IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 11, 2019, accepted August 12, 2019, date of publication August 15, 2019, date of current version August 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2935462

Compressive Sensing-Based Clustering Joint Annular Routing Data Gathering Scheme for Wireless Sensor Networks

YICONG YUAN[1], WEI LIU[2], TIAN WANG[3], QINGYONG DENG[4,5], ANFENG LIU[1], AND HOUBING SONG[6]

[1]School of Computer Science and Engineering, Central South University, Changsha 410083, China
[2]School of Informatics, Hunan University of Chinese Medicine, Changsha 410208, China
[3]College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China
[4]Key Laboratory of Hunan Province for Internet of Things and Information Security, Xiangtan University, Xiangtan 411105, China
[5]College of Information Engineering, Xiangtan University, Xiangtan 411105, China
[6]Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114, USA

Corresponding author: Anfeng Liu (afengliu@mail.csu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61772554, Grant 61602398, Grant 61672447, and Grant 61572435, in part by the Hunan Provincial National Natural Science Foundation of China under Grant 2017JJ3316 and Grant 2019JJ50592, and in part by the Hunan Province Science and Technology Project Funds under Grant 2018TP1036.

ABSTRACT Compressed sensing technology is one of the effective techniques to effectively reduce the amount of data transmission in wireless sensor networks. Compressed sensing technology can reduce the amount of data that a node undertakes from n to m, where $m \ll n$, but we still hope to further reduce the amount of data that the node bears to improve network lifetime. In this paper, a Compressive Sensing based Clustering Joint Annular Routing Data Gathering (CS-CARDG) scheme is proposed to improve the network life. The key technology adopted by CS-CARDG scheme is: data is collected by cluster. The network first forms a cluster, and each node in the cluster sends the data packet to the cluster head. Each cluster forms m-dimensional data according to the requirements of the compressed sensing technology to ensure that the data can be recovered. When the cluster head node routes the m-dimensional data to the sink, the CS-CARDG scheme adopts a two-stage routing scheme with the same ring routing and shortest path that is completely different from the previous scheme. The same ring routing means that the cluster heads with the same number of sink hops are routed around the ring for one week to route the compressed sensing data of the same ring to a node in the ring. In this way, each sub-dimension data in the same ring is routed to the corresponding node of each ring through the same-loop route, and then the shortest-circuit strategy of the second phase is started. That is, from the outermost ring, the same fractal data is sequentially compressed from the outside to the inside, and is routed to the sink by the shortest path. In this round of data collection, the number of data packets that the nodes in the near-sink one-hop range bears is only m, and the nodes in the near-sink region directly send data to the sink node, thereby reducing the amount of data that the node bears to $\frac{m}{k}+1$, where k is the number of nodes within the node's broadcast radius.. In this paper, the compressed sensing strategies proposed in the past are compared by detailed theoretical analysis. The theoretical analysis results show that the CS-CARDG strategy proposed in this paper can effectively reduce the amount of data carried by nodes. This scheme reduces the amount of data in the network from $\left(1+\frac{r}{l_2}\right)\lambda+\left(1+\frac{r}{l_3}\right)\lambda+\ldots+\frac{m}{k}+d$ to $\frac{m}{k}+1$. Reduced the amount of data by at least 20%. In the network with $R = 480$ m, the energy utilization rate can reach more than 90%.

INDEX TERMS Wireless sensor networks, annular routing, compressed sensing, clustering, energy utilization.

I. INTRODUCTION

Currently, the number of sensing-based devices deployed on the Internet of Things (IoT) [1]–[4] is increasing, according Ref. [5]. The current sensing-based devices have far exceeded the number of humans, and are growing at a rapid rate. With the development of microprocessor technology [6], the computing and storage capabilities of these sensing- based devices have been greatly

The associate editor coordinating the review of this article and approving it for publication was Yuyu Yin.

enhanced [7]–[10]. For example, today's IPhone computing power is comparable to that of a personal computer 10 years ago [5]. Moreover, the size of the current sensing devices is getting smaller and smaller, thus giving it a wider range of applications [11]–[13]. As a result, the current network structure has undergone major changes [14]–16], the network architecture has seen a trend from the center to the decentralization, Edge computing [15], [17], 18], Fog computing [5], [14], [19], [20] is the change of this calculation mode. The Wireless Sensor Network [21]–[24] is an important part of the edge network. It consists of a network system consisting of multi-hop or self-organizing sensor nodes that collaborate and perceive relevant information in the coverage area and finally pass the information to the corresponding applications [25]–[27]. Wireless networks are widely used in military, agricultural, traffic information, ecological, infrastructure, and industrial applications [28]–[30]. On the military side, based on the randomness and dense distribution of wireless networks, it can be applied to harsh operational environments, such as for monitoring battlefields, etc. [31]. In agriculture, wire-less sensor networks can monitor the irrigation of crops and the migration of poultry, etc., and it has many applications in electrical automation and positioning technology [32], [33].

Wireless sensor networks are typically deployed in critical, or hazardous locations, where the energy is battery powered and extremely difficult to replace [3], [23], [34], [35]. Therefore, the energy in the wireless sensor network is very limited. How to save the energy of the node and extend the life of the network as much as possible is the most important research content in the wireless sensor network. The researchers also put forward many related researches [3], [36], [37].

Reducing the amount of data that wireless sensor networks need to transmit plays a crucial role in wireless sensor networks [38]–[40], because in wireless sensor networks, data communication is the most important energy consumption of energy consumption in the network, accounting for 70% of total energy consumption [3], [11], [21], [24], [31], [37]. Therefore, how to reduce the amount of data that nodes need to transmit is an important method to effectively reduce node energy consumption. Many methods for reducing data have been proposed [21] [24], [31], [33], [36], [37]. These methods include a wide range of methods, and there are data reduction methods for the network routing layer. For example, the simplest shortest path method reduces the energy consumption of the network by reducing the length of the routing path [41]. Another important way to reduce data is the data aggregation method [37], [42], [43]. This type of method uses time and space correlation between data sensed by sensing nodes. Therefore, when multiple perceived data packets meet, data fusion can be used to reduce redundancy between data packets to reduce nodes. The target of the amount of data that needs to be transmitted [37], [42], [43]. The data aggregation method is particularly effective for data collection of an aggregate fusion function [37]. The aggregate data fusion function refers to the max, min, and average functions in a class of monitoring data [37]. In such an application, applications need to know the maximum, minimum, and average values perceived in the network [37]. However, the data that is specifically perceived by each node is not concerned. In such a network, after the data of any multi-node meets, it can be merged into one data packet. Thus, the amount of data sent by each node in the network can be reduced to 1 [37]. However, data fusion is very limited in applications where there is redundancy between perceptual data, but the network with weak correlation between data is very poor.

Compressive Sensing is a newly proposed novel signal sampling theory which can effectively reduce the amount of data transmitted by nodes, and is very suitable for the sparsely characteristics of wireless sensor networks. The core idea is to recover a finite-dimensional signal from a small set of linear measurements when the signal is sparse in the base or dictionary [44]. According to the general data collection method, there are n nodes in the network, and the data readings available for the n nodes are available $X = [x_1, x_2, x_3, \ldots, x_i, \ldots, x_n]$ where x_i is the data reading for sensor Node i. According to the theory of non-compressive sensing, the amount of data of each node needs to go through a multi-hop route to reach the sink. In this kind of data collection, the amount of data undertaken by the far sink node is small, and the node within one hop of the near sink area bears the forwarding of all the node data in the network, that is, n data packets, thereby causing the node within the range of the near one sink. The amount of data undertaken is the largest, forming the so-called hotspots area. According to the theory of compressed sensing [45], in the network using perceptual theory, the node needs to transmit an m-dimensional observation value where $m \ll n$. In this way, the amount of data that each node bears is m packets. Compared with the non-CS method, the maximum amount of data that the nodes bear is significantly reduced, and the network lifetime is extended [45].

Although the compressed sensing technology can significantly reduce the amount of data in the near-sink area, the network life is prolonged, but the amount of data in this area is not fundamentally reduced, and the energy consumption is not balanced. The goal of this paper is to further reduce the amount of data in the near-sink region based on the theory of compressed sensing, which can further improve the network lifetime. So in this paper, a Compressive Sensing based Clustering Joint Annular Routing Data Gathering (CS-CARDG) scheme is proposed to further reduce the amount of data carried by the node to improve network lifetime. The main innovations of this paper's work compared to previous scheme are as follows:

(1)A Compressive Sensing based Clustering Joint Annular Routing Data Gathering (CS-CARDG) scheme is proposed to improve network performance, so that the amount of data per node near the sink area is reduced to $\frac{m}{k} + 1$, where k is the number of nodes within the node's broadcast radius.

(2) Based on clustering compression sensing, The CS-CARD scheme uses a two-stage routing scheme with the same-loop routing and shortest path that is completely different from the previous scheme. Therefore, the maximum amount of data that the nodes in the network can bear is significantly down. In CS-CARDG scheme, data is collected by cluster. The network first forms a cluster. Each node in the cluster sends the data packet to the cluster head. Each cluster forms m-dimensional data according to the requirements of the compressed sensing technology to ensure that the data can be recovered. When the cluster head node performs data routing to the sink for the m-dimensional data, the CS-CARDG scheme adopts a two-stage routing scheme combining the same-ring routing and the shortest path. The same-ring routing means that the cluster heads with the same number of sink hops are routed around the ring for one week to route the compressed sensing data of the same ring to a node in the ring. In this way, each sub-dimension data is routed to the corresponding node of each ring through the same-ring route, and then the shortest-circuit strategy of the second stage is started. That is, from the outermost ring, the same fractal data is sequentially compressed from the outside to the inside, and the shortest path is from the sink. In this round of data collection, the number of data packets that the nodes in the near-sink one-hop range bears is only m, and the nodes in the near-sink region directly send data to the sink node, thereby reducing the amount of data that the node bears to $\frac{m}{k}+1$. To the best of our knowledge, in the wireless sensor network, the amount of data carried by the node was reduced to $\frac{m}{k}+1$ for the first time.

(3) This paper compares the previous proposed compressed sensing strategies through detailed theoretical analysis. The amount of data per node near the sink area is reduced to $\frac{m}{k}+1$, which proves that CS-CARDG strategy can effectively reduce node commitment. The amount of data has greatly improved the network life. It is proved that the CS-CARDG scheme can effectively reduce the amount of data carried by nodes and greatly improve the network life. After comparative analysis, it is found that the data volume of the network is reduced from $\left(1+\frac{r}{l_1}\right)\lambda+\left(1+\frac{r}{l_2}\right)\lambda+\left(1+\frac{r}{l_3}\right)\lambda+\ldots+\frac{m}{k}$ to $\frac{m}{k}+1$, reducing the amount of data by at least 20%. In a network with $R=480$ m, the energy utilization of the CS-CARDG scheme can usually reach more than 90%.

The rest of this paper is organized as follows: Section II reviews related works. Section III describes the network model and defines problem statements of this paper. In Section IV, the detailed design of CS-CARDG scheme is presented. The results of the theoretical analysis are given in Section V. We conclude this paper in Section VI.

II. RELATED RESEARCH

There are also many strategies for saving energy consumption, such as MAC protocol, network layer strategy, application layer strategy, and cross layer optimization strategy. At present, due to the rapid development of artificial intelligence technology [46] [47], it is also used in the network to improve network performance. The fundamental goal of these strategies is to save energy as much as possible without affecting the wireless sensor's perception of the surrounding environment [21], [24], [31], [33], [36] [37]. For example, the main strategy for reducing node energy consumption at the MAC layer is to use the duty cycle mechanism [13], [31] [33], [36], [37]. Because the energy consumption of the node in the active state is several orders of magnitude higher than the energy consumption in the sleep state. Therefore, in order to save energy, the node should be in a sleep state as much as possible to save energy consumption [13], [31], [33], [36], [37]. Thus, the node works in a periodic active/sleep rotation to save energy. Obviously, the longer the node is in the sleep state in a cycle, the more energy the node saves and the longer the life. However, this may have other properties for the network, such as increasing the delay of data transmission [13], [31], [33], [36], [37]. Since the maximum energy consumption of the node is the energy consumption of the data communication, the data that the node needs to transmit can be reduced, thereby effectively reducing the energy consumption [48]. There are many studies to reduce the energy consumption of nodes from how to reduce the amount of data that nodes need to transmit. The strategies to reduce the amount of data that nodes need to transmit are data fusion and data compression sensing, and the compression-based method based on this paper is one of the most important methods which is emphasized in this section.

