sensors

MDPI

Article

A Practical Data-Gathering Algorithm for Lossy Wireless Sensor Networks Employing Distributed Data Storage and Compressive Sensing

Ce Zhang, Ou Li, Guangyi Liu * and Mingxuan Li

National Digital Switching System Engineering and Technological R&D Center, Zhengzhou 450002, China; cezhang@foxmail.com (C.Z.); zzliou@126.com (O.L.); seu_lmx@foxmail.com (M.L.)

* Correspondence: liuguangyi1982@163.com

Received: 8 August 2018; Accepted: 20 September 2018; Published: 24 September 2018

Abstract: Reliability and energy efficiency are two key considerations when designing a compressive sensing (CS)-based data-gathering scheme. Most researchers assume there is no packets loss, thus, they focus only on reducing the energy consumption in wireless sensor networks (WSNs) while setting reliability concerns aside. To balance the performance–energy trade-off in lossy WSNs, a distributed data storage (DDS) and gathering scheme based on CS (CS-DDSG) is introduced, which combines CS and DDS. CS-DDSG utilizes broadcast properties to resist the impact of packet loss rates. Neighboring nodes receive packets with process constraints imposed to decrease the volume of both transmissions and receptions. The mobile sink randomly queries nodes and constructs a measurement matrix based on received data with the purpose of avoiding measuring the lossy nodes. Additionally, we demonstrate how this measurement matrix satisfies the restricted isometry property. To analyze the efficiency of the proposed scheme, an expression that reflects the total number of transmissions and receptions is formulated via random geometric graph theory. Simulation results indicate that our scheme achieves high precision for unreliable links and reduces the number of transmissions, receptions and fusions. Thus, our proposed CS-DDSG approach effectively balances energy consumption and reconstruction accuracy.

Keywords: WSNs; CS; distributed data storage; packet loss rate; energy efficiency

1. Introduction

As the perceptual layer of the Internet of Things (IoT) [1,2], wireless sensor networks (WSNs) [3] are widely deployed for purposes such as environment monitoring [4], industry automation [5] and military reconnaissance [6]. WSNs consist of many sensors and play a key role in sensing and gathering data from the surrounding environment. Because of harsh environments and energy-limited nodes, there are two key considerations in WSNs design: reliability and energy efficiency. In addition, nodes that are closer to the sink require more forwarding tasks than others, resulting in higher energy consumption as well as a reduction in the lifetime of the entire network.

Compressive sensing (CS) theory [7,8] provides a new method for reducing communication energy consumption. CS points out that, for the compressible signals in WSNs, a small collection of linear projections is sufficient to achieve near-perfect reconstruction, which reduces energy consumption and prolongs network lifetime. Thus, a considerable amount of research has been conducted concerning ways to utilize CS to gather data in WSNs. The CS-based data-gathering schemes in [9–11] obtained the member node readings utilizing fixed routing, in which ordinary nodes forward compressed data to the static sink node through multi-hops. Lou et al. [9] and Lou et al. [10] combined CS and routing protocols to reduce the number of transmissions. In [11–15], the use of sparse measurement

matrix is investigated to reduce the number of nodes involved in data gathering. Introducing CS effectively reduces the energy required for communication and distributes energy consumption loads more evenly. However, if a parent node (which holds a combination of child node readings) loses its packet, then all the information from the child nodes is also lost. Hence, unreliable links have a serious impact on data gathering and make it difficult to reliable gather data reliably through a centralized sink node. Additionally, Kong et al. [16] reported that unreliable links are widespread in WSNs, where the average packet loss rate is 40–50%. Thus, assuming completely reliable links is unfeasible and oversimplifies the problem.

To resolve this problem, distributed data storage (DDS) [17–19] is proposed to enable reliable data gathering by employing redundancy. In contrast to a centralized sink, a mobile sink collects data from a small subset of the total nodes to recover all the data. It is worth mentioning that DDS effectively reduces the impact of packet loss on data gathering because there is no static routing, although few researchers have focused on this advantage. However, DDS requires a large number of transmission tasks to ensure sufficient redundancy, which is potentially catastrophic for nodes with energy limitations. Thus, it is imperative to investigate effective ways to apply DDS for data gathering with the dual purposes of resisting packet loss and reducing the number of transmissions.

To address this problem, many studies have been carried out on this topic. In [20–22], CS is combined with DDS to exploit the advantages of both technologies. The goal of Talari et al. [20] was to reduce the number of transmissions by exploiting the spatial correlations of nodes based on CS with the broadcast properties of wireless channels. In this scheme, the nodes store received data and broadcast the data with a given probability. The performance of data reconstruction was further improved in [21]. Yang et al. [21] found that the number of receptions was higher than the number of transmissions. Hence, Yang et al. [21] focused on reducing the total number of both transmissions and receptions simultaneously. In [22], both the spatial and temporal correlations of nodes are exploited to reduce the number of transmissions. All the above studies take advantage of broadcast routing and consider how to reduce the transmission energy cost. However, compared with fixed routing, such as tree routing and cluster routing, broadcasting data consumes more reception energy because neighboring nodes receive broadcast data whether they need it or not. For example, in [20–22], the neighboring nodes first receive the broadcasting data and then determine whether to merge the data based on certain conditions. Consequently, broadcasting data consumes large amount of reception energy, although the received data are rarely merged. Furthermore, none of these studies consider the problem of packet loss; instead, they make the unrealistic assumption that the wireless links are completely reliable.

Tackling the abovementioned consideration, two challenges must be resolved. The first involves how to effectively reduce the quantity of data disseminated (transmissions and receptions), especially the number of receptions rather than the number of fusions. The second problem is related to reducing the impact of lossy links (namely, the packet loss rate) on data reconstruction. To solve these two challenges, a distributed data storage and gathering algorithm based on compressive sensing (CS-DDSG) is proposed utilizing CS and DDS. Relying on collected data, the mobile sink generates a sparse measurement matrix aimed at reducing communication energy consumption. Furthermore, it is proven that the measurement matrix satisfies the restricted isometry property (RIP) [23]. Based on random geometric graph theory, an expression of the total number of transmissions and receptions is formulated to analyze the energy consumption of CS-DDSG.

The reminder of this paper is organized as follows. In Section 2, we commence by reviewing the CS theory and introduce the network model. In Section 3, we present the proposed CS-DDSG algorithm, describe the formulation of the measurement matrix and provide a proof that this matrix can satisfy RIP. Based on the proposed scheme, we formulate the expression of the total number of transmissions and receptions in Section 4. We present our simulations and their results and investigate the performance of CS-DDSC in Section 5. Finally, concluding remarks are provided in Section 6.

2. Preliminaries and Network Model

In this section, we introduce CS theory and then describe the network model and our motivation.

2.1. Compressed Sensing

In WSNs, assume that N sensor readings are denoted by $\mathbf{X} = (x_1, \cdots, x_N)^{\mathrm{T}}$, where $x_i, i \in [1, N]$ denotes the reading of node i with K-sparse representation at a basis $\mathbf{\Psi} \in \mathbb{R}^{N \times N}$:

$$\mathbf{X} = \mathbf{\Psi}\boldsymbol{\theta}, \tag{1}$$

where $\boldsymbol{\theta} \in \mathbb{R}^N$ is a coefficient vector corresponding to the sparse basis $\mathbf{\Psi}$. $\mathbf{X}$ is K-sparse and compressive if the vector $\boldsymbol{\theta}$ has at most $K(K \leq N)$ nonzero coefficients or $(N - K)$ smallest coefficients can be ignored.

We assume the measurement matrix is $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$ and is uncorrelated with the basis $\mathbf{\Psi}$, then the CS measurements of $\mathbf{X}$ can be expressed as follows:

$$\boldsymbol{Y} = \mathbf{\Phi}\boldsymbol{X} = \mathbf{\Phi}\mathbf{\Psi}\boldsymbol{\theta} = \mathbf{\Theta}\boldsymbol{\theta}, \tag{2}$$

where $M \ll N$ and $\mathbf{\Theta} = \mathbf{\Phi}\mathbf{\Psi}$ is a sensing matrix. The original signal $\boldsymbol{X}$ can be reconstructed with an overwhelming probability from M measurements by $\boldsymbol{l_1}$-norm minimization as follows:

$$\begin{aligned} &\min: \hat{\boldsymbol{X}} = \min\|\boldsymbol{X}\|_1 \\ &\text{s.t.:} \boldsymbol{Y} = \mathbf{\Phi}\boldsymbol{X}, \end{aligned} \tag{3}$$

where $\hat{\mathbf{X}}$ denotes the reconstructed sparse signal of $\boldsymbol{X}$.

To reconstruct $\mathbf{X}$, two factors must be considered: (1) $\mathbf{X}$ is compressive at $\mathbf{\Psi}$; and (2) $\mathbf{\Phi}$ must satisfy the RIP with $\boldsymbol{M} \geq \boldsymbol{ck}\lg(\boldsymbol{N}/\boldsymbol{k})$. Therefore, $\boldsymbol{K}$-sparse $\mathbf{X}$ satisfies the following condition:

$$(1 - \boldsymbol{\varepsilon})\|\boldsymbol{\theta}\|_2^2 \leq \|\mathbf{\Phi}\boldsymbol{\theta}\|_2^2 \leq (1 + \boldsymbol{\varepsilon})\|\boldsymbol{\theta}\|_2^2, \tag{4}$$

where $c, \boldsymbol{\varepsilon} \in (0,1)$, while $\mathbf{\Phi}$ satisfies RIP with the parameter $\boldsymbol{\varepsilon}$.

2.2. Network Model

We consider a single-sink WSN consisting of $\boldsymbol{N}$ battery-powered sensors. The sensors are deployed in a square area with a boundary length of 1. We assume all nodes have an identical transmission radius of $\boldsymbol{r_t}$, and that any two nodes can communicate with each other if their Euclidian distance $\boldsymbol{d}$ satisfies $\boldsymbol{d} \leq \boldsymbol{r_t}$. To guarantee the network connectivity, $\boldsymbol{r_t}$ should also satisfy the following condition [24]:

$$\boldsymbol{r_t^2} > \boldsymbol{S} \cdot \ln(\boldsymbol{N})/(\boldsymbol{\pi N}), \tag{5}$$

where $\boldsymbol{S}$ denotes the deployment area and $\boldsymbol{S} = \mathbf{1} \times \mathbf{1}$. Let $\boldsymbol{X_{N \times 1}} = (\boldsymbol{x_1}, \cdots, \boldsymbol{x_N})^{\boldsymbol{T}}$ denotes the $\boldsymbol{N}$ node readings. Since the readings are spatiotemporally correlative with each other, $\boldsymbol{X}$ can be compressed on an orthogonal basis $\mathbf{\Psi} = (\boldsymbol{\phi_{i,j}})_{N \times N}$. The fast Fourier transform (FFT) orthonormal basis is adopted as the sparse representation basis in this paper. Let $\mathbf{\Phi} = (\boldsymbol{\varphi_{i,j}})_{M \times N}$ denote the measurement matrix. The measurement vector $\boldsymbol{Y} \in \mathbb{R}^{\boldsymbol{M \times 1}}$ can be computed with Equation (2). Furthermore, we introduce the expression of $\mathbf{\Phi}$ in Section 3. Thus, the CS-DDSG network model coincides with the CS model.

