Research Article

A Strategy Based on Cooperative Transmission for Minimizing Delivery Delay in WSN

Qingyu Yang, Xinyu Yang, Yu Wang, and Dou An

School of Electronic and Information Engineering, Xian Jiaotong University, No. 28 Xianning West Road, Xian, Shaanxi 710049, China

Correspondence should be addressed to Xinyu Yang; yxyphd@mail.xjtu.edu.cn

Received 13 June 2013; Revised 13 September 2013; Accepted 27 September 2013

Academic Editor: Prabir Barooah

Copyright © 2013 Qingyu Yang 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.

To achieve the purpose of energy conservation, various sleep scheduling approaches, such as duty cycle, are applied in wireless sensor networks (WSN). However, the duty-cycle mechanism results in data delivery latency, which is critical to monitor applications. To minimize the delay caused by sleeping nodes in the transmission path, we propose to “hop over” the sleeping nodes based on the range extension of cooperative transmission (CT). The transmission delay models for the random duty-cycled WSN and optimized fixed duty-cycled WSN under cooperative operation are formulated, and an algorithm named (delay-tolerant cooperative transmission DTCT) is presented for the selection of transmission modes to avoid waiting for the sleeping nodes to wake up. The energy consumption model under direct transmission (DT) and CT mode is also presented. Theoretical analysis shows that sleep latency can be greatly reduced in the cooperative scheme, and it is validated by simulations that it outperforms the traditional store-and-forward (DT) mode in delivery latency. Especially, CT reduces 67% and 14.3% of the transmission delay in random and optimized fixed low duty-cycled WSN, respectively, and DTCT algorithm saves energy by 11.29% in random low duty-cycled WSN.

1. Introduction

A wireless sensor network (WSN) is composed of a large number of intelligent sensor nodes. How to save the limited energy of the battery-powered sensors has been a focus of investigation [1–3]. Idle listening of radio is the major source of battery drain. Most existing studies propose to reduce energy consumption by turning off the transceivers and scheduling the nodes to wake up periodically, which is known as duty-cycle scheme [4, 5].

Scheduling of the nodes, though benefits energy conservation, causes transmission delay in WSN. For example, a transmitter typically adopts a store-and-forward mode when the next node is in sleep state. This means that a message packet destined to the next sleeping node, when it arrives, will be buffered and will wait to be transmitted until the node wakes up. If the packet meets a large number of sleeping nodes on the path, the sleep latency will greatly increase the end-to-end transmission delay.

Therefore, developing efficient methods to minimize the sleep latency caused by the store-and-forward mechanism is of great importance. Many efforts have been made to address this issue [6, 7]. In [8, 9], geographic information routing is exploited to find the lowest delay path. Swain et al. [10] constructed a broadcast tree from sink to each node and arranged the state of nodes in the tree to reduce the transmission delay. Cooperative transmission (CT) is a novel model introduced recently that allows combining partial messages to decode a complete message [11, 12]. Under the same bit error rate (BER), the required signal noise ratio (SNR) is lower in CT due to the diversity gain than in direct transmission without diversity. Thus, the signal can be decoded at a farther distance and the reduced SNR requirement can be used to extend the transmission range.

We propose here to minimize the sleep latency using CT strategy. Briefly, a receiver, when in sleep state, will be hopped over, and the packet is directly delivered to the active node two hops away from the sender. The sleep latency on the path can thus be avoided under this cooperative mode. In this study, transmission delay models for random and fixed duty-cycled WSN under DT and CT operations are formulated, respectively. In random duty-cycled WSN, we assume that
each node is able to predict the status of its neighbors by pseudorandom number generation (PRNG) algorithm [13]. Specifically, each node in the network needs to know the seeds used by its neighbors in their pseudorandom generators; the details of the prediction process will be discussed in Section 3.2.3. Theoretical analysis and numerical simulations are performed to evaluate the efficiency of the models in reducing sleep latency on the path. An algorithm which adopts the CT strategy is presented for the aim of minimizing the delivery delay by avoiding waiting for sleeping nodes to wake up. The algorithm is named (delay-tolerant cooperative transmission DTCT). The energy consumption model of DT and CT mode is also presented.

The remainder of this paper is organized as follows. Section 2 briefly reviews the background of the duty-cycle model and derives the delay model under the store-and-forward mode. In Section 3, the delay and energy consumption model in random duty-cycle scenario and optimized fixed duty-cycle scenario under the cooperative mode are formulated and the DTCT algorithm is presented. Section 4 presents the experimental results. Section 5 is a review of the related work. Finally, we give our conclusions in Section 6.