(1) The data fusion method is an important method to effectively reduce the amount of data, thereby reducing the energy consumption of nodes in WSNs [37], [42], [43]. The basic principles of data fusion are: Due to the high node density of wireless sensor network deployment, there is a correlation between the nodes' perceived data, that is, there is redundancy between the data. Therefore, when two or more data packets meet in the process of routing to the sink, data fusion can be performed, and redundant data between the data packets can be reduced, so that the amount of data that the node needs to transmit can be reduced [37], [42], [43]. For example: In the monitoring of the surrounding environment such as temperature, humidity, or events, the temperature and humidity perceived by the nodes in the same area may be the same, or there is correlation, or the same event is perceived. In the data fusion strategy, the general data fusion method is when the two data packets meet in the routing process, the data packet length before the data fusion is v_i and v_{j}, and the data volume that needs to be transmitted by the data fusion is $v_{\mathrm{a}}=\emptyset\left(v_i+v_j\right)$ where $\emptyset$ is a coefficient greater than 0 and less than 1. The most special data fusion method is no matter how many data packets before data fusion. After data fusion, it is still a data packet. The data length before data fusion is $v_1, v_2, \ldots, v_i \ldots v_k$. The length of the fusion is only the length of one packet v_{o}. This data fusion method is often referred to as the convergecast scheduling strategy [37], [43]. In order to effectively reduce the energy consumption of the

node, the scheduling principle often adopted by the converge cast scheduling strategy is a two-stage scheduling strategy, that is, the data transmission of the node is divided into a data receiving phase and a data transmitting phase [37], [43]. In the receiving phase of data transmission, the node only receives the data sent by the son node, and after receiving the data packets of all the son nodes and merging into one data packet, the node only performs one data transmission. After the node sends data, it no longer receives the data of the son node. In this round of data collection, any node sends data at most once. Thus, the energy consumption of the node can be minimized. In such a convergecast scheduling strategy, the network is generally converted into a tree structure. The data transmission starts from the leaf node and is transmitted to the sink node layer by layer. In this type of research, in addition to reducing the energy consumption of nodes, reducing the time required for data collection is also a key factor. In Ref. [37], Li et al. proposed a convergecast scheduling strategy based on unequal clustering. In their strategy, the clusters in the far sink area are small, and the clusters in the near sink area are large. Each node first sends its own packet to the cluster head. After the cluster head is merged into a data packet, the data is merged and routed between the cluster heads to the sink. Since the cluster of the far sink is small, the cluster of the far sink first completes the data collection in the cluster, and starts the data collection between the clusters in advance. The clusters near the sink are large. When the data collection in the cluster is completed, the inter-cluster routing of the far sink just arrives at the data packet. Therefore, after the cluster head node completes the data collection in the cluster, the inter-cluster data routing continues. In this way, the time required for data collection is reduced, and at the same time, the node only needs one active/sleep rotation in one round of data collection to reduce energy consumption [50]. In the previous strategy, after the cluster and data collection, the node is transferred to sleep, and then transferred to active during the inter-cluster data routing, which requires a secondary active/sleep rotation [37].

The commonly used data fusion is that a type of two or more data packets meet in a route and then merge into a fusion mode that is smaller than the original data packet and does not merge into one data packet [51]. In such data fusion, therefore, the most important thing is to make the data meet as much as possible when routing to the sink, because only data packets can be merged in the routing process to reduce the amount of data. Villas *et al.* [42] is an effective routing strategy based on the above ideas to increase the probability of data fusion to reduce the amount of data. In their DRINA strategy, routing is based on the shortest hop routing principle [52], [53]. That is, according to the hop count diffusion protocol during network initialization, each node in the network obtains the minimum number of hops required to reach the sink. Then, if the node generates data, it routes along the shortest-radial radial sink, that is, each node selects the node whose neighbor node is closer to sink to route the next hop [53]. However, unlike the ordinary shortest

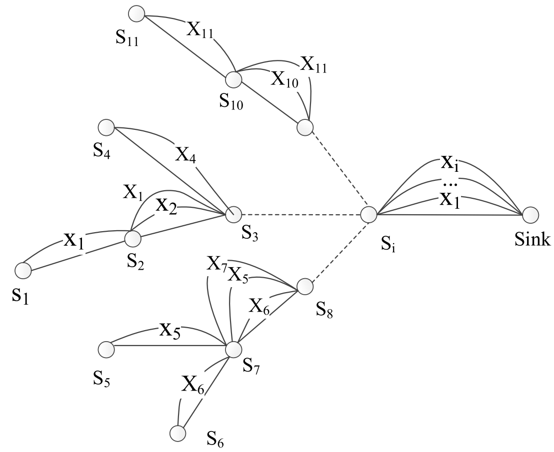

FIGURE 1. Non-CS scheme data collection method.

path, once the routing path is formed, the number of hops of the node on the routing path becomes 0. And spread out this information, so that the nodes around the path will be routed along the path when there is data routing. In this way, the data can be routed along the same path, increasing the probability of data packets meeting, thereby increasing the probability of data fusion and reducing the amount of data that the node needs to transmit. We proposed a ring-based data fusion strategy in Ref. [49] to maximize data fusion to reduce the amount of data that nodes need to transmit. Our adoption strategy is: Select an area in the network to form a ring route, and the data of other area nodes in the network are redirected to the ring for routing. When the data is routed to the ring, it is routed along the ring for a week and then routed to the sink. This allows all the data in the network to be merged before reaching the sink, thus reducing the amount of data to a minimum [49].

(2) Data collection scheme based on compressed sensing

American scholars Romberg and Donoho first proposed compressed sensing technology in 2006 [54], which has aroused widespread concern in academia. It can improve network transmission efficiency and improve network performance. Compressed sensing theory is not limited by the strict requirements of the classical Nyquist sampling theorem on sampling frequency. It can be compressed while sampling, and the result is compressed signal [44], [45], [54], [55]. The process is essentially to reduce the dimensionality of the signal. An important prerequisite for compressed sensing is the sparsity of the signal. The definition of sparsity is that if there are at most K elements that are not zero in a certain signal x, then the signal is considered to be K-sparse and can be expressed as $\|x\|_0 \leq K$ [54]. The data collected by the sensor node from the environment is not sparse in general, but can pass a certain transform domain according to a certain mapping relationship [56]. That is, the projection to an orthogonal transform base is sparse or nearly sparse, and the sparse signal is a compressible signal. The following is a graphical comparison of the compressed sensing method and other data collection.

Fig. 1 gives the data amount of the nodes under the non-CS scheme data collection mode. In the non-CS collection process, peripheral node data is sent to the sink through chain routing. s_1 sends its own packet x_1 to s_2, s_2 needs to pass two packets x_1 and x_2 to s_3, s_4 sends its own packet x_4 to s_3. Then s_3 passes x_1, x_4 and x_2 together to the next node, and so on, eventually collecting data to the sink node. In such a data collection method, packets of all nodes in the network are transmitted to the sink through the node s_i closest to the sink. Therefore, the node s_i bears the largest amount of data, and the number of packets it undertakes is the number of nodes n in the network. However, the nodes farthest from the sink, such as s_1, s_4, s_5, s_6, s_{11} only need to transmit one data packet, that is, sending their own data packets to their parent node. Therefore, in such a data collection method, the amount of data assumed between nodes is very uneven, and thus the amount of data undertaken by the near sink node is often n times the amount of data assumed by the farthest node. Its energy consumption is also nearly n times, which leads to the early death of nodes near the sink, and the early death of the near sink node often leads to the failure of the entire network. As shown in Fig. 1, if the node s_i dies early, the data of all other nodes in the network cannot be routed to the sink, thus causing the entire network to die. Therefore, the network lifetime depends on the time when the first node in the network dies, so the lifetime and energy efficiency of the network are not high under this data collection mode.

CS data collection is a data collection method based on compressed sensing theory [44], [45], [54]. The mathematical expression for the data collection process of compressed sensing scheme is as follows:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_m \end{bmatrix} = \begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1N} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{m1} & \phi_{m2} & \ldots & \phi_{mn} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{bmatrix} \tag{1}$$

where n is the total number of nodes in the network, $\mathrm{X} = [x_1, x_2, \ldots x_n]$ is original signals, Φ is m^*n dimension observation matrix. $\mathrm{Y} = [y_1, y_2 \ldots y_m]$ is a m-dimensional observation vector obtained by compression observation.

The compressed data collection method can be represented by the graph shown in Fig. 2. To use compressed sensing for data collection, you need to send the measured value, that is, the sum of the weights of the data, to the sink node, s_1 sends $\phi_{i1} * x_1 (1 \leq i \leq m) m$ packets to s_2, Then s_2 sends m packets of $\phi_{i1} * x_1 + \phi_{i2} * x_2 (1 \leq i \leq m)$ to s_3, s_4 sends $\phi_{i4} * x_4 (1 \leq i \leq m)$ to s_3, s_3 received $2m$ packets, then send $\phi_{i1} * x_1 + \phi_{i2} * x_2 + \phi_{i4} * x_4 (1 \leq i \leq m) m$ packets to the next node. Then, the last sink node receives $\sum_{j=1}^{N} \phi_{ij} * x_j (1 \leq i \leq m)$, a total of m observations. It can be seen that in the compressed sensing mode, the amount of data undertaken by each node is m data packets. Since m is much smaller than the number n of nodes in the network, the compressed sensing strategy can effectively reduce the data volume of the node and improve the network lifetime.

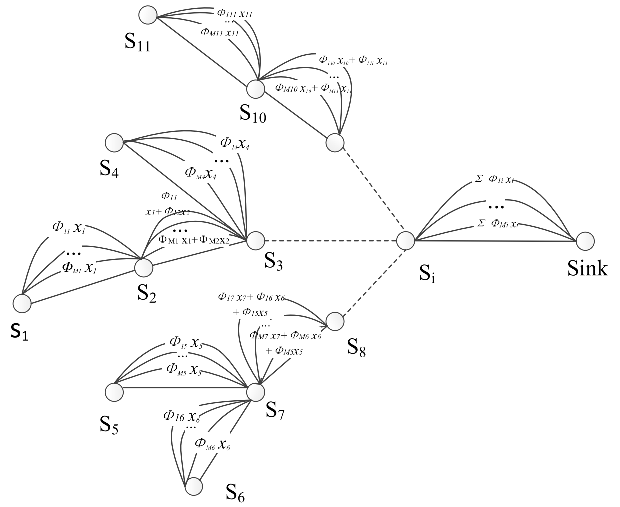

FIGURE 2. CS scheme data collection method.

It can be seen from Fig. 1 and Fig. 2 that the method of compressed sensing causes the amount of data that each node in the network bears to be m packets. But in fact, in non-CS scheme, the amount of data that the nodes in far sink area assume is still relatively small. For example, node s_1, s_4, s_5, s_6, s_{11} in Fig. 1 only needs to transmit one data packet, but in the compressed sensing, it needs to transmit $m$data packets, which means that the amount of data carried by these far sink nodes is increased. Therefore, some researchers have proposed a hybrid data collection scheme to reduce node energy consumption. That is, in the far sink area, when the amount of data undertaken by each node is less than m packets, a non-CS scheme is adopted. When the number of data packets undertaken by the node is greater than m data packets, the CS scheme is adopted, so that the maximum amount of data that the node bears is m data packets. This can reduce the energy consumption of the nodes in the far sink area.

Although a hybrid data collection method can reduce the amount of data that the far sink area nodes bear. But on the whole, the significance of doing so is not great. Because of this hybrid approach, the maximum amount of data that a node in the network assumes is still m packets. The lifetime of the network depends on the node with the most energy consumption in the network. Therefore, this method strictly reduces the energy consumption of some nodes, but the network life has not been improved as a whole. Therefore, we propose a multi-Strip Data Gathering (MSDG) scheme in Ref [45] that the amount of data that the node bears is less than m. The MSDG scheme is actually a further study on the hybrid strategy. The main point of the MSDG scheme is: Divide the network into many strips, in each strip, only the nodes in the center of the strip adopt CS scheme data collection mode, and the amount of data carried by the nodes is m data packets. The other areas of the strip are CS scheme data collection methods, and the amount of data carried by the nodes is less than m data packets. Thus, in a strip, only a small number of nodes bear the amount of data for m packets,

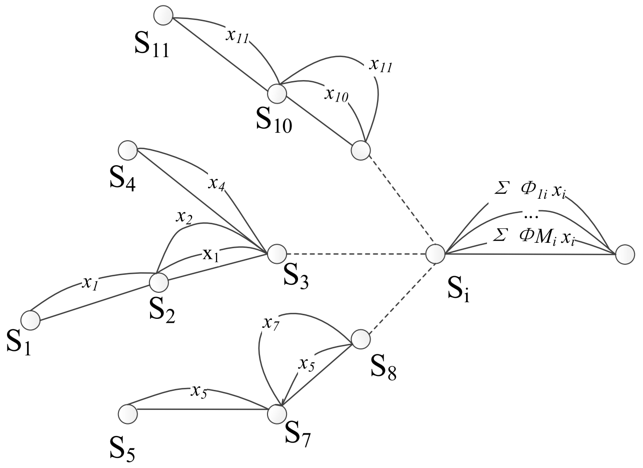

FIGURE 3. Non-CS and CS hybrid data collection scheme.

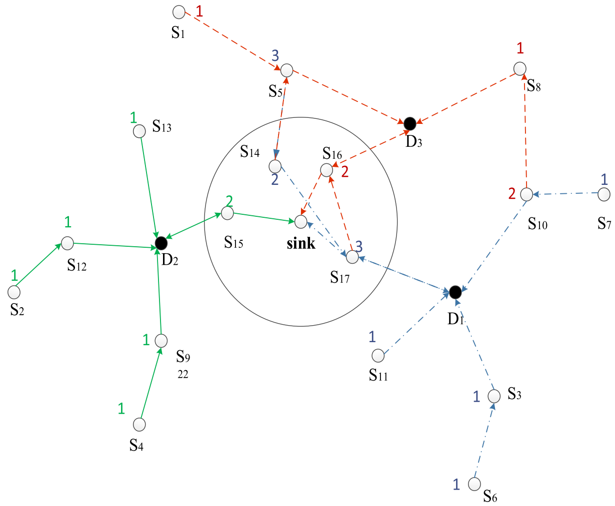

FIGURE 4. DCSS scheme diagram.

and most nodes bear less than m packets. If the strip keeps changing in the network, it will make the node sometimes become the center of the strip. At this time, the amount of data undertaken is m packets, and most of the time is non-strip center, and the amount of data is less than m packets. After the rotation, the average amount of data undertaken by the node is less than m packets. To the best of our knowledge, the MSDG scheme is the first study to reduce the amount of data carried by the node to less than m. However, the network structure of the MSDG scheme is complex and may affect its practicability.