In addition, we define the normalized mean absolute error (NMAE) metric to evaluate the accuracy of reconstruction accuracy:

$$\mathrm{NMAE} = \frac{\|\hat{\boldsymbol{X}} - \boldsymbol{X}\|_{\boldsymbol{2}}}{\|\boldsymbol{X}\|_{\boldsymbol{2}}} = \frac{\sqrt{\sum_{\boldsymbol{n=1}}^{\boldsymbol{N}} (\hat{\boldsymbol{x}}_{\boldsymbol{n}} - \boldsymbol{x}_{\boldsymbol{n}})^{\boldsymbol{2}}}}{\sqrt{\sum_{\boldsymbol{n=1}}^{\boldsymbol{N}} \boldsymbol{x}_{\boldsymbol{n}}{}^{\boldsymbol{2}}}}, \tag{6}$$

Equation (6) shows that the smaller the NMAE is, the better performance the algorithm can achieve.

2.3. Motivation

In this subsection, we investigate the impact of packet loss on the CS recovery performance relying on the fixed routing. Figure 1 presents the performance of the CDG [9] algorithm with cluster topology in unreliable links. In this scheme, there are 100 nodes and the member nodes forward the packets to the cluster head via a one-hop route. When the packet loss rate is 10%, the recovery accuracy is worse than the accuracy in the ideal link. Furthermore, increasing the measurements cannot improve the algorithm's performance. For $M = 50$ measurements, Figure 2 indicates that the accuracy declines with the increase of packet loss rate.

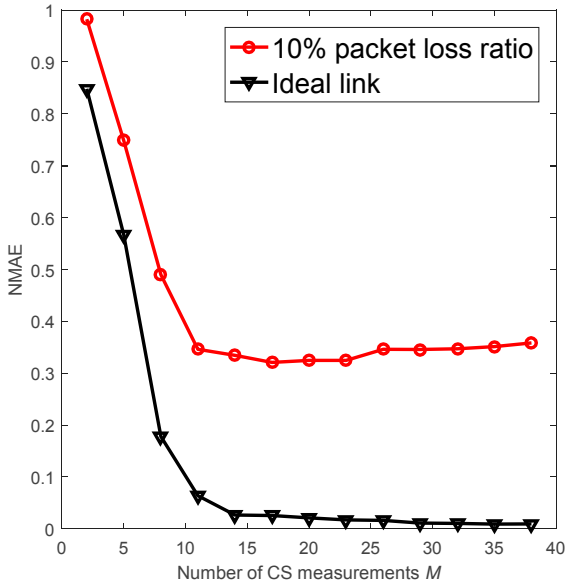

Figure 1. Performance of CDG with ideal link and lossy link.

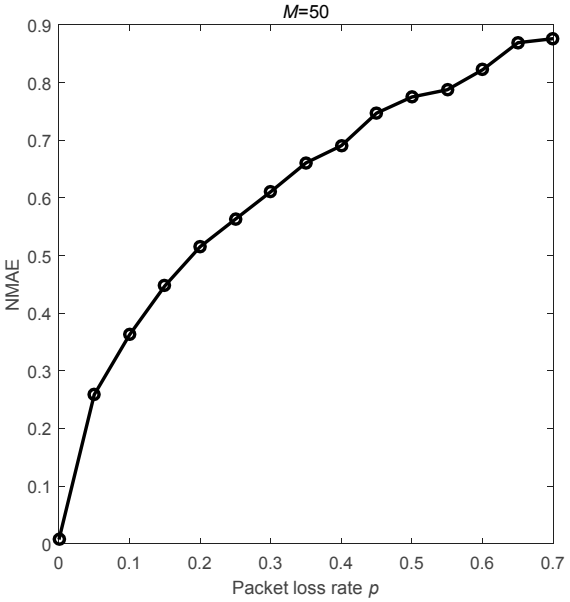

Figure 2. The relationship between the packet loss rate and the NMAE.

We consider one of the clusters containing N_1 nodes. For the CDG algorithm with fixed routing, the cluster head receives the data vector $\mathbf{X}_{N_1\times 1} = (x_1, \cdots, x_i, \cdots, x_{N_1})^T$ in reliable links. The measurements Y can be represented as

$$\mathbf{Y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{pmatrix} = \begin{pmatrix} \phi_{11} & \cdots & \phi_{1N_1} \\ \vdots & & \vdots \\ \phi_{M1} & \cdots & \phi_{MN_1} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_{N_1} \end{pmatrix}. \tag{7}$$

If the packet of node i is missing due to unreliable links, then its cluster head will receive $\mathbf{X}'_{N_1\times 1} = (x_1, \cdots, x'_i, \cdots, x_{N_1})^T$ and the measurement Y can be represented as

$$\mathbf{Y}' = \begin{pmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_M \end{pmatrix} = \begin{pmatrix} \phi_{11} & \cdots & \phi_{1N_1} \\ \vdots & & \vdots \\ \phi_{M1} & \cdots & \phi_{MN_1} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x'_i \\ \vdots \\ x_{N_1} \end{pmatrix}. \tag{8}$$

According to Equations (7) and (8), one missing packet affects every element y_i of the measurement vector. Thus, the sink recovers all the data X using Y' and Φ, which leads to an imprecise or invalid reconstruction. Furthermore, the accuracy is even worse under tree-based routing. This deficiency occurs because if one packet of a parent node is missing, then all the information from its child nodes is lost too. Additionally, simply increasing the number of measurements or the number of retransmissions does not help much in improving the recovery accuracy. Therefore, the CS-based algorithm is sensitive to packet loss. In the next section, we investigate how to resist unreliable links, while using fewer transmissions and receptions by utilizing broadcasting properties.

3. Proposed CS-DDSG Scheme

3.1. Procedures of CS-DDSG

Based on the network model, we propose CS-DDSG to avoid packet loss and reduce the total number of transmissions and receptions, as presented in Figure 3. The procedures involved in CS-DDSG are detailed below.

Stage 1. Initialization. The proposed scheme requires precise time to help nodes to cooperate with each other. Assuming the network is synchronized and slotted based on Reference Broadcast Synchronization (RBS) [25], which can achieve the goal of high accuracy and energy-efficiency. At the beginning of data gathering, each node senses a data x_i and generates a coefficient $\varphi_i = \mathbf{1}$. Then, each node i forms an initial packet, denoted by $S(i)$ which defines has two components:

$$S(i) = \begin{cases} S(i).\text{id} = [i] \\ S(i).\text{data} = x_i \end{cases}. \tag{9}$$

The component $S(i)$.id stores the node ID of nodes and $S(i)$.data stores the readings.

Stage 2. Broadcasting. After a fixed and long enough period of time for synchronization and initialization, N_s, $(N_s < N)$ nodes are randomly selected as source nodes with a probability p_1 in this stage. The source nodes broadcast their own packets and do not receive any packets. If an ordinary node $m(m \in [\mathbf{1}, N])$ is located with the communication range of the source node $n(n \in [\mathbf{1}, N])$ and

has not received a packet before, then node m receives the data broadcasted by node n and updates its packet as follows:

$$S(m) = \begin{cases} S(m).\text{id} = [m, n] \\ S(m).\text{data} = x_m + x_n \end{cases}. \quad (10)$$

If node m has already received any other broadcast data, then this node stops receiving data; in other words, each node receives only one broadcast packet.

Stage 3. Forwarding. In the following, only the receiving nodes from Stage 2 continue to broadcast their updated packets to neighboring nodes with the probability p_2. Similarly, the neighboring nodes around the forwarding nodes will receive a packet only if they have not received any prior packets. These new receiving nodes broadcast their updated packets as described above. Actually, the Stage 2 and Stage 3 could start simultaneously. Nodes get the packets of source nodes in Stage 2 and then decide whether to broadcast immediately. Thus, the neighboring nodes of those forwarding nodes could update their packets relying on the packets of source nodes or forwarding nodes. Finally, the forwarding operation will stop until there are no new reception nodes. Because of the reception condition and the small probability p_2, in practice, the forwarding process stops after repeating only a few times, which is analyzed in Section 5 in detail.

Stage 4. Visiting. The mobile sink starts the visiting phase after a fixed and sufficiently long period, which can be preset according to the number of nodes N. M nodes are randomly queried by the mobile sink to extract the corresponding information, i.e., the measurement vector Y and the measurement matrix Φ. Finally, the entire network's readings X can be reconstructed from Y and Φ based on Equation (3). The entire pseudocode of CS-DDSG is presented in Algorithms 1 and 2.

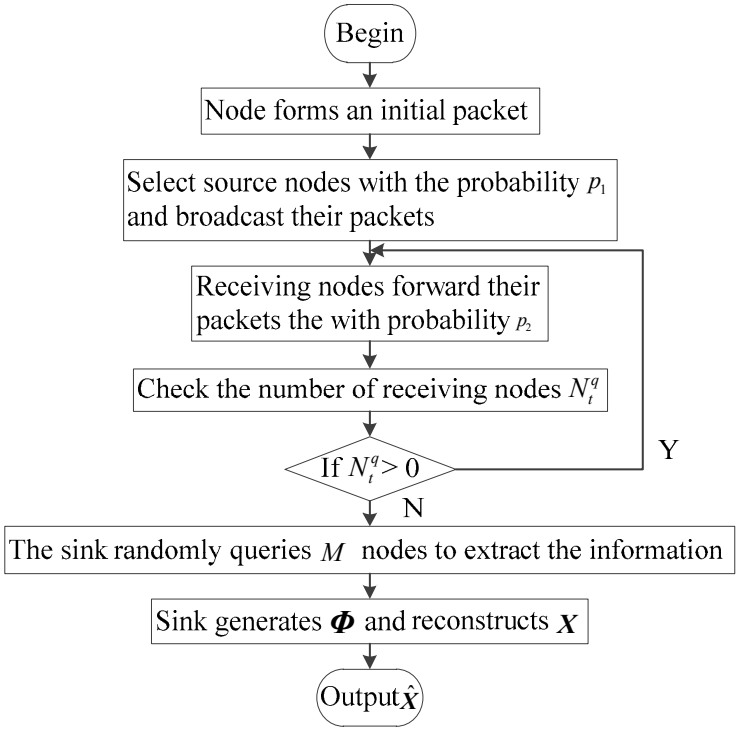

Figure 3. Flow chart of CS-DDSG.

