

Article

# Data Aggregation Based on Overlapping Rate of Sensing Area in Wireless Sensor Networks

Xiaolan Tang <sup>1</sup>, Hua Xie <sup>1</sup>, Wenlong Chen <sup>1,\*</sup>, Jianwei Niu <sup>2,3</sup> and Shuhang Wang <sup>4</sup>

<sup>1</sup> College of Information Engineering, Capital Normal University, Beijing 100048, China; tangxl@cnu.edu.cn (X.T.); 2151002064@cnu.edu.cn (H.X.)

<sup>2</sup> State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China; niujianwei@buaa.edu.cn

<sup>3</sup> Research Institute of Beihang University in Shenzhen, Shenzhen 518057, China

<sup>4</sup> Schepens Eye Research Institute, Harvard Medical School, Boston, MA 02114, USA; shuhang\_wang@meei.harvard.edu

\* Correspondence: chenwenlong@cnu.edu.cn; Tel.: +86-10-6890-1370 (ext. 232)

Received: 26 April 2017; Accepted: 20 June 2017; Published: 29 June 2017

**Abstract:** Wireless sensor networks are required in smart applications to provide accurate control, where the high density of sensors brings in a large quantity of redundant data. In order to reduce the waste of limited network resources, data aggregation is utilized to avoid redundancy forwarding. However, most of aggregation schemes reduce information accuracy and prolong end-to-end delay when eliminating transmission overhead. In this paper, we propose a data aggregation scheme based on overlapping rate of sensing area, namely AggOR, aiming for energy-efficient data collection in wireless sensor networks with high information accuracy. According to aggregation rules, gathering nodes are selected from candidate parent nodes and appropriate neighbor nodes considering a preset threshold of overlapping rate of sensing area. Therefore, the collected data in a gathering area are highly correlated, and a large amount of redundant data could be cleaned. Meanwhile, AggOR keeps the original entropy by only deleting the duplicated data. Experiment results show that compared with others, AggOR has a high data accuracy and a short end-to-end delay with a similar network lifetime.

**Keywords:** wireless sensor networks; data aggregation; overlapping rate of sensing area; data accuracy

## 1. Introduction

Wireless sensor networks (WSNs) consist of a large quantity of sensor nodes to offer a variety of services, such as environmental monitoring and security surveillance [1,2]. Nowadays, WSNs are considered as one of the most promising technologies for cyber manufacturing systems in Industrial Internet of Things (IIoT) [3]. In smart factories, WSNs serve for intelligent industrial control applications in harsh environments [4,5]. In order to provide highly reliable and realtime transmission, sensor nodes are often densely distributed in monitoring areas. However, the high density of node deployment causes lots of redundant data, and hence their forwarding brings in a large waste of the limited power and bandwidth, resulting in low energy efficiency and short network lifetime.

In order to avoid the transmissions of redundant information, data aggregation is required in WSNs. In most of aggregation schemes, the whole network is separated into several areas like grids according to the geographical coordinates, and then the data collected by sensors in each area are aggregated by a particular node [6]. Because of the possible random distribution of nodes as well as the fixed size and shape of the aggregation area, the similarity of data collected by different sensors in one area is not close, which affects the performance of aggregation. Additionally, since there may be multiple hops from an ordinary node to an aggregation node, the redundant data might be forwarded

for several hops and hence lead to a high energy cost. Besides, an aggregation node has to wait for collecting all sensor data in its area before aggregation, which results in a long end-to-end delay [7,8].

In this paper, we propose a novel data aggregation scheme based on overlapping rate of sensing area in WSNs, named AggOR, to achieve energy-efficient data collection and keep high data accuracy. With respect to the sensing ranges, several nodes construct a gathering area if their overlapping rates of sensing area are no less than a preset threshold. In this way, the data in a gathering area have relatively high correlation and hence would be aggregated efficiently to eliminate redundancy. Moreover, there are only one or two levels of nodes in a gathering area. Usually a lower-level node transfers its sensing data to an upper-level node which aggregates and then delivers data to the sink. Therefore, redundant data are removed immediately after one-hop forwarding, which prohibits more energy consumption of redundancy relay. For further energy saving, an appropriate neighbor node at the same level could also be selected as the aggregation node if subject to particular conditions in the aggregation rules.

The main advantages of our scheme are listed below. (1) Construct gathering areas according to the overlapping rate of sensing area. It helps to remove a large quantity of duplicated data and keep almost all entropy of the original information; (2) Three aggregation rules take full advantage of data aggregation by selecting aggregating nodes from candidate parent nodes or appropriate neighbor nodes. They limit the hops of redundancy forwarding and decrease the transmission overhead as much as possible; (3) A large quantity of experiments show that AggOR scheme keeps a high accuracy of data, improves energy efficiency and achieves a quick data collection.

The rest of this paper is organized as follows. In Section 2, some related work on data aggregation in WSNs is discussed. Definitions and aggregation rules used in AggOR are introduced in Section 3, and Section 4 details the implementation of AggOR. Experimental results are analyzed in Section 5, and Section 6 concludes this paper.

## 2. Related Work

Data aggregation is utilized in WSNs to diminish the resource consumption of redundant data when delivering information from sensor nodes to the sink. In specific, data aggregation schemes could be classified into three kinds, i.e., tree-based aggregation [9–12], hybrid aggregation [13], and cluster-based aggregation [14–22].

In classical tree-based data aggregation schemes, a spanning tree rooted at the sink is constructed firstly, and then data are forwarded from leaves to root along the paths in the tree. In Tiny AGgregation (TAG) [10], after leaf nodes send their own data to their parent, the parent node aggregates data from its children and delivers the aggregated data to the root. Obviously, TAG is inefficient in case of dynamic topologies or link/device failures. In [11], Deligiannakis et al. propose an aggregation tree construction/reorganization algorithm to minimize energy cost. By calculating and sending a small set of intuitive statistics, a parent node may be substituted by one of its sibling nodes based on attachment cost. In [12], an adaptive spanning tree algorithm (AST) is proposed, which adaptively builds and adjusts an aggregation spanning tree. Owing to the strategies of random waiting times and alternative father nodes, AST establishes a relatively balanced spanning tree with flexible adjustments. Considering that a single packet, as the output of aggregation algorithm at a given level of the tree, may stand for all the data coming from a subtree, if it is lost, the entropy from this subtree might be lost as well.

As a typical hybrid data aggregation scheme, Tributary-Delta [13] combines the advantages of tree and multi-path by implementing them simultaneously in different regions of the network. It supports region adjustment in response to network condition changes, and determines the number of useful aggregates in the scenario. However, it may have a high overhead because of frequent update of the data gathering structure.

Compared with tree-based and hybrid schemes, cluster-based aggregation schemes usually have good scalability and high energy efficiency [14,15]. Considering that the cluster heads, which are close

to the sink, relay data for others, in [16], Li et al. propose an energy-efficient unequal clustering scheme (EEUC). Cluster heads are elected by localized competition, and the competition range becomes small when it is near the base station. Therefore, those clusters closer to the sink have smaller sizes than others, and the energy consumption of cluster heads is balanced. Even though, the cluster maintenance is somewhat difficult.

In recent years, some cluster-based schemes analyze various factors to select a cluster head from several candidates. In DHCR [17], energy consumption, adjustment degree and exact distance from sensors to the base station are three main parameters for cluster head selection. Multi-hop routing and clustering are combined to decrease the number of control packets. In [18], Leu et al. propose REAC-IN to evenly distribute cluster heads based on the residual energy of each sensor and the average energy of sensors in the cluster. Together with isolated node checking considering power and distance, REAC-IN improves the cluster head selection process and avoids node isolation. To save energy for cluster reformation, Yi and Yang propose Hamilton energy-efficient routing protocol (HEER) [19]. Members in each cluster are linked on a Hamilton Path, and take turns to work as cluster head. In this way, no cluster reformation is required. However, the clusters are formed like LEACH in the first round, which may cause energy hole problem; the condition that all members in a cluster could communicate with each other is strict; the intra-cluster nodes in the Hamilton Path transmitting data in turn prolong the end-to-end delay.

Moreover, some existing studies exploit spatial correlation to set aggregation areas. In YEAST algorithm, those nodes detecting the same event are grouped in a cluster and the cluster head is the node closest to the sink [20]. The cluster is divided into spatially correlated cells, and only one node within each cell transfers its data to its cluster head, which stand for all the sensing data in this cell. Cells can be resized dynamically according to the application requirements. However, the bigger cells become, the less the entropy is. Since a representative node's sensing area could not cover the whole cell, YEAST causes low accuracy to some extent. For several synchronous events, DRINA [21] tends to maximize the number of aggregation nodes and decrease the overhead of control packets. Nodes sensing the same event form a cluster, and a route for a new event is connected with an already established route which has the shortest path between them. Experiment results show that DRINA has a high aggregation rate, and reliable data aggregation and transmission. Nevertheless, it increases the overload of nodes in existing routes, and thus leads to unbalanced energy consumption and even the energy-hole problem.

