

Research Article

Comprehensive Optimization of Energy Consumption and Delay Performance for Green Communication in Internet of Things

Jiaze Wang,¹ Chunhua Hu,^{2,3} and Anfeng Liu¹

¹*School of Information Science and Engineering, Central South University, Changsha 410083, China*

²*Key Laboratory of Hunan Province for Mobile Business Intelligence, Hunan University of Commerce, Changsha 410205, China*

³*Mobile E-Business Collaborative Innovation Center of Hunan Province, Hunan University of Commerce, Changsha 410205, China*

Correspondence should be addressed to Chunhua Hu; huchunhua777@163.com and Anfeng Liu; afengliu@mail.csu.edu.cn

Received 16 December 2016; Accepted 22 February 2017; Published 23 March 2017

Academic Editor: Ning Zhang

Copyright © 2017 Jiaze Wang 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.

Energy efficiency as well as fast data transmission is vital to green communications-based applications for Internet of Things (IoT). Wireless sensors, which constitute one of the important parts of IoT, adopt duty cycle operating mode to save energy. Although duty cycle operating mode will decrease the energy consumption of sensor nodes, it leads to a larger communication delay. In this paper, a utility-based adaptive duty cycle (UADC) routing algorithm is proposed to increase energy efficiency, reduce transmission delay, and keep long lifetime at the same time. First, UADC routing algorithm adopts a comprehensive performance evaluation function to evaluate the utility of choosing different relay nodes. Then it selects the node which maximizes the utility of the system to perform data relay. The utility function synthesizes comprehensive indexes like the reliability, energy consumption, and delay of the node. UADC routing algorithm adopts a high-duty cycle operating mode in the areas which have more remaining energy to decrease the delay. And a low-duty cycle operating mode in the energy-strained areas is adopted to ensure a long lifetime. The simulation results also prove the significant performances of our proposed algorithms.

1. Introduction

Internet of Things (IoT) connects physical objects such as thermostats, refrigerators, and vehicles to the Internet over the wireless medium and enables them to share information and coordinate operations [1–4]. With applications in many domains that include transportation, healthcare, manufacturing, home automation, and the power grid, IoT plays a remarkable role in improving the quality of life and growing the world's economy [5–8]. Wireless sensors are one of the important parts of IoT [9–11]. According to deploy wireless sensors in the areas to be detected, by means of self-organizing network, these wireless sensors send the sensed data back to the information center to be stored and analyzed in real time [12, 13], so that human beings can get ubiquitous perception. However, since wireless sensors generally use the battery for power supply, their energy is limited [14, 15]. Researchers have proposed various measures to prolong the lifetime of wireless sensors. Among them, one of the most effective ways is to make nodes using the periodic work and

sleep operating mode [16, 17]. The sensor nodes can turn off the wireless antenna and data transmission device when in the sleep state. Thus the energy consumption of the node is 1/1000 of the working state, greatly saving the energy.

The events or objects the sensor nodes perceived last for seconds, minutes, or even hours; on the other hand, the sensor nodes periodically rotate their work patterns in milliseconds, so that they do not affect the observation of surrounding physical phenomena while saving a lot of energy. However, the use of duty cycle work pattern enormously impacts the communication delay. The underlying reason is that when the node is in sleep mode, no data are received or sent. Accordingly, when there are data to send, communication will only be achieved when the node is awake, thus increasing the delay. Additionally, if the duty cycle of a node is decreased, though it costs less energy when the duty cycle increases, the delay increases as well.

On the other hand, the quality of wireless communication link is not as good as that of the wired one, where its reliability is relatively low. The research shows that the reliability of

some wireless link is less than 90% or even lower [16–19]. When selecting the relay node, the node should go with the one of high reliability which requires less retransmission times to the energy consumption and acquire a minor delay [16, 17]. Thus, selecting relay with high-quality links has important significance [9].

