IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 7, 2021, accepted January 21, 2021, date of publication January 26, 2021, date of current version February 2, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3054630

An Efficient Data Aggregation Scheme Based on Differentiated Threshold Configuring Joint Optimal Relay Selection in WSNs

ANG LI[1], WEI LIU[2,3], LIJUN ZENG[1], CHUNJI FA[3], AND YAN TAN[1]
[1]School of Computer Science and Technology, Hunan Institute of Technology, Hengyang 421002, China
[2]School of Informatics, Hunan University of Chinese Medicine, Changsha 410208, China
[3]School of Computer Science and Engineering, Central South University, Changsha 410083, China

Corresponding author: Yan Tan (tanyan@hnit.edu.cn)

This work was supported in part by the Natural Science Foundation of Hunan Province, China under Grant 2019JJ40064, in part by the Scientific Research Project of Education Department of Hunan Province under Grant 19B142, in part by the Second Batch of Cooperative Education Project of Industry University Cooperation, in 2018, by the Department of Higher Education of the Ministry of Education, under Grant 201802076013, in part by the Innovation and Entrepreneurship Training Plan for College Students in Hunan Province in 2019 with project name: campus service of Knight escort agency, and in part by the Research Project of Teaching Reform in Colleges and Universities in Hunan Province in 2019 under Grant name Research on the Construction of Maker Education System in Applied Local Colleges and Universities.

ABSTRACT In Wireless Sensor Networks (WSNs), when multiple data packets meet during routing to the sink, redundant data can be removed through data fusion, thereby reducing the amount of data transmitted, and increasing the life of the network. However, how to increase the data fusion rate as much as possible and ensure that the delay is lower than the deadline is a challenge issue. To solve this problem, a Differentiated Threshold Configuring joint Optimal Relay Selection based Data Aggregation (DTC-ORS-DA) scheme is proposed, which can significantly reduce redundant data routing and guarantee the delay for WSN. The main innovation is as follow: (1) In DTC-ORS-DA scheme, there are two thresholds: data volume threshold N_y and time threshold T_y. Data routing can only be performed when N_y or T_y of the node meet the threshold requirements, so that the data can be fully integrated to minimize the amount of data to be transmitted. More importantly, DTC-ORS-DA adopts differentiated threshold settings based on the characteristics of unbalanced energy consumption in WSNs, and sets a smaller threshold in the far sink area with sufficient energy, so that data packets can be routed quickly. And the near sink area where the energy is tight uses a larger threshold to maximize data fusion, so that the combination can make the data fusion high, the energy is effectively used, and the delay is small. (2) We propose a priority-based relay selection algorithm, which enables child nodes to dynamically select the parent node with the highest priority based on the number of data packets, waiting time, and remaining energy. In the process of routing, the probability of nodes with many data packets or long waiting time being selected as transmission relay is high, which can either increase the data fusion rate or reduce the delay. Finally, the performance comparison with Common data collection Scheme (CS) proves that, the DTC-ORS-DA scheme reduces the average delay by 10.74%-19.91%, increases the life cycle by 9.81% at most, and the energy utilization rate is increased by 6.67%-9.48%.

INDEX TERMS Wireless sensor networks, data aggregation, energy efficient, delay, lifetime.

I. INTRODUCTION

With the rapid development of micro-processing technology, more and more sensing devices are connected to the Internet of Thing (IoT) [1]–[3]. By 2020, the number of devices connected to the IoT has exceeded 20 billion [4], [5]. The data generated by these IoT devices reaches 2.5 quintillion bytes every day [4], [5]. On the other hand, the Artificial Intelligence (AI) technology has enriched the ability of humans to use data [6]–[8], thus making humans' desire for sensing and collection data more urgent [9]–[11]. Therefore, many sensing devices are increasingly being deployed in various applications to sense and obtain data [12]–[14], making edge computing, fog computing and other emerging

The associate editor coordinating the review of this manuscript and approving it for publication was Qin Liu.

computing models fully developed [15]–[17]. Compared with cloud computing [18], IoT devices with weak computing power can offload tasks to edge servers [19]–[21], making more applications more dependent on IoT devices for sensing and obtaining data [22]–[24]. Among them, Wireless Sensor Networks (WSNs) are the earliest researched networks [25]–[27]. In such a network, many sensing nodes are widely deployed in the area that needs to be monitored through various methods. The communication distance of sensor nodes is limited, and the node deployment density is high. These nodes are constructed into a network through self-organization, to perceive the area that needs to be monitored, obtain data, and send it to a special node called sink through multi-hop routing [22], [28]. In recent years, the rise of edge computing has promoted the development of IoT. These sensor networks can also offload their own computing tasks to edge servers and even the cloud, forming the so-called Sensor-Cloud computing system [29], [30].

Limited energy is the shortcoming of sensor nodes [22], [31], [32]. Because sensor nodes are powered by batteries, due to economic cost and volume constraints, the batteries are not very large, and their power is limited [22]. In general, sensor nodes are difficult to replace and recharge once deployed [13], [16]. Therefore, how to be energy efficient in various applications of WSNs is an important research issue [22], [24], [26]. The largest energy consumption of WSNs is the consumption of data transmission by nodes, so the most effective means to reduce energy consumption is to reduce the amount of data that needs to be transmitted [33]–[35]. In fact, there has been quite a lot of researches on how to reduce the amount of data that needs to transmit. The research in this article is based on the predecessors and proposes a more effective data fusion strategy to reduce the amount of transmitted data.

An effective way to reduce the amount of data is data aggregation [34]–[36]. The basic idea of data aggregation is as follows. Due to the high density of nodes deployed in WSNs, there is redundancy between the data of various physical phenomena that it perceives. So, when multiple data packets meet in the routing to the sink, data fusion of multiple data packets can be carried out, and the data volume after data fusion is surely less than the sum of the data volume of the original multiple data packets, which can reduce the amount of data that needs to transmit [33], [35]. Since the data fusion method can reduce the transmission load without reducing the amount of information [36], this method is widely adopted in WSNs since it has been proposed [35]. In the initial research, the main consideration is how to maximize the number of data fusions. The main representatives of this type of research are cluster based WSNs. In such a network, the network is divided into multiple clusters [37], [38]. Each cluster has a special node called Cluster Head (CH), and the other nodes within the cluster are called member nodes. Member nodes send their data to CH, so that after CH receives all the member nodes and performs data fusion, then the data is sent to sink through multi-hop routing. Since only multiple data packets meet to remove their redundant data through data aggregation, the key to data aggregation is to allow as many data packets as possible to meet during the routing process, so that the amount of data can be reduced the most [38]. Jiang *et al.* [34] proposed a dynamic ring-based routing scheme. In their method, the data of the nodes on the network are first routed to the nodes on the ring, and then the data of these nodes are routed around the ring for full data fusion is then routed to sink, which can maximize data fusion [34]. However, this type of research only considers how to reduce the amount of data borne by the node, and the delay of the data reaching the sink is not considered, so it is not suitable for applications that require a deadline for the time the data packet arrives at the sink. Another type of research that considers data fusion and reduces delay [37], when multiple data packets meet, they can be fused into one data packet, so that after optimization, each node only needs to send one data packet in a round of data collection. This type of data fusion generally uses TDMA, which allocates active slots to each node in advance, and only in the awake state can the node has data operations, and when there is no data operation, it sleeps to save energy [39]. This data fusion solution generally uses data routing from the bottom of the network to the sink layer by layer. Each node collects data from its child nodes and merges them into a data packet and then routes forward. The effective allocation of data operation slots to nodes can effectively reduce the time required for data collection, that is, reduce the convergecast time [39].

However, convergecast requires data to be generated periodically, however, in real-life, there are not many WSNs that meet this condition [39]. Most WSNs are networks that generate non-periodic data. In such a network, node data is generated by random events or environmental changes [40]. Moreover, these data packets are required to arrive at the sink as quickly as possible, and even have a deadline when the data packets arrive at the sink. The data packets arriving at the sink later than the deadline are invalid [40]. Therefore, in this case, the data packet is required to reach the sink quickly, that is, the data packet is required to be routed as fast as possible. Data fusion requires as many data packets as possible to meet in the routing, so as to take advantage of data fusion to remove redundant data and reduce the amount of transmitted data. Therefore, during the routing process, the previous data packets stay on the node and wait for the subsequent data to be fused. Obviously, this method will increase the delay of data transmission [40]. Therefore, Li *et al.* [40] proposed a compromise method. The method they proposed is to set two thresholds: queue length threshold and waiting time threshold. When the node receives the data packet, it does not forward it immediately, but to see whether the node's threshold is met. If the length of the data packets queued by the node to be sent is greater than the threshold, it means that the number of data packets currently encountered has reached a certain value, indicating that the node with sufficient data fusion can send data. And if the queuing queue of the node is less than the threshold, but the node's data packet has been waiting for longer than the threshold. If it continues to wait at this time, the data packet will exceed the deadline, and at this time, the node also sends data packets [40]. Taking such a method can ensure that the delay of the data packet to the sink is not greater than the deadline, and on the other hand, wait for as many data packets as possible for data fusion, thus achieving the compromise optimization of the data fusion and delay [40].

Although the method of setting the threshold for each node can achieve better results. However, this method still has the following aspects that deserve improvement. First, in this method, the selection of the relay node is not considered when the node sends data, which leads to the random selection of the relay node. Obviously, if the node chooses the longest queued node as the relay, it will improve the network performance. Because, if all the child nodes choose the node with the longest queue as the relay, those nodes that already have data packets will get more data packets on the one hand, so the more data integration is, on the other hand, such nodes can easily meet the conditions for data transmission, which can speed up data routing at the same time. Finally, by adopting this method, data packets will be concentrated on fewer nodes, which will not only enable full data fusion, but also obtain the advantage that the previous strategy did not have. The number of nodes that have data operations on the network will be reduced, which will reduce the interference and conflict during data operations.

The disadvantage of the previous strategy is that all nodes use the same threshold. The purpose of setting the threshold is to let the data packet stay on the node for a period of time, rather than forwarding it as it arrives. Obviously, if the threshold set by the node is larger, the longer the data packet stays on the node, the more conducive to data fusion, which can effectively reduce the amount of data. However, because the data packet stops on each node for a long time, the deadline of the data packet may not be guaranteed. Therefore, although setting an optimized threshold can improve the performance of WSNs, this is far from enough. Because there is a special situation in WSN that the energy consumption of the network is unbalanced. Because in WSNs, sink is the center of the entire data collection, all node data needs to be routed to sink. Such a ''many-to-one'' data collection mode causes the nodes near the sink to bear much more data than other areas, forming a so-called ''energy hole'' and premature death. When the network dies prematurely due to ''energy hole'', its remaining energy is as high as 80% [34]. Therefore, how to make full use of the remaining energy in the network to further improve network performance is a challenge issue.