In the above aggregation schemes, those nodes located in a particular area usually compose a cluster. If the area is too large, the similarity of data gathered by member nodes is small; if the area is too small, the advantage of data aggregation is degraded. In order to balance aggregation efficiency and data accuracy, we explore the relation of sensing areas of nodes to deal with redundant data, and utilize a threshold of overlapping sensing area to guarantee high accuracy after data aggregation.

### 3. Network Model and Aggregation Rules

#### 3.1. Network Model

In this paper, we assume that all sensor nodes have the same sensing radius, denoted by  $R_S$ , and the same communication radius, denoted by  $R_C$  ( $R_C > 2R_S$ ) [23]. The data collected by sensor networks may be periodic sensing information, such as the average temperature, or information triggered by specific events, such as fire alerts. Our scheme focuses on the periodic data collection, and assumes that the amount of information collected is the same for all the nodes, which is a common assumption about data collection [24]. The data collected by each node is denoted by  $d$ . Since the sink usually has sufficient power, we only consider the energy cost of sensor nodes in our scheme. Because data sending (the amount of data sent out from node  $v_i$  is denoted by  $d_i^T$ ) and receiving (the amount of data received by node  $v_i$  is denoted by  $d_i^R$ ) consume most of the energy, small energy consumptions such as the cost of data processing are ignored. Therefore we focus on the transmission overhead in

AggOR. For simplicity, we assume that total energy consumption of node  $v_i$  is  $E_i^C = d_i^T E_T + d_i^R E_R$ , where  $E_T$  and  $E_R$  are the energy costs for sending and receiving per unit data, respectively. The primary symbols used in AggOR are listed in Table 1.

Table 1. Symbols.

Symbol	Description
$R_S$	Sensing radius of a sensor node.
$R_C$	Communication radius of a sensor node.
$E_T$	Consumed energy of sending per unit data.
$E_R$	Consumed energy of receiving per unit data.
$d$	The size of data collected by a sensor node.
$OR_{i,j}$	Overlapping rate of sensing area of two nodes $v_i$ and $v_j$ .
$OR_T$	Threshold of overlapping rate of sensing area.
$GA_i$	Gathering area with $v_i$ as the gathering node.
$GN$	Gathering node, which aggregates the data collected in a gathering area.
$CG_i$	Candidate gathering node set of $v_i$ .
$CP_i$	Candidate parent node set of $v_i$ , including all the upper-level nodes that could communicate with $v_i$ directly.
$N_i$	Neighbor node set of $v_i$ , which consists of the nodes at the same level that could communicate with $v_i$ directly.
$L_i$	Level of $v_i$ , and $L_i \in N^+$ .
$d_i^T$	The total amount of data transferred from $v_i$ .
$d_i^R$	The total amount of data received by $v_i$ .
$E_{GN_i}$	The energy consumed for delivering aggregated data from the gathering node $v_i$ .
$E_{GA_i}$	The total energy cost of the nodes in the gathering area $GA_i$ .
$E_i^k$	The total energy cost of transmitting data of node $v_k$ to the sink via another node $v_i$ .
$FN_n$	Free nodes at the level $n$ .

Before presenting the details of AggOR, several definitions are introduced as follows.

**Definition 1.** *Transmission hierarchy diagram: The diagram is a directed acyclic graph, including all the sensors, the possible communication paths, and the hierarchy levels (denoted by  $L$ ). It is similar to a tree structure rooted at the sink, but the parent node is not unique.*

For data collection in a dense network, we only care about those nodes which can communicate with the sink through one-hop or multi-hop transmissions. For two levels  $L_i$  and  $L_i + 1$  in the diagram,  $L_i$  is called upper-level and  $L_i + 1$  is called lower-level. In other words, the value of upper-level is smaller than that of lower-level. As shown in Figure 1, there are  $m$  sensor nodes in the network as well as  $v_0$  as the sink. The edges show the communication chances between nodes. Specifically, the solid lines indicate the possible parent-child relations between lower-level and upper-level nodes, while dotted lines show the neighbor relations between nodes at the same level.

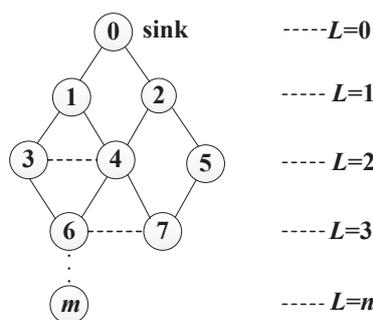


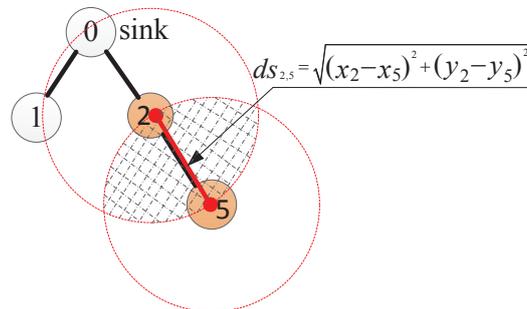
Figure 1. Transmission hierarchy diagram.

**Definition 2.** *Overlapping rate of sensing area,  $OR_{i,j}$ : The ratio of the overlapping sensing area of two nodes  $v_i$  and  $v_j$  to a node's entire sensing range ( $\pi R_S^2$ ).*

Furthermore, considering that the two nodes with a larger overlapped sensing area probably have a larger similarity of the collected data, we assume that the amount of duplicated data at two sensors is proportional to the overlapping rate of sensing area. Our aggregation only removes redundant data, and thus the entropy of the sensing data in the whole network is not lost. In other words, the aggregation works like a lossless compression approach whose compression ratio is the overlapping rate of sensing area.

In order to construct gathering areas, a threshold of overlapping rate of sensing area, denoted by  $OR_T$ , is utilized. The assignment of  $OR_T$  affects the aggregation efficiency. The smaller  $OR_T$  is, the larger the gathering area is, but the smaller the amount of duplicated data between nodes is. Further analysis of  $OR_T$  is in Section 5.3.2.

As shown in Figure 2, take the overlapping sensing area of  $v_2$  and  $v_5$  in Figure 1 as an example. Two dashed circles represent the sensing ranges of two nodes, respectively, and the hatched area is the overlapping sensing area, denoted by  $S_C$ . Therefore the overlapping rate of sensing area of  $v_2$  and  $v_5$  is computed by  $OR_{2,5} = \frac{S_C}{\pi R_S^2}$ . In order to calculate  $S_C$ , we denote the distance between  $v_i$  and  $v_j$  by  $ds_{i,j}$ . According to geometric theory, the overlapping sensing area is  $S_C = 2R_S^2 \arccos \frac{ds_{i,j}}{2R_S} - \frac{ds_{i,j}}{2} \sqrt{4R_S^2 - ds_{i,j}^2}$ . To remove the complicated calculation of inverse trigonometric functions, using curve fitting mechanism [25],  $S_C$  could be computed as  $S_C = 0.34ds_{i,j}^2 - 2.26R_S ds_{i,j} + \pi R_S^2$ . Therefore,  $OR_{i,j} = \frac{0.34ds_{i,j}^2}{\pi R_S^2} - \frac{2.26ds_{i,j}}{\pi R_S} + 1$ .



**Figure 2.** Overlapping rate of sensing area.

**Definition 3.** *Gathering area,  $GA$ : A gathering area is composed of a gathering node and several member nodes, and the overlapping rate of sensing area between the gathering node and each member node equals or is larger than  $OR_T$ .*

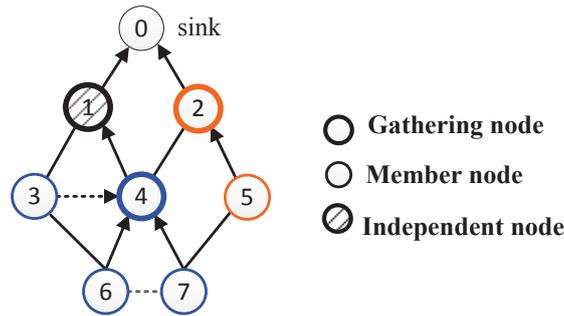
The gathering node is responsible for collecting and aggregating all the sensor data in the gathering area and then sending the result toward the sink, while the member node transfers its data to the gathering node. A gathering area with node  $v_i$  as its gathering node and  $v_j, \dots, v_k$  as its member nodes is expressed by  $GA_i = (v_i, \{v_j, \dots, v_k\})$ . One node belongs to at most one gathering area. If a node  $v_k$  does not find a gathering node,  $v_k$  turns into an independent node and forms a gathering area by itself as  $GA_k = (v_k, \emptyset)$ . If a node is not a gathering node, it is called non-gathering node; if a node does not join in a gathering area, it is called free node.