The selection of the relay node is also associated with the energy of the node. Obviously, the node with higher residual energy should be selected for data forwarding, since selecting the nodes with less residual energy leads to an unbalanced energy consumption between nodes, resulting in the early death of some nodes. In most studies, network lifetime is defined as the first dead time of the node [20, 21]. Therefore, nodes with more residual energy should be chosen to balance energy consumption, which is beneficial to prolonging the service life.

In summary, when selecting the next-hop node in routing, we should consider various factors comprehensively. The problem is a multiobjective optimization problem [16, 17]. There have been some related studies, where utility function is a widely used research method [11, 19]. Its principles are as follows: one or more performance indicators are transformed into a target value by the utility function, and the node with the highest target values will be selected as the relay node when routing. Xiao et al. [11] put forward the concept of utility function. In their studies, utility function is described as the synthesis of link reliability and cost. Some other researches on the comprehensive optimization also used utility function [19]. But there are some areas which have not been discussed in previous studies: (1) many studies only combine one or two performance indicators, so the results of their integrated optimization are limited. Obviously, it will be of great significance to IoT communication if energy, delay, and reliability could be all optimized at the same time. However, few studies have been done to achieve that. (2) The previous studies often only optimized the utility function. Since each node only gets the information about itself and its data forwarding set nodes, even if each node gets the optimal utility, overall, it is not necessarily optimized. Therefore, unlike previous studies, UADC not only optimizes the comprehensive performance index but also optimizes the network delay from the global perspective. This algorithm is comprehensively optimized after overall optimization, so that the algorithm gets a better performance than the aforementioned optimization algorithm. In conclusion, the main contributions of this work are listed as follows.

(1) Increasing the duty cycle of the node can effectively decrease the delay. Therefore, if the duty cycle can be effectively increased without affecting other performance, the delay can be lowered. With the energy balance seen as a whole, the node adaptive duty cycle adjustment strategy is proposed to balance the energy consumption of nodes in distinct areas of the network. Thus, the energy consumption of nodes can be extended as much as possible without exceeding that of the largest energy consumption nodes in the network, while the duty cycle of the nodes with remaining energy is greatly ensured. It has wide applicability in that it effectively reduces delay and can be applied to many additional strategies.

(2) Because the duty cycle of most of the nodes can be improved effectively in the network, a novel utility-based adaptive duty cycle (UADC) routing algorithm is proposed to improve the energy efficiency while reducing the communication delay and keep a long lifetime. The energy consumption and reliability of the link are analyzed and considered in this paper through the utility function. Theoretically, the maximum number of retransmission times of energy consumption per unit transmission bit is obtained so that the data transmission energy efficiency is maximized. In the design of the benefit function, the forwarding nodes with small unit transmission bit energy consumption, high remaining energy, and minor delay result in a superior benefit value. Therefore, those with high energy consumption, low residual energy, and huge delay should be avoided. Consequently, the energy efficiency will be brought up, with the energy consumption being balanced and the data transmission delay being small. Meanwhile, the UADC routing algorithm can obtain optimization effect from the whole to partial synthesis by choosing the link that maximizes the system's profit function and the adaptive duty cycle.

(3) Through our extensive simulation study, we demonstrate that both delay and energy efficiency can be enhanced simultaneously using the proposed UADC routing algorithm. Comparing with the previous optimization approaches, UADC routing algorithm optimizes the network energy from the global and local levels. At the global level, UADC can take full advantage of the remaining energy of nodes to increase duty cycle to reduce delay. At the local level, by optimizing the function, UADC chooses the node with higher energy efficiency and more residual energy to make the network energy consumption less and more balanced. Theoretical and experimental results show that UADC routing algorithm can improve energy utilization rate in more than 32.3% with the delay reduces by 77%.

The rest of this paper is structured as follows. In Section 2, the related works are reviewed. The system model is presented in Section 3. A novel UADC routing algorithm is proposed in Section 4. Section 5 shows the experimental results and comparison. Finally, we conclude this paper in Section 6.

2. Related Work

Internet of Things (IoT) is composition by large number of sensor nodes, which plays a more and more important role in national production and life [22–24]. Although tremendous efforts have been made in advancing IoT technologies, green IoT still faces significant challenges from many aspects [24–29]. In general, the challenges mainly include the following aspects.

