weight of a reading $r_t$ is defined as the number of equal or similar (according to *LocTmp* function) consecutive readings, $r_{t-1}$, previously collected in the same period.

Therefore, after searching local temporal correlation, *LocTmp* allows $S_i$ to transform the initial vector of readings, $R_i$, to a set of readings, $R'_i$, associated to their corresponding weights as follows: $R'_i$=$\{(r'_1, wgt(r'_1)),\ (r'_2, wgt(r'_2)), \ldots, (r'_k, wgt(r'_k))\}$, where $k \le \mathcal{T}$.

Based on the set $R'_i$, we provide the following definitions:

**Definition 3 (Cardinality of the set $R'_i$, $|R'_i|$)** The cardinality of the set $R'_i$ is equal to the number of elements in $R'_i$, i.e. $|R'_i| = k$.

**Definition 4 (Weighted Cardinality of the set $R'_i$, $wgt_c(R'_i)$)** The weighted cardinality of the set $R'_i$ is equal to the sum of all reading weights in $R'_i$ as follows: $wgt_c(R'_i) = \sum_{j=1}^{|R'_i|} wgt(r'_j) = \mathcal{T}$, where $r'_j \in R'_i$.

### 3.2 Spatial Correlation Between Sensors

In WSNs, data sensed by the sensor nodes are spatially correlated due to their densely deployment. Hence, it is important to exploit the spatial correlation of data in sensor network in order to reduce the energy consumption in sensors, while conserving the integrity of these data.

Mostly, a sensor node $S_i$ is represented by its position ($x_i$,$y_i$), its sensing range ($S_r$) and its transmission range ($T_r$). In this paper, we assume that all sensor nodes has the same sensing and transmission range. Then, we use the Euclidean distance ($E_g$) to calculate the geographical distance between two nodes $S_i$ and $S_j$ as follows:

$$E_g(S_i, S_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

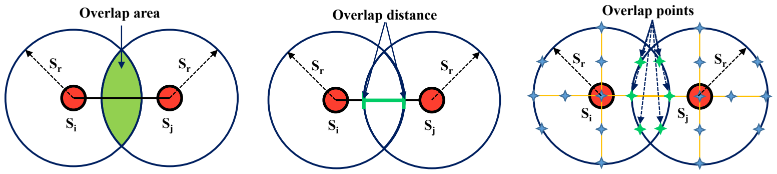
After that, we define the neighboring nodes of $S_i$:

**Definition 5** (neighbor): $S_j$ is a neighbor node of $S_i$ if the Euclidean distance between $S_i$ and $S_j$ is less than the twice of sensing range as follows:

$$E_g(S_i, S_j) \leq 2 \times S_r$$

Finally, we assume that $V_i$ is the set of neighbors of $S_i$.

Generally, there are three main categories to search the spatial correlation between neighboring sensors. The first category, such proposed in [1,2], exploits the overlap area between two sensor nodes (Fig. 1(a)). The second category, such in [3], calculates the spatial correlation based on the distance overlap between the sensors (Fig. 1(b)). The last category, such as [4], defines a number of primary points in the circle disk of the sensing range, then it calculates the number of points in the common area between the two sensors (Fig. 1(c)).



(a) Overlap area searching. (b) Overlap distance searching. (c) Overlap points searching.

**Fig. 1** Spatial correlation techniques between two sensors.

In this paper, we focus on the second category of spatial correlation which is simple and more flexible compared to other categories. However, in order to make the constraint for the spatial correlation between sensors more difficult to satisfy, we define a spatial correlation threshold called $C_{sp}$. Therefore, the spatial correlation between two sensors is defined as follows:

**Definition 6** (Spatial correlation between two sensors): Two given sensors $S_i$ and $S_j$, where $S_j$ is a neighbor node of $S_i$ (i.e. $S_j \in V_i$), are spatially correlated if and only if:

$$E_g(S_i, S_j) \leq C_{sp} \tag{1}$$

where $C_{sp}$ is a threshold determined by the application and it takes values in $[0, 2 \times S_r]$. Then, we assume that $V_i'$ is the set of all spatially correlated nodes with $S_i$.

Based on the definition 6, we can dynamically change the threshold $C_{sp}$ depending on the criticality of the monitored environmental; if the phenomenon is critical, the decision makers can decrease $C_{sp}$ in order to decrease the number of spatially correlated nodes for each node, i.e. $|V_i'|$ decreases; else, the decision makers can increase the threshold $C_{sp}$ when the phenomenon is less critical.

## 3.3 Temporal Correlation Between Sensors

In addition to the local temporal correlation in each sensor, the readings collected by nearby sensor nodes can be also temporally correlated. The temporal correlation among sensor nodes is to find out the sensors that collect similar readings at the same period time. Similarity metrics, such as Euclidean and Cosine distances, is one of the methods which can be used to identify sensor nodes that are temporally correlated. These metrics are generally used at the CHs level. Once high temporal correlation between two sensors is found, the CH should schedule these sensors in order to remove the redundancy in the network. In this paper, we focus on the Euclidean distance which is widely used in various domains.

In mathematics, the Euclidean distance is the straight line distance between two vectors of data. Let us first consider two data sets $R_i'$ and $R_j'$ generated by the two sensor nodes $S_i$ and $S_j$ respectively in the same period $p$. Then, in order to compute the Euclidean distance between $R_i'$ and $R_j'$, we must retransform the set $R_i'$ (resp. $R_j'$) to a vector as follows:

$$R_i' = [\ \underbrace{r_1', \ldots, r_1'}_{wgt(r_1')\text{ times}},\ \underbrace{r_2', \ldots, r_2'}_{wgt(r_2')\text{ times}}, \ldots,\ \underbrace{r_k', \ldots, r_k'}_{wgt(r_k')\text{ times}}\ ]$$

where $|R_i'| = |R_j'| = \mathcal{T}$.

Finally, we can calculate the Euclidean distance between the two vectors $R_i'$ and $R_j'$ based on the following equation:

$$E_d(R_i', R_j') = \sqrt{\sum_{k=1}^{\mathcal{T}} (r_{i_k}' - r_{j_k}')^2} \text{ where } r_{i_k}' \in R_i' \text{ and } r_{j_k}' \in R_j'$$

### *3.3.1 Distance Normalization*

The normalization of data is an essential process when using the distance functions. The objective of the normalization process is to scale all vectors of data to have the same variation then, to perform an exact comparison among these vectors. In this paper, we use Gaussian normalization to normalize data generated by the sensors. First, we calculate the Euclidean distance for each pair of data vectors in the network:

$$\mathbb{E}_d = \{E_d(R'_1, R'_2), E_d(R'_1, R'_3), \ldots, E_d(R'_{N-1}, R'_N)\}$$

where $N$ is the total number of sensors. Then, we can apply the Gaussian normalization using the following formula:

$$E'_d(R'_i, R'_j) = \frac{E_d(R'_i, R'_j) - \overline{Y}}{6 \times \sigma} + \frac{1}{2} \tag{2}$$

where $\overline{Y}$ is the mean of all distances and $\sigma$ is the standard deviation of pairwise distance over all data.

Thus, $R'_i$ and $R'_j$ are said to be redundant if $E'_d(R'_i, R'_j) \leq C_{tp}$, where $C_{tp}$ is a user defined threshold for the temporal correlation.

### 3.4 Spatial-Temporal Correlation Between Sensors

In PSNs, the spatial-temporal correlation happens when two nodes that are close geographically take similar readings in a period [5]. In this section, our objective is to search all pairs of sensors that are spatially-temporally correlated then to switch, in a later time, some sensors to the sleep mode in order to reduce redundant sensing and communication.