**Definition 4.** *Candidate parent nodes of  $v_i$ ,  $CP_i$ : A set consists of the nodes at level  $L_i - 1$  and in the communication range of  $v_i$ .*

**Definition 5.** *Neighbor nodes of  $v_i$ ,  $N_i$ : A set includes the nodes at level  $L_i$  which can communicate with  $v_i$ .*

**Definition 6.** Candidate gathering nodes of  $v_i$ ,  $CG_i$ : A set includes all the nodes which might be the gathering node of  $v_i$ . In AggOR,  $CG_i \subseteq CP_i \cup N_i$ .

As an instance, Figure 3 shows a transmission hierarchy diagram having three gathering areas, i.e.,  $GA_4 = (v_4, \{v_3, v_6, v_7\})$ ,  $GA_2 = (v_2, \{v_5\})$ , and  $GA_1 = (v_1, \emptyset)$ . The arrows indicate the directions of data transfers. Therefore,  $v_3, v_6$  and  $v_7$  send their data to  $v_4$  which aggregates these data with its own data, and then sends the results to the sink;  $v_5$  transfers its data to  $v_2$ , and  $v_2$  aggregates these data;  $v_1$  sends its data to the sink directly.



**Figure 3.** An instance of wireless sensor network (WSN) with three gathering areas.

### 3.2. Aggregation Rules

We have three aggregation rules for the gathering area construction. Although in general,  $E_R$  is slightly smaller than  $E_T$  which is relevant with the transmission distance, to simplify the explanations of these rules, we assume  $E_R = E_T = E$  and  $E \cdot d = e$ .

**Rule 1.** In gathering areas, the upper-level nodes have priorities over the lower-level nodes to be selected as gathering nodes.

We evaluate the validity of Rule 1 in an instance gathering area with an upper-level node  $v_j$  and a lower-level node  $v_i$ . Apparently,  $L_i = L_j + 1 \geq 2$ .

Case 1:  $v_i$  is the gathering node. After  $v_j$  joins the gathering area  $GA_i$ ,  $v_j$  as the member node sends  $d$  data to  $v_i$ . Thus  $d_j^R = 0$ , and  $d_j^T = d_i^R = d$ . After data aggregation,  $v_i$  transmits additional  $\gamma \cdot d$  ( $0 < \gamma = 1 - OR_{i,j} < 1$ ) data as well as its own data  $d$ . Hence,  $d_i^T = (1 + \gamma)d$ . Since these data need to be forwarded to the sink through  $L_i$  hops. The energy consumed for delivering aggregated data from the gathering node  $v_i$  is  $E_{GN_i} = (2L_i - 1)d_i^T \cdot E = (2L_i - 1)(1 + \gamma)e$ . Therefore, the total energy consumption of those nodes in the gathering area  $GA_i$  is computed by

$$\begin{aligned} E_{GA_i} &= d_j^T \cdot E + d_i^R \cdot E + E_{GN_i} \\ &= [(2L_i - 1)(1 + \gamma) + 2]e. \end{aligned} \quad (1)$$

Case 2:  $v_j$  is the gathering node. Similarly, the total energy cost of data delivery from those nodes in  $GA_j$  to the sink is

$$\begin{aligned} E_{GA_j} &= [(2L_j - 1)(1 + \gamma) + 2]e \\ &= [(2L_i - 3)(1 + \gamma) + 2]e. \end{aligned} \quad (2)$$

Obviously  $E_{GA_j} < E_{GA_i}$ . Because only replicated data are cleaned in aggregation, no matter which node is the gathering node, the quantities of information after aggregation are the same. As a result of a higher level of the gathering node, Case 1 has more energy consumed than Case 2. Thus selecting the gathering nodes from the upper-level nodes is better than from the lower-level nodes.

**Rule 2.** After the lower-level nodes complete the construction of gathering areas, the free nodes at the upper level firstly select their gathering nodes from the candidate parent nodes. If no suitable candidate parent node exists, then select from the neighbor nodes.

Assume that  $v_k$  (the current node) is selecting a gathering node.  $v_j$  is its candidate parent node, while  $v_i$  is a neighbor node of  $v_k$ . Hence  $L_k = L_i = L_j + 1$ .

Case 1:  $OR_{i,k} \geq OR_T$  and  $OR_{j,k} \geq OR_T$ . In this case,  $v_k$  select  $v_i$  or  $v_j$  to be its gathering node. If  $v_i$  is the gathering node of  $v_k$ ,  $v_i$  receives data of size  $d$  from  $v_k$  and transfers additional  $\alpha \cdot d$  ( $0 < \alpha = 1 - OR_{i,k} < 1$ ) data as well as its own data after aggregation. The total energy cost of transmitting data from  $v_k$  to the sink via  $v_i$  is

$$E_i^k = [(2L_k - 1)\alpha + 2]e. \tag{3}$$

Otherwise, if  $v_k$  chooses  $v_j$  as its gathering node, the total energy cost for data delivery from  $v_k$  to the sink via  $v_j$  is

$$E_j^k = [(2L_k - 3)\beta + 2]e, \tag{4}$$

where  $0 < \beta = 1 - OR_{j,k} < 1$ .

Comparing  $E_i^k$  and  $E_j^k$ , we know that if  $\alpha \geq \beta$  (in other words,  $OR_{i,k} \leq OR_{j,k}$ ), then  $E_i^k > E_j^k$ , and  $E_i^k$  is larger than  $E_j^k$  by at least  $2\beta e$ . When  $\alpha < \beta$ , considering  $L_k \geq 2$ , we obtain that if  $E_i^k \geq E_j^k$ , then  $\frac{\beta}{3} \leq \alpha < \beta$ . Accordingly, if  $\alpha < \frac{\beta}{3}$ ,  $E_i^k$  might be smaller than  $E_j^k$ . Note that to guarantee a relatively high data correlation,  $OR_T$  is often larger than 0.4 (as discussed in Section 5.3.2), and thus  $0 < \alpha, \beta < 0.6$ . If  $\alpha < \frac{\beta}{3} < 0.2$  ( $OR_{i,k} > 0.8$ ), then  $E_i^k$  has some probability to be smaller than  $E_j^k$ . In our experiments, it is rare to achieve this strict condition in networks. Therefore, in general,  $E_i^k > E_j^k$ . The gathering node selection from the candidate parent node is more energy-efficient than from the neighbor node.

For instance, we discuss about how  $v_3$  (the current node) selects its gathering node from  $v_1$  and  $v_4$ . In Figure 4,  $L_3 = 2$ ,  $OR_{1,3} = 0.5$ ,  $OR_{3,4} = 0.7$ . The total energy cost of transmitting data of  $v_3$  to the sink via  $v_1$  is  $E_1^3 = [(2L_3 - 3)(1 - OR_{1,3}) + 2]e = 2.5e$ , while that of selecting  $v_4$  as the gathering node of  $v_3$  is  $E_4^3 = [(2L_3 - 1)(1 - OR_{3,4}) + 2]e = 2.9e$ . Therefore,  $E_4^3 > E_1^3$ , which is consistent with above analysis.

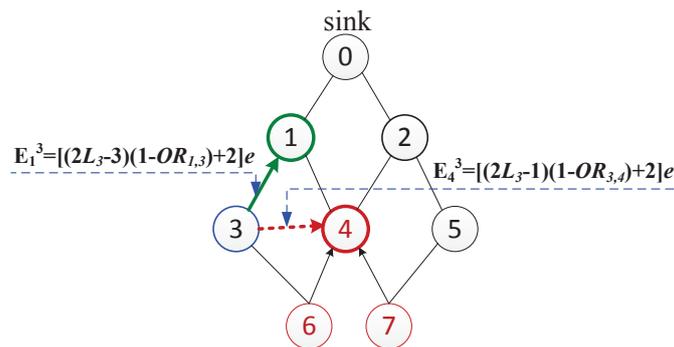


Figure 4. Gathering node selection for  $v_3$ .

Case 2:  $OR_{i,k} < OR_T$  and  $OR_{j,k} \geq OR_T$ .  $v_k$  chooses  $v_j$  as its gathering node, according to Rule 1. From Case 1 and Case 2, when  $v_k$  gets  $OR_{j,k} \geq OR_T$ ,  $v_k$  selects its gathering node from the candidate parent nodes, and does not need to calculate the overlapping rates of sensing area with its neighbor nodes.

