Computer Networks 185 (2021) 107701

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Reliable spatial and temporal data redundancy reduction approach for WSN

Zaid Yemeni [a], Haibin Wang [a,b,*], Waleed M. Ismael [a], Yanan Wang [a], Zhengming Chen [a]

[a] *College of Internet of Things (IoT) Engineering, Hohai University, Changzhou, Jiangsu, China*
[b] *China Institute of Atomic Energy, Beijing, China*

ARTICLE INFO

Keywords:
Data redundancy reduction
Data aggregation
WSN
Temporal redundancy
Spatial redundancy

ABSTRACT

Data generated by sensors are inherently apt to spatial and temporal redundancy owing to the proximity of sensors that could sense the same environment or react to the same event. The massively generated data leads to reducing the life span of the sensors in particular, and the network in general. To minimize the effect of such generated data, we develop an approach to reducing the spatial and temporal data redundancy while maintaining the life of the sensor that results in prolonging the lifetime of the network with balancing data reliability. The proposed approach relies on two levels. The first level represents the end node, and it is responsible for reducing the temporal data redundancy and minimizing the data transmission using the Kalman filter for data estimation. The second level represents the sink or base station, which works in synchronization with the end nodes. This level is responsible for minimizing the spatial data redundancy based on two algorithms, namely Sink Level Grouping Algorithm (SLGA) and Sink Level Aggregation Algorithm (SLAA). The obtained results demonstrate that the proposed approach outperformed Prefix Frequency Filtering (PFF) and Redundancy Elimination Data Aggregation (REDA) algorithms in terms of spatial and temporal data redundancy and accuracy with acceptable results of energy consumption.

1. Introduction

Recent advances in sensing technology enable us to continuously monitor real-world phenomena. Sensor nodes are often deployed over geographical areas to monitor either dynamic or non-dynamic environments. Wireless Sensor Networks (WSNs) gave rise to the development of various IoT applications , including environmental monitoring, natural disasters, smart agriculture, industrial, etc [1–7]. In IoT, sensor nodes continuously report their instantaneous readings to base stations (i.e., sink nodes). However, data transmission over the network is the main factor in energy dissipation and communication overhead in WSNs [8,9]. Sensor nodes mainly depend on batteries as a power supply. Therefore, prolonging network lifetime became a significant limitation and an objective principle because it is almost impossible to replace or re-charge the battery of sensor nodes in some real-world monitoring environment [10–15]. The lifetime of WSNs also to large-extent is affected by way of sensor nodes deployment over a large geographical area (i.e., dense or sparse). The densely deployed sensor nodes cause redundant data. Indeed, more redundant data means more data reliability, and redundant data transmission requires energy. As opposed to, less redundant data leads to less reliable data and requires less energy for transmission.

WSNs are more vulnerable to failure and malfunction owing to the limitation of sensor node processing ability, limited transceiver range, and the capacity of its battery, [16–18]. To address such problems in WSNs, various data aggregation approaches have been introduced. Data aggregation techniques have gained considerable attention for minimizing the number of transmissions from the sensor nodes to the base station. The main goal of data aggregation is to effectively aggregate data starting from data source up to the base station. Moreover, data aggregation increases data reliability and prolongs the lifetime of WSNs [19–23].

The proposed approach is developed to minimize data redundancy and power dissipation while maintaining data reliability in WSNs. The proposed approach is a prediction-based approach for redundancy reduction that exploits the Kalman filter. In other words, the proposed approach exploits the decoupled Kalman filter for estimating the next reading both at the source and the base station. So that the sensor nodes transmit their immediate readings. If there is a deviation with their Kalman-filter-based estimated values (> e_{max}, a user-defined threshold). At the source, the ELDRR algorithm is responsible for reducing the temporal redundancy. At the base station, the SLGA algorithm is responsible for reducing the spatial data redundancy exploiting the

[*] Corresponding author at: College of Internet of Things (IoT) Engineering, Hohai University, Changzhou, Jiangsu, China.
E-mail addresses: yemenizaid@hhu.edu.cn (Z. Yemeni), wanghaibin@hhuc.edu.cn (H. Wang), waleed.m@hhu.edu.cn (W.M. Ismael), 2529910662@qq.com (Y. Wang), 19871348@hhu.edu.cn (Z. Chen).

https://doi.org/10.1016/j.comnet.2020.107701
Received 13 June 2020; Received in revised form 2 November 2020; Accepted 19 November 2020
Available online 13 December 2020

grouping technique for spatially related sensor nodes, and the SLAA algorithm is to aggregate the readings of each group individually.

The remainder of this paper is organized as follows: In Section 2, state-of-the-art proposed approaches are presented for data aggregation techniques in WSNs, including Kalman filter and data similarity technique. A system description have been discussed in Section 3. In Section 4, a comprehensive overview of the proposed approach is described. Next, in Section 5, we present the results of the proposed approach evaluated against two state-of-the-art aggregation approaches. Finally, in Section 6, we summarize with a conclusion, along with future work.

2. Related work

One of the challenges that WSN suffers from is data redundancy. In other words, transmitting duplication of data leads to different issues, including energy dissipation, network overhead, and network bottleneck. Data aggregation is one of the techniques that are commonly used to reduce the transmission of the total data over the network, thus saving energy and reduce network bottlenecks [24–28]. Several state-of-the-art approaches employed various data aggregation techniques to reduce data redundancy. Authors in [29] proposed a data redundancy elimination approach for data aggregation in WSN, which exploits the Support Vector Machine (SVM) for eliminating redundant data. They also exploited the locality-sensitive hashing (LSH) algorithms to detect and remove outliers. The approach was simulated using AADL with OSATE, and the results were analyzed in terms of aggregation, the average of the packet delivery ratio, energy consumption, and accuracy. Other authors in [30] suggested a Redundancy Elimination Data Aggregation algorithm (REDA) in WSNs. The proposed technique is a combination of [31] and [32] with considering the energy consumption level of sensor nodes in each phase as a new feature. The proposed algorithm is based on hierarchically structured networks. The cluster heads (CHs) are selected and changed depending on the residual of energy by generating a lookup table and transmitting it to the cluster members. The data transmission depends on the change of the pattern and the cluster member significance to transmit the data if and only if its pattern code had changed. Although this approach has saved up to 40% of network energy consumption, the data reliability was compromised. In [33], a novel data aggregation algorithm called Redundancy Elimination for Accurate Data Aggregation (READA) is introduced. Benefiting from the tree-based approach, READA organized the network into clusters. Besides, each cluster will act as a cluster head. To eliminate the duplicated data, READA applies a grouping and compression mechanism. Additionally, to detect outlier READA exploits a prediction model derived from cached values to confirm whether any outlier is genuinely an event that has happened. Although this approach applies grouping and compression mechanisms, there are still some problems: including delay, loss of data, and accuracy problems [29]. Also, authors in [34] proposed the Energy Efficient High Accuracy (EEHA) scheme for making secure data aggregation. The idea of this scheme is to achieve accurate aggregated data without solidifying the secret readings sensed by end nodes. According to [35], this technique is developed to improve security and data accuracy with more massive energy consumption. In [36], the authors get the benefit of the latest theoretical advances in complex networks to propose an automated solution to improve the topology of WSNs by utilizing centrality metrics to detect the redundant links and nodes in a WSN and assist on to shut down them safely. This proposed solution can work in centralized and decentralized system models. Choosing a centralized or a decentralized centrality metric is driven by the application goal. This approach eliminates data redundancy by turning off some redundant nodes overlapping each other's sensing range. Turning off redundant nodes may lead to some problems, including data reliability and network connectivity. To ensure reliability and prolong the lifetime of sensor nodes, in [37], a novel scheme for data aggregation based on trust and reputation model is proposed. To increase accuracy, the proposed model selects secure paths from sensor nodes to the base station. According to [29], this approach limitations are the transmission overhead and the loss of accuracy while eliminating the data redundancy. Data reduction is one of the widely used strategies to eliminate data redundancy. To find an appropriate solution for network bottleneck problems, authors, in Ref. [6], proposed a real-time data reduction approach called NECtar. Based on the data type, NECtar selects the suitable data reduction algorithms, such as sampling, piecewise approximation, perceptually important points (PIP) algorithm, selective forwarding, and data change detection. Moreover, other authors employed two tiers for solving the issue of energy consumption of IoT devices [9]. The proposed approach functions on two tiers, namely getaway, and network edge tiers. Their approach employs perceptually important points (PIP) to deal with time-series data. Interval restriction, dynamic caching, and weight sequence selection techniques are integrated with PIP algorithms. At the second tier, they proposed data fusion based on optimal dataset selection. Some other authors take advantage of edge networking to perform data reduction. Both [6] and [9] suffers from computational complexity as it requires longer run time to identify the essential points that contribute in the overall shape of time series data in each iteration [38], and its algorithm complexity is high [39]. In Ref. [24], the authors proposed an in-networking double-layered data reduction approach for the Internet of Things. Their approach consists of two layers. The first layer is for data filtering based on the Kalman filter and the second layer is responsible for data fusion. All the above-mentioned data reduction approaches take into account the elimination of temporal data redundancy and neglect the spatial data redundancy. In [40], a data redundancy removal strategy is proposed for minimizing the number of data transmission. The proposed strategy aims to remove redundant data before transmitting them by using data mining. The proposed strategy deals with both temporal and spatial redundancy at the sensor nodes. Although this strategy helps solve data redundancy problems, it compromises the time aspect and reliability of data. Aiming to decrease WSN data transmission and increase energy-efficient zoom-in zoom-out (ZIZO) is presented in [41]. ZIZO works based on two WSN levels. A compression method, namely index-bit-encoding (IBE) was used at the sensor level to aggregate similar readings before transmitting them to the second level. While in the second level, the sampling rate of sensors was optimized by applying the sampling rate adjustment (SRA) process. Despite the proposed work achieved good results in reducing energy consumption, the data reliability is ignored. The authors in [42] proposed a cluster-based framework to control the redundant transmission over WSNs using the statistical test. The proposed framework aims to remove the redundant data before transmitting them to the base station. The proposed framework is cluster-based, which means the redundant data generated by the sensor node itself is sent first to the cluster head before eliminating them. Moreover, this framework does not take into consideration the vital importance of data reliability. In [43] a novel Spatial Correlation-based Data Redundancy Elimination for Data Aggregation (SCDRE) is proposed to eliminate the data redundancy at two levels: source level and aggregator level. The source level used a simple similarity function to eliminate redundant data. In contrast, the aggregator level is based on the correlation coefficient for data redundancy elimination and aggregation. Similar to many state-on-the-arts approaches, this approach compromises the aspect of data reliability.