### 2. Delay Model under the Store-and-Forward Mode

#### 2.1. Duty-Cycle Model

Assume that all the nodes in WSN operate in two states, sleep and active, and the state of each node is independent. A duty cycle is the percent of time that a node spends in an active state as a fraction of the total time under consideration. The node switches between the two states periodically. For simplicity, the node state is expressed with variable $S_n^i$ as follows:

$$S_n^i = \begin{cases} 
0, & \text{node } i \text{ in sleep state at the } n \text{th time slot}, \\
1, & \text{node } i \text{ in active state at the } n \text{th time slot}. 
\end{cases} \quad (1)$$

Note that nodes can send and receive messages or listen to the channel in active state, whereas they shift to sleep state and meanwhile turn off their radio units when the active time slot elapses. Here, we assume the sleeping and active time slots are denoted as random values $w$ and $d$ uniformly distributed in the range of $[1, S]$ and $[1, K]$, respectively, where $S$ and $K$ are the maximum sleeping and active time, respectively.

The probability of the node in active state is thus derived as follows:

$$P_1 = \frac{E(w)}{E(d) + E(w)} = \frac{\sum_{w=1}^{S} w \times Pr(T_1 = w)}{\sum_{d=1}^{K} d \times Pr(T_0 = d) + \sum_{w=1}^{S} w \times Pr(T_1 = w)} \quad (2)$$

where $E(w)$ and $E(d)$ represent the mean time of active state and sleep state; $T_0$ and $T_1$ represent the duration of two states, respectively. The probabilities of active state and sleep state are expressed as $Pr(T_1 = w)$ and $Pr(T_0 = d)$, respectively. $\rho = (S + 1)/(K + 1)$ is the active factor.

Similarly, the probability of node in sleep status is given as follows:

$$P_0 = 1 - P_1 = \frac{1}{1 + \rho}. \quad (3)$$

#### 2.2. Delay Model under Store-and-Forward Mode

Store-and-forward is a frequently used information transmission mechanism in modern communication systems, in which message packets are transmitted from a starting node to a destination node with one or several intermediate nodes in between. A packet is received and stored by an intermediate node, and a copy is sent to the destination node. If the intermediate node is connected to more than one node, it sends a copy of the packet to each node that leads to the destination. In duty-cycled WSN, the packet will be stored if the intermediate node is in sleep state and will wait to be transmitted when the node wakes up. This process is also called direct transmission (DT) in this paper.

The delay model under the store-and-forward mode can be derived as follows. For any node $v_i$ on the path $R_k = \{v_1, v_2, \ldots, v_j, v_{j+1}, \ldots, \text{sink}\}$, message packets will be buffered when the next-hop node $v_{j+1}$ is in sleep state and will wait to be sent until $v_{j+1}$ wakes up. The average sleep latency of $v_i$ is as follows:

$$\overline{SL}^{\text{DT}} = \sum_{k=1}^{K} k \times Pr(WT_{i+1} = k), \quad (4)$$

where $Pr(WT_{i+1} = k)$ is the probability of $v_i$ waiting for $k$ time slots until $v_{j+1}$ wakes up. It can be expressed as follows:

$$Pr(WT_{i+1} = k) = \sum_{j=1}^{K-k} \frac{1}{K} \times \frac{\rho}{1 + \rho} \times \frac{2}{S+1} \quad (5)$$

where $P_{10}(\cdot)$ is transfer probability of node $v_{j+1}$ switching from active state to sleep state. Figure 1 illustrates the time schedule that node $v_i$ receives packet at time $t = 0$ and wait $k$ time slot if the next node $v_{j+1}$ is active at $j$ slots ago.
3. Delay and Energy Consumption Model under CT Mode

3.1. Basic Idea. In WSN, CT is widely used to improve data transmission through cooperation among network nodes, which takes advantage of the broadcast nature of wireless channels. Because of the diversity gain, CT can achieve higher quality transmission while lower BER at receivers; thus, the required SNR is lower than that in direct transmission without diversity under the same BER. Hence, the signal can be decoded at a farther distance with diversity gain and the reduced SNR requirement can be used to extend the transmission range.

Accordingly, CT is used here to tackle the transmission delay described in Section 2. Briefly, if the next node is in sleep state, the sender will collaborate with relays and forward the packet to the node after the next which is in active state, and there is no need to wait for the next sleeping node to wake up. In this way, the transmission range can be greatly extended. As illustrated in [14], when two neighboring nodes participate in the cooperative transmission, the range is 2.71 times that in direct transmission. This range extension can evidently reduce the sleep latency in transmission path. In this section, we consider to minimize the delivery delay by introducing CT in two scenarios: random duty-cycled WSN and optimized fixed duty-cycled WSN. Besides, we also proposed the energy consumption model under DT and CT mode.

3.2. Random Duty-Cycle Scenario

3.2.1. Delay Model. Random duty-cycle schedule is mentioned in Section 2, and we assume that the traffic in WSN is low. Thus, the delay in the route can be considered to result from the latency of the nodes in sleep state. Meanwhile, the queuing latency and collision on each node due to congestion can be neglected. In addition, each node delivers message in one time slot and the value of the slot is regular.