To solve this problem, a Differentiated Threshold Configuring joint Optimal Relay Selection based Data Aggregation (DTC-ORS-DA) scheme is proposed in this paper to reduce redundant data routing and guarantee the delay for WSN. The main innovation of our work is as follow:

1) In DTC-ORS-DA scheme, a data aggregation scheme with differentiated threshold configuration is proposed. Based on the characteristics of uneven energy consumption in WSNs, a small threshold is set in the far sink area with sufficient energy, so that nodes can quickly meet the transmission and fusion conditions to reduce the delay. In the near sink area where energy is tight, a larger threshold is adopted, so that the data packets stay on the node for a long time, so as to maximize data fusion and effectively reduce the amount of transmitted data. Since the range of the near sink area is small, and the routing in most far areas is faster, the overall delay is smaller than the strategy with a fixed threshold on the whole network. Therefore, the adaptive threshold strategy proposed in this paper can not only fully integrate data to reduce the amount of transmitted data, but also make the delay smaller.

2) A priority-based optimal relay selection algorithm is proposed by considering factors such as the number of existing data packets, waiting time, remaining energy, which can effectively improve the fusion effect and reduce conflicts. In the proposed algorithm, those nodes that already have a large number of data packets, or the node with the longest waiting time are more likely to be selected as the relay node. This has the following advantages compared with the previous strategy: first, if the parent node with a large number of data packets is selected as the relay node, then sufficient data fusion can be obtained to reduce the amount of data and increase the life of the network; at the same time, choosing such a node as a relay node is the easiest to reach the threshold, which can reduce the time that the data packets stay and reduce the delay. What's more, in such an algorithm, those nodes with more remaining energy are selected as relay nodes, and as long as the node is selected as a relay node, its priority is higher than other nodes with empty packets, thereby reducing interference and conflict of nodes, improve network performance.

3) Extensive simulation results by comparing DTC-ORS-DA scheme with CS scheme illustrate that, the DTC-ORS-DA scheme proposed in this paper outperforms better. Its average delay is reduced by 10.74%-19.91%, the life cycle is increased by 9.81%, and the energy utilization is increased by 6.67%-9.48%.

The rest of this paper is organized as follows: In Section II, the related work is introduced. The system model and problem statement are presented in Section III. Then, the DTC-ORS-DA scheme is introduced in Section IV. Performance analysis is presented in Section V. Finally, Section VI provides conclusion.

II. RELATED WORK

Data collection is the most important function in WSNs. A large amount of data forms the basis of various applications [41], [42]. With the development of artificial intelligence technology, more potential value can be obtained from a large amount of data, which promotes the widespread application of IoT devices [31], [32]. However, data collection in WSNs is not an easy task, and many factors affect the performance of data collection [13], [16]. In addition to the common security attacks in data collection, the energy consumption and delay in the data collection process are the most important issues in data collection [40]. Researchers have done a lot of work in this area. Next, we discuss them separately.

1) Data collection that mainly considers data fusion. In this type of research, the main research goal is how to fully carry out data fusion, thereby effectively reducing the amount of data undertaken by nodes and improving lifetime. The idea of data fusion is: because there is redundant information between the data perceived between WSNs nodes, when multiple redundant data packets meet, the redundant data in the multiple data packets can be removed. Therefore, the data packet capacity of data fusion is smaller than the data packet capacity before fusion [33]–[40]. Several representative works are given below.

In the initial research, various data fusion algorithms are proposed [33], [35], [36], which are applied the to the existing data routing strategy. The simplest data fusion strategy is to send the data fusion algorithm to each node without modifying the data routing. For example, the data fusion is used along the path with shortest routing algorithm. In such a data fusion strategy, when multiple data packets meet, the data fusion algorithm is used to calculate the data, and the redundant data packets are deleted before routing forward. Obviously, in such a strategy, since the data routing is not specifically designed, the probability of data packets meeting is not particularly high. Therefore, the effect of data fusion is not particularly good. Subsequent researchers deliberately proposed a special data routing strategy for data fusion to increase the probability of multiple data packets meeting, to fully integrate data and reduce the amount of data that need to send.

Leandro Aparecido Villas *et al.* [43] proposed a DRINA algorithm to increase the probability of data packets meeting during the routing process, by making as many data packets as possible along the same data routing path to the sink, instead of data packets independently route to sink along different data routing paths. The DRINA algorithm is an improved data fusion strategy of the shortest routing algorithm [43]. In the original shortest routing algorithm, each node selects the node with the smallest number of hops from the sink within its communication range as the relay node. The DRINA algorithm uses the same strategy of selecting relay. However, in the DRINA algorithm, once a node forms a route to the sink, the number of hops from the nodes on this route to the sink becomes 0, so that the nodes near this route path will route to this path instead of opening up a routing path separately [43]. This will increase the chance of data packets meeting, thereby increasing the data fusion rate and reducing the amount of data [43].

Jiang *et al.* [34] proposed a scheme called Ring Based Correlation Data Routing (RBCDR), which allows data packets of the entire network to meet for data fusion, so that the advantages of data fusion can be maximized. The method adopted by RBCDR is to establish a route called a ring [34]. Except for nodes near the sink that directly send data to the sink, other nodes first route to the selected ring. When all the data in the network is are sent to the ring, the data packet is routed along the ring for a cycle. In this way, all the data in the network will meet, so that the data in the network is fully integrated, and the amount of data that the node needs to bear is greatly reduced [34]. Moreover, the ring established in the RBCDR scheme is a region with sufficient energy at a distance from the sink, so it can make full use of the energy of the remote sink region. After a large number of experiments and theoretical analysis, the authors found that the RBCDR scheme can more than double the life of the network compared with previous strategies [34].

There is a method to reduce the amount of data that is different from the above-mentioned data fusion. This method uses a representative mechanism [44]. Since sensor nodes are sensing physical phenomena such as temperature and humidity, the physical quantities perceived by neighboring nodes sometimes differ very little. Therefore, in this case, it is not necessary to transmit the data of each node to the sink. Instead, some nodes in the neighboring area are selected to represent the physical phenomena observed in the entire area. Liu *et al.* [44] proposed a multi-representative re-fusion (MRRF) approximate data collection approach to further reduce the amount of data. In MRRF approach, multiple nodes with similar readings form a set and a representative node represents the perceived value of the set. On this basis, as much of these data as possible are gathered in the routing process for secondary data fusion, which can further reduce the amount of data the node bears [44].

Dong *et al.* proposed a reliability and multipath encounter routing (RMER) strategy in event monitoring networks [45]. Their main work is as follows. First, because the energy of the nodes in the far sink area is surplus, and for event monitoring, if more nodes are selected to monitor the event, the higher the accuracy of the recovery event, and the larger the cluster radius in the far sink area is set to improve the monitoring accuracy. The cluster radius near the sink area is small to save energy. Each cluster is represented by the cluster head to monitor the events of the entire cluster. Secondly, multiple cluster heads send monitoring data to the sink through multiple routes [45]. Different from other data routing, multi-path routing merges into a routing path to send data to the sink before reaching the hotspots area of the network, so that the data generated by the same event will reach the sink via the same route, thereby increasing the probability of data fusion most [45].

2) Data collection research that considers data fusion and considers the time required for data collection. For example, sometimes the network needs to query the maximum, minimum, average temperature, humidity, and other data in the network [37], [39]. In this application, an infinite number of data packets can be merged into one data packet. In such applications, it is expected that the required data can be obtained in the shortest possible time after the query of the application is sent, that is, the time required for data collection is the shortest [37], [39]. Therefore, in this kind of data collection, on the one hand, it is necessary to maximize the

data integration as much as possible, and at the same time make the sink collect the data of the entire network in the shortest time. There is a special term in such a data collection called convergecast, and the time required for the completion of the entire network data collection is called convergecast time [37], [39]. Convergecast is aimed at a network where each node generates only one data packet during a data collection cycle [37], [39]. The general convergecast uses a bottom-up data collection method to generate a tree from the network. The data collection starts from the leaf nodes, and then collects data from the sink layer by layer, and finally the sink receives the data of all nodes in the network [39]. Another type of convergecast data collection strategy is to use a clustering method. In this method, the network is divided into multiple clusters, and the data collection is divided into two stages. One is data collection within the cluster, and the nodes in the cluster sent data to the cluster head (CH), and the CH merges the data into a data packet [37]; the second is data collection between clusters, at this time, only the CH owns the data packet in the network, like the previously described convergecast method, data collection starts from the edge of the network to the sink layer by layer [37]. Li *et al.* [37] proposed a network architecture with unequal radius clustering for clusters near sinks with a large radius and far sink clusters for data collection [37]. In such a clustering network, the far sink cluster is smaller, and the time spent on data collection within the cluster is less, so the inter-cluster data aggregation scheduling process can be started earlier [37]. The cluster near the sink is larger, and it takes more time to collect data in the cluster, so that when the data from the outer layer is transferred to the inner layer, the data collection in the cluster is completed. The method proposed by Li *et al.* [37] makes the state transition of the node only required once, thereby effectively reducing energy consumption [37].

3) Consider the sufficiency of data fusion and consider the deadline of data packets. This type of research is more complicated than the two types of researches discussed above. Because, in the first type of research, there is no need to consider the delay of data routing at all, but only need to maximize the data fusion rate. In the second type of research, it considers a special type of data fusion strategy. The research in this paper is more complicated than the previous two types of researches. In such a network, the number of data packets generated by the node is uncertain, and the data packet is recorded as 0 from the moment of generation, and then the time required to reach the sink is less than the deadline. Therefore, the problems faced by this type of research are twofold, that is, to perform data fusion as fully as possible to reduce the amount of data, and to route to the sink as quickly as possible. And these two goals are often conflicting. To make data fusion better, it is necessary to let the data packet stay on the node for a long time to wait for other data packets to arrive for data fusion, but the data packet stays on the node for a long time, it will increase its delay. In response to this situation, Huang *et al.* [46] proposed a method to change the duty cycle for duty cycle base WSNs. The method they proposed is an improvement of the DRINA algorithm, that is, they use the DRINA algorithm to form a route. But the differences are as follows: One is that the first routing path to the sink sets the hop to the sink to 0 to form a backbone route, and nearby nodes will route to this path, thereby increasing the probability of data encounters in the routing; the second is to set the duty cycle of the nodes on the backbone path to be relatively large, so that the data packets can be routed on such a path faster, which can achieve better results in both energy consumption and delay [46]. Li *et al.* [40] also proposed the solution of two thresholds like this paper. However, the threshold value in their research is fixed, and the selection of relay nodes is not optimized, so the performance is not as good as this paper. Therefore, in this article, we have optimized these two aspects and proposed an Differentiated Threshold Configuring joint Optimal Relay Selection based Data Aggregation (DTC-ORS-DA) scheme to further optimize the network performance.

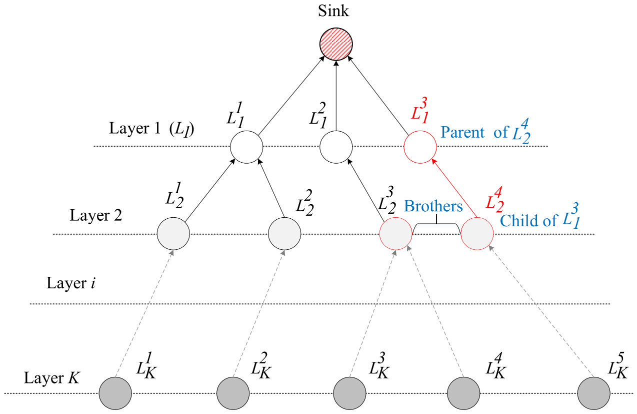


FIGURE 1. Network model.