3. System description

assume we have a network that contains a number of sensor nodes (end-nodes) (N), the nodes are homogeneous and densely distributed in an under-observation phenomenon, as shown in Fig. 1. By using cooja simulation environment, this can be accomplished by randomly assigning an X-coordinate and Y-coordinate in a zone of R x Rm^2. It

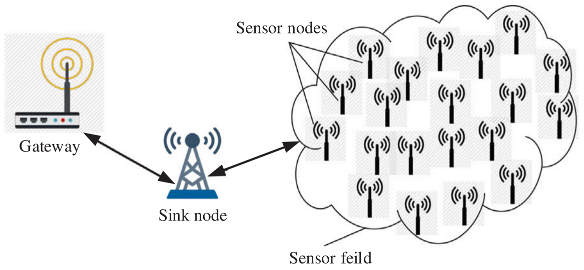

Fig. 1. Network topology architecture.

is worthy to notice that all nodes in the network are stationary. In the other hand, a number of sink nodes (M) are positioned in the network field to collect the data from the sensor node in a systematical manner. After deploying the sensor nodes over the network field, the first step of the proposed (ELRR) algorithm is to reduce the number of transmitted data, as discussed in Section 4. On the other hand, once the data has reached the sink node, the sink level grouping algorithm (SLGA) will directly divide the enter network (in contact with) into groups based on the number of neighbors nodes, starting with the nodes that have the most number of neighbors to the smallest. It is useful to know that the division is done periodically based on the desire of the user, as we discussed in detail in Section 4. After the completion of the nodes grouping comes the role of the sink level aggregation algorithm (SLAA), which purifies and refines data from spatial redundancy, as we discussed in more detail in Section 4.

4. Proposed approach

Assume a network consists of N sensors denoted by $S = \{s_1, s_2, \ldots, s_N\}$, which are deployed randomly, and M sink nodes located in a monitoring area. At each time, t, each sensor sends the dynamic data stream to the sink. Each sensor is not required to send all its actual readings. In other words, each sensor has to send the actual readings to the sink if the deviation between the actual reading and its estimated value is greater than the user-defined threshold (e_{max}).

In this paper, we propose an approach for reducing the spatial and temporal data redundancy while maintaining the reliability of data. Fig. 2 depicts an illustrative demonstration of the proposed approach architecture. The proposed approach operates at two levels, namely the end-node level and sink-node level. At the end-node level, the proposed approach reduces temporal data redundancy and data transmission over a network by applying the Kalman filter to the actual readings to obtain the estimated values. Then, a user-defined value is used to find the deviation between the actual readings and estimated ones. The readings that have a higher deviation than the threshold will be sent to the sink. Otherwise, the data is discarded, and the estimated value at the sink is used instead. At the sink-node level, the proposed exploits two mechanisms (grouping mechanism and data aggregation mechanism). The proposed approach uses a grouping mechanism to group the nodes that are highly correlated based on analyzing the nodes' spatial correlation to reduce the spatial redundancy resulted from the densely deployed sensor nodes. It is worthy to note that the proposed approach applies the grouping mechanism immediately after the proposed approach set up or to adapt the dynamic changes of a network. Firstly, in the grouping process, all end nodes are listed in a list, along with their spatially correlated neighbors. Then, a node with the highest number of neighbors is chosen to form the first group. This step is carried out to decrease the number of groups. Next, the next node with the highest number of nodes is selected to form another group, considering that the selected node is not in the previously formed group to avoid forming groups of already grouped nodes and minimizing the overhead. This process continues traversing the list to the end of the list. Based on the formed groups, the proposed approach exploits the data aggregation mechanism. As shown in Fig. 4 during the data aggregation process, the proposed approach works in two modes, namely an active mode and passive mode. In the active mode, the proposed approach exploits the Kalman filter to estimate the readings discarded by the end nodes, which occurs if the deviation between the actual reading and its estimated value is smaller than or equal to the user-defined threshold (e_{max}). In the passive mode, the proposed approach relies on the readings received from the end nodes. In both modes, the readings of each group are aggregated to represent the state of the environment under observation.

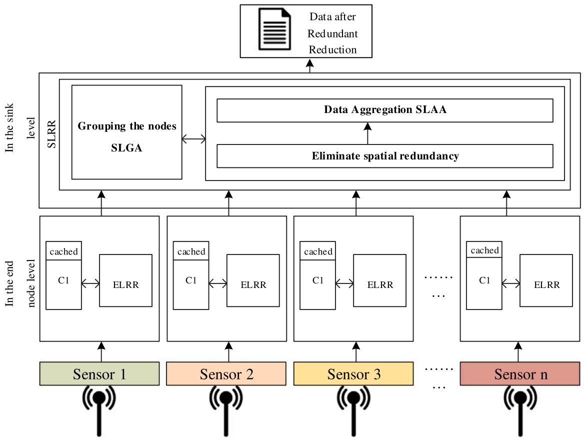

Fig. 2. The proposed approach architecture.

4.1. End Node Level Redundancy Reduction (ELRR)

ELRR is developed for reducing the temporal redundancy of each sensor node. As shown in Fig. 3, it is based on two techniques. The first is data change detection. Data change detection, as specified by [44], is defined as the process of identifying differences in the state of an object or phenomenon by observing it at different times. In ELRR, data detection is the process of determining the difference between the previous and current reading. ELRR caches the first reading of a sensor node ($cached_x$) and passes it to the sink. Then, each reading at time t is compared with the $cached_x$ value. If no change is detected, the current reading z_t is discarded, else the $cached_x$ is updated with the current reading z_t.

The second technique is the deviation between the actual readings and their estimated values. The idea behind this technique is that if the data change is detected, the current reading z_t is passed to the Kalman filter to find its estimated value (e_t). The deviation between the actual current reading z_t and its Kalman-filter-based estimated value is calculated according to Eq. (1). If deviation dev_t is greater than the user-defined threshold e_{max}, then z_t is sent to the sink else the z_t value is discarded, and the corresponding estimated value at the sink is used instead.

$$dev_t = |z_t - e_t| \tag{1}$$

where dev_t is the deviation between the actual reading and it estimated value, z_t is the current reading at time t, and e_t is estimated value by Kalman filter at time t. We designed the end-node algorithms to be optimal to avoid computational overhead, which can affect the node's lifetime. So that the complexity of end node algorithm is $O(1)$. The Algorithm 1 represents the end-node level, and the table summarizes the parameters and variables used (see Table 1).

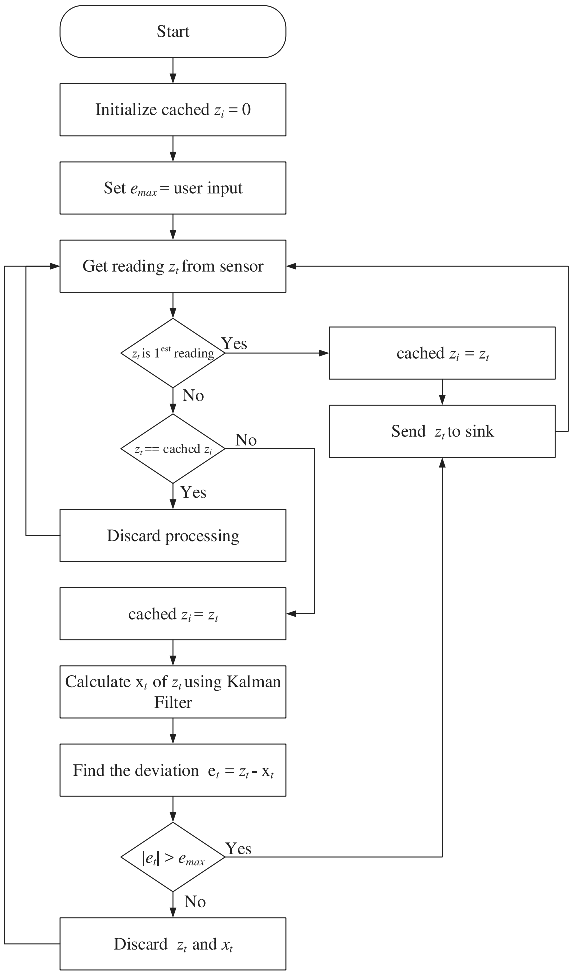
Start
Initialize cached $z_i = 0$
Set e_{max} = user input
Get reading z_t from sensor
z_t is 1^{est} reading
Yes
cached $z_i = z_t$
Send z_t to sink
No
z_t == cached z_i
No
Yes
Discard processing
cached $z_i = z_t$
Calculate x_t of z_t using Kalman Filter
Find the deviation $e_t = z_t - x_t$
$|e_t| > e_{max}$
Yes
No
Discard z_t and x_t

Fig. 3. The operational workflow of the end node.