(1) *The Energy Consumption Efficiency.* The energy consumption efficiency is the most important challenge of sensor nodes all the time. Researchers have made a lot of efforts to minimize the energy consumption of nodes and carried out researches from the MAC layer to the network layer in many aspects. Firstly, optimizing the MAC layer can effectively improve the utilization of energy, because the transmission in wireless network involves many factors, such

as transmit power mode, transmission distance, transmission power, and packet size [11, 14, 16, 17, 20, 30, 31]. Therefore, the energy consumption of the nodes can be optimized if the transmission power of nodes is optimized. Similarly, the choice of the appropriate transmission range has also expressed the same meaning. The routing process for nodes to transmit data to the sink is faced with the problem of one-hop length selection. In such a problem, nodes can send packets through the expansion of the transmission power, and with fewer hops to reach sink, nodes can also adopt the opposite approach. This is an optimization problem. Chen et al. [19] have studied how to determine one-hop length and corresponding transmit power; in order to minimize the total energy consumption, they proposed a method of how to select the optimized one-hop length. However, it should be pointed out that this way is not necessarily able to improve network lifetime although it can minimize the total energy consumption. Because the network lifetime is often defined as the time when the first node dies, lifetime is determined by the energy consumption of the node with the highest energy consumption in the network. Therefore, the strategy of minimizing the total energy consumption of the network is often not necessarily to maximize the network lifetime [30].

Obviously, if the parameter optimizations of the MAC layer and the network layer are combined to conduct the so-called cross-layer optimization, the network performance can be improved better. Based on above thought, by optimizing the deployment of the number of nodes, nodes' deployment location, and nodes' transmission structure, Xiao et al. [11] and Zhang et al. [18] carried out cross-layer optimization from the MAC layer and the network layer at the same time, which resulted in a better network performance [31].

Actually, the transmission mode and the transmission power of nodes are also related to the data transmission rate. Thus it is related to the data transmission delay. Studies [20] have shown that when the node adopts a high data transmission rate, its energy consumption will increase very fast. But at this time, the rate of data transmission is improved, thus making the data transmission delay reduced. This kind of research can be found in our previous work [20].

There are many studies on the energy efficiency of the network layer [11, 31]. In the study of network layer, it is generally assumed that the parameters of the MAC layer are fixed, and the main researches are routing algorithm and strategy. In early research work, the research on routing layer mainly focuses on how to minimize the energy consumption of network routing. For example, Ettus proposed the so-called MTE (minimum transmission energy) routing scheme [32]; in such routing strategy, the principle of selecting relay nodes is to minimize the total energy consumption of transmitting a packet from the source to the sink. But later studies have found that the strategy of minimizing the total energy consumption of routing often makes nodes which are often selected in the routing continue to be selected in the later routing, so that the energy consumption between nodes is not balanced and some nodes die early, thus affecting network lifetime. The routing algorithm in the later not only considers the total energy consumption of the routing, but also considers choosing the node which has the

maximum residual energy as relay node, so that the network energy consumption is balanced, thus improving the network lifetime. In many studies, the cost function is adopted to measure the consumption of different relay nodes, and then the least consumed nodes are chosen as relay nodes. In fact, these cost functions are equivalent to utility functions. In our previous work [30], we summarized the features of cost function used in routing. Then, we point out the principle of designing cost function in theory and give the standard of evaluating the cost function, which has good guiding significance for routing based on cost function [30].

