Research Article

Network Coding-Based Real-Time Retransmission Scheme in Wireless Sensor Networks

Chang Zhu, Yanqin Zhu, and Lingzhi Li

School of Computer Science & Technology, Soochow University, Suzhou 215006, China

Correspondence should be addressed to Yanqin Zhu; yqzhu@suda.edu.cn

Received 26 September 2014; Revised 29 March 2015; Accepted 30 March 2015

Academic Editor: Amiya Nayak

Copyright © 2015 Chang Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In wireless sensor networks (WSNs), wireless links are unreliable and sensor nodes may be in sleep mode. Thus, many applications which require reliable broadcast cannot work properly if they lost some packets. In order to make sure every sensor node in the network receives all packets completely and correctly, retransmission of lost packets is indispensable. Many retransmission methods with network coding have been proposed but they do not catch coding opportunity. In this paper, real-time retransmission algorithm based on network coding (NCRR) is proposed to make the average number of transmissions as less as possible. During the transmission of original packets, we detect whether a certain time slot is suitable for retransmitting a coded packet based on recovery ratio. We analyze the number of transmissions with traditional retransmission and with network coding. Compared with existing approaches, simulation results show that our algorithm can effectively reduce the average number of transmissions and improve the transmission efficiency.

1. Introduction

Wireless sensor networks have been widely used in many scenarios. They can perceive and interact with the physical world for different purpose, such as military surveillance [1], habitat monitoring [2], structural monitoring [3], and medical applications [4]. The sensors in a WSN are expected to work for a long period of time once deployed, and it is hard to physically get them back to do some updates. However, updating the software running on those sensors or adding new function to those sensors [5] is often inevitable. Reprogramming the network needs to reliably transmit many packets to every sensor node in the network.

Network coding (NC) has been proposed in [6] for many years. It has become an efficient approach to improving the system throughput in wireless networks. The core ideal of network coding is allowing every node between a source and a destination to encode the content of multiple packets into a single packet using several techniques such as packet XORs and linear coding. Two network coding methods can be distinguished in the literature, namely, opportunistic network coding [7] and full network coding [8]. In XORs coding, a coded packet consists of the coding vector information and the encoded data. Thus, upon receiving a coded packet, the receiver knows which packets are encoded together and how to decode the packet with the available packets at the receiver.

In wireless sensor networks, it is usually necessary to update the code running on sensors, which requires reliable dissemination of large data to each sensor with energy efficiency. Because of sleep scheduling designed for energy efficiency, some sensors are in sleep mode. They cannot receive some packets at some time slots during data dissemination. At the same time, due to the unreliability of wireless links, a sensor may not successfully receive a packet even when it is in the active mode. Therefore, retransmission of such packets to those sensors is necessary, which consumes more energy and increases the delay of data dissemination.

In this paper, considering the sleep scheduling and unreliability of links, we proposed a real-time retransmission scheme with XORs coding. We aim at making each sensor in the network successfully receive the whole set of disseminated packets with minimum number of retransmissions. Thus, energy consumption can be reduced and throughput can be improved. In order to achieve this goal, it is important to maximize the expected number of active sensors that can decode out one lost packet at each time slot in the recovery...
process. The contribution of our work is summarized as follows:

(i) Two factors (e.g., sleep scheduling and link unreliability) have been considered in WSNs. We propose a real-time retransmission algorithm based on network coding with the detection of recovery ratio to make sure every active sensor node can recover one lost packet from the retransmitted packet.

(ii) We analyze the influence of sensors’ sleep probability and links’ packet loss probability on the average number of retransmissions. Then we still implement our algorithm on NS2 simulator. The results show that our scheme can reduce the average number of retransmissions.

The rest of this paper is organized as follows. In Section 2, the related work has been presented. In Section 3, we present the system model and parameters. We introduce our real-time retransmission method in Section 4. We evaluate the performance of our algorithm in Section 5. Conclusion could be found in Section 6.