Case 3:  $OR_{i,k} \geq OR_T$  and  $OR_{j,k} < OR_T$ . Node  $v_k$  has two ways to send its data. One option is that  $v_k$  becomes an independent node while  $v_j$  is its relay node, and the total energy consumption of transmitting data of  $v_k$  to the sink via  $v_j$  is

$$E_j^k = (2L_k - 1)e. \quad (5)$$

The other option is taking  $v_i$  as  $v_k$ 's gathering node, and the total energy cost is shown in Equation (3). Hence  $E_i^k - E_j^k = [2 - (2L_k - 1)(1 - \alpha)]e$ . If  $(2L_k - 1)OR_{i,k} > 2$ , then  $E_i^k < E_j^k$ , and  $v_k$  chooses  $v_i$  as its gathering node. In one word, if there is no candidate parent node  $v_j$  subject to  $OR_{j,k} \geq OR_T$ , a neighbor node  $v_i$  with  $OR_{i,k} \geq OR_T$  and  $(2L_k - 1)OR_{i,k} > 2$  is selected as the gathering node of  $v_k$ . Furthermore, if no such neighbor node exists,  $v_k$  becomes an independent node.

Case 4:  $OR_{i,k} < OR_T$  and  $OR_{j,k} < OR_T$ . Considering the weak similarity, it is not necessary to aggregate the data from these nodes. Therefore,  $v_k$  becomes an independent node, delivering its data to the sink without aggregation.

In conclusion, for free nodes, their candidate parent nodes have high priorities to be the gathering nodes. If all the candidate parent nodes have lower overlapping rates of sensing area than the threshold, then the neighbor nodes are considered to aggregate data.

**Rule 3.** *Data from every node is aggregated at most once, and in relay node selection, the non-gathering nodes take priorities over the candidate parent node with the most residual energy.*

Since data similarity is small between different gathering areas, data aggregation inter gathering areas probably has no significant advantages. Additionally, the data collection and processing for further aggregations may prolong the end-to-end delay. Consequently, in AggOR, taking into account the original data from each node, the aggregation executes at most once.

Obviously, the gathering nodes consume more power than the member nodes and the independent nodes, and thus are not suitable as relay nodes which need to contribute extra energy for data forwarding. Selecting non-gathering nodes as relays helps to balance the energy consumption in the entire network and prolong the network lifetime. If no non-gathering node exists, take the candidate parent node with the most remaining energy as the forwarder.

#### 4. Implementation of AggOR Scheme

The transmission hierarchy diagram of WSN is constructed based on hello messages exchange between sensor nodes. Hello message includes the sender's ID, coordinates, residual energy and the level. At the beginning, all sensor nodes initialize their levels as infinity, and the sink floods hello message including its level 0. Then other nodes update their levels after receiving hello messages. In specific, after receiving a hello message,  $v_i$  checks if the difference of its stored level  $L_i$  and the level in the hello message is larger than 1. If it is true,  $v_i$  updates  $L_i$  with the level in hello message plus 1, and then disseminates its hello message with the updated level; otherwise, if the level in the message is  $L_i - 1$ ,  $v_i$  adds the ID in the message to its candidate parent node set  $CP_i$ ; if the levels are the same,  $v_i$  inserts the ID to its neighbor node set  $N_i$ .

A topology under construction is illustrated in Figure 5 where the numbers are levels of nodes. Take node  $v_i$  as an example; its  $CP_i$  and  $N_i$  are illustrated.

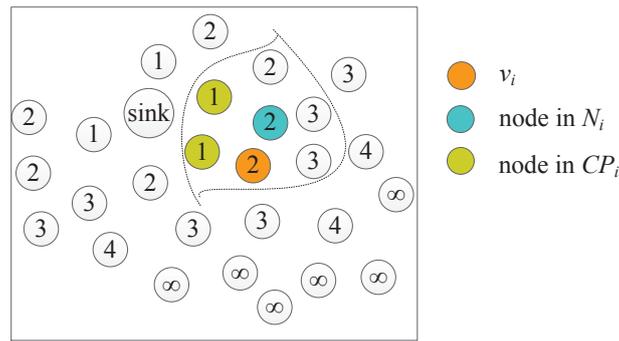


Figure 5. An instance topology under construction.

#### 4.1. Gathering Area Construction

After transmission hierarchy diagram is completed, sensor nodes start to form the gathering area in a distributed manner. Gathering area construction begins from the lower-level free nodes to the upper-level layer by layer and follows the three aggregation rules in Section 3.2. In the construction process, a free node may be chosen as a gathering node in a new gathering area, or join in an existing gathering area as a member node, or become an independent node. Take  $v_i$  as an instance; its gathering area construction algorithm is shown in Algorithm 1, a core of which is finding the candidate gathering node set as shown in Algorithm 2.

---

#### Algorithm 1: Gathering Area Construction.

---

**Input:** the transmission hierarchy diagram, a free node  $v_i$

**Output:**  $GA_i$

**if** a set of nodes  $M$  request to be member nodes of  $GA_i$  **then**

    | send replies to  $M$  to construct a gathering area  $GA_i = (v_i, M)$ ;

**else**

**if**  $L_i > 1$  **then**

        |  $CG_i \leftarrow$  Algorithm 2;

**if**  $CG_i = \emptyset$  **then**

            |  $v_i$  is an independent node,  $GA_i = (v_i, \emptyset)$ ;

**else**

            | select  $v_t \in CG_i$  with the maximum residual energy to be the gathering node of  $v_i$ ;

            | send the request to be a member of  $GA_t$ ;

**end**

**else**

        |  $v_i$  is an independent node,  $GA_i = (v_i, \emptyset)$ ;

**end**

**end**

**return**  $GA_i$ ;

---

The network is initialized that all the sensor nodes are free nodes and there are  $(n + 1)$  levels in the transmission hierarchy diagram. Levels are numbered as  $0, 1, 2, \dots, n$ . Then the nodes at level  $n$  start constructing gathering areas firstly. For a node  $v_i$  whose level is larger than 1, it calculates the overlapping rates of sensing area and obtains its candidate gathering node set  $CG_i$  through Algorithm 2. If  $CG_i$  is empty,  $v_i$  becomes an independent node; otherwise,  $v_i$  chooses the node  $v_t$  in  $CG_i$  which has the most residual energy as its gathering node, and sends the request to be a member node in the gathering area  $GA_t$ . For the free nodes at level 1, it is unnecessary to select the sink as gathering node, and their overlapping rates of sensing area with their neighbor nodes cannot be larger than 2 (Rule 2). Therefore each free node at level 1 turns into an independent node.

**Algorithm 2:** Finding Candidate Gathering Node Set.

---

```

Input:  $CP_i, N_i$ 
Output:  $CG_i$ 
initialize:  $CG_i \leftarrow \emptyset$ ;
for  $\forall v_j \in CP_i$  do
    if  $\exists OR_{i,j} \geq OR_T$  then
         $CG_i \leftarrow CG_i \cup \{v_j\}$ ;
    end
end
if  $CG_i = \emptyset$  then
    for  $\forall v_k \in N_i$  do
        if  $OR_{i,k} \geq OR_T$  and  $(2L_i - 1)OR_{i,k} > 2$  then
             $CG_i \leftarrow CG_i \cup \{v_k\}$ ;
        end
    end
end
return  $CG_i$ ;

```

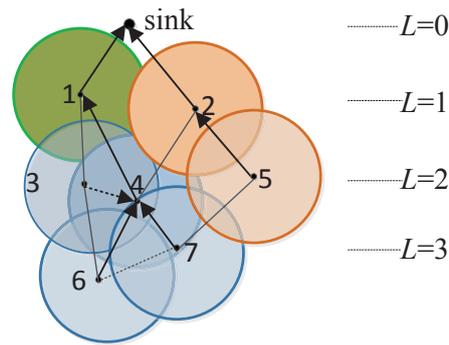
---

Algorithm 2 returns  $CG_i$ , as the set of candidate gathering nodes of  $v_i$ . For every candidate parent node  $v_j$ , if  $OR_{i,j} \geq OR_T$ ,  $v_j$  is included in  $CG_i$ . If there is no candidate gathering node selected from the candidate parent nodes,  $v_i$  calculates the overlapping rates of sensing area with its neighbor nodes. If a neighbor node  $v_k$  satisfies  $OR_{i,k} \geq OR_T$  and  $(2L_i - 1)OR_{i,k} > 2$ ,  $v_k$  is included in  $CG_i$ .

#### 4.2. Data Routing

In the process of data routing, member nodes send data to their gathering nodes through one hop transmission, while gathering nodes aggregate data collected in their gathering areas and then send them to the sink through the energy-efficient paths. Additionally, independent nodes send their own data to the sink without aggregation. The energy-efficient paths are established according to Rule 3, in which the non-gathering nodes are the first choice for relay nodes and then the candidate parent nodes with the most residual energy are selected as forwarders.