In compressed sensing, the Gaussian matrix is commonly used as the observation matrix, while the literature [57] uses the sparse binary matrix as the observation matrix for the first time. Sparse binary matrix is composed of 0 and 1. Each matrix has a fixed number of 1 in each column, and all other elements are 0. We assume here that this fixed amount is d. The advantage of using this matrix is that some nodes do not need to provide data when collecting data, and can reduce the complexity of coding. The number of data packets that need to be sent by each sensor node is reduced from m to d, so a distributed compressed sparse sampling scheme (DCSS) is proposed [57]. The main research idea of the DCSS scheme is: Using the compressed sensing method, in the compressed sensing method, data collection is performed separately for each dimension of data. Since some nodes do not generate data on a certain dimension of data, nodes without data generation do not need to participate in this dimension of data collection, thereby reducing the number of nodes participating in data collection. Only need to send the node with data to the specified m sensor nodes, denoted as $D_1, D_2 \ldots D_m$. These m sensor nodes can then route data to the sink. Although this method has the potential to reduce the number of participating data collection nodes, the CS-CARDG scheme proposed in this paper is more advantageous. The main reasons are as follows: (a) For the nodes in the network, although the amount of data that the node itself needs to send is reduced to d, the node still needs to forward the data of other nodes, resulting in an increase in the amount of data. (b) For the nodes within the range of one sink from the sink, the CS-CARDG scheme proposed in this paper makes the amount of data carried by the nodes in the range of near-sink one hop in one round of data collection is $\frac{m}{k}+1$. In the DCSS scheme, the data amount sent to the sink by only m specified sensor nodes is $\frac{m}{k}+1$. However, if the area where the sink is located in a certain dimension of data collection, the node will bear a lot of data growth (see Fig. 4).

III. SYSTEM MODEL

A. NETWORK MODEL

There are n evenly distributed sensor nodes in the wireless network. The sink nodes are located in the network center for data collection. The density of sensor nodes distributed in the network is ρ. The life of the network can be defined by 'round'. At a certain time, the sensor node sends a data packet, and finally transmits the data packet to the sink node by using compressive sensing in a certain path.

B. ENERGY CONSUMPTION MODEL

Both signal transmission and reception produce energy consumption. The energy model used in this paper is the same as in [58]. The model used is free space and multipath fading channel (FM) model.

The energy consumption $E_{Tx}(i)$ of the L-bit information transmitted by s_i is:

$$E_{Tx}(i) = \begin{cases} \left(E_{elec} + \epsilon_{fs} \times d_i^2\right) \times L, d_i < d_0, \\ \left(E_{elec} + \epsilon_{mp} \times d_i^4\right) \times L, d_i \geq d_0, \end{cases} \tag{2}$$

The energy consumption of s_i receiving L-bit information is $E_{Rx}(i)$, as it is shown in Eq.3.

$$E_{Rx}(i) = E_{elec} \times L, \tag{3}$$

where $\epsilon_{fs} d_i^2$ and $\epsilon_{mp} d_i^4$ are the amplifier energies, d_i is the Euclidean distance between the transmitter and the receiver, and E_{elec} is the parameter of the transmitting or receiving circuit (ie the energy consumption of transmitting or receiving

TABLE 1. Network parameter.

Symbol	Description	Value
d_0	Distance threshold(m)	87
r	Communication radius(m)	40
E_{elec}	Transmitting circuit parameters(*nJ/bit)*	50
ε_{fs}	Power amplification for the free space(*pJ/bit/m*)^2	10
E_{init}	Initial energy(J)	0.5
ε_{amp}	Power amplification for the multi-path fading(*pJ/bit/m*^{4})	0.0013

a message), E_{amp} is the parameter describing the amplifier in the transmitting circuit [59]. In the Eq.2, for the energy consumed to transmit data, when the value of d_i is smaller than the threshold d_0, the free space model is used. Otherwise, use a multipath model. In Eq.3, it can be seen that the energy consumed to transmit data is only related to the amount of data transmitted, regardless of the distance.

The total energy consumption is:

$$E\left(i\right) = E_{Tx}\left(i\right) + E_{Rx}\left(i\right). \tag{4}$$

The parameters used in this paper are consistent with the literature [60], as shown in the following table:

C. DATA COLLECTION MODEL

In this paper, we mainly use two data collection models to collect data, one is based on compressed sensing data collection model, and the other is based on clustered routing data collection model. The following two models are analyzed.

(1) Compressed sensing model. In this strategy, except for the node within the range of one sink, the data is sent directly to the sink node, and other nodes adopt compressed sensing to collect data. As in Ref. [57], the sparse binary matrix is used as the observation matrix for data compression. The number of nodes in the network is n. After the observation matrix is compressed, the sink node collects m data packets to ensure data recovery.

(2) Clustering model. Clustering routing is for plane routing, which divides nodes in the network into two categories: cluster head nodes and member nodes. The cluster head node is located at the center of the cluster, and the nodes in the cluster send data to the cluster head node, so that the direct communication between the network node and the sink node is reduced, and the network energy consumption is largely alleviated. For nodes in the cluster, these nodes only need to communicate with the cluster head node, and do not need to communicate with other nodes, so it is convenient to maintain routing information, reduce the amount of information, and reduce consumption.

D. SELECTION AND ESTABLISHMENT OF OBSERVATION MATRIX

The strategy proposed in this paper is mainly based on the improvement of routing strategy based on compressed sensing theory, and the observation matrix is the key to the theory of compressed sensing. The specific expression of compressed sensing data collection is as follows:

$$\begin{bmatrix} \mathbf{y_1} \\ \mathbf{y_2} \\ \vdots \\ \mathbf{y_m} \end{bmatrix} = \begin{bmatrix} \phi_{\mathbf{11}} & \phi_{\mathbf{12}} & \cdots & \phi_{\mathbf{1n}} \\ \phi_{\mathbf{21}} & \phi_{\mathbf{22}} & \cdots & \phi_{\mathbf{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{\mathbf{m1}} & \phi_{\mathbf{m2}} & \cdots & \phi_{\mathbf{mn}} \end{bmatrix} \begin{bmatrix} \mathbf{x_1} \\ \mathbf{x_2} \\ \vdots \\ \mathbf{x_n} \end{bmatrix} \tag{5}$$

where $X = [x_1, x_1, \ldots x_n]$ represents the original signal, and $Y = [y_1, y_2, \ldots y_m]$ represents the signal obtained after compression, Φ represents the observation matrix.

As with the DCSS algorithm we use a sparse binary matrix as the measurement matrix Φ. The sparse binary matrix is composed of 0 and 1. The matrix has a fixed number of 1 per column. We assume here the fixed number is d. Its specific establishment is as follows:

Step 1: For each column, $d(1 \le d \le m)$ integers are randomly generated, that is, d rows are selected, and the value of the selected row is set to 1.

Step 2: If there are duplicate numbers in the randomly generated integer, that is, select the same row in a column, repeat step 1 until there is no repetition.

E. PROBLEM DESCRIPTION

Through previous analysis and research, we can find that to optimize the network and improve the network life, the focus is on the data bearing of the nodes closer to the sink node. Because the number of these nodes is limited, and other nodes want to transmit data to the sink node, they have to pass through these nodes, often these nodes are the nodes with the most energy consumption of the network. It is usually possible to refer to the time elapsed by the node with the first energy exhaustion of the network as the lifetime of the network [61]. Assuming that the initial energy of this node is E_i, the energy consumed by one round of data transmission is e_i, then the lifetime of the network is the quotient of the two. Therefore, extending the life of the entire network is to extend the life cycle of the node with the largest energy consumption, that is, to minimize the energy consumption of each node. The specific mathematical expression for this problem is as follows:

$$max(T) = \max \min_{0<i\le n} \left(\frac{E}{e_i}\right) = min \max_{0<i\le n} (e_i) \tag{6}$$

Energy utilization is also an important indicator to measure network conditions. A reasonable routing strategy can improve network utilization. The energy consumption of network nodes is as balanced as possible, which can improve energy utilization. The expression for energy utilization is as follows:

$$\max\left(\eta\right) = max(\sum_{i=1}^{n} E^{i}_{con} / \sum_{i=1}^{n} E^{i}_{init}) \tag{7}$$

In summary, the above problem can be described as Eq.8.

$$\begin{cases} max(T) = \max \underset{0<i\leq n}{min} \left(\frac{E}{e_i}\right) = \underset{0<i\leq n}{min}\ \underset{0<i\leq n}{max}\ (e_i) \\ \max\ (\eta) = max(\sum_{i=1}^{n} E_{con}^{i} / \sum_{i=1}^{n} E_{init}^{i}) \end{cases} \quad (8)$$

IV. DESIGING OF CS-CARDG SCHEME

A. RESEARCH MOTIVATION

After the analysis in II, it can be seen that collecting data using the non-CS method will make the load of the nodes located within one hop of the sink node too large, making the energy consumption extremely unbalanced. Collecting data using the CS scheme makes the energy consumption more balanced than that of the non-CS scheme. However, since the "many-to-one" transmission characteristic of the nodes has not changed, the closer the node closer to the sink node, the larger the amount of data that needs to be received. This in turn leads to greater energy consumption and still significantly reduces network life.

In the DCSS scheme, sink node needs to receive m measurements to recover all data. Therefore, we specify m sensor nodes according to the observation matrix and mark these sensor nodes as $D_1, D_2 \ldots D_m$. Fig.4 shows the schematic diagram of the DCSS scheme. If $m = 3$, $d = 1$, then select 3 sensor nodes D_1, D_2, D_3, and the number of data packets that each node needs to send is 1. First, a sparse binary matrix Φ is generated by sink node, and each sensor node in the network stores it locally. For each sensor node $s_j (1 \leq j \leq n)$, if $\phi_{ij} \neq 0$, the sensor node s_j sends its reading x_j to the specified sensor D_i by the shortest path algorithm. As shown in Fig. 4, D_1 collects data of sensor nodes $S_{11}, S_3, S_6, S_5, S_7$, D_2 collects data of sensor nodes $S_2, S_{12}, S_9, S_4, S_{13}, S_{15}$, D_3 collects data of sensor nodes $S_1, S_{17}, S_{16}, S_8, S_{10}, S_{14}$. For sensors D_1, D_2, D_3, each calculates and stores the sum of the data it receives. Finally, D_1, D_2, D_3 send the result to the sink through the shortest path, and the sink performs the recovery of the compressed data.

Fig. 4 shows the number of packets that each sensor node needs to bear. For node S_{17}, it not only sends its own data to D_3, but also forwards the data from S_5. D_1 also sends data to the sink node via S_{17}, resulting in the number of packets of S_{17} being 3. Other nodes have the same problem, for node S_{10}, S_7 will forward the data to D_1 through S_{10}, and S_{10} will also send its own data to D_3, resulting in the number of data packets of S_{10} being 2. In an actual network, the packet data to be forwarded by each node is much larger than this value.

It can be seen from the above analysis that although the DCSS scheme is improved compared with the previous schemes, only m sensor nodes send m data packets to the sink node. However, when the other sensor nodes in the network (except for the m sensor nodes) send d data packets to the m sensor nodes, part of the data packets of the d data packets will pass through the near sink area (such as the sensor node S_{17} in Fig. 4). Because the area of the near sink area is πr^2, the number of nodes in this area is $k = \pi r^2 \rho$, where r is the sending radius of the node and ρ is the node density. As long as there is a need to transmit the i-th dimension data of the node in this area, the route of the i-th dimension must pass through the near sink area. Even if the node in the near sink area does not exist in a certain dimension, such as the j-th dimension data, the j-th dimension data may still be routed through the near sink area. Let the probability of routing through the near sink area be P_s. Then, when the data of the nodes in the network is sent to m nodes, the load to the near sink area is $P_s m$. And finally the m sensor nodes will also pass through the near sink area when transmitting data to the sink, thus, the load of the near sink area is m. The node of the near sink area also sends kd packets to m sensor nodes. In summary, the load of the near sink area is $(1 + P_s)\, m + kd$, which is the ideal load. However, in the actual environment, when other nodes in the network transmit data to the m nodes through the near sink area, there is often more than one hop, which may be two hops, or three hops pass through the near sink area. Therefore, the load caused to the near sink area will be multiplied and the network life will be reduced. In addition, it is very difficult to form a route for data collection in such a scheme. Because, in advance, you need to know which nodes have data. Then, you need to design complex routes to construct all the nodes with the i-th dimension data in a routing path. The data on each node is dynamically changing, so constructing such a fine route is very difficult and costly. How to solve the problem of increasing the amount of data caused by forwarding, how to make the network energy consumption more balanced, how to further reduce the amount of data in the near sink area, this is our research motivation.

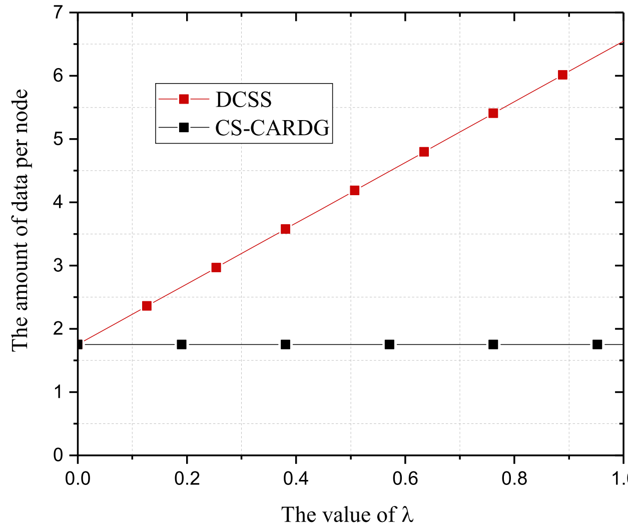

FIGURE 5. Comparison of DCSS and CS-CARDG data volume.