Based on equations 1 and 2, we say that two sensors $S_i$ and $S_j$, collecting the set of readings $R'_i$ and $R'_j$ respectively, are spatially-temporally correlated at the period $p$ if and only if:

$$E_g(S_i, S_j) \times E'_d(R'_i, R'_j) \leq C_{sptp} \tag{3}$$

where $C_{sptp} = C_{sp} \times C_{tp}$ and $E_g(S_i, S_j) \leq C_{sp}$ and $E'_d(R'_i, R'_j) \leq C_{tp}$.

Algorithm 1 describes our technique to find pairs of sensors that are spatially-temporally correlated. The CH searches which neighbors of each sensor $S_i$ are spatially (line 6) and temporally (line 8) correlated with $S_i$.

### 3.5 Selection of Thresholds

Obviously, the efficiency of our technique is highly related to the selection of thresholds $\delta$, $C_{sp}$ and $C_{tp}$. In a realistic application, these thresholds allow us to define an appropriate level of service. For instance, increasing the values of such thresholds leads to decrease the accuracy of the sent information and reduces more the size of data sent. On the other hand, the lowest the values of these thresholds are, the better relevant decisions and analysis could be made but the *LocTmp* function will not be efficient in reducing the amount of sent data. Therefore, selecting the appropriate values of thresholds is very essential in our technique. Indeed, we believe that these values should be determined by the decision makers or experts depending on the application requirements. For instance, in disaster monitoring applications, like volcano or seismic, thresholds must be lower than weather monitoring applications. Therefore, these parameters are based on the application criticality and the studied phenomenon. On the other hand, these parameters can also be adapted online and using the context aware sensing.

**Algorithm 1** Spatial-Temporal Correlation Algorithm.

```
Require: Set of sensors: $\mathbb{S} = \{S_1, S_2...S_N\}$, Set of their readings sets: $\mathbb{R} = \{R'_1, R'_2...R'_N\}$, $C_{sp}$, $C_{tp}$.
Ensure: All pairs of sensors ($S_i$,$S_j$) that are spatially-temporally correlated.
L ← ∅
for each sensor $S_i \in \mathbb{S}$ do
  for each sensor $S_j \in \mathbb{S}$ such that $S_j \neq S_i$ do
    compute $E_g(S_i, S_j)$
    if $(E_g(S_i, S_j) \leq 2 \times S_r)$ then
      if $(E_g(S_i, S_j) \leq C_{sp})$ then
        compute $E'_d(R'_i, R'_j)$
        if $(E'_d(R'_i, R'_j) \leq C_{tp})$ then
          $L \leftarrow L \cup \{(S_i, S_j)\}$
        end if
      end if
    end if
  end for
end for
```

## 4 Sleep Scheduling Strategies

After searching all pairs of spatially-temporally correlated sensors into a cluster, we propose, in this section, two scheduling strategies that allow sensors to work alternatively. The first strategy is based on the set cover problem while the second takes into account the correlation degree and the residual energy of sensors when searching the set of active sensors. In each strategy, a set of sensor nodes is selected, based on some criteria, to collect data in the network while switching other sensors into sleep mode.

### 4.1 Set Cover (SC) Strategy

The first strategy for scheduling sensor nodes is based on the Set Cover (SC) problem. In general, the SC problem consists in organizing sensor nodes into mutually exclusive subsets that are activated successively, where each subset ensures the coverage of the area of interest. Some real-world applications of SC problem include railway and airline crew scheduling, network discovery and phasor measurement unit placement [6]. In our case of PSNs, we apply the SC over the set of correlated sensors in order to divide all sensors into disjoint sensor subsets while every subset being able to completely cover the whole area of interest.

The SC problem can be formally defined in this paper as follows:

Given a set of $N$ sensors $\mathbb{S} = \{S_1, S_2, \ldots, S_N\}$ and the list $\mathbb{L} = \{(S_i, S_j)/ E_g(S_i, S_j) \times E'_d(R'_i, R'_j) \leq C_{sptp}\}$ of all pairwise spatially-temporally correlated sensors. The goal of SC is to find the list $\mathbb{D}$ which contains all the subsets $\mathbb{X} \subseteq \mathbb{S}$, where each $\mathbb{X}$ will cover, in terms of spatial-temporal correlation, all the sensors in $\mathbb{S}$. In other words, organizing sensor nodes into mutually exclusive subsets $\mathbb{X}$ where $\mathbb{X}$ ensures the coverage of the whole area of interest. Then, these subsets are activated successively.

Theoretically, the SC can be formulated as a binary integer programming problem as follows:

$$\begin{aligned} &\textit{Minimize} \sum_{j=1}^{N} S_j \\ &\textit{Subject to} \sum_{j=1}^{N} a_{ij} S_j \geq 1, \quad i = \{1, \ldots, N\} \\ &\qquad\qquad S_j \in \{0,1\},\, j = \{1, \ldots, N\} \end{aligned}$$

Our startegy operates into rounds, where each round is composed of $|\mathbb{D}|$ successive periods where $|\mathbb{D}|$ is the number of disjoint subsets found after applying the set cover technique. At each period only one subset is activated while the remaining nodes go to sleep mode. After each round, where all the subsets have been activated for one period and successively, the list of subsets will be updated based on the new spatio-temporal correlation between nodes.

**Illustrative example:** we consider a set of 6 sensors: $\mathbb{S} = \{S_1, S_2, S_3, S_4, S_5, S_6\}$, with the list of spatially-temporally correlated sensors: $\mathbb{L} = \{(S_1, S_2), (S_1, S_3), (S_1, S_4), (S_1, S_5), (S_2, S_6), (S_3, S_4), (S_3, S_6), (S_4, S_5)\}$. This leads to the following mathematically formulation of SC problem:

$$\begin{array}{rl} \textit{Minimize:} & S_1 + S_2 + S_3 + S_4 + S_5 + S_6 \\ \textit{Subject to:} & S_1 + S_2 + S_3 + S_4 + S_5 \qquad\;\; \geq 1 \\ & S_1 + S_2 \qquad\qquad\qquad\;\; + S_6 \geq 1 \\ & S_1 \qquad\;\; + S_3 + S_4 \qquad\; + S_6 \geq 1 \\ & S_1 \qquad\;\; + S_3 + S_4 + S_5 \qquad\;\; \geq 1 \\ & S_1 \qquad\qquad\;\; + S_4 + S_5 \qquad\;\; \geq 1 \\ & \qquad\; S_2 + S_3 \qquad\qquad\; + S_6 \geq 1 \end{array}$$

By applying the SC problem [7], we can find two feasible scheduling where evey sensor $S_i$ equals to 1 (in active mode) in at most one scheduling [1]:

- *Scheduling 1*: $S_1 = S_6 = 1$ and $S_2 = S_3 = S_4 = S_5 = 0$.

[1] The values 0 and 1 of the sensor mean that it can be on sleep or active mode respectively.

- *Scheduling 2*: $S_2 = S_4 = 1$ and $S_1 = S_3 = S_5 = S_6 = 0$.

Therefore, we can divide $\mathbb{S}$ into two disjoint subsets of sensors as follows: $\mathbb{L} = \{L_1 = \{S_1, S_6\}, L_2 = \{S_2, S_4\}\}$. Consequently, the current round will consist, by applying our SC strategy, in three periods where in each period the sensors in one subset $L_i$ wil be active. Otherwise, all the sensors are active in the first period. Fig. 2 shows the active sensors in each period in the $Round1$.