In the transmission path $R_h$, when a node $v_1$ prepares to report to the sink, the shortest route can be found as $R_h = \{ v_1, v_2, \ldots, v_i, v_{i+1}, \ldots, \text{sink} \}$. Any node $v_i$ that receives packets to forward has to wait $k$ time slots if the next node $v_{i+1}$ is in sleep state. To reduce the sleep latency, $v_i$ can select relays from its neighboring nodes and send packets directly to the active node $v_{i+2}$ which is two hops away from sender $v_i$. This process is shown in Figure 2.

There will be three cases that the transmitter $v_i$ meets: (i) the next node $v_{i+1}$ is in sleep state while the next-next node $v_{i+2}$ is active; (ii) $v_{i+1}$ and $v_{i+2}$ are both sleeping, but $v_{i+2}$ wakes up earlier than $v_{i+1}$; (iii) $v_{i+1}$ and $v_{i+2}$ are both sleeping while $v_{i+2}$ wakes up earlier. For the first two cases, CT strategy can be used to deal with the sleep latency. For the last case, the transmitter can only send the packet to the next hop in a store-and-forward manner. The delay models for the three cases are as follows.

Case 1. When a node $v_i$ receives a packet at $t = 0$, it has to wait $k$ time slots for the sleeping $v_{i+1}$ to wake up. Collaborating with relays, $v_i$ can forward the packet to $v_{i+2}$ which is awake.

As a result, $k$ time slots are saved. Figure 3 gives the sleep schedule of Case 1. The probability of this case is expressed as follows:

$$ q_1 = \Pr(WT_{i+1} = k) \cdot P_1^{i+2} $$

$$ = \left( \frac{K-k}{K} \right) \cdot \left( \frac{2}{S+1} \right)^2 \cdot \left( \frac{1}{1+\rho} \right)^k, \quad (6) $$

where $P_1^{i+2}$ is the probability of $v_{i+2}$ in the active state.

Case 2. $v_{i+1}$ and $v_{i+2}$ are both sleeping and $v_{i+2}$ wakes up earlier. Under the cooperative mode, $v_i$ chooses to transmit the packet to $v_{i+2}$ after waiting only $m$ time slots. Figure 4 presents the sleep schedule of Case 2, and the probability of this case can be expressed as follows:

$$ q_2 = \Pr(WT_{i+1} = k) \cdot \Pr(WT_{i+2} = m \leq k) $$

$$ = \Pr(WT_{i+1} = k) \cdot \sum_{m=1}^{k} \Pr(WT_{i+2} = m), \quad (7) $$

where $\Pr(WT_{i+2} = m \leq k)$ denotes the probability of node $v_i$ waiting $m$ slots for $v_{i+2}$ to wake up and $m < k$. Then, the above equation can be transformed to the following:

$$ q_2 = \left( \frac{K-k}{K} \right) \cdot \left( \frac{2}{S+1} \right) \cdot \frac{1}{1+\rho} \cdot \sum_{m=1}^{k} \left( \frac{K-m}{K} \right) \cdot \left( \frac{2}{S+1} \right) \cdot \frac{1}{1+\rho}. \quad (8) $$

Case 3. $v_{i+1}$ and $v_{i+2}$ are both sleeping and $v_{i+2}$ wakes up earlier, which means that waiting time $m$ is larger than $k$. 

![Figure 2: An example of cooperative transmission.](image)

![Figure 3: A schedule of nodes in Case 1.](image)
Node $v_j$ has no choice but to wait $k$ time slots before sending packet to $v_{i+1}$. The sleep schedule of Case 3 is given in Figure 5, and the probability is as follows:

$$d_j^i = \Pr(WT_{i+1}^j = k) \cdot \Pr(WT_{i+2}^j = m > k)$$

$$= \Pr(WT_{i+1}^j = k) \cdot \sum_{m=k+1}^{K} \Pr(WT_{i+2}^j = m)$$

$$= \left[ \frac{K-k}{K} \right] \times \left[ \frac{2}{S+1} \right] \times \left( 1 + \frac{\rho}{1+\rho} \right) \cdot \sum_{m=k+1}^{K} \left[ \frac{K-m}{K} \right] \times \left[ \frac{2}{S+1} \right] \times \left( 1 + \frac{\rho}{1+\rho} \right).$$

(9)

In conclusion, under the CT mode, the average sleep latency of forwarders sending packets to the downstream node is as follows:

$$SL = \frac{K}{k=1} q_j^i + \frac{K}{k=1} m \times d_j^i + \frac{K}{k=1} k \times d_j^i.$$  

(10)