(2) *Research on Data Transmission Reliability.* In most studies, the node transmitting mode is (0, 1) mode. In this mode, if the receiving node is within the sending radius of the sending node, the data can be received absolutely [14, 20, 26]. However, the nodes outside the sending radius cannot receive the data completely. Obviously this is a simplification of the model. Thus there are many studies using the non-(0, 1) model. In the non-(0, 1) model, the probability of receiving the data is different with different distances from the sending node. The probability of receiving data in the nodes near the sending node is very high, while the probability of successful reception decreases with the increase of the distance. Therefore, the factors affecting the reliability of data transmission are also very much. First, the factor affecting the transmission reliability of the link is the transmit power of the node. In the same transmission distance, the greater the transmit power of the node is, the higher the reliability of data transmission is and the more the energy consumption is. Due to the increased reliability of data transmission, the energy consumption for successful transmission of unit bits also can be optimized in unreliable wireless links. The optimization results were given in our previous work [14, 20, 33].

On the other hand, the reliability of data transmission is also related to delay. Although wireless link is unreliable, many methods can be used to guarantee the reliability of routing in network layer. The main adopted methods are as follows. (a) Adopting retransmission mechanism [26]: in such studies, if a node fails when transmitting packets, it will improve its reliability by sending the packets again or repeatedly. Obviously, each transmission takes time. So if the retransmission time is more, then the data transmission delay will be larger. The retransmission mechanism is also divided into many kinds; some adopt ACK feedback mechanism, and others do not use ACK feedback mechanism and adopt automatically retransmitting repeatedly method. These studies can be found in [11]. Retransmission mechanism also involves energy consumption. If the node retransmits packets multiple times, then it will consume multiple energy. Thus, the optimal relation among reliability, delay, and energy consumption is very complex. (b) Adopting encoding mode: in such studies, the sending node encodes the data before transmitting, and the receiving node decodes it to get the original data. Because of the use of redundant coding, it is possible to receive the correct data even if the receiver misses some of the data [14]. (c) Cooperative communication methods [13]: cooperative communication method is developed from MIMO (Multiple-Input Multiple-Output) method. It is an effective method

to combat fading of wireless network, increase transmission rate, and improve the reliability of communication systems. Thus, the real time and reliability of data transmission are improved by cooperative communication method.

(3) *Delay Research on Data Transmission.* Through the previous analysis we know that data transmission of delay involves multiple aspects and they are summarized as follows. (a) Delay in MAC layer: if the node adopts a larger transmission power, then the one-hop distance will be larger. Thus the number of hops required to reach sink is reduced, thereby reducing delay. Even without the increase in one-hop distance of nodes, due to the improvement of the reliability of data transmission, thus reducing the number of retransmissions, the delay will be effectively reduced [16, 17]. In addition, according to [20], increasing the data transmission rate of the nodes can also effectively reduce the delay. (b) Delay in network layer: because the deployment of sensor nodes is dense, therefore, when transmitting packets, the node can choose a node from several forwarding nodes to relay. At this time the node may select the nearest node from the sink to reduce the number of hops, thus reducing the delay.

The above studies assumed that nodes are in work state all the time. But in order to save energy, in many applications nodes are working in a duty cycle state [11, 16, 34]. In such networks, the data transmission delay is greater than the delay in the previous research. The reason is that nodes are periodically in sleep and work states. Therefore, when nodes have data to transmit, its forwarding nodes may be all in sleep state, resulting in sending node which needs to wait for its forwarding nodes to be awake before data transmission. Because the node's duty cycle period is much larger than the time used for data transmission, data transmission delay in the network whose nodes are in duty cycle mode is much higher than the network whose nodes are always working. And in this way, when node has data to send, the node that wakes up first is not necessarily the node which is the nearest to the sink; it may be very far from the sending node; thus it is a dilemma that the sending nodes should continue to wait for the nodes closer to the sink to wake up or send the data to the awake node immediately. If the node sends the data immediately, the distance of each hop is less; thus the data need more hops to reach the sink. And the delay of each hop is larger, which results in a very large end-to-end delay. If the node chooses to wait for the nodes closer to the sink node to wake up, although the distance of each transmission is longer, the number of hops needed to reach the sink node is less. But maybe the sending node will find that the awake node is farther away from the sink node than before after waiting for some time. At this time, in the case of the increased delay, we did not get better results. What is more, if the sending node needs to wait for the node near to the sink for a very long time, then its delay increases much more. Altogether these lead to the problem to choose relay nodes more complex [16, 17].