Algorithm 1: End Node Level Redundancy Reduction (ELRR)

```
Input: The readings of sensor node
Output: Filtered Readings
set $e_{max}$ ← user input
set $cached_x$ ← first reading of the sensor
set $z_t$ ← the new sensor readings
if $z_t = cached_x$ then
    Exit
else
    $e_t$ ← calculate estimated value based on Kalman filter
    if $e_t = z_t$ then
    $cached_x \leftarrow e_t$
    Exit
    else
        $dev_t = |z_t - e_t|$
        if $dev_t > e_{max}$ then
            send $z_t$ to aggregation center
            $delcached_x$
            $cached_x \leftarrow z_t$
        else
            $cached_x \leftarrow z_t$
            exit
```

Table 1
The summary of Algorithm's 1 parameters.

Parameter	Definition
e_{max}	User defined absolute maximum error
$cached_x$	The first reading of sensor node
z_e	The current reading of sensor node
e_t	The Kalman filter estimated value
dev_t	Deviation between the actual reading and it Estimated value

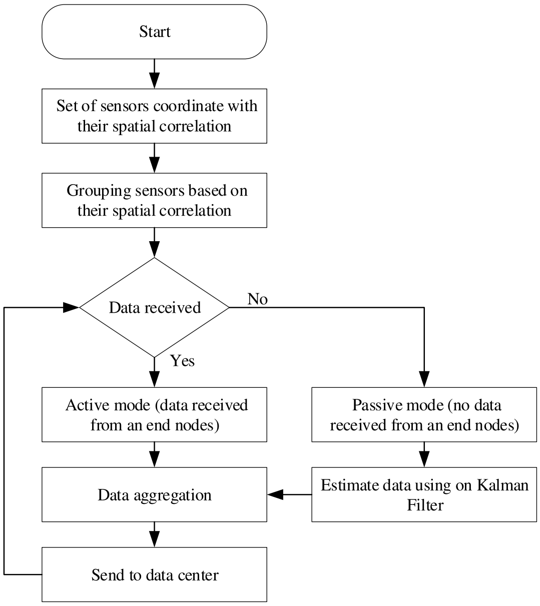
Start
Set of sensors coordinate with their spatial correlation
Grouping sensors based on their spatial correlation
Data received
No
Yes
Active mode (data received from an end nodes)
Passive mode (no data received from an end nodes)
Data aggregation
Estimate data using on Kalman Filter
Send to data center

Fig. 4. Sink operational flowchart.

4.2. Sink Level Redundancy Reduction (SLRR)

In a wireless sensor network, all the data collected by sensors is transmitted to the sink node. The processing unit, energy, and transmitter of the Sink node are more significantly impacted than the end node. In this paper, we proposed two algorithms for the sink node. The first algorithm we called sink level grouping algorithm, which is responsible for grouping the sensor nodes periodically depending on their vicinity to other nodes. The second algorithm takes the output of the sink level grouping algorithm of grouping. It works continuously to check the group members' similarity. Then, based on similarity, the algorithm will aggregate the group members' data. It is worthy to note that the sink will not receive all data captured by the end nodes since the data is filtered at each end node, and only some of the data will be sent to the sink. The filtered data at the end node will be simultaneously estimated at the sink using the Kalman filter. Below we show the structure of the proposed algorithms and discuss them in more detail.

4.2.1. Grouping algorithm

After setting up the approach for the first time at the sink node, the grouping algorithm initializes collecting information about sensor nodes (e.g., location coordinates and sensing ranges). Then, it groups sensor nodes in groups, and each group consists of sensor nodes that are high spatially correlated. The grouping algorithm depends primarily on the type of application and mobility of the end nodes. In some applications, the grouping process is only done once if sensor nodes are immobile. In other applications, the grouping process is performed periodically to adapt to the changes in the network structure. For instance, if the network is immobile, the grouping algorithm is only initiated for one time. However, if the network is mobile, the grouping algorithm is required to re-operated. To solve the problem of common

Table 2
WSN example.

Node ID	Neighbors	Node ID	Neighbors
1	[5,11]	14	[15,3]
2	[4,3,11,15,19]	15	[14,3,2,22,6]
3	[15,14,11,22,6,2,4]	16	[5,10]
4	[19,8,3,13,2,7]	17	[20,25,8,18,21,9,12]
5	[1,11,16]	18	[25,8,7,21,17]
6	[15,22,3]	19	[10,4,8,2,7]
7	[25,4,19,8,18,13]	20	[25,17,12]
8	[10,4,19,18,7,17]	21	[18,17,25]
9	[10,24,12,17,23]	22	[3,6,15]
10	[9,16,8,19]	23	[9,24,12]
11	[5,2,1,3,]	24	[9,23]
12	[20,9,23,17]	25	[20,18,7,17,21]
13	[4,7]		

neighbors and lessen the time of the grouping process as well as the number of groups, the grouping algorithm starts with the sensor with the highest number of neighbors.

Fig. 5 depicts a network contains 25 sensors. The sensors nodes in the network are randomly deployed. Firstly, we find the neighbors of each sensor separately, as shown in Table 2. As shown in Fig. 5 and Table 2, we notice that the sensors densely deployed produce spatial redundancy inevitably. Therefore, to reduce the spatial redundancy, the grouping algorithm aims to aggregate the readings of the sensor nodes that are spatially correlated. After grouping spatially-correlated sensor nodes, the grouping algorithm investigates the common neighbors among groups and specifies the closest group to which they belong.

For example, in Table 2, node 24 with its neighbors are neighbors of node 23. In other words, $S_{24} \subseteq S_{23}$, which means sensor 24, is a subgroup of sensor 23. In the same way nodes, 13 and its neighbors are contained in node 7. In this case, the groups are unionized. After merging groups, there are still common neighbors between groups, and the grouping algorithm will investigate that and attach the senor to the group that has a higher correlation. For example, node 14 and 6 in Table 2 appear to have node number 15 as a common neighbor. To find the closest group to which the node 15 belongs, the distance is calculated based on Eq. (3), and the shortest distance is selected to add the node to the closest group.

The grouping algorithm is based on the structure of the network as an input. The grouping algorithm investigate the whole network to find the common neighbors between groups according to Eq. (2).

$$D = S_i Z - S_j Z \tag{2}$$

Where S_i, S_j are two nodes with intersection and Z is the Common neighbor between S_i and S_j.

$$dist(S_i Z, S_j Z) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{3}$$

According to Eqs. (2) and (3), there are three cases for the Z node based on the distance calculated to each group. If the Z node is close to S_i, the Z node will be added to the S_i, if the Z node is close to S_j, the Z node will be added to S_j group. In case, the distance is equal, the grouping algorithm will add the Z node either to S_i or S_j as Eq. (4).

$$Z \in \begin{cases} Z \in S_i & if D > 0 \\ Z \in S_j & if D < 0 \\ Z \in S_i or S_j & otherwise \end{cases} \tag{4}$$

It is worthy to note that when a node is selected to form a group, all its neighbors will not be able to form any group and this reduces the number of groups. If there are intersections (common neighbors). In this case, the grouping algorithm will calculate the neighbors according to geographical proximity based on the location of the node. Algorithm 2 works on the sink node, and its complexity is $O(n^2)$. Table 3 summarizes the used parameters and variables.

Algorithm 2: Node Grouping

```
Input: The total sensors of a given network $S = \{s_1, s_2, ..., s_n\}$
Output: The List of groups
set $Q \leftarrow []$
set $nL \leftarrow []$
set $gL \leftarrow []$
for each $s_i \in L$ do
    for each $s_j \in L$ and $s_j \neq L_i$ do
        if $s_i \subseteq s_j$ then
            N ← $s_i$ $N$.insertneighbor($s_j$)
        else if $s_j \subseteq s_i$ then
            $N \leftarrow s_j$ $N$.insertneighbor($s_i$)
    if $N$ not $\in nL$ then
        $nL$.add(N)
for each $s_k \in nL$ do
    if $s_k$ not $\in Q$ then
        gL.append ($s_k$) for each $g_i \in s_k nieghbors$ do
            if $g_i$ not $\in Q$ then
                Q.append ($g_i$)
```

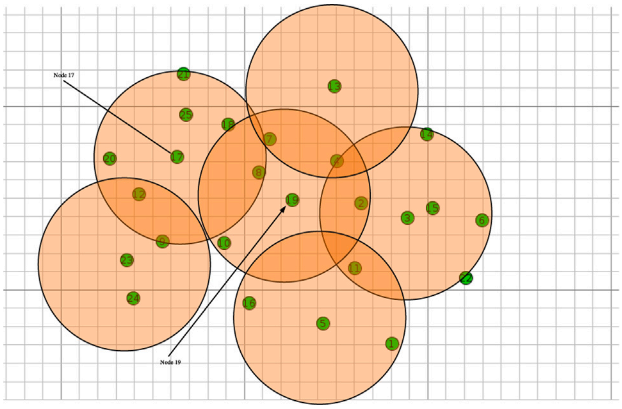

Fig. 5. Grouping example.

Table 3
The summary of Algorithm's 2 parameters.

Parameter	Definition
Q	Represents the final output groups
nL	A list to represent non repeated members
gL	A list to represent the final generated groups
L	The list of initial groups created by cooja simulator
s_i	A member group of L
s_j	A member group of L
N	A list to represent not contained groups
Sk	Represents a member group of nL
g_i	Represents a group neighbors

4.2.2. *Aggregating algorithm*

The aggregation algorithm is developed to reduce the spatial redundancy among spatially correlated sensor nodes as grouped by the grouping algorithm. The aggregation algorithm receives the readings from end nodes to be aggregated. It is mainly based on the Kalman filter to perform reading estimation for the reading that is filtered by the end node. It requires to be pre-configured to work in synchronization with end nodes. It runs each time to check each group to find out whether readings are received from the end nodes or not. If there are no readings received, it exploits the Kalman filter to estimate the filtered out readings at the end nodes. Then, based on the estimated data (passive mode) or received readings (active mode), it aggregates the readings of each group by taking the average according to Eq. (6) at time t. After that, the aggregated values of groups at time t is checked. If the aggregated data still has spatial redundancy, it is aggregated over again.

$$CV = \begin{cases} est & if CV = \Phi \\ Kalmanfilter.Update & otherwise \end{cases} \tag{5}$$

where CV represents the current reading of an individual end node, est represents the estimated value by Kalman filter and is a function used to update Kalman filter estimation based on the arrived reading from the end node.