3.2.2. Energy Consumption Model. In the duty-cycled WSN, node energy consumption includes three modules: sensor module with low power, microprocessor module, and wireless transceiver module; in which the energy consumption of sensor module and microprocessor module is small and identical in DT and CT modes. We focus on the energy cost of wireless transceiver module in this section, which includes the energy consumption of packet transmission, reception, and idle listening. Note that the node in this paper can predict the status of the next hop, next two hops and the surrounding nodes through pseudorandom number generation (PRNG) algorithm; the energy consumption of idle interception under DT and CT is equal. We focus on the energy consumption of packet transmission and reception under the two modes. Figure 6 is a simple transmission model of DT and CT, where A, B, C, and D are nodes in the WSN, and the routes under DT and CT are A-B-C and A-D-C, respectively. Define the energy consumption under the two modes as follows:

$$E_{DT} = P_A^{DT} + P_B^{DT} + R_B^{DT} + R_C^{DT},$$

$$E_{CT} = P_A^{CT} + P_D^{CT} + R_D^{CT} + R_C^{CT},$$

where $P_A^{DT}$, $P_B^{DT}$, $P_A^{CT}$, and $P_D^{CT}$ are energy consumption of nodes A and B sending packet directly to B and C, with A sending packet to cooperation node D, $R_B^{DT}$, $R_C^{DT}$, $R_D^{CT}$, and $R_C^{CT}$ are energy consumption of nodes B, C, and D receiving packet. Moreover, assume that the distribution density of nodes in the network is constant; thus the energy consumption of each node receiving packet is identical, which is $R_B^{DT} + R_C^{DT} = R_D^{CT} + R_C^{CT}$. Hence, the difference of transmitting energy consumption under DT and CT is equal to the difference of total energy consumption under the two modes.

Transmitting Energy Consumption under DT. Notice that signal can be decoded at the target nodes, only if the SNR of the packet received at the target node is larger than the minimum value (SNR$_{min}$) requested by the receiver. In other words, $P_A^{DT}$ should meet the following condition:

$$\text{SNR} = \frac{P_A^{DT} E\left[\alpha^2\right] d_{AB}^{-1}}{P_\eta} \geq \text{SNR}_{min},$$

(12)

where $\alpha$ is the random attenuation coefficient of the channel, $P_\eta$ is the power of additive Gaussian white noise, and $d_{AB}$ is the distances between A and B. We can get $P_A^{DT}$ in the same way; hence, the minimum sending energy consumption under DT is obtained as follows:

$$P_A^{DT} + P_B^{DT} = P_\eta \text{SNR}_{min} d_{AB}^{-1} + P_\eta \text{SNR}_{min} d_{BC}^{-1}.$$  

(13)

Transmitting Energy Consumption under CT. Notice that CT process includes two phases: (i) transmitting node A shares the packet with cooperation node D and (ii) A and D work together and send packet to the target node C through cooperative transmission.
For the first phase, the energy consumption is similar to that under DT mode; according to (13), \( P^C_T \) can be expressed as follows:

\[
P^C_T = \frac{P_y \text{SNR}_{\text{min}}}{E[\alpha^2]} d^4_{\text{AD}}, \quad (14)
\]

For the second phase, source node A works with the cooperation node D, sending packet to the target node C through cooperative transmission. The transmitting energy consumed in this process is indicated as \( P^C_{AD} \). Based on our previous work [15], the minimum energy consumption of the second phase under CT mode can be expressed as follows:

\[
P^C_{AD} = \frac{P_y \text{SNR}_{\text{min}}}{E[\alpha^2]} \left( d^3_{AC} + d^3_{DC} \right).
\]

We can see that the energy consumption of the second phase of CT highly depends on \( d_{DC} \), which is the distance between cooperation node and target node. The shorter the distance is, the smaller the energy consumption in second phase of CT is.

In conclusion, under CT mode, the minimum transmitting energy consumption is as follows:

\[
P^C_T + P^C_{AD} = \frac{P_y \text{SNR}_{\text{min}}}{E[\alpha^2]} d^4_{\text{AD}} + \frac{P_y \text{SNR}_{\text{min}}}{E[\alpha^2]} \left( d^3_{AC} + d^3_{DC} \right).\quad (16)
\]

3.2.3. DTCT Algorithm. Based on the model in Section 3.2.1, we propose an algorithm which incorporates CT strategy into the random duty-cycled WSN to reduce sleep latency. The algorithm is named (delay-tolerant cooperative transmission DTCT). Here, we give the network model and the assumptions about the algorithm first and then introduce the DTCT algorithm in detail.

Note that WSN can be denoted by a graph \( G(V, E) \), \( V \) is the nodes set and \( E \) is the set of edges between nodes. If there exists a direct link \( e(i,j) \) between the two nodes \( i \) and \( j \), then \( e(i,j) \in E \). In a duty-cycled WSN, nodes switch between sleep and active states periodically. We use \( V(n) = \{ i | i \in V \land S_{n-1} = 1 \} \) and \( E(n) = \{ e(i,j) | e(i,j) \in E \land (S_{n-1} \cap S_{n} = 1) \} \) to represent the awake nodes and available links at the \( k \)th time slot. Thus, the duty-cycled WSN is written as \( G(n) = (V(n), E(n)) \). PRNG [13] is applied to predict the next active time. The basic idea of PRNG is as follows. Assume that each node \( i \) in the network holds a local PRNG for deciding the time slots, which is initialized with a local seed. Each node also maintains a copy of the PRNG used by its neighbors, initialized with the seed they are using. Each time a node \( i \) wakes up, the node sends a short synchronization packet containing its local seed \( S_i \) and a newly generated random number \( S_i \). For \( S_i \), first, it is used for calculating the (relative) time of the next active slot of node \( i \), which is derived by \( 1 + S_i \mod K \). Second, an active neighbor \( j \) of node \( i \) uses \( S_i \) to synchronize its copy of node \( i \)'s PRNG. Notice that the activation of node in network is random; each pair of nodes can be eventually active at the same time. Thus, nodes in the network are able to know each other's seed.