For such a complex problem, using the utility function is becoming an inevitable choice. There are some other utility-based researches. The benefit function of early research is usually a single performance index. For example, Chen et al. [19] made utilization efficiency defined as network lifetime

per unit deployment cost. Recent studies have combined multiple indexes as a utility function. For example, there are researches integrated the residual energy and the total energy consumption as a cost function, while in [11], in addition to the reliability of the link, the other influencing factors are represented by cost and then integrated them into the utility function to optimize. But from a global perspective, optimization of utility function is a kind of optimization in local level, and it is a kind of optimization of each node in its own local field of vision. But from the whole point of view it is not necessarily the best way. Therefore, in this paper, utility-based adaptive duty cycle (UADC) routing algorithm carried out optimization on the basis of the global level. Then the local level is optimized by using the utility function of multiple factors. Thus, the effect of comprehensive optimization can be obtained from both local and global perspective.

3. System Model

3.1. Network Model

- (1) We consider a WSN which is made up of N sensor nodes with a sink node situated at the center. The initial duty cycle of each sensor node is ϕ_{ini} . As soon as the source node generates a packet, the packet will be transmitted to sink node by the UADC algorithm we proposed.
- (2) For the purpose of reducing the energy consumption. There are two possible working states of each sensor: the active state and the dormant state. When the sensor is in the active state, it will turn on the whole functional modules including sensing, listening, receiving, and transmitting. In the dormant state, the sensor cannot perform any function except for a wake-up timer.
- (3) We divide the time into the same time slots and the network is based on synchronization. It is proved that the time slots are large enough in previous studies [1, 34], so we can ignore the time synchronization error.
- (4) The wireless communication is unreliable, and the existence of collision is dealt with by the CSMA/CA mechanism. It is proved that the quality of the link varies very slowly as time goes by, so it is reasonable that the link reliability can be evaluated by the probability of the successful transmission derived from historical data.
- (5) In this paper, we adopt the retransmission-allowed setting when transmitting packets, which means that when a node transmits packets, the packets can be transmitted multiple times to improve the success rate and the utility.

3.2. *Utility Model.* When the node is transmitting packets, the node will assign an initial benefit to each packet. Once a packet is generated from the source node, its benefit will decrease linearly as the time goes by until it arrives at the sink node. The packet will be discarded if its benefit reaches zero.

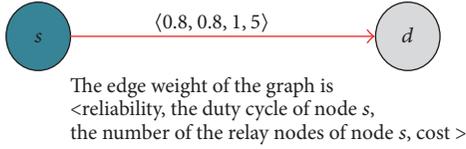


FIGURE 1: An example of UADC routing.

Regardless of whether the packet delivery is successful or not, the delivery in each hop will cost energy and cause a cost. The utility can be calculated as the remaining benefit subtracts the whole cost. We define the remaining benefit in Definition 1 and define utility in Definition 2.

Definition 1. Suppose $r(t)$ is the remaining benefit of a packet, which is a benefit function decreasing linearly as time goes by if the packet is transmitted to the sink node successfully; otherwise, the remaining benefit is zero. We denote the initial benefit as α and denote the benefit value decreased in every time slot as λ . Thus the remaining benefit can be calculated as

$$r(t) = \begin{cases} \alpha - \lambda \cdot t & \text{successful delivery;} \\ 0 & \text{failed delivery,} \end{cases} \quad (1)$$

where t is the lifetime of the packet. The benefit of packet is maximum when a packet is newly generated ($t = 0$). The maximum live time of the packet is α/λ ; if the whole delivery time is beyond α/λ , the packet will be discarded; thus $r(t)$ will be zero.

Definition 2. The total transmission cost is denoted as c ; suppose u is the utility of a packet delivery, which equals the remaining benefit subtracting the total transmission cost of the packet delivery. The utility can be calculated as

$$u = r(t) - c. \quad (2)$$

In the definition, the utility is calculated using the dimensionless calculation mode. Suppose a packet delivery from the source node ξ_i to the sink node ξ_{i+1} . If the packet arrives at the sink node successfully with the delay t_s , the utility would be $r(t_s) - c$; the utility would be $(0 - c)$ if it fails. The utility value is determined by the path reliability, the benefit, the transmission cost, and the delivery delay.