Algorithm 1 The CS-DDSG algorithm

Input:
The probability of selecting source nodes: P_1;
The probability of forwarding: P_2;
The number of measurements: M;
Output:
Measurement vector: $\boldsymbol{y}$;
Measurement matrix: $\boldsymbol{\Phi}$;
Stage 1:
for i = 1:N
$S(i)$.id = [i];
$S(i).data = x_i$;
end for
Stage 2:
Nodes select themselves with the probability p_1 and broadcast their packets;
$N_2 = 0$;
for i = 1:$N \cdot p_1$
for j = 1:N
if node i receives the broadcasting data from node j
$S(j).id$ = [j,i];
$S(j).data = x_j + x_i$;
$N_2 = N_2 + 1$;
end if
end for
end for
Stage 3:
The receiving nodes in Stage 2 forward their update packets with probability p_2.
$N_3 = N_2 p_2$;
for $loop$ = 1:max
if $N_3 \leq 1$
break
end if
if node j forwards its packets
for i = 1: N
if node i has not received a packet and hears node j
$S(i).id$ = [i,j];
$S(i).data = x_i + x_j$;
$N_3 = N_3 + 1$;
end if
end for
end if
The reception nodes in the stage 3 forwarding their packets with probability p_2.
end for
Stage 4:
The mobile sink queries M nodes to generate $\boldsymbol{\Phi}$ and $\boldsymbol{Y}$.
$\boldsymbol{\Phi}$ = ***zeros*** (M, N)
if node i_k are queried
$\boldsymbol{\Omega}_k = S(i_k)$.id;
$\boldsymbol{\Phi}(k, \boldsymbol{\Omega}_k) = 1$;
end if
Return $\boldsymbol{\Phi}$ and $\boldsymbol{Y}$.

Algorithm 2 CS Reconstruction

Input:
Measurement vector: $\boldsymbol{y}$;
Measurement matrix: $\boldsymbol{\Phi}$;
Output:
Reconstructed vector: $\hat{X}$
1: Sink creates $\boldsymbol{Y}$ and BDM $\boldsymbol{\Phi}$ based on $\boldsymbol{y}_i$ and $\boldsymbol{\Phi}_i$;
2: $\hat{\boldsymbol{\theta}} = \arg\min \|\boldsymbol{\theta}\|_1$ s.t. $\boldsymbol{Y} = \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{\theta}$;
3: $\hat{\boldsymbol{X}} = \boldsymbol{\Psi}\hat{\boldsymbol{\theta}}$

3.2. Selection of Parameters

In this subsection, we investigate the values of the parameters $\boldsymbol{r_t}$ and $\boldsymbol{p_2}$. We consider a network with $\boldsymbol{N = 400}$ nodes, which are randomly deployed over an area of size $\boldsymbol{S = 1 \times 1}$ in this paper. As described in Section 2, to ensure the network connectivity, $\boldsymbol{r_t}$ must satisfy the condition in Equation (5). Thus, $\boldsymbol{r_t > 0.069}$; we set $\boldsymbol{r_t = 0.075}$.

In Stage 3 of CS-DDSG, nodes forward their updated packets with a probability $\boldsymbol{p_2}$ and all neighboring nodes can receive this data. For the sake of an appropriate $\boldsymbol{p_2}$ that reduces the number of transmissions $\boldsymbol{N_t}$ and increases the proportion of reception nodes $\boldsymbol{P_r}$ simultaneously, we simulate $\boldsymbol{N_r}$ and $\boldsymbol{P_r}$ versus $\boldsymbol{p_2}$ by setting $\boldsymbol{p_1 = 0.2}$ and $\boldsymbol{r_t = 0.075}$ as shown Figure 4, where all normal nodes stop receiving any data after merging one packet. As Figure 4 shows, as $\boldsymbol{p_2}$ increases, the values of $\boldsymbol{N_t}$ and $\boldsymbol{P_r}$ both increase. Furthermore, $\boldsymbol{P_r}$ increases almost linearly with $\boldsymbol{p_2}$. Thus, when $\boldsymbol{p_2 = 0.32}$, 98% nodes receive a broadcast packet. Moreover, as $\boldsymbol{p_2}$ increase beyond 0.32, $\boldsymbol{N_r}$ increases less, while $\boldsymbol{P_r}$ increases sharply. Therefore, the appropriate value for $\boldsymbol{p_2}$ is 0.32, because that value provides a balanced trade-off between the number of transmissions and the percentage of receiving nodes.

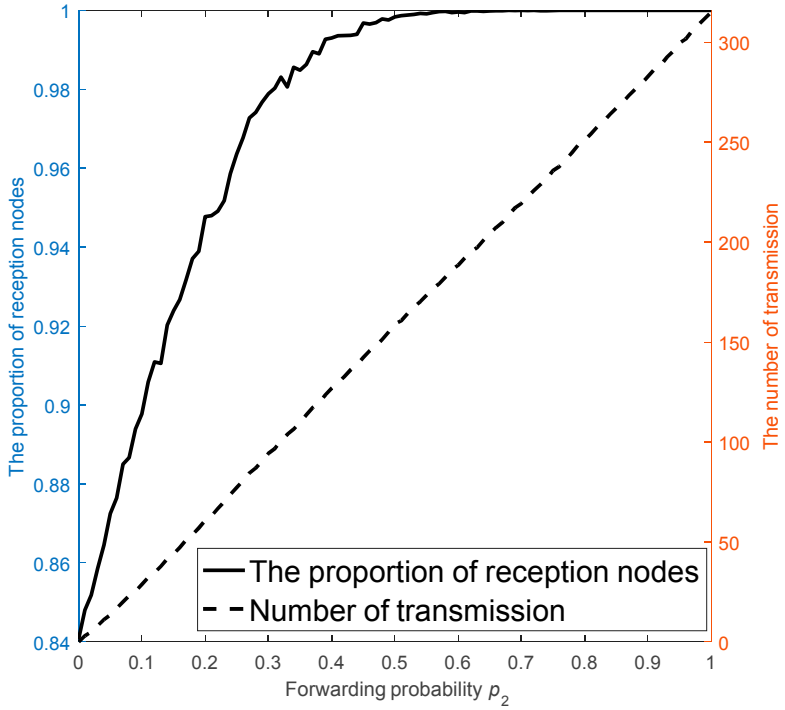

Figure 4. The impact of forwarding probability on the number of transmitting and receiving nodes.

3.3. Measurement Matrix Formulation

In this subsection, we present the formulation procedure for the measurement matrix. As we introduced above, in Stage 4, after the mobile sink queries the M nodes, which are denoted by $(n_{i_1}, n_{i_2} \cdots, n_{i_k}, \cdots n_{i_M}), i_1 < i_2 < \cdots < i_M, i_k \in [1, N]$, the measurement matrix Φ is constructed based on the M packets. Suppose Ω_k is the index of node ID and its definition is expressed as follows:

$$\Omega_k = S(n_{i_k}).\text{id}. \tag{11}$$

Initially, Φ is an all-zero $M \times N$ matrix, then Φ is formulated at this step which is given by Equation (12):

$$\blacksquare(k, j) = \begin{cases} 1, & j = \Omega_k \\ 0, & otherwise \end{cases}. \tag{12}$$

For example, assume there are five nodes in the network (i.e., $N = 5$). If the mobile sink queries two nodes (i.e., $M = 2$), then Φ can initially be expressed as follows:

$$\Phi_{2\times 5} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{13}$$

Suppose that nodes 2 and 4 are selected by the sink, and their packets components are as follows:

$$\begin{aligned} S(2).\text{id} &= [2, 5] \\ S(4).\text{id} &= [1, 4], \end{aligned} \tag{14}$$

then $\varphi_{1,2} = \varphi_{1,5} = 1$ and $\varphi_{2,1} = \varphi_{2,4} = 1$. Finally, the matrix Φ becomes:

$$\Phi_{2\times 5} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix}. \tag{15}$$

Moreover, the measurement vector Y is expressed as follows:

$$Y = (S(2).\text{data}, S(4).\text{data})^{\mathrm{T}}. \tag{16}$$

Obviously, Φ is a sparse matrix, whose sparsity degree is influenced by p, p_1 and p_2. Furthermore, Equation (12) indicates that Φ is constructed by relying on the gathered data, which precludes the need to measure lost data. Thus, Y is not influenced by lost packets at all. Therefore, CS-DDSG is resistant to the packet loss rate.

3.4. Does the Measurement Matrices Satisfy RIP?

The structure of measurement matrix Φ is random and relies on the receiving nodes. Thus, CS-DDSG avoids measuring the lost nodes and avoids the packet loss. The question is: Does Φ obey RIP to utilize the CS theory? Unfortunately, it is an NP-hard problem to prove the RIP property of a matrix. However, Yang et al. [21] reported that recovery performance can be guaranteed with high probability when the rows of the measurement matrix are linearly independent. We investigate this proposition below.

The rows of Φ are linearly dependent when one of the following two situations occurs.

Case 1. *Any row φ_k can be expressed as a linear combination of other rows.*

Proof. The measurement coefficient is 1; thus, if $\boldsymbol{\varphi_k}$ can be expressed as a linear combination of rows $\boldsymbol{\varphi_{k_1}, \cdots, \varphi_{k_q}}$, $\boldsymbol{q = 2, \cdots, N-1}$, they satisfy the following:

$$\boldsymbol{\varphi_k = \varphi_{k_1} + \cdots + \varphi_{k_q}}. \tag{17}$$

Suppose $\boldsymbol{I_k = \{j | \varphi_{k,j} \neq 0\}}$ and $\boldsymbol{I_{k_i} = \{j | \varphi_{k_i,j} \neq 0\}}$; if the condition of Equation (17) is satisfied, then $\boldsymbol{I_k = \cup_{i=1}^{q} I_{k_i}}$ and we can obtain

$$\boldsymbol{|I_k| > |I_{k_i}|, i \in [1, q]}, \tag{18}$$

where $|\cdot|$ denotes the number of elements in the set. Thus, Equation (17) can be satisfied when one of the following two situations occurs. The first situation would occur if node $\boldsymbol{k}$ were to receive packets from nodes $\boldsymbol{k_1, \cdots, k_q}$ and merge their packets. However, this situation contradicts the reception condition under which each node receives one packet. Thus, the condition of Equation (17) cannot occur.

The second situation would occurs when node $\boldsymbol{k_2}$ receives a packet from node $\boldsymbol{k_1}$ and node $\boldsymbol{k_3}$ receives a packet from node $\boldsymbol{k_2}$. It follows that node $\boldsymbol{k_q}$ receives a packet from node $\boldsymbol{k_{q-1}}$. Finally, node $\boldsymbol{k}$ receives the packet from node $\boldsymbol{k_q}$. According to the condition in Equation (10), $\boldsymbol{I_{k_q}}$ satisfies the following:

$$\boldsymbol{I_{k_q} = \{k_q\} \cup \left(\cup_{i=1}^{q-1} I_{k_i} \right)}. \tag{19}$$

After node $\boldsymbol{k}$ updates its packet, $\boldsymbol{I_k}$ satisfies:

$$\boldsymbol{I_k = \{k\} \cup I_{k_q}}. \tag{20}$$

Obviously, $\boldsymbol{k \in I_k}$ but $\boldsymbol{k \notin I_{k_q}}$. Thus, $\boldsymbol{I_k = \{k\} \cup \left(\cup_{i=1}^{q} I_{k_i} \right)}$ and Equation (17) is false.
Consequently, it can be concluded that no rows can be linearly expressed by other rows. □

Case 2. *Any two rows $\boldsymbol{\varphi_i}$ and $\boldsymbol{\varphi_j}$ are linearly dependent.*

Proof. $\boldsymbol{\varphi_i}$ and $\boldsymbol{\varphi_j}$ are linearly dependent if and only if they are precisely the same. However, according to the reception condition, each node receives only one packet and merges with its own unique packet. Therefore, although node $\boldsymbol{i}$ and node $\boldsymbol{j}$ may receive the same broadcasting packet from a common neighboring node, their packets will still be different. Therefore, none of the rows are linearly dependent. □

In conclusion, the rows of the measurement matrix $\boldsymbol{\Phi}$ are linearly independent; consequently, in CS-SSDG, $\boldsymbol{X}$ can be reconstructed from $\boldsymbol{Y}$ with a very high probability.