An instance of the construction of gathering areas and the data routing is depicted in Figure 6, which is the part in dotted circle of Figure 5. We suppose that  $v_4$  has more residual energy than  $v_3$ , and  $OR_{3,4} = 0.7 > OR_T$ . All the nodes in the network are initialized as free nodes, i.e.,  $FN_3 = \{v_6, v_7\}$ ,  $FN_2 = \{v_3, v_4, v_5\}$  and  $FN_1 = \{v_1, v_2\}$ . At the beginning,  $v_6$  and  $v_7$ , at the highest level 3, calculate the overlapping rates of sensing area with their candidate parent nodes. We get  $OR_{4,6} > OR_{3,6} > OR_T$  and  $OR_{4,7} > OR_T > OR_{5,7}$ . Thus  $CG_6 = \{v_3, v_4\}$  and  $CG_7 = \{v_4\}$ . Due to more residual energy,  $v_4$  is selected by  $v_6$  and  $v_7$  as their gathering node,  $GA_4 = (v_4, \{v_6, v_7\})$ . Then  $FN_3 = \emptyset$  and  $FN_2 = \{v_3, v_5\}$ . Next the free nodes  $v_3$  and  $v_5$  at level 2 construct their gathering areas. Similarly, regarding the candidate parent nodes,  $OR_{1,3} < OR_T$ ,  $OR_{2,3} < OR_T$  and  $OR_{2,5} = OR_T$ . Therefore,  $CG_3 = \emptyset$  and  $CG_5 = \{v_2\}$ . Node  $v_5$  chooses  $v_2$  as its gathering node,  $GA_2 = (v_2, \{v_5\})$ . Since  $CG_3 = \emptyset$ ,  $v_3$  further calculates overlapping rate of sensing area with its neighbor node  $v_4$ , and gets  $(2L_3 - 1) \times OR_{3,4} = 3 \times 0.7 > 2$ . Because there is no other neighbor node,  $CG_3 = \{v_4\}$ , and  $v_3$  chooses  $v_4$  as its gathering node,  $GA_4 = (v_4, \{v_3, v_6, v_7\})$ . After that,  $FN_2 = \emptyset$  and  $FN_1 = \{v_1\}$ .  $v_1$  at the level 1 becomes an independent node, and  $FN_1 = \emptyset$ . For the data transmission,  $v_3$ ,  $v_6$  and  $v_7$  send their data to  $v_4$ ,  $v_4$  aggregates data in  $GA_4$  and then chooses  $v_1$  (non-gathering node) to relay its data to the sink. Node  $v_5$  sends data to  $v_2$  which aggregates and forwards data to the sink, while  $v_1$  sends its data to the sink directly.



**Figure 6.** An example of AggOR implementation.

### 4.3. Complexity Analysis

In order to testify the validity and efficiency of AggOR, we analyze its complexity in terms of computation, message and storage complexity. The computation complexity of Algorithm 2 (finding candidate gathering node set) is  $O(\max_{v_i \in [1, m]} (|CP_i \cup N_i|))$ , where  $m$  is the number of sensor nodes in the network. It means in the worst case, a node  $v_i$  visits all its candidate parent nodes  $CP_i$  and its neighbor nodes  $N_i$  to find its candidate gathering nodes  $CG_i$ . Accordingly the computation complexity of Algorithm 1 (gathering area construction) is  $O(\max_{v_i \in [1, m]} |CP_i \cup N_i| + |CG_i|)$ . Since the number of elements in  $CP_i$ ,  $N_i$  and  $CG_i$  are all less than  $m$ , the complexity of our algorithms is  $O(m)$ .

In the process of transmission hierarchy diagram construction, sink starts flooding hello message and other nodes broadcast it after reception. Thus the control message cost is  $m$ . When constructing gathering areas, node  $v_i$  sends join message to its potential gathering node, and after receiving acceptance message from it,  $v_i$  sends acknowledgement message to join the gathering area. Hence in this process the message cost is  $3m$ . Overall, the message complexity of AggOR is  $O(m)$ .

For the control information, since each node stores its overlapping rates of sensing area with nearby nodes, candidate parent node set, neighbor node set, level and remaining energies of its candidate gathering nodes, the storage complexity is  $O(m)$ . In addition, for the data packets, every member node only carries its own data, while each gathering node caches the data from its member nodes. Considering that data aggregation is implemented at one-hop distance, the number of data packets collected by a gathering node is usually far less than  $m$ , with the storage complexity  $O(m)$ .

To sum up, compared with EEUC, whose message complexity is  $O(m)$ , and HEER, whose message complexity is  $O(m)$  and computation complexity is exponential, our distributed scheme AggOR has relatively low complexities of computation, message and storage.

## 5. Performance Evaluation

### 5.1. Network Configurations

We evaluate the performance of AggOR scheme on OPNET Modeler [26] network simulation platform. The network configurations are listed in Table 2. Note that sensor nodes are evenly distributed in the monitoring field.

We select a typical scheme EEUC and a newly proposed scheme HEER to be our comparisons. EEUC is a distributed cluster mechanism where cluster heads are elected by localized competition. Through a function of the competition range  $R_{comp}$  which is decided by the distance to the base station, several tentative cluster heads are elected to compete for final cluster heads. After the cluster head selection, other nodes join in their closest cluster heads. HEER, as a chain-based protocol, constructs clusters like LEACH, and establishes a Hamilton Path in each cluster to set an order for sensors to transmit data. In order to evaluate different gathering node selection methods, we take a variation of

AggOR as comparison, in which the gathering nodes are only selected from candidate parent nodes (not considering the neighbor nodes), named AggOR-CP.

**Table 2.** Simulation parameters.

Parameter	Value
Scenario (m <sup>2</sup> )	100 × 100
Number of sink node	1
Number of sensor nodes, $N$	40, 80, 120, 160 and 200
Sensing radius (m)	25
Communication radius (m)	52
Data collection cycle (s)	60
$OR_T$	0.5

Note that the data sizes after aggregation are not the same in different schemes. In AggOR, after a gathering node aggregates  $x$  data packets, the amount of output data is  $p(0 < p < x \cdot d)$ , which is decided by the overlapping rates of sensing area in this gathering area. Since the data aggregation function only removes duplicated data relevant with the overlapping sensing area, the whole data obtained by the sink are complete and accurate. However, in EEUC and HEER, a packet with a fixed amount of data is output by aggregating several data packets.

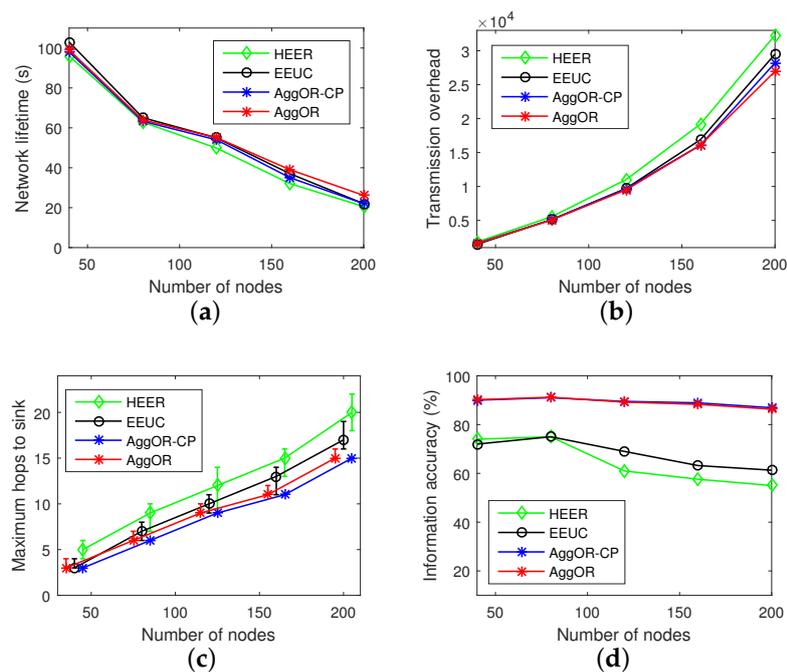
The following metrics are used for the performance evaluation.

- (1) Network lifetime: the time interval from the beginning of the network to the death of the first node.
- (2) Transmission overhead: the total amount of data transmitted in one data transmission round. It indicates the energy consumption of data sending and receiving in the whole network.
- (3) Maximum number of hops to the sink: the maximum number of hops from sensor nodes to the sink in the network. More hops mean a longer time for which the sink has to wait to collect all the data in the scenario. Hence it implies the data delivery delay.
- (4) Information accuracy: the ratio of the amount of information collected by the sink to the amount of information in all raw data.