III. SYSTEM MODEL AND PROBLEM STATEMENT

A. SYSTEM MODEL

The network model adopted in this paper is a planar wireless sensor network, which consists of a sink node and N sensor nodes. Here, we abstract the network as a tree typology, as shown in Fig. 1. Among them, the sink is a special node, and is the root of this tree. It receives data sent by other leaf nodes. Assuming that the nodes in the wireless sensor network can be divided into K layers according to the number of hops from the sink, each layer has m nodes, and the i-th node of the k-th layer is represented as $\mathcal{L}_k^i$. In the small network shown in Figure 1, there are a total of K layers, and three nodes in the 1st layer, four nodes in the 2nd layer, five nodes in the K-th layer. And $\mathcal{L}_1^3$ is the parent node of $\mathcal{L}_2^4$, $\mathcal{L}_2^4$ is the child node of $\mathcal{L}_1^3$. $\mathcal{L}_2^4$ and $\mathcal{L}_2^3$ are brothers, because they are in the same layer, and also in the communication of each other. In the tree network of this article, each node has one or more child nodes and parent nodes, and data is transmitted layer by layer from the child node to the parent node in a multi-hop route, and finally reaches the sink.

The entire network shown in Figure 1 can be represented by the following matrix:

$$\mathbb{T} = \begin{bmatrix} \mathcal{L}_1^1 & \mathcal{L}_1^2 & \cdots & \mathcal{L}_k^{m1} \\ \mathcal{L}_2^1 & \mathcal{L}_2^2 & \cdots & \mathcal{L}_k^{m2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{L}_K^1 & \mathcal{L}_K^2 & \cdots & \mathcal{L}_K^M \end{bmatrix} \tag{1}$$

Each sensor node adopts a cyclic periodic sleep/wake working mode, that is, the entire data collection process is evenly divided into multiple cycles, $\mathbb{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k, \ldots\}$. Each cycle can be subdivided into data packet generation cycle c_g and aggregation and transmission cycle c_t, $\mathcal{C}_k = \left\{c_g^k, c_t^k\right\}$. In the packet generation cycle c_g, each node has a probability of generating a packet. In the aggregation and transmission cycle c_t, the node decides whether to transmit the aggregated packet to the parent node. The aggregation and transmission of data packets occurs in two adjacent layers. Suppose the sending node is $\mathcal{L}_k^s$, the receiving node is $\mathcal{L}_{k-1}^r$, when two nodes wake up in the same cycle, they can send and receive data without waiting. At this time, we call these two nodes periodic synchronization.

In the packet generation cycle c_g, suppose the probability of the node generating a packet is p_d, the probability of not generating a packet is $1 - p_d$. For the node $\mathcal{L}_k^i$, if $\mathcal{L}_k^i = 1$ means that the node generates a data packet, if $\mathcal{L}_k^i = 0$ means the node does not generate a data packet, so the probability model for generating data in this article can be expressed as:

$$P\left\{\mathcal{L}_k^i = x\right\} = (p_d)^x \left(1 - p_d\right)^{1-x}, \quad x = 0, 1 \tag{2}$$

In aggregation and transmission cycle c_t, nodes aggregate data and then transmit it to the next hop, because data aggregation deletes invalid and redundant information from the original data. Therefore, the data volume d' is less than the original data amount d after aggregation. If the aggregation ratio is λ, then the data amount d' can be calculated as:

$$d'_{\mathcal{L}_k^i} = \lambda d_{\mathcal{L}_k^i}. \tag{3}$$

B. CONFLICT AVOIDANCE MODEL

To avoid interference and collisions during data transmission, we stipulate that nodes cannot send and receive data in the same time slot. If two nodes want to send data in the same time slot, they must satisfy the following formula:

$$\begin{cases} \mathcal{D}_{\mathcal{L}_m^i, \mathcal{L}_n^j} > r_{\mathcal{L}_m^i} + r_{\mathcal{L}_n^j} & \text{if } m \neq n \\ \mathcal{L}_m^i \text{ and } \mathcal{L}_n^j \text{ are not brothers} & \text{if } m = n \end{cases} \tag{4}$$

$\mathcal{D}_{\mathcal{L}_m^i, \mathcal{L}_n^j}$ represents the distance between $\mathcal{L}_m^i$ and $\mathcal{L}_n^j$, $r_{\mathcal{L}_m^i}$ and $r_{\mathcal{L}_n^j}$ are transmission radius of $\mathcal{L}_m^i$ and $\mathcal{L}_n^j$, m and n are layers where $\mathcal{L}_m^i$ and $\mathcal{L}_n^j$ located. In general, when parent and child nodes or sibling nodes within the same transmission range send data at the same time, conflicts are likely to occur. Therefore, the conflict between sibling nodes and parent-child nodes can be avoided through the above formula.

TABLE 1. System Parameters.

Notation	Description
$\mathcal{L}_k^i$	The *i-th* node in the *k-th* layer
$\mathcal{C}_k$	Working cycle
c_g	Time period for data packet generation
c_t	Time period for data packet transmission
p_d	Packets generation probability
$d_{\mathcal{L}_k^i}$	Delay of the *i-th* node in the *k-th* layer
$\mathcal{D}_{\mathcal{L}_m^i, \mathcal{L}_n^j}$	Distance between $\mathcal{L}_m^i, \mathcal{L}_n^j$
$r_{\mathcal{L}_m^i}$	Radius of $\mathcal{L}_m^i$
N_y	Threshold for data packets number
T_y	Threshold for waiting time
d_0	Threshold distance (87m)
E_{elec}	Energy loss of transmitting circuit (50 nJ/bit)
ε_{fs}	Energy for power amplification (10 pJ/bit/m^2)
ε_{amp}	Energy for power amplification (0.0013 pJ/bit/m^4)

C. ENERGY CONSUMPTION MODEL

In this article, the energy consumption of sending data and receiving data is shown in the following formula:

$$\begin{cases} E = lE_{elec} + l\varepsilon_{fs} d^2 & \text{if } d < d_0 \\ E = lE_{elec} + l\varepsilon_{amp} d^4 & \text{if } d > d_0 \end{cases} \tag{5}$$

Here, E_{elec} is the energy loss of transmitting circuit, if transmission radius is less than threshold d_0, power amplification loss adopts the free space model; if radius is larger than the threshold, multipath attenuation model is adopted. ε_{fs} and ε_{amp} are the energy required for power amplification in these two models, l indicates the number of data bits.

In addition to the above-mentioned parameters, other parameters involved in this article are shown in Table 1. Some of the parameter values are obtained during calculations, so they are not provided in the table.

D. PROBLEM STATEMENT

1) MINIMIZE DATA COLLECTION DELAY

The purpose of this research is to optimize the average delay. Assuming that there are K layers in the network, and each layer has N nodes, the average delay is minimized as:

$$Min\ D = Min \left(\frac{\sum_{i=2}^{K} \sum_{j=1}^{N} D_i^j}{\sum_{i=2}^{K} \sum_{j=1}^{N}} \right) \tag{6}$$

2) MINIMIZE LIFE CYCLE OF NETWORK

The life of the network is defined as the death time of the first node. Suppose the average energy consumption of the *i-th* node in the network is ${}_i$, its initial energy is $\acute{E}_i$, and there are N nodes in the network. In order to maximize the life of the network, the life of the first dead node in the network should be maximized. Therefore, maximizing the network life cycle can be expressed as:

$$Max\ \eta = Max \left(\underset{1 \leq i \leq \mathcal{N}}{Min} \left(\frac{\acute{E}_i}{\omega'_i} \right) \right) \tag{7}$$

3) MAXIMIZE ENERGY UTILIZATION

Energy utilization refers to the ratio of the energy consumed by the network to the initial energy of the entire network. Use $E_{d,i}$ to represent the energy consumption of the node when the network dies. The energy utilization is maximized as:

$$Max\ \Gamma = Max\left(\frac{\sum_{i=2}^{K}\sum_{j=1}^{N} E_{d,i}}{\sum_{i=2}^{K}\sum_{j=1}^{N} \acute{E}_i}\right) \tag{8}$$

The research objectives are summarized as follows:

$$\begin{cases} Min\ D = Min\left(\frac{\sum_{i=2}^{K}\sum_{j=1}^{N} D_i^j}{\sum_{i=2}^{K}\sum_{j=1}^{N}}\right) \\ Max\ \eta = Max\left(\underset{1\leq i\leq \mathcal{N}}{Min}\left(\frac{\acute{E}_i}{\omega' i}\right)\right) \\ Max\ \Gamma = Max\left(\frac{\sum_{i=2}^{K}\sum_{j=1}^{N} E_{d,i}}{\sum_{i=2}^{K}\sum_{j=1}^{N} \acute{E}_i}\right) \end{cases} \tag{9}$$

IV. DESIGN OF THE DTC-ORS-DA SCHEME

A. MOTIVATION AND OVERVIEW

In this paper, we propose an efficient data aggregation strategy, which can greatly improve the delay and energy efficiency of data transmission. Specifically, the proposed strategy is improved through the following two aspects. First, we introduce the data packets threshold N_y and the waiting time threshold T_y in the data aggregation, and sets the thresholds of nodes in different layers to differentiate, so that packets can be sent to the next hop faster to avoid conflicts. Second, when selecting the next hop, we comprehensively consider factors such as the amount of data, waiting time, and remaining energy, so that the next hop node with the highest priority can be selected. In this way, data packets of nodes can be aggregated on a path, and nodes on the path can perform data aggregation operations on the aggregated data packets, which can greatly reduce the amount of data in the network and optimize energy consumption.

In this section, we explain the basic idea of the proposed strategy, and compare it with the previous Common Scheme (CS) to illustrate our research motivation. In the CS strategy, if the current node wants to send data to the next hop, it must wait until all child nodes have collected the data. As shown on the left side of the Figure 2, node $\mathcal{L}_2^2$ must collect the data of the node $\mathcal{L}_3^2$ and $\mathcal{L}_3^3$ before it can be aggregated and sent to the sink, and the node $\mathcal{L}_2^1$ must waiting for node $\mathcal{L}_3^1$. In the CS strategy, the data must be sent to the upper node after the lower node uploads the data. Therefore, it is a bottom-up process. If the lower-level nodes have not uploaded data, the upper-level nodes must wait. Obviously, the closer the node to the sink, the longer its waiting time, and when the network is large, the delay is very large.

Based on the above analysis, we have made the first step to improve the data fusion strategy. The idea is to add the data packets threshold N_y and the waiting time threshold T_y to the nodes in the network. When the amount of data owns by the node is greater than the threshold N_y or the waiting time of the node is greater than the threshold T_y, and the node satisfies the conflict avoidance model in formula (4), it can send data to its parent node.

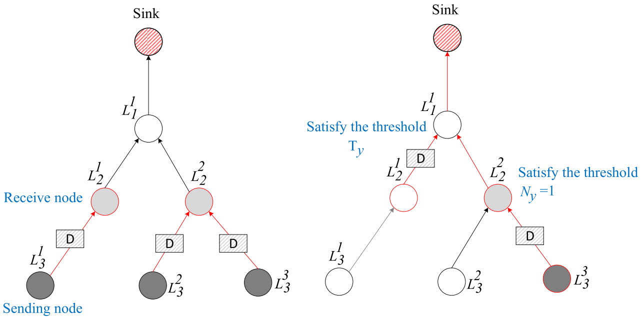


FIGURE 2. Packets transmission in DTC-ORS-DA scheme and CS.