2. Related Work

Network coding has caused wide public concern over the recent years and was used in different types of information networks. In a unicast network with multiple hops, Pfletschinger et al. [9] proposed a random linear network coding scheme based on opportunistic listening. A data dissemination protocol has been proposed in WSNs by Hou [10]. The protocol used adaptive network coding to reduce broadcast traffic in the process of code update. Shwe et al. [11] further researched the protocol proposed by Hou and improved the performance. They have deployed an efficient neighbor discovery protocol to find out all neighbors of a node for reducing power consumption in the network.

Network coding with XORs operation in wireless broadcast has been studied widely. Lu et al. proposed ONCSB and ONCMB [12] algorithms in broadcast transmission to improve throughput. ONCBT algorithm has been proposed in [13]; it led to higher transmission efficiency and lower packet transmission delay. Nguyen et al. [14] showed the advantage of the proposed XORs coding scheme over the traditional wireless broadcast in the bandwidth efficiency through simulation and theoretical analysis. Network coding is also used in mobile wireless sensor network [15] to adapt for the dynamics velocity of mobile nodes. However, all the above algorithms cannot be used in wireless sensor network, because they did not consider sleep scheduling.

As the computation ability and memory of sensor nodes are very limited, it is not suitable to do complicated calculations. Since the complexity of linear encoding and decoding is very high, it is more appropriate to use XORs operation whose encoding and decoding operations are much simpler in WSNs.

To achieve a reliable transmission, all the receivers must make sure that they have received all information correctly. Since wireless communication links are unreliable in general, the reliability of packet delivery is achieved through packet retransmission using automatic repeat request (ARQ) [16] or its combinations with forward error correction (FEC), known as the hybrid automatic repeat request (HARQ) [17]. However, both schemes retransmit lost packets one by one, and the number of receivers which can benefit from each retransmission is small. This leads to more retransmissions and low system bandwidth and energy efficiency. Thus they cannot be used as retransmission method in wireless sensor networks.

In wireless sensor networks, sleep scheduling has been introduced in sensor nodes for the sake of saving energy which is the difference from other networks. In [18, 19], sleep scheduling has been considered when retransmitting lost packets. Both of their algorithms can reduce the number of retransmissions and energy consumption. However, the number of packets that has been transmitted is constant when retransmitting a coded packet. At the retransmission timeslot, network maybe cannot reach the maximum throughput. That is to say, not every active sensor node can recover one lost packet from the retransmitted coded packet. To solve this problem we proposed an algorithm that can calculate the throughput when retransmitting a coded packet. If throughput satisfies requirements, then we will choose lost packets to encode.

3. System Model and Parameters

3.1. System Model. In this paper, the network model is a clustering wireless sensor network. In the network, there is a sink node and several clusters. For simplicity, we just talk about one cluster.

(1) One cluster has a cluster head (CH) and m ordinary sensor nodes. Let the set \( R = \{ R_1, R_2, \ldots, R_m \} \) denote sensor nodes. The cluster head broadcasts a set \( P = \{ P_1, P_2, \ldots, P_n \} \) of \( n \) packets to \( R \).

(2) CH knows the receiving state of all sensor nodes. This can be accomplished by using Acknowledgments/Negative Acknowledgments (ACK/NCK). The receiving states include (a) whether a packet is lost or not; (b) identity of lost packets and corresponding nodes. For simplicity, we assume all the ACK/NCK are instantaneous and never lose.

(3) Packet lost rate at sensor node \( R_i \) follows the Bernoulli distribution with parameter \( l_i \). In addition, the packets lost at different sensor nodes are uncorrelated.

(4) Each sensor node \( R_i \) chooses to sleep with probability \( s_i \) which follows the Bernoulli distribution.

**Definition 1** (sleep vector \( v_j \)). Each sensor node \( R_i \) produces a sleep vector \( v_j = \{ a_j \mid a_j \in \{0,1\} \} \). If \( R_i \) chooses to sleep at time slot \( j \), \( a_j \) is 0. Otherwise, \( a_j \) is 1.

**Definition 2** (sleep matrix \( V \)). This matrix consists of \( v_j \). Each row represents the sleep state of \( R_i \). Columns represent time slots. Element \( v_{ij} \) in \( V \) represents whether \( R_i \) is active or not at time slot \( j \).
Definition 3 (receiving state matrix $S$). This matrix will change according to ACK/NCK and $V$. In this matrix, rows represent the receiving state of sensor nodes, and columns represent the receiving state of broadcasting packets. When the broadcasting packets are successfully received in any sensor node, the corresponding position $b_{ij}$ in the matrix will be marked 0, or else it is 1.