The aggregating algorithm exploits different variables and lists to perform the aggregation operation. The T pointer is used to traverse the list of the groups starting from the head of the list LG and AG is the aggregated value of each group at time t, which is calculated according to Eq. (6). AGL is a list used to store the total aggregated values of all groups at time t. First, the aggregating algorithm finds the head of each group, which is selected arbitrarily by the grouping algorithm, and then checks its current readings. If the reading is filtered at the end node, the estimated value produced by the Kalman filter is used instead. Otherwise, the current reading is aggregated with the readings of neighboring nodes readings, and the Kalman filter is update accordingly. Second, each member el of each group is investigated in the same way. If the current reading of el is none, the estimated value by Kalman filter is used. Otherwise, the current reading of el is used, and the Kalman filter is updated according to the current reading. After checking all members in the group, the average value of the group member reading is calculated according to Eq. (6).

$$AVG_j(t) = \frac{\sum_{i=1}^{m} z_i(t)}{m} \tag{6}$$

where $AVG(t)$ is the total aggregated value of the whole group members readings at time t, $z_i(t)$ is the current reading at time t of a sensor i in the currently processed group, and m is the number of nodes in a specific group. The aggregated value is added to the aggregation list (AGL), which represents the aggregated values of all groups at time t. The aggregating algorithm continues traversing the LG list until the end. The final output of the aggregating algorithm is a list (AGL) representing the aggregated values of all groups. Algorithm 3 describes the implementation of Data Aggregation and Table 4 summarizes the used parameters. Algorithm 3 works on the sink node , and its complexity is $O(m * n)$, where m is the number of outer iterations and n is the number of inner iterations.

5. Experiments and results

To evaluate the effectiveness of the proposed approach, the proposed approach is evaluated against two aggregation methods, namely REDA and PFF, in terms of data redundancy reduction (both spatial and temporal data redundancy), data reliability, and energy consumption.

5.1. Datasets

The datasets are randomly generated using the Cooja simulator for 500 sensors. For each sensor, 1100 readings are generated with a fixed sample rate of 5 readings per minute to represent a real-like temperature between 19 °C and 40 °C. For clarification, Table 5 represents samples of generated datasets selected arbitrarily for different sensors. It is worthy to note that the generated datasets contain both spatial and temporal redundancy for the evaluation of the proposed approach and counterparts (REDA and PFF).

Algorithm 3: Aggregation Algorithm.

```
Input: List of groups $LG$
Output: Aggregated Data
set $T \leftarrow LG$.head
set $AG \leftarrow 0$
set $AGL \leftarrow []$
while $T$ is not None do
    $GH \leftarrow$ findGroupHead($T$)
    $CV \leftarrow$ getCurrentReading()
    if $CV$ is None then
        $CV \leftarrow$ Kalmanfilter
        $AG = AG + CV$
    else
        $AG = AG + CV$
        Kalmanfilter.update()
    for each el $\in T$.getnieghbor() do
        $CV \leftarrow el$.getCurrentReading()
        if $CV$ is None then
            $CV \leftarrow$ Kalmanfilter
            $AG{=}AG + CV$
        else
            $AG{=}AG + CV$
            Kalmanfilter.update()
    $AGL$.append($AVG(AG)$)
    reset $AG \leftarrow 0.0$
    $T \leftarrow T$.next
```

Table 4
The summary of Algorithm's 3 parameters.

Parameter	Definition
T	A pointer to traverse the list of the groups starting from the head of the list LG
AG	Represents the aggregated value
AGL	Represents a list to store the whole aggregated values of all groups
CV	Represents the current reading
GH	Represents the head of a specific group

Table 5
Selected sensors' readings.

Reading no.	s_1	s_{10}	s_{100}	s_{150}	s_{300}	s_{400}	s_{500}
1	19.013	19.044	19.007	19.000	19.032	19.028	19.008
100	20.848	21.112	20.980	20.888	20.929	21.097	20.972
200	22.493	22.725	22.740	23.055	22.684	22.960	22.803
300	24.524	24.532	24.800	25.015	24.583	25.285	24.572
400	26.508	26.230	26.789	27.073	26.394	27.034	26.693
500	28.343	27.874	28.795	29.083	28.543	28.661	28.364
600	30.306	29.782	30.732	30.876	30.409	30.650	30.290
700	32.305	31.747	32.453	32.730	32.539	32.714	32.373
800	34.306	33.877	34.419	34.485	34.416	34.546	34.395
900	36.079	35.859	36.487	36.577	36.247	36.294	36.224
1000	37.909	37.863	38.322	38.192	38.194	37.966	38.270
1100	39.982	39.925	39.996	39.998	39.887	39.964	39.957

5.2. Simulation setting

Two simulation environments (Cooja and Python) are used to simulate the proposed, REDA and PFF. The Cooja simulator is used to create different WSN scenarios. Fourteen WSN scenarios with a different number of sensors ranging from 50 sensors to 500 sensors are used. The sensor type used in each WSN scenario is zolertia. Each WSN is assumed to be randomly deployed over a specific area. Table 6 summarizes the details of each WSN scenario. Python is used to write the simulation code for the proposed approach, REDA, and PFF.

Table 6
The summary of the structure of networks for the proposed approach evaluation.

No.	WSN	READINGS
1	50	1100
2	75	1100
3	100	1100
4	125	1100
5	150	1100
6	175	1100
7	200	1100
8	225	1100
9	250	1100
10	275	1100
11	300	1100
12	350	1100
13	400	1100
14	500	1100

Table 7
The initial estimated values for elected sensors.

Sensor	Initial x values				
s_1	19.013	19.029	19.041	19.055	19.090
s_2	19.016	19.022	19.032	19.096	19.123
s_3	19.000	19.043	19.054	19.071	19.100
s_4	19.036	19.053	19.071	19.081	19.090
s_5	19.005	19.023	19.026	19.056	19.085

Table 8
The parameters of the proposed approach, REDA and PFF.

Proposed approach	REDA	PFF		
emax	p	P	e	td
0.1	5	5	0.03	0.35
0.2	7	5	0.05	0.4
0.3	10	5	0.1	0.5

Besides, since the Kalman filter needs prior information to work correctly and to synchronize the Kalman filter at the end node level and the sink level, we adopt the same initial values and parameters (A, H, Q, and P). Table 7 summarizes the initial values for some chosen sensors.

The Kalman filter parameters (A and H) at the end node level and the sink level are initialized as follows:

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

where Q is considered as white Gaussian noise and the target noise is P. Q and P were initialized as follows:

$$Q = \begin{bmatrix} 2.25e^{-6} & 4.50e^{-6} \\ 4.50e^{-6} & 9.00e^{-6} \end{bmatrix}$$

$$P = \begin{bmatrix} 0.000001 & 0 \\ 0 & 0 \end{bmatrix}$$

In our simulation, the proposed approach, REDA and PFF are assumed to depend on setup parameters. Table 8 summarizes the parameters of the proposed approach, REDA and PFF, respectively.

5.3. Study results and evaluation

The performance of the proposed approach is evaluated based on the results obtained from the two levels as follows:

5.3.1. Spatial data redundancy

In this section, the spatial data redundancy is analyzed for the proposed approach, REDA, and PFF. The spatial data redundancy of the three approaches is evaluated in comparison with the spatial data redundancy of the input data in percentage terms. The spatial data redundancy is calculated according to Eq. (7) at each time t. Then the total spatial data redundancy is summed up to represent the whole datasets.

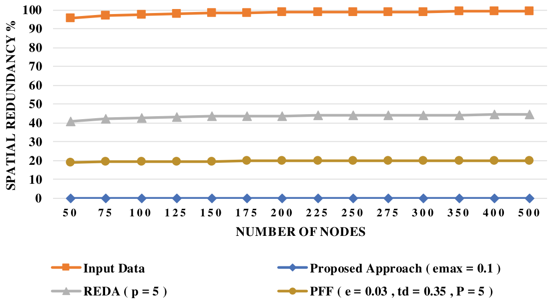

Fig. 6. Spatial redundancy reduction of proposed approach, REDA, and PFF with $e_{max} = 0.1$.

$$SR(t) = 1 - \frac{\mathrm{NSR(t)}}{\mathrm{NR(t)}} * 100 \tag{7}$$

where SR is the spatial data redundancy at time t, NSR is the similar readings at time t, and NR is the total readings at time t.

Table 9 summarizes the experimental results of spatial data redundancy of the proposed approach, REDA, and PFF for the fourteen networks. It shows the spatial redundancy of the input data and the proposed approach, REDA and PFF based on different parameters, respectively. According to Table 9, the input data of different WSN scenarios contains spatial redundancy ranging from 97.08% to 99.49%. It is seen that the proposed approach achieves zero spatial redundancy for all e_{max} values (0.1,0.2,0.3) in comparison with the spatial redundancy of the input data. Regarding REDA, the experiment results contain spatial redundancy in the range between 40.77% and 44.49% for the interval (5), in the range between 38.25% and 41.99% for the interval (7), and in the range between 37.73% and 41.49% for the interval (10) compared with the spatial redundancy of the input data. It is noted that the proposed approach outperforms REDA in terms of spatial redundancy.