B. SCHEME OVERVIEW

1) INTRODUCTION OF THE SCHEME

The nodes in the network are evenly distributed. First, the network is evenly clustered. There are a total of n nodes,

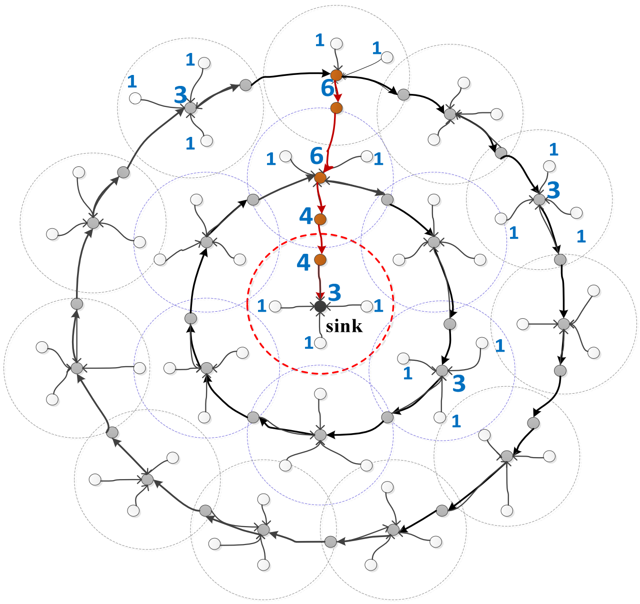

FIGURE 6. **CS-CARDG strategy diagram.**

and the number of nodes in each cluster is k. The density of the nodes in the cluster is ρ, and the size of the cluster is πr^2, where r is the communication radius. The cluster head is located at the center of the cluster, that is, the nodes in the cluster can directly communicate with the cluster head. The number of rings is p, and re-clustering is performed once every h round of data transfer.

Fig. 6 is a general strategy diagram of CS-CARDG. In this strategy, nodes within one hop from the sink node send data directly to the sink node, and other nodes use compressed sensing for data collection. We draw on the DCSS routing method to collect the data in different dimensions. For each row $i(1 \leq i \leq m)$ of the measurement matrix Φ, the sparse binary matrix is routed as follows:

For each node in the cluster, if $\phi_{ij} \neq 0(1 \leq i \leq m, 1 \leq j \leq n)$, $\phi_{ij} \cdot x_j$ means that its reading is sent directly to the cluster head node, and the cluster head node calculates the sum of the obtained data $\sum \phi_{ij} \cdot x_j$. For a cluster head node, if $\phi_{ij} \neq 0$ then add its own data to the sum obtained above. Otherwise, it is not added. Concatenate the cluster head nodes into a number of rings, and specify a cluster head for each ring as the initial data collection node. Data collection is performed around the ring from the initial node, and the collected data is added. Then, data is collected from the initial node of the outermost ring to the inner layer, and the data is added and finally sent to the sink node.

Finally, the sink node obtains $\sum_{j=1}^{N} \phi_{ij} * x_j(1 \leq i \leq M)$, a total of m observations, and the sink node performs data recovery.

We use the dynamic ring method to select the starting node on the ring every r rounds. After each h $(h > r)$ round, we cluster again, select other nodes in the cluster as the cluster head, and re-establish the ring.

2) CS-CARDG SCHEME VS. DCSS SCHEME

as shown in Fig.6, if under the same conditions, ie $m = 3$, $d = 1$, the number of packets assumed by each node in the cluster in the CS-CARDG scheme is 1, and the number of packets undertaken by the cluster head node is 3. For the near-sink area, only one node in the cluster bears 4 packets, and the other nodes bear 1 packet. Since the starting node not only transmits 3 data packets along the ring to the next node, but also transmits 3 data packets from the outside to the inside to the sink node. So the number of packets to be assumed by the starting node is 6. The nodes in the clusters passing through the packet from the outside to the inside of the sink node must not only send their own data packets to the cluster head node, but also transmit 3 packets sent by the cluster head node. So the number of packets undertaken is 4.

It can be seen from the comparison between Fig.4 and Fig.6. For a node in a cluster, the node does not need to forward data sent by other nodes, so that the amount of data of each member node in the cluster is d, which is 1. For the near sink area, the total number of packets is reduced from 9 to 7, for an actual network, this reduced amount of data will be larger. From the theory of compressed sensing, it is known that at least m measured values are needed to recover the original data. In the CS-CARDG scheme, only one node in each round directly communicates with the sink node to send m data packets, so that the amount of data in this area is minimized.

In the DCSS algorithm, if the sensor node needs to forward data to the D_1, D_2, D_3 through the near sink node, the probability is λ. Then, in the network environment shown in Fig.4 and Fig.6, as for the near-sink node, the relationship between the amount of data of the DCSS scheme and the CS-CARDG scheme and λ is shown in Fig.5.

As can be seen from Fig.5, the data capacity of the DCSS scheme increases with the increase of λ, while the data capacity of the CS-CARDG scheme remains unchanged. In addition, it can be seen that for the near sink node, when $\lambda = 0$ (When the sensor node sends data to D_1, D_2, D_3, it does not pass through the near sink area.), the data capacity of DCSS scheme is the same as that of CS-CARDG scheme. In other cases, the data carrying capacity of the DCSS scheme is greater than the data carrying capacity of the CS-CARDG scheme.

C. THE DESIGN OF CS-CARDG SCHEME

The CS-CARDG strategy divides the network into clusters for data collection. Except for the nodes with the sink node as the cluster head, the other nodes collect data based on the compressed sensing. The specific routing strategy is as follows:

(1) The selection of the initial starting node for each ring route. Select the cluster head node farthest from the sink node as the starting node of the outermost ring and record it as A_p. Select the node closest to the sink node from the cluster head node closest to the A_p node as the starting node A_{p-1} of the next ring, and so on,

and determine the starting node $A_i(1 \leq i \leq p)$ of each ring.

(2) Creating a ring route. Starting from the starting node $A_i(1 \leq i \leq p)$, according to the [62] right-hand rule, in the cluster head closest to the distance $A_i(1 \leq i \leq p)$, select a cluster head, which is the cluster head closest to the sink distance and the distance from A_i to sink as the next destination. Use the shortest path algorithm to reach this cluster head and sum the data, finally return to A_i.

(3) For member nodes, if $\phi_{ij} \neq 0(1 \leq i \leq m, 1 \leq j \leq n)$, the data is directly sent to the cluster head node.

(4) For the cluster head node, the data collected in the cluster is added. If $\phi_{ij} \neq 0$, its own data is added to the sum obtained above. Otherwise, it is not added.

(5) For points within one hop from the sink node, the data is sent directly to the sink node.

(6) Routing on the ring. After the nodes in the cluster send the data to the cluster head node, each ring starts to collect the sub-dimensional compressed sensing data from the starting node one week, and the collected data is added, and finally the starting node is collected. After m times of collection, the starting node gets m packets.

(7) The data on the ring is collected to the sink node. Starting from the outermost starting node A_p, the data is sent to the sending node A_{p-1}, the data is summed, routed from the outer to the inner layer, and finally the data is sent to the sink node. After the sink node receives m observations, the sink node performs data recovery.

D. FEATURES AND ADVANTAGES OF THE CS-CARDG STRATEGY

Such a data collection strategy has the following characteristics:

(1) Clustering the network reduces the direct communication between nodes and sink nodes in the network, which helps to reduce the data capacity of the near sink node. The nodes in the cluster send data to the cluster head in a compressed sensing manner. Different from the previous compression sensing collection, each node needs to send m data packets. We use a sparse binary matrix as the observation matrix here. Since there are d 1s per column in the observation matrix, each node in the cluster only needs to send $d(d < m)$ packets.

(2) Using compressed sensing for data collection, the sink node needs to collect at least m measurements to ensure data recovery. In the CS-CARDG strategy, only one node in each round communicates directly with the sink node. That is, in the point within the range of one sink from the sink, only one node bears the number of packets m. Therefore, the average amount of data assumed by each node in the cluster with the sink node as the cluster head is minimized.

(3) In this way, the amount of data to be carried on the ring node is larger than other nodes. Therefore, using the dynamic ring method, the starting node is selected again on the ring every r rounds. Each time h (h>r) is transmitted, the cluster is re-clustered, and other nodes in the cluster are selected as cluster heads to re-establish the ring.

The main advantages of the CS-CARDG strategy are as follows:

(1) The data capacity of the near sink node is greatly reduced, and the network lifetime is prolonged. In the previous collection algorithms, neither the non-CS strategy nor the CS strategy changed the many-to-one characteristics of network data collection. Therefore, it is impossible to fundamentally reduce the data carrying capacity of the near sink point. And for the better algorithm DCSS proposed now, when the sensor node sends data to the m cluster head nodes, it will pass through the near sink area, and finally the m cluster head nodes will pass through the near sink area when transmitting data to the sink, which will still increase the data volume of this area. In the CS-CARDG strategy, only one node in each round communicates directly with the sink node, so that the average amount of data assumed by each node in the cluster with the sink node as the cluster head reaches the minimum $\frac{m}{k} + 1$.

(2) Increasing network energy utilization and more balanced energy consumption. In this network, using the idea of dynamic ring, the starting node is re-selected on the ring every r rounds, and each time $h(h>r)$ is transmitted, it is re-clustered. This makes the energy consumption of nodes of other clusters approximately equal except for the near sink area, while the energy consumption of the near sink area is minimized. As long as the energy of other clusters is not exhausted, data can be transferred to the sink. And it will not appear, the energy in the near-sink area is exhausted, causing the entire network to be unavailable, causing a lot of energy waste. As d is smaller, the difference in energy consumption between other clusters and near-sink clusters is smaller, and the energy consumption of the network is more balanced, and the energy utilization rate is higher. As for DCSS scheme, it causes the network energy consumption to be unbalanced due to the different amount of data that each node bears.

E. THE DETAILED DESIGN OF CS-CARDG STRATEGY

V. PERFORMANCE ANALYSISP

This chapter mainly analyzes the data volume and energy consumption of non-CS scheme, CS scheme, DCSS scheme and CS-CARDG scheme, and then compares and analyzes the advantages of CS-CARDG scheme.

Assume the following energy analysis premise: the nodes in the network are evenly distributed, the number of nodes in the cluster is the same, the cluster size is the same, the cluster head is located at the center of the cluster, and the nodes in

Algorithm 1 CS-CARDG Strategy

Input: Raw data of the sensor nodes
Output: recovered data by sink node

Step 1: Selection of the starting node of the initial ring route.
Select the cluster head farthest from the sink node as the starting node of the outermost ring A_p
$A = \{A_p\}$; // A represents the set of starting nodes for each ring.
While $A_p \neq sink$
    Select the node A_{p-1} closest to the sink in the cluster head node adjacent to A_p
    $A = A \cup \{A_{p-1}$;
    $A_p = A_{p-1}$;
End while
Step 2: Create a ring route
For the initial node $A_t (1 \leq t \leq p)$ of each ring
$C = A_t$;
While $C.next \neq A_t$(No closed rings)
    Choose B that meets the criteria. (B satisfies the right-hand rule, B is the cluster head adjacent to C, and the distance from B to sink is the closest to the distance from C to sink.)
    $C.next = B$;
    $C = B$;
End while
End for
Step 3: Node routing from the sink node 1 hop range
For the node is within 1 hop from the sink node
    Send data directly to the sink node
End for
Step 4: Data collection of nodes (except for points within 1 hop from the sink node)
Sink generates a sparse binary matrix Φ, which is stored locally by each sensor node in the network
For each row i of Φ do
    For each sensor node $s_j (1 \leq j \leq n)$,
        If $\phi_{ij} \neq 0$
            The sensor node s_j sends its reading x_j to the cluster head node of the cluster by the shortest path algorithm
        End if
    End for
    For each cluster head node s_j
        If $\phi_{ij} \neq 0$
            $Sum_j =$ The sum of the data received by the cluster head node;
            $Sum_j+ =$ Cluster head node data x_j;
        Else
            $Sum_j =$ The sum of the data received by the cluster head node;
        End if
    End for
    For each node A_t in A
        Transfer data along the ring to the next cluster head node and calculate their sum until the return node is returned, and the resulting sum is Sum_t
    End for
    For (k=p;k>0;k++)
        A_k sends the data on the ring and Sum_t to A_{k-1}
        $SUM_i+ = Sum_t$
    End for
End for
Step 5: Data Recovery
The sink node receives m measured values, and the sink performs recovery of compressed data.

TABLE 2. Symbol Description.

Symbol	Description
R	Network radius
θ_k	Center angle of the sector
l	The distance from the node to the sink node
r	Communication radius (node emission radius)
ρ	Node density
n	Total number of nodes in the network
m	Measuring the number of rows in the matrix
σ	The correlation coefficient between n and m, $m = \sigma n$, σ is usually 0.05
τ	Packet size
λ	Event generation rate
d_l	The average amount of data of the node (the distance from the sink is l)
S_l	Area area (the distance from the sink is l)
D_l	Number of packets (the distance from the sink is l)
N_l	Number of nodes (the distance from the sink is l)
Q_l	Node data volume (the distance from the sink is l)
E_l	Node energy consumption (the distance from the sink is l)
k	Number of nodes in the cluster
d	The number of each column 1 of a sparse binary matrix
b	the value that makes $l+br$ just less than R.
l_m	the distance between this node and D_m

the cluster can communicate directly with the cluster head. The nodes in the network are homogeneous and static.

Table 2 shows the specific meanings of the symbols appearing in the following analysis.

A. THE AMOUNT OF DATA ANALYSIS OF DIFFERENT SCHEME

1) THE AMOUNT OF DATA BY NON-CS SCHEME

Theorem 1: In a network with radius R, the communication radius is r and the event rate is λ. The distance between node n_x and sink is l, and $l = hr + x$, (h is 0, 1, 2, 3...) The data volume of each data packet is τ, and the data is collected by non-CS scheme. The amount of data the node bears is

$$d_l = ((b+1) + \frac{(b(1+b)r}{2l})\lambda\tau \tag{9}$$

where b is the value that makes $l + br$ just less than R.