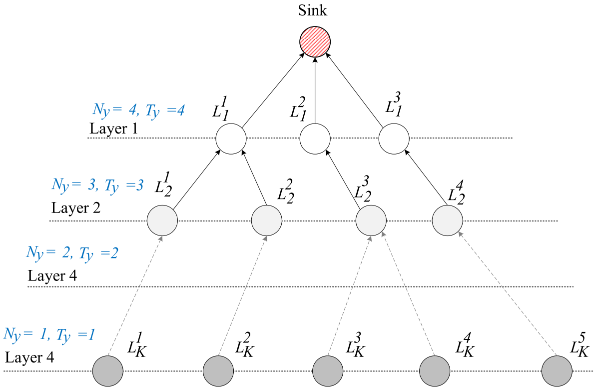


FIGURE 3. Network with different thresholds.

As shown on the right side of Figure 2, the node $\mathcal{L}_3^3$ sends data to the node $\mathcal{L}_2^2$ when the threshold condition is met, and in the same time slot, the node $\mathcal{L}_2^1$ also meets the time threshold condition and the conflict avoidance model, so the node $\mathcal{L}_2^1$ can send data to $\mathcal{L}_1^1$. Compared to the transmission method on the left, our improved scheme on the right can realize almost parallel data transmission. In our proposed strategy, some nodes close to the sink can send data to the sink first, thereby reducing the average delay. In addition, by analyzing the entire network, it is found that nodes separated by two layers can transmit data at the same time, thus making the number of layers of the network has a much smaller effect on the average delay.

When controlling data fusion through the threshold N_y and T_y, the configuration of the entire network is shown in Figure 3. Since the node located at the higher layers is farther away from the sink, in order to send the data of the nodes farther away from the sink faster and avoid large delays, we set the data packets threshold N_y and the waiting time threshold T_y to be very small, so that each data packet can be sent to the next hop faster. For the area closer to the sink, we set larger data packets threshold N_y and the waiting time threshold T_y, so that more data packets can be aggregated to achieve a better fusion effect. Therefore, in this article, we use the last layer as a benchmark, let the threshold of nodes at the last layer be very small, set the same data volume threshold and time threshold for each node of the same layer, and then gradually increase the threshold. Finally, the threshold of nodes in the layers close to the sink is larger, and the threshold of nodes far away from the sink layer is smaller. In this way, it realizes a fast forwarding at a long distance to reduce delay, and a high degree of fusion processing at a short distance to reduce energy consumption. This strategy of ''fast sending at far, high aggregation at near'' can realize the comprehensive optimization of network energy and delay.

In addition, how to select the next hop is also a problem to be solved. Therefore, we propose the algorithm of selecting the appropriate parent node according to the priority. We hope that the node can meet the data threshold requirement in a short time to reduce the delay. Therefore, the amount of data is considered as a priority factor. In the same layer of the network, some nodes may have been working and cannot get a rest, while other nodes may not receive data all the time, resulting in an imbalance in energy consumption. Nodes that consume a lot of energy die early, affecting the function of the entire network. Therefore, we use energy consumption as an influencing factor, which can prevent some nodes from consuming too much energy, thereby prolonging the life cycle. The node waiting time is considered as a factor, so that the node can receive data without waiting too long, which balances delay and energy consumption. Through the above analysis, it can be concluded that the priority is based on the comprehensive consideration of data volume, waiting time, and remaining energy. Therefore, the priority formula for selecting the next hop is calculated as follows:

$$\mathcal{P} = \alpha \frac{N'}{N_y} + \beta \frac{T'}{T_y} + \gamma \frac{E'}{E_y}, \alpha + \beta + \gamma = 1 \tag{10}$$

And N_y is the data packets threshold, T_y is the waiting time threshold, E_y is the total consumed energy, and the actual data packet of the node is N', waiting time is T', residual energy is E'. α, β, γ are the weights in the formula.

In summary, the data aggregation method we proposed in this paper adopts a differentiated threshold. When a node meets the conditions for sending data, it transmits data to the parent node with the highest priority.

B. ANALYSISI OF THE ALGORITHM

In this section, we give a specific node selection algorithm based on priority. First, we explain some symbols in the algorithm. Suppose the *n-th* node at the *m-th* layer is $\mathcal{L}_m^n$, it has three status, $Status_{\mathcal{L}_m^n} = 0$ represents that the node $\mathcal{L}_m^n$ is in listening and waiting state; $Status_{\mathcal{L}_m^n} = 1$ represents that the node is in the data sending status; $Status_{\mathcal{L}_m^n} = 2$ represents that the node is in the data receiving status. U_m is the set of nodes at the *m-th* layer where $Status_{\mathcal{L}_m^n} = 2$.$N_{y_m}^n$ and $T_{y_m}^n$ are packets threshold and waiting time threshold at the *m-th* layer. $H(\mathcal{L}_m^n)$ is the child node of $\mathcal{L}_m^n$,$F(\mathcal{L}_m^n)$ is the parent node of $\mathcal{L}_m^n$. N' is the data packets of the node, T' is the timer, λ is the data aggregation, P is the priority value.

Algorithm 1 Next Relay Selection Algorithm With Priority

```
1: For each node $\mathcal{L}_m^n$ Do
2:     Let $Status_{\mathcal{L}_m^n} = 0$
3:     For each $H(\mathcal{L}_m^n)$ of $\mathcal{L}_m^n$ Do
4:         If $Status_{H(\mathcal{L}_m^n))} = 2$ then
5:             Let $H(\mathcal{L}_m^n) \in \mathrm{U}_{m-1}$
6:         End if
7:     End for
8:     Foreach $F(\mathcal{L}_m^n)$ of $\mathcal{L}_m^n$ Do
9:     If $\mathcal{P}(F(\mathcal{L}_m^n))$ is max and $Status_{F(\mathcal{L}_m^n)} \neq 2$ then
10:        Let $Status_{F(\mathcal{L}_m^n)} = 2$
11:        Let $F(\mathcal{L}_m^n) \in \mathrm{U}_{m+1}$
12:    End if
13: End for
14: For each $\mathcal{L}_m^n$ $from 1$ $to n_i$ Do
15:    If $N'(\mathcal{L}_m^n) \geq Ny_m^n$ or $T'(\mathcal{L}_m^n) \geq Ty_m^n$ then
16:        If $\mathrm{U}_{m-1} = \emptyset$ and $\mathrm{U}_{m+1} \neq \emptyset$ then
17:            If $Status_{\mathcal{L}_m^n} = 0$ then
18:                Let $Status_{\mathcal{L}_m^n} = 1$
19:                Let $N'(F(\mathcal{L}_m^n)) = N'(F(\mathcal{L}_m^n)) + \lambda N'(\mathcal{L}_m^n)$
20:            End if
21:        End if
22:    End if
23: End for
```

The basic idea of the algorithm is: first give each node an initial state, $Status_{\mathcal{L}_m^n} = 0$. When the node $\mathcal{L}_m^n$ and all its child nodes have no data operations, the node $\mathcal{L}_m^n$ can enter the dormant state to save energy. Each round of data collection starts from the leaf nodes. The entire collection process starts from bottom to top, and continues to collect from each layer up until all the data in the network is collected by the sink. First, traverse all the child nodes of the node $\mathcal{L}_m^n$, and put the child nodes whose state is 2 into the set U_{m-1}, as shown in the algorithm in lines 3-7. Next, traverse all the parent nodes of node $\mathcal{L}_m^n$, change the state of the parent node with the highest priority to 2 and then put it into the set U_{m+1}, as shown in the algorithm in lines 8-13. Finally, traverse the nodes in each layer, change the state of the node whose state is 0, meet the threshold requirement and its child node and parent node are not in state 2 to 1, and send the data of node $\mathcal{L}_m^n$ to its parent node with highest priority, as shown in lines 14-23 of the algorithm.

C. ILLUSTRATION OF THE ALGORITHM

Next, we take the case shown in Figure 4 as an example to explain the above algorithm in detail. As shown in the figure, the data packet threshold N_y and waiting time threshold T_y of Layer k are both 1, which means that the node can send data after it has a data or waits for one time slot. Data is sent from Layer k to Layer *k-1*. The circle indicates the state of the node. 0 means the node is in the listening and waiting state, 1 means the node is in the sending state, and 2 means the node is in the receiving state. First, all nodes are in the

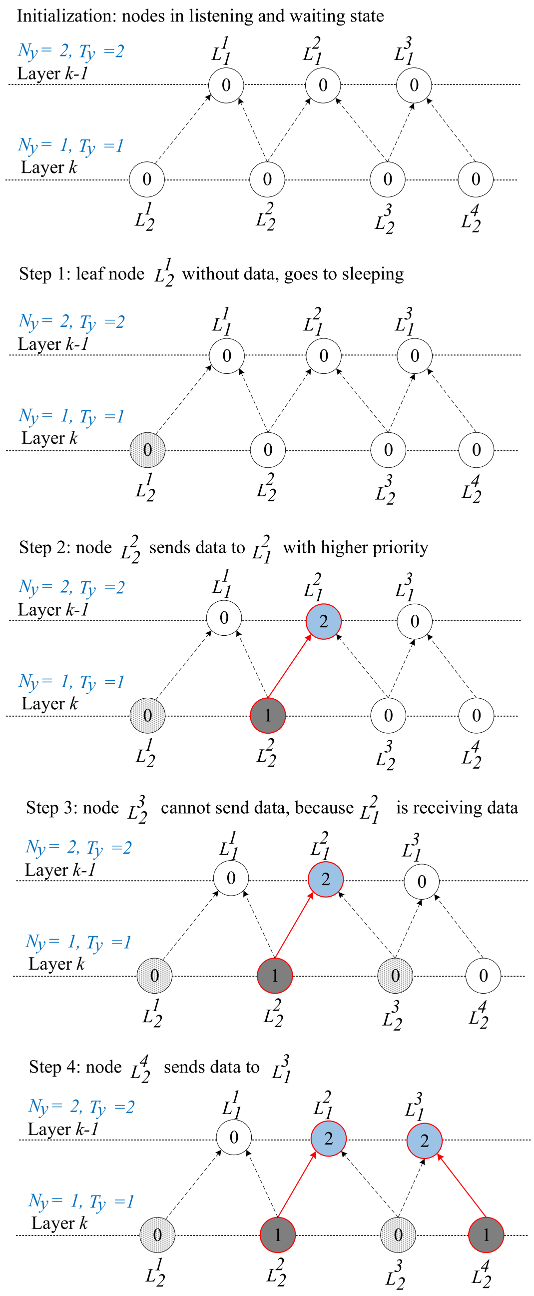


FIGURE 4. Illustration of the algorithm.

listening and waiting state during initialization. Then, start sending data from the node $\mathcal{L}_2^1$. First, $\mathcal{L}_2^1$ has no data to send, and it is a leaf node, so transfer $\mathcal{L}_2^1$ into sleeping state to save energy, as shown in step 1 in the figure. Then determine the node $\mathcal{L}_2^2$, the node $\mathcal{L}_2^2$ has data to send, and it has two father nodes $\mathcal{L}_1^1$ and $\mathcal{L}_1^2$. The two parent nodes are both in a waiting state. At this time, according to the priority formula, $\mathcal{L}_1^2$ has a higher priority as the next hop node, so $\mathcal{L}_1^2$ is selected as the receiving node and switch its state to 2. Then the node $\mathcal{L}_2^2$ sends data to $\mathcal{L}_1^2$, as shown in step 2 in the figure. Then, judge the node $\mathcal{L}_2^3$, although $\mathcal{L}_2^3$ has data to send, but because its parent node $\mathcal{L}_1^2$ is in the receiving state, in order to avoid conflicts, $\mathcal{L}_2^3$ keeps waiting, as shown in step 3 in the figure.

