

Article

A Data-Gathering Scheme with Joint Routing and Compressive Sensing Based on Modified Diffusion Wavelets in Wireless Sensor Networks

Xiangping Gu ^{1,2}, Xiaofeng Zhou ^{1,*} and Yanjing Sun ³¹ College of Computer and Information, Hohai University, Nanjing 210098, China; xpgu2010@hyit.edu.cn² Faculty of Electronic Information Engineering, Huaiyin Institute of Technology, Huai'an 223003, China³ School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China; yjsun@cumt.edu.cn

* Correspondence: zhouxif@hhu.edu.cn; Tel.: +86-135-0514-0317

Received: 8 February 2018; Accepted: 26 February 2018; Published: 28 February 2018

Abstract: Compressive sensing (CS)-based data gathering is a promising method to reduce energy consumption in wireless sensor networks (WSNs). Traditional CS-based data-gathering approaches require a large number of sensor nodes to participate in each CS measurement task, resulting in high energy consumption, and do not guarantee load balance. In this paper, we propose a sparser analysis that depends on modified diffusion wavelets, which exploit sensor readings' spatial correlation in WSNs. In particular, a novel data-gathering scheme with joint routing and CS is presented. A modified ant colony algorithm is adopted, where next hop node selection takes a node's residual energy and path length into consideration simultaneously. Moreover, in order to speed up the coverage rate and avoid the local optimal of the algorithm, an improved pheromone impact factor is put forward. More importantly, theoretical proof is given that the equivalent sensing matrix generated can satisfy the restricted isometric property (RIP). The simulation results demonstrate that the modified diffusion wavelets' sparsity affects the sensor signal and has better reconstruction performance than DFT. Furthermore, our data gathering with joint routing and CS can dramatically reduce the energy consumption of WSNs, balance the load, and prolong the network lifetime in comparison to state-of-the-art CS-based methods.

Keywords: compressive sensing; wireless sensor networks (WSNs); diffusion wavelets; ant colony algorithm; data gathering

1. Introduction

Wireless Sensor Networks (WSNs) generally consist of a large number of sensor nodes and a sink node deployed in the detected environment to monitor various physical characteristics of the real world, such as temperature, voltage, wind direction, and so on. Furthermore, WSNs should have a long enough lifetime to successfully fulfill the monitoring task. However, sensor nodes are limited in terms of computational ability, communication bandwidth, and energy availability. In many cases, such as where there is an inaccessible or hostile field, a battery is difficult to recharge [1]. Therefore, how to gather sensor network data in an energy-efficient way becomes the crucial problem in practical applications. For instance, [2] indicates the importance of promoting the efficiency of intra-car multihop WSNs.

A great many research works have addressed energy-efficient challenges for WSNs from the point of view of sleep schedule [3], topology control [4], area coverage [5], mobile sink [6], and data gathering [7]. However, the core of the study is to take advantage of data-gathering and routing algorithms in WSNs to promote network performance and balanced load.

We can leverage the spatial characteristics in sensor node readings from real deployments, which is an essential technique to decrease data transmission costs while preserving relatively high recovery accuracy in the sink node [1]. In other words, the spatial correlation property of a sensor node leads to inherent data sparsity in some areas, such as wavelet domain and DCT domain. In order to solve the sparsity of such signals, compressive sensing (CS) is exploited as a novel signal-processing paradigm that provides an efficient compressive method and recovers sparse or compressible signals [8–11].

In recent years, a large number of CS-based techniques have been investigated to gather data on WSNs [12–19]. However, most of the current CS-based methods neglect the topology structure or property of sensor node readings, such as spatial characteristics. Therefore, the performance of the algorithm is limited. In this paper, the spatial property of sensor node readings is exploited to strengthen the performance of networks, considering the topology structure and sensor nodes' distance. Thus, we take advantage of spatial correlations of sensor node readings to further promote the efficiency of the data-gathering algorithm.

So far, various CS-based data-gathering schemes have been presented to promote the networks' efficiency for data gathering in WSNs [12–15]. (See Section 2 for more details.) For instance, [12] proposes a data-gathering scheme that diminishes the bottleneck of the sink in WSNs. In [13], the spatial correlation of sensor node readings is leveraged to improve the performance of WSNs. Wu et al. propose a sparsest representation basis, which makes full use of the spatial property of sensor node readings [14]. Quer et al. introduce a framework for data gathering considering CS and principal component analysis (PCA) simultaneously. In this scheme, the spatial characteristics of sensor node readings are utilized to enhance the recovery accuracy [15].

The routing protocol in data-gathering schemes has also attracted much attention in the past few years [16–19]. In [16], multi-chain CS-based data gathering is presented by considering the routing protocol from the sensor nodes to the sink of WSNs. A special routing algorithm corresponding to the topology structure is given in [17]. In [18], a random walk routing approach is introduced to reduce the energy consumption of the whole network. In [19], Xie et al. propose a cluster-based routing algorithm for data gathering in WSNs. However, these approaches do not consider the property of sensor node readings. Additionally, the optimal routing algorithm is not considered.

In this paper, a sparser basis based on improved diffusion wavelets is provided, aiming to exploit the sensor node's spatial correlation features. Moreover, a novel data-gathering algorithm with joint CS and modified ant colony routing is presented. More specifically, the contributions of this paper are as follows.

- Improved diffusion wavelets are used as the basis for data gathering, which makes full use of sensor nodes' spatial correlations;
- A sparse measurement matrix is used, where the non-zero components of each row denote the offspring nodes of one projection node, which only requires a fraction of nodes to participate in each measurement task, leading to a dramatic decrease in the energy consumption.
- Modified ant colony routing compressively considering the next hop sensor node's residual energy and path length is proposed; to speed up the algorithm coverage ratio and avoid a local optimal, the pheromone impact factor is improved. Furthermore, a novel data-gathering scheme that integrates MST (Minimum Spanning Tree), modified ant colony routing, and compressive sensing is provided;
- Theoretical proof is shown on the equivalent measurement matrix to satisfy the RIP.
- The superiority of our approach is demonstrated by numerical experiments with synthetic and real sensor data. The simulations show that our sparse basis sparsify the signal. In addition, our approach can accurately reconstruct the original signal, thereby reducing the energy consumption of WSNs and balancing the network load.

The remainder of this paper is organized as follows. First, in Section 2, related work is introduced. Section 3 discusses preliminaries on CS theory and the system model. Section 4 analyzes a lot of

WSNs datasets and signal spatial correlation characteristics. Furthermore, the proper sparse basis is designed using a modified diffusion wavelet. Moreover, it provides the modified ant colony routing algorithm. In addition, we describe a novel compressive data-gathering algorithm with joint improved ant colony algorithm and MST. Section 5 proves that the equivalent measurement matrix satisfies the RIP. The numerical results of synthetic and real datasets are analyzed in Section 6. Conclusions and future work are given in Section 7.

2. Related Work

Recently, numerous CS-based data-gathering approaches have been provided to increase the network's lifetime by decreasing the amount of transmission data and balancing load of the whole WSNs [12,13,20,21]. Luo et al. first analyzed the CS-based data-gathering scheme (plain CS) in large-scale WSNs, namely, compressive data gathering (CDG) [12]. In [12], Luo et al. prove that the network capacity is proportional to the sparsity level of sensor node readings. However, this method also demonstrates that an increase in the number of measured samples leads to growth of transmission costs compared to the non-CS approach. CStorage [13] exploits the spatial correlation of sensor node readings along with CS to reduce the number of transmission data points, aiming at prolonging the WSNs' lifetime. Bajwa et al. introduce compressive wireless sensing, whereby a distributed, matched-source channel communication architecture was proposed for energy-efficient estimation of sensor node readings [20]. However, sensor node readings' spatial correlation features in [20] are not considered. The authors of [21] show that hybrid CS can achieve high network throughput, while plain CS may not yield a significant improvement in throughput because of the dense measurement matrix.

Up to now, the interaction between routing and compression data gathering has seriously hindered the progress of CS in WSNs [12,22]. These data-gathering methods jointly use routing and CS to mitigate the data throughput. The authors of [22] address the data collection problem in WSNs, with routing used in combination with CS to transmit random data projections. However, this technique demonstrates that a sparse measurement matrix not only deviates from RIP property, but does not minimize transport energy consumption for each CS measurement. Nevertheless, in [12], Luo et al. claim that measurement matrix $[I \ R]$ has good RIP; in addition, CDG can further reduce communication costs for both chain-type and tree-type routing. Alternatively, this method requires a large number of sensor nodes involved in each projection gathering, bringing about high transmission costs. To significantly reduce data traffic and save power, distributed multi-chain compressive sensing based on a routing algorithm in WSNs is presented. However, the measurement matrix is still a dense measurement matrix [16]. In [23], a distributed sparse random measurement is proposed, where the sparse or compressible signal can be recovered. Wang et al. consider that some sensor nodes instead of all of them are required in each measurement. Furthermore, the authors of [23] show that sparse random projection significantly enhances network performance. The communication costs can be reduced to $O(\log N)$ packets per sensor, but this strategy utilizes random routing in the networks, which mitigates the energy efficiency of WSNs. In [14] a sparsest random scheduling for compressive data gathering in WSNs is provided to satisfy RIP property; a sparse basis is designed based on the measurement matrix and sensor node readings. The scheme claims that it can achieve good results in real datasets, but the routing is not considered. Routing and CS are also incorporated into [17], which considers special random routing of a given network topology. However, sensor nodes' compressible ration and routing schemes considerably affect the transport costs. In fact, this method is impractical since sensor nodes are randomly deployed in the field of WSNs applications. In [18], Zheng et al. introduces an algorithm based on a random walk that is independent of network topology. Nevertheless, those techniques in [14,17,18,23] suffer from the fact that the routing of the projection to the sink node is not optimal, which leads to an extra transmission bottleneck for the sensor nodes near the sink node in WSNs. [19] follows the dense matrix used in [12] and hybrid CS in [21]; a cluster-based data-gathering scheme is proposed. This method indicates that it reduces the transport cost and balances the traffic load of the network, but the same issue of a dense matrix exists

in [19]. Furthermore, there is no theoretical proof that the equivalent measurement matrix satisfies the RIP in [19].

Existing CS-based data-gathering algorithms adopt random, chain-type and tree-type routing in which numerous sensor nodes participate in each measurement. However, few studies take into account optimal routing in CS-based data gathering. Moreover, design the sparse basis does not make full use of the spatial correlation of the sensor nodes and network topology. In addition, there is no theoretical guarantee that the equivalent sensing matrix obeys RIP in most research works.