The basic idea of DTCT is that a forwarder chooses the transmission strategy, either DT or CT, according to the status of the next two nodes. When the source node needs to transmit message to sink, it constructs a route \( R_k = \{ v_1, v_2, \ldots, v_j, v_{i+1}, \ldots, \text{sink} \} \), and DTCT is applied to the nodes responsible for delivering packets to sink on the route. If a sender \( v_i \) receives a packet to transmit and the next node \( v_{i+1} \) is awake, the packet will be sent to \( v_{i+1} \) directly in the DT manner. Whereas, if \( v_{i+1} \) is sleeping, \( v_i \) does not need to wait for it to wake up but sends the packet to the next awake node \( v_{i+2} \) based on the range extension of CT. This will definitely cut down the waiting time.

After deciding the transmission strategy, the sender needs to select relays from the neighboring nodes to cooperate with. The sender in Cases 1 and 2 is named DN (dominate node). According to [14], the transmission range is extended to 2.71 times when two relays participate in. This means that DN needs to choose relays in its neighborhood for cooperation, hops over the next node, and transmits message to the node after the next. Two factors should be taken into account when selecting the best relays: (i) node state and energy cost of CT, and only awake nodes are qualified to be the potential relays, and (ii) the lowest energy consumption node is selected to launch CT. As mentioned before, PRNG algorithm is conducted to predict the state of neighbor nodes; the active ones are considered as potential relays and defined as set \( R_{\text{potential}} \). To achieve the lowest energy consumption when launching CT, based on the energy consumption model of CT before, we need to find a node \( D \in R_{\text{potential}} \) which satisfies the following equation:

\[
\text{Min} \left\{ \frac{P_y \text{SNR}_{\text{min}}}{E[\alpha^2]} d^3_{\text{AD}} + \frac{P_y \text{SNR}_{\text{min}}}{E[\alpha^2]} \left( d^3_{AC} + d^3_{DC} \right) \right\}. \quad (17)
\]

The variables are consistent with the ones in (16). The pseudocode of the DTCT algorithm is presented in Algorithm 1.

3.3. Optimized Fixed Duty-Cycle Scenario. To deal with the delivery delay in nonrandom duty-cycled WSN, in this section, we introduce CT to a specific optimized fixed duty-cycled WSN: concentric ring for the grid topology, which is proposed by Lu et al. in [16]. We first review the concentric ring schedule and then conduct CT to it to reduce the average delivery delay.

In [16], the author aims to minimize the transmission latency at the given duty cycling requirement. Assume that, in a low traffic duty-cycled WSN, the node should be kept active on an average of \( 1/k \) of the time slots, where \( k \) is the schedule length and each node is assigned to be active at one of the \( k \) slots, while it can transmit any packet at any slot. Each node in the network should only receive the packet at its active time slot. In addition, every pair of the nodes is equally likely to communicate. The network wide delay is characterized by the delay diameter which is defined as the maximum value of the shortest delay between each pair of nodes in the grid. The optimal assignment on a ring can seem as a set of concentric rings with interconnecting bridges, and the outer most ring is given a sequential assignment going in the clockwise direction starting at 0. For the other rings, a slot assignment is chosen that provides the best delay diameter for
(1) Check the sleep schedule of next hop \(v_{i+1}\) when \(v_i\) receiving message;
(2) if \(S_{i+1} = 1\) then
(3) do DT; \{DT stands for direct transmission.\}
(4) else
(5) check the sleep schedule of node \(v_{i+2}\);
(6) if \(S_{i+2} = 1\) then
(7) do CT; \{CT stands for cooperative transmission.\}
(8) else
(9) Compare the waiting-time of \(v_{i+1}\) and \(v_{i+2}\);
(10) if \(WT_{i+1} \geq WT_{i+2}\) then
(11) wait for \(WT_{i+2}\) slots and do CT;
(12) else
(13) wait for \(WT_{i+1}\) slots and do DT;
(14) end if
(15) end if
(16) end if
(17) Function DT
(18) Direct Transmission: node \(v_i\) transmits packets to \(v_{i+1}\) directly
(19) End function DT
(20) Function CT
(21) if \(|RS_{\text{potential}}| = 0\) then
(22) do DT;
(23) else
(24) pick up relay \(R_i\) from \(RS_{\text{potential}}\) satisfying (17);
(25) end if
(26) End Function CT

Algorithm 1: Delay-tolerant cooperative transmission algorithm at node \(v_i\).