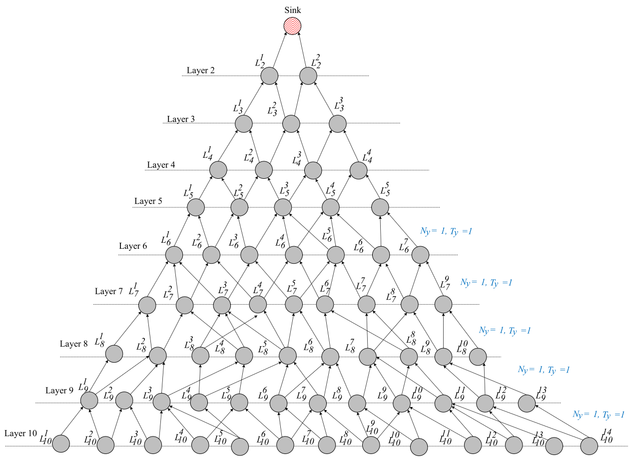


FIGURE 5. Network with $p_d = 100\%$.

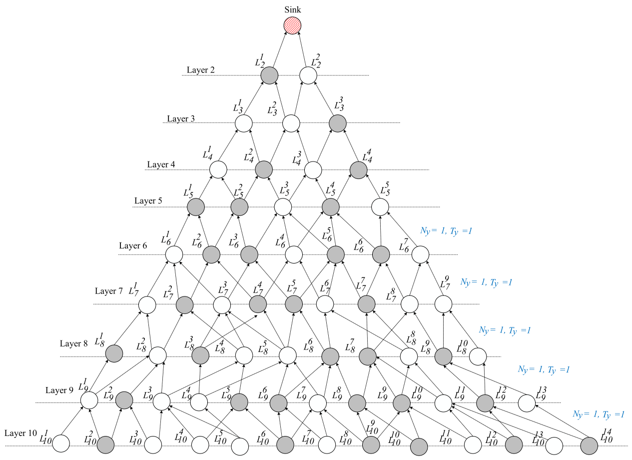


FIGURE 6. Network with $p_d = 50\%$.

Finally, determine the operation of $\mathcal{L}_2^4$, $\mathcal{L}_2^4$ has data to send, and he has only one parent node $\mathcal{L}_1^3$, so the node $\mathcal{L}_1^3$ switches to receiving state and send data to it. As can be seen from the figure, our method can avoid conflicts.

D. AN EXAMPLE OF THE SCHEME

According to the previous discussion, our solution is very complicated and challenging, because the value of N_y and T_y for each layer is uncertain, which makes it difficult to find the optimal solution. And the same applies to the values of the priority weights α, β, γ. In this section, we simulate the process of data transmission through theoretical analysis and research. First, we set the threshold of the five layers closest to the leaf to 1, and then selected different thresholds for comparison.

As shown in Figure 5 and Figure 6, a ten-layer network with different data generation probability p_d is constructed. The data packet generation probability p_d of each threshold case is 50% and 100% respectively. Suppose the priority weights are $\alpha = 1$, $\beta = 0$, $\gamma = 0$, that is, only consider the data packets. And energy consumed for transmitting an

TABLE 2. Cases of thresholds.

	L2	L3	L4	L5	L6-10
Case 1	2	3	4	5	1
Case 2	3	4	5	6	1
Case 3	4	5	6	7	1
Case 4	2	2	2	2	1
Case 5	3	3	3	3	1

TABLE 3. Network performance when $p_d = 100\%$.

	CS	Case 1	Case 2	Case 3	Case 4	Case 5
Delay	11.56	9.81	9.98	10.71	9.76	9.95
Times	$66e_s$+66 e_r.	$80e_s$+ $80e_r$.	$79e_s$+ $79e_r$.	$76e_s$+ $76e_r$.	$81e_s$+ $81e_r$.	$80e_s$+ $80e_r$.

TABLE 4. Network performance when $p_d = 50\%$.

	CS	Case 1	Case 2	Case 3	Case 4	Case 5
Delay	10.58	8.67	9.63	10.15	8.51	9.48
Times	$53e_s$+53 e_r.	$57e_s+$ $57e_r$.	$56e_s+$ $56e_r$.	$53e_s+$ $53e_r$.	$58e_s+$ $58e_r$.	$57e_s+$ $57e_r$.

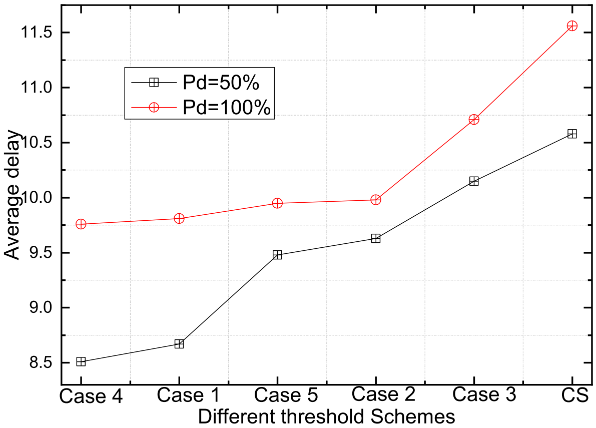


FIGURE 7. Average delay of CS and DTC-ORS-DA under different p_d.

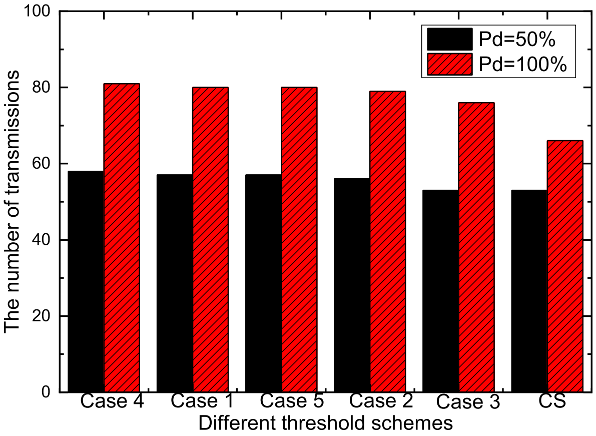


FIGURE 8. Number of transmission times of CS and DTC-ORS-DA under different p_d.

packet is e_s, energy consumed for receiving a packet is e_r, then the result is shown as follows:

As shown in Fig. 5, when the probability of each node generating data is 100%, the delay and transmission times can be obtained, as shown in Table 3.

As shown in Figure 6, when the probability of each node generating data is 50%, the delay and transmission times can be obtained, as shown in Table 4.

According to the data in Table 3 and Table 4, Figure 7 is obtained. As shown in Figure 7, we can see that the greater the threshold is set, the greater the delay, because the node takes longer to reach the threshold, which shows that the delay is related to the value of the threshold. The figure also shows the relationship between the delay and the probability of packet generation p_d. The larger the p_d, the greater the delay. When $p_d = 100\%$, delay is reduced by $7.35\% \sim 15.57\%$ compared with the CS scheme. When $p_d = 50\%$, the delay is reduced by $4.06\% \sim 19.56\%$.

According to the data in Table 3 and Table 4, Figure 8 is obtained. Since the number of transmissions is the same as the number of receptions, then we use the number of transmissions to illustrate. As shown in Figure 8, we can see that the smaller the threshold value is, the greater the number of sending and receiving data. Because when the threshold is small, some nodes close to the sink can send data to the sink in advance, increasing the number of sending and receiving.

Analysis of energy consumption. The energy consumption of the network is only related to the amount of data in the network and the distance between nodes, and the parameters that affect the amount of data are the probability of packet generation p_d and the data aggregation ratio λ. Therefore, when the p_d and the λ are fixed to 1, the data volume and data distribution of the two schemes are the same, and the energy consumption is the same. When p_d or λ is less than 1, the data distribution is different, because the number of data aggregations in this paper is greater than CS. So, the energy consumption of the DTC-ORS-DA proposed in this paper is reduced.

Analysis of life cycle and energy utilization. In the CS scheme, the node can only send data packets to the default parent node. Some nodes have a large number of child nodes, and some have a small number of child nodes, so the energy consumption of nodes in each layer is not balanced. In our proposed scheme, each node has multiple parent nodes to choose. When the priority is based on the remaining energy of the node, the energy consumption of the nodes in each layer is relatively balanced, which ensures that the life cycle. The life cycle and energy utilization are increased.

V. PERFORMANCE ANALYSIS

A. METHODS AND SETTINGS

We consider an experiment network consisting of a sink node and 500 sensor nodes. These sensor nodes are randomly deployed in the network with the center of the sink node, and the transmission radius of each node is 60 meters. Each sensor

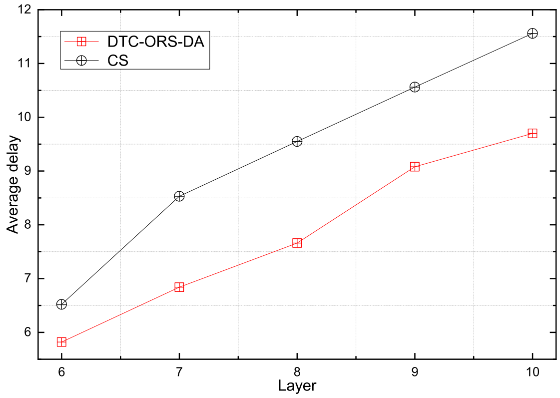


FIGURE 9. **Average Delay of DTC-ORS-DA and CS under different layers.**

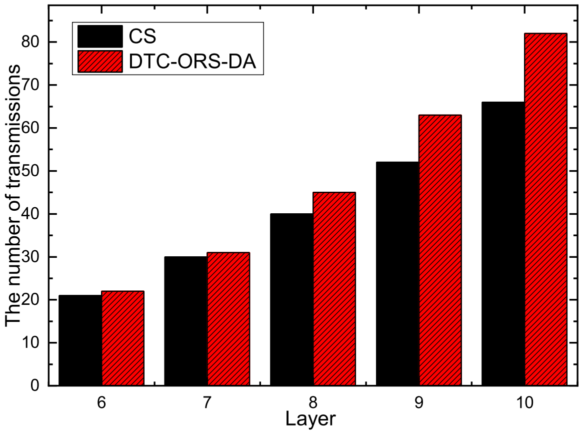


FIGURE 10. **Transmission times of DTC-ORS-DA and CS under different layers.**

node obtains data and send to the sink through multiple-hop routing. At the same time, we compare the DTC-ORS-DA scheme with Common Scheme (CS) in terms of delay, life cycle, and number of transmissions. The CS scheme is to send its aggregated data to its default parent node, and each time the parent node is fixed. When all the data packets are received by the sink, the simulation experiment is considered to be over. We regard a cycle as a unit time, which includes a packet generation period and an aggregation and transmission period.