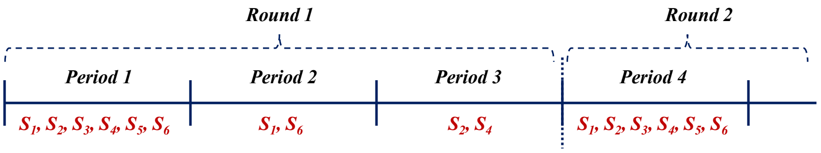


**Fig. 2** Active sensors during periods in the round.

At the begining of each perio the CH is reponsible to send notifications to activate or switch off the nodes. Finally, we note that the number of periods during a round can be fixed based on the temporal variations and observations.

### 4.2 Correlation Degree and Residual Energy (CDRE) Strategy

In this section, we propose a second strategy for scheduling in sensor networks based on spatio-temporal correlation while taking into account the residual energy. We call this strategy Correlation Degree and Residual Energy (CDRE) strategy.

The CDRE strategy also operates into rounds where each round is fixed to two periods. In the first period of each round, the CH searches the set of sensors to be active in the second period, based on the CDRE strategy. For this we consider the following notations:

- The list of spatially-temporally correlated sensors with their correlation degrees: $\mathbb{L} = \{\big((S_i, S_j), C_{ij}(S_i, S_j)\big)\ such\,that\ C_{ij}(S_i, S_j) = E_g(S_i, S_j) \times E'_d(R'_i, R'_j)\ and\ C_{ij}(S_i, S_j) < C_{sptp}\}$.
- $E_{r_i}$, the residual energy of the sensor $S_i$.

The CDRE strategy can be expressed using the Algorithm 2. First, we order the pairs of sensors according to the increasing order of their spatial-temporal correlation degree. Therefore, we start with the pair of sensors having the highest correlation degree (line 2). Then for the first period of each round, for each pair of nodes ($S_i$,$S_j$), we selct the sensor which has the higher residual energy to be an active sensor in the second period, and the second one will go to sleep mode (lines 4-14). The objective of this part of the algorithm is to select from **correlated nodes** the nodes to be activated in the next period and having the highest remaining energy. In a second step, for the nodes that do not have any correlation with other sensors, they must be in active mode for next rounds (lines 15-19). After that, all the nodes in set $\mathbb{E}$ will send their readings of the first period to the sink and be activated for the next period, while the others do not sen d their readings and go to sleep node for the next period. The objective here is to remove the redundancy among the data sent to the sink in the first period, contrarily to the SC strategy which sends all the readings sets.

**Algorithm 2** CDRE Strategy Algorithm.

```
Require: Set of sensors: $\mathbb{S} = \{S_1, S_2 ... S_N\}$, Set of their readings sets: $\mathbb{R} = \{R'_1, R'_2 ... R'_N\}$, List of correlated sensors: $\mathbb{L}$, round.
Ensure: void.
 1: $\mathbb{E} \leftarrow \emptyset$
 2: $\mathbb{L} \leftarrow$ sort($\mathbb{L}$); // $\mathbb{L}$ is sorted in increasing order of the sensors correlation degree
 3: if First round then
 4:   while $\mathbb{L}$ is not empty do
 5:     $((S_i,S_j), C_{ij}(S_i,S_j))$ is the first element in $\mathbb{L}$
 6:     consider $E_{r_i} > E_{r_j}$ [2]
 7:     if $S_j \notin \mathbb{E}$ then
 8:       if $S_i \notin \mathbb{E}$ then
 9:         $\mathbb{E} \leftarrow \mathbb{E} \cup \{S_i\}$
10:       end if
11:       remove all elements $(S_y,S_j)$ and $(S_j,S_y)$ from $\mathbb{L}$ such that $S_y \neq S_i$
12:     end if
13:     remove $((S_i,S_j), C_{ij}(S_i,S_j))$ from $\mathbb{L}$
14:   end while
15:   for each $S_i \in \mathbb{S}$ do
16:     if $S_i$ has no correlated $S_j$ in $\mathbb{E}$ then
17:       $\mathbb{E} \leftarrow \mathbb{E} \cup \{S_i\}$
18:     end if
19:   end for
20:   for each $S_k \in \mathbb{E}$ do
21:     $Send\_to\_Sink(R'_k)$
22:   end for
23:   for each $S_k \in \mathbb{S}$ such that $S_k \notin \mathbb{E}$ do
24:     $Sleep\_message\_to(S_k)$
25:   end for
26: else
27:   for each $S_k \in \mathbb{S}$ do
28:     if $Sleep\_message\_to(S_k)$ then
29:       $S_k$ enter in sleep mode
30:     else
31:       Active mode
32:     end if
```

[2] We consider that $S_i$ has more energy than $S_j$. In the opposite case, we do the same thing by inverting the indices.

```
    end for
end if
```

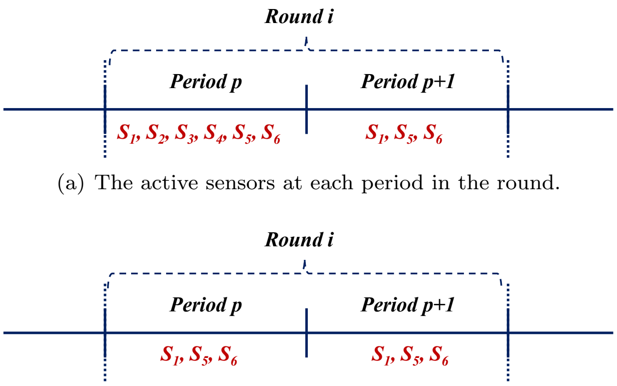
**Illustrative example:** Recall the sensors $S_1$ to $S_6$ in the set $\mathbb{S}$ in the example above with the ordered list of correlated pairs degree as follows: $\mathbb{L} = \{((S_1, S_3), C_{1,3} = 0.5), ((S_4, S_5), C_{4,5} = 0.9), ((S_3, S_4), C_{3,4} = 1.8), ((S_3, S_6), C_{3,6} = 2.1), ((S_1, S_5), C_{1,5} = 2.5), ((S_1, S_2), C_{1,2} = 2.9), ((S_1, S_4), C_{1,4} = 3.3), ((S_2, S_6), C_{2,6} = 3.5)\}$. Then, we consider that the sensors have the following residual energies at the beginning of the round $i$: $E_{r_1} = 8.1\,mJ$, $E_{r_2} = 8.3\,mJ$, $E_{r_3} = 7.6\,mJ$, $E_{r_4} = 6.9\,mJ$, $E_{r_5} = 7.8\,mJ$, $E_{r_6} = 7.9\,mJ$.