3. Preliminaries

3.1. CS Theory

CS is a novel technique that is used to compress and recover an original signal when it has sparse representation in some domains. As above, we consider the sparse signal $X \in \mathbb{R}^N$, describing the sensor node readings in WSNs with N nodes. is represented as follows:

$$X = \Psi S = \sum_{i=1}^N \psi_i s_i, \quad (1)$$

where $\Psi = [\psi_1, \psi_2, \dots, \psi_N] \in \mathbb{R}^{N \times N}$ is some basis matrix, and $S \in \mathbb{R}^N$ is coefficient vector. When can be represented as a linear combination of K column vectors of Ψ with $K \ll N$, it means that the signal is K -sparse, which demonstrates that it has only K nonzero components or $(N - K)$ smallest components can be ignored. In this case, Ψ is referred to as the sparse basis. Thus, the information can be compressed by simple linear projection. This projection matrix $\Phi \in \mathbb{R}^{M \times N}$ is an $M \times N$ measurement matrix, $Y \in \mathbb{R}^M$ is the measurement vector, and $K < M < N$. The compressed measurements can be described as

$$Y = \Phi x = \Phi \Psi S = AS, \quad (2)$$

where A denotes the equivalent sensing matrix.

The measurement matrix needs to satisfy the RIP conditions [24].

Definition 1. (RIP [10,11]): A matrix Φ satisfies the restricted isometric property of order K if there exists a $\delta_K \in (0, 1)$ such that

$$(1 - \delta_K) \|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta_K) \|x\|_2^2 \quad (3)$$

for all K -sparse vectors $S \in \mathbb{R}^N$.

However, the reconstruction of the original signal x from y is a problem. Candès and Tao [9] and Donoho [10] have shown that signal X can be reconstructed via ℓ_1 as

$$\hat{x} = \underset{x}{\operatorname{argmin}} \|x\|_1 \quad \text{s.t. } Y = \Phi X, \quad (4)$$

where Φ satisfies RIP and $M \geq O(K \cdot \log(N/K))$, X can be recovered successfully with high probability. In addition, there are a large number of reconstruction algorithms such as the basis pursuit (BP) algorithm [24], orthogonal matching pursuit (OMP) algorithm [25], CsaMP [26], Stage-wise Orthogonal Matching Pursuit (StOMP) [27], gOMP [28], and so on.

3.2. System Model

We describe WSNs where N sensor nodes are randomly deployed in a square area. The system model is represented by a connected graph $G(V, E)$, where the vertex set V denotes the nodes in the networks, and the edge set E denotes the wireless links between the different nodes. Node i can communicate with node j if they are involved in the communication range. We assume that

the single hop distance d_{ij} between node i and node j can be represented as a Euclidean distance. At a sampling instant, each sensor node i takes a measurement x_i ; the goal of the data gathering in WSNs is to collect sufficient information to reconstruct the N -dimensional signal $X = [x_1, \dots, x_N]^T$. In this paper, the energy consumption model is similar to that in [29], namely, Equation (5) and Equation (6). When the distance between transmission node i and receive node j is greater than d_0 , the multi-path fading model is utilized. When the distance is less than d_0 , the free-space model is adopted.

$$E_{Ti}(L, d) = \begin{cases} E_{elec} \times L + E_{amp} \times L \times d^4, & d \geq d_0 \\ E_{elec} \times L + E_{fs} \times L \times d^2, & d < d_0 \end{cases} \quad (5)$$

$$E_{Rj}(L) = E_{elec} \times L, \quad (6)$$

where $E_{Ti}(L, d)$ and $E_{Rj}(L)$ describe the energy consumption of transporting and receiving the L bit data packet. E_{elec} denotes the power expended to run the transmitter or receiver circuitry of the sensor node. E_{amp} and E_{fs} represent energy consumption for a multi-path fading amplifier and free-space amplifier, respectively.

4. Our Proposed Algorithm

4.1. Datasets

The spatial correlation features among the different sensor nodes, which can be exploited to considerably reduce transmission costs in WSNs in [15] by analyzing the signal features utilizing PCA technique. However, in this paper, we follow the information theory as in [30] to evaluate spatial correlation characteristics of real sensor nodes' data in WSNs. The entropy concept in information theory denotes the information involved in datasets. We investigate the properties of the signal in view of the spatial marginal and conditional entropy because the gap between marginal entropy and conditional entropy demonstrates data compressibility. We extract five different matrixes from DEI [31], IntelLab [32], LUCE-EPFL [33], CitySense [34], and OrangeLab [35], which are deployed in campus, indoor, and urban city environments. These data features are summarized in Table 1.

Table 1. Details of datasets in WSNs.

Name	Time Period	Physical Signal	Matrix Size	Frame Length
LUCE-EPFL	12–15 January 2007	Temperature, Humidity, Solar Radiation, Wind, Water	81 nodes \times 856	5 min
IntelLab	28 February 2004–5 April 2004	Temperature, Humidity, Light, Voltage	54 nodes \times 500	30 s
CitySense	14 October 2009–21 November 2009	Temperature, Wind	8 nodes \times 887	60 min
DEI	19–22 March 2009	Temperature, Humidity, Light	29 nodes \times 781	5 min
OrangeLab	26–27 August 2008	Temperature, Light, Voltage	75 nodes \times 65	15 min

Let $x_{i,t}$ represents the i sensor node readings at slot t . The signal can also be expressed using matrix-type Equation (7), where the rows are the i th sensor node readings and the columns are sensor node readings at t slot.

$$X = \begin{bmatrix} x_{0,0} & \dots & x_{0,t} & \dots & x_{0,T-1} \\ x_{1,0} & \dots & x_{1,t} & \dots & x_{1,T-1} \\ \dots & \dots & \dots & \dots & \dots \\ x_{N-1} & \dots & x_{N-1,t} & \dots & x_{N-1,T-1} \end{bmatrix} \quad (7)$$

4.1.1. Spatial Marginal and Conditional Entropy

Sensor node readings are different from the signals of a 2D image, which has a discrete sequence range from 0 to 255, while the signal collected from WSNs is continuous. Therefore, we need to preprocess the signals before calculating spatial marginal and conditional entropy. Take temperature sequence [24.23 24.66 24.75 23.65 23.70 24.12 22.78 22.23 22.16 22.96 21.29 21.11] for example: the data

can be divided into Q equal sections (select proper equal sections depending on the real data; here, Q is 4); we calculate the occurrence probability per section so that the marginal entropy yield is 1.9591 bits.

The spatial marginal entropy is defined as the entropy of different nodes sensing the state at slot t . Suppose α_k is the occurrence frequency of state z_k in the t column Z_t . The probability of $P(z_k)$, spatial marginal entropy $H(Z_t)$, and conditional entropy $H(Z_{i,t}|Z_{j,t})$ are expressed by Equations (8)–(10) respectively:

$$P(z_k) = \lim_{N \rightarrow \infty} \frac{\alpha_k}{N} \quad (8)$$

$$H(Z_t) = - \sum_{t=0}^{Q-1} P(z_t) \cdot \log_2 P(z_t) \quad (9)$$

$$H(Z_{i,t}|Z_{j,t}) = H(Z_{i,t}, Z_{j,t}) - H(Z_{j,t}). \quad (10)$$

4.1.2. Spatial Compressibility

In this section, we pick up two temperature subsets from LUCE-EPFL and CitySense that are representative of the other datasets to approximately estimate spatial marginal and conditional entropy. The cumulative distribution functions (CDF) are provided in Figures 1 and 2. In Figure 1, the red star curve represents CDF of marginal entropy, while the blue triangle curve is CDF of conditional entropy on LUCE-EPFL datasets. As can be seen in Figure 1, the values of marginal and conditional entropy are less than 3 bits. It is worth noting that conditional entropy is only about 0.7 bit. There is a big gap between the CDF curve of conditional entropy and the CDF curve of marginal entropy, which demonstrates that data storage space can be compressed considerably. Similarly, in Figure 2, the CDF of conditional entropy from CitySense is much smaller than that of marginal entropy. The same phenomenon with CitySense indicates the spatial compressibility of the sensor nodes' readings in WSNs. We especially study five different datasets divided into classes based on the physical conditions and calculate the marginal entropy per signal. The results are recorded in Table 2. In Table 2, the first column indicates the marginal temperature entropy of different datasets. The entropy range is 2.5 bit to 3.1 bit. The other columns are humidity, solar radiation, wind, water, light, and voltage. The minimum marginal entropy is 0.592 bit of DEI light, while the maximum is 3.256 bit of CitySense wind in general. Therefore, it is inadvisable to store sensor node readings with 32 bit or 64 bit, that is to say, sensor node readings are compressible to some extent.

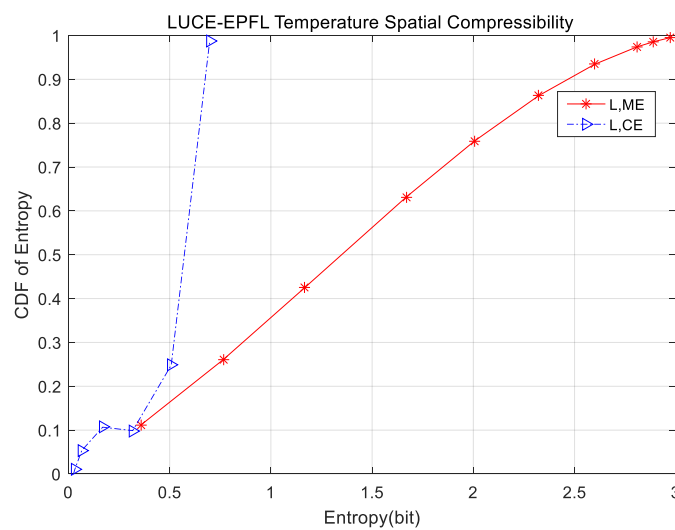


Figure 1. Spatial compressibility of LUCE-EPFL temperature.

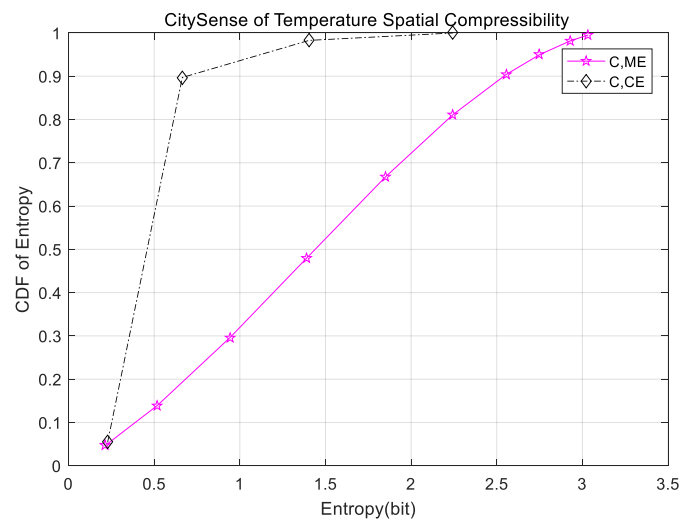


Figure 2. Spatial compressibility of CitySense temperature.

Table 2. Analysis of the marginal entropy (in bits) of different datasets in WSNs.