Figure 9 compares the average delay of the DTC-ORS-DA scheme with the CS scheme under different layers in a cycle, where $\lambda = 1, p_y = 1, \alpha = 1$. The DTC-ORS-DA scheme in this article selects the threshold in Case 1. It can be seen that the delay in Case 1 of the DTC-ORS-DA scheme is less than that of the CS scheme, and the delay is reduced by $10.74\% \sim 19.91\%$.

Figure 10 compares the times of sending data between the DTC-ORS-DA scheme and the CS scheme under different layers in a cycle. It can be seen that the number of data sending times of the DTC-ORS-DA scheme is greater than that of the CS scheme under each layer, and the larger the layer number, the more the data transmissions increases, and the increase in the number of transmissions reduces the delay.

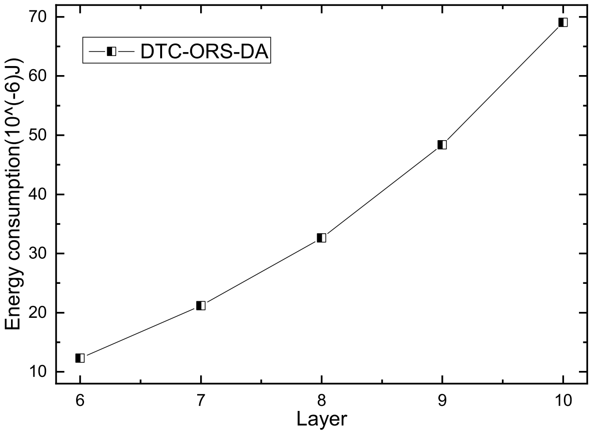


FIGURE 11. **Energy consumption of DTC-ORS-DA under different layers.**

As shown in Figure 11, when p_d and λ are fixed to 1, the energy consumption of the two schemes is the same, so only the energy consumption of the DTC-ORS-DA scheme under different layers is drawn.

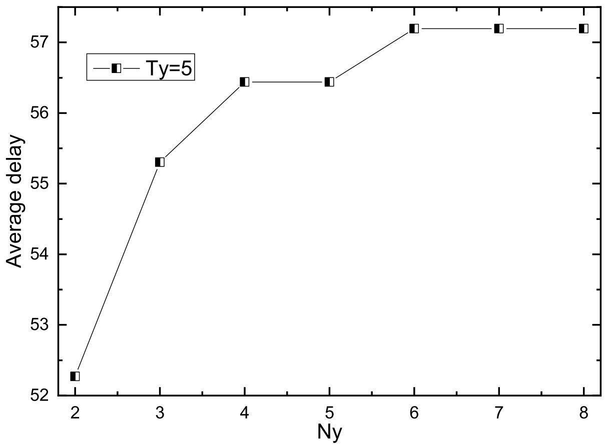
B. COMPARASION OF DIFFERENT THRESHOLDS

In this section, we tested the impact of different thresholds, p_d, λ on latency and energy consumption. We change the threshold of nodes in four layers near the sink. In the case of fixed T_y and different N_y, as shown in Figure 12(a). When N_y is set to a small value, the average delay increases accordingly. When N_y gradually increases, the average delay becomes stable, and the average delay is saturate under the larger packet volume threshold. And in the case of fixed N_y and variable T_y, as shown in Fig. 12(b). When T_y increases, the average delay also increases, and there is no tendency to stabilize. From the above figures, it can be seen that when N_y and T_y are set to be small, both N_y and T_y have an impact on the average delay, while N_y has little effect on the average delay at a larger threshold, and T_y has a larger effect on the average delay.

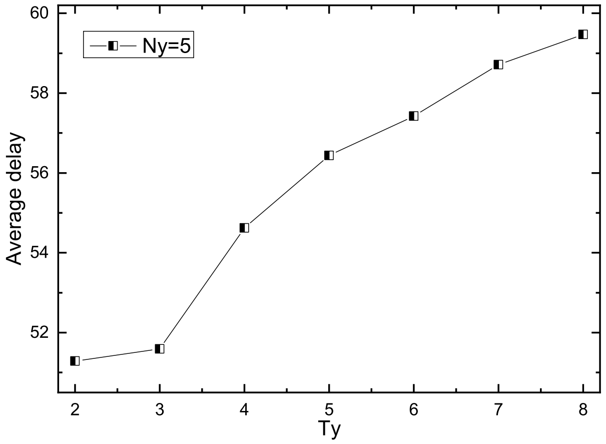
As shown in Figure 13, we compared the delay variation of different threshold cases under different packet generation probabilities p_d in five cycles. It can be seen that as the threshold increases, the average delay also increases correspondingly. When the p_d of a data packet becomes larger, the average delay also increases.

As shown in Figure 14, we compared the number of times for sending data for different threshold cases under different packet generation probabilities p_d in five cycles. It can be seen that when the p_d increases, the times of sending data also increases. When the layers close to the sink take a smaller threshold, the number of times to send data increases. When the threshold is larger, the number of times to send data is relatively small.

We compared the average delay changes of different threshold cases under different data aggregation ratios λ in



(a). Impact of N_y on average delay



(b). Impact of T_y on average delay

FIGURE 12. **(a). Impact of N_y on average delay (b). Impact of T_y on average delay.**

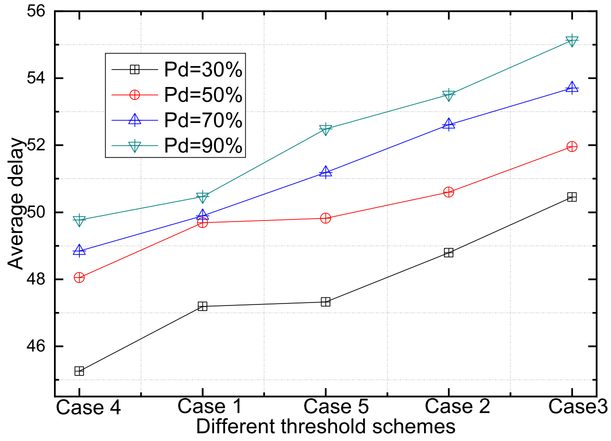


FIGURE 13. **Average delay of different threshold cases with different p_d.**

five cycles. It can be seen from Figure 15 that in different threshold cases, when the data aggregation ratio λ becomes larger, the average delay is reduced accordingly. Because a small data aggregation ratio λ reduces the amount of data in the network, most nodes cannot meet the data threshold, and it must wait until the time meets the time threshold. Especially when the threshold is in Case 3, the degree of delay reduction is great. In summary, the data aggregation ratio λ has a great impact on the average delay.

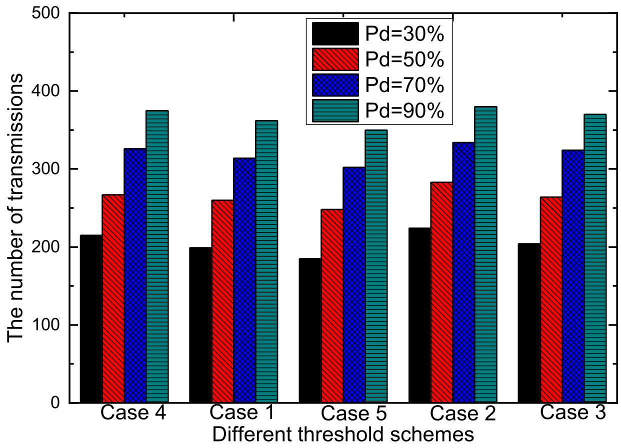


FIGURE 14. **Average transmission times of different threshold cases with different p_d.**

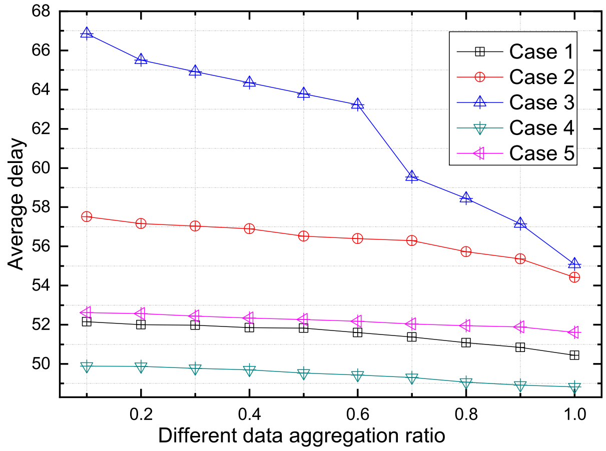


FIGURE 15. **Average delay of different threshold cases with different λ.**

As shown in Figure 16, we compare the impact of changes in λ and p_d on energy consumption under the threshold of Case 1. When the data aggregation rate λ becomes larger, the energy consumption of the entire network becomes larger. Because λ and p_d have an impact on the amount of data in the network, which has an effect on energy consumption.

As shown in Figure 17, we compared the impact of the distance between nodes on energy consumption. It is obvious that the energy consumption of nodes increases with distance.

We compared the data volume of the DTC-ORS-DA scheme and the CS scheme under different data aggregation ratios λ in five cycles. It can be seen from Figure 18 that the DTC-ORS-DA scheme has a smaller data transmission volume than the CS scheme at different data aggregation rates, so the energy consumption is lower than that of the CS scheme. The data volume is reduced by up to 82.64%.

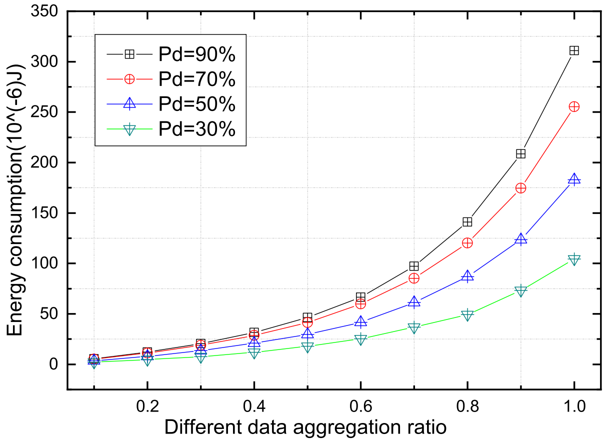


FIGURE 16. **Energy consumption under different λ and p_d.**

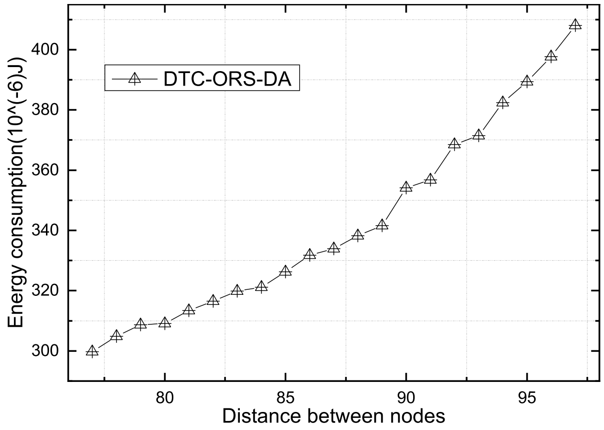


FIGURE 17. **Impact of distance between nodes on energy consumption.**

C. COMPARISON OF PRIORITY PARAMETERS

In this section, we compare the effects of different priority methods and no-priority cases on the delay and the energy consumption of nodes in three layers close to the sink.