4. Formulating the Expression of the Total Number of Transmissions and Receptions

Compared with the mainstream algorithms [15,20,21], the proposed scheme CS-DDSG reduces the number of transmissions and receptions rather than the number of fusions. In this section, we formulate the total number of transmissions $\boldsymbol{N_{Ttot}}$ and receptions $\boldsymbol{N_{Rtot}}$ based on the random geometric graph (RGG) mode [26] and the torus convention [27] to investigate the efficiency in reducing $\boldsymbol{N_{Ttot}}$ and $\boldsymbol{N_{Rtot}}$.

According to Section 3, $\boldsymbol{N_{Ttot}}$ and $\boldsymbol{N_{Rtot}}$ can be expressed as follows:

$$\begin{aligned} \boldsymbol{N_{Ttot}} &= \boldsymbol{N_t^P + N_t = N_s + \sum_{q=1}^{N_f} N_t^q} \\ \boldsymbol{N_{Rtot}} &= \boldsymbol{N_r^P + N_r = N_r^P + \sum_{q=1}^{N_f} N_r^q}, \end{aligned} \tag{21}$$

where $\boldsymbol{N_t^P}$ and $\boldsymbol{N_r^P}$ denote the number of transmitting and reception nodes in Stage 2, respectively. $\boldsymbol{N_t}$ and $\boldsymbol{N_r}$ denote the number of transmitting and receiving nodes in Stage 3, respectively. Similarly,

N_t^q and N_r^q represent the number of transmitting and receiving nodes in the qth forwarding of Stage 3, respectively. N_f denotes the number of forwarding iterations. In Stage 2, N_s nodes are selected to broadcast, thus $N_t^P = N_s$. Because the receiving nodes in Stage 3 forward their packet with the probability p_2, N_r^{q-1} and N_t^q satisfy the following:

$$N_t^q = N_r^{q-1} \cdot p_2. \tag{22}$$

When $N_t^{q*} = N_r^{q*-1} \cdot p_2 \leq 0$, no node forwards packets and the forwarding process is completed. Thus, $N_f = q^* - 1$, $N_t = \sum_{q=1}^{N_f} N_t^q, N_r = \sum_{q=1}^{N_f} N_r^q$. Additionally, $N_r^0 = N_r^P$ and $N_t^0 = N_s$. Next, we formulate the expression of N_r^P, N_t^q and N_r^q.

4.1. Formulating N_r^P

Proposition 1. *The number of receptions in Stage 2 N_r^P is:*

$$N_r^P = N_s N \pi r_t^2 - C_{N_s}^2 \pi^2 N r_t^4. \tag{23}$$

Proof. According to the procedures of Stage 2, N_r^P equals the number of neighboring nodes around all the source nodes $N_{s,nei}$ minus the number of nodes N_{r2} located in the overlapping communication region of the two sources nodes. This relation occurs because each node receives just one packet and the number of receptions for those nodes is counted twice, thus N_r^P can be represented as follows:

$$N_r^P = N_{s,nei} - N_{r2}. \tag{24}$$

The average number of neighboring nodes for all source nodes $N_{s,nei}$ is expressed as follows:

$$N_{s,nei} = N_s N \frac{\pi r_t^2}{S} = N_s N \pi r_t^2. \tag{25}$$

In Figure 5, the red circle denotes the communication region and S_2 represents the shaded area jointly covered by the two source nodes. A and B are two intersections. When the distance between two source nodes $d(O, O')$ satisfies $0 < d(O, O') \leq 2r_t$, N_{r2} exists. Thus, the probability p_L of an existing communication between the two nodes is expressed as follows:

$$p_L = p\{d(O, O') \leq 2r_t\} = \frac{\pi (2r_t)^2}{S} = 4\pi r_t^2. \tag{26}$$

In the N_s source nodes, an average of N_L nodes pairs satisfy the condition in Equation (26) (i.e., N_L source nodes pairs can communicate with each other). The expressions for N_L and N_{r2} are, respectively, as follows:

$$N_L = C_{N_s}^2 \cdot p_L = C_{N_s}^2 \cdot 4\pi r_t^2. \tag{27}$$

$$N_{r2} = N_L \times N \times \frac{\overline{S}_2}{S}. \tag{28}$$

Because the nodes are uniformly distributed and $0 < d(O, O') \leq 2r_t$, the probability $p\{d \leq x\}$ is equal to

$$F_1(x) = p\{d \leq x\} = \frac{\pi x^2}{\pi (2r_t)^2} = \frac{x^2}{4r_t^2}. \tag{29}$$

Thus, the probability density function (PDF) $f_1(x)$ is

$$f_1(x) = F_1{}'(x) = \frac{x}{2r_t}. \tag{30}$$

In this case, the area $S_2/2$ equals the area of sector OAB minus the area of triangle OAB:

$$\begin{aligned} S_2 &= 2\left(\frac{r_t^2}{2} \times 2\arccos\frac{d}{2r_t} - \frac{1}{2} \times \frac{d}{2} \times 2\sqrt{r_t^2 - \frac{d^2}{4}}\right) \\ &= 2r_t^2\arccos\frac{d}{2r_t} - \frac{d}{2}\sqrt{4r_t^2 - d^2}. \end{aligned} \tag{31}$$

Thus, the expected area of S_2 is calculated as follows:

$$\overline{S}_2 = \int_0^{2r_t} S_2 f_1(x)dx = \int_0^{2r_t} \left(2r_t^2\arccos\frac{x}{2r_t} - \frac{x}{2}\sqrt{4r_t^2 - x^2}\right)\frac{x}{2r_t^2}dx = \frac{\pi}{4}r_t^2. \tag{32}$$

Combining Equations (27), (28) and (32), N_{r2} can be formulated as:

$$N_{r2} = N_L \times N \times \frac{\overline{S}_2}{S} = C_{N_s}^2 \cdot 4\pi r_t^2 \cdot N \cdot \frac{\pi}{4}r_t^2 = C_{N_s}^2 \pi^2 N r_t^4. \tag{33}$$

Finally, we substitute Equations (25) and (33) into Equation (24), to obtain the representation of N_r^P:

$$N_r^P = N_s N \pi r_t^2 - C_{N_s}^2 \pi^2 N r_t^4. \tag{34}$$

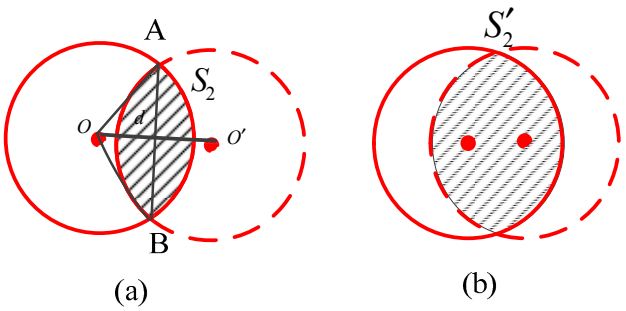

Figure 5. Diagram of two communication nodes: (**a**) $r_t < d(O, O') \leq 2r_t$; and (**b**) $0 < d(O, O') \leq r_t$.

4.2. Formulating N_r^q

Figure 6 shows the forwarding procedure of Stage 3, where n_t^q denotes the transmitting node in the qth forwarding; its communication range is represented by the black circle. Node n_t^{q-1} broadcasts its packet in the $(q-1)$th forwarding process. Because of the reception conditions, the nodes located in area S_3 can receive the forwarded packet broadcast by n_t^q. Let N_{r1}^q denotes the number of receiving nodes in area S_3.

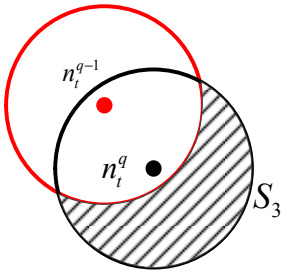

Figure 6. Diagram of forwarding packets.

Besides, there are two situations should be considered:

1. Case 1: As presented in Figure 7, there are two broadcasting nodes n_{t1}^{q-1} and n_{t2}^{q-1} in the $(q-1)$th forwarding, while their communication ranges are represented by the two red circles. This case can be divided into two situations via the distance $d\left(n_{t1}^{q-1}, n_{t2}^{q-1}\right)$: (a) $0 < d\left(n_{t1}^{q-1}, n_{t2}^{q-1}\right) < r_t$; and (b) $r_t < d\left(n_{t1}^{q-1}, n_{t2}^{q-1}\right) < 2r_t$. Taking the first situation as an example, if node n_t^q is located in the black area S_4, the nodes in the shadow area S_5 can receive packets from nodes n_t^q or n_{t2}^{q-1}, thus the number of receptions of those nodes is counted twice, and that value should be subtracted from N_{r1}^q. Suppose that the number of receiving nodes in areas such as S_5 is N_{r2}^q and that the number in areas such as S_7 is N_{r3}^q.
2. Case 2: Similarly, there are two transmitting nodes n_{t1}^q and n_{t2}^q in the qth forwarding, whose communication ranges are represented by two black circles in Figure 8. This case can be divided into two situations via the distance $d\left(n_{t1}^q, n_t^{q-1}\right)$: (a) $0 < d\left(n_{t1}^q, n_t^{q-1}\right) < r_t$; and (b) $r_t < d\left(n_{t1}^q, n_t^{q-1}\right) < 2r_t$. Taking the first situation as an example, when node n_{t2}^q is distributed in the black area S_8, the nodes located in shadow area S_9 receive one of the packets broadcasted by node n_{r1}^q or n_{r2}^q. Thus, the number of receptions for those nodes is counted twice, which should be subtracted from N_{r1}^q. Suppose that the number of reception nodes in areas such as S_9 is N_{r4}^q and the number in areas such as S_{10} is N_{r5}^q.

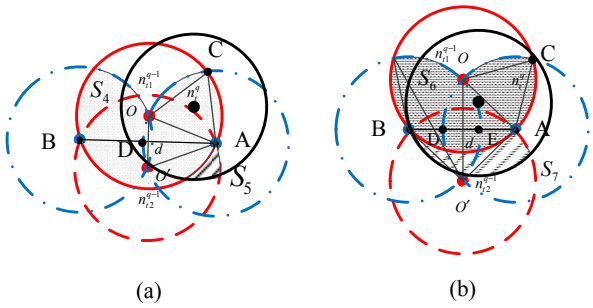

Figure 7. The diagram of Case 1: (**a**) $0 < d\left(n_{t1}^{q-1}, n_{t2}^{q-1}\right) < r_t$; and (**b**) $r_t < d\left(n_{t1}^{q-1}, n_{t2}^{q-1}\right) < 2r_t$.