Proof: On the one hand, the nodes in the area where the node n_x is located itself will generate data, and on the other hand, the nodes in the area forward the data from the area at $l+r, l+2r, \ldots l+br$.

The amount of data generated by the area where the n_x node is located is $Q_l = D_l\tau = d_x\theta_k l\rho\lambda\tau$.

The amount of data forwarded by this area is $(\theta_k\ (L+r)\ \rho\ d_x + \theta_k\ (L+2r)\ \rho d_x + \ldots + \theta_k\ (L+br)\ \rho d_x)\lambda\tau$.

So the total amount of data that this area bears is $(\theta_k\ (L+r)\ \rho d_x + \theta_k\ (L+2r)\ \rho d_x + \ldots + \theta_k\ (L+br)\ \rho d_x + d_x\theta_k l\rho)\lambda\tau$.

Then the average amount of data each node bears is $((b+1) + \frac{(b(1+b)r}{2l})\lambda\tau$. □

2) THE AMOUNT OF DATA BY CS SCHEME

*Theorem 2:*In a network with radius R, the communication radius is r, The distance between node n_x and sink is l, and $l = hr + x$ The data volume of each data packet is τ, and the data is collected by CS scheme. The amount of data received by the node is

$$(1+\frac{r}{l})m\tau \tag{10}$$

Proof: In CS scheme, the node in the area where the n_x node is located receives data from the node at $l+r$ (distance from the sink node). The number of packets sent by each node is m. So the number of nodes, the number of packets, and the amount of data at $l+r$ are as follows:

$$N_{l+r} = \theta_k(l+r)\rho d_x \tag{11}$$

$$D_{l+r} = N_{l+r}m = \theta_k(l+r)\rho d_x m \tag{12}$$

$$Q_{l+r} = D_{l+r}\tau = \theta_k(l+r)\rho d_x m\tau \tag{13}$$

The number of nodes in the area where the n_x node is located is

$$N_l = \theta_k l\rho d_x \tag{14}$$

Therefore, the amount of data received by each node in the area where the n_x node is located is:

$$d_l = \frac{Q_{l+r}}{N_l} = \frac{\theta_k(l+r)\rho d_x m\tau}{\theta_k l\rho d_x} = \left(1+\frac{r}{l}\right)m\tau \tag{15}$$

In CS scheme, the number of packets sent by each node is m, and $m = \sigma n$, σ is usually between 0.05 and 0.25 [63], so the amount of data sent by each node is $0.05n\tau$. □

3) THE AMOUNT OF DATA BY DCSS SCHEME

Theorem 2: In a network with radius R, the communication radius is r, a node n_x within a hop range of sink, the distance between this node and $D_1, D_2 \ldots D_m$ is $l_1, l_2, l_3 \ldots l_m$, and $l = hr + x$, the probability that the node sends data is λ, and the data amount of each packet is τ. In the DCSS scheme, the amount of data that the node within a hop range of sink is

$$d = \left(1+\frac{r}{l_1}\right)\lambda\tau + \left(1+\frac{r}{l_2}\right)\lambda\tau + \ldots + \left(1+\frac{r}{l_m}\right)\lambda\tau + \frac{m\tau}{k} + d\tau \tag{16}$$

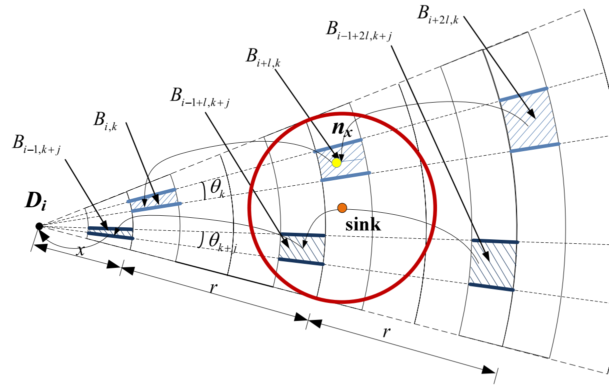

FIGURE 7. Data collection diagram.

where k is the number of nodes in the range of one hop from the sink node, $k = \pi r^2\rho$.

Proof: Nodes within one hop from the sink node do not only forward data from $D_1, D_2 \ldots D_m$, but $D_1, D_2 \ldots D_m$ also pass through these nodes when collecting data. Therefore, the total number of bearers of the node is greater than or equal to m. □

Figure 7 is a schematic diagram of data collection. For D_1, if the distance between a node n_x and D_1 within a range of one sink is l, If there is data transmission at $l+r$, Data transmission probability is λ, Then n_x needs to forward data from $l+r$, the amount of data received is

$$Q_{l+r} = D_{l+r}\tau = \theta_k(l+r)\rho d_x\lambda\tau \tag{17}$$

So the amount of data received by each node in the area where the n_x node is located is

$$d_l = \frac{Q_{l+r}}{N_l} = \frac{\theta_k(l+r)\rho d_x\lambda\tau}{\theta_k l\rho d_x} = \left(1+\frac{r}{l_1}\right)\lambda\tau \tag{18}$$

Similarly, the amount of data received by each node when $D_2 \ldots D_m$ ollects data can be obtained. Therefore, when $D_1, D_2 \ldots D_m$ collects data, the amount of data received by each node in the area where the n_x node is located is:

$$\left(1+\frac{r}{l_1}\right)\lambda\tau + \left(1+\frac{r}{l_2}\right)\lambda\tau + \ldots + \left(1+\frac{r}{l_m}\right)\lambda\tau \tag{19}$$

When $D_1, D_2 \ldots D_m$ sends a data packet to the sink node, the total number of packets to be taken from the point within one hop of the sink is m. Then the average amount of each node is m/k, the amount of data generated by the node itself in the near sink area is kd, so in summary, the average amount of data that each node needs to bear is:

$$\left(1+\frac{r}{l_1}\right)\lambda\tau + \left(1+\frac{r}{l_2}\right)\lambda\tau + \ldots + \left(1+\frac{r}{l_m}\right)\lambda\tau + \frac{m\tau}{k} + d\tau \tag{20}$$

The result is proved.

4) THE AMOUNT OF DATA BY CS-CARDG SCHEME

a. The amount of data per node within one hop of the sink node

TABLE 3. Contrast the amount of data between DCSS scheme and CS-CARDG scheme.

	DCSS	CS-CARDG
The amount of node data within one hop from the sink distance	$\left(1+\frac{r}{l_1}\right)\lambda\tau+\cdots+\left(1+\frac{r}{l_m}\right)\lambda\tau + \frac{m\tau}{k}+d\tau$	$\frac{m\tau}{k}+\tau$

Since only one node in each round communicates directly with the sink node, and the number of packets forwarded by it is m, for a point within one hop from the sink node (a node with a sink node as the cluster head), each node needs to send its own 1 packet., a total of m data packets from other nodes need to be forwarded, so a total of $m+k$ data packets are required. The average amount of data d_x assumed by each node is:

$$d_x = (\frac{m}{k}+1)\tau \tag{21}$$

where $k = \pi r^2 \rho$

b. The amount of data sent by each ring starting node

The starting node of each ring shall send data to the next node on the ring, and also send data to the sink node from the outside to the inside. So the number of packets it wants to send is $2m$, so the amount of data it sends is $2m\tau$.

c. The amount of data per node on other nodes on the ring

Since the nodes on the ring are all cluster head nodes, the number of data packets that the cluster head node needs to send is m, so the amount of data sent by it is $m\tau$.

Since each node in the cluster directly sends data to the cluster head node, the number of data packets sent by each node is d, and the number of nodes in one cluster is k. Therefore, the number of packets received by the node on the ring (except the starting node) is kd, and the amount of data is $kd\tau$.

B. CONTRAST THE AMOUNT OF DATA OF DIFFERENT SCHEME

As can be seen from Table 3, the amount of node data within the one-hop range of the DCSS scheme and the sink distance is largely larger than the CS-CARDG scheme. This is because the $D_1, D_2 \ldots D_m$ nodes in the DCSS scheme, when collecting data, largely pass the data within the range of the sink one hop, so it will result in a packet size greater than m/k at these points. The CS-CARDG scheme can guarantee that the average packet of these nodes is $\frac{m}{k}+1$. This shows the advantage of the CS-CARDG scheme in terms of load capacity.

Then the amount of data reduced compared to the two is

$$\left(1+\frac{r}{l_1}\right)\lambda\tau+\ldots+\left(1+\frac{r}{l_m}\right)\lambda\tau-\tau+d\tau \tag{22}$$

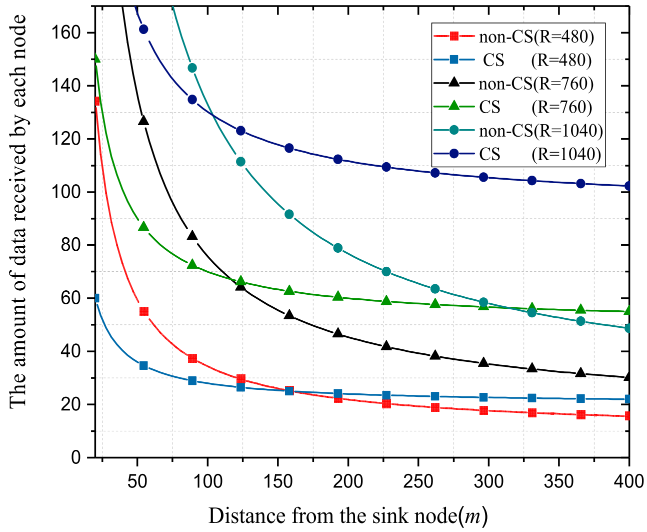

FIGURE 8. Comparison of non-CS and CS data volume.

Then the percentage reduction is

$$\frac{\left(1+\frac{r}{l_1}\right)\lambda\tau+\ldots+\left(1+\frac{r}{l_m}\right)\lambda\tau-\tau+d\tau}{\left(1+\frac{r}{l_1}\right)\lambda\tau+\ldots+\left(1+\frac{r}{l_m}\right)\lambda\tau+\frac{m\tau}{k}+d\tau} \tag{23}$$

Taking the extreme value can result in an average reduction of at least 20% of the amount of data each node has to bear.

As shown in Fig.8, the closer the distance to the sink node, the larger the amount of data, and the faster the amount of data increases. It can be clearly seen from the Fig.8 that the CS scheme has a smaller data load than non-CS scheme when the nodes are closer to the sink node. However, when the distance is far to a certain extent, the CS scheme has a larger data carrying capacity than non-CS. At the same time, it can be seen from the figure that for a network with even nodes distributed, the radius of the whole network is larger, and the number of nodes is larger, so the data carrying capacity of each node is larger.

Fig.9 shows the relationship between the data capacity of the node and the total number of nodes in the network within one hop distance from the sink. The greater the number of network nodes, the greater the amount of data carried by these three schemes. It can be clearly seen that the data carrying capacity of the non-CS schemes is the largest, followed by the CS schemes. It can also be seen from Table 3 that the data carrying capacity of the DCSS schemes is also larger than the CS-CARDG schemes. Therefore, the data carrying capacity of the CS-CARDG schemes is the smallest. This shows that the CS-CARDG schemes is very helpful to alleviate the node load in the range of sink one hop, which verifies the advantages of this schemes.

Fig. 10 shows the relationship between the data capacity of the node and the distance within the range of one sink from the sink. It can be seen that, for nodes near the sink area, the closer the distance between the node and the sink node, the larger the data carrying capacity. For the CS-CARDG

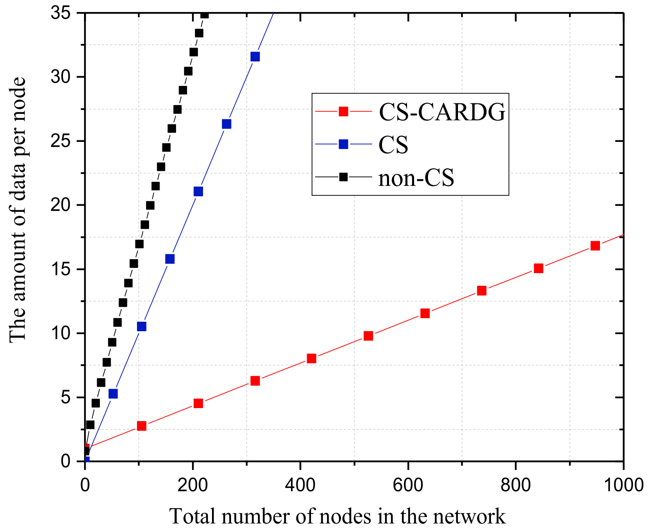

FIGURE 9. Comparison of cs-CARDG, CS, non-CS data volume.

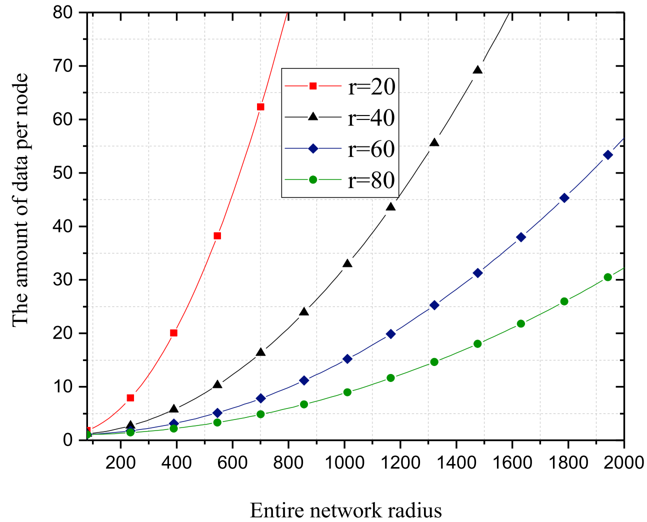

FIGURE 11. Network radius and data volume diagram.