Figure 7: An example of concentric ring.

that ring. The optimal assignment is shown in Figure 7, where \(k = 5\) and the grid size is \(4 \times 4\).

To minimize the delivery delay, we introduce CT to this optimized fixed schedule. Figure 8 is an example of the optimal concentric ring with CT, where the red and green lines are DT path and CT path, respectively. Assume that node A has a packet to send to node C at slot 0. Under DT mode, the shortest delay path is A-B-C. A can only transmit to B at slot 3; the delay from A to B is 3 slots; then, B needs to wait for 4 slots until C wakes up; the delay from B to C is 4 slots. Thus, the total delay under DT mode is 7 slots. By introducing CT to the grid, node in the network can select the neighbors to cooperate with. A could send the packet to D first; the delay between them is 1 slot, and then they transmit the packet to C together by CT; they need to wait 1 slot until C wakes up; thus, the total transmission latency is 2 slots. Hence, the shortest delay path is A-D-C. Similarly, when node E has packet to send to node H, the shortest transmission delay under DT mode and CT mode is 8 slots and 3 slots, respectively. Apparently, the delivery delay is reduced by introducing CT. The energy consumption of this schedule also follows the model proposed in Section 3.2.2.

4. Performance Evaluation

4.1. Simulation Setup. The delay models described in Sections 2 and 3 are simulated on Matlab platform. The shortest route between source and destination is identified by DSR algorithm and the packet transmission is bidirectional. We define the path loss exponent \(\lambda = 2\) and the additive white Gaussian noise variance \(N_0 = -70\) dbm. For cooperative communication, the signals are modulated by binary phase shift keying (BPSK) at BER of \(10^{-3}\). Theoretical analysis of the delay model under the CT mode is performed, and then the DTCT algorithm is evaluated by comparing the performance of the DT and CT strategies.

4.2. Evaluation Results of Random Duty-Cycle Scenario. In this section, the performance of CT mode in random duty-cycled WSN scenario is evaluated. Firstly, the numerical results of CT model under different parameters (maximum sleep time slots \(K\) and maximum active time slots \(S\)) are simulated, and then the performances of the two delay models in random schedule are compared in the aspects of transmission and transmission energy consumption. Finally, transmission delay under the circumstances of different numbers of neighbor nodes in the CT model is analyzed.

4.2.1. Numerical Results. We define a metric: sleep latency gain (SLG), which is the ratio between sleep latency under the store-and-forward and CT modes.

Figure 9 illustrates the SLG in Cases 1 and 2. Apparently, SLG is constantly greater than 1, indicating that CT strategy can save the time of waiting for the next sleeping node to
wake up. The maximum SLG is 12.07 at $K = 16$ and $S = 60$. For a given value of $k$, SLG is positively correlated to $S$. The reason is that the probability of the node selecting CT strategy increases with the increase of $S$. The more frequently the senders encounter Cases 1 and 2, the greater the sleep latency can be reduced by using CT. Nevertheless, for a given $S$ and a growing $K$, SLG increases at first and then decreases slightly, which is because the probability of the sender meeting Cases 1 and 2 decreases with the lower duty cycles. Consequently, the sender has less chances to transmit messages cooperatively.

Figure 10 depicts the SLG with different $K$ and $S$ in all the three cases. We can see that the minimum SLG is 1.078 when $S = 10$ and $K = 60$, which validates that DTCT reduces sleep latency in all the cases. Compared with that in Figure 9, the tendency of SLG decreasing after an initial increase is more obvious for a given $S$ and a growing $K$. The major reason is that when $K$ reaches a certain point, the probability of the sender meeting Case 3 is very large. In summary, CT strategy minimizes sleep latency evidently.

4.2.2. The End-to-End Delay versus Node Duty Cycle.

Figure 11 shows how duty cycle affects the end-to-end delay in either DTCT or store-and-forward transmission. The average end-to-end delay decreases when the duty cycle increases in both modes, and the delays of the two modes are very close to each other when the duty cycle is greater than 50%. This is because the probability of the next node to be sleeping gets smaller with the increase of duty cycle, and thus the forwarder is more likely to transmit a packet directly to the next node. Similarly, the average delay increases obviously when the duty cycle decreases. The average delay in DTCT is 3.5 times that in the store-and-forward transmission when the duty cycle is 5%. Obviously, DTCT is more adaptable in low duty-cycled WSN, in which nodes sleep longer and conserve more energy.

4.2.3. Transmission Energy Consumptions versus Node Duty Cycle.

Figure 12 presents the average transmission energy consumption as the duty cycle increases. Clearly, DTCT could reduce the energy consumption by 11.29% on average. The reduction of energy consumption is mainly because, in the process of cooperation node selection, DTCT always selects the lowest energy consumption node as the cooperator.