3.2. Basic Idea of Network Coding. In this paper, we use network coding with XOR operation to accomplish retransmission under the condition of sleep scheduling. As shown in Figure 1, it is an example of XOR network coding. There are one CH and four sensor nodes.

At the beginning, CH sends four packets A, B, C, and D to all sensor nodes. After broadcasting, due to either sleep scheduling or link unreliability, packets A, D are lost in node $R_1$; Packet A is lost in node $R_2$; Packet C is lost in node $R_3$; Packet D is lost in node $R_4$. To make sure every sensor node works properly, retransmission approach is needed.

For traditional retransmission approach, CH needs to rebroadcast A, B, C, and D packets in sequence. However, network coding with XOR operation considers the state of all lost packets at each sensor node to adjust sending scheme. In Figure 1, we just need to retransmit two times. Firstly, we can calculate $A \oplus B \oplus C$ (encoding process) and broadcast the combined packet. Then, node $R_1$ can recover packet A via $(A \oplus B \oplus C) \oplus (B \oplus C) = A$ (decoding process); sensor nodes $R_2$ and $R_3$ can recover packets A and C in the same way. Secondly, CH should broadcast packet D. Then, $R_1$ and $R_4$ can recover packet D. Apparently, network coding combines different lost packets and reduces the number of retransmissions.

4. Real-Time Retransmission Method

4.1. Description of NCRR Algorithm. NCRR algorithm adopts synchronous feedback mechanism and updates matrix $S$ dynamically. When a sensor node receives a coded packet, it may recover what it wants. The coding opportunity will change at every time slot. However, because of link unreliability and sleep scheduling, not every sensor node can recover a lost packet from the retransmission. So we propose a detection method to make sure each active sensor node can recover a lost packet from the coded packet. If we make detection in every timeslot, the cost is very large. We choose to make detection every $T$ timeslots. Algorithm 1 is the pseudocode of our algorithm.

Firstly, CH broadcasts an original packet. If the active sensor nodes receive the packet, they will send ACKs to CH. Then, CH updates matrix $S$ which presents the receiving state of every sensor node. Secondly, we should make a decision whether to execute detection mechanism based on detection interval $T$. Next, a coded packet or an original packet will be transmitted. In that procedure, lost packets should be selected based on matrices $S$ and $V$. The details about how to choose lost packets will be described in the next paragraph. If the result returned by encoding algorithm is null, we should broadcast original packets at next timeslot. At the same time, we will prolong the detection interval. It can be considered as a punishment, because we believe that the network will not be suitable for encoding during the recent timeslots. Otherwise, the result is a coded packet which will be retransmitted at next timeslot.

4.2. Encoding Scheme. As shown in Figure 1, network coding can reduce the number of retransmissions. However, how to choose lost packets to combine a coded packet is the key point. It can directly impact the preference of wireless sensor network.

Definition 4. In order to detect whether to encode lost packets, one defines $\theta$ as recovery ratio. Consider

$$\theta = \frac{k}{\sqrt{V}}.$$
where \( k \) is the number of packets which can be decoded by active sensor nodes. \( S' \) is the submatrix of \( S \) and \(| S' | \) is the number of active nodes. If \( \theta = 1 \), every active sensor node can recover a wanted packet. Otherwise, we will not execute retransmitting and return null. Algorithm 2 is the pseudocode of encoding algorithm.

Our encoding algorithm is based on hash search. The algorithm has three steps: (1) calculate hash value according to hash function; (2) create hash table based on hash value; (3) choose packets which satisfy requirements to encode into a retransmission packet. Next we will introduce encoding algorithm with a concrete example.

Table 1(a) is the sleep matrix \( V \) and Table 1(b) is the receiving state of each sensor. At this point, CH has broadcasted seven packets. Then, it needs to retransmit the lost packets in time slot \( t \). Searching the matrix \( V \), we know the active node set \( Q \) is \( \{R_1, R_2, R_4, R_5, R_6\} \). Next we can encode lost packets based on \( S' \) which is a submatrix of \( S \) shown in Table 2.