As is shown in Figure 1, we consider a packet sent from node ξ_s to node ξ_d . The initial benefit is 50, and the benefit value decreased in each time slot is 1. The transmission delay can be calculated as $d_s = f(0.8, 1)$. The remaining benefit $r_s(t)$ will be $0.8 * (50 - d_s * 1) = 40 - 0.8d_s$. Therefore, the utility can be calculated as $u_s = 40 - 0.8d_s - 5 = 35 - 0.8d_s$.

3.3. Energy Consumption Model. Since sensor node's energy resource is constrained, and it is extremely difficult to replace or recharge the batteries of sensor node, energy efficiency is another vital issue. For the purpose of saving energy and the sensor nodes not working all the time, it would be better if the nodes are set to sleep-wake cycle to work periodically; that is,

TABLE I: Network parameters.

Symbol	Description	Value
t_{duty}	Data collection period	100 ms
γ	Time slot duration	2 ms
w_t	Transmission power consumption	0.0511 W
w_r	Reception power consumption	0.0588 W
w_s	Sleep power consumption	2.4×10^{-7} W
t_{pr}	Preamble duration	0.26 ms
t_{ack}	ACK window duration	0.26 ms
t_p	Packet duration	0.93 ms
ϕ	Duty cycle	Calculation specific
ϕ_{ori}	The original duty cycle nodes adopt	Calculation specific

the sensor nodes can periodically switch on and off due to a normalized duty cycle. The duty cycle (denoted as ϕ) of a sensor node is defined as its active/working period.

$$\phi = \frac{t_a}{t_a + t_{\text{off}}}, \quad (3)$$

where t_a is the duration when the sensor node is active and t_{off} is the time duration when the sensor node is dormant. When a node is in active state, it can detect event and communicate with its neighbor nodes. And when it is in dormant state, it will close all the communication unit; therefore, it can neither detect event nor communicate with other nodes.

When a node is in active state, the power consumption is mainly composed of two parts: the power used to send or receive packets and the power to perform the Low-Power Listening (LPL) operations. When a node is in dormant state, the power consumption is composed of the wake-up timer.

The main parameters of system model adopted in this paper are similar to those in [30]. The device-specific values are taken from the internal data-sheet of a prototype sensor node by Thales. The values of the parameters are listed in Table 1.

3.4. Problem Statement. Similar to the utility-based routing in the previous researches [1], our goal is to design a routing scheme which has maximum utility of each packet delivery under the premise of not reducing the life time of the system. In this paper, we consider a WSN which is made up of N sensor nodes with a sink node situated at the center and n source nodes. The initial duty cycle of each sensor node is ϕ_{ini} . The initial benefit is μ and the benefit decay coefficient is ρ . Then, the issue is how to adjust the duty cycle of each node and determine the next-hop node to transmit packets, on the purposing of maximizing u_f .

Definition 3. The utility from the source node to the sink node is denoted as u_f . u_f depends on the delay and the routing selection. Let u_i stand for the utility from the source node ξ_i to the sink node; then the utility can be expressed as

$$\max(u_f) = \max \sum_{i \in \text{source } e} u_i. \quad (4)$$

And for the convenience of readers to understand this paper, Notations section summarizes the notations used in this paper.

4. Main Design of UADC Scheme

In this section, we focus on the duty cycle and delay analysis. First, we calculate the energy consumption of each node according to the data load. Then we calculate the duty cycle each node can adopt and adjust each node's duty cycle according to adaptive control of duty cycle (ACDC) algorithm. After adjusting the duty cycle of each node to decrease the end-to-end delay, we propose the utility-based adaptive duty cycle routing algorithm, that is, UADC.