4.2.4. The End-to-End Delay versus the Number of Neighbors.

Figure 13 illustrates the correlation between the average end-to-end delay and the number of neighbor nodes of the forwarder. As we can see, the number of neighbor nodes affects DTCT performance more evidently when the duty cycle is 10% (low-duty-cycled WSN) than when the duty cycle is 30% or 50%. This is consistent with the result shown in Figure 11 that DTCT plays a more important role in
minimizing the end-to-end transmission delay in low-duty-cycled WSN. When ND has only a small number of neighbor nodes, the potential relays will be less. More neighbor nodes lead to a higher probability of conducting CT and reducing sleep latency. Additionally, we find that the number of neighbor nodes, when it reaches up to 4, has little influence on transmission delay in high duty-cycled WSN. Whereas, this number is 6 in low duty-cycled WSN. The results suggest that DTCT performs better in low-duty-cycled WSN.

4.3. Evaluation Results of Optimized Fixed Duty-Cycle Scenario. The performances of the two transmission modes DT and CT in the optimized fixed duty-cycled WSN are compared in the aspects of end-to-end transmission delay and transmitting energy consumption. The topology of concentric ring is shown in Figure 7, where the length of the time slots is five and the size of the grid is $4 \times 4$. Consider that the probability of each node sending a packet to the neighbors at each time slot is equal. Assume that distance between the adjacent nodes is 1. Notice that the shortest delay path for each pair of nodes in the grid is asymmetric; thus, the total number of the results is 256.

Figure 14 illustrates the end-to-end delay under DT and CT modes, which are presented by the red and blue curves, respectively. The average delay under DT and CT modes is 6.02 slots and 5.16 slots, respectively. The transmission delay is reduced by 14.3% after introducing CT. Besides, the delay diameter also decreases from 12 to 11 under CT mode. Figure 15 presents the transmitting energy consumption under DT and CT modes, the average values of which are $7.72 \times 10^{-7}$ and $8.20 \times 10^{-7}$. Notice that the energy consumption is increased by almost six percent when introducing CT to the grid, which is tolerable. The results suggest that CT could also reduce the delivery delay in optimized fixed duty-cycled WSN while the energy consumption is not increasing excessively.

5. Related Work

Currently, many researchers have addressed the issue of transmission delay in duty-cycled WSN from the following three perspectives: (1) finding the lowest latency path by...
routing algorithms for a given duty cycle, (2) scheduling the node duty cycle for a given path, and (3) finding the route and scheduling the duty cycle jointly.

For the first perspective, in [6], Ratnaraj et al. form sub-networks and regard the end-to-end delay as route metric to select the forwarding node so as to find the path with the lowest delay. The lukewarm forwarding routing algorithm is proposed based on spanning tree in [13]. In [17], the routing algorithm is organized in layered structure and packets are delivered to the awake-earliest node according to the layer order, so there is no fixed path. In addition, geographic information routing is exploited to find the lowest delay path more easily in [8, 9]. The research [8] puts forward a new metric as the standard of selecting the next node to achieve minimal transmission delay, and a new metric routing with the sleep-wake scheduling of each node is presented in [9]. Reference [18] proposes a (Coded Anycast Packet Forwarding CAPF) scheme which could not only reduce the end-to-end delay but also improve the reliability in WMNs. The research [19] considers the issue of minimizing the end-to-end delay bounds in WSN. The authors show that, by using a centralized algorithm to decide the link schedule and (Coordinated Earliest Deadline First CEDF), a delay bound is derived for each flow in the network.

For the second perspective, Lu et al. [16] try to arrange the wake-up time of nodes and introduce different heuristic algorithms to reduce transmission delay. The LETED approach is proposed to solve the clock drift problem of nodes in [20], in which the nodes on the path can arrange their own transceiver time slots according to the schedules of the last hop and the next hop. The research [10] constructs broadcast tree from sink to each node and schedules the state of nodes to reduce the transmission delay. The research [21] constructs connected dominating set in the network and forms a backbone composed of core nodes. The core nodes are scheduled to be in the active state while the noncore nodes in the sleep state. Data transmits through core nodes, which further forward the received information to the surrounding noncore nodes. In this way, transmission delay can be avoided.

For the third perspective, [22] presents (minimum latency joint scheduling and routing MLSR) algorithm to find nonintersecting paths to realize the minimization of delay. The research [7] solves the routing and scheduling problems by using convex optimization theory and finally realizes the maximization of network lifetime.

Nevertheless, the above three approaches transmit packets in the traditional store-and-forward manner, which leads to end-to-end delay in the path. In this paper, we adopt a cooperative way to transmit message. To the best of our knowledge, our study is the first to exploit the CT strategy in reducing transmission delay in duty-cycled WSN.