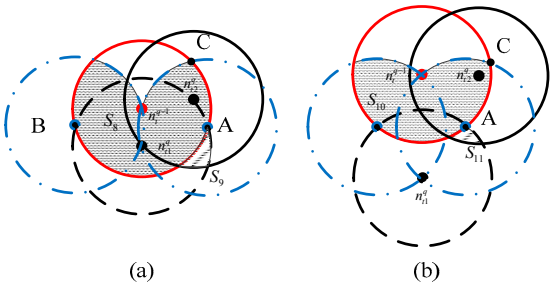

Figure 8. The diagram of Case 2: (**a**) $0 < d\left(n_{t1}^{q-1}, n_{t2}^{q-1}\right) < r_t$; and (**b**) $r_t < d\left(n_{t1}^{q-1}, n_{t2}^{q-1}\right) < 2r_t$.

In conclusion, the number of receptions N_r^q for Stage 3 in the q^{th} forwarding can be expressed as follows:

$$N_r^q = N_{r1}^q - N_{r2}^q - N_{r3}^q - N_{r4}^q - N_{r5}^q. \tag{35}$$

Next, we formulate the expression of N_{r1}^q, N_{r2}^q, N_{r3}^q, N_{r4}^q and N_{r5}^q.

4.2.1. Calculating N_{r1}^q

As shown in Figure 6, the nodes in shadow area S_3 would receive the packet. Thus, N_{r1}^q is calculated as follows:

$$N_{r1}^q = N_t^q \times N \times \frac{\overline{S}_3}{S}. \tag{36}$$

In the above formula, $\overline{S}_3$ can be expressed as follows:

$$\overline{S}_3 = \pi r_t^2 - \overline{S'}_2. \tag{37}$$

Because the nodes are uniformly distributed and $0 < d(O,O') \leq r_t$, the probability $p\{d \leq x\}$ equals

$$F_2(x) = p\{d \leq x\} = \frac{\pi x^2}{\pi r_t^2} = \frac{x^2}{r_t^2}. \tag{38}$$

Thus, the PDF $f_2(x)$ is

$$f_2(x) = F'_2(x) = \frac{2x}{r_t^2}. \tag{39}$$

Combining Equations (31), (37) and (39), we obtain

$$\overline{S}'_2 = \int_0^{r_t} \left(2r_t^2 \arccos\frac{x}{2r_t} - \frac{x}{2}\sqrt{4r_t^2 - x^2}\right)\frac{2x}{r_t^2}dx = \left(\pi - \frac{\sqrt{3}}{4}\right)r_t^2. \tag{40}$$

$$\overline{S}_3 = \pi r_t^2 - \overline{S'}_2 = \pi r_t^2 - \left(\pi - \frac{3\sqrt{3}}{4}\right)r_t^2 = \frac{3\sqrt{3}}{4}r_t^2. \tag{41}$$

Thus, we obtain

$$N_{r1}^q = N_t^q N \frac{3\sqrt{3}}{4}r_t^2. \tag{42}$$

4.2.2. Calculating N_{r2}^q

Next, we formulate the expression of N_{r2}^q. As presented in Figure 7a, the value of N_{r2}^q is the number of receive node in area S_5, thus we have

$$N_{r2}^q = C_{N_t^{q-1}}^2 \times {p'}_L \times N_t^q \times \frac{\overline{S}_4}{\pi r_t^2} \times N \times \frac{\overline{S}_5}{S}, \tag{43}$$

where $\overline{S}_4$ denotes the expected area of the black region, $\overline{S}_5$ denotes the expected area of the shadow region, and ${p'}_L$ denotes the probability that the distance between two nodes satisfies $0 \leq d(O,O') \leq r_t$. Thus, we have the following:

$${p'}_L = p\{d(O,O') \leq r_t\} = \frac{\pi {r_t}^2}{S} = \pi r_t^2. \tag{44}$$

As shown in Figure 7a, the area S_4 equals twice the area of region ACD minus the half intersection area of circle A and B, i.e., $\overline{S}_2/2$, plus the half intersection area of circle O and O', i.e., $\overline{S'}_2/2$. The area of region ACD equals to the area of sector ACD plus the area of sector OAC minus the area of triangle OCA; thus,

$$S_{ACD} = \pi r_t^2 \times \frac{\frac{\pi}{3} + \arcsin\left(\frac{d}{2r_t}\right)}{2\pi} + \pi r_t^2 \times \frac{\frac{\pi}{3}}{2\pi} - \frac{1}{2} \times r_t \times \frac{\sqrt{3}r_t}{2} = \frac{r_t^2}{2}\left[\frac{\pi}{3} + \arcsin\left(\frac{d}{2r_t}\right)\right] + \frac{\pi}{6}r_t^2 - \frac{\sqrt{3}}{4}r_t^2. \tag{45}$$

$$S_4 = S_{ACD} - \frac{\overline{S}_2}{2} + \frac{\overline{S'}_2}{2} = \frac{r_t^2}{2}\left[\frac{\pi}{3} + \arcsin\left(\frac{d}{2r_t}\right)\right] + \frac{\pi}{6}r_t^2 - \frac{\sqrt{3}}{4}r_t^2 - \frac{\overline{S}_2}{2} + \frac{\overline{S'}_2}{2}. \tag{46}$$

Combining Equations (39) and (46), we obtain

$$\begin{aligned}\overline{S}_4 &= \int_0^{r_t} 2 \times \left[\frac{r_t^2}{2}\left(\frac{\pi}{3} + \arcsin\frac{x}{2r_t}\right) + \frac{\pi r_t^2}{6} - \frac{\sqrt{3}r_t^2}{4}\right] f_2(x)dx - \frac{\pi r_t^2}{8} + \frac{1}{2}\left(\pi - \frac{3\sqrt{3}}{4}\right)r_t^2 \\ &= \frac{\pi r_t^2}{2} - \frac{\pi r_t^2}{8} + \frac{1}{2}\left(\pi - \frac{3\sqrt{3}}{4}\right)r_t^2 = \frac{7\pi r_t^2}{8} - \frac{3\sqrt{3}r_t^2}{8},\end{aligned} \tag{47}$$

where $2\int_0^{r_t} x\arcsin\frac{x}{2r_t}dx = \left(\frac{\sqrt{3}}{2} - \frac{\pi}{6}\right)r_t^2$. According to the method in [28], we can get the approximate value of $\overline{S}_5$, i.e., $\overline{S}_5 \simeq S_{5,\max} = \pi r_t^2/6$. Finally, combining Equations (43), (44) and (47), we obtain

$$N_{r2}^q = \frac{C_{N_t^{q-1}}^2 N N_t^q}{6}\left(\frac{7\pi}{8} - \frac{3\sqrt{3}}{8}\right). \tag{48}$$

4.2.3. Calculating N_{r3}^q

The expression of N_{r3}^q is similar to that of N_{r2}^q:

$$N_{r3}^q = C_{N_t^{q-1}}^2 \times p_L \times N_t^q \times \frac{\overline{S}_6}{\pi r_t^2} \times N \times \frac{\overline{S}_7}{S}, \tag{49}$$

As shown in Figure 7b, because $r_t \le d(O,O') \le 2r_t$, $\overline{S}_6$ is calculated as follows:

$$\begin{aligned}\overline{S}_6 = 2S_{ACD} &= \int_{r_t}^{2r_t} 2 \times \left[\frac{r_t^2}{2}\left(\frac{\pi}{3} + \arcsin\frac{x}{2r_t}\right) + \frac{\pi r_t^2}{6} - \frac{\sqrt{3}r_t^2}{4}\right] f_1(x)dx \\ &= \int_{r_t}^{2r_t} \frac{\pi}{3}x dx + \frac{1}{2}\int_r^{2r} x\arcsin\frac{x}{2r_t}dx - \frac{\sqrt{3}}{4}\int_r^{2r} x dx \\ &= \left(\frac{13\pi}{24} - \frac{\sqrt{3}}{2}\right)r_t^2,\end{aligned} \tag{50}$$

where $\frac{1}{2}\int_r^{2r} x\arcsin\frac{x}{2r_t}dx = \frac{\pi}{24}r_t^2 - \frac{\sqrt{3}}{8}r_t^2$, and S_7 is expressed as

$$\overline{S}_7 = \overline{S}_2 - \overline{S}_{12}, \tag{51}$$

where $\overline{S}_{12}$ is the intersection area of circle O and circle O' when $r_t \le d(O,O') \le 2r_t$, thus

$$\overline{S}_{12} = \int_{r_t}^{2r_t}\left(2r_t^2\arccos\frac{x}{2r_t} - \frac{x\sqrt{4r_t^2 - x^2}}{2}\right)\frac{x}{2r_t^2}dx = \frac{3\sqrt{3}r_t^2}{16}, \tag{52}$$

and

$$\overline{S}_7 = \overline{S}_2 - \overline{S}_{12} = \left(\frac{\pi}{4} - \frac{3\sqrt{3}}{16}\right)r_t^2. \tag{53}$$

Combining Equations (49), (50) and (53), we obtain

$$N_{r3}^q = \frac{C_{N_t^{q-1}}^2 4r_t^2 N N_t^q}{\pi}\left(\frac{13}{24} - \frac{\sqrt{3}}{2\pi}\right)\left(\frac{\pi}{4} - \frac{3\sqrt{3}}{16}\right). \tag{54}$$

4.2.4. Calculating N_{r4}^q and N_{r5}^q

As illustrated in Figure 8, the two black circles denote the communication range of two transmitting nodes, n_{t1}^q and n_{t2}^q, in the q forwarding. The red circle denotes the communication

range of transmission node n_t^{q-1} in the $q-1$ forwarding. The calculation of N_{r4}^q and N_{r5}^q is similar to that of N_{r2}^q and N_{r3}^q:

$$N_{r4}^q = C_{N_t^q}^1 C_{N_t^{q-1}}^1 {p'}_L \times C_{N_t^q-1}^1 \frac{\overline{S}_8}{\pi r_t^2} \times N\frac{\overline{S}_9}{S}, \tag{55}$$

$$N_{r5}^q = C_{N_t^q}^1 C_{N_t^{q-1}}^1 p_L \times C_{N_t^q-1}^1 \frac{\overline{S}_{10}}{\pi r_t^2} \times N\frac{\overline{S}_{11}}{S}, \tag{56}$$

where S_8 and S_{10} denote the area of the black region and S_9 and S_{11} denote the area of the shadow region. Compared with Figures 7 and 8, we have the following:

$$\begin{cases} \overline{S}_8 = \overline{S}_4, \overline{S}_9 = \overline{S}_5 \\ \overline{S}_{10} = \overline{S}_6, \overline{S}_{11} = \overline{S}_7 \end{cases}. \tag{57}$$

Thus, the expressions of N_{r4}^q and N_{r5}^q are:

$$N_{r4}^q = \frac{N_t^q N_t^{q-1} \pi N r_t^4 \left(N_t^q - 1\right)}{6} \left(\frac{7\pi}{8} - \frac{3\sqrt{3}}{8}\right) \tag{58}$$

$$N_{r5}^q = 4N_t^q N_t^{q-1} N \pi r_t^4 \left(N_t^q - 1\right) \left(\frac{13}{24} - \frac{\sqrt{3}}{2\pi}\right) \left(\frac{\pi}{4} - \frac{3\sqrt{3}}{16}\right) \tag{59}$$