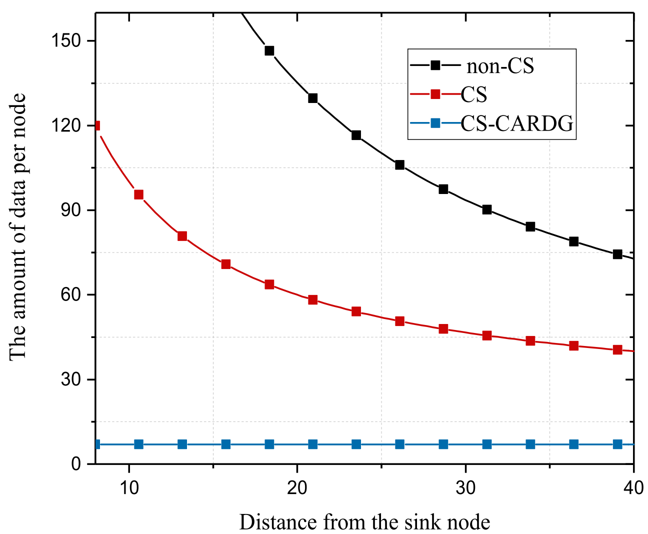

FIGURE 10. Comparison of cs-CARDG, CS, non-CS data volume.

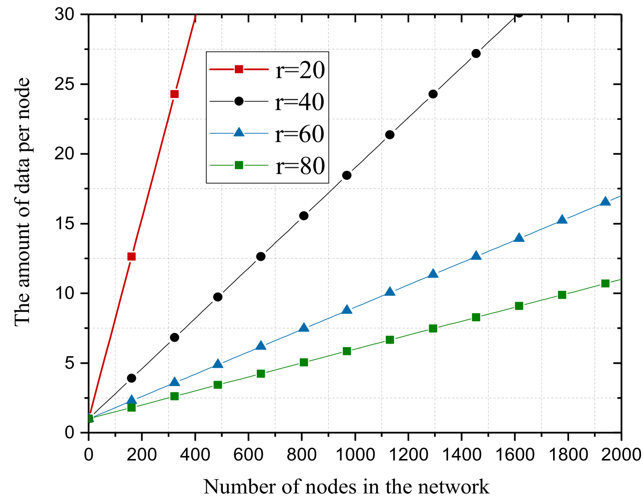

FIGURE 12. Relationship between number of network nodes and the amount of data.

scheme, the average data capacity of the near-sink area node is unchanged and the smallest.

C. CS-CARDG SCHEME DATA CARRYING CAPACITY ANALYSIS

Since the data carrying capacity of the node within the range of one sink is often the largest, we use the average data volume of the nodes in this range as the research object to conduct research and analysis.

As shown in Fig.11, the larger the radius of the network, the greater the data carrying capacity of the node within one hop of the sink. It can be seen that the larger the number of nodes in the network, the greater the amount of data carried by each node. The larger the communication radius of the network, the smaller the load capacity of each node. Because the data carrying capacity of each cluster is certain for a certain network, the more nodes in the cluster, the smaller the amount of data of each node.

As shown in Fig.12, it can be seen that the greater the number of nodes in the network, the greater the data carrying capacity of the points within the range of one sink of the sink. The larger the number of network nodes, the larger the value of m, so the greater the data load per node.

As shown in Fig.13, the greater the density of nodes in the network, the smaller the data capacity of each node. As the network density increases, the number of nodes in each cluster increases, and the data capacity of each node decreases.

As shown in Fig.14, the larger the communication radius of the network, the smaller the data capacity of each node. The greater the number of nodes in the network, the greater the data carrying capacity of each node.

As can be seen from Fig.15, in the case where the total number of nodes in the network is constant, the more the

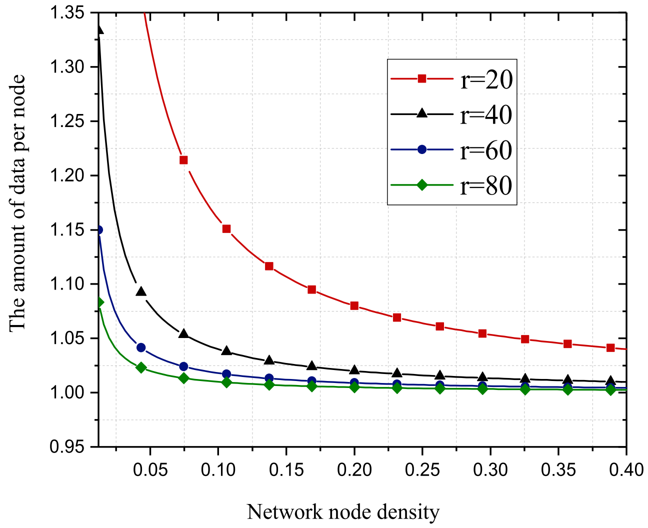

FIGURE 13. Network node density and data volume diagram.

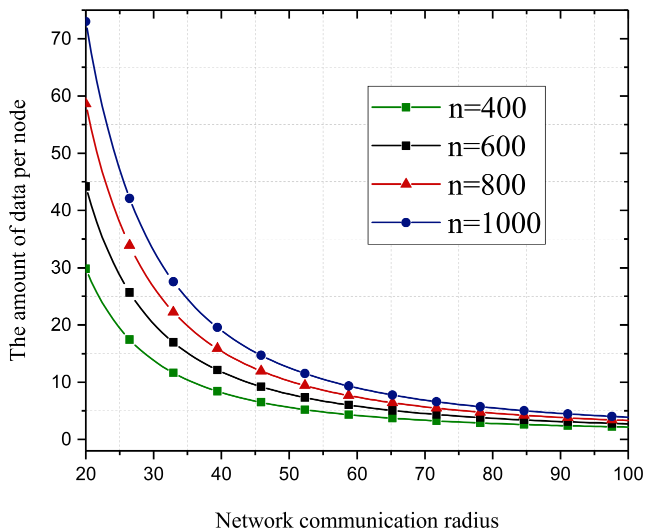

FIGURE 14. Network communication radius and data volume diagram.

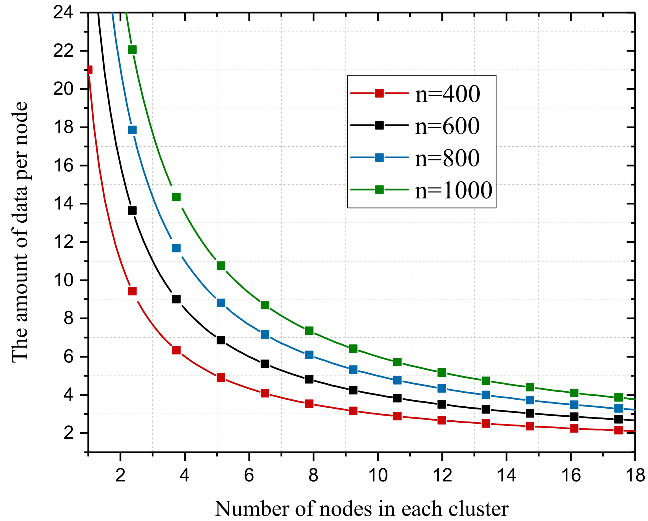

FIGURE 15. Relationship between the number of nodes in the cluster and the amount of data.

number of nodes in the cluster, the smaller the amount of data carried by each node.

D. DIFFERENT SCHEME ENERGY ANALYSIS

1) NON-CS SCHEME ENERGY CONSUMPTION OF PER NODE

Reasoning 1: It can be obtained by the Eq.2 and Eq.3, the communication radius is r, the distance between the node and the sink is l, $l = hr + x$, and the data is collected by the non-CS scheme, then the energy consumption of the single node is:

$$\begin{cases} E_l = (2d_l - \tau)E_{elec} + d_l \varepsilon_{fs} d^2 & if\ d < d_0 \\ E_l = (2d_l - \tau)E_{elec} + d_l \varepsilon_{amp} d^2 & if\ d \geq d_0 \end{cases} \tag{24}$$

where $d_l = \left((b+1) + \frac{(b(1+b)r}{2l}\right)\lambda\tau$, here d represents the Euclidean distance of the transmitter and receiver. Other symbols have the same meaning as Eq.2 and Eq.3.

2) CS SCHEME ENERGY CONSUMPTION OF PER NODE

Reasoning 2: The communication radius is r, the distance between the node and the sink is l, $l = hr + x$. When the data is collected by the CS scheme, the energy consumption of the single node is:

$$\begin{cases} E_l = (m + \dfrac{rm}{l} + m)\tau E_{elec} + m\tau\varepsilon_{fs} d^2 & if\ d < d_0 \\ E_l = \left(m + \dfrac{rm}{l} + m\right)\tau E_{elec} + m\tau\varepsilon_{amp} d^2 & if\ d > d_0 \end{cases} \tag{25}$$

3) DCSS SCHEME ENERGY CONSUMPTION OF PER NODE

Reasoning 3: The data is collected using the DCSS algorithm, and the energy consumption of each node within one hop from the sink node is:

$$\begin{cases} E_l = 2d_x E_{elec} + d_x \varepsilon_{fs} d^2 & if\ d < d_0 \\ E_l = 2d_x E_{elec} + d_x \varepsilon_{amp} d^2 & if\ d \geq d_0 \end{cases} \tag{26}$$

where, $d_x = \left(1 + \frac{r}{l_1}\right)\lambda\tau + \ldots + \left(1 + \frac{r}{l_m}\right)\lambda\tau + \frac{m\tau}{k} + d\tau$

Proof: According to Eq.16, the amount of data sent and received by each node within one hop of the sink node is $d_x = \left(1 + \frac{r}{l_1}\right)\lambda\tau + \ldots + \left(1 + \frac{r}{l_m}\right)\lambda\tau + \frac{m\tau}{k} + d\tau$, so the energy consumed by each node to send data is:

$$E_{Tx}(i) = \begin{cases} \left(E_{elec} + \epsilon_{fs} \times d^2\right) \times d_x, & d < d_0, \\ \left(E_{elec} + \epsilon_{mp} \times d^2\right) \times d_x, & d \geq d_0, \end{cases} \tag{27}$$

The energy consumed to receive the data is:

$$E_{Rx}(i) = E_{elec} \times d_x \tag{28}$$

where, $d_x = \left(1 + \frac{r}{l_1}\right)\lambda\tau + \ldots + \left(1 + \frac{r}{l_m}\right)\lambda\tau + \frac{m\tau}{k} + d\tau$

The total energy consumption is obtained by adding the energy consumed by the transmitted data to the energy consumed by the received data. □

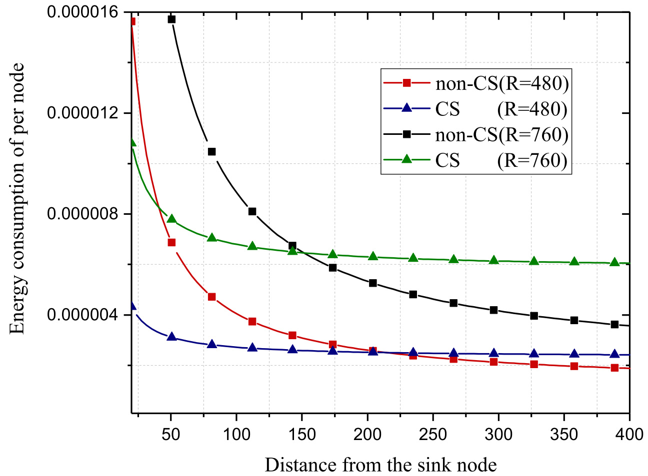

FIGURE 16. Comparison of energy consumption between non-CS and CS nodes.

4) CS-CARDG SCHEME ENERGY CONSUMPTION OF PER NODE

Reasoning 1: The data is collected using the CS-CARDG scheme, and the energy consumption of each node within one hop from the sink node is:

$$\begin{cases} E_l = \left(\dfrac{2m}{\pi r^2 \rho}+2\right)\tau E_{elec} + (\dfrac{m}{\pi r^2 \rho}+1)\tau \varepsilon_{fs} d^2 & if\ d < d_0 \\ E_l = (\dfrac{2m}{\pi r^2 \rho}+2)\tau E_{elec} + (\dfrac{m}{\pi r^2 \rho}+1)\tau \varepsilon_{amp} d^2 & if\ d > d_0 \end{cases} \tag{29}$$

Proof: It can be obtained from Eq.21 that the amount of data transmitted and received by each node within one hop of the sink node is $d_x = \frac{m\tau}{\pi r^2 \rho} + \tau$, so the energy consumed by each node to send data is:

$$E_{Tx}(i) = \begin{cases} \left(E_{elec} + \epsilon_{fs} \times d^2\right) \times (\dfrac{m\tau}{\pi r^2 \rho} + \tau), & d < d_0, \\ \left(E_{elec} + \epsilon_{mp} \times d^2\right) \times (\dfrac{m\tau}{\pi r^2 \rho} + \tau), & d \geq d_0, \end{cases} \tag{30}$$

The energy consumed to receive data is:

$$E_{Rx}(i) = E_{elec} \times (\frac{m\tau}{\pi r^2 \rho} + \tau) \tag{31}$$

The total energy consumption is obtained by adding the energy consumed by the transmitted data to the energy consumed by the received data. □

E. ENERGY CONSUMPTION ANALYSIS

As can be seen in Fig.16, the closer the distance to the sink, the greater the energy consumption of the node. The CS scheme has less energy consumption than the non-CS scheme in the range closer to the sink node, and the energy consumption of the non-CS scheme is smaller than the CS scheme after a certain distance.