Name	Temperature	Humidity	Solar Radiation	Wind	Water	Light	Voltage
LUCE-EPFL	2.971	2.810	2.450	2.484	2.991	—	—
IntelLab	2.543	1.629	—	—	—	2.151	1.015
CitySense	3.034	—	—	3.256	—	—	—
DEI	2.589	2.510	—	—	—	0.592	—
OrangeLab	2.832	—	—	—	—	1.193	1.836

4.2. Modified Diffusion Wavelets

Conventional CS-based data-gathering approaches generally assume that sensor node readings have perfect sparse features under FFT, DWT, DCT, etc. To make full use of the spatial correlation property, which is demonstrated by experiments in Section 4.1, we take diffusion wavelets [36] as the sparse basis considering the spatial correlation of sensor node readings in WSNs. One is the nodes degree, and the other is the distance between the different sensor nodes. In addition, an improved QR decomposition of Givens transform is introduced to set up the sparse basis. Then, we describe how to construct the modified diffusion wavelets in detail. However, diffusion wavelets are affected significantly by the diffusion operator, which is equivalent to the wavelet function of a discrete wavelet transform. Diffusion is utilized as a smoothing and scaling technique to enable multi-scale and coarse-grained application. The detailed steps are shown in Algorithm 1.

Step 1: Suppose that $G(V, E)$ denotes a graph with N sensor nodes deployed in the monitoring environment, as indicated in Section 3.2. Diffusion wavelets are introduced to set up an orthonormal basis for functions supported by the topology graph of WSNs. In this section, we take a random deployment of WSNs to explain this process. WSNs' topology of 640 sensor nodes is shown in Figure 3. In Figure 3, the hexagram vertex represents the sensor node, while the blue edge denotes the wireless link.

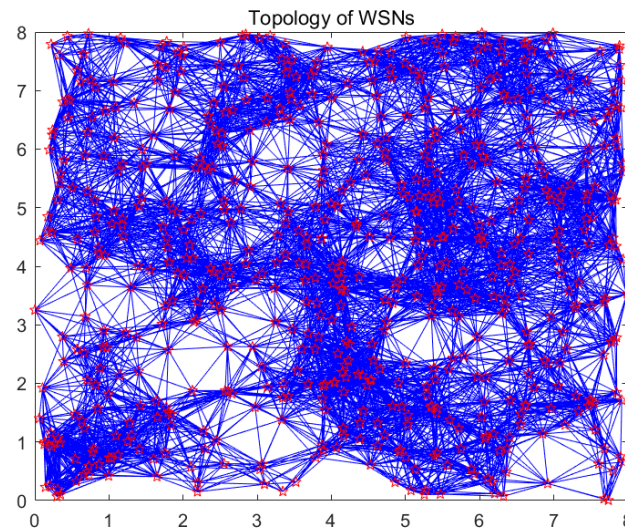


Figure 3. Topology of WSNs.

Step 2: Calculate the weight adjacency matrix of $G(V, E)$, which is denoted as $\Omega = [w_{i,j}]$. $w_{i,j}$ is the weight of the edge in the graph. In this section, we consider two different cases of weight. The sensor node degree is chosen as the weight in the first scheme, while the distance between sensor nodes is taken into consideration to exploit the spatial correlation features, aiming to mitigate the load of WSNs in another scheme. In the former case, we show an example of a graph and corresponding weight adjacency matrix in Figure 4. In the latter case, the weight function is given below in Equation (11), which follows a similar method to [37].

$$w_{ij} = \begin{cases} d_{ij}^{\gamma} & , i \neq j, d_{ij} \leq r \\ \chi & , \text{otherwise} \end{cases} \quad (11)$$

where r is the maximum distance among the sensor nodes that can directly communicate by a single hop. d_{ij} is the Euclidean distance between node i and node j . γ is a negative number, while χ is a small positive number.

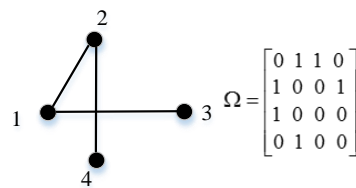


Figure 4. An example of the weighted adjacency matrix of a graph.

Step 3: Generate a normalized Laplacian matrix of $G(V, E)$: $\Lambda = [\lambda_{ij}]$. In [38], Chung et al. indicates that Λ is the degree of correlations among different function values provided at the vertices of the graph $G(V, E)$. In the first schedule, we denote λ_{ij} using Equation (12), while the other schedule considering spatial correlation implements Equation (13). Generally speaking, an eigenvalue or eigenvector shows the special correlations at some scale. We need to split the space of Λ if we decompose the signal sampled of the $G(V, E)$ in a multi-scale.

$$\lambda_{ij} = \begin{cases} 1, & i = j \\ -\frac{w_{ij}}{\sqrt{\sum_u w_{iu} \sum_u w_{uj}}}, & \text{otherwise} \end{cases} \quad (12)$$

$$\lambda_{ij} = \begin{cases} 1 - \frac{\chi}{\sum_u d_{ij}^u}, & i = j \\ -\frac{d_{ij}^{\chi}}{\sqrt{\sum_u d_{iu}^{\chi} \sum_u d_{uj}^{\chi}}}, & \text{otherwise} \end{cases} \quad (13)$$

Step 4: However, the diffusion operator O stems from Λ , where O shares the same eigenvalues as Λ (less than 1). The diffusion operator is $O = I - \Lambda$ or $O = \Lambda/2$; in this paper, we choose the first expression.

Step 5: Consequently, recursively raise O to power 2, and delete the diminishing eigenvalues with a threshold. Step by step, this approach splits the space spanned by the eigenvectors. Let the initial space of O be $x_0 = \mathbb{R}^N$, which is represented by scale space $\{x_j\}_{j \in N}$ and wavelet space $\{V_j\}_{j \in N}$. Wavelet space V_j is different between x_j and x_{j+1} . Then, we derive Equation (14):

$$x_{j+1} = x_0 \oplus V_0 \oplus V_1 \oplus \cdots \oplus V_j. \quad (14)$$

Here, steps 5.1–5.5 accomplish the modified QR decomposition, where $[O]_{x_a}^{x_b}$ indicates the column space of matrix O denoted by basis x_b at scale b , and row space is denoted by basis at scale a , $[x_b]_{x_a}$ represents basis x_b denoted on the basis x_a .

Step 6: In the end, the diffusion wavelet basis Ψ is the concatenation of the scale functions and wavelet functions.

Algorithm 1 Modified diffusion wavelets.

Input: the number of sensor nodes N , communication radius r , decomposition level η , precision ε and MQR function.

Output: sparse basis Ψ .

1 generate a graph $G(V, E)$

2 compute weight adjacency matrix $\Omega = [w_{ij}]$ according to the vertex degree/Equation (11)

3 calculate normalized Laplacian matrix Λ relying on Equation (12)/Equation (13)

4 generate diffusion operator $O = I - \Lambda$

5 recursively raising O to power 2

5.1 for $\eta = 0$ to $\eta - 1$

5.2 $[x_{j+1}]_{x_j}, [O]_{x_0}^{x_1} \leftarrow \text{MQR}([O^{2^\eta}]_{x_j}^{x_j}, \varepsilon)$

5.3 $O_{j+1} := [O^{2^{\eta+1}}]_{x_{j+1}}^{x_{j+1}} \leftarrow [x_{j+1}]_{x_j} [O^{2^{\eta+1}}]_{x_j}^* [x_{j+1}]_{x_j}^*$

5.4 $[\Psi_\eta]_{x_j} \leftarrow \text{MQR}(I_{\langle x_j \rangle} - [x_{j+1}]_{x_j} [x_{j+1}]_{x_j}^*, \varepsilon)$

5.5 end for

6 concatenation of the scale functions and wavelet functions is regarded as the sparse basis Ψ .

MQR Function:

$Q, R \leftarrow \text{MQR}(B, \varepsilon)$

Input: B : $N \times N$ sparse matrix, ε

Output: Q, R matrix, possibly sparse, such that $B =_\varepsilon QR$

(1) Q is orthogonal

(2) R is upper triangular up to a permutation

(3) The columns of Q ε -span the space spanned by the columns of B

Figure 3 denotes the topology of 640 nodes of WSNs, and also represents some scale functions. To visualize the wavelets' function, we plot Figures 5 and 6 using the first scheme (Equation (12)) and the second scheme (Equation (13)), respectively. Figure 5a introduces the second-level wavelet function, while Figure 5b is the 10th-level wavelet function for the former schedule. Figure 6 represents the second schedule considering the spatial correlation of sensor node readings in WSNs. Figure 6a,b indicate first- and 10th-level wavelet functions, respectively. Obviously, the second scheme is set up on a sparser basis, for it can capture the sensor node's relationship and network topology properties and thus gain valuable information from the real world.

4.3. Modified Ant Colony Routing Algorithm

In this paper, in order to decrease the whole network transmission load and prolong the network lifetime, we provide a modified ant colony routing algorithm, where to speed up the convergence rate and avoid local optimal of the algorithm, pheromone impact factor is improved. Here, we select the energy consumption model described in Section 3.2. The traditional ant colony optimization algorithm selects the next hop depending on Equation (15) [39]:

$$p_{ij}^{\theta} = \begin{cases} \frac{[\tau_{ij}(t)]^{\zeta} [\rho_{ij}(t)]^{\xi}}{\sum_{v \in allowed_{\theta}} [\tau_{ik}(t)]^{\zeta} [\rho_{ij}(t)]^{\xi}}, & j \in allowed_{\theta} \\ 0, & others \end{cases}, \quad (15)$$

where $\tau_{ij}(t)$ denotes the pheromone information on edge (i, j) , while $\rho_{ij}(t)$ is the heuristic information on edge (i, j) . ζ and ξ are impact factors demonstrating the importance degree of the pheromone information and heuristic information. In order to speed up the convergence rate and avoid local optimal, impact factor ζ is modified as in Equation (16):

$$\zeta = \mu(1 + e^{-10 \times (iter / totiter)^{10}}), \quad (16)$$

where μ is a small positive constant $\in (0, 1]$; $iter$ and $totiter$ refer to current iterations and total iterations, respectively. In Equation (16), ζ gradually becomes smaller as the number of iterations increases. In other words, the proportion of pheromones will diminish when the number of iterations rises.

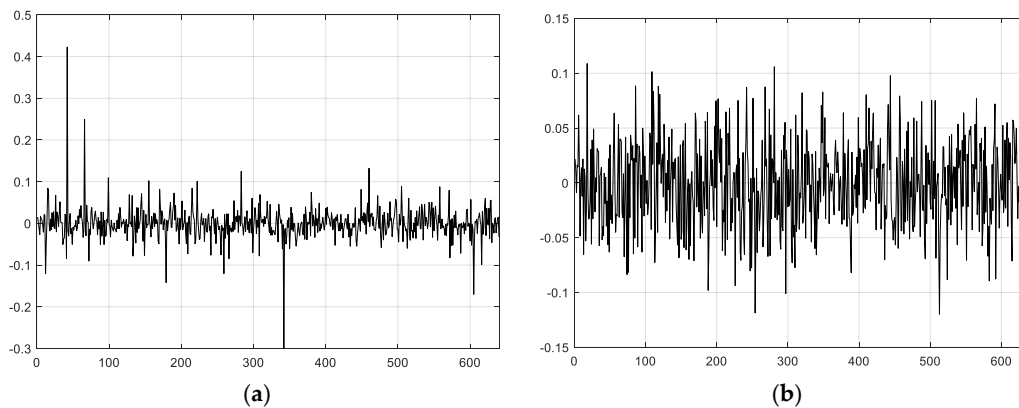


Figure 5. Wavelet functions of the first scheme: (a) 2nd-level wavelet function; (b) 10th-level wavelet function.