Considering that sensor density may influence the performance of data aggregation, we will discuss about this issue in Section 5.3.1. In addition, the threshold of overlapping rate of sensing area is a significant factor affecting the size of gathering area and the energy efficiency of AggOR scheme. Therefore, we will analyze the influences of this threshold in Section 5.3.2. Moreover, we consider three scenarios corresponding to different shapes of monitoring field and different locations of the sink in the network. In Scenario SP, the nodes are deployed in a pyramid field, of which the top is the sink. In Scenario SC, the sink is placed at the center of a circular field. Scenario SS has a square field with the sink in the top-left corner. Note that in all the scenarios, sensor nodes are uniformly deployed. We will analyze the sources of gathering nodes in AggOR and AggOR-CP with different  $OR_T$ , in Section 5.3.3.

## 5.2. Experiment Results

We evaluate the performances of four schemes, i.e., HEER, EEUC, AggOR-CP and AggOR, and the results are illustrated in Figure 7. Note that the experiments are conducted in Scenario SP. With the number of sensors increasing, the density of nodes does not change. In other words, the results are obtained under different network scales with the same density. Specifically, the networks with 40, 80, 120, 160 and 200 nodes cover approximately 20%, 40%, 60%, 80% and 100% of the whole 100 × 100 m<sup>2</sup> scenario, respectively.

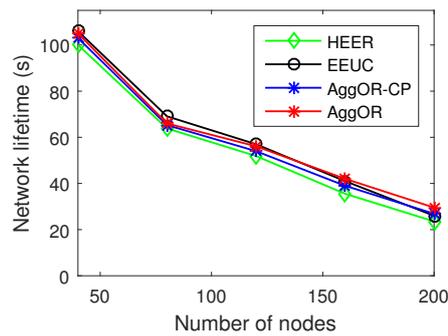


**Figure 7.** Experiment results. (a) network lifetime; (b) transmission overhead; (c) maximum hops to the sink; (d) information accuracy.

As Figure 7a shows, when there are 40 nodes in the network, EEUC has the longest lifetime of the network, and AggOR tightly follows. However, with the number of nodes increasing, the lifetime of AggOR which is longer than AggOR-CP, gradually exceeds EEUC from the scale of 120 nodes, and the lifetime of HEER is the shortest. Even though in EEUC scheme, the clusters output a single length-fixed packet after aggregation, which is smaller than the output of gathering nodes in AggOR, the main reasons of the result lie in two aspects. (1) The redundant data is forwarded for several hops in EEUC while redundancy is only relayed once in AggOR; (2) Compared with EEUC in which the cluster heads may be at lower levels, in AggOR, the gathering nodes are mainly the upper-level nodes, and thus sensor data are always forwarded up avoiding the back and forth relay.

Note that there are different ways to define the dead time of the network. If the sensing range of the first dead node is covered by others, the network might continue to work. Therefore, beside the death of the first node, we analyze the network lifetime, which takes the time when some area cannot be sensed any longer or some data cannot be delivered to the sink as the dead time. The results are shown in Figure 8. The network lifetimes of all the four schemes in Figure 8 are a little longer than or the same as those in Figure 7a, because the sensing areas of some first dead nodes are covered by others and some first dead nodes are the gathering nodes or the forwarders for others. However, the trends of the results in these two figures are similar, both showing that our scheme AggOR achieves a similar network lifetime to EEUC and a little longer lifetime than HEER.

Figure 7b demonstrates that the transmission overhead in AggOR is a little smaller than those in AggOR-CP and EEUC schemes when the number of nodes is 80. As the number of nodes increases, the transmission overheads of HEER, EEUC and AggOR-CP are increasing faster than AggOR. AggOR has a smaller transmission overhead than AggOR-CP, EEUC and HEER by about 4%, 10% and 17% respectively in the scenario with 200 nodes. HEER always has the highest overheads among the compared schemes.



**Figure 8.** Another definition of network lifetime.

Because the maximum hops in multiple tests for HEER is not stable, we use median values rather than average values in Figure 7c. From the figure, AggOR-CP has the smallest and stable number of hops, while AggOR in some rare cases has more hops than others. This is because in our scenarios with  $OR_T = 0.5$ , all the gathering nodes in AggOR-CP are the upper-level nodes, while there are some neighbor nodes as gathering nodes in AggOR, slightly increasing the number of hops in data transmission. As the number of nodes increases, the maximum hops of EEUC and HEER schemes are more unstable and larger than AggOR, especially HEER. When there are 200 nodes, the maximum hops in AggOR is smaller than EEUC and HEER by 12% and 25% respectively. The main reason is that member nodes send data to their gathering nodes by just one hop both in AggOR and AggOR-CP. By contrast, there exists multi-hop routing in clusters of EEUC, and in HEER the members transfer their data to cluster heads following Hamilton Paths, which increase hops to the sink and prolong the end-to-end delay.

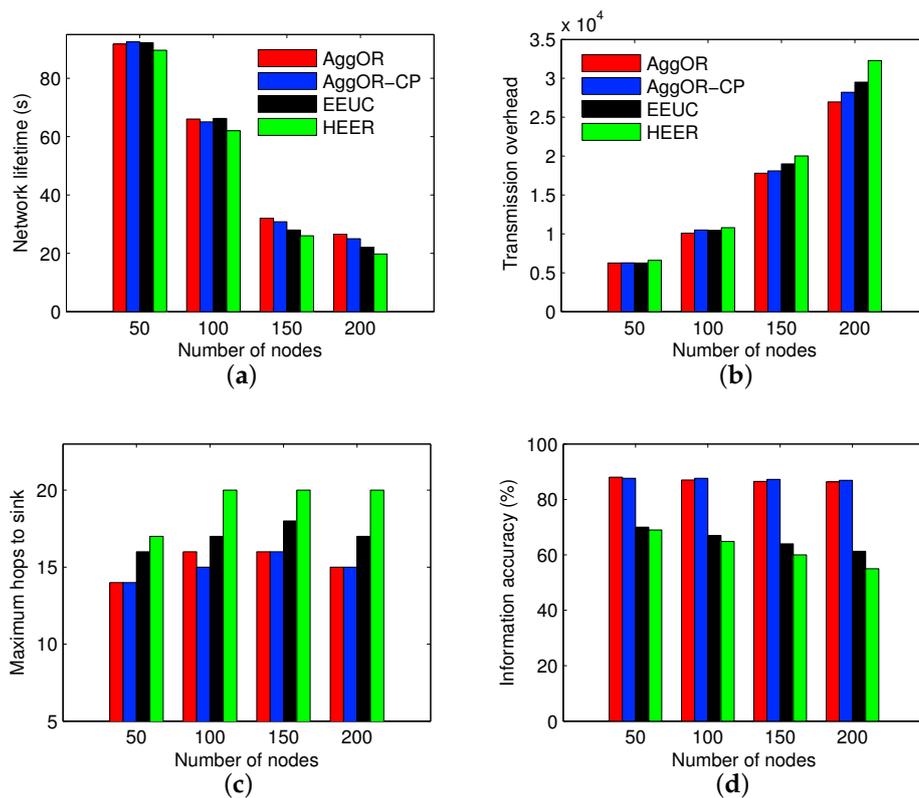
As Figure 7d illustrates, in AggOR the ratio of information accuracy is the highest (around 88%) among those four schemes and very similar to AggOR-CP, which is higher than EEUC and HEER by 38% and 48% respectively. This is because aggregating lots of data into a small fixed amount of data and aggregating the same data for several times both affect the information accuracy. In HEER, the end node of Hamilton Path transmits its data to its neighbor which is closer to the cluster head, and the neighbor aggregates data into one packet of a constant size. This process continues until the data reaches the cluster head. Therefore, an original data may be aggregated for several times. In EEUC, the clusters aggregate all member data into one packet with fixed size, regardless of the specific redundancy ratios. By contrast, AggOR aggregates one data only once, and does not lose any information. Note that the ratio of information accuracy in AggOR is not 100% because of the redundant data.

In conclusion, AggOR scheme achieves an energy-efficient and quick data collection, while ensuring a high data accuracy. Moreover, AggOR has a greater advantage over other schemes when the network scale rises, because the efficiency of redundancy clearing within only one-hop forwarding is apparent in a large scale scenario.

### 5.3. Parameter Analysis

#### 5.3.1. Sensor Density Analysis

We analyze the effects of node density on the performances of all the four schemes in the fixed  $100 \times 100 \text{ m}^2$  zone area. The numbers of nodes are 50, 100, 150 and 200, respectively. Accordingly, the sensor density ranges from 0.005 to 0.02 nodes per square meter. The results are illustrated in Figure 9.