In the following, only the energy consumption of CS-CARDG is analyzed, and the nodes within one hop of the sink are taken as the research object. As can be seen

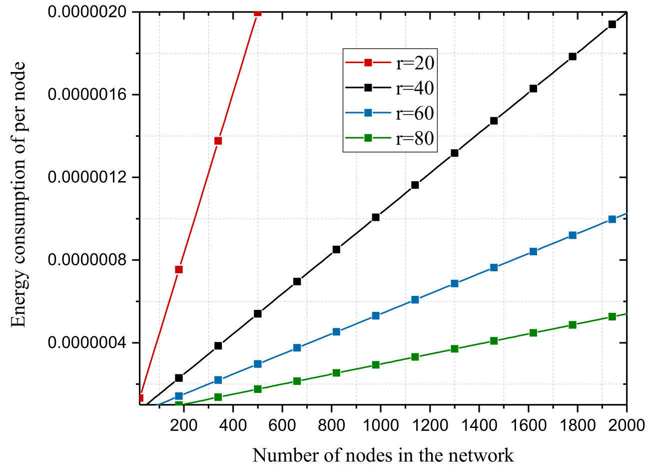

FIGURE 17. Relationship between number of network nodes and energy consumption.

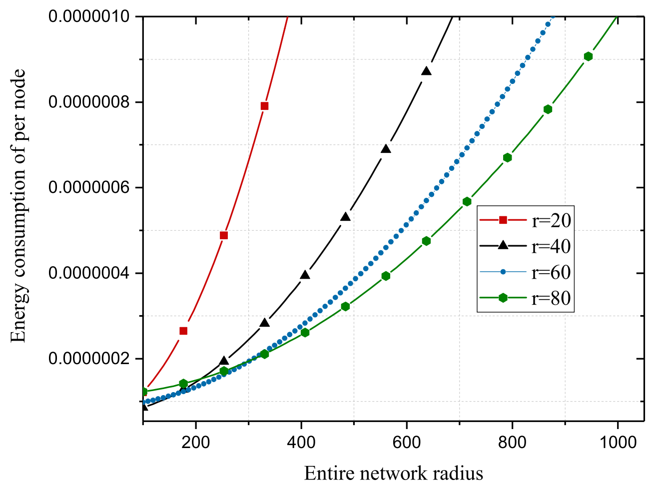

FIGURE 18. The relationship between network radius and energy consumption.

from Fig.17, the greater the number of nodes in the network, the greater the energy consumption of each node within one hop of the sink. The greater the communication radius of the network, the smaller the energy consumption of each node.It can be seen from Fig.18 that the larger the radius of the entire network, the greater the energy consumption of each node within the range of one sink from the sink. As shown in Fig.19, the more nodes in each cluster, the less energy the node consumes. The more total the number of network nodes, the greater the energy consumption of the network. As shown in Fig.20, the greater the communication radius of the network, the smaller the energy consumption of each node.

F. CS-CARDG SCHEME ENERGY EFFICIENCY

The CS-CARDG scheme is rotated and re-clustered by nodes, so that the energy consumption of other nodes in the cluster is approximately equal except for the cluster with the sink as the cluster head. Then the average energy consumption of each

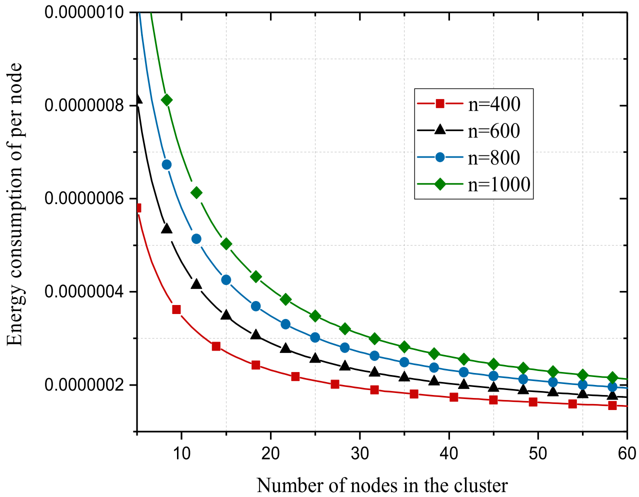

FIGURE 19. Relationship between number of nodes in the cluster and energy consumption.

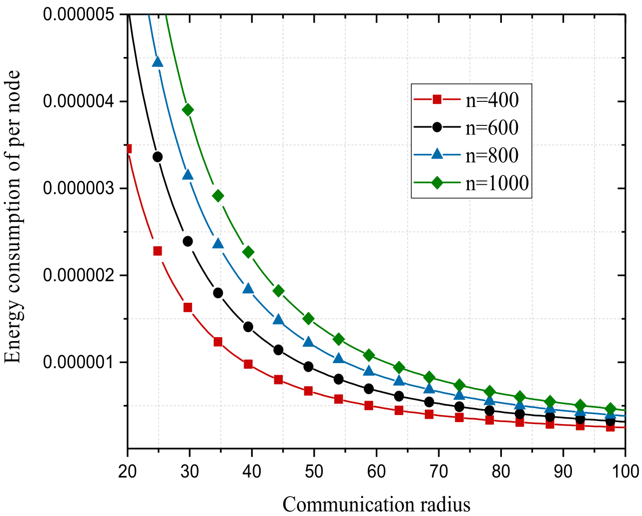

FIGURE 20. Relationship between communication radius and energy consumption.

cluster after the c round is:

$$\sum_{f=1}^{c}\sum_{z=1}^{k}\frac{e_{fz}}{k} \tag{32}$$

where e_{fz} represents the energy consumed by the zth node in the f-th round, and k represents the number of nodes in the cluster.

We assume that D_i is the total energy consumption of a node within each cluster, then:

$$\sum_{f=1}^{c}\sum_{z=1}^{k}\frac{e_{fz}}{k}=\frac{cD_i}{k} \tag{33}$$

$$\sum_{f=1}^{c}\sum_{z=1}^{k}\frac{e_{fz}}{ck}=\frac{D_i}{k} \tag{34}$$

In summary, we can replace the average energy consumption of a single node in different rounds with the average energy consumption of nodes in each cluster in a single round.

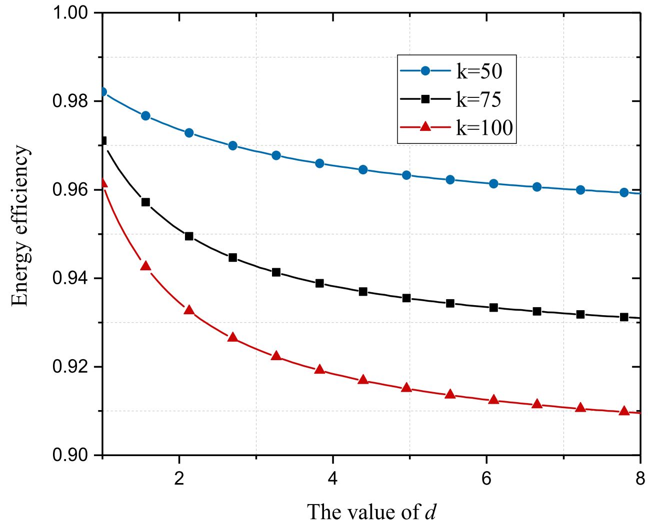

FIGURE 21. CS-CARDG strategy energy efficiency.

Theorem 3: In the CS-CARDG scheme, the energy utilization of the network is:

$$\eta = 1-\frac{\left(\bar{E_1}-\bar{E_2}\right)k}{\bar{E_1}n} \tag{35}$$

where, $\bar{E_1}=\frac{u(kd-d+m)\tau+E_{elec}kd\tau}{k}$, $\bar{E_2}=\frac{uk\tau+E_{elec}m\tau}{k}$, k represents the number of nodes in the cluster, and n represents the total number of nodes in the network. When $d_i < d_0$, $u=E_{elec}+\epsilon_{fs}\times d_i^2$, when $d_i \geq d_0$, $u=E_{elec}+\epsilon_{mp}\times d_i^4$.

Proof: The number of packets received by each node within the range of one sink is m, and the number of packets sent is k, so the average energy consumption of each node in this range is:

$$\bar{E_2}=\frac{uk\tau+E_{elec}m\tau}{k} \tag{36}$$

For other clusters, the cluster head node needs to receive kd data packets and send m data packets, while the nodes in the cluster need to send d data packets, so the average energy consumption of each node is:

$$\bar{E_1}=\frac{u\left(kd-d+m\right)\tau+E_{elec}kd\tau}{k} \tag{37}$$

Since $m-d>0$, and the value of kd is often greater than m, the percentage of remaining energy of the node is:

$$\frac{\bar{E_1}-\bar{E_2}}{\bar{E_1}} \tag{38}$$

Then the remaining energy of the entire network is:

$$\frac{\left(\bar{E_1}-\bar{E_2}\right)kE_{Init}}{\bar{E_1}nE_{init}} \tag{39}$$

Among them, E_{Init} represents the initial energy of each node. So the energy efficiency is $\eta=1-\frac{(\bar{E_1}-\bar{E_2})k}{\bar{E_1}n}$. □

We experimented on a network with a network radius of 480 m and got the following results. As shown in Fig.21, k represents the number of nodes in each cluster. As k increases, the energy utilization of the network decreases. d represents the number of 1 in each column of the observation matrix. It can be seen that as d decreases, the energy utilization of the network increases, and as d increases, the energy utilization rate begins to a fixed value. As can be seen from the Fig.21, the energy utilization rate of the network is often above 90%. A significant improvement has been achieved compared to the 10% energy utilization rate proposed in [49].

VI. CONCLUSION

Compressive sensing technology can significantly reduce the amount of data in the near-sink area, which makes the network life extended, but does not fundamentally reduce the amount of data in this area. In this paper, we propose Compressive Sensing based Clustering Joint Annular Routing Data Gathering (CS-CARDG) scheme. In this scheme, the network first forms a cluster. Each node in the cluster sends data packets to the cluster head. Each cluster forms m-dimensional data according to the requirements of the compressed sensing technology to ensure that the data can be recovered. When the cluster head node routes the m-dimensional data to the sink, the CS-CARDG scheme adopts a two-stage routing scheme with the same loop routing and shortest path which are different from the previous schemes. Through the performance analysis, we can see that the advantage of this scheme is that the amount of data in the near-sink area is reduced to a minimum, and the energy utilization rate is significantly improved.