With regards to PFF, the experiment results show that PFF accomplishes spatial redundancy ranging from 19.14% to 19.90% for parameters (p = 5, e = 0.03, td = 0.35), ranging from 19.14% to 19.90% for parameters (p = 5, e = 0.05, td = 0.4), and ranging from 19.29% to 19.92% for parameters (p = 5, e = 0.1, td = 0.5). It can be noted that the proposed approach outperforms PFF in terms of spatial redundancy.

Figs. 6 and 7 depict the spatial data redundancy reduction calculated based on Eq. (8) for the experiment results shown in Table 9 and represent an illustrative demonstration for the evaluation of the proposed approach against REDA and PFF. As it can be noted, the proposed approach outperforms both REDA and PFF in terms of spatial redundancy reduction, but REDA achieves a higher spatial redundancy than PFF.

$$RRP = (SI - SR) + (100 - SI) \tag{8}$$

where RRP is the redundancy reduction percentage, SI is the spatial data redundancy of the input data, and SR is the spatial data redundancy of the obtained results (see Figs. 8 and 9).

5.3.2. Temporal data redundancy

Temporal redundancy is known as performing an action more than one time. In our simulation, we assume that Z is the input data, where Z is a vector $\{z_1, \dots, z_n\}$. Temporal redundancy is calculated according to the following snippet of code:

```
For each z(i) in Z Do
If |z(i+1)−z(i)|< threshold Then
m=m+1
TR=(m/n)∗100
```

Table 9
Comparison of spatial redundancy of the proposed approach, REDA and PFF.

Number of nodes	Input data	Proposed approach			READ			PFF		
		emax = 0.1	emax = 0.2	emax = 0.3	p = 5	p = 7	p = 10	e = 0.03, td = 0.35, p = 5	e = 0.05, td = 0.4, p = 5	e = 0.1, td = 0.5, p = 5
50	95.73	0.00	0.00	0.00	40.77	38.25	37.73	19.14	19.14	19.29
75	97.08	0.00	0.00	0.00	42.11	39.60	39.08	19.41	19.41	19.52
100	97.81	0.00	0.00	0.00	42.83	40.32	39.81	19.56	19.56	19.64
125	98.12	0.00	0.00	0.00	43.15	40.64	40.13	19.62	19.62	19.69
150	98.43	0.00	0.00	0.00	43.45	40.94	40.43	19.69	19.69	19.74
175	98.65	0.00	0.00	0.00	43.67	41.16	40.65	19.73	19.73	19.78
200	98.81	0.00	0.00	0.00	43.82	41.32	40.81	19.76	19.76	19.80
225	98.92	0.00	0.00	0.00	43.93	41.43	40.92	19.78	19.78	19.82
250	99.03	0.00	0.00	0.00	44.04	41.53	41.03	19.81	19.81	19.84
275	99.11	0.00	0.00	0.00	44.11	41.61	41.11	19.82	19.82	19.85
300	99.18	0.00	0.00	0.00	44.18	41.68	41.18	19.84	19.84	19.86
350	99.28	0.00	0.00	0.00	44.29	41.78	41.28	19.86	19.86	19.88
400	99.36	0.00	0.00	0.00	44.37	41.87	41.36	19.87	19.87	19.89
500	99.49	0.00	0.00	0.00	44.49	41.99	41.49	19.90	19.90	19.92

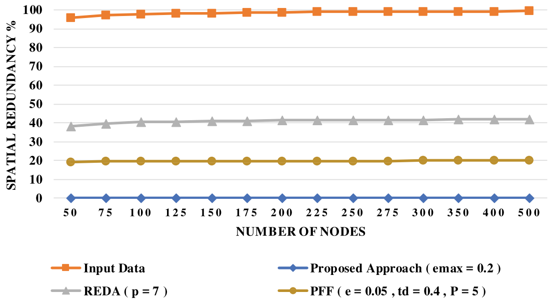

Fig. 7. Spatial redundancy reduction of proposed approach, REDA, and PFF with $e_{max} = 0.2$.

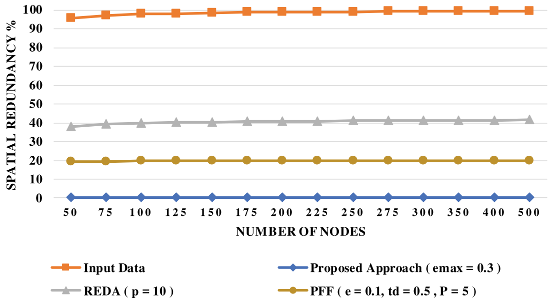

Fig. 8. Spatial redundancy reduction of proposed approach, REDA, and PFF with $e_{max} = 0.3$.

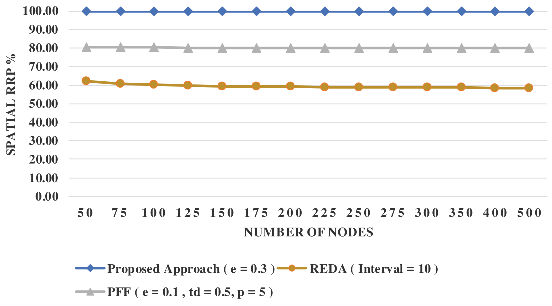

Fig. 9. Spatial redundancy reduction percentage of proposed approach, REDA, and PFF with $e_{max} = 0.3$.

where m is the number of temporal redundancy occurrence, n is the total number of readings, and TR is the total temporal redundancy.

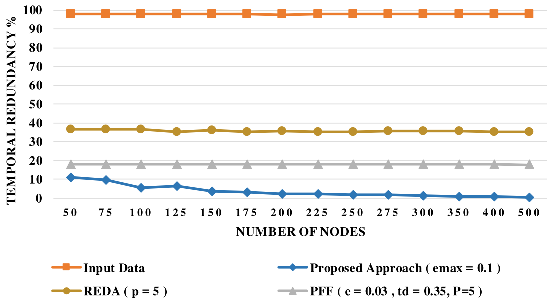

Fig. 10. Temporal redundancy reduction of proposed approach, REDA, and PFF with $e_{max} = 0.1$.

Table 10 summarizes the experiment results for different networks scenarios. It shows that the temporal data redundancy of the proposed approach, REDA and PFF in percentage terms. Based on the obtained results, the input data contains temporal redundancy ranging between 97.95% and 97.96%. It can be seen that the proposed approach accomplishes a lower percentage of temporal redundancy for the different values of e_{max}. For $e_{max} = 0.1$, the proposed approach achieves temporal redundancy in the range from 0.40% to 11.29%. The proposed approach also achieves temporal redundancy ranging between 0.40% to 11.26% and ranging between 0.40% to 11.24% for $e_{max} = 0.2$ and $e_{max} = 0.3$, respectively. Regarding REDA, temporal redundancy is evaluated according to different pre-configurational pattern codes (5, 7, 10). For the pattern code (5), REDA achieves temporal redundancy ranging between 35.24% and 36.87%. In the pattern code (7), REDA shows temporal redundancy ranging between 36.06% and 37.05%. With regards to the pattern code (10), REDA accomplishes temporal redundancy ranging between 37.95% and 37.96%. PFF is also evaluated with different parameters. For parameters (p = 5, e = 0.03, td=0.35), PFF shows temporal redundancy ranging between 17.88% and 17.97%. It also achieves temporal redundancy ranging between 17.96% to 17.97%, and 14.02% and 14.46% for parameters (p = 5, e = 0.05, td = 0.4) and (p = 5, e = 0.1, td = 0.5), respectively.

Figs. 10–12 illustrate the obtained results shown in Table 10. As it can be seen, REDA archives a higher temporal redundancy than the proposed approach. This is attributed to the fact that (REASON). PFF shows a higher temporal redundancy than the proposed approach, but lower than REDA. It is noted that PFF shows a higher temporal redundancy than the proposed approach and a lower temporal redundancy than REDA (see Fig. 13).

5.3.3. Data accuracy

To ensure the reliability of the obtained results, data accuracy is calculated as the deviation between the mean of the actual readings and

Table 10
Comparison of temporal redundancy of the proposed approach, REDA and PFF.

Number of nodes	Input data	Proposed approach			READ			PFF		
		emax = 0.1	emax = 0.2	emax = 0.3	p = 5	p = 7	p = 10	e = 0.03, td = 0.35, p = 5	e = 0.05, td = 0.4, p = 5	e = 0.1, td = 0.5, p = 5
50	97.96	11.29	11.26	11.24	36.87	37.05	37.96	17.97	17.97	14.46
75	97.96	9.64	9.62	9.59	36.54	36.56	37.96	17.98	17.98	14.23
100	97.96	5.50	5.50	5.53	36.71	36.79	37.96	17.98	17.98	14.49
125	97.95	6.57	6.55	6.58	35.47	36.29	37.95	17.97	17.97	14.22
150	97.95	3.52	3.51	3.51	36.10	36.76	37.95	17.96	17.96	14.06
175	97.95	3.18	3.18	3.17	35.23	36.29	37.95	17.96	17.96	14.16
200	97.95	2.37	2.36	2.37	35.89	36.29	37.95	17.96	17.96	14.14
225	97.95	2.26	2.26	2.27	35.40	36.35	37.95	17.96	17.96	14.15
250	97.95	1.74	1.74	1.74	35.48	36.24	37.95	17.95	17.96	14.11
275	97.95	1.58	1.59	1.58	35.56	36.22	37.95	17.94	17.96	14.05
300	97.95	1.13	1.13	1.13	35.94	36.33	37.95	17.93	17.97	14.09
350	97.95	0.95	0.95	0.95	35.63	36.06	37.95	17.92	17.97	13.99
400	97.95	0.73	0.73	0.73	35.45	36.06	37.95	17.91	17.96	14.02
500	97.95	0.40	0.40	0.40	35.24	36.21	37.95	17.88	17.96	14.03

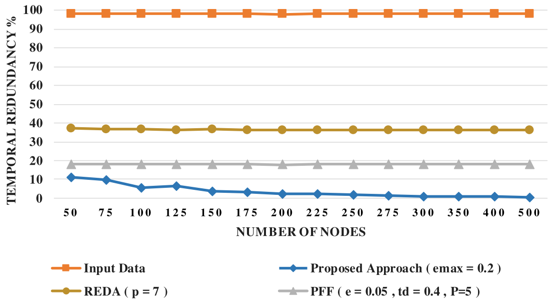

Fig. 11. Temporal redundancy reduction of proposed approach, REDA, and PFF with $e_{max} = 0.2$.