In conclusion, by combining Equations (35), (42), (48), (54), (58) and (59), we can obtain the expression of N_r^q:

$$\begin{aligned} N_r^q = {} & N_t^q N \tfrac{3\sqrt{3}}{4} r_t^2 - \frac{C_{N_t^{q-1}}^2 N N_t^q}{6}\left(\tfrac{7\pi}{8} - \tfrac{3\sqrt{3}}{8}\right) - \frac{C_{N_t^{q-1}}^2 4r_t^2 N N_t^q}{\pi}\left(\tfrac{13}{24} - \tfrac{\sqrt{3}}{2\pi}\right)\left(\tfrac{\pi}{4} - \tfrac{3\sqrt{3}}{16}\right) \\ & - \frac{N_t^q N_t^{q-1} \pi N r_t^4 \left(N_t^q - 1\right)}{6}\left(\tfrac{7\pi}{8} - \tfrac{3\sqrt{3}}{8}\right) - 4N_t^q N_t^{q-1} N \pi r_t^4 \left(N_t^q - 1\right)\left(\tfrac{13}{24} - \tfrac{\sqrt{3}}{2\pi}\right)\left(\tfrac{\pi}{4} - \tfrac{3\sqrt{3}}{16}\right). \end{aligned} \tag{60}$$

4.3. The Formulation of N_{Ttot} and N_{Rtot}

Theorem 1. *Assume that all N sensor nodes are deployed randomly and uniformly in a distributed WSNs with a boundary length of 1, and each node has a transmission range of r_t. If we gather data based on CS-DDSG scheme, then N_{Ttot} and N_{Rtot} are, respectively, expressed as follows:*

$$N_{Ttot} = N_s + \sum_{q=1}^{N_f} N_t^q, \tag{61}$$

$$\begin{aligned} N_{Rtot} = {} & N_s N \pi r_t^2 - C_{N_s}^2 \pi^2 N r_t^4 + \textstyle\sum_{q=1}^{N_f} N_t^q N \frac{3\sqrt{3}}{4} r_t^2 - \sum_{q=1}^{N_f} \frac{C_{N_t^{q-1}}^2 N N_t^q}{6}\left(\frac{7\pi}{8} - \frac{3\sqrt{3}}{8}\right) \\ & - \textstyle\sum_{q=1}^{N_f} \frac{C_{N_t^{q-1}}^2 4r_t^2 N N_t^q}{\pi}\left(\frac{13}{24} - \frac{\sqrt{3}}{2\pi}\right)\left(\frac{\pi}{4} - \frac{3\sqrt{3}}{16}\right) - \sum_{q=1}^{N_f} \frac{N_t^q N_t^{q-1} \pi N r_t^4 \left(N_t^q - 1\right)}{6}\left(\frac{7\pi}{8} - \frac{3\sqrt{3}}{8}\right) \\ & - \textstyle\sum_{q=1}^{N_f} 4N_t^q N_t^{q-1} N \pi r_t^4 \left(N_t^q - 1\right)\left(\frac{13}{24} - \frac{\sqrt{3}}{2\pi}\right)\left(\frac{\pi}{4} - \frac{3\sqrt{3}}{16}\right), \end{aligned} \tag{62}$$

where $N_r^0 = N_r^P, N_t^0 = N_s, N_t^q = N_r^{q-1} \times p_1$ and $N_f = q^* - 1$, where q^* satisfies $N_t^{q^*} = N_r^{q^*-1} \times p_1 \leq 0$. The expression for N_r^P is given in Equation (34).

Proof. As presented in the above derivation, we can obviously obtain Equation (61) based on Equation (21) and the correlative description in Stage 2 of CS-DDSG. Furthermore, by combining Equations (21), (34) and (60), we can obtain the expression of Equation (62). □

5. Performance Evaluation and Analysis

To evaluate the effectiveness of CS-DDSG, we ran simulations in MATLAB 2012b. The simulation parameters were set as shown in Table 1. Furthermore, we adopted the FFT orthonormal basis and the orthogonal matching pursuit (OMP) method for the reconstruction algorithm. We used the real sensor readings extracted from the GreenOrbs [29] system.

Table 1. Default Simulation Parameters.

	Parameters	Value
N	The total number of sensors	400
a	Boundary length	1
p_2	The probability of forwarding in Stage 3	0.32
r_t	Communication radius	0.075

In this paper, we present the performance comparations of CS-DDSG, Compressive Sensing Data storage (CStorage) [20], Improved CStorage (ICStorage) [21], Compressed Network Coding based Distributed data Storage (CNCDS) [21] and Direct Cluster-Based Compressive Sensing Data Collection (DCCS) [15] on unreliable links. These first four schemes all combine DDS and CS to gather data. CStorage, ICStorage and CNCDS are concerned with reducing the number of transmission and fusions. In CStorage, intermediate nodes receive the broadcasting packets when they first receive, and then, they forward the received packet with a given probability. The intermediate nodes in ICStorage forward their own readings rather than the received source nodes readings. In the CNCDS scheme, the intermediate nodes receive broadcast packets only if the receiving node does not share any node IDs with the corresponding transmitting node. We also analyze the numbers of transmissions, receptions and fusions involved in the first four algorithms. DCCS combines CS and cluster topology to reduce the total power consumption with no consideration of packet loss rate. All member nodes gather data and transmit to cluster heads, where the CS measurements and measurement matrices are generated and send to sink directly. Additionally, we discuss the impact of packet loss rate, the number of measurements and the proportion of source nodes on the performance of CS-DDSG. The simulation results shown are the average values from 1000 runs.

First, we evaluate the performance on unreliable links when $p\mathbf{1} = \mathbf{0.3}, M = \mathbf{50}$, as shown in Figure 9. It can be seen that: (1) As p increases, the reconstruction accuracy of all the algorithms decreases in Figure 9a. When $p \leq \mathbf{0.6}$, the NMAEs of the four algorithms are stable and increase gradually, which indicates that CS-DDSG is effective at resisting the packet loss. Although the packet loss rate impacts the nodes receiving broadcasting packets, the sink still gathers enough packets to recover the data. In addition, the sink constructs the measurement matrix based on received packets, which avoids the need to measure the lost nodes and reduces the impact of unreliable links on measurement vector Y. However, the performance of DCCS is poor with an increase in p. Sink cannot find the lossy nodes and still reconstructs data based on the original measurement matrices. Thus, DCCS is sensitive to p. (2) CS-DDSG outperforms the other algorithms. This improved performance occurs because in CS-DDSG, nodes receive only one packet which is broadcasted by its neighbor nodes in CS-DDSG. Thus, the measurement vectors have the characteristic of strong spatial correlation, which is utilized by CS to recover the data. However, in the other algorithms, nodes would fuse packets from distant nodes as long as the receipt condition is satisfied, which leads to a weak spatial correlation of measurement vectors. Thus, CS-DDSG outperforms the other algorithms.

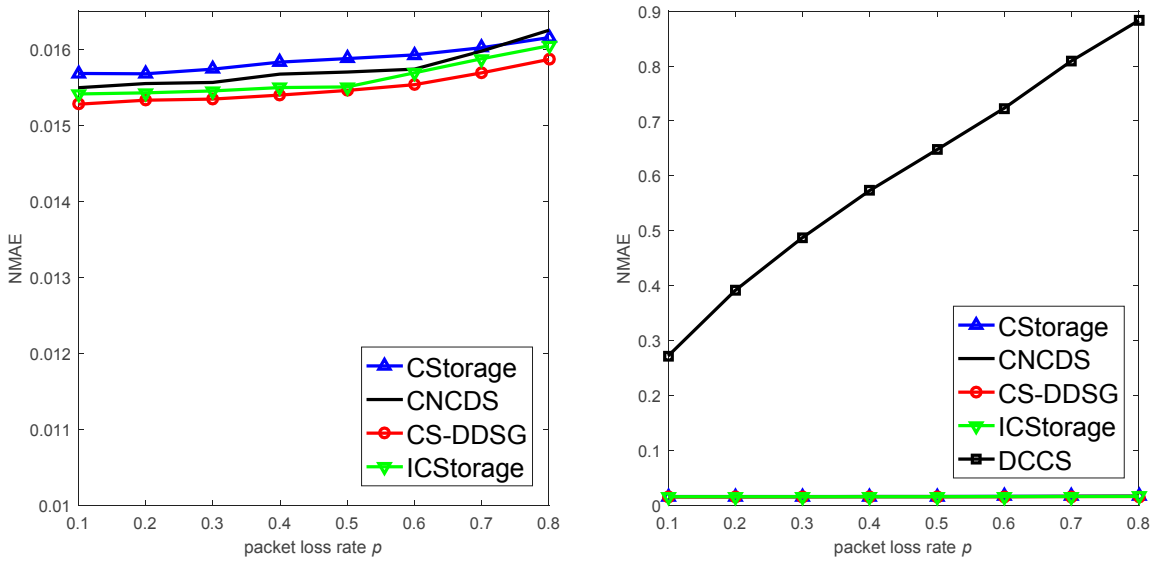

Figure 9. Performance of the different algorithms in unreliable links: (**a**) DDS-based algorithms; and (**b**) comparison between DDS-based algorithms and cluster-based algorithm.

We present the total number of transmissions, receptions and fusions of the four algorithms in Figure 10 when $p = 0.3$ and $p_1 = 0.15$. CS-DDSG requires fewer transmissions, receptions and fusions than do the CNCDS, CStorage and ICStorage schemes. This is because nodes in CS-DDSG receive packets only the first time and broadcast their packets with the probability p_2, after which then they do not receive any data. However, for CNCDS, CStorage and ICStorage, nodes continue to receive packets as long as the reception condition is satisfied. CStorage and ICStorage in particular focus on reducing the number of transmissions. Moreover, compared with CNCDS, CStorage and ICStorage, CS-DDSG scheme reduces N_{Ttot} by up to 23.9%, 42.5% and 67.8%, respectively, and reduces N_{Rtot} by up to 73.8%, 80.2% and 89.9%, respectively.

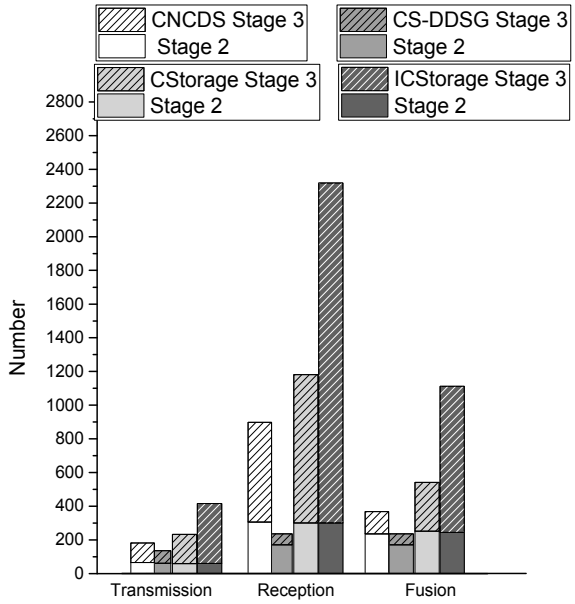

Figure 10. The total number of transmissions, receptions and fusions in Stages 2 and 3.