- *Step 1*: We start by the correlated pair ($S_1$,$S_3$). Since $S_1$ has more energy than $S_3$, $S_3$ will be switched to the sleep mode in the next period while $S_1$ will be added to the list of active sensors: $\mathbb{E} = \{S_1\}$. Then, we remove the pairs of sensors that contains $S_3$, i.e. ($S_3$,$S_4$) and ($S_3$,$S_6$). The remaining elements in $\mathbb{L} = \{((S_4, S_5), C_{4,5} = 0.9), ((S_1, S_5), C_{1,5} = 2.5), ((S_1, S_2), C_{2,6} = 2.9), ((S_1, S_4), C_{1,4} = 3.3), ((S_2, S_6), C_{1,2} = 3.5)\}$.
- *Step 2*: The first element in $\mathbb{L}$, i.e. ($S_4$,$S_5$), is treated similarly to ($S_1$,$S_3$): we add $S_5$ to the list of active sensors and we switch $S_4$ to the sleep mode then, we remove all elements that contains $S_4$ from $\mathbb{L}$. Therefore, $\mathbb{E} = \{S_1, S_5\}$ and $\mathbb{L} = \{((S_1, S_5), C_{1,5} = 2.5), ((S_1, S_2), C_{2,6} = 2.9), ((S_2, S_6), C_{1,2} = 3.5)\}$.
- *Step 3*: Since $S_1$ and $S_5$ are both in $\mathbb{E}$, we remove the pair ($S_1$,$S_5$) from $\mathbb{L}$ because they will be both in active mode. Therefore, $\mathbb{L} = \{((S_1, S_2), C_{2,6} = 2.9), ((S_2, S_6), C_{1,2} = 3.5)\}$.
- *Step 4*: Independent from residual energies of the sensors $S_1$ and $S_2$, $S_2$ should be switched to the sleep mode because $S_1$ will be considered as active sensor in the next period. Hence, we remove elements from $\mathbb{L}$ that contains $S_2$: $\mathbb{L} = \{\}$.
- *Step 5*: We add the sensor $S_6$ to the set $\mathbb{E}$ since it does not have any correlated sensor in $\mathbb{E}$.

Finally, the set of active sensors and the readings sets sent from the CH to the sink, at each period in the round $i$, are shown in Fig. 3(a) and Fig. 3(b) respectively. In the first period, all the sensors are active while the CH only sends the sets which are not redundant, i.e. sensors $\in \mathbb{E}$. On the other hand, all readings sets coming from the active sensors will be send to the sink in the second period in the round.

## 5 Simulation Results

In this section, we look at the performance of our spatial-temporal correlation mechanism under the two proposed scheduling strategies. In our simulations, we implemented both strategies based on a Java based simulator. We ran the simulator based on real sensor readings of temperature collected by 46 sensors and provided by the Intel Berkeley Research lab [8]. Fig. 4 shows a map of



(a) The active sensors at each period in the round.

(b) The readings sets sent from the CH to the sink at each period in the round.

**Fig. 3** Illustrative example of the actives sensors and their readings sets during a round.

the placement of sensors in the lab. The sensors are Mica2Dot with weather boards that collect temperature values once every 31 seconds. We assume that the network is divided into two clusters, which have $CH_1$ and $CH_2$ as cluster-heads respectively, where the sensors send their data periodically to their appropriate cluster-head. We compared our results to those of PPMC proposed in [9]. Table 1 shows the parameters used in our simulations.

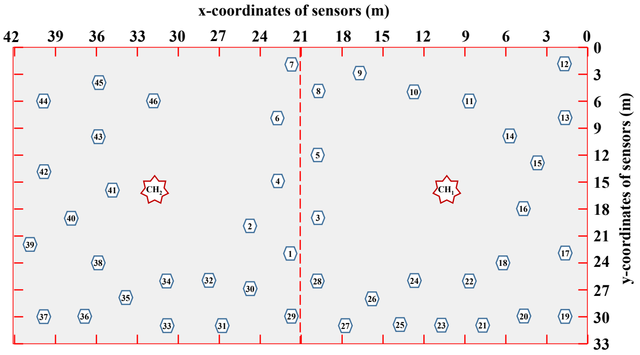


**Fig. 4** Distribution of sensors and CHs in the Intel Laboratory.

### 5.1 Performance Evaluation at Sensor Node

In this section, we evaluate the performance of our mechanism with SC and CDRE strategies at the sensor nodes level, compared to the PPMC [9] and

| Parameter | Description | Value |
|---|---|---|
| $\mathcal{T}$ | Number of readings per period | 200, 500, 1000 |
| $S_r$ | Sensor sensing range | 5, 10, 15, 20 meters |
| $\delta$ | *LocTmp* similarity threshold | 0.03, 0.05, 0.07, 0.1 |
| $C_{sp}$ | Spatial correlation threshold | $2 \times S_r$, $\frac{5 \times S_r}{3}$, $\frac{4 \times S_r}{3}$, $S_r$ |
| $C_{tp}$ | Temporal correlation threshold | 0.35, 0.4, 0.45, 0.5 |
| $n$ | Number of sensors in each cluster | 23 |
| $t_{PPMC}$ | Similarity threshold used for PPMC | 0.9 |
| $E_i$ | Initial energy for each sensor | 10 mJ |

**Table 1** Simulation environment.

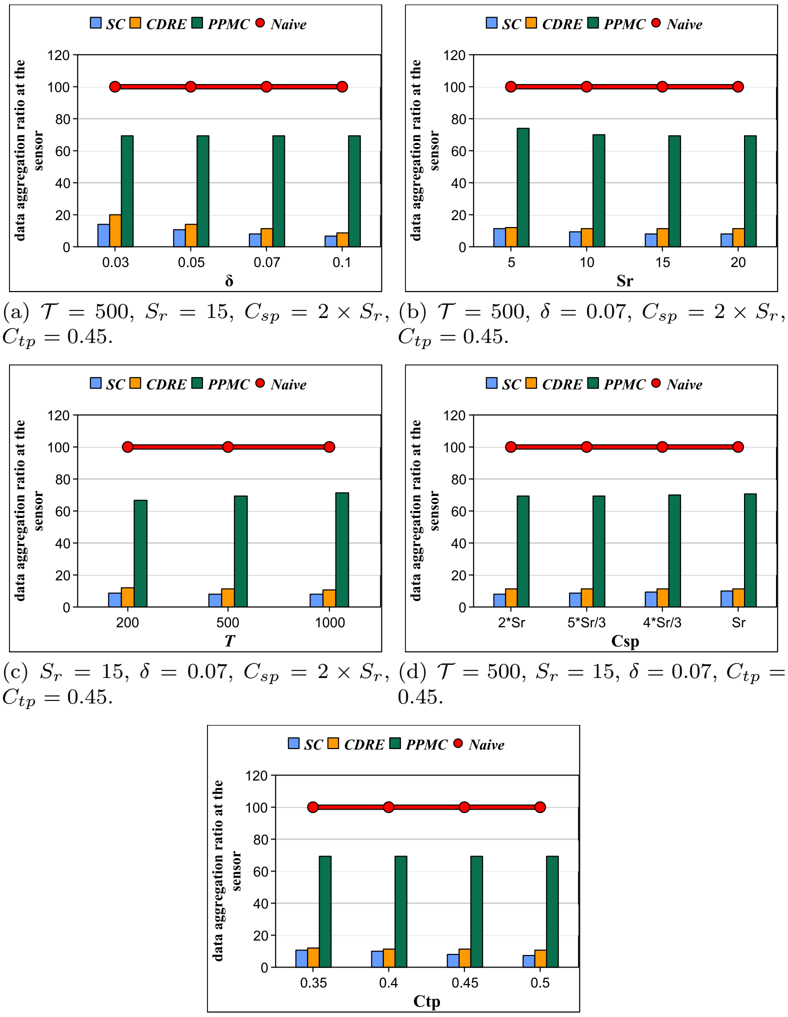
the naïve method (i.e. the classic method where all readings collected by the sensors are sent to the sink without any processing). Indeed, we present only the results for one cluster, i.e. the second cluster with $CH_2$, in the case when the performance metric gives similar results for the two clusters. In addition, the results for each metric shown in the next figures represent the average of all sensors in each cluster. At this level, we have considered three performance metrics.