4.1. Energy Consumption Analysis. In this subsection, we need to calculate the data load of each node first and then calculate the energy consumption of each node according to the data load. The value is given by theoretical analysis.

In planar networks, the network radius is denoted as R , the data generation rate is λ , and each node generates one data packet in a collection period, t_{duty} . According to [35], the data amount of the node whose distance to sink is d m can be calculated as L :

$$\begin{aligned} L_r^d &= \left(z + \frac{z(z+1)r}{2d} \right) \lambda, \\ L_r^t &= \left((z+1) + \frac{z(z+1)r}{2d} \right) \lambda, \end{aligned} \quad (5)$$

where $z = \lfloor (R-d)/r \rfloor$.

In layered networks, the data amount of a node is determined by the number of its parent nodes. Assume that the number of parent nodes of node ξ_i is ψ_i ; the data load L_t^i of node ξ_x can be calculated as

$$\begin{aligned} L_r^i &= \left(\sum_{m=1}^{\psi_i} \text{load}_m \right) \lambda, \\ L_t^i &= L_r^i + \lambda. \end{aligned} \quad (6)$$

The number of child nodes of node ψ_m is denoted as n_m ; thus load_m satisfies

$$\text{load}_m = \frac{L_t^i}{n_m}. \quad (7)$$

Therefore, the node which has more parent nodes or closer to the source node usually loads more data. The nodes which load more data will cost more energy. If all the nodes adopt the same duty cycle, there will be much energy left when the network dies. Assume that the initial total energy of node ξ_i is denoted as E_i ; the remaining energy of node ξ_i when the network dies is denoted as E_r ; then the energy utilization ratio ω_i of node ξ_i can be calculated as $(E_i - E_r)/E_i$. Through the previous analysis, we know that the energy utilization ratio can be improved by using the remaining energy. And the way to make full use of the remaining energy is to improve the duty cycle of the nodes. Thus we need to adjust the duty cycle of each node according to its remaining energy.

Theorem 4. *The total energy consumption of node ξ_i in one cycle satisfies*

$$E_{to} = \sigma_{LPL}^i \cdot t_{\text{duty}} + \sigma_r^i L_r^i + \sigma_t^i L_t^i. \quad (8)$$

Proof. The average power used for transmitting an alert packet satisfies

$$\sigma_t^i = w_t t_p + \left(\frac{\phi t_{\text{duty}}}{(t_{\text{pr}} + t_p)} + \frac{1}{2} \right) (w_t t_r + w_r t_{\text{ak}}), \quad (9)$$

where $w_r t_p$ is the energy spent for transmitting a data packet and $(\phi t_{\text{duty}}/(t_{\text{pr}} + t_p) + 1/2)(w_t t_r + w_r t_{\text{ak}})$ is the energy consumption for the preamble transmission in one period [18]. We take the arithmetic mean of the best cases and the worst cases as the average of preamble transmissions.

The energy used to receive a packet is $w_r t_p$ and the energy used to receive the preamble and the transmission of the ACK message [30] is $(w_r t_{\text{pr}} + w_t t_{\text{ak}})$. Thus, the power used for receiving a packet satisfies

$$\sigma_r^i = w_r t_p + (w_r t_{\text{pr}} + w_t t_{\text{ak}}). \quad (10)$$

According to [30], the power used for LPL satisfies

$$\sigma_{LPL}^i = w_r \phi + w_s (1 - \phi) - \tau_t^i - \tau_r^i. \quad (11)$$

Because we have calculated the energy used for transmitting and receiving the data in (9) and (10) in active state, subtracting τ_t^i and τ_r^i from σ_{LPL}^i is necessary in the above formula. Obviously, the node which is closer to sink node loads more data, so the time duration for transmitting or receiving packets is more and the time duration for performing LPL operations is shorter. σ_{LPL}^i will decrease with the increase of τ_t^i and τ_r^i .