We compared the delay of different threshold cases under $\alpha = 1$ in five cycles. It can be seen from Figure 19 that when $\alpha = 1$, the delay is obviously smaller than that of no priority, because the priority is based on the number of data packets, and the node can receive enough data without continuing wait, thereby reducing the delay.

We compared the delay of different threshold cases under $\beta = 1$ in five cycles. It can be seen from Figure 20 that when $\beta = 1$, the delay is significantly less than that of no priority. But for cases slightly larger than $\alpha = 1$, for example, in the Case 3, the delay is significantly smaller than the case of $\alpha = 1$. Because the priority is based on the waiting time of the node, the waiting time of the node is not too long. When the threshold is set to a large value, there is no need to wait until the time threshold, and the delay is reduced.

We compared the delay of different threshold cases under $\gamma = 1$ in five cycles. It can be seen from Figure 21 that when $\gamma = 1$, the delay has no advantage compared with the case of no priority. Under certain threshold settings, the delay is even greater, because the priority is mainly chosen to balance the energy consumption, no optimization for delay.

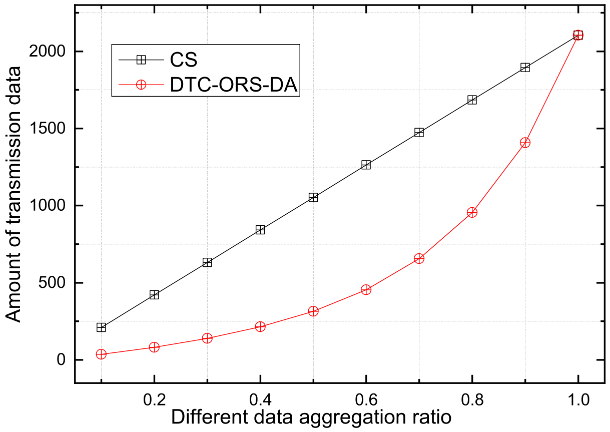


FIGURE 18. **Amount of transmission data in DTC-ORS-DA and CS with different λ.**

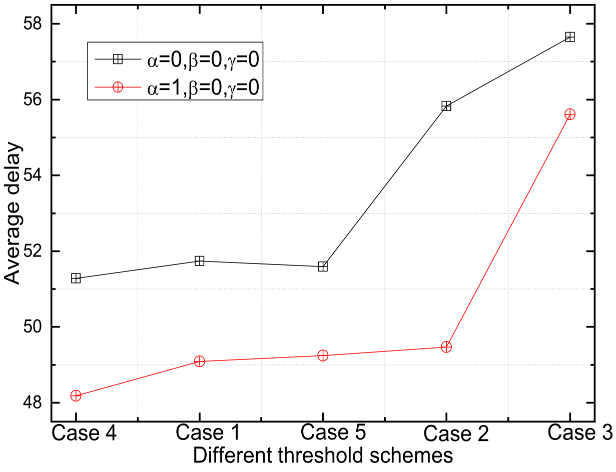


FIGURE 19. **Impact of priority weight α on delay.**

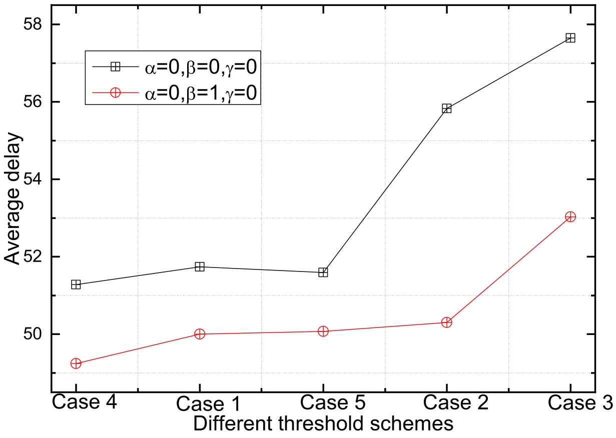


FIGURE 20. **Impact of priority weight β on delay.**

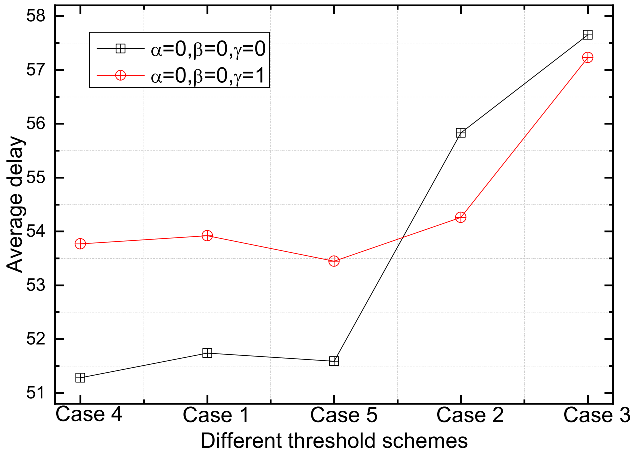


FIGURE 21. Impact of priority weight γ on delay.

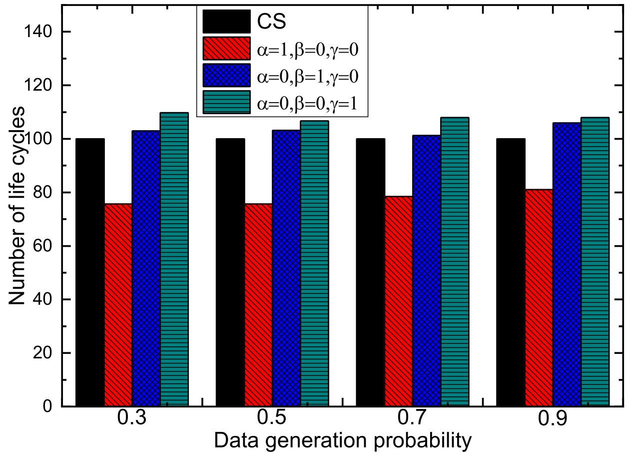


FIGURE 22. Impact of different priority weights on life cycle.

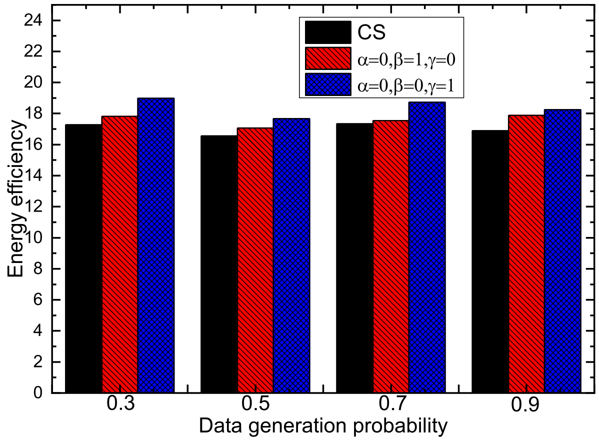


FIGURE 23. Impact of different priority weights on energy utilization.

We compared the life cycle under different p_d and different priority weights in five cycles. It can be seen from Figure 22 that when $\alpha = 1$, the life cycle of the network is reduced. When $\gamma = 1$, $\beta = 1$, the life cycle of the network is increased, and the life cycle is increased by 9.81% at most.

We compared the energy utilization efficiency of Case 1 under different p_d and different priority weights in five cycles. It can be seen from Figure 23 that when $\gamma = 1$ or $\beta = 1$, the energy utilization rate Γ of the entire network is improved, and the energy utilization rate of the DTC-ORS-DA scheme is increased by 6.67% $\sim$ 9.48% compared to the CS scheme.

VI. CONCLUSION

In this paper, we propose an efficient data aggregation strategy. It can be applied in a network with tree topology, which can significantly improve the performance in the delay and life cycle. In this scheme, we first proposed two key aggregation thresholds N_y and T_y. They limit the data packets and waiting time for data fusion, so that nodes can perform data fusion faster, thereby reducing latency. In addition, a priority-based node selection algorithm is presented, which enables the child nodes to dynamically select the parent node with the highest priority based on the number of data packets, waiting time, and remaining energy. Finally, we compare the proposed solution with the normal CS scheme. The experimental results are completely consistent with our expectations. Our solution reduces the average delay by 10.74%-19.91%, increases the life cycle by up to 9.81%, and increases the energy utilization by 6.67%-9.48%.

REFERENCES

[1] A. Li, W. Liu, S. Zhang, and M. Xie, "Fast multicast with adjusting transmission power and active slots in software define IoT," *IEEE Access*, vol. 8, pp. 226352–226369, 2020.

[2] X. Zhu, Y. Luo, A. Liu, M. Z. A. Bhuiyan, and S. Zhang, "Multiagent deep reinforcement learning for vehicular computation offloading in IoT," *IEEE Internet Things J.*, early access, Nov. 26, 2020, doi: 10.1109/JIOT.2020.3040768.

[3] O. Yan, A. Liu, N. Xiong, and T. Wang, "An effective early message ahead join adaptive data aggregation scheme for sustainable IoT," *IEEE Trans. Netw. Sci. Eng.*, early access, Oct. 29, 2020, doi: 10.1109/TNSE.2020.3033938.

[4] B. Marr. (2019). *How Much Data DoWe Create Every Day? The Mind-Blowing Stats Everyone Should Read*. Forbes. [Online]. Available: https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/#4c8deee860ba

[5] (2020). *Cisco Annual Internet Report (2018-2023) White Paper*. Accessed: May 30, 2020. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[6] X. Zhu, Y. Luo, A. Liu, W. Tang, and M. Z. A. Bhuiyan, "A deep learning-based mobile crowdsensing scheme by predicting vehicle mobility," *IEEE Trans. Intell. Transp. Syst.*, early access, Oct. 7, 2020, doi: 10.1109/TITS.2020.3023446.

[7] Y. Liu, T. Wang, S. Zhang, X. Liu, and X. Liu, "Artificial intelligence aware and security-enhanced traceback technique in mobile edge computing," *Comput. Commun.*, vol. 161, pp. 375–386, Sep. 2020.

[8] Q. Liu, P. Hou, G. Wang, T. Peng, and S. Zhang, "Intelligent route planning on large road networks with efficiency and privacy," *J. Parallel Distrib. Comput.*, vol. 133, pp. 93–106, Nov. 2019.

[9] T. Li, A. Liu, N. N. Xiong, S. Zhang, and T. Wang, "A trustworthiness-based vehicular recruitment scheme for information collections in distributed networked systems," *Inf. Sci.*, vol. 545, pp. 65–81, Feb. 2021.

[10] S. Huang, J. Gui, T. Wang, and X. Li, "Joint mobile vehicle–UAV scheme for secure data collection in a smart city," *Ann. Telecommun.*, pp. 1–22, Aug. 2020, doi: 10.1007/s12243-020-00798-9.