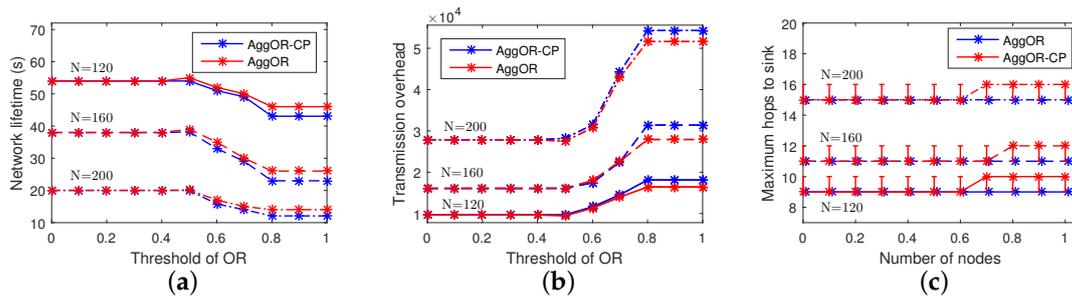
**Figure 9.** Results with different sensor densities. (a) network lifetime; (b) transmission overhead; (c) maximum hops to the sink; (d) information accuracy.

In Figure 9, a high density leads to short network lifetimes and large transmission overheads for all the schemes, while the maximum hops to sink and the information accuracy do not change a lot. Additionally, as the sensor density increases, AggOR scheme, which always presents a high data accuracy and short transmission route in the experiments, begins to show longer lifetime and better energy efficiency than EEUC and HEER. On the whole, AggOR performs the best among the compared schemes with different sensor densities, especially in high-density sensor networks.

### 5.3.2. Analysis of the Threshold of Overlapping Rate of Sensing Area

In order to show the impacts of the threshold of overlapping rate  $OR_T$  on the overall performances of AggOR and AggOR-CP, we conduct a series of experiments in Scenario SP with  $OR_T$  ranging from 0 to 1. Figure 10 shows the simulation results. The solid line, intermittent line and chain dotted line indicate the three scenarios with 120, 160 and 200 nodes, respectively. In Figure 10, the results are stable when  $OR_T \leq 0.4$  or  $OR_T \geq 0.8$ , because in our dense WSN, all the overlapping rates of sensing area between two nodes are larger than 0.4 and smaller than 0.8.

In Figure 10a, when  $OR_T \leq 0.4$ , all gathering nodes are candidate parent nodes and almost no independent node exists. When  $OR_T$  increases to 0.5, the network lifetime prolongs a little, because the gathering areas are constructed efficiently with the larger overlapping sensing area as well as some neighbor nodes selected as gathering nodes. When  $OR_T$  continues to increase, less candidate parent nodes and more neighbor nodes are selected as the gathering nodes, which increases energy cost. Additionally, more independent nodes also limit the benefits of data aggregation. Therefore, with a medium value 0.5 of  $OR_T$ , AggOR reaches the longest network lifetime. Moreover, the network lifetime of AggOR is superior to that of AggOR-CP.



**Figure 10.** Results with different  $OR_T$ . (a) network lifetime; (b) transmission overhead; (c) maximum hops to the sink.

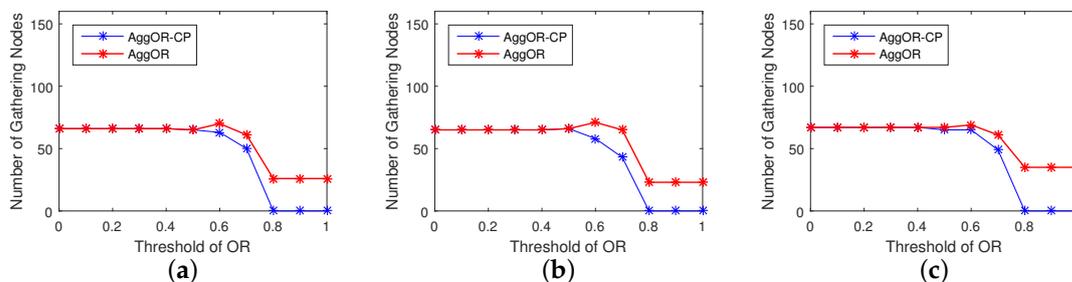
Figure 10b depicts that, when there are 120 or 200 nodes, the smallest transmission overhead appears in AggOR when  $OR_T$  is 0.5. It is obvious that the larger  $OR_T$  is, the higher transmission overhead there is due to less aggregation. In addition, transmission overhead in AggOR is smaller than AggOR-CP, and large scale leads to a bigger difference between them. It is consistent with our previous analysis that the neighbor nodes working as gathering nodes cost less energy for data transmission than independent nodes.

Data collection and aggregation by the gathering nodes cause a little delay apparently, and the neighbor nodes as the gathering nodes further increase the hops to the sink. When  $OR_T$  rises, the median value of maximum hops to the sink in AggOR is bigger than AggOR-CP, as shown in Figure 10c. However, the largest difference between them appears when  $OR_T$  is larger than 0.6 and the difference is one hop, which implies that there are no candidate parent nodes suitable to be gathering nodes for some node on the longest path to the sink.

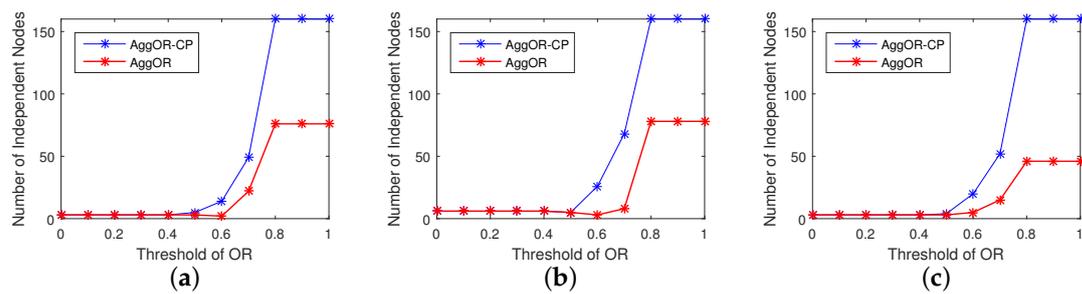
### 5.3.3. Gathering Node Analysis

Considering that the number of gathering nodes indicates the efficiency of data aggregation, we analyze the number of gathering nodes and the number of independent nodes, in three scenarios, i.e., SP, SC and SS, with 160 nodes. The results are illustrated in Figures 11 and 12.

In different scenarios, the numbers of two kinds of nodes are similar with the same  $OR_T$ . Comparing the numbers of gathering nodes in AggOR and AggOR-CP in Figure 11, we get that when  $OR_T$  is smaller than 0.5, no neighbor nodes are selected as gathering nodes. In Figure 12, when  $OR_T$  increases from 0.5, AggOR-CP has more independent nodes than AggOR. When  $OR_T$  is larger than 0.8, the numbers of gathering nodes and the numbers of independent nodes both reach plateaus. Specifically, all nodes in AggOR-CP are independent nodes, while AggOR has some neighbor nodes selected as gathering nodes, and hence reduces the transmission overhead.



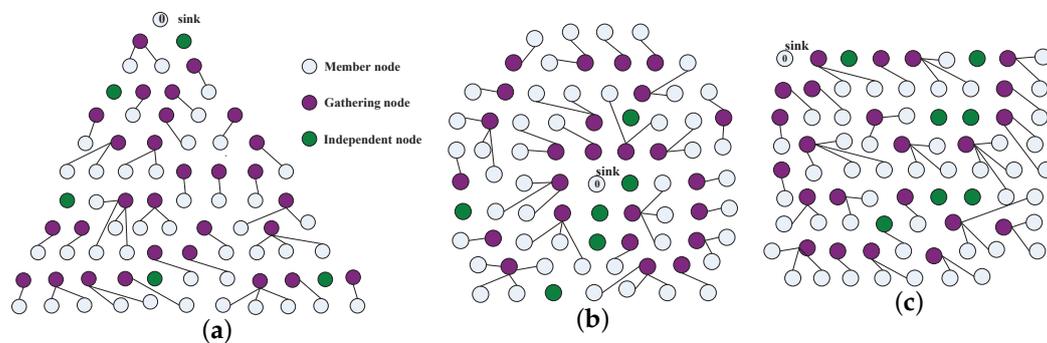
**Figure 11.** The numbers of gathering nodes in three scenarios. (a) Scenario SP; (b) Scenario SC; (c) Scenario SS.



**Figure 12.** The numbers of independent nodes in three scenarios. (a) Scenario SP; (b) Scenario SC; (c) Scenario SS.

Figure 13 shows the sketch maps of three scenarios, i.e., SP, SC and SS, with 80 nodes in AggOR. The edges indicate the relationships between gathering nodes and member nodes. Due to the limited spaces, to avoid confusion caused by crossing lines, we do not show the data transmission paths from the gathering nodes to the sink. From the figure, we get that the gathering nodes are usually located near their member nodes, and most of them are closer to the sink than member nodes.