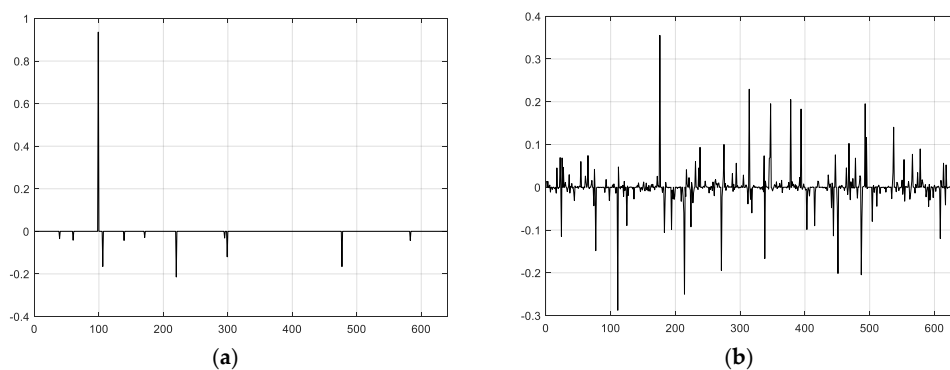


Figure 6. Wavelet functions of the second scheme: (a) First-level wavelet function; (b) 10th-level wavelet function.

Furthermore, to yield optimal routing by the ant colony algorithm, in this subsection, a sensor node's residual energy and path length are taken into consideration simultaneously. So, the fitness value of each routing is presented as follows:

$$Fitness = \beta(E_{ave} \times E_{min}) + \sigma Len_{\theta}^{-iter}, \quad (17)$$

where E_{ave} indicates the average residual energy, while E_{min} represents the node minimal energy of ants passing through the path. Len_{θ}^{-iter} denotes the reciprocal of path length for given θ th ant and $iter$ th iterations. β and σ are $\in [0, 1]$ constants, and $\beta + \sigma = 1$. Consequently, the path with the largest fitness function value is chosen as the optimal routing, thus balancing the network load and prolonging the network lifetime. Specifically, the modified ant colony algorithm is shown in Algorithm 2.

Algorithm 2. Modified ant colony algorithm.

Input: the number of sensor nodes N , the power expended to run the transmitter or receiver circuitry of sensor node E_{elec} , energy consumption of multi-path fading amplifier E_{amp} , energy consumption of free-space amplifier E_{fs} , distance threshold d_0 , impact factors of pheromone information ς , impact factors of heuristic information ξ , μ is a small positive constant $\in (0, 1]$, pheromone information on edge (i, j) τ , heuristic information on edge (i, j) ρ , β and σ are $\in [0, 1]$ constants.

Output: optimal routing *Routing*.

```

1 Initialization routing  $R(Path, iter, \theta)$ , energy for each node and tabu
2 calculate distance  $d_{ij}$  of different nodes,  $\rho_{ij} = 1/d_{ij}$ 
3 while maximum iterations has not be reached
4 for  $\theta=1: \Theta'$ 
5 compute  $allowed_{\theta}$  according to the node communication radius.
6 generate transition probability  $p_{ij}^{\theta}$  based on Equations (15) and (16)
7 choose the next hop node, relying on  $p_{ij}^{\theta}$ , modify routing and tabu
8 the destination node or not? If not, go back to step 2, or proceed to step 9
9 update the node residual energy based on Equations (5) and (6), routing depending on Equation (17)
10 end for
11 end while
12 return the optimal routing Routing.
```

4.4. Compressive Data Gathering

WSNs are utilized for gathering physical signal from the real world in practical applications. Without using CS theory, which is the simplest method, a data-gathering scheme with the help of the tree topology is shown in Figure 7a. In order to dramatically decrease communication costs and prolong the network lifetime, the authors of [12] consider that the sink node receives only M packets instead of N packets of original data from the whole network. In the end, at the sink, CS theory is used to reconstruct the original data. For the CDG algorithm, each node in the WSN multiplies its readings x_j using the corresponding j column vector of basis matrix Φ . Next, the sensor node adds them to its own readings after receiving all same-size vectors from descendent nodes and transmitting the final results to its parent node with M packets. Let us illustrate the product of CDG in Figure 8, where Φ is $M \times N$ matrix, and each column corresponds to one weight sum. In the plain CS [11], all nodes in WSNs transmit M packets and each has equal transmission costs; therefore, each CS measurement cost remains relatively high. An example of the plain CS mechanism is given in Figure 7b. It is obvious that for these approaches (non-CS and plain CS), the former transmits fewer packets compared with plain CS from the point of view of child nodes. In [14], Wu et al. provides the hybrid CS method, where non-CS is chosen when the number of packets is less than or equal to M ; alternatively, plain CS is used. Figure 7c illustrates the idea of hybrid CS. In Figure 7c, thin circles indicate a forward node using non-CS, while thick circles denote the gathering node using plain CS.

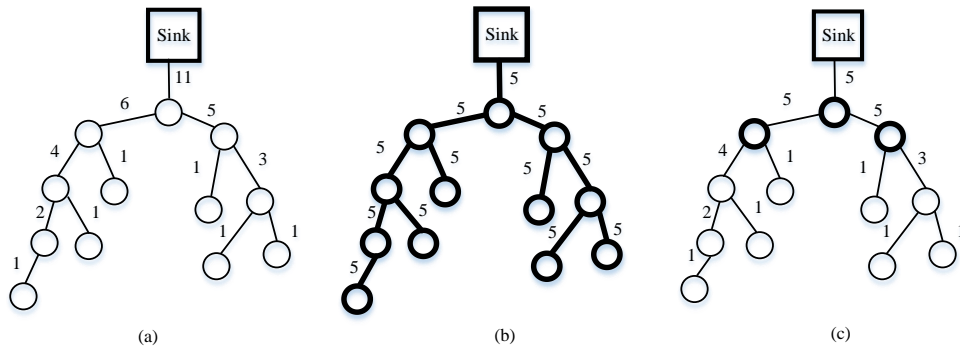


Figure 7. (a–c) Comparison of the three different data-gathering schemes.

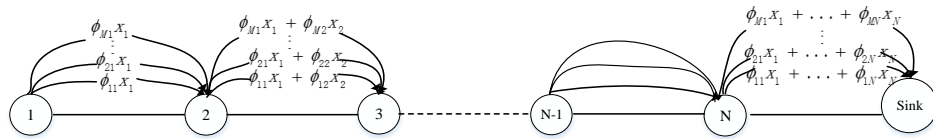


Figure 8. Compressive data gathering (CDG).

4.5. Data Gathering with Sparse Random Projections

The aforementioned methods (plain CS and hybrid CS) still experience great challenges for all nodes involved in each measurement; in other words, the two mechanisms follow the dense matrix. Therefore, Wang et al. [23] propose a distributed data-gathering algorithm according to sparse random projections. In this algorithm, M nodes are randomly chosen to collect M weight sums in WSNs. Each projection sensor node collects one weight sum. Now, we explain how to accomplish data gathering in detail by means of Figure 9. Suppose that node 5 is a projection node, and $\phi_{ij} \neq 0$ of nodes 10, 15, 20, and 26. Then, node 5 initializes the projection by querying nodes 10, 15, 20, and 26. These nodes reply to node 5's query with their readings x_j in one packet. Finally, node 5 gathers all the data with its own data: $\sum_{j=1}^N \phi_{ij}x_j$ and transmits them to the sink node via the shortest path. Meanwhile, the transmission costs for each measurement reduce dramatically from $O(N)$ in a dense matrix to $O(\log N)$ in a sparse random matrix.

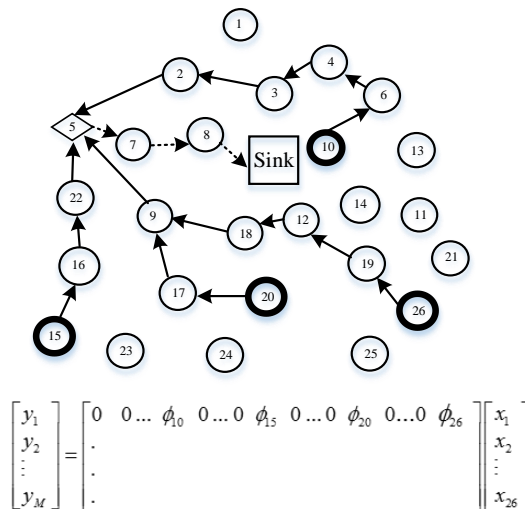


Figure 9. An example of data gathering per random projection node using Φ .

4.6. A Novel Data-Gathering Scheme with Joint Routing and CS

Obviously, according to the analysis, the network load in hybrid CS is unbalanced. Specifically, sensor nodes near the sink node will consume more energy than those far from the sink node because of forwarding data more times. This results in sensor nodes near the sink dying earlier. However, [23] does not consider the total network costs for each random projection. To avoid the drawbacks, one can leverage the advantages of the algorithms; in this section, we present our data-gathering strategy combining joint routing and CS.

Firstly, randomly choose M projection nodes in the network with probability $\frac{M}{N}$, which follows [23]. In the CS theory, the sink node needs M measurements to reconstruct the original data. Therefore, these M projection nodes will be selected as the gathering node, defined as g_1, g_2, \dots, g_m , to collect one random measurement y_i , and transmit y_i to the sink node. Then, distribute non-zero elements in each row of measurement matrix Φ as uniformly as possible to guarantee the sparse features of the measurement matrix; the number of non-zero elements in each row should equal to $\left\lceil \frac{N}{M} \right\rceil$, which is related to Algorithm 3's step 1. Additionally, each column of measurement matrix represents a sensor node, so if a column of the matrix has full zero elements, the data from its special sensor node should be thrown away. ϕ_i , the column vector of measurement matrix Φ is required to store each sensor node memory in advance. Now, an example of measurement matrix is given in Equation (18), where $M = 5$, and $N = 10$.