Furthermore, we investigate the fusion proportion of the total number of receptions. As presented in Figure 11, only 41% of the receiving nodes in CNCDS merge the received packets; the authors consider that only 41% of nodes lose energy. In fact, 59% of the receiving nodes also consume energy because they would receive the broadcast packet first and then determine whether the condition of CNCDS are satisfied; the received packets will be merged only if they satisfy the condition. Thus, energy is consumed even when the received packets are not fused. However, the number of receptions in [21] is the same as the number of fusions, which is less counted. Similarly, 46% and 48% of the receiving reception nodes in CStorage and ICStorage merge the packets, respectively. In CS-DDSG, all received nodes are fused and no redundancy occurs because the nodes receive packets only once. Thus, the energy consumption of CS-DDSG receiving nodes is much smaller than that of the other algorithms. In conclusion, CS-DDSG effectively reduces both the number of transmissions and receptions.

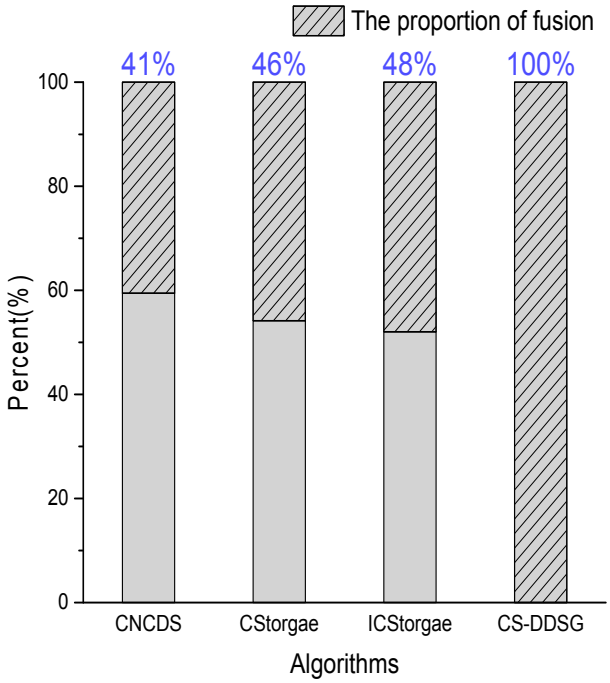

Figure 11. The fusion proportion of the total number of receptions.

Figure 12 presents the number of fusions and receiving nodes during each forwarding round when $p_1 = 0.15$. The forwarding process of CS-DDSG repeats five times, until no node remains to accept the broadcast packets, while CNCDS, CStorage and ICStorage repeat six, nine and twelve times, respectively. The network employing CS-DDSG has the fastest convergence and characteristics of efficiency due to the strictest reception conditions. Moreover, most of the data fusion occurs during Stage 2, and subsequently the number of fusions rapidly decreases in Stage 3 except in ICStorage.

In Figure 13, we investigate the recovery performance of the algorithms when $p_1 = 0.3$ and the number of measurements M, which is queried by the mobile sink, ranges from 15 to 150. It can be observed that, with an increase in M, the recovery accuracy of ICStorage, CStorage, CNCDS and CS-DDSG are improved and equivalent, while the performance of CS-DDSG becomes slightly better when $M \geq 100$. This improvement occurs because the more information that is gathered, the better is the reconstruction accuracy. According to Equation (12), the sink constructs measurement matrix Φ based on the packets fused by the forwarding nodes. The forwarding nodes of CS-DDSG receive only one packet, and the Φ is sparser than that in the others algorithms. Consequently, less information is gathered and fewer nodes contribute to data recovery for CS-DDSG. However, with an increase in M, more information is gathered and the gaps separating the four algorithms decrease. When $M > 100$,

CS-DDSG outperforms the four DDS-based algorithms due to the strong spatial correlation of the measurement vector. Moreover, the reconstruction accuracy of DCCS is the best when M is large enough and there is no packet loss. All nodes in DCCS participate in gathering data and DCCS adopts dense measurement matrix in clusters. Thus, more information is gathered. In addition, performance tends to be stable as M increases.

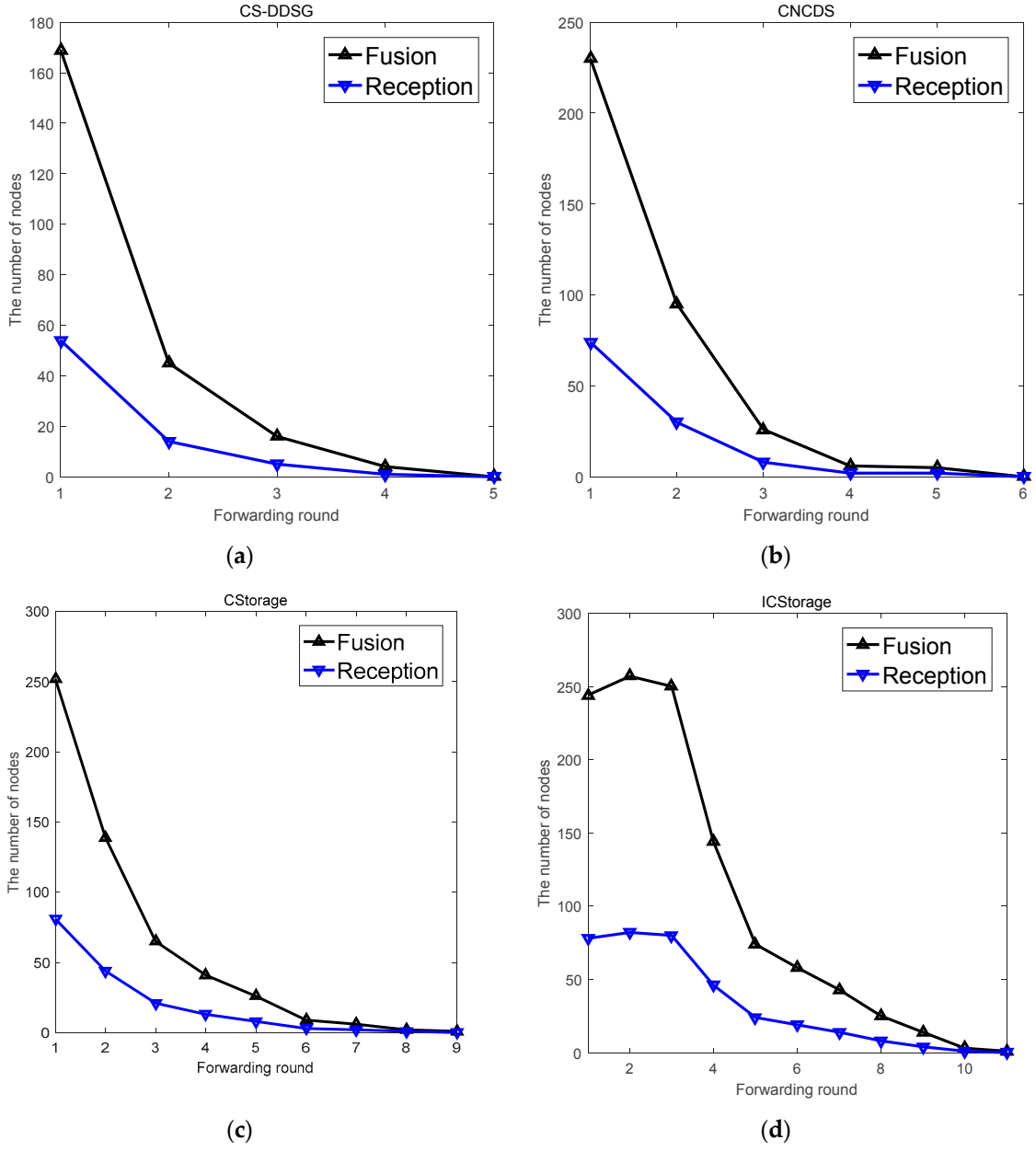

Figure 12. The number of fusions and receiving nodes during the forwarding process: (**a**) CS-DDSG; (**b**) CNCDS; (**c**) CStorage; and (**d**) ICStorage.

Figure 14 shows the performance of CS-DDSG under different packet loss ratios p and probabilities p_1 when $M = 40$. As p increases, the value of NMAE remains stable, i.e., $NMAE \approx 0.014$. This result indicates that CS-DDSG effectively resists the packet loss and maintains high reconstruction accuracy even when unreliable links exist. Additionally, its accuracy is not influenced by p_1 due to the very sparse measurement matrix.

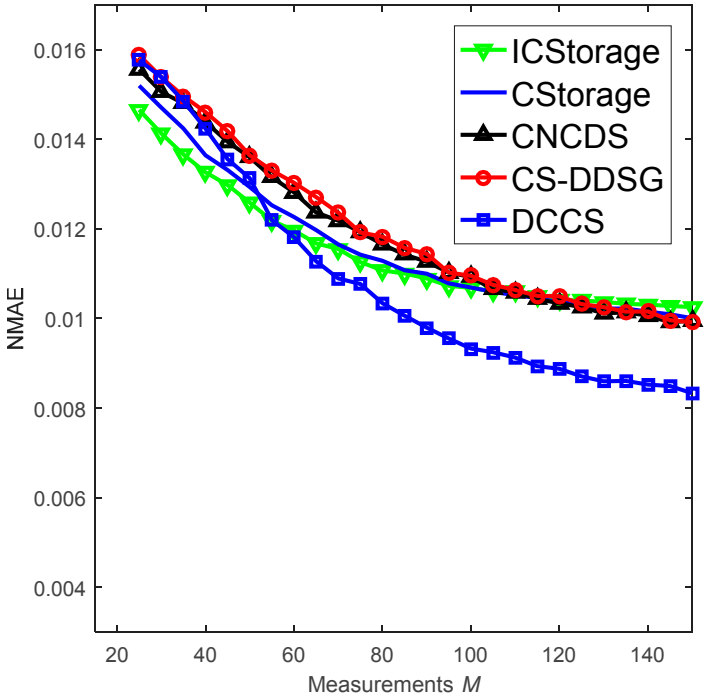

Figure 13. Performance of the algorithms when $p_1 = 0.3$.

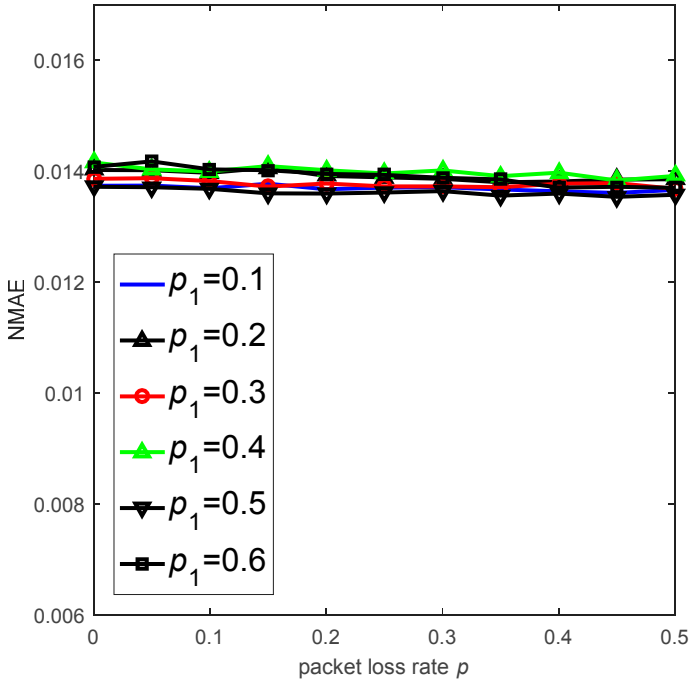

Figure 14. Performance of CS-DDSG with different p_1 and p.