**Step 1.** Calculate hash value of the lost packets. We regard each column as a binary sequence. The hash function is formula (2), where \( M \) is the number of active sensor nodes. The relationship between hash value and binary sequence is one-to-one correspondence. Consider

\[
H(i) = \sum_{j=1}^{M} 2^{j-1} b_{ij}, \quad i = 1, 2, \ldots, N. \tag{2}
\]

**Step 2.** Establish a hash table which is shown in Table 3 and sort the hash value in descending order. First row denotes hash value of lost packets; second row is the number of packets who have the same hash value; third row presents lost packets.

**Step 3.** Select packets which satisfy (1) to produce a retransmitting packet. We denote \( E \) as a set of packets which need encoding. First, we choose the biggest hash value and add corresponding packet to set \( E \). Second, we select hash values which are in the neighborhood of set \( E \) until neighborhood is empty. Finally, it is time to make a decision whether to return a coded packet or a null value based on \( \theta \). We define \( Y \) as neighborhood of \( H(i) \). The formula of neighborhood is as follows:

\[
Y = \left\{ \sum_{j} 2^{j-1} \mu_j \mid \mu_j \in \{0, 1\}, \exists \mu_j \neq 0 \right\}
\]

subject to \( j \in \{1, 2, \ldots, |S'|\} \), \( \forall s'_{ij} = 0 \)

**Algorithm 2: Encoding algorithm.**

\[
\begin{align*}
\text{Input:} & \quad \text{matrix } S \text{ and } V, \text{ timeslot } t \\
\text{Output:} & \quad \text{a coded packet or NULL} \\
(1) & \quad w = t + 1 \\
(2) & \quad \text{For } j = 1 \text{ to } m \text{ //select active nodes} \\
(3) & \quad \text{If } y_{wj} = 1 \\
(4) & \quad \text{Q.add}(R_j) \text{ //Q is a set of active nodes} \\
(5) & \quad \text{Endif} \\
(6) & \quad \text{Endfor} \\
(7) & \quad \text{Produce submatrix } S' \text{ based on } Q \text{ and } S \\
(8) & \quad \text{For } i = 1 \text{ to } N \text{ //calculate hash value} \\
(9) & \quad H(i) = \sum_{j=1}^{M} 2^{j-1} b_{ij} \\
(10) & \quad \text{Endfor} \\
(11) & \quad \text{Select packet to encode based on } H(i) \\
(12) & \quad \theta = k / |S'| \\
(13) & \quad \text{If } (\theta = 1) \\
(14) & \quad \text{Return coded packet} \\
(15) & \quad \text{Else} \\
(16) & \quad \text{Return null} \\
(17) & \quad \text{End If}
\end{align*}
\]

**Table 1:** (a) Sleep matrix in CH. (b) Received state of each sensor.

|       |   |   |   |   |   |   |   |
|-------|---|---|---|---|---|---|---|
|       | \( t_1 \) | \( t_2 \) | \( t_3 \) | \( t_4 \) | \( t_5 \) | \( t_6 \) | \( t_7 \) |
| \( R_1 \) | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| \( R_2 \) | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| \( R_3 \) | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| \( R_4 \) | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| \( R_5 \) | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| \( R_6 \) | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| \( R_7 \) | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |

|       | \( P_1 \) | \( P_2 \) | \( P_3 \) | \( P_4 \) | \( P_5 \) | \( P_6 \) | \( P_7 \) | \( P_8 \) |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| \( R_1 \) | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| \( R_2 \) | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| \( R_3 \) | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| \( R_4 \) | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| \( R_5 \) | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| \( R_6 \) | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| \( R_7 \) | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |

**Table 2:** Submatrix of \( S \).

|       | \( P_1 \) | \( P_2 \) | \( P_3 \) | \( P_4 \) | \( P_5 \) | \( P_6 \) | \( P_7 \) | \( P_8 \) |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| \( R_1 \) | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| \( R_2 \) | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| \( R_3 \) | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| \( R_4 \) | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| \( R_5 \) | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| \( R_6 \) | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 |