### *5.1.1 Percentage of Data Readings Sent from Each Sensor to its CH*

In this section, our objective is to show how our mechanism can decrease the data readings collected by each sensor node and then sent to the $CH_2$. Fig. 5 shows the percentage of data collected, then sent, by each sensor node when varying one parameter each time and fixing the others as shown in Fig. 5 (a to e). The obtained results show that PPMC can reduce from 25% to 33% the data sent to the $CH_2$, while, our mechanism with SC and CDRE strategies can reduce, respectively, up to 93% and 90% the data sent, compared to the naïve technique which always sends all data collected (i.e. 100%) . This means that our approach can effectively eliminate the redundancy in data collection while searching all sensors that generate spatially-temporally correlated data. Furthermore, we can also notice that the SC strategy gives better results, in terms of reducing the data sent by each sensor, than CDRE strategy in all cases.

Several observations can be made based on the results in Fig. 5:

- By increasing the threshold $\delta$ in Fig. 5(a), each sensor can reduce, using the SC and CDRE strategies, up to 90% and 87% respectively the readings sent to the $CH_2$ compared to PPMC. These results are obtained due to the fact that *LocTmp* will find more similar readings when $\delta$ increases.
- By increasing its sensing range as shown in Fig. 5(b), each sensor sends less readings to the $CH_2$ using the two proposed strategies. For instance, when $S_r$ increases from 5 to 20, a sensor node decreases its readings sent from 11.4% to 8.4% using SC strategy. This happens because, when $S_r$ increases, each sensor will have more neighboring, thus correlated, sen-



(a) $\mathcal{T} = 500$, $S_r = 15$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

(b) $\mathcal{T} = 500$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

(c) $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

(d) $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{tp} = 0.45$.

(e) $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$.

**Fig. 5** Percentage of data readings sent from each sensor to the $CH_2$.

sors. Consequently, more sensors will be switched to the sleep mode, thus, decreasing the percentage of the collected and sent readings.

- By increasing $\mathcal{T}$ from 200 to 1000 in Fig. 5(c), the percentage of readings sent decreases using SC and CDRE strategies while it increases using PPMC. The reason for this is that the $\delta$ threshold used in *LocTmp* in the two strategies which finds, then eliminates, more redundancy when $\mathcal{T}$ increases. Contrarily, PPMC does not apply any processing on the collected data which increases the readings sent to the $CH_2$ when $\mathcal{T}$ increases.
- By decreasing the spatial correlation threshold ($C_{sp}$) in Fig. 5(d), the percentage of readings sent to the $CH_2$ increases in the three approach, i.e.

SC, CDRE and PPMC. This result is logical since we make the constraint for the neighboring sensors more difficult to satisfy (see definition 6). Consequently, the number of active sensors in each period will tend to increase. It is also important to notice that our strategies reduce, in all cases, the readings sent to $CH_2$ as compared to those sent using PPMC.

- By increasing the temporal correlation threshold ($C_{tp}$) in Fig. 5(e), SC and CDRE strategies allow each sensor node to decrease its data readings sent to the $CH_2$. This is because, the Euclidean distance between sets of readings will be more easily satisfied therefore, more sensors will be switched to the sleep mode.

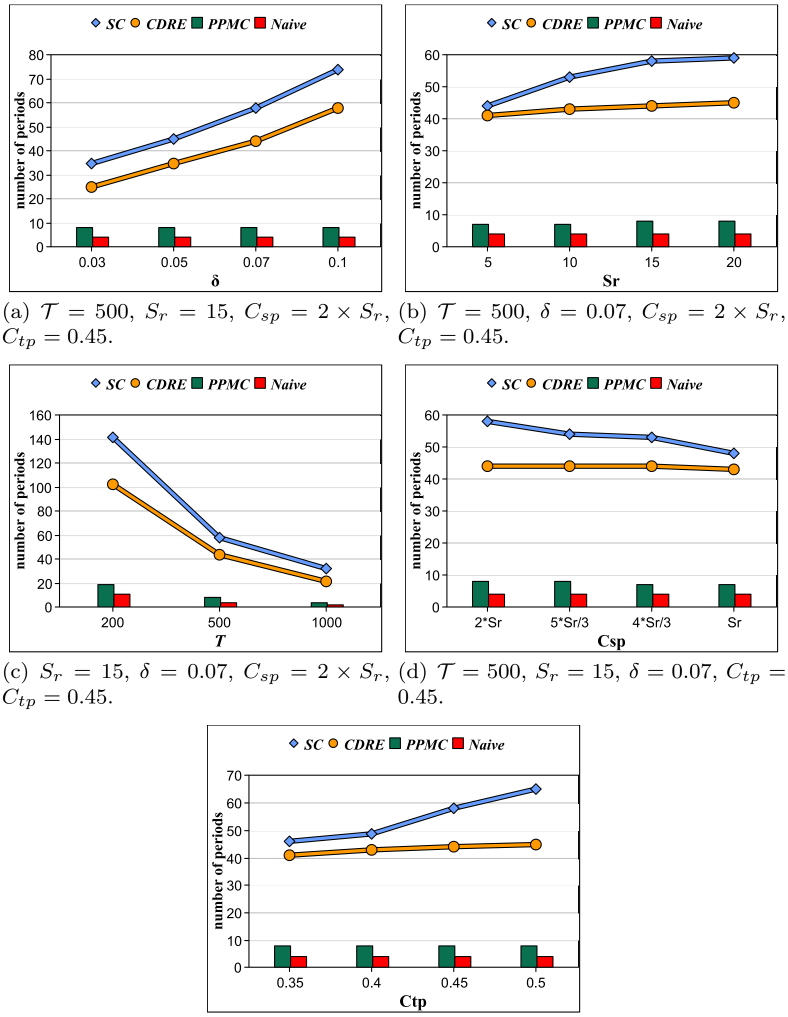
### *5.1.2 Lifetime of the Sensor Node*

In this section, our objective is to study the energy consumption at the sensor nodes level. Therefore, we fixed the initial energy for all sensor nodes to $E_i$. Then, we applied our strategies, PPMC and Naïve approaches while varying, each times, one parameter and fixing the others as done in Fig. 5. Fig. 6 shows the lifetime of each sensor in terms of number of periods in which the sensor is operational, i.e. its residual energy is positive. The obtained results show clearly that our mechanism, with the proposed strategies, can efficiently reduce the energy consumption of the sensor and extend its lifetime. This is because, our mechanism eliminates the redundancy among collected data and reduces the readings sent to the CH (see Fig. 5). Although the PPMC can extend, in the best case, the lifetime of a sensor by two times compared to the Naïve approach, our strategies significantly outperform the results of PPMC. We can also notice that, the SC strategy gives better results in terms of keeping the sensor node operating for long time compared to CDRE strategy.

In WSNs, the energy consumption in the sensor node is proportionally to the amount of data sent by the sensor. Consequently, when the sensor sends more data the CH, its energy will be more consumed and vice versa. Hence, the observations made based on the results of Fig. 5 can be similarly made for the energy consumption in the sensor in the Fig. 6. Table 2 shows how many times the sensor node can extend, using our strategies, its lifetime in the worst and the best cases by fixing one parameter as shown in Fig. 6(a to e), compared to PPMC and Naïve approaches.

| *Our Strategy* | *Compared Strategy* | $\delta$ | $S_r$ | $\mathcal{T}$ | $C_{sp}$ | $C_{tp}$ |
|---|---|---|---|---|---|---|
| *SC* | *PPMC* | 4 → 9 | 6 → 7 | 7 → 8 | 6 → 7 | 5 → 8 |
| | *Naïve* | 8 → 18 | 11 → 14 | 12 → 16 | 12 → 14 | 11 → 16 |
| *CDRE* | *PPMC* | 3 → 7 | 5 → 6 | 5 → 6 | 5 → 6 | 4 → 6 |
| | *Naïve* | 6 → 14 | 10 → 11 | 9 → 11 | 10 → 11 | 10 → 11 |

**Table 2** Lifetime comparisons between our strategies, PPMC and Naïve approaches. (Worst case → Best case).



(a) $\mathcal{T} = 500$, $S_r = 15$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

(b) $\mathcal{T} = 500$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

(c) $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

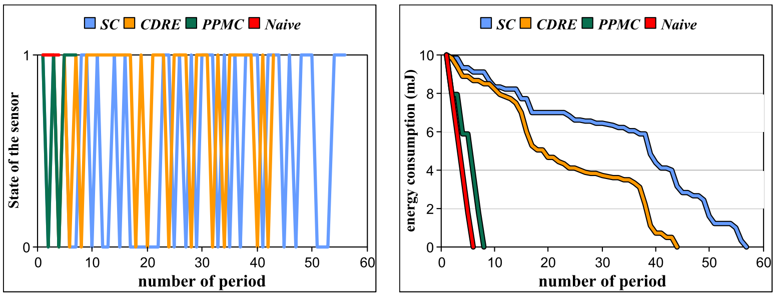
(d) $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{tp} = 0.45$.