Secondly, each row ϕ_{g_i} of measurement matrix Φ corresponds to one projection node. However, the number of each row coefficient is N , which is assigned to the size of network. $\phi_{ij} \neq 0$ indicates that j th candidate nodes belong to the i th projection node's. Here, this subsection corresponds to step 3 in Algorithm 3.

$$\Phi = \begin{bmatrix} 0 & 0 & 0 & \phi_{1,4} & 0 & \phi_{1,6} & 0 & 0 & 0 & 0 \\ \phi_{2,1} & 0 & \phi_{2,3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \phi_{3,5} & 0 & 0 & \phi_{3,8} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \phi_{4,7} & 0 & 0 & \phi_{4,10} \\ 0 & \phi_{5,2} & 0 & 0 & 0 & 0 & 0 & 0 & \phi_{5,9} & 0 \end{bmatrix} \quad (18)$$

Subsequently, we set up the routing, which is from the offspring sensor nodes to the projection nodes and the projection nodes to the sink node, respectively. Based on the MST algorithm, access all candidate sensor nodes of a given projection node. In the first stage, the projection node is considered one root node tree. In the step 4 initialization stage in Algorithm 3, $Tree_i$ is assigned by i , and the temporary variable $temp$ also yields i . Then steps 5–12 use the MST algorithm to construct the tree, adding the candidate nodes step by step. If $temp$ is not empty, step 6 deletes the top node of the $temp$ queue and puts its neighbor node in the $Tree$ and $temp$. The next step is to delete them from Can_i if they belong to Can_i . Note that these candidate nodes must be directly connected to the parent node by a single hop. If there are still some candidate nodes not involved in the tree, the *Dijkstra* algorithm [40] is proposed, aiming to find the shortest path from the residual nodes to the tree (steps 14–19), and we add the residual candidate nodes (steps 20–22).

Finally, this loop of 13–26 lines will repeat until Can_i is empty. The modified ant colony routing technique is utilized to transmit packets of projection nodes to the sink node, namely step 27 of Algorithm 3. Consequently, Algorithm 3 terminates by generating the optimal routing between the projection nodes and the sink node, and an M routing tree from the projection nodes to their own candidate nodes. Our novel algorithm (Algorithm 3) is shown in more detail. The modified ant colony algorithm jointly considers the sensor node's residual energy and the path length, which will not only balance the whole network load, avoiding nodes near the sink node dying earlier, but will prolong the network lifetime. In this way, the transmission costs should be greatly decreased compared to hybrid CS.

Figure 10 is taken as an example to describe the details of Algorithm 3. It can be seen from Figure 10 that the thick circle indicates a projection node, the thin circle is a sensor node, the arc with an arrow represents the routing from the projection node to the sink node, and the dashed line, solid line, dotted and dashed line, and thick line describe four different projection routings. Sensor nodes 6, 12, 26, and 39 are chosen as projection nodes in the networks. Suppose that node 39 is the projection node in projection 4 routing, which has non-zero coefficients 9, 11, 20, 28, 31, 32, 33, 34, 35, 38, 39 of row vector ϕ_4 in the measurement matrix. Here, projection node 39 wants to establish a tree using Algorithm 3. Above all, node 39 considers itself a tree with only one node. Select its neighbor nodes 33 and 34, which can be directly coupled with node 39 by a single hop in the following step. Next, node 33 retrieves candidate node 32 as its neighbor node, so node 32 is added to the tree, while node 34 finds node 28 is also its neighbor node; similarly, node 28 is involved in the tree. However, in the coming stage, there are no candidate nodes directly connected with the tree. Therefore, the *Dijkstra* algorithm is utilized to find the shortest path from the residual candidate nodes to the generated tree. Then nodes 31 and 38 are all joined to the tree, and nodes 35 and 36 are linked to node 28 by the shortest path. Again, nodes 20, 9, and 11 are involved in the tree using the MST algorithm. Finally, node 39 queries an optimal routing to the sink node for the projection node 39, which is $39 \rightarrow 33 \rightarrow 25 \rightarrow 18 \rightarrow 8 \rightarrow \text{Sink}$, rather than $39 \rightarrow 19 \rightarrow 26 \rightarrow 11 \rightarrow 2 \rightarrow \text{Sink}$ using the *Dijkstra* algorithm. The reason is that the modified ant colony chooses the next hop considering not only residual energy but also path length. So, node 33 chooses node 25, which has more residual energy compared with node 19. The same reasoning is applied to node 25 and node 18. Node 1 near the sink node forwards data many times from the other nodes, leading to considerable energy reduction. Therefore, in order to ease the burden on the network, node 8 chooses a direct path to the sink node, instead of node 1, although node 1 is actually the shortest path. Similarly, the routing constructed in projections 1, 2, and 3 is shown in Figure 10. At the sink node, sensor signal reconstruction is implemented by Algorithm 4. In the process of the sensor signal, the gOMP [28] recovery algorithm is utilized, where r_{it} denotes residual error, it is iteration. \emptyset represents the null set, a'_j is j th of A . Λ_{it} denotes the sets of indexes of it iteration.

Algorithm 3. Our proposed algorithm.

Input: $G(V, E)$
Output: $\text{Tree}\{1, 2, \dots, M\}, \text{Opt}$

1 randomly select M sensor nodes g_1, g_2, \dots, g_M in the network probability $\frac{M}{N}$, generate Φ

2 **for** $i = 1 : M$

3 query candidate nodes $\text{Can}_i(\phi_{ij} \neq 0)$ of projection nodes i

4 initialization $\text{Tree}_i \leftarrow i, \text{temp}_i \leftarrow i$

5 **while** !empty(temp) **do**

6 $\text{CanNode} \leftarrow \text{Del}(\text{temp})$

7 **if** CanNode is i 's candidate node

8 $\text{Tree} \leftarrow \text{CanNode}$

9 $\text{temp} \leftarrow \text{CanNode}$

10 $\text{Del}(\text{Int}_i, \text{CanNode})$

11 **end if**

12 **end while**

13 **while** !empty(Can_i) **do**

14 **for** all residual candidate nodes $r \in \text{Can}_i$

15 $\text{Path}(r) \leftarrow$ find a shortest path to Tree_i using the *Dijkstra* algorithm

16 **if** $\text{shortestpath} > \text{Path}(r)$

17 $\text{shortestpath} \leftarrow \text{Path}(r)$

```

18 end if
19 end for
20  $Tree \leftarrow shortestpath$ 
21  $temp \leftarrow shortestpath$ 
22  $Del(Can_i, shortestpath)$ 
23 while !empty(temp) do
24 go back to steps 7–11
25 end while
26 end while
27  $Opt_{tr} \leftarrow$  Optimal routing from  $i$  to the sink node using Algorithm 2
28 return  $Tree\{1, 2, \dots, M\}$ 
29 end for

```

Algorithm 4. Sensor signal reconstruction.

```

1 Input: received data  $X$ , measurement matrix  $\Phi$ , the number of atom is  $n_{atom}$ 
2 Output: reconstruct data  $\hat{x}$ 
3 generate sparse basis  $\Psi$  using Algorithm 1
4 collect data  $Y$  in the network using Algorithm 3
5  $A \leftarrow \Psi * \Phi$ 
6 initialization residual error  $r_0 = y$ ,  $\Lambda_0 = \emptyset$ ,  $A_0 = \emptyset$ ,  $it = 1$ 
7 compute  $\langle r_{i-1}, a_j' \rangle$ , select the largest  $la$  values from  $\langle r_{i-1}, a_j' \rangle$ ; these values correspond to  $A$ 's column indexes  $j$ , constructing set  $J_0$ 
8 set  $\Lambda_{it} = \Lambda_{it-1} \cup J_0$ ,  $A_{it} = A_{it-1} \cup a_j$  (for all  $j \in J_0$ )
9  $\hat{x}_{it} = (A_{it}^T A_{it})^{-1} A_{it}^T y$ 
10 update  $r_{it} = y - A_{it} \hat{x}_{it}$ 
11  $it = it + 1$ , if  $it \leq K$  go back to step 7, or proceed to step 12
12 reconstruct  $\hat{x}$ , which is the generation value of the last iteration  $\hat{x}_{it}$ .

```

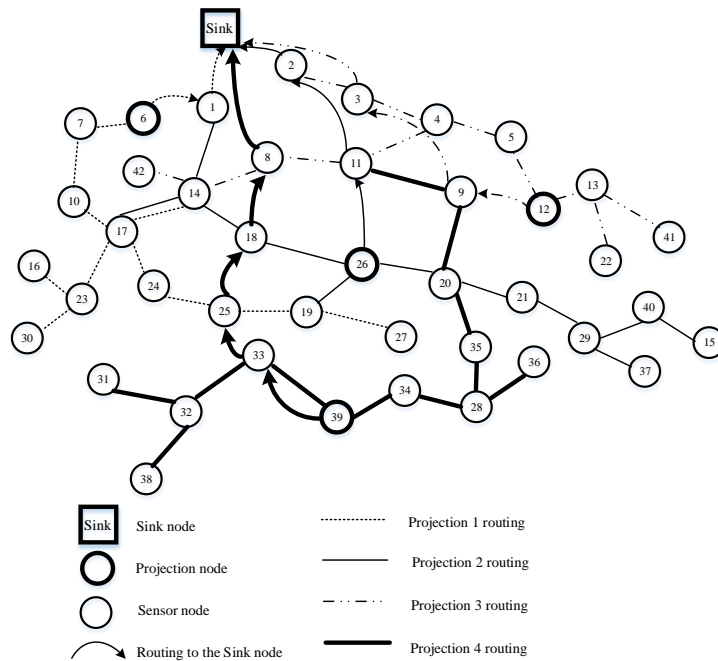


Figure 10. Our proposed algorithm, where $N = 42$ and $M = 4$.

5. Theoretical Analysis

In this section, we prove that the equivalent sensing matrix A follows the RIP property. We first propose some definitions and corollaries.

Definition 2. (Sub-Gaussian [41]) A random variable \wp is called sub-Gaussian if there exists a constant $c > 0$ such that

$$E(\exp(\wp t')) \leq \exp(c^2 t'^2 / 2) \quad (19)$$

holds for all $t' \in \mathbb{R}$. We use the notation $\wp \sim \text{Sub}(c^2)$ to denote that \wp satisfies Equation (19).

Theorem 1. ([41]) Suppose $\wp_{\partial} = [\wp_{\partial 1}, \wp_{\partial 2}, \dots, \wp_{\partial M}]'$, where each $\wp_{\partial i}$ is i.i.d. $\wp_{\partial i} \sim \text{Sub}(c^2)$ and $E(\wp_{\partial i}^2) = \sigma^2$. Then

$$E(\|\wp_{\partial i}\|_2^2) = M\sigma^2 \quad (20)$$

Furthermore, for any $\alpha' \in (0, 1)$, $\beta' \in [c^2/\sigma^2, \beta_{\max}]$, there exists a constant c^* such that

$$P(\|\wp\|_2^2 \leq \alpha' M\sigma^2) \leq \exp(-M(1 - \alpha')^2 / c^*) \quad (21)$$

and

$$P(\|\wp\|_2^2 \leq \beta' M\sigma^2) \leq \exp(-M(\beta' - 1)^2 / c^*) \quad (22)$$

Lemma 1. Fix $\delta \in (0, 1)$, $A = \Psi\Phi$ satisfies $(1 - \delta_K)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta_K)\|x\|_2^2$ for all N -dimensional K -sparse signal x .