**Table 3:** Hash table.

|        |       |       |       |       |       |       |       |
|--------|-------|-------|-------|-------|-------|-------|-------|
| Hash value | 21 | 20 | 18 | 13 | 12 | 10 | 2 |
| Number of sensors | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| \( P_1 \) | \( P_7 \) | \( P_1 \) | \( P_3 \) | \( P_2 \) | \( P_6 \) | \( P_4 \) | \( P_5 \) |
|        |       |       |       |       |       |       |       |
neighbourhood of 21. Then, the neighbourhood is empty. The sum of \( H(7) \) and \( H(4) \) is 31 which is equal to \( 2^{|S'|} - 1 \). So, the value of \( k \) is equal to \(|S'|\) and \( \theta \) is 1. Encoding algorithm will return an encoded packet \((p_6 \oplus p_7)\).

4.3. Mathematic Analysis of NCRR. We will analyze the impact of loss probability and sleep scheduling on network coding gain. We define the average number of transmissions as the analysis parameter. Let \( \omega_n \) denote the average number of transmissions with traditional retransmission approach and \( \omega_t \) is the average number of transmission with NCRR.

**Lemma 5.** The average number of transmissions without network coding required for transmitting sufficient large \( n \) packets to \( m \) receivers is

\[
\omega_n = \sum_{i_1, i_2, \ldots, i_m} (-1)^{\sum_{i=1}^{m} i - 1} 1 - \prod_{j=1}^{m} |u_j|,
\]

where \( i_1, i_2, \ldots, i_m \in \{0, 1\} \) and \( \exists j \neq 0, u_j = 1 - (1 - l_j)(1 - s_j) \).

**Proof.** According to [14], every node is active at any time slot. The average number of transmissions when we transmit sufficient large \( n \) packets to \( m \) receiver is

\[
\omega_t = \sum_{i_1, i_2, \ldots, i_m} (-1)^{\sum_{i=1}^{m} i - 1} 1 - \prod_{j=1}^{m} |u_j|,
\]

where \( l_j \) is lost rate. However, except link unreliability, sensor nodes may be in sleep mode during retransmission process. So probability that sensor nodes can successfully receive a coded packet is \((1 - l_j)(1 - s_j)\). That is to say, the probability that sensor nodes lose a packet is \( 1 - (1 - l_j)(1 - s_j) \). Therefore, the number of transmissions without network coding is \((4)\).

**Lemma 6.** The average number of transmissions with XOR network coding required for transmitting sufficient large \( n \) packets to \( m \) receivers is

\[
\omega_b = \frac{1}{1 - \max_{i \in \{1, 2, \ldots, m\}} \{u_i\}},
\]

where \( u_i = 1 - (1 - l_i)(1 - s_j) \).

**Proof.** According to [14], every node is active at any time slot. The average number of transmissions with XOR network coding when we transmit sufficient large \( n \) packets to \( m \) receiver is \( 1 / (1 - \max_{i \in \{1, 2, \ldots, m\}} \{l_i\}) \) where \( l_i \) is lost rate. Thus, considering sleep scheduling, the number of transmissions with XOR network coding is \((6)\).

We define network coding gain as

\[
\varphi = \frac{\omega_n - \omega_b}{\omega_n}.
\]

5. Performance Evaluation

In this section, we test the performance of our proposed NCRR algorithm on NS2 simulator. In our simulation, a cluster-based WSN is randomly generated with the fixed number of sensors. There is a single hop between the sink node and cluster heads. In one cluster, the distance is also one hop between a cluster head and all ordinary sensor nodes. The whole distribution mainly has two stages. Firstly, the sink node sends packet to cluster heads. At the same time, all the sensor nodes keep sleeping to save energy. If one of the cluster heads completely receives packets, it will wake up all sensor nodes in its cluster. Then it will do the same work like the sink node. The sleep pattern is a fixed sleep mode. Each node in one cluster will produce a sleep vector just including 0 and 1 according to its sleep probability when network is initialized. They will send it to the cluster head and choose to sleep according to the vector. We divide packets into several batches. Each batch has \( M \) packets. We compare NCRR algorithm with three schemes: (1) traditional retransmission with ARQ; (2) low power data retransmission scheme based on network coding [19] (LPDR); (3) network coding-based loss recovery [20] (NCLR).