(e) $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$.

**Fig. 6** Lifetime of each sensor in the second cluster ($\mathrm{CH}_2$).

### *5.1.3 Variation of the State and the Energy of the Sensor during Periods*

In this section, we show an example of a sensor activitiy variation during periods by applying our strategies, PPMC and Naïve approaches. We take the sensor that has an *id* equals to 35 located in the second cluster, then we study the variation of its state, i.e. active or sleep, and its residual energy during the periods, for some fixed parameters shown in Fig. 7. Based on the results of Fig. 7(a), we can see that the state of the sensor varies, when applying our strategies, from 1 (i.e. active mode) to 0 (i.e. sleep mode) during the periods more dynamically than with other techniques. Our strategies confirm also the efficient reduction of the redundancy between the sensors correlated to the sensor '35' by switching it to the sleep mode more often than with the other techniques. On the other hand, Fig. 7(b) shows how the residual energy of the sensor varies depending on the state of the sensor; if the sensor is in active mode, its residual energy decreases in order to collect the data and send it to the CH; else, it remains fixed until the next period. We can also observe that the residual energy of the sensor can remain fixed using the SC strategy in many successive periods, i.e. from periods 18 to 22 in Fig. 7(b) for example. This happens because the number of periods in each round changes dynamically using the SC strategy where the sensor can be active at most in two periods in a round.



(a) Variation of sensor state's. (b) Variation of sensor energy.

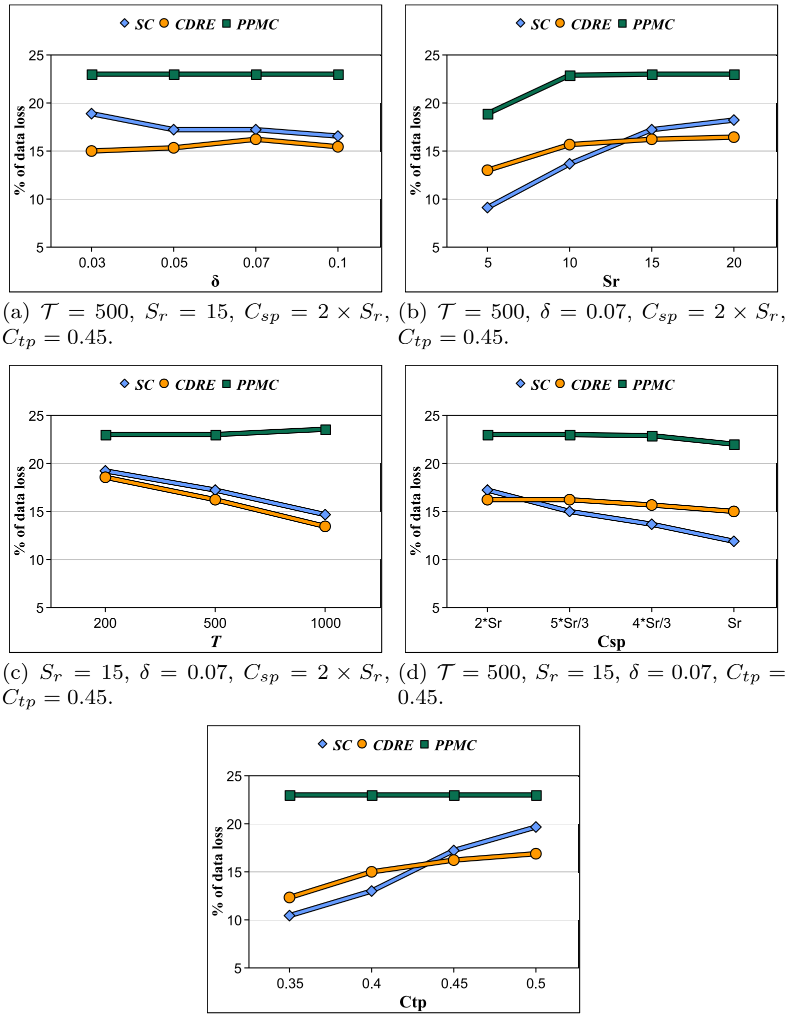
**Fig. 7** Variation of sensor activity during periods, Sensor id = 35, $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

## 5.2 Performance Evaluation at CH Nodes

In this section, we evaluate the performance of our approach at the CHs level. We have taken four performance metrics.

### *5.2.1 Data Accuracy*

Scheduling sensor nodes in the network without losing the integrity of the information is an important challenge for the WSN. Data accuracy usually represents the "data loss rate" measure. It is an evaluation of the readings taken by the sensor nodes whose values (or similar values) do not reach the sink. Fig. 8 shows the results of data accuracy for SC, CDRE and PPMC for different values of parameters considered in our simulation. We can observe that our strategies always give better results for data accuracy compared to PPMC. This is because, the Pearson coefficient used in PPMC calculates the distance between two data sets based on the summation of readings while the Euclidean distance, used in our strategies, calculates the distance between every two readings in the data sets. This makes the loss of data in our strategies less than that in PPMC. We can also notice that the results of data accuracy using CDRE strategy is better, in the most cases, than those obtained using SC strategy. The reason for that is the sensor sends, using the two strategies, at most two data sets in a round while the round in SC contains more periods than that in CDRE (see illustrative examples for SC and CDRE strategies).



(a) $\mathcal{T} = 500$, $S_r = 15$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

(b) $\mathcal{T} = 500$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

(c) $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

(d) $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{tp} = 0.45$.

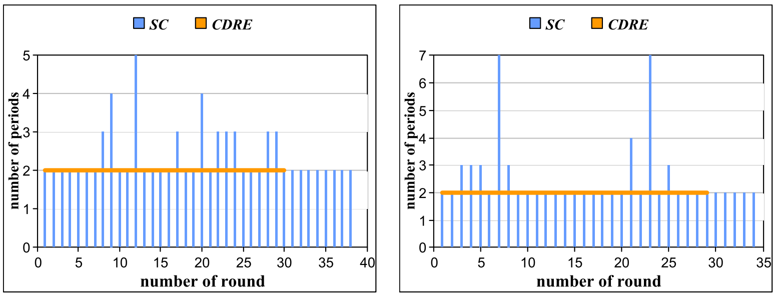
(e) $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$.