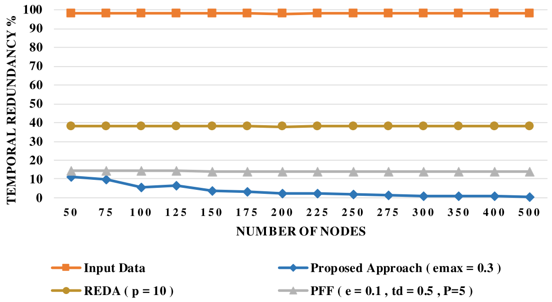

Fig. 12. Temporal redundancy reduction of proposed approach, REDA, and PFF with $e_{max} = 0.3$.

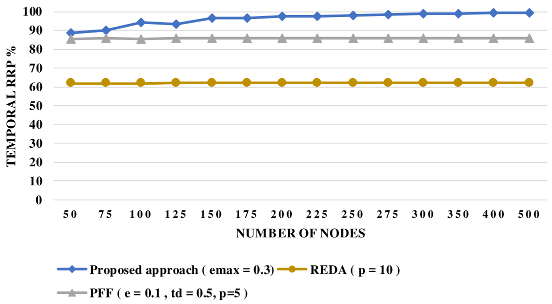

Fig. 13. Temporal redundancy reduction percentage,of proposed approach, REDA, and PFF with $e_{max} = 0.3$.

the mean of the obtained results after applying the proposed approach, REDA and PFF. It is worthy to note that less deviation means higher accuracy and vise versa. Data accuracy is calculated for the entire datasets according to Eq. (9).

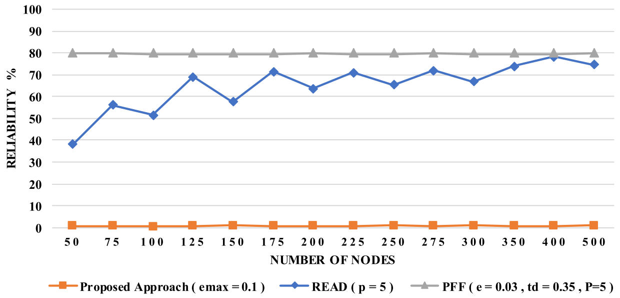

Fig. 14. Data accuracy of proposed approach, REDA, and PFF with $e_{max} = 0.1$.

$$TA = \left| \frac{AM - EM}{AM} \right| * 100 \tag{9}$$

where TA is the total accuracy of the entire datasets, AM is the actual mean of the input data, and EM is the estimated mean of the obtained results.

Table 11 shows the evaluation of data accuracy of the proposed approach, REDA and PFF. According to Table 11, the proposed approach accomplishes data accuracy ranging between 0.44% and 1.03% for the values of $e_{\max} = \{0.1, 0.2, 0.3\}$. Regarding REDA, it achieves data accuracy in the range from 38.20% to 78.23% for the pattern code (5) and in the range from 38.21% to 78.23% for the pattern codes (7,10). With regards to PFF, it shows data accuracy up to 80.03% for parameters (p = 5, e = 0.03, td=0.35) and (p = 5, e = 0.05, td=0.4). For parameters (p = 5, e = 0.1, td=0.5), PFF shows data accuracy ranging between 80.01% and 80.02%.

Figs. 14–16 depict an illustrative demonstration for the obtained results shown in Table 11. We can see that the proposed approach achieves the highest data accuracy than REDA and PFF, while PFF achieves the worst data accuracy than the proposed approach and REDA.

5.3.4. Energy consumption

To ensure the efficiency of the proposed approach, we evaluate the proposed approach against both REDA and PFF. The energy consumption is calculated for the transmission of each reading from end nodes to the sink with taking into consideration the modes of CPU (active mode and idle mode) according to Eqs. (10) and (11) [45].

$$energy = charge * voltage \tag{10}$$

$$charge = \frac{current * (cpu + cpu\_idle)}{RTIMER\_ARCH\_SECOND} \tag{11}$$

Table 11
Comparison of data reliability of the proposed approach, REDA and PFF.

Number of Nodes	Proposed approach			READ			PFF		
	emax = 0.1	emax = 0.2	emax = 0.3	p = 5	p = 7	p = 10	e = 0.03, td = 0.35, p = 5	e = 0.05, td = 0.4, p = 5	e = 0.1, td = 0.5, p = 5
50	0.69	0.69	0.69	38.20	38.21	38.21	80.03	80.03	80.02
75	0.68	0.68	0.67	56.09	56.09	56.09	80.03	80.03	80.02
100	0.44	0.44	0.44	51.48	51.48	51.48	80.03	80.03	80.01
125	0.68	0.68	0.67	69.15	69.15	69.15	80.03	80.03	80.02
150	0.84	0.84	0.84	57.68	57.68	57.68	80.03	80.03	80.01
175	0.77	0.77	0.76	71.57	71.57	71.57	80.03	80.03	80.02
200	0.80	0.80	0.80	63.68	63.68	63.68	80.03	80.03	80.01
225	0.81	0.81	0.80	71.13	71.13	71.13	80.03	80.03	80.01
250	0.95	0.95	0.95	65.45	65.45	65.45	80.03	80.03	80.01
275	0.77	0.77	0.76	71.97	71.97	71.97	80.03	80.03	80.01
300	0.90	0.90	0.90	66.90	66.90	66.90	80.03	80.03	80.01
350	0.73	0.73	0.73	73.93	73.93	73.93	80.03	80.03	80.01
400	0.76	0.76	0.76	78.23	78.23	78.23	80.03	80.03	80.01
500	1.03	1.03	1.03	74.56	74.56	74.56	80.03	80.03	80.01

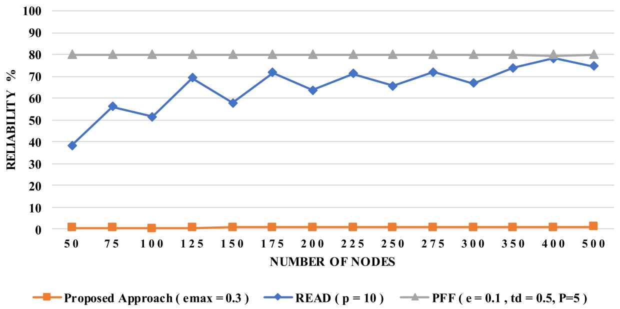

Fig. 15. Data accuracy of proposed approach, REDA, and PFF with $e_{max} = 0.2$.

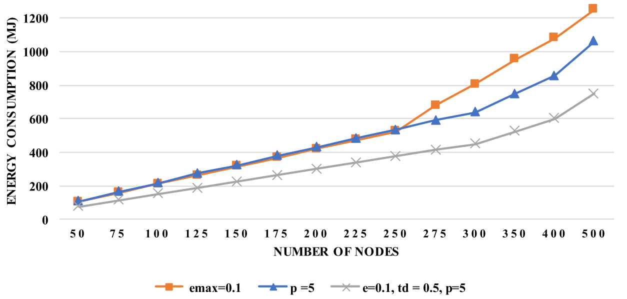

Fig. 17. Energy consumption of proposed approach, REDA, and PFF with $e_{max=0.1}$.

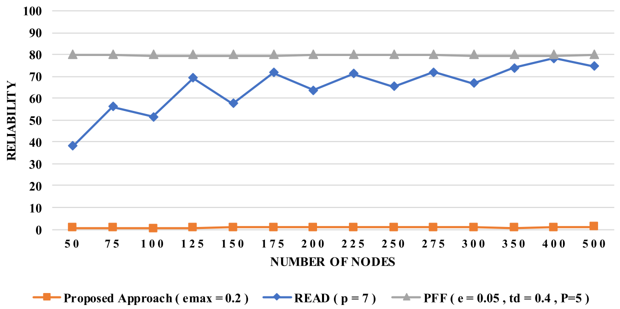

Fig. 16. Data accuracy of proposed approach, REDA, and PFF with $e_{max} = 0.2$.

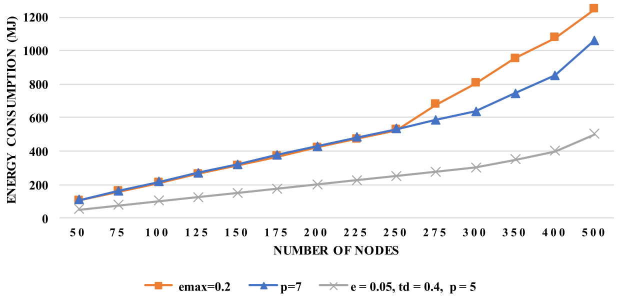

Fig. 18. Energy consumption of proposed approach, REDA, and PFF with $e_{max} = 0.2$.