Proof. As mentioned above, diffusion operator O generated by Λ , while O has the same eigenvalues as Λ (less than 1). The modified diffusion wavelets are the concatenation of the scale functions and wavelet functions. Hence, we believe that the entries of the representative basis Ψ are randomly sequenced, which is denoted by $\wp_1, \wp_2, \wp_3, \dots, \wp_N$. Moreover, since the non-zero entries for each row of measurement matrix Φ are mutually independent, the same probability is chosen for the projection nodes. Therefore, each row of A independently selects elements at random from Ψ . Alternatively, we suppose that A is generated by independent and identically distributed (i.i.d.) random variables $\wp_{\partial 1}, \wp_{\partial 2}, \wp_{\partial 3}, \dots, \wp_{\partial M}$.

The next step is normalization, $A = \sqrt{\frac{N}{M}}[\Theta_1, \Theta_2, \dots, \Theta_M]^T$; in addition, we obtain Equations (23) and (24), which follow a similar idea as [18]:

$$E(\Theta^T(i, j)^2) = \left(\sqrt{\frac{N}{M}}\right)^2 E(\Theta(j, i)^2) = \frac{1}{M} \quad (23)$$

$$E(\Theta(i, j)^T) = \sqrt{\frac{N}{M}} E(\Theta(j, i)) = 0. \quad (24)$$

Accordingly, we yield

$$\begin{aligned} & E(\|Y\|_2^2) \\ &= E\left(\sum_{i=1}^M \left(\langle \sqrt{\frac{N}{M}} \Theta_i, x \rangle\right)^2\right) \\ &= \sum_{i=1}^M E\left(\sum_{j=1}^N (\Theta^T(i, j)x_j)^2\right) \\ &= \sum_{i=1}^M (E(\Theta^T(i, j)^2))\|x\|_2^2 + 2 \sum_{j=1}^N \sum_{\kappa \neq j} E(\Theta^T(i, j))E(\Theta^T(i, \kappa)x_j x_{\kappa}) \end{aligned} \quad (25)$$

Here, we bring Equations (23) and (24) into Equation (25), and Equation (25) can be expressed as Equation (26):

$$E(\|Y\|_2^2) = \|x\|_2^2. \quad (26)$$

Alternatively, we set $\alpha' = 1 - \delta$ and $\beta' = 1 + \delta$. Hence, Equations (21) and (22) can be expressed as follows:

$$P(\|Y\|_2^2 \leq (1 - \delta)E(\|Y\|_2^2)) \leq \exp(-M\delta^2/c^*) \quad (27)$$

$$P(\|\Phi x\|_2^2 \leq (1 - \delta)\|x\|_2^2) \leq \exp(-M\delta^2/c^*) \quad (28)$$

and

$$P(\|\Phi x\|_2^2 \geq (1 + \delta)\|x\|_2^2) \leq \exp(-M\delta^2/c^*). \quad (29)$$

However, there are (N, K) possible K -dimensional subspaces of A relying on Sterling's approximation. So, the inequality is given as $(N, K) \leq (eN/K)^K$. Thus, the probability of signal x needs to obey the conditions of Equation (30):

$$(eN/K)^K \cdot 2e^{(-M\delta^2/c^*)} = 2e^{(-M\delta^2/c^* + K\log(N/K) + 1)}, \quad (30)$$

such that

$$(1 - \delta_K)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta_K)\|x\|_2^2. \quad (31)$$

Finally, $M = O(K\log(N/K))$ is selected to follow the RIP property with the probability approximation to 1, which completes the proof.

6. Simulation Results

In the following section, we evaluate the performance of our scheme by experiments. We choose the dataset from DEI [30], described in Section 4.1 and synthetic data. We evaluate our scheme mainly in terms of the sparse basis comparison; the reconstruction performance of the novel mechanism; the reconstruction error for different schemes; the energy consumption based on non-CS, plain CS, hybrid CS and our proposed algorithm (sparse basis is based on distance); and network lifetime performance between the different schemes and our algorithms. In our simulations, all programs have been run in the Matlab platform. Moreover, $E_{elec} = 50$ nJ/bit, $E_{amp} = 60$ pJ/bit/m⁴, $E_{fs} = 100$ pJ/bit/m², $L = 1024$ bits, initial energy $E_0 = 5$ J. Figure 11 indicates the temperature signal from sensor nodes 1, 2, 3, and 4 in the detected area, and the frame length is 781. It is obvious that the signals have high spatial correlations, which is also demonstrated in Section 3.1.

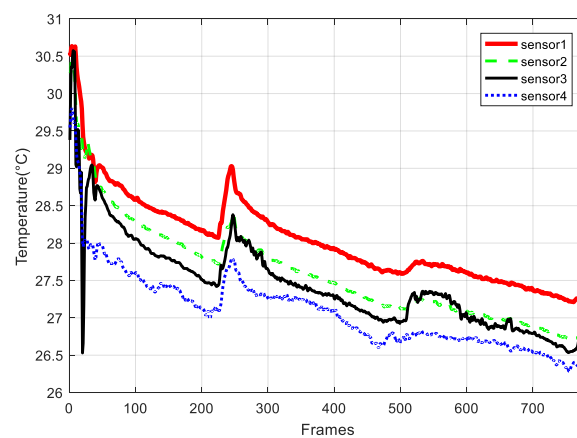


Figure 11. Original signal with different sensor nodes from DEI datasets.

6.1. Sparse Comparison

We compare the DFT sparse basis, diffusion wavelets based on degree (the first scheme), and diffusion wavelets based on distance (the second scheme) presented in Section 4.2. However, since there are not enough real data, we choose synthetic datasets [42] to accomplish the following simulations. Thus, Figure 12 is plotted in Matlab software. In Figure 12a denotes DFT coefficients, Figure 12b is the diffusion wavelet coefficients based on degree, and Figure 12c represents diffusion wavelet coefficients based on distance. Here, we select 650 frame length data. As can be seen from Figure 12, DFT does not sparsify the sensor signal, and the value of most of the coefficients is approximately 0.1 instead of 0. However, Figure 12b,c can all sparsify the sensory data, while the energy of the latter is more concentrated without interference signal compared with the former. On the whole, diffusion wavelets based on distance represent better performance because they exploit the spatial correlation features among sensor nodes of the networks.

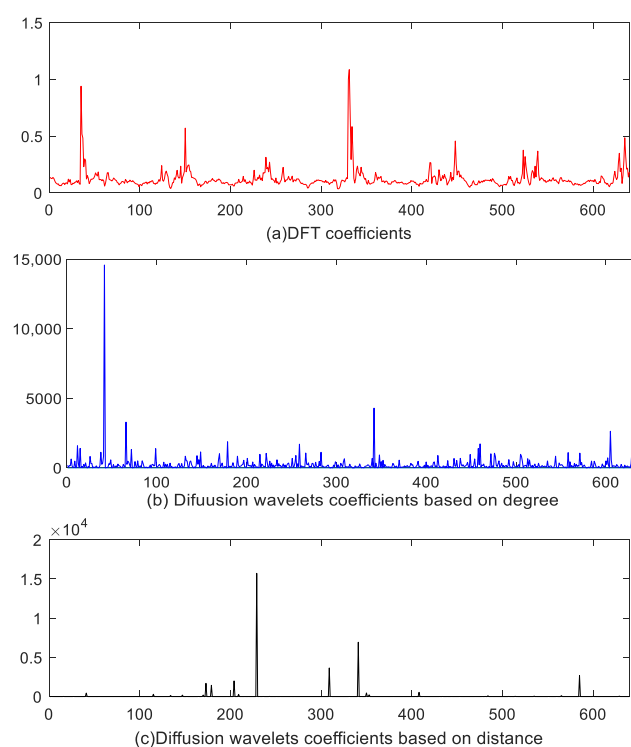


Figure 12. Comparison of different represent basis (a) DFT coefficients (b) coefficients based on degree (c) coefficients based on distance.

6.2. Performance of Reconstruction Signal

We implement our experiment using reconstruction algorithm gOMP. In our experiment, we choose modified diffusion wavelets based on distance and sparse matrix as our represent basis and measurement matrix, respectively. In order to show the recovery quality of the proposed algorithm, we plot Figure 13, which illustrates the error of the reconstruction signal decoding at the sink node. In our experiment, we define the reconstruction error as in Equation (32):

$$\varepsilon_1 = \|\mathbf{x} - \hat{\mathbf{x}}\|_2 \cdot \|\mathbf{x}\|_2^{-1}. \quad (32)$$

We select 750 humidity readings, which are described in Figure 13a. Then, Figure 13b shows the recovery performance of sensor node readings with 256 CS measurements. This also demonstrates that 256 CS measurements can reconstruct the original signal with a relative recovery error of $\varepsilon_1 = 0.0306$.

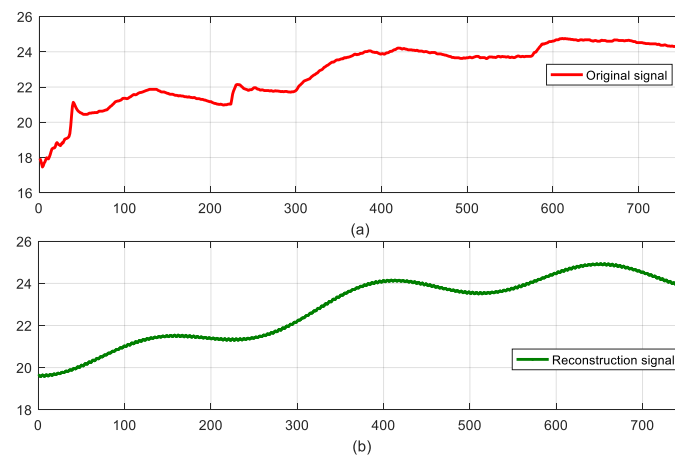


Figure 13. Comparison between original signal and reconstruction signal (a) original signal (b) reconstruction signal.

6.3. Reconstruction Error for Different Schemes

Figure 14 compares the reconstruction error for DFT, diffusion wavelets based on degree, and diffusion wavelets based on distance. Here, we again extract the sensory data [42]. In Figure 14, the reconstruction error is high under DFT sparse basis, where the error is up to 0.34 when the number of CS measurements is 20. However, the recovery error is less than 0.1 when the number of CS measurements is 220. In other words, the method does not accurately recover the signal. The blue curve with triangles in Figure 14 indicates the reconstruction performance of diffusion wavelets based on degree. It is noted that the scheme can sparsity the signal, and can recover the original signal with smaller reconstruction error than the DFT at the sink node. The reconstruction error changes steadily along with the increase of the number of CS measurements. The reason is that the approach only considers some featured such as the network topology, namely the number of neighbor nodes for a given node, without capturing the geographical position features of the neighbor nodes. Accordingly, the diffusion wavelets based on distance take advantage of the favorable conditions to promote recovery, as is shown in the red star curve in Figure 14.