Overall, the larger the sensor density is, the better AggOR performs. In a relatively dense network, a larger network scale leads to a greater advantage of AggOR over EEUC and HEER. Additionally,  $OR_T$  has a significant effect on the performance of AggOR, which peaks at  $OR_T = 0.5$  in our experiments. In real scenarios, an appropriate  $OR_T$  can be obtained through sampling analysis during preliminary study. Furthermore, AggOR has a longer network lifetime and a smaller traffic overhead than AggOR-CP in the three scenarios with different sensor deployments and different locations of the sink.



**Figure 13.** Sketch maps of three scenarios. (a) Scenario SP; (b) Scenario SC; (c) Scenario SS.

## 6. Conclusions

Wireless sensor networks are deployed to support a variety of precise monitoring applications in smart factories, and require energy-efficient and no-entropy-loss data aggregation. In this paper, we propose a data aggregation scheme based on the overlapping rate of sensing area, named AggOR. In the transmission hierarchy diagram, some candidate parent nodes as well as appropriate neighbor nodes, whose overlapping rates of sensing area are not smaller than a preset threshold  $OR_T$ , may be selected as the candidate gathering nodes. It guarantees that the sensor data in a gathering area are extremely correlative, and there exist a large amount of redundant data to be cleaned. Member nodes transfer their data to gathering nodes through one hop, and only duplicated data are removed by aggregation, ensuring a short end-to-end delay and a high data accuracy. A large quantity of experiments on OPNET modeler show that AggOR has a better data accuracy and a shorter delay than compared schemes, while keeping similar network lifetime.

However, the specific relation between  $OR_T$  and network density still requires further study. In addition, considering the occurrence of multiple events at the same time [27], how to optimize multi-event data collection by analyzing overlapping sensing area is another research topic for the future.

**Acknowledgments:** We gratefully acknowledge the support from the National Natural Science Foundation of China (61502320, 61373161 & 61572060), Science & Technology Project of Beijing Municipal Commission of Education in China (KM201410028015), Youth Backbone Project of Beijing Outstanding Talent Training Project (2014000020124G133), 973 Program (2013CB035503), CERNET Innovation Project (NGII20151004, NGII20160316), Beijing Advanced Innovation Center for Imaging Technology, Cultivation Object of Young Yanjing Scholar of Capital Normal University and Science Foundation of Shenzhen City in China (JCYJ20160419152942010).

**Author Contributions:** X.T. conceived and designed the study; H.X. performed the experiments and wrote the paper; W.C. and J.N. supervised the research; S.W. revised the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Yang, J.; Zhou, J.; Lv, Z.; Wei, W.; Song, H. A real-time monitoring system of industry carbon monoxide based on wireless sensor networks. *Sensors* **2015**, *15*, 29535–29546.
2. Tang, X.; Pu, J.; Gao, Y.; Xiong, Z.; Weng, Y. Energy-efficient multicast routing scheme for wireless sensor networks. *Trans. Emerg. Telecommun. Technol.* **2014**, *25*, 965–980.
3. Jeschke, S.; Brecher, C.; Song, H.; Rawat, D. *Industrial Internet of Things: Cyber Manufacturing Systems*; Springer: Berlin, Germany, 2017.
4. Shu, L.; Mukherjee, M.; Hu, L.; Bergmann, N.; Zhu, C. Geographic Routing in Duty-cycled Industrial Wireless Sensor Networks with Radio Irregularity. *IEEE Access* **2016**, *4*, 9043–9052.
5. Shu, L.; Wang, L.; Niu, J.; Zhu, C.; Mukherjee, M. Releasing network isolation problem in group-based industrial wireless sensor networks. *IEEE Syst. J.* **2015**, *1*, 1–11.
6. Kumar, A.; Baksh, R.; Thakur, R.K.; Singh, A.P. Data aggregation in wireless sensor networks. *Int. J. Sci. Res.* **2014**, *3*, 249–251.
7. Singh, S.P.; Sharma, S.C. A survey on cluster based routing protocols in wireless sensor networks. *Comput. Sci.* **2015**, *45*, 687–695.
8. Zeb, A.; Islam, A.K.M.M.; Zareei, M.; Mamoon, I.A.; Mansoor, N.; Baharun, S.; Katayama, Y.; Komaki, S. Cluster analysis in wireless sensor networks: The ambit of performance metrics and schemes taxonomy. *Int. J. Distrib. Sens. Netw.* **2016**, *12*, 4979142.
9. Fasolo, E.; Rossi, M.; Widmer, J.; Zorzi, M. In-network aggregation techniques for wireless sensor networks: A survey. *IEEE Wirel. Commun.* **2007**, *14*, 70–87.
10. Madden, S.; Franklin, M.J.; Hellerstein, J.M.; Hong, W. TAG: A tiny aggregation service for ad-hoc sensor networks. *ACM SIGOPS Oper. Syst. Rev.* **2002**, *36*, 131–146.
11. Deligiannakis, A.; Kotidis, Y.; Stoumpos, V.; Delis, A. Building efficient aggregation trees for sensor network event-monitoring queries. *Int. Conf. GeoSens. Netw.* **2009**, 5659, 63–76.
12. Zhang, Y.; Pu, J.; Liu, X.; Chen, Z. An adaptive spanning tree-based data collection scheme in wireless sensor networks. *Int. J. Distrib. Sens. Netw.* **2015**, *2015*, 637387.
13. Manjhi, A.; Nath, S.; Gibbons, P.B. Tributaries and deltas: Efficient and robust aggregation in sensor network stream. In Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, Baltimore, MD, USA, 14–16 June 2005; pp. 287–298.
14. Shukla, K.V. Research on energy efficient routing protocol LEACH for wireless sensor networks. *Int. J. Eng.* **2013**, *2*, 1–5.
15. Younis, O.; Fahmy, S. HEED: A hybrid, energy-efficient, distributed clustering approach for Ad hoc sensor networks. *IEEE Trans. Mob. Comput.* **2004**, *3*, 660–669.
16. Li, C.; Ye, M.; Chen, G.; Wu, J. An energy-efficient unequal clustering mechanism for wireless sensor networks. In Proceedings of the 2005 IEEE International Conference on Mobile Adhoc and Sensor Systems Conference (MASS), Washington, DC, USA, 7 November 2005; pp. 597–604.
17. Sabet, M.; Naji, H.R. A decentralized energy efficient hierarchical cluster-based routing algorithm for wireless sensor networks. *AEU Int. J. Electron. Commun.* **2015**, *69*, 790–799.

18. Leu, J.S.; Chiang, T.H.; Yu, M.C.; Su, K.W. Energy efficient clustering scheme for prolonging the lifetime of wireless sensor network with isolated nodes. *IEEE Commun. Lett.* **2015**, *19*, 259–262.
19. Yi, D.; Yang, H. HEER-A delay-aware and energy efficient routing protocol for wireless sensor networks. *Comput. Netw.* **2016**, *104*, 155–173.
20. Villas, L.A.; Boukerche, A.; de Oliveira, H.A.B.F.; de Araujo, R.B.; Loureiro, A.A.F. A special correlation aware algorithm to perform efficient data collection in wireless sensor networks. *Ad Hoc Netw.* **2011**, *12*, 10–30.
21. Villas, L.A.; Boukerche, A.; Ramos, H.S. DRINA: A lightweight and reliable routing approach for in-network aggregation in wireless sensor networks. *IEEE Trans. Comput.* **2013**, *64*, 676–689.
22. Gherbi, C.; Aliouat, Z.; Benmohammed, M. An adaptive cluster approach to dynamic load balancing and energy efficiency in wireless sensor networks. *Energy* **2016**, *114*, 647–662.
23. Pu, J.; Gu, Y.; Zhang, Y.; Chen, J.; Xiong, Z. A Hole-Tolerant Redundancy Scheme for Wireless Sensor Networks. *Int. J. Distrib. Sens. Netw.* **2012**, *2012*, 184–195.
24. Imon, S.K.A.; Khan, A.; Francesco, M.D.; Das, S.K. Energy-efficient randomized switching for maximizing lifetime in tree-based wireless sensor networks. *IEEE ACM Trans. Netw.* **2015**, *23*, 1401–1415.
25. Xiang, M.; Wang, P.; Luo, Z. Node classification method based on integrative support degree in WSN. *Comput. Eng.* **2010**, *36*, 97–99.
26. George, T.; Trevor, C. Simulation tools for multilayer fault restoration. *IEEE Commun. Mag.* **2009**, *47*, 128–134.
27. Dong, M.; Ota, K.; Liu, A. RMER: Reliable and Energy-Efficient Data Collection for Large-Scale Wireless Sensor Networks. *IEEE Internet Things J.* **2016**, *3*, 511–519.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).