The number of packets in one batch is 25, \( l_i = 0.2 \), and \( s_j = 0.05 \). The number of sensor nodes in one cluster varies in the range of \( [4, 20] \). As shown in Figure 2, when the number of sensor nodes increases, the average number of transmissions increases too. LPDR and NCLR are as good as our scheme and better than traditional scheme. But our scheme has the best performance. It needs the least average number of transmissions.

Sleep scheduling is a feature of WSNs. Now we study its effect on network coding gain. We compare the network coding gain obtained through simulation with the analytical network coding gain shown in Figure 3. We assume \( M = 50 \); \( l_i = 0.2 \) (\( 2 \leq i \leq 5 \)) and \( l_i = 0.3 \) (\( 6 \leq i \leq 10 \)); \( s_1 \) varies in the range of \([0.1, 0.7] \) and \( s_j = s_1 + 0.04 \) (\( 2 \leq i \leq 5 \)), and \( s_j = s_1 + 0.08 \) (\( 6 \leq i \leq 10 \)). The result shows that simulation is very close to analytical result. We also can conclude that if many sensor nodes are in sleep mode, the number of transmissions will increase. In other words, the higher sleep probability cannot necessarily result in better performance.

In order to reduce number of retransmissions, detection method is proposed to make sure that every active sensor node can recover a lost packet from one retransmission. We
assume $M = 50, I_i = 0.1$, and $s_i = 0.1$. Figure 4 shows that the number of sensor nodes which can recover a lost packet from one retransmission is the largest with our algorithm, because we just retransmit encoded packets only when the recovery ratio satisfies requirement. It cannot reach the largest value, because, even as a sensor node in active mode, it may not receive the retransmitted packet.

The cache of a sensor node is sometimes very limited and the number of packets in one batch should be adjusted to different conditions. We study the impact of $M$ on the average number of transmissions with $l_i = 0.1$, $s_i = 0.2$. When $M$ changes from 15 to 100, we test the average number of transmissions. The result in Figure 5 shows that no matter how $M$ changes, our scheme is the most stable. After transmitting one batch of packets, LPDR and NCLR begin to recover lost packets. They do not check how many sensor nodes can recover a lost packet. So they cannot make sure that every active node can recover a lost packet. And they may need more transmission times to complete the whole dissemination. Our retransmission algorithm is not strictly based on one batch of packets. In order to make full use of the opportunity of retransmission, we could combine packets in two batches together. So our algorithm has lower average number of transmissions.

We still study the average delays of four schemes in Figure 6. Obviously, the bigger the lost probability, the greater the average delay. The traditional scheme has the maximum delay, because it needs the largest number of transmissions. The other three schemes have similar performance. But LPDR and NCLR are a little better than our scheme, because we add detection mechanism which needs extra time.

6. Conclusion

In this paper, we propose a NCRR algorithm combining with sleep scheduling in WSN. Our retransmission algorithm is not strictly based on one batch of packets. Retransmitting a coded packet is embedded in the process of transmission of original packets. We make sure that every active sensor node can recover a lost packet from a retransmission through the value of $\theta$. Simulation results reveal that, compared with existing approaches, the approach in this paper can effectively reduce the average number of transmissions and advance the retransmission efficiency.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.
Acknowledgments

This paper is supported by the National Science Foundation of China under Grants no. 61070170 and no. 61373164 and Research Project of Jiangsu Province under Grant no. BY2013030-06.

References

[1] T. He, S. Krishnamurthy, J. A. Stankovic et al., “Energy-efficient surveillance system using wireless sensor networks,” in Proceedings of the 2nd International Conference on Mobile Systems, Applications, and Services (MobiSys ’04), pp. 270–283, ACM, 2004.

[2] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, “An analysis of a large scale habitat monitoring application,” in Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys ’04), pp. 13–24, ACM, November 2004.

[3] N. Xu, S. Rangwala, K. K. Chintalapudi et al., “A wireless sensor network for structural monitoring,” in Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys ’04), pp. 214–226, ACM, November 2004.