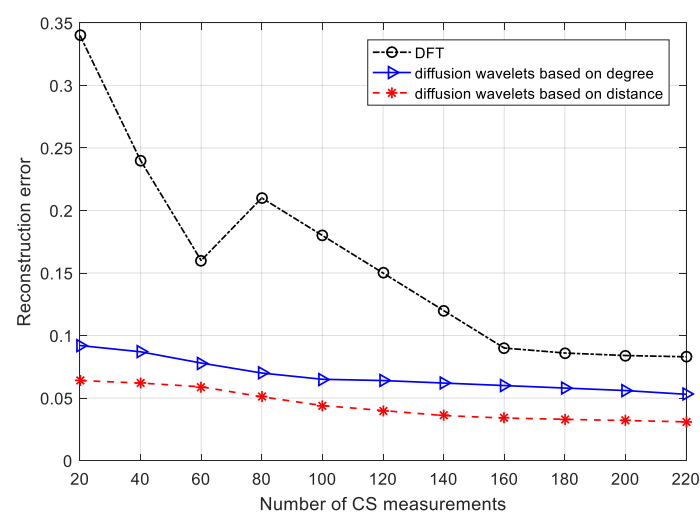


Figure 14. Reconstruction error vs. number of CS measurements.

6.4. Energy Consumption Evaluation

To illustrate the efficiency of our proposed data-gathering technique, we compare the results obtained from the non-CS, plain CS, and hybrid CS data-gathering schemes for data to the sink node using *Dijkstra* algorithm and our proposed algorithm, where measurement $M \in [30\ 220]$ and the frame length of data is 750. The number of sensor nodes is 80. From Figure 15, it can be seen that non-CS consumes the most energy, about 3.91×10^3 J, and the value is unchanged as the number of CS measurements steadily increases because the scheme does not adopt the CS technique. In addition, hybrid CS consumes about 1.22×10^3 J compared with 1.51×10^3 J for plain CS when the number of CS measurements is 50. Both display an upward trend with the rise in the number of CS measurements, but the energy consumption of hybrid CS is always less than that of plain CS. The difference is about 0.46×10^3 J when the number of CS measurements is at its maximum, 220. The reason is that the hybrid CS scheme uses the CS mechanism or not based on the number of transmission packets. Therefore, hybrid CS takes advantage of non-CS and plain CS to reduce the energy consumption of the network. However, the data-gathering scheme using the *Dijkstra* algorithm consumes less energy than hybrid CS because the former makes full use of our modified diffusion wavelets and sparse random matrix. When the number of measurements is about 110, the gap between hybrid CS and *Dijkstra* reaches a maximum; when the number of CS measurements is 220, the difference is minimal. Our proposed data-gathering scheme using diffusion wavelets based on distance and a modified ant colony algorithm shows better performance compared with other methods. It is obvious that it considers a sensor node's distance to exploit spatial correlation and sensor nodes' residual energy and path length are jointly taken into consideration. Thus, the scheme consumes less energy. When the number of CS measurements achieves the maximum (220), the advantage of this approach is more prominent: the transmission cost is about 2.18×10^3 J. This is far less than for the other four schemes in terms of energy consumption. From Figures 14 and 15, we can observe that although the recovery error has become smaller, the transmission cost has increased a lot. Therefore, there is a trade-off between energy consumption and recovery error.

Moreover, to further evaluate the performance of the proposed algorithm, Figure 16 is plotted, where energy consumption comparisons are shown among different data-gathering schemes with a change in the number of sensor nodes. It can be seen from Figure 16 that the non-CS technique always consumes the most energy as the number of sensor nodes increases from 100 to 800 because it does not use any compressive methods. However, plain CS dramatically reduces the network energy consumption compared with the non-CS. The reason is that it takes advantage of compressive sensing. Obviously, the energy consumption of hybrid CS decreases compared with plain CS. The number of sensor nodes is fewer than 400 and the difference between plain CS and hybrid CS is small; when the number of sensor nodes is 800, the gap between plain CS and hybrid CS is largest. That is to say, the energy efficiency of hybrid CS is more significant with an increase in the number of sensor nodes. Additionally, the transmission costs of the *Dijkstra* algorithm are lowered because of the sparse measurement matrix, as is shown in Figure 16. The performance of our proposed algorithm is better than all of the abovementioned schemes. Especially when the number of sensor nodes is greater than 500, the performance of our algorithm is more significant. This is why sparse represent basis is used and our improved ant colony algorithm is presented in the process of data gathering.

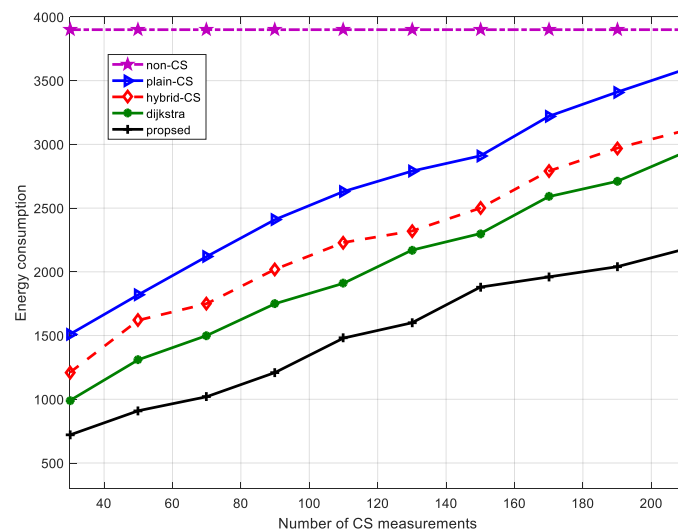


Figure 15. Energy consumption comparison of different data-gathering schemes with a change in the number of CS measurements.

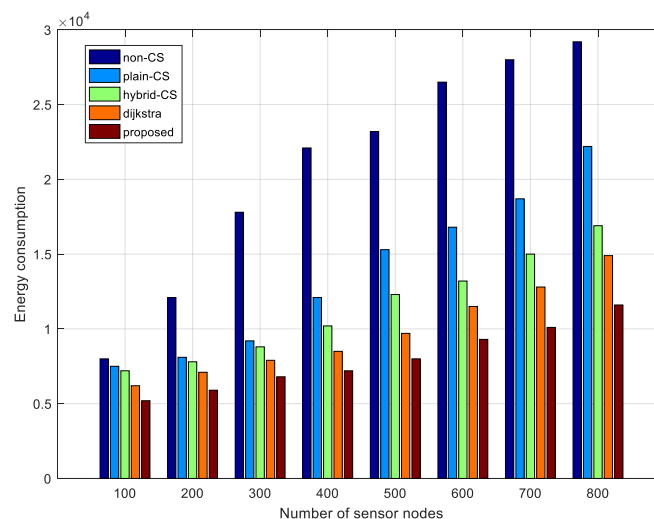


Figure 16. Energy consumption comparison of different data-gathering schemes vs. the number of sensor nodes.

6.5. Network Lifetime Performance

To complete the evaluation of the network lifetime performance, in this subsection we suppose that the time of the first dead node corresponds to the network lifetime. Figure 17 represents the relationship between the network lifetime and the number of CS measurements among non-CS, plain CS, hybrid CS, *Dijkstra*, and our proposed algorithm, while Figure 18 demonstrates the relationship between the network lifetime and the number of sensor nodes based on the different CS schemes and our algorithm. In Figure 17, we see that the network lifetime of the three data-gathering schemes non-CS, plain CS, and hybrid CS is short, and the number of CS measurements varies from 30 to 220. However, the network lifetime of *Dijkstra* and our algorithm is longer than in the above three schemes. The main reason is that the two methods also utilize a sparse measurement matrix. However, the advantage of our proposed algorithm is that the projection sensor nodes select the optimal routing to the sink node, aiming to lessen the transport load of the nodes nearest the sink node. Furthermore, the improved ant colony algorithm considers the residual energy of the sensor nodes, guaranteeing the load balance of the whole network, as is observed in Figure 17. We plot Figure 18 comparing the network lifetime of

the different schemes with a change in the number of sensor nodes. It is obvious that our proposed algorithm is better than the other techniques. In Figure 18, we see that the red bar denotes the network lifetime of our algorithm, which is greater than that found with the other algorithms. When the number of sensor nodes is 800, in terms of the network lifetime, non-CS is about 200, plain CS is about 500, hybrid CS is about 620, *Dijkstra* is about 850 and our proposed algorithm is about 2600, respectively. Even though the number of sensor nodes is small, ~100–200, the network lifetime of the other schemes is far shorter than that of our proposed algorithm.

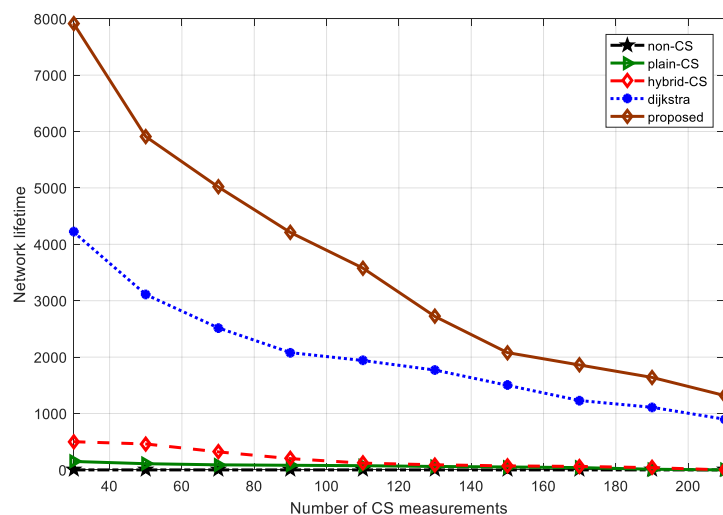


Figure 17. Network lifetime comparison of different data-gathering schemes vs. number of CS measurements.

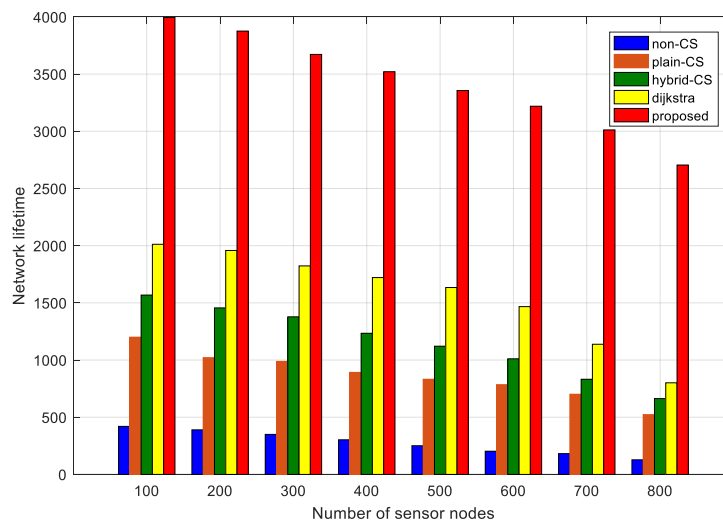


Figure 18. Network lifetime comparison of different data-gathering schemes vs. number of sensor nodes.