Finally, we investigate how the proportion of source nodes p_1 impact the recovery accuracy in Figure 15. The simulation results show the following: (1) When $p_1 = 0.4$, the value of NMAE decreases as M increases because more nodes participate in data reconstruction as M increases. (2) When M is fixed, CS-DDSG performance is improved and the trend of NMAE values is very close to the value of p_1, varying from 0 to 0.6. This effect occurs because, when there are more source nodes, more nodes will receive broadcast packets before the sink obtains data. Hence, the amount of information used for reconstruction increases. However, due to the reception condition, the measurement matrix Φ is

sparse. Thus, information is increasingly limited. As a result, the trends of the NMAE values are close to the different value of p_1.

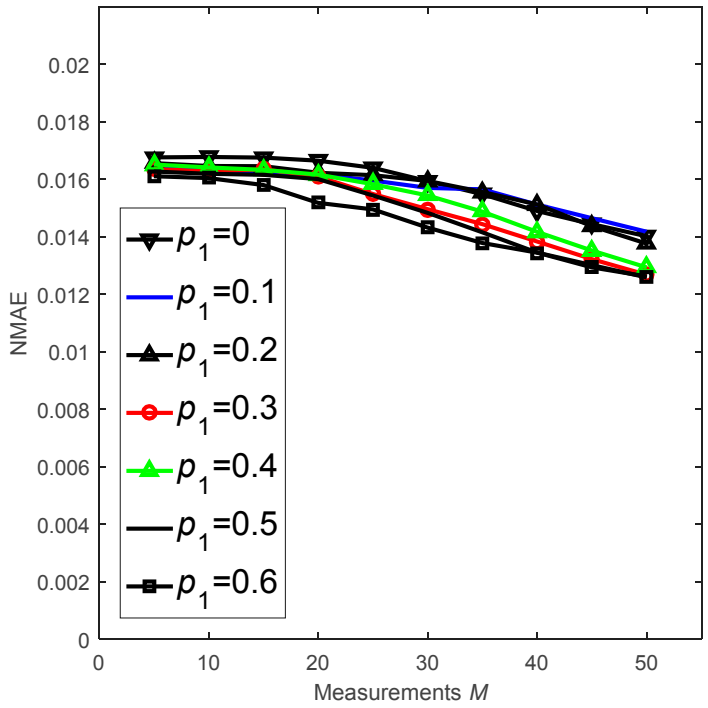

Figure 15. Performance of CS-DDSG with different value of p_1 and M.

6. Conclusions

In this paper, the data gathering problem is investigated in lossy WSNs using the simple but efficient proposed CS-DDSG algorithm that combines CS theory and DDS. Compared with other correlative and mainstream strategies, CS-DDSG balances the energy consumption and reconstruction performance effectively. In our proposed algorithm, nodes are selected to be source nodes with the probability p_1 to broadcast their packets. The neighboring nodes around the source nodes receive the broadcasting nodes and update their own packets, which are broadcasted with the probability p_2. Then, all receiving nodes forward their updated packets with the probability p_2. The process will be repeated a few times until there are no receiving nodes. Each receiving node receives only one packet. In this way, the numbers of transmissions and fusions are reduced, and the CS reconstruction accuracy is guaranteed. Moreover, the expression of the total number of transmissions and receptions is formulated via RGG. The simulation results and analysis validate that CS-DDSG outperforms the other algorithms in unreliable links.

In addition, we investigate how the measurements M, the packet loss p and the probability p_1 influence the performance of CS-DDSG. In future research, we plan to explore the possibility of temporal correlations of node readings. Another potential extension of this work is to more strictly demonstrate that the measurement matrix satisfies the RIP.

Author Contributions: Conceptualization, O.L. and G.L.; Methodology, C.Z. and M.L.; Software, G.L. and M.L.; Writing—Original Draft Preparation, C.Z.; Project Administration, G.L.; and Funding Acquisition, G.L. and O.L.

Funding: This work was supported in part by the National Science and Technology Major Projects of China under grant No. 2016zx03001010 and National Natural Science Foundation of China No. 61601516.

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

Nomenclature

WSNs	Wireless Sensor Networks
CS	Compressive Sensing
IoT	Internet of Things
RIP	Restricted Isometry Property
FFT	Fast Fourier Transform
OMP	Orthogonal Matching Pursuit
DDS	Distribute Data Storage
NMAE	Normalized Mean Absolute Error
RGG	Random Geometric Graph
PDF	Probability Density Function

References

1. Jesús, R.M.; José-Fernán, M.; Pedro, C.; Lourdes, L. Combining wireless sensor networks and semantic middleware for an internet of things-based sportsman/woman monitoring application. *Sensors* **2013**, *13*, 1787–1835.
2. Atzori, L.; Iera, A.; Morabito, G. The internet of things: A survey. *Comput. Netw.* **2010**, *54*, 2787–2805. [CrossRef]
3. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: A survey. *Comput. Netw.* **2002**, *38*, 393–422. [CrossRef]
4. Wu, M.; Tan, L.; Xiong, N. Data prediction, compression, and recovery in clustered wireless sensor networks for environmental monitoring applications. *Inf. Sci.* **2016**, *329*, 800–818. [CrossRef]
5. Zhou, M.; Fortino, G.; Shen, W.; Jobin, M.J.; Bhattacharyya, R. Guest editorial: Special section on advances and applications of internet of things for smart automated systems. *IEEE Trans. Autom. Sci. Eng.* **2016**, *13*, 1225–1229. [CrossRef]
6. Đurišić, M.P.; Tafa, Z.; Dimić, G.; Milutinović, V. A survey of military applications of wireless sensor networks. In Proceedings of the Mediterranean Conference on Embedded Computing (MECO), Bar, Montenegro, 19–21 June 2012; pp. 196–199.
7. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [CrossRef]
8. Baraniuk, R. Compressive sensing. *IEEE Signal Process. Mag.* **2007**, *56*, 4–5.
9. Luo, C.; Wu, F.; Sun, J.; Chen, C.W. Compressive Data Gathering for Large-scale Wireless Sensor Networks. In Proceedings of the International Conference on Mobile Computing and Networking, Beijing, China, 20–25 September 2009; pp. 145–156.
10. 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, New York, NY, USA, 23–27 May 2010; pp. 1–6.
11. Wang, W.; Garofalakis, M.; Ramchandran, K. Distributed Sparse Random Projections for Refinable Approximation. In Proceedings of the International Symposium on Information Processing in Sensor Networks, Cambridge, MA, USA, 25–27 April 2007; pp. 331–339.
12. Wu, X.; Xiong, Y.; Yang, P.; Wan, S.; Huang, W. Sparsest random scheduling for compressive data gathering in wireless sensor networks. *IEEE Trans. Wirel. Commun.* **2014**, *13*, 5867–5877.
13. Han, L.Y.; Eftekhari, A.; Wakin, M.B.; Rozell, C.J. The Restricted Isometry Property for Block Diagonal Matrices. In Proceedings of the Information Sciences and Systems, Baltimore, MD, USA, 23–25 March 2011; pp. 1–31.
14. Leinonen, M.; Codreanu, M.; Juntti, M. Sequential compressed sensing with progressive signal reconstruction in wireless sensor networks. *IEEE Trans. Wirel. Commun.* **2015**, *14*, 1622–1635. [CrossRef]
15. Nguyen, M.T.; Teague, K.A.; Rahnavard, N. CCS: Energy-efficient data collection in clustered wireless sensor networks utilizing block-wise compressive sensing. *Comput. Netw.* **2016**, *106*, 171–185. [CrossRef]
16. Kong, L.; Xia, M.; Liu, X.Y.; Wu, M.Y.; Liu, X. Data Loss and Reconstruction in Sensor Networks. In Proceedings of the IEEE INFOCOM, Turin, Italy, 14–19 April 2013; pp. 1654–1662.
17. Kong, Z.; Aly, S.; Soljanin, E. Decentralized coding algorithms for distributed storage in wireless sensor networks. *IEEE J. Sel. Areas Commun.* **2010**, *28*, 261–267.

18. Zeng, R.; Jiang, Y.; Lin, C.; Fan, Y.; Shen, X. A distributed fault/intrusion-tolerant sensor data storage scheme based on network coding and homomorphic fingerprinting. *IEEE Trans. Parallel Distrib. Syst.* **2012**, *23*, 1819–1830. [CrossRef]
19. Ren, Y.; Oleshchuk, V.; Li, F.Y. A Scheme for Secure and Reliable Distributed Data Storage in Unattended WSNs. In Proceedings of the IEEE Global Telecommuncations Conference, Miami, FL, USA, 6–10 December 2010; pp. 1–6.
20. Talari, A.; Rahnavard, N. CStorage: Distributed Data Storage in Wireless Sensor Networks Employing Compressive Sensing. In Proceedings of the IEEE GLOBECOM, Kathmandu, Nepal, 5–9 December 2011; pp. 1–5.
21. Yang, X.; Tao, X.F.; Dutkiewicz, E.; Huang, X.J.; Guo, Y.J.; Cui, Q.M. Energy-efficient distributed data storage for wireless sensor networks based on compressed sensing and network coding. *IEEE Trans. Wirel. Commun.* **2013**, *12*, 5087–5099. [CrossRef]
22. Gong, B.; Cheng, P.; Chen, Z.; Ning, L.; Gui, L.; Hoog, F.D. Spatiotemporal compressive network coding for energy-efficient distributed data storage in wireless sensor networks. *IEEE Commun. Lett.* **2015**, *19*, 803–806. [CrossRef]
23. Candes, E.J.; Tao, T. Decoding by linear programming. *IEEE Trans. Inf. Theory* **2005**, *51*, 4203–4215. [CrossRef]
24. Gupta, P.; Kumar, P.R. Critical Power for Asymptotic Connectivity in Wireless Networks. In Proceedings of the IEEE Conference Decision and Control, Tampa, FL, USA, 16–18 December 1998; pp. 1106–1110.
25. Elson, J.; Girod, L.; Estrin, D. Fine-grained network time synchronization using Reference broadcasts. *SIGOPS Oper. Syst. Rev.* **2002**, *36*, 147–163. [CrossRef]
26. Penrose, M. *Random Geometric Graphs*, 5th ed.; Oxford University Press: Oxford, UK, 2004; pp. 90–102, ISBN 9780198506263.
27. Hall, P. *Introduction to the Theory of Coverage Process*; John Wiley and Sons: Hoboken, NJ, USA, 1988; pp. 26–39, ISBN 9781584350675.
28. Yu, C.W. Computing subgraph probability of random geometric graphs with applications in quantitative analysis of ad hoc networks. *IEEE J. Sel. Areas Commun.* **2009**, *27*, 1056–1065.
29. Mo, L.; He, Y.; Liu, Y.; Zhao, J.; Tang, S.J.; Li, X.Y.; Dai, G. Canopy Closure Estimates with Greenorbs: Sustainable Sensing in the Forest. In Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 7), Berkeley, CA, USA, 4–6 November 2009; pp. 99–112.