As shown in Table 12 , the proposed approach shows energy consumptions ranging 105.08 mJ and 1252.15 mJ for $e_{max} = 0.1$, ranging between 104.75 mJ and 1247.62 mJ for $e_{max} = 0.2$, and ranging between 101.70 mJ and 1210.10 mJ. On the other hand, REDA achieves energy consumption in the range from 108.08 mJ to 1063.17 mJ for the pattern code (5), in the range from 107.02 mJ to 1058.80 mJ for the pattern code (7), and in the range from 105.94 mJ to 1054.39 mJ for the pattern code (10). PFF shows energy consumption ranging between 75.04 mJ and 750.11 mJ for parameters(p = 5, e = 0.03, td=0.35), ranging between 49.84 mJ and 498.50 mJ for parameters (p = 5, e = 0.05, td=0.4) and ranging between 38.73 mJ and 386.89 mJ for parameters (p = 5, e = 0.1, td=0.5). From Figs. 17–19 it can be observed that the proposed approach and REDA have shown no significant changes as the e_{max} and p change. In contrast, PFF has shown significant improvement in energy consumption when there is an increase in td and e values. Although PFF has achieved better energy consumption performance, it has shown lower data reliability than the proposed approach.

Depending on results obtained, it can realize that the proposed approach outperforms PFF and REDA in terms of accuracy and spatial and temporal data redundancy reduction. REDA performs the aggregation at two levels. Firstly, it gives a pattern code for each sensor on the network based on its first reading. Then the later reading will not be sent to the chosen CH unless it is not in the range of its given pattern code. Secondly, at the CH level, the CH compares and finds the nodes with the same pattern codes and selects the node with the highest energy to transmit its data. The selected sensors will then send their actual readings to the CH, which will transmit them directly to the BS. Then, all sensor nodes update their energy levels, and according to that, a new CH will be chosen depending on the level of nodes energy. This iteration operation leads to good data reduction at the sensor level

Table 12
Comparison of energy consumption of the proposed approach, REDA and PFF.

Number of nodes	Proposed approach			REDA			PFF		
	emax = 0.1	emax = 0.2	emax = 0.3	p = 5	p = 7	p = 10	e = 0.1, td = 0.5, p = 5	e = 0.05, td = 0.4, p = 5	e = 0.03, td = 0.35, p = 5
50	105.08	104.75	101.70	108.08	107.02	105.94	75.04	49.84	38.73
75	157.62	157.10	154.93	163.92	161.69	159.42	112.46	74.77	58.03
100	210.16	209.45	206.57	215.22	213.46	211.63	150.00	99.74	77.36
125	262.71	261.88	258.22	273.86	269.85	265.84	187.32	124.59	96.58
150	315.25	314.23	309.82	321.60	319.36	317.09	224.75	149.50	115.92
175	367.80	366.63	361.51	380.07	375.75	371.32	262.14	174.38	135.28
200	420.35	418.97	413.11	428.40	425.58	422.68	299.68	199.28	154.66
225	472.89	471.30	464.61	484.68	480.54	476.33	337.30	224.15	173.92
250	525.44	523.66	516.20	533.85	530.83	527.81	374.76	249.11	193.36
275	678.62	676.17	666.56	590.02	585.74	581.43	412.25	274.05	212.76
300	806.68	803.65	791.77	639.32	636.16	633.01	449.70	299.00	232.12
350	955.80	952.06	938.14	748.54	743.89	739.29	524.76	348.76	270.72
400	1079.79	1075.60	1059.97	857.46	851.48	845.49	599.67	398.59	309.50
500	1252.15	1247.62	1210.10	1063.17	1058.80	1054.39	750.11	498.50	386.89

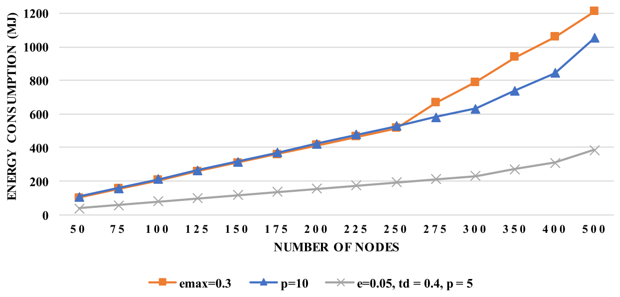

Fig. 19. Energy consumption of proposed approach, REDA, and PFF with $e_{max=0.3}$.

while it keeps the same pattern code. But the removal of the nodes with similar pattern codes at the CH decries the accuracy due to increasing the deviation of the actual data to aggregated data.

In the same way, PFF performs its aggregation process depending on two levels of sensor level and cluster head level. Firstly, at the sensor level, calculating the similarity of the current readings will take place to cached the readings at every period. The current reading will be removed, and the weight of similar readings will increase by 1. if the difference calculated is equal or less than the given threshold. So, a substantial data redundancy reduction will be made at the sensor level due to this deletion operation. Secondly, at the cluster head, number of sets will be received at the end of the user-defined period of time. Similarly, the cluster head will delete one of the similar sets if the similarity is in the range of given distance. Same as REDA, the deletion operation will lead to an increase between the deviation of the actual data and aggregated data. Nevertheless, the proposed approach has realized lower data redundancy and higher data accuracy due to the fact it omits the transmission of redundant data and estimates them at the sink level for keeping quality and data accuracy.

Once it comes to the energy consumption term, both REDA and PFF outperform the proposed approach due to several reasons. Firstly both of REDA and PFF have applied deletion operation in two levels sensor and cluster head levels. Secondly, due to the fluctuation of the acquired data of some sensors starting from the network with 275 sensor nodes. In other words, the total number of data transmission to the sink of the proposed approach is higher than PFF and REDA.

6. Conclusion

In this paper, we propose an approach for WSN data redundancy reduction. The proposed approach is relying on two levels, namely, end-node level redundancy reduction and sink level redundancy reduction. The first level is dedicated to temporal data redundancy reduction depending on two techniques, namely, data change detection and the deviation (>user defined threshold) of real readings from their estimated values. The essential goal is to reduce the temporal data redundancy, which in turn reduces the data transmission to the sink. The second level is sink level redundancy reduction, and it aims to reduce the spatial data redundancy by applies two techniques, namely, sink level grouping algorithm and sink level aggregation algorithm. The proposed approach is compared with two different aggregation approaches, namely REDA and PFF, with networks of various sizes. Based on the obtained results, the proposed approach has shown the highest efficiency in spatial and temporal redundancy reduction with achieving the right level of data accuracy. Although the counterparts outperform the proposed approach in terms of energy efficiency, the proposed approach achieves an acceptable level of energy efficiency up to 51.6%. It is worthy to notice that both REDA and PFF aggregation approaches involve eliminating data at the aggregation process. The proposed approach adapted a data aggregation technique that is based on a statistical model, a mean of aggregated value for its low computational load, and to maintain data reliability. Even though the proposed approach presents a good accuracy, reliability, and data redundancy reduction, future work will investigate the anomaly and faulty data detection.

CRediT authorship contribution statement

Zaid Yemeni: Conceived the presented idea, Methodology, and the original draft, Writing - review & editing the manuscript, Worked as a coordinator to incorporate the contributions of co-authors into the paper. **Haibin Wang:** Direct supervisor of the whole project, Devised the project and the main conceptual ideas. **Waleed M. Ismael:** Wrote most of the codes in this manuscript, Writing and revising the original draft. **Yanan Wang:** Helped in writing the portion of the code and wrote the corresponding sections, Reviewed and edited the manuscript several times. **Zhengming Chen:** Formulated the comparison tables and wrote the introduction section.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is in part supported by the Fundamental Research Funds for the Central Universities, China (B200202216) and in part supported by Innovation Foundation of Radiation Application, China Institute of Atomic Energy (KFZC2020010401).

References

[1] M.A. Mahmood, W.K. Seah, I. Welch, Reliability in wireless sensor networks: A survey and challenges ahead, Comput. Netw. 79 (2015) 166–187.

[2] Y.-G. Yue, P. He, A comprehensive survey on the reliability of mobile wireless sensor networks: Taxonomy, challenges, and future directions, Inf. Fusion 44 (2018) 188–204.

[3] A. Cruz, J.P. Lousado, A survey on wearable health monitoring systems, in: 2018 13th Iberian Conference on Information Systems and Technologies (CISTI), IEEE, 2018, pp. 1–6.

[4] A. Adeel, M. Gogate, S. Farooq, C. Ieracitano, K. Dashtipour, H. Larijani, A. Hussain, A survey on the role of wireless sensor networks and IoT in disaster management, in: Geological Disaster Monitoring Based on Sensor Networks, Springer, 2019, pp. 57–66.

[5] B. Rashid, M.H. Rehmani, Applications of wireless sensor networks for urban areas: A survey, J. Netw. Comput. Appl. 60 (2016) 192–219.

[6] A. Papageorgiou, B. Cheng, E. Kovacs, Real-time data reduction at the network edge of internet-of-things systems, in: 2015 11th International Conference on Network and Service Management (CNSM), IEEE, 2015, pp. 284–291.

[7] E. Vahedi, M. Bayat, M.R. Pakravan, M.R. Aref, A secure ecc-based privacy preserving data aggregation scheme for smart grids, Comput. Netw. 129 (2017) 28–36.

[8] Y. Fathy, P. Barnaghi, R. Tafazolli, An adaptive method for data reduction in the internet of things, in: 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), IEEE, 2018, pp. 729–735.

[9] L. Feng, P. Kortoçi, Y. Liu, A multi-tier data reduction mechanism for IoT sensors, in: Proceedings of the Seventh International Conference on the Internet of Things, 2017, pp. 1–8.

[10] K. Guleria, A.K. Verma, Comprehensive review for energy efficient hierarchical routing protocols on wireless sensor networks, Wirel. Netw. 25 (3) (2019) 1159–1183.