[4] R. K. Ganti, P. Jayachandran, T. F. Abdelzaher, and J. A. Stankovic, “Satire: a software architecture for smart attire,” in Proceedings of the 4th International Conference on Mobile Systems, Applications and Services (MobiSys ’06), pp. 110–123, ACM, Uppsala, Sweden, June 2006.

[5] C.-J. M. Liang, R. Musaloiu-E, and A. Terzis, “Typhoon: a reliable data dissemination protocol for wireless sensor networks,” in Wireless Sensor Networks, vol. 4913 of Lecture Notes in Computer Science, pp. 268–285, Springer, Berlin, Germany, 2008.

[6] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, “Network information flow,” IEEE Transactions on Information Theory, vol. 46, no. 4, pp. 1204–1216, 2000.

[7] S. Katti, H. Rahul, H. Hu et al., “XORs in the air: practical wireless network coding,” ACM SIGCOMM Computer Communication Review, vol. 36, no. 4, pp. 243–254, 2006.

[8] J.-S. Park, M. Gerla, D. S. Lun, Y. Yi, and M. Médard, “Codecast: a network-coding-based ad hoc multicast protocol,” IEEE Wireless Communications, vol. 13, no. 5, pp. 76–81, 2006.

[9] S. Pfletschinger, M. Navarro, and C. Ibars, “Energy-efficient data collection in WSN with network coding,” in Proceedings of the IEEE GLOBECOM Workshops (GC Wkshps ’11), pp. 394–398, IEEE, Houston, Tex, USA, December 2011.

[10] I.-H. Hou, “Adapcode: adaptive network coding for code updates in wireless sensor networks,” in Proceedings of the 27th Conference on Computer Communications (INFOCOM ’08), IEEE, April 2008.

[11] H. Y. Shwe, X. Jiang, and S. Horiguchi, “Efficient network coding for power saving in wireless sensor networks,” in Proceedings of the 15th Asia-Pacific Conference on Communications (APCC ’09), pp. 36–39, IEEE, Shanghai, China, October 2009.

[12] J. Lu, C.-K. Wu, S. Xiao, and R. Zhang, “Efficient broadcast transmission algorithms based on opportunistic network coding,” Journal on Communications, vol. 33, no. 1, pp. 64–70, 2012.

[13] J. Lu, S. Xiao, and C.-K. Wu, “Opportunistic network coding based delay-sensitive broadcast transmission algorithm,” Acta Electronica Sinica, vol. 39, no. 5, pp. 1214–1219, 2011.

[14] D. Nguyen, T. Tran, T. Nguyen, and B. Bose, “Wireless broadcast using network coding,” IEEE Transactions on Vehicular Technology, vol. 58, no. 2, pp. 914–925, 2009.

[15] L. Li, S. Zhang, Z. Yang, and Y. Zhu, “Data dissemination in mobile wireless sensor network using trajectory-based network coding,” International Journal of Distributed Sensor Networks, vol. 2013, Article ID 852472, 10 pages, 2013.

[16] M. Naijoh, S. Sampe, N. Morinaga, and Y. Kamio, “ARQ schemes with adaptive modulation/TDMA/TDD systems for wireless multimedia communication services,” in Proceedings of the International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC ’97), pp. 709–713, September 1997.

[17] R. G. Kermode, “Scoped hybrid automatic repeat request with forward error correction (SHARQFC),” ACM SIGCOMM Computer Communication Review, vol. 28, no. 4, pp. 278–289, 1998.

[18] X. Wang, J. Wang, and Y. Xu, “Data dissemination in wireless sensor networks with network coding,” EURASIP Journal on Wireless Communications and Networking, vol. 2010, Article ID 465915, 2010.

[19] S. Long, J. Liu, G. Jiang, and Y. Gao, “Low power consumption data retransmission strategy in WSNs based on network coding,” Transducer and Microsystem Technologies, vol. 32, no. 8, pp. 35–38, 2013.

[20] Z.-H. Zhou and L. Zhou, “Efficient loss recovery based on network coding in multicast networks,” Journal of Electronics and Information Technology, vol. 34, no. 8, pp. 1962–1967, 2012.