6. Conclusion

This paper addresses the issue of transmission delay in duty-cycled WSN. Cooperative transmission is employed to reduce sleep latency. The transmission delay models under both store-and-forward and cooperative modes in random duty-cycled WSN are established. In this scenario, each node in the network needs to know the seeds used by the neighbors when predicting the status of the neighbors by PRNG algorithm. DTCT algorithm is proposed for forwarders to make choice of transmission modes between direct store-and-forward and cooperative communication for the purpose of minimizing sleep latency. The cooperative transmission is also introduced to a specific fixed duty-cycled WSN. Simulations results validate that cooperative transmission outperforms the traditional store-and-forward mode in end-to-end transmission delay. In particular, CT reduces 67% and 14.3% of the transmission delay in random and optimized fixed low duty-cycled WSN, and DTCT algorithm saves energy by 11.29% in random duty-cycled WSN. In conclusion, our approach contributes to transmission delay reduction in both random and fixed low duty-cycled WSN.

Acknowledgment

This work is supported in part by the Fundamental Research Funds for the Central Universities under Grant xjj2011078.

References

[1] F.-C. Jiang, D.-C. Huang, C.-T. Yang, C.-H. Lin, and K.-H. Wang, “Design strategy for optimizing power consumption of sensor node with min(n,t) policy m/g/1 queuing models,” International Journal of Communication Systems, vol. 25, no. 5, pp. 652–671, 2012.

[2] M. Sanaullah Chowdhury, N. Ullah, M. A. Ameen, and K. S. Kwak, “Framed slotted aloha based mac protocol for low energy critical infrastructure monitoring networks,” International Journal of Communication Systems, 2012.
[3] N. Ullah, M. S. Chowdhury, P. Khan, and K. S. Kwak, “Multi-hop medium access control protocol for low energy critical infrastructure monitoring networks using wake-up radio,” *International Journal of Communication Systems*, 2012.

[4] J. Polastre, J. Hill, and D. Culler, “Versatile low power media access for wireless sensor networks,” in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (ser. SenSys ’04)*, pp. 95–107, ACM, New York, NY, USA, November 2004.

[5] Y. Sun, O. Gurewitz, and D. B. Johnson, “Ri-mac: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks,” in *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (ser. SenSys ’08)*, pp. 1–14, ACM, New York, NY, USA, 2008.

[6] S. Ratnaraj, S. Jagannathan, and V. Rao, “OEDSR: optimized energy-delay sub-network routing in wireless sensor network,” in *Proceedings of the 2006 IEEE International Conference on on Networking, Sensing and Control (ICNSC ’06)*, pp. 330–335, April 2006.

[7] F. Liu, C. Tsui, and Y. J. Zhang, “Joint routing and sleep scheduling for lifetime maximization of wireless sensor networks,” *IEEE Transactions on Wireless Communications*, vol. 9, no. 7, pp. 2258–2267, 2010.

[8] K. Wang, L. Wang, C. Ma, L. Shu, and J. Rodrigues, “Geographic routing in random duty-cycled wireless multimedia sensor networks,” in *Proceedings of the IEEE Globecom Workshops (GC ’10)*, pp. 230–234, December 2010.

[9] Z. Jiang, J. Wu, and R. Ito, “A metric for routing in delay-sensitive wireless sensor networks,” in *Proceedings of the IEEE 7th International Conference on Mobile Adhoc and Sensor Systems (MASS ’10)*, pp. 272–281, November 2010.

[10] A. Swain, R. Hansdah, and V. Choubian, “An energy aware routing protocol with sleep scheduling for wireless sensor networks,” in *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications (AINA ’10)*, pp. 933–940, April 2010.

[11] W. Guo, J. Liu, Y. Liu, L. Zheng, and G. Zhang, “Exact capacity analysis of multi-relay multuser cooperative networks based on twostep selection,” *International Journal of Communication Systems*, vol. 26, no. 5, pp. 662–673, 2013.

[12] T. A. Tsiatis, “Performance of wireless multihop communications systems with cooperative diversity over fading channels,” *International Journal of Communication Systems*, vol. 21, no. 5, pp. 559–565, 2008.

[13] R. Beraldi, R. Baldoni, and R. Prakash, “A biased random walk routing protocol for wireless sensor networks: the lukewarm potato protocol,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 11, pp. 1649–1661, 2010.

[14] J. W. Jung and M. A. Ingram, “Residual-energy-activated cooperative transmission (REACT) to avoid the energy hole,” in *Proceedings of the IEEE International Conference on Communications Workshops (ICC ’10)*, pp. 1–5, May 2010.

[15] Y. Wang, X. Yang, S. Yang et al., “Towards energy-efficient cooperative routing algorithms in wireless networks,” in *Proceedings of the IEEE Consumer Communications and Networking Conference*, pp. 79–84, January 2013.

[16] G. Lu, N. Sadagopan, B. Krishnamachari, and A. Goel, “Delay efficient sleep scheduling in wireless sensor networks,” in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’05)*, vol. 4, pp. 2470–2481, March 2005.