[11] Z. Yemeni, J. Shu, X. Zhang, L. Liu, A dbn approach to predict the link in opportunistic networks, in: Recent Developments in Intelligent Computing, Communication and Devices, Springer, 2019, pp. 575–587.

[12] A. Hawbani, X. Wang, A. Abudukelimu, H. Kuhlani, Y. Al-sharabi, A. Qarariyah, A. Ghannami, Zone probabilistic routing for wireless sensor networks, IEEE Trans. Mob. Comput. 18 (3) (2018) 728–741.

[13] H. Harb, A. Makhoul, S. Tawbi, R. Couturier, Comparison of different data aggregation techniques in distributed sensor networks, IEEE Access 5 (2017) 4250–4263.

[14] G. Han, L. Liu, J. Jiang, L. Shu, G. Hancke, Analysis of energy-efficient connected target coverage algorithms for industrial wireless sensor networks, IEEE Trans. Ind. Inf. 13 (1) (2015) 135–143.

[15] L. Cheng, J. Niu, C. Luo, L. Shu, L. Kong, Z. Zhao, Y. Gu, Towards minimum-delay and energy-efficient flooding in low-duty-cycle wireless sensor networks, Comput. Netw. 134 (2018) 66–77.

[16] J. Ludeña-Choez, J.J. Choquehuanca-Zevallos, E. Mayhua-López, Sensor nodes fault detection for agricultural wireless sensor networks based on nmf, Comput. Electron. Agric. 161 (2019) 214–224.

[17] P. Patil, U. Kulkarni, Svm based data redundancy elimination for data aggregation in wireless sensor networks, in: 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), IEEE, 2013, pp. 1309–1316.

[18] A. Hawbani, X. Wang, Y. Sharabi, A. Ghannami, H. Kuhlani, S. Karmoshi, Lora: Load-balanced opportunistic routing for asynchronous duty-cycled wsn, IEEE Trans. Mob. Comput. 18 (7) (2018) 1601–1615.

[19] S. Randhawa, S. Jain, Energy-efficient fuzzy-logic-based data aggregation in wireless sensor networks, in: Information and Communication Technology for Sustainable Development, Springer, 2020, pp. 739–748.

[20] W.M. Ismael, M. Gao, A.A. Al-Shargabi, A. Zahary, An in-networking double-layered data reduction for internet of things (iot), Sensors 19 (4) (2019) 795.

[21] A.A. Agarkar, M. Karyakarte, H. Agrawal, Post quantum security solution for data aggregation in wireless sensor networks, in: 2020 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2020, pp. 1–8.

[22] Y. Wang, H. Chen, X. Wu, L. Shu, An energy-efficient sdn based sleep scheduling algorithm for wsns, J. Netw. Comput. Appl. 59 (2016) 39–45.

[23] S. Chen, C. Zhao, M. Wu, Z. Sun, H. Zhang, V.C. Leung, Compressive network coding for wireless sensor networks: Spatio-temporal coding and optimization design, Comput. Netw. 108 (2016) 345–356.

[24] J. Lei, H. Bi, Y. Xia, J. Huang, H. Bae, An in-network data cleaning approach for wireless sensor networks, Intell. Autom. Soft Comput. 22 (4) (2016) 599–604.

[25] V. Sharma, S. Kumar, S. Bhushan, An overview of data redundancy reduction schemes in wsns, in: 2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA)(Fall), IEEE, 2017, pp. 1–8.

[26] B. Pourghebleh, N.J. Navimipour, Data aggregation mechanisms in the internet of things: A systematic review of the literature and recommendations for future research, J. Netw. Comput. Appl. 97 (2017) 23–34.

[27] N.-T. Nguyen, B.-H. Liu, V.-T. Pham, Y.-S. Luo, On maximizing the lifetime for data aggregation in wireless sensor networks using virtual data aggregation trees, Comput. Netw. 105 (2016) 99–110.

[28] F. Giroire, J. Moulierac, T.K. Phan, F. Roudaut, Minimization of network power consumption with redundancy elimination, Comput. Commun. 59 (2015) 98–105.

[29] S. Yadav, R.S. Yadav, Redundancy elimination during data aggregation in wireless sensor networks for iot systems, in: Recent Trends in Communication, Computing, and Electronics, Springer, 2019, pp. 195–205.

[30] S. Khriji, G.V. Raventos, I. Kammoun, O. Kanoun, Redundancy elimination for data aggregation in wireless sensor networks, in: 2018 15th International Multi-Conference on Systems, Signals & Devices (SSD), IEEE, 2018, pp. 28–33.

[31] H. Çam, S. Özdemir, P. Nair, D. Muthuavinashiappan, H.O. Sanli, Energy-efficient secure pattern based data aggregation for wireless sensor networks, Comput. Commun. 29 (4) (2006) 446–455.

[32] H.O. Sanli, S. Ozdemir, H. Cam, Srda: secure reference-based data aggregation protocol for wireless sensor networks, in: IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall. 2004, Vol. 7, IEEE, 2004, pp. 4650–4654.

[33] K. Khedo, R. Doomun, S. Aucharuz, et al., Reada: Redundancy elimination for accurate data aggregation in wireless sensor networks, Wirel. Sensor Netw. 2 (04) (2010) 300.

[34] H. Li, K. Lin, K. Li, Energy-efficient and high-accuracy secure data aggregation in wireless sensor networks, Comput. Commun. 34 (4) (2011) 591–597.

[35] S. Randhawa, S. Jain, Data aggregation in wireless sensor networks: Previous research, current status and future directions, Wirel. Pers. Commun. 97 (3) (2017) 3355–3425.

[36] D.C. Mocanu, M.T. Vega, A. Liotta, Redundancy reduction in wireless sensor networks via centrality metrics, in: 2015 IEEE International Conference on Data Mining Workshop (ICDMW), IEEE, 2015, pp. 501–507.

[37] M. Kumar, K. Dutta, Ldat: Lftm based data aggregation and transmission protocol for wireless sensor networks, J. Trust Manage. 3 (1) (2016) 2.

[38] T.-c. Fu, Y.-k. Hung, F.-l. Chung, Improvement algorithms of perceptually important point identification for time series data mining, in: 2017 IEEE 4th International Conference on Soft Computing & Machine Intelligence (ISCMI), IEEE, 2017, pp. 11–15.

[39] U. Jugel, Z. Jerzak, G. Hackenbroich, V. Markl, Vdda: automatic visualization-driven data aggregation in relational databases, VLDB J. 25 (1) (2016) 53–77.

[40] S. Kumar, V.K. Chaurasiya, A strategy for elimination of data redundancy in internet of things (iot) based wireless sensor network (wsn), IEEE Syst. J. 13 (2) (2018) 1650–1657.

[41] A. Al Sayyid, H. Harb, M. Ruiz, L. Velasco, Zizo: A zoom-in zoom-out mechanism for minimizing redundancy and saving energy in wireless sensor networks, IEEE Sens. J. (2020).

[42] G. Ahmed, X. Zhao, M.M.S. Fareed, M.R. Asif, S.A. Raza, Data redundancy-control energy-efficient multi-hop framework for wireless sensor networks, Wirel. Pers. Commun. 108 (4) (2019) 2559–2583.

[43] R. Maivizhi, P. Yogesh, Spatial correlation based data redundancy elimination for data aggregation in wireless sensor networks, in: 2020 International Conference on Innovative Trends in Information Technology (ICITIIT), IEEE, 2020, pp. 1–5.

[44] P.R. Coppin, M.E. Bauer, Digital change detection in forest ecosystems with remote sensing imagery, Remote Sens. Rev. 13 (3–4) (1996) 207–234.

[45] A. Benayache, A.B. Benayache, How to calculate total energy consumption using cooja, 2017, URL https://stackoverflow.com/questions/45644277/how-to-calculate-total-energy-consumption-using-cooja.

Zaid Yemeni He received his Bachelor of Science (BS.c), under the Faculty of Computer Science and Engineering, from Al-Ahgaff University, Yemen, in 2010. He completed his Master of Science (MS.c), in field of IoT, from Nanchang Hangkong University in 2018. Recently, he joined the Faculty of Information and Communication Engineering, Hohai University to continue his study to Doctor of Philosophy (Ph.D.) in field of IoT. His research interests include WSN reliability, opportunistic networks and deep learning.

Haibin Wang used to do Ph.D. and worked as a Postdoctoral Fellow at the University of Saskatchewan, Canada. He is currently the Vice Head and an Associate Professor with the Department of Microelectronics, Hohai University. He has authored more than 20 journal articles and more than ten patents and software copyright. His research interests include, but are not limited to, radiation effects in electronic circuits and systems, and deep learning reliability. He is a Technical Program Committee Member of IEEE IRPS/NSREC/ESREF conferences, a Reviewer of IEEE TNS/VLSI journals, and a Reviewer of NSFC grants.

Waleed M. Ismaeel He received his BS.c in Computer Science from Thamar University, Yemen, in 2006. In 2009, he received a postgraduate diploma in Geoinformatics from ITC institute, Hollande. He completed his MS.c in Geoinformatics, Osmania University, India. Now, he is pursuing his Ph.D. in Information and Communication Engineering, majoring in IoT Engineering. His research interests include WSN reliability, Data fusion, Geoinformatics, and deep learning.

Wang Yanan is studying at Hohai University as an undergraduate student in the program of IoT Engineering. He has put a lot of effort in computer programming.

Zhengming Chen is currently a full professor with the College of Internet of Things Engineering, Hohai University. He received his bachelor and Ph.D. degrees from Zhejiang University in 1987 and 2001, respectively. His research interests include deep learning, digital modeling, simulation and visualization of IoT devices, digital geometric processing technology in digital orthopedics.