7. Conclusions and Future Work

Conventional CS data-gathering schemes design the sparse represent basis such as DCT, DFT, and wavelets transform do not consider the network topology and sensor nodes' spatial information. Therefore, in our mechanism, diffusion wavelets based on sensor nodes' degree and different nodes' distance considering the above factors are proposed. Additionally, to further reduce the transport costs in WSNs, a sparse measurement matrix is utilized and MST and modified ant colony routing are jointly applied to mitigate energy consumption and balance the network load, especially lowering

the transmission costs for those nodes nearest the sink node. Experimental results have shown that our sparse basis can sparsify the signal well. Our methods can also accurately reconstruct the original signal. Moreover, the reconstruction error of our scheme is less than DFT. Compared with existing data-gathering approaches, our proposed algorithm not only minimizes the energy consumption of the network, but prolongs the network lifetime. Furthermore, our algorithm has a theoretical guarantee of recovering the original signal, and the sink node needs to gather $M = O(K \log(M/K))$ measurements so as to recover the original signal.

Future work should consider the following aspects: On the one hand, sensor node readings have not only spatial correlation, but temporal correlation, so our future work will extend the spatial-temporal field. On the other hand, [43] demonstrates that optimization projection will generate better recovery performance than random projection, so a possible extension to this work will consider how to design the optimization projection so as to reduce the energy consumption.

Acknowledgments: This work was supported by the National Natural Science Foundation of China (51504214, 61771417, 51504255), the Fundamental Research and Development Foundation of Jiangsu Province (BE2105040), the Natural Science Foundation of Jiangsu Province of China (BK20150204), the National Key Research and Development Program (2016YFC0801403), and a China Postdoctoral Science Foundation Grant (2015M581884).

Author Contributions: Xiangping Gu designed the algorithm, performed the theoretical analysis, and wrote the entire manuscript. Xiaofeng Zhou implemented the simulation and contributed to the manuscript preparation. Yanjing Sun contributed to polishing the revised manuscript and provided suggestions on simulation evaluation.

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

References

1. Anastasi, G.; Conti, M.; Francesco, M.D.; Passarella, A. Energy conservation in wireless sensor networks: A Survey. *Ad Hoc Netw.* **2009**, *7*, 537–568. [\[CrossRef\]](#)
2. Hashemi, M.; Si, W.; Laifenfeld, M.; Starobinski, D.; Trachtenberg, A. Intra-car multihop wireless sensor networking: A case study. *IEEE Commun. Mag.* **2014**, *52*, 182–191. [\[CrossRef\]](#)
3. Cheng, C.T.; Chi, K.T.; Lau, F.C.M. An energy-aware scheduling scheme for wireless sensor networks. *IEEE Trans. Veh. Technol.* **2010**, *59*, 3427–3444. [\[CrossRef\]](#)
4. Aziz, A.A.; Sekercioglu, Y.A.; Fitzpatrick, P.; Ivanovich, M. A survey on distributed topology control techniques for extending the lifetime of battery powered wireless sensor networks. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 121–144. [\[CrossRef\]](#)
5. Wu, P.; Xiao, F.; Sha, C.; Huang, H.; Wang, R.; Xiong, N. Node scheduling strategies for achieving full-view area coverage in camera sensor networks. *Sensors* **2017**, *17*, 1303. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Zhao, M.; Ma, M.; Yang, Y. Efficient data gathering with mobile collectors and space-division multiple access technique in wireless sensor networks. *IEEE Trans. Comput.* **2011**, *60*, 400–417. [\[CrossRef\]](#)
7. Li, X.; Tao, X.; Mao, G. Unbalanced expander based compressive data gathering in clustered wireless sensor networks. *IEEE Access* **2017**, *5*, 7553–7566. [\[CrossRef\]](#)
8. Wang, Y.; Yang, Z.; Li, F.; Wen, H.; Shen, Y. CS²-collector: A new approach for data collection in wireless sensor networks based on two-dimensional compressive sensing. *Sensors* **2016**, *16*, 1318. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Candes, E.J.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509. [\[CrossRef\]](#)
10. Donoho, D. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [\[CrossRef\]](#)
11. Baraniuk, R. Compressive sensing. *IEEE Signal Proc. Mag.* **2007**, *24*, 118–121. [\[CrossRef\]](#)
12. Luo, C.; Wu, F.; Sun, J.; Chen, C.W. Efficient measurement generation and pervasive sparsity for compressive data gathering. *IEEE Trans. Wirel. Commun.* **2010**, *9*, 3728–3738. [\[CrossRef\]](#)
13. Talari, A.; Rahnavard, N. CStorage: Decentralized compressive data storage in wireless sensor networks. *Ad Hoc Netw.* **2016**, *37*, 475–485. [\[CrossRef\]](#)
14. Wu, X.G.; Xiong, Y.; Wan, S.; Huang, W. Sparsest random scheduling for compressive data gathering in wireless sensor networks. *IEEE Trans. Wirel. Commun.* **2014**, *13*, 5867–5877. [\[CrossRef\]](#)
15. Quer, G.; Masiero, R.; Pillonetto, G.; Rossi, M.; Zorzi, M. Sensing, compression, and recovery for wsns: Sparse signal modeling and monitoring framework. *IEEE Trans. Wirel. Commun.* **2012**, *11*, 3447–3461. [\[CrossRef\]](#)

16. Salim, A.; Osamy, W. Distributed multi chain compressive sensing based routing algorithm for wireless sensor networks. *Wirel. Netw.* **2015**, *21*, 1379–1390. [[CrossRef](#)]
17. Wang, X.; Zhao, Z.F.; Xia, Y. Compressed sensing for efficient random routing in multi-hop wireless sensor networks. *Int. J. Commun. Netw. Distrib. Syst.* **2011**, *7*, 275–292. [[CrossRef](#)]
18. Zheng, H.F.; Yang, F.; Tian, X.; Gan, X.; Wang, X. Data Gathering with compressive sensing in wireless sensor networks: A random walk based approach. *IEEE Trans. Parallel Distrib. Syst.* **2015**, *26*, 35–44. [[CrossRef](#)]
19. Xie, R.T.; Jia, X.H. Transmission-efficient clustering method for wireless sensor networks using compressive sensing. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *5*, 806–815.
20. Bajwa, W.; Haupt, J.; Sayeed, A.; Nowak, R. Compressive wireless sensing. In Proceedings of the 5th International Conference on Information Processing in Sensor Networks (IPSN), Nashville, TN, USA, 19–21 April 2006; pp. 134–142.
21. Luo, J.; Xiang, L.; Rosenberg, C. Does compressed sensing improve the throughput of wireless sensor networks. In Proceedings of the IEEE International Conference on Communications, Cape Town, South Africa, 23–27 May 2010; pp. 1–6.
22. Quer, G.; Masiero, R.; Munaretto, D.; Rossi, M. On the interplay between routing and signal representation for compressive sensing in wireless sensor networks. *Inf. Theory Appl. Workshop* **2009**, *10*, 206–215.
23. Wang, W.; Garofalakis, M.; Ramchandran, K. Distributed sparse random projections for refinable approximation. In Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN), Cambridge, MA, USA, 25–27 April 2007; pp. 331–339.
24. Candès, E.; Romberg, J.; Tao, T. Decoding by linear programming. *IEEE Trans. Inf. Theory* **2005**, *51*, 4203–4215. [[CrossRef](#)]
25. Tropp, J.A.; Gilbert, A.C. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inf. Theory* **2007**, *53*, 4655–4666. [[CrossRef](#)]
26. Needell, D.; Tropp, J.A. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmon. Anal.* **2009**, *26*, 301–321. [[CrossRef](#)]
27. Donoho, D.L.; Tsai, Y.; Drori, I.; Starck, J.L. Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit. *IEEE Trans. Inf. Theory* **2012**, *58*, 1094–1121. [[CrossRef](#)]
28. Wang, J.; Kwon, W.; Shim, B. Generalized orthogonal matching pursuit. *IEEE Trans. Signal Process.* **2012**, *60*, 6202–6216. [[CrossRef](#)]
29. Heinzelman, W.; Chandrakas, A.; Balakrishnan, H. Energy-efficient communication protocol for wireless sensor networks. In Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, 4–7 January 2000; pp. 3005–3014.
30. Thomas, M.C. *Elements of Information Theory*, 2nd ed.; Machinery Industry Press: Beijing, China, 2008; pp. 7–12.
31. Casari, P.; Castellani, A.P.; Cenedese, A.; Lora, C.; Rossi, M.; Schenato, L.; Zorzi, M. The “wireless sensor networks for city-wide ambient intelligence (WISE-WAI)” project. *Sensors* **2009**, *9*, 4056–4082. [[CrossRef](#)] [[PubMed](#)]
32. IntelLab. Available online: <http://www.select.cs.cmu.edu/data/labapp3/index.html> (accessed on 12 September 2016).
33. EPFL LUCE SensorScope WSN. Available online: <http://sensorscope.epfl.ch/> (accessed on 16 October 2017).
34. CitySense. Available online: <http://www.citysense.net> (accessed on 18 October 2017).
35. Watteyne, T.; Barthel, D.; Dohler, M.; Augeblum, I. Sense and sensitivity: A large-scale experimental study of reactive gradient routing. *Meas. Sci. Technol.* **2010**, *21*, 124001–124009. [[CrossRef](#)]
36. Coifman, R.; Maggioni, M. Diffusion wavelets. *Appl. Comput. Harmon. Anal.* **2006**, *21*, 53–94. [[CrossRef](#)]
37. Lv, C.; Wang, Q.; Yan, W.; Zhao, R. A sparse representation method of 2-D sensory data in wireless sensor networks. In Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Taipei, Taiwan, 23–26 May 2016; pp. 1–6.
38. Chung, F. Spectral Graph Theory. In *CBMS Regional Conference Series in Mathematics*; American Mathematical Society: Providence, RI, USA, 1997; pp. 1–21.
39. Dorigo, M.; Maniezzo, V.; Colomi, A. Ant system: Optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man Cybern. Part B* **1996**, *26*, 29–41. [[CrossRef](#)] [[PubMed](#)]
40. Dijkstra, E.W. A note on two problems in connexion with graphs. *Numer. Math.* **1959**, *1*, 269–271. [[CrossRef](#)]

41. Davenport, M.A. Random Observation on Random Observations: Sparse Signal Acquisition and Processing. Ph.D. Thesis, Rice University, Houston, TX, USA, 2010.
42. Zordan, D.; Quer, G.; Zorzi, M.; Rossi, M. Modeling and Generation of Space-Time Correlated Signals for Sensor Network Fields. In Proceedings of the Global Telecommunications Conference, Houston, TX, USA, 5–9 December 2011; pp. 1–6.
43. Elad, M. Optimized projection for compressed sensing. *IEEE Trans. Signal Process.* **2007**, *55*, 5695–5702. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).