**Fig. 8** Data accuracy at the $CH_2$.

In general, the data accuracy dependent on the percentage of data sent by the sensors (see results in Fig. 5) and on the number of active sensors during the periods; when the data sent to the sink or the number of active sensors increase, the data accuracy increases. Therefore, the following observations can be made based on the results of Fig. 8: (1) data loss measure increases when the sensing range of the sensor ($S_r$) or the temporal correlation threshold ($C_{tp}$) increase (Fig. 8(b and e)). (2) the data accuracy increases when the number of collected readings during a period ($\mathcal{T}$) increases or the spatial correlation threshold ($C_{sp}$) decreases (Fig. 8(c and d)).

### *5.2.2 Variation of Periods Number during Rounds*

In this section, we show how the number of periods changes, using our proposed strategies, after each round for the two clusters in the network, for some fixed parameters. Using SC strategy, the CH calculates, at the beginning of each round, the maximum number of periods for the current round based on the set covering problem. Otherwise, the number of periods is always equal to 2 for each round using CDRE strategy. The obtained results of the two clusters, represented by their cluster-heads CH$_1$ and CH$_2$ respectively, are shown in Fig. 9(a) and Fig. 9(b) respectively. While each round always consists of two periods using CDRE, the number of periods dynamically varies in each round using SC as shown in the figures. We can also observe that: (1) the round can up to 7 periods using SC strategy. This reflect the high level of redundancy existing in the network where SC can efficiently eliminate this redundancy. (2) the sensors in the first cluster are more spatially-temporally correlated compared to those in the second cluster. This leads to extend, using the two strategies, the lifetime of the first cluster more than that of the second cluster.

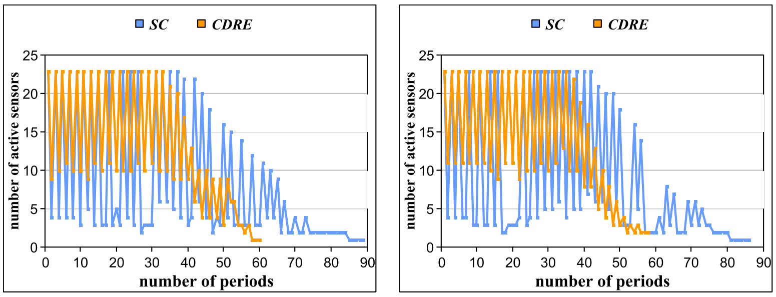


(a) CH$_1$ (b) CH$_2$

**Fig. 9** Variation of periods number in each round, $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2\times S_r$, $C_{tp} = 0.45$.

### *5.2.3 Variation of Active Sensors Number during Periods*

In this section, our main goal is to show how our strategies are able to schedule the activities of the sensor nodes for the two clusters. Fig. 10 shows the number of active sensors in each cluster and in each period using SC and CDRE strategies, for the fixed parameters shown in the figure. The number of active sensors can affect the lifetime of the network, the data latency and the coverage of the monitored area. As we can see, each strategy successfully schedules the sensor nodes in each cluster dynamically after each period according to its own scheduling mechanism. We can notice that, the SC strategy reduces the number of active sensors, in each cluster, at each period to the minimum while the CDRE strategy selects the set of active sensors that balance the energy distribution in each cluster. Consequently, the obtained results confirm the proper behavior of our strategies.



(a) $\mathrm{CH}_1$

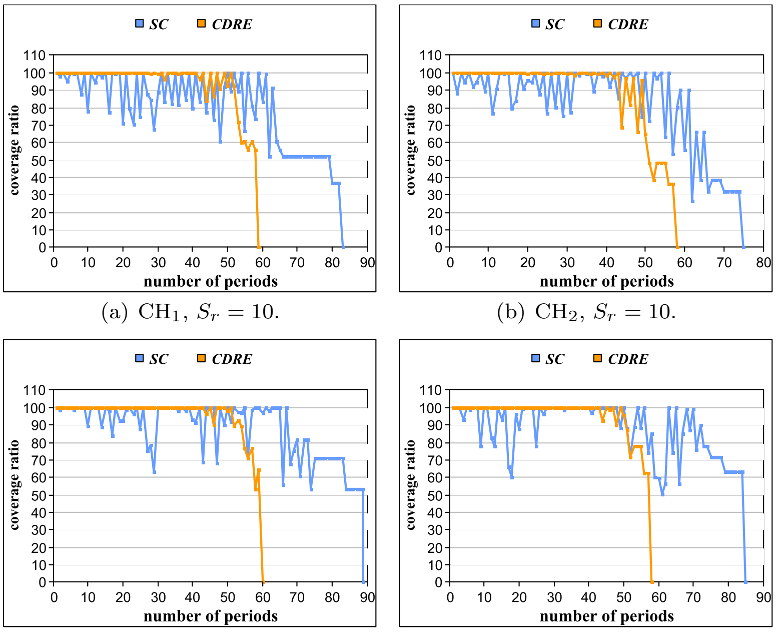
(b) $\mathrm{CH}_2$

**Fig. 10** Variation of active sensors number during each period, $\mathcal{T} = 500$, $S_r = 15$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

### *5.2.4 Coverage Variation during Periods*

Conserving the network energy while preserving the maximal coverage of the region of interest is an important challenge in WSNs. In Fig. 11, we show how much of the area of each cluster is covered after each period by applying our strategies. The sensing range of a sensor is varied from 10 in Fig. 11(a and b) to 15 in Fig. 11(c and d), while the other parameters remain fixed. The obtained results show that the two proposed strategies provide sufficient coverage for the clusters during each period. Therefore, we can consider that our mechanism with the two proposed strategies can efficiently extend the network lifetime while preserving the integrity of data and the coverage of the observed area.

Based on the results in Fig. 11, several observations can be made:



(a) $\text{CH}_1$, $S_r = 10$.

(b) $\text{CH}_2$, $S_r = 10$.

(c) $\text{CH}_1$, $S_r = 15$.

(d) $\text{CH}_2$, $S_r = 15$.

**Fig. 11** Coverage ratio for each cluster, $\mathcal{T} = 500$, $\delta = 0.07$, $C_{sp} = 2 \times S_r$, $C_{tp} = 0.45$.

- the CDRE strategy provide more coverage for the two clusters compared to SC strategy. This is because, the number of active sensors in each period using CDRE strategy is greater than that using SC strategy.
- the coverage ratio for each cluster increases when the sensor sensing range increases.

### 5.3 Further Discussions

In this section, we give further consideration to our proposed mechanism. We compare the obtained results for both strategies SC and CDRE. We give some directions to which strategy to choose and under which conditions and circumstances of the application.

From the sensor lifetime point of view, both strategies SC and CDRE significantly improve the lifetime of the sensor (Fig. 6). However, SC allows sensor to extend more its lifetime, from 7% to 45%, compared to CDRE. Therefore, if the application needs to conserve the energy and extend the network lifetime as long as possible, SC strategy is more suitable.

From the data accuracy point of view, CDRE can save, in the most cases, the integrity of the collected data more than SC. This is because the number of active sensors in each period using CDRE is greater than that in SC, which increases the accuracy of the data sent to the sink. Consequently, when the priority of the application is to ensure a high level of data accuracy, CDRE is more suitable.

From the coverage of the interest area point of view, CDRE can practically cover the whole monitored area during all periods of the network lifetime, while SC can ensure a satisfactory coverage, i.e. more than 70% in the most cases, of the network area. Hence, if the application does not permit flexibility regarding coverage of the network, CDRE is more suitable.

## 6 Conclusion & Future Work

In this paper, we proposed an efficient mechanism in order to search the spatial and temporal correlation between data collected by the sensors in a periodic sensor network. Then, in order to schedule sensors to work alternatively, we proposed two scheduling strategies in order to switch the sensors into sleep/active mode during the periods. The first strategy, called SC, is based on the set cover problem while the second strategy, called CDRE, takes into account the correlation degree and the residual energy of the sensors when scheduling the network. We demonstrated through simulation on real data readings the efficiency of our mechanism, under the two proposed strategies, in sensor networks in terms of extending network lifetime while conserving the quality of the collected data and the coverage of the monitored area.