[11] X. Li, J. Tan, A. Liu, P. Vijayakumar, N. Kumar, and M. Alazab, "A novel UAV-enabled data collection scheme for intelligent transportation system through UAV speed control," *IEEE Trans. Intell. Transp. Syst.*, early access, Dec. 17, 2020, doi: 10.1109/TITS.2020.3040557.

[12] Q. Liu, Y. Tian, J. Wu, T. Peng, and G. Wang, "Enabling verifiable and dynamic ranked search over outsourced data," *IEEE Trans. Services Comput.*, early access, Jun. 11, 2019, doi: 10.1109/TSC.2019.2922177.

[13] G. Li, F. Li, T. Wang, J. Gui, and S. Zhang, "Bi-adjusting duty cycle for green communications in wireless sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2020, no. 1, pp. 1–55, Dec. 2020, doi: 10.1186/s13638-020-01767-5.

[14] B. Liang, X. Liu, H. Zhou, V. C. M. Leung, A. Liu, and K. Chi, "Channel resource scheduling for stringent demand of emergency data transmission in WBANs," *IEEE Trans. Wireless Commun.*, early access, Dec. 8, 2020, doi: 10.1109/TWC.2020.3041471.

[15] C. Zhou, Y. Gu, S. He, and Z. Shi, "A robust and efficient algorithm for coprime array adaptive beamforming," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1099–1112, Feb. 2017.

[16] W. Mo, T. Wang, S. Zhang, and J. Zhang, "An active and verifiable trust evaluation approach for edge computing," *J. Cloud Comput.*, vol. 9, no. 1, pp. 1–9, Dec. 2020, doi: 10.1186/s13677-020-00202-w.

[17] C. Zhou, Y. Gu, X. Fan, Z. Shi, G. Mao, and Y. D. Zhang, "Direction-of-arrival estimation for coprime array via virtual array interpolation," *IEEE Trans. Signal Process.*, vol. 66, no. 22, pp. 5956–5971, Nov. 2018.

[18] Q. Liu, Y. Peng, J. Wu, T. Wang, and G. Wang, "Secure multi-keyword fuzzy searches with enhanced service quality in cloud computing," *IEEE Trans. Netw. Service Manage.*, early access, Dec. 17, 2020, doi: 10.1109/TNSM.2020.3045467.

[19] M. Yu, A. Liu, N. N. Xiong, and T. Wang, "An intelligent game based offloading scheme for maximizing benefits of IoT-edge-cloud ecosystems," *IEEE Internet Things J.*, early access, Nov. 23, 2020, doi: 10.1109/JIOT.2020.3039828.

[20] M. Shen, A. Liu, G. Huang, N. N. Xiong, and H. Lu, "ATTDC: An active and trace-able trust data collection scheme for industrial security in smart cities," *IEEE Internet Things J.*, early access, Jan. 5, 2021, doi: 10.1109/JIOT.2021.3049173.

[21] W. Huang, K. Ota, M. Dong, T. Wang, S. Zhang, and J. Zhang, "Result return aware offloading scheme in vehicular edge networks for IoT," *Comput. Commun.*, vol. 164, pp. 201–214, Dec. 2020.

[22] X. Liu, M. S. Obaidat, C. Lin, T. Wang, and A. Liu, "Movement-based solutions to energy limitation in wireless sensor networks: State of the art and future trends," *IEEE Netw.*, early access, Sep. 30, 2020, doi: 10.1109/MNET.011.2000445.

[23] S. Huang, A. Liu, S. Zhang, T. Wang, and N. Xiong, "BD-VTE: A novel baseline data based verifiable trust evaluation scheme for smart network systems," *IEEE Trans. Netw. Sci. Eng.*, early access, Aug. 7, 2020, doi: 10.1109/TNSE.2020.3014455.

[24] S. Liu, G. Huang, J. Gui, T. Wang, and X. Li, "Energy-aware MAC protocol for data differentiated services in sensor-cloud computing," *J. Cloud Comput.*, vol. 9, no. 1, pp. 1–33, Dec. 2020, doi: 10.1186/s13677-020-00196-5.

[25] X. Liu, H. Song, and A. Liu, "Intelligent UAVs trajectory optimization from space-time for data collection in social networks," *IEEE Trans. Netw. Sci. Eng.*, early access, Aug. 19, 2020, doi: 10.1109/TNSE.2020.3017556.

[26] X. Liu, P. Lin, T. Liu, T. Wang, A. Liu, and W. Xu, "Objective-variable tour planning for mobile data collection in partitioned sensor networks," *IEEE Trans. Mobile Comput.*, early access, Jun. 1, 2020, doi: 10.1109/TMC.2020.3003004.

[27] Q. Liu, G. Wang, F. Li, S. Yang, and J. Wu, "Preserving privacy with probabilistic indistinguishability in weighted social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 5, pp. 1417–1429, May 2017.

[28] S. Huang, Z. Zeng, K. Ota, M. Dong, T. Wang, and N. Xiong, "An intelligent collaboration trust interconnections system for mobile information control in ubiquitous 5G networks," *IEEE Trans. Netw. Sci. Eng.*, early access, Nov. 17, 2020, doi: 10.1109/TNSE.2020.3038454.

[29] T. Wang, H. Luo, X. Zeng, Z. Yu, A. Liu, and A. K. Sangaiah, "Mobility based trust evaluation for heterogeneous electric vehicles network in smart cities," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 19, 2020, doi: 10.1109/TITS.2020.2997377.

[30] D. Chen, Z. Zhao, X. Qin, Y. Luo, M. Cao, H. Xu, and A. Liu, "MAGLeak: A learning-based side-channel attack for password recognition with multiple sensors in IIoT environment," *IEEE Trans. Ind. Informat.*, early access, Dec. 18, 2020, doi: 10.1109/TII.2020.3045161.

[31] X. Liu, A. Liu, T. Wang, K. Ota, M. Dong, Y. Liu, and Z. Cai, "Adaptive data and verified message disjoint security routing for gathering big data in energy harvesting networks," *J. Parallel Distrib. Comput.*, vol. 135, pp. 140–155, Jan. 2020.

[32] F. Lyu, H. Zhu, H. Zhou, L. Qian, W. Xu, M. Li, and X. Shen, "MoMAC: Mobility-aware and collision-avoidance MAC for safety applications in VANETs," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10590–10602, Nov. 2018.

[33] E. Zeydan, D. Kivanc, C. Comaniciu, and U. Tureli, "Energy-efficient routing for correlated data in wireless sensor networks," *Ad Hoc Netw.*, vol. 10, no. 6, pp. 962–975, Aug. 2012.

[34] L. Jiang, A. Liu, Y. Hu, and Z. Chen, "Lifetime maximization through dynamic ring-based routing scheme for correlated data collecting in WSNs," *Comput. Electr. Eng.*, vol. 41, pp. 191–215, Jan. 2015.

[35] J. E. Barcelo-Llado, A. M. Perez, and G. Seco-Granados, "Enhanced correlation estimators for distributed source coding in large wireless sensor networks," *IEEE Sensors J.*, vol. 12, no. 9, pp. 2799–2806, Sep. 2012.

[36] F. Davoli, M. Marchese, and M. Mongelli, "Non-linear coding and decoding strategies exploiting spatial correlation in wireless sensor networks," *IET Commun.*, vol. 6, no. 14, pp. 2198–2207, Sep. 2012.

[37] Z. Li, Y. Liu, A. Liu, S. Wang, and H. Liu, "Minimizing convergecast time and energy consumption in green Internet of Things," *IEEE Trans. Emerg. Topics Comput.*, vol. 8, no. 3, pp. 797–813, Jul. 2020.

[38] A. Shahraki, A. Taherkordi, Ø. Haugen, and F. Eliassen, "Clustering objectives in wireless sensor networks: A survey and research direction analysis," *Comput. Netw.*, vol. 180, Oct. 2020, Art. no. 107376.

[39] M. Bakshi, B. Jaumard, and L. Narayanan, "Optimum ConvergeCast scheduling in wireless sensor networks," *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5650–5661, Nov. 2018.

[40] X. Li, W. Liu, M. Xie, A. Liu, M. Zhao, N. Xiong, M. Zhao, and W. Dai, "Differentiated data aggregation routing scheme for energy conserving and delay sensitive wireless sensor networks," *Sensors*, vol. 18, no. 7, p. 2349, Jul. 2018.

[41] J. Gui, X. Dai, and X. Deng, "Stabilizing transmission capacity in millimeter wave links by Q-learning-based scheme," *Mobile Inf. Syst.*, vol. 2020, pp. 1–17, Feb. 2020, doi: 10.1155/2020/7607316.

[42] Q. Liu, Y. Peng, S. Pei, J. Wu, T. Peng, and G. Wang, "Prime inner product encoding for effective wildcard-based multi-keyword fuzzy search," *IEEE Trans. Services Comput.*, early access, Sep. 1, 2020, doi: 10.1109/TSC.2020.3020688.

[43] L. A. Villas, A. Boukerche, H. S. Ramos, H. A. B. F. de Oliveira, R. B. de Araujo, and A. A. F. Loureiro, "DRINA: A lightweight and reliable routing approach for in-network aggregation in wireless sensor networks," *IEEE Trans. Comput.*, vol. 62, no. 4, pp. 676–689, Apr. 2013.

[44] A. Liu, X. Liu, T. Wei, L. T. Yang, S. Rho, and A. Paul, "Distributed multi-representative re-fusion approach for heterogeneous sensing data collection," *ACM Trans. Embedded Comput. Syst.*, vol. 16, no. 3, p. 73, 2017.

[45] M. Dong, K. Ota, and A. Liu, "RMER: Reliable and energy-efficient data collection for large-scale wireless sensor networks," *IEEE Internet Things J.*, vol. 3, no. 4, pp. 511–519, Aug. 2016.

[46] M. Huang, A. Liu, N. N. Xiong, T. Wang, and A. V. Vasilakos, "A low-latency communication scheme for mobile wireless sensor control systems," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 2, pp. 317–332, Feb. 2019.

ANG LI received the master's degree in computer application technology from Central South University, China. He is currently an Associate Professor with the School of Computer Science and Technology, Hunan Institute of Technology, China. His research interests include crowd sensing networks, wireless sensor networks, and bioinformatics.

WEI LIU received the Ph.D. degree in computer application technology from Central South University, Changsha, China, in 2014. He is currently an Associate Professor and a Senior Engineer with the School of Informatics, Hunan University of Chinese Medicine, Changsha. He has published more than 40 articles in related fields. His research interests include complex network analysis, software engineering, and medical informatics.

LIJUN ZENG received the master's degree in computer application technology from Central South University, Changsha, China. He is currently an Associate Professor with the School of Computer Science and Technology, Hunan Institute of Technology. His research interests include intelligent information processing, pattern recognition, wireless sensor networks, and bioinformatics.

CHUNJI FA is currently pursuing the master's degree with the School of Computer Science and Engineering, Central South University, China. His research interest includes wireless sensor networks.

YAN TAN received the master's degree in computer application technology from the University of South China, Hengyang, China. She is currently a Lecturer with the School of Computer Science and Technology, Hunan Institute of Technology, China. Her current research interests include services-based networks, crowd-sensing networks, and wireless sensor networks.

•••