REFERENCES

[1] H. Teng, Y. Liu, A. Liu, N. N. Xiong, Z. Cai, T. Wang, and X. Liu, "A novel code data dissemination scheme for Internet of Things through mobile vehicle of smart cities," *Future Gener. Comput. Syst.*, vol. 94, pp. 351–367, May 2019.
[2] T. Shu, W. Liu, T. Wang, Q. Deng, M. Zhao, N. N. Xiong, X. Li, and A. Liu, "Broadcast based code dissemination scheme for duty cycle based wireless sensor networks," *IEEE Access*, vol. 7, pp. 105258–105286, 2019.
[3] J. Li, W. Liu, T. Wang, H. Song, X. Li, F. Liu, and A. Liu, "Battery-friendly based relay selection scheme to prolong lifetime for sensor nodes in Internet of Things," *IEEE Access*, vol. 7, pp. 33180–33201, 2019.
[4] Y. Liu, A. Liu, T. Wang, X. Liu, and N. N. Xiong, "An intelligent incentive mechanism for coverage of data collection in cognitive Internet of Things," *Future Gener. Comput. Syst.*, vol. 100, pp. 701–714, Nov. 2019.
[5] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the suitability of fog computing in the context of Internet of Things," *IEEE Trans. Cloud Comput.*, vol. 6, no. 1, pp. 46–59, Jan./Mar. 2018.
[6] A. Aloisio, F. Ameli, A. Anastasio, P. Branchini, F. Di Capua, R. Giordano, V. Izzo, and G. Tortone, "uSOP: A microprocessor-based service-oriented platform for control and monitoring," *IEEE Trans. Nucl. Sci.*, vol. 64, no. 6, pp. 1185–1190, Jun. 2017.
[7] Y. Liu, A. Liu, X. Liu, and X. Huang, "A statistical approach to participant selection in location-based social networks for offline event marketing," *Inf. Sci.*, vol. 480, pp. 90–108, Apr. 2019.
[8] H. Teng, W. Liu, T. Wang, A. Liu, X. Liu, and S. Zhang, "A cost-efficient greedy code dissemination scheme through vehicle to sensing devices (V2SD) communication in smart city," *IEEE Access*, vol. 7, pp. 16675–16694, 2019.
[9] Y. Ren, W. Liu, T. Wang, X. Li, N. N. Xiong, and A. Liu, "A collaboration platform for effective task and data reporter selection in crowdsourcing network," *IEEE Access*, vol. 7, pp. 19238–19257, 2019.
[10] M. Huang, W. Liu, T. Wang, H. Song, X. Li, and A. Liu, "A queuing delay utilization scheme for on-path service aggregation in services oriented computing networks," *IEEE Access*, vol. 7, pp. 23816–23833, 2019.
[11] T. Wang, H. Luo, W. Jia, A. Liu, and M. Xie, "MTES: An intelligent trust evaluation scheme in sensor-cloud enabled industrial Internet of Things," *IEEE Trans. Ind. Inform.*, to be published. doi: 10.1109/TII.2019.2930286.
[12] Q. Wang, W. Liu, T. Wang, M. Zhao, X. Li, M. Xie, M. Ma, G. Zhang, and A. Liu, "Reducing delay and maximizing lifetime for wireless sensor networks with dynamic traffic patterns," *IEEE Access*, vol. 7, pp. 70212–70236, 2019.
[13] Q. Li, A. Liu, T. Wang, M. Xie, and N. Xiong, "Pipeline slot based fast rerouting scheme for delay optimization in duty cycle based M2M communications," in *Peer-to-Peer Networking Applications*. New York, NY, USA: Springer, 2019, pp. 1–32. doi: 10.1007/s12083-019-00753-z.
[14] X. Xiang, W. Liu, T. Wang, M. Xie, X. Li, H. Song, A. Liu, and G. Zhang, "Delay and energy-efficient data collection scheme-based matrix filling theory for dynamic traffic IoT," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, p. 168, 2019. doi: 10.1186/s13638-019-1490-5.
[15] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the edge: A scalable IoT architecture based on transparent computing," *IEEE Netw.*, vol. 31, no. 5, pp. 96–105, Aug. 2017.
[16] X. Peng, J. Ren, L. She, D. Zhang, J. Li, and Y. Zhang, "BOAT: A block-streaming app execution scheme for lightweight IoT devices," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1816–1829, Jun. 2018.
[17] L. Hu, A. Liu, M. Xie, and T. Wang, "UAVs joint vehicles as data mules for fast codes dissemination for edge networking in smart city," in *Peer-to-Peer Networking and Applications*. New York, NY, USA: Springer. doi: 10.1007/s12083-019-00752-0.
[18] X. Sun, W. Liu, T. Wang, Q. Deng, A. Liu, N. Xiong, and S. Zhang, "Two-hop neighborhood information joint double broadcast radius for effective code dissemination in WSNs," *IEEE Access*, vol. 7, pp. 88547–88569, 2019.
[19] J. Xu, K. Ota, M. Dong, A. Liu, and Q. Li, "SIoTFog: Byzantine-resilient IoT fog networking," *Frontiers Inf. Technol. Electron. Eng.*, vol. 19, no. 12, pp. 1546–1557, 2018.
[20] T. Wang, J. Zhou, A. Liu, M. Z. A. Bhuiyan, G. Wang, and W. Jia, "Fog-based computing and storage offloading for data synchronization in IoT," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4272–4282, Jun. 2019.
[21] J. Tan, W. Liu, M. Xie, H. Song, A. Liu, M. Zhao, and G. Zhang, "A low redundancy data collection scheme to maximize lifetime using matrix completion technique," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 5, Dec. 2019. doi: 10.1186/s13638-018-1313-0.
[22] W. Zhang, W. Liu, T. Wang, A. Liu, Z. Zeng, H. Song, and S. Zhang, "Adaption resizing communication buffer to maximize lifetime and reduce delay for WVSNs," *IEEE Access*, vol. 7, pp. 48266–48287, 2019.
[23] Y. Liu, A. Liu, N. N. Xiong, T. Wang, and W. Gui, "Content propagation for content-centric networking systems from location-based social networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published. doi: 10.1109/TSMC.2019.2898982.
[24] J. Tan, W. Liu, T. Wang, S. Zhang, A. Liu, M. Xie, M. Ma, and M. Zhao, "An efficient information maximization based adaptive congestion control scheme in wireless sensor network," *IEEE Access*, vol. 7, pp. 64878–64896, 2019.
[25] M. Huang, W. Liu, T. Wang, Q. Deng, A. Liu, M. Xie, M. Ma, and G. Zhang, "A game-based economic model for price decision making in cyber-physical-social systems," *IEEE Access*, to be published. doi: 10.1109/ACCESS.2019.2934515.
[26] J. Ren, Y. Zhang, N. Zhang, D. Zhang, and X. Shen, "Dynamic channel access to improve energy efficiency in cognitive radio sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 5, pp. 3143–3156, May 2016.
[27] T. Wang, L. Qiu, G. Xu, A. K. Sangaiah, and A. Liu, "Energy-efficient and trustworthy data collection protocol based on mobile fog computing in Internet of Things," *IEEE Trans. Ind. Inform.*, to be Published. doi: 10.1109/TII.2019.2920277.
[28] X. Deng, G. Li, M. Dong, and K. Ota, "Finding overlapping communities based on Markov chain and link clustering," *Peer-Peer Netw. Appl.*, vol. 10, no. 2, pp. 411–420, 2017.
[29] L. Yin, J. Gui, Z. Zeng, "Improving energy efficiency of multimedia content dissemination by adaptive clustering and D2D multicast," *Mobile Inf. Syst.*, vol. 2019, Mar. 2019, Art. no. 5298508. doi: 10.1155/2019/5298508.
[30] X. Deng, T. He, L. He, J. Gui, and Q. Peng, "Performance analysis for IEEE 802.11s wireless mesh network in smart grid," *Wireless Pers. Commun.*, vol. 96, no. 1, pp. 1537–1555, Sep. 2017.
[31] F. Wang, W. Liu, T. Wang, M. Zhao, M. Xie, H. Song, X. Li, and A. Liu, "To reduce delay, energy consumption and collision through optimization duty-cycle and size of forwarding node set in WSNs," *IEEE Access*, vol. 7, pp. 55983–56015, 2019.
[32] G. Zhang, T. Wang, G. Wang, A. Liu, W. Jia, "Detection of hidden data attacks combined fog computing and trust evaluation method in sensor-cloud system," *Concurrency Comput., Pract. Exper.*, to be published. doi: 10.1002/cpe.5109.

[33] Y. Liu, A. Liu, X. Liu, and M. Ma, "A trust-based active detection for cyber-physical security in industrial environments," *IEEE Trans. Ind. Inform.*, to be published. doi: 10.1109/TII.2019.2931394.

[34] X. Liu, M. Zhao, A. Liu, and K. K. L. Wong, "Adjusting forwarder nodes and duty cycle using packet aggregation routing for body sensor networks," *Inf. Fusion*, vol. 53, pp. 183–195, Jan. 2020.

[35] S. Zhang, X. Mao, K. Choo, T. Peng, and G. Wang, "A trajectory privacy-preserving scheme based on a dual-K mechanism for continuous location-based services," *Inf. Sci.*, to be published. doi: 10.1016/j.ins.2019.05.054.

[36] X. Xiang, W. Liu, N. N. Xiong, H. Song, A. Liu, and T. Wang, "Duty cycle adaptive adjustment based device to device (D2D) communication scheme for WSNs," *IEEE Access*, vol. 6, pp. 76339–76373, 2018.

[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.*, to be published. doi: 10.1109/TETC.2018.2844282.

[38] P. Guo, X. Liu, S. Tang, and J. Cao, "Enabling coverage-preserving scheduling in wireless sensor networks for structural health monitoring," *IEEE Trans. Comput.*, vol. 65, no. 8, pp. 2456–2469, Aug. 2017.

[39] Y. Yin, W. Zhang, Y. Xu, H. Zhang, Z. Mai, and L. Yu, "QoS prediction for mobile edge service recommendation with auto-encoder," *IEEE Access*, vol. 7, pp. 62312–62324, 2019.

[40] Y. Yin, L. Chen, Y. Xu, J. Wan, H. Zhang, and Z. Mai, "QoS prediction for service recommendation with deep feature learning in edge computing environment," in *Mobile Networks Applications*. New York, NY, USA: Springer, 2019. doi: 10.1007/s11036-019-01241-7.

[41] P. Guo, X. Liu, J. Cao, and S. Tang, "Lossless in-network processing and its routing design in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6528–6542, Oct. 2017.

[42] 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.

[43] 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.

[44] C. Lv, Q. Wang, W. Yan, and J. Li, "A sparsity feedback-based data gathering algorithm for wireless sensor networks," *Comput. Netw.*, vol. 141, pp. 145–156, Aug. 2018.

[45] Z. Li, Y. Liu, M. Ma, A. Liu, X. Zhang, and G. Luo, "MSDG: A novel green data gathering scheme for wireless sensor networks," *Comput. Netw.*, vol. 142, pp. 223–239, Sep. 2018.

[46] C. Zhang, Y. Lin, L. Zhu, A. Liu, Z. Zhang, and F. Huang, "CNN-VWII: An efficient approach for large-scale video retrieval by image queries," *Pattern Recognit. Lett.*, vol. 123, pp. 82–88, May 2019.

[47] W. Liu, P. Zhuang, H. Liang, J. Peng, and Z. Huang, "Distributed economic dispatch in microgrids based on cooperative reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2192–2203, Jun. 2018.

[48] Y. Yin, L. Chen, Y. Xu, and J. Wan, "Location-aware service recommendation with enhanced probabilistic matrix factorization," *IEEE Access*, vol. 6, pp. 62815–62825, 2018.

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

[50] H. Gao, H. Miao, L. Liu, J. Kai, and K. Zhao, "Automated quantitative verification for service-based system design: A visualization transform tool perspective," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 28, no. 10, pp. 1369–1397, 2018.

[51] H. Gao, K. Zhang, J. Yang, F. Wu, and H. Liu, "Applying improved particle swarm optimization for dynamic service composition focusing on quality of service evaluations under hybrid networks," *Int. J. Distrib. Sensor Netw.*, vol. 14, no. 2, pp. 1–14, 2018.

[52] H. Gao, W. Huang, X. Yang, Y. Duan, and Y. Yin, "Toward service selection for workflow reconfiguration: An interface-based computing solution," *Future Gener. Comput. Syst.*, vol. 87, pp. 298–311, Oct. 2018.

[53] X. Wang, L. T. Yang, L. Kuang, X. Liu, Q. Zhang, and M. J. Deen, "A tensor-based big-data-driven routing recommendation approach for heterogeneous networks," *IEEE Netw.*, vol. 33, no. 1, pp. 64–69, Jan./Feb. 2019.

[54] E. J. Candés, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.

[55] Q. Deng, H. Zeng, J. Zhang, S. Tian, J. Cao, Z. Li, and A. Liu, "Compressed sensing for image reconstruction via back-off and rectification of greedy algorithm," *Signal Process.*, vol. 157, pp. 280–287, Apr. 2019.

[56] X. Wang, L. T. Wang, X. Xie, J. Jin, and M. J. Deen, "A cloud-edge computing framework for cyber-physical-social services," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 80–85, Nov. 2017.

[57] S. Li and H. Qi, "Distributed data aggregation for sparse recovery in wireless sensor networks," in *Proc. IEEE Int. Conf. Distrib. Comput. Sensor Syst.*, Mar. 2013, pp. 62–69.

[58] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

[59] A. E. A. A. Abdulla, H. Nishiyama, and N. Kato, "Extending the lifetime of wireless sensor networks: A hybrid routing algorithm," *Comput. Commun.*, vol. 35, no. 9, pp. 1056–1063, 2012.

[60] P. Bonnet, J. Gehrke, and P. Seshadri, "Towards sensor database systems," in *Proc. Int. Conf. Mobile Data Manage.* Berlin, Germany: Springer, 2001, pp. 3–14.

[61] A.-F. Liu, P.-H. Zhang, and Z.-G. Chen, "Theoretical analysis of the lifetime and energy hole in cluster based wireless sensor networks," *J. Parallel Distrib. Comput.*, vol. 71, no. 10, pp. 1327–1355, 2011.

[62] F. Kuhn, R. Wattenhofer, and A. Zollinger, "An algorithmic approach to geographic routing in ad hoc and sensor networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 51–62, Feb. 2008.

[63] C. Luo, J. Sun, F. Wu, and C. W. Chen, "Compressive data gathering for large-scale wireless sensor networks," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2009, pp. 145–156.

YICONG YUAN is currently pursuing the degree with the School of Computer Science and Engineering, Central South University, China. Her research interest includes wireless sensor networks.

WEI LIU received the Ph.D. degree in computer application technology from Central South University, China, in 2014. He is currently an Associate Professor and a Senior Engineer with the School of Informatics, Hunan University of Chinese Medicine, China. His research interests include software engineering, data mining, and medical informatics. He has published more than 20 papers in related fields.

TIAN WANG received the B.Sc. and M.Sc. degrees in computer science from Central South University, in 2004 and 2007, respectively, and the Ph.D. degree from the City University of Hong Kong, in 2011. He is currently an Associate Professor with the National Huaqiao University of China. His research interests include wireless sensor networks, social networks, and mobile computing.

QINGYONG DENG received the master's degree in signal and information processing from Xiangtan University, China, in 2009, and the Ph.D. degree from the Beijing University of Posts and Telecommunications (BUPT), China, in 2019. He is currently a Lecturer with the College of Information Engineering, Xiangtan University, China. He has published more than ten referred journal papers in these fields. His current research interests include the Internet of Things, compressed sensing, wireless networks, and 5G.

ANFENG LIU received the M.Sc. and Ph.D. degrees in computer science from Central South University, China, in 2002 and 2005, respectively. He is currently a Professor with the School of Computer Science and Engineering, Central South University, China. He is also a member (E200012141M) of the China Computer Federation (CCF). His major research interests include cloud computing, fog computing, and wireless sensor networks.

HOUBING SONG (M'12–SM'14) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA, in August 2012. In August 2017, he joined the Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL, USA, where he is currently an Assistant Professor and the Director of the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us). He served on the faculty of West Virginia University, from August 2012 to August 2017. In 2007, he was an Engineering Research Associate with the Texas A&M Transportation Institute. He has authored more than 100 articles. His research interests include cyber-physical systems, cybersecurity and privacy, the Internet of Things, edge computing, big data analytics, unmanned aircraft systems, connected vehicle, smart and connected health, and wireless communications and networking.

Dr. Song is a Senior Member of the ACM. He was the very first recipient of the Golden Bear Scholar Award, the highest campus-wide recognition for research excellence at the West Virginia University Institute of Technology (WVU Tech), in 2016. He serves as an Associate Technical Editor for the *IEEE Communications Magazine*. He is an Editor of four books, including *Smart Cities: Foundations, Principles and Applications* (Hoboken, NJ, USA: Wiley, 2017), *Security and Privacy in Cyber-Physical Systems: Foundations, Principles and Applications* (Chichester, U.K.: Wiley-IEEE Press, 2017), *Cyber-Physical Systems: Foundations, Principles and Applications* (Boston, MA, USA: Academic Press, 2016), and *Industrial Internet of Things: Cybermanufacturing Systems* (Cham, Switzerland: Springer, 2016).