As future work, we will adapt our proposed mechanism to take into consideration *reactive* periodic sensor network, where sensor nodes operate with different sampling rates. In periodic applications the dynamics of the monitored condition or process can slow down or speed up; the sensor node can further save energy by adapting its sampling rates to the changing dynamics of the condition or process.

## 7 Conflict of Interest

The authors declare that they have no conflict of interest.

## References

1. C. Liu and G. Cao, *Spatial-Temporal Coverage Optimization in Wireless Sensor Networks*, IEEE Trans. Mob. Comput., Vol. 10, No. 4, pp. 465-478, 2011.
2. K. Karuppasamy and V. Gunaraj, *Optimizing Sensing Quality with Coverage and Lifetime in Wireless Sensor Networks*, International Journal of Engineering Research & Technology, Vol. 2, No. 2, pp. 1-7, 2013.
3. M.-H. Tsai and Y.-M. Huang, *A Sub-Clustering Algorithm Based on Spatial Data Correlation for Energy Conservation in Wireless Sensor Networks*, Journal of Sensors, Vol. 14, No. 11, pp. 21858-21871, 2014.
4. A. Idrees, K. Deschinkel, M. Salomon and R. Couturier, *Coverage and Lifetime Optimization in Heterogeneous Energy Wireless Sensor Networks*, In ICN 2014, 13-th Int. Conf. on Networks, Nice, France, p. 49-54, 2014.

5. L.A. Villas, A. Boukerche, D.L. Guidoni, H.A.B.F. de Oliveira, R.B. de Araujo and A.A.F. Loureiro, *An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks*, Journal of Computer Communications, Vol. 36, No. 2013, pp. 1054-1066, 2013.
6. J. Xu, M.HF. Wen, V.OK Li and K.-C. Leung, *Optimal PMU placement for wide-area monitoring using chemical reaction optimization*, in Proc. IEEE Innovative Smart Grid Technologies Conference (ISGT), Washington DC, U.S., p. 1-6, 2013.
7. D. Jonhson, *Approximation Algorithms for Combinatorial Problem*, Journal of Computer and System Sciences, Vol. 9, pp. 256-278, 1974.
8. S. Madden, *Intel Berkeley Research lab*, http://db.csail.mit.edu/labdata/labdata.html, 2004.
9. S. Dhimal and K. Sharma, *Energy Conservation in Wireless Sensor Networks by Exploiting Inter-Node Data Similarity Metrics*, International Journal of Energy, Information and Communications, Vol. 6, No. 2, pp. 23-32, 2015.
10. N. Boopal, S. Gunasekaran and V. A. Mangai, *A Survey Of Spatiotemporal Data Compression In Wireless Sensor Networks*, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol. 4, No. 4, pp. 1182- 1185, 2015.
11. A.R. Pagar and D.C. Mehetre, *A Survey on Energy Efficient Sleep Scheduling in Wireless Sensor Network*, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 5, No. 1, pp. 557-562, 2015.
12. A. Makhoul, H. Harb and D. Laiymani, *Residual Energy-based Adaptive Data Collection Approach for Periodic Sensor Networks*, Ad Hoc Networks, Vol. 35, pp. 149-160, 2015.
13. H.K. Singh and J. Bharti, *A Novel Solution for Sleep Scheduler in Wireless Sensor Networks*, International Journal Of Advanced Smart Sensor Network Systems (IJASSN), Vol. 2, No. 1, pp. 13-19, 2012.
14. A. S. Bhosale, S.R. Khajure and M.S. Sharma, *Efficient Data Collection in Wireless Sensor Networks using Spatial Correlation Algorithm*, International Journal on Recent and Innovation Trends in Computing and Communication, Vol. 3, No. 2, pp. 418-423, 2015.
15. K. Liu, Y. Zhuang, Z. Wang and J. Ma, *Spatiotemporal Correlation Based Fault-Tolerant Event Detection in Wireless Sensor Networks*, International Journal of Distributed Sensor Networks, Vol. 2015, No. Article ID 643570, pp. 14 pages, 2015.
16. X. Piao, Y. Hu, Y. Sun, B. Yin and J. Gao, *Correlated Spatio-Temporal Data Collection in Wireless Sensor Networks Based on Low Rank Matrix Approximation and Optimized Node Sampling*, Sensors, Vol. 14, pp. 23137-23158, 2014.
17. F. Gielow, G. Jakllari, M. Nogueira and A. Santos, *Data similarity aware dynamic node clustering in wireless sensor networks*, Ad Hoc Networks, Vol. 24, pp. 29-45, 2015.
18. S. Chen, C. Zhao, M. Wu, Z. Sun and J. Jin, *Clustered Spatio-Temporal Compression Design for Wireless Sensor Networks*, 24th International Conference on Computer Communication and Networks (ICCCN), IEEE, p. 1-6, 2015.
19. L.A. Villas, A. Boukerche, D.L. Guidoni, H.A.B.F. de Oliveira, R.B. de Araujo and A.A.F. Loureiro, *A spatial correlation aware algorithm to perform efficient data collection in wireless sensor networks*, Ad Hoc Networks, Vol. 12, pp. 69-85, 2014.
20. B. Paczek and M. Berna, *Uncertainty-based information extraction in wireless sensor networks for control applications*, Ad Hoc Networks, Vol. 14, pp. 106-117, 2014.
21. D. Baum and CIO Information Matters, *Big Data, big opportunity*, http://www.oracle.com/us/c-central/cio-solutions/informationmatters/big-data-big-opportunity/index.html.
22. Bloomberg Business, *Sensor Networks Top Social Networks for Big Data*, http://www.bloomberg.com/bw/technology/content/-sep2010/tc20100914_284956.htm, 2010.
23. T. Wang, *Research on Data Aggregation Technology Based on Wireless Sensor Networks*, International Journal of Future Generation Communication and Networking, Vol. 9, No. 1, pp. 127-134, 2016.
24. H.-Y. Kim, *An energy-efficient load balancing scheme to extend lifetime in wireless sensor networks*, Cluster Computing, No. DOI 10.1007/s10586-015-0526-9, pp. 1-5, 2016.
25. L. Quan, S. Xiao, X. Xue and C. Lu, *Neighbor-Aided Spatial-temporal Compressive Data Gathering in Wireless Sensor Networks*, Communications Letters, IEEE, Vol. PP, No. 99, pp. 1, 2016.

26. H. Harb, A. Makhoul and R. Couturier, *An enhanced k-means and anova-based clustering approach for similarity aggregation in underwater wireless sensor networks*, IEEE Sensors journal, Vol. 15, No. 10, pp. 54835493, 2015.
27. L.M.L. Oliveira and J.J.P.C. Rodrigues, *Wireless Sensor Networks: a Survey on Environmental Monitoring*, Journal of Communications, Vol. 6, No. 2, pp. 143-151, 2011.
28. E. G. Julie and S.T. Selvi, *Development of Energy Efficient Clustering Protocol in Wireless Sensor Network Using Neuro-Fuzzy Approach*, The Scientific World Journal, Vol. 2016, No. Article ID 5063261, pp. 8 pages, 2016.
29. T. A. Alghamdi, *Cluster Based Energy Efficient Routing Protocol for Wireless Body Area Network*, Trends in Applied Sciences Research, Vol. 11, No. 1, pp. 12-18, 2016.