According to [36], D_t^i can be calculated as

$$\tau_t^i = \left\{ w_s \left[\frac{(1-\phi)t_{\text{duty}}}{2} + t_{\text{pr}} + t_{\text{ak}} \right] + w_r t_{\text{pr}} \right\} \cdot \frac{L_t^i}{t_{\text{duty}}}. \quad (12)$$

According to [36], τ_r^i can be calculated as

$$\tau_r^i = \left[(t_{\text{ak}} + t_p) w_s + w_r t_{\text{pr}} \right] \cdot \frac{L_r^i}{t_{\text{duty}}}. \quad (13)$$

□

4.2. Duty Cycle Calculation. According to the energy consumption of each node under SDC scheme, that is, when the nodes in the network are initialized with the same duty cycle, we can calculate the largest duty cycle (denoted as ϕ_L) that each node can adopt on the premise of not reducing the lifetime of network. Then, according to the largest duty cycle of each node, we can perform the adaptive control of duty cycle (ACDC) algorithm we proposed in the next subsection to adjust the duty cycle of each node. ACDC algorithm is able to decrease the transmission delay, which results in a higher utility when transmitting packets. And the duty cycle of each node will reach an equilibrium point after several

adjustments, which is able to achieve a very low packet delay and guaranteed the lifetime of the network at the same time.

Suppose all the nodes except for the sink node have the same energy in the beginning. E_{\max} is the maximum energy consumption of the nodes in a certain flow of network and for another node, ξ_x , whose energy consumption is denoted as E_x in the same flow. The difference between E_{\max} and E_x is the remaining energy of node ξ_x :

$$E_x^\Delta = E_{\max} - E_x. \quad (14)$$

The energy that node ξ_x keeps when the network is dead can be used to prolong its active time in a period and the lifetime of the network is guaranteed. As the energy remains more, the active time of the node can prolong more.

Theorem 5. *Suppose a certain flow Θ in the network is composed of n sensor nodes. The set of sensor nodes is $\{\xi_1^\Theta, \xi_2^\Theta, \dots, \xi_n^\Theta\}$. When all the nodes are initialized with the same duty cycle, assume node ξ_n^Θ is the node which consumes most energy, and for another node, $\xi_n^\Theta (x \neq k)$, in the same flow, on the premise of not harming the lifetime of the network, the largest duty cycle node ξ_x^Θ can adopt satisfies*

$$\Phi_L^x = \frac{\mathcal{A}(L_r^x - L_r^{\max}) + \Phi_{\max} L_r^{\max} \mathcal{B} + \Phi_{\max} \mathcal{C}}{L_r^x \mathcal{B} + \mathcal{C}}, \quad (15)$$

where \mathcal{A} , \mathcal{B} , and \mathcal{C} are presented as (20), (21), and (22), respectively.

Proof. According to Theorem 4, the largest energy consumption of flow Θ satisfies

$$E_\Theta^{\max} = \sigma_{\text{LPL}}^{\max} \cdot t_{\text{duty}} + \sigma_r^{\max} L_r^{\max} + \sigma_t^{\max} L_t^{\max}, \quad (16)$$

where $\sigma_{\text{LPL}}^{\max}$, σ_r^{\max} , and σ_t^{\max} can be calculated by (2), (9), and (8), respectively.

Similarly, the energy consumption of node ξ_x^Θ satisfies

$$E_\Theta^x = \sigma_{\text{LPL}}^x \cdot t_{\text{duty}} + \sigma_r^x L_r^x + \sigma_t^x L_t^x. \quad (17)$$

Then the remaining energy of node ξ_x^Θ can be calculated as

$$\begin{aligned} E_x^\Delta &= E_{\max} - E_x \\ &= (\sigma_{\text{LPL}}^{\max} - \sigma_{\text{LPL}}^x) \cdot t_{\text{duty}} + (\sigma_r^{\max} L_r^{\max} - \sigma_r^x L_r^x) \\ &\quad + (\sigma_t^{\max} L_t^{\max} - \sigma_t^x L_t^x). \end{aligned} \quad (18)$$

The node ξ_x^Θ can obtain its largest duty cycle Φ_{\max}^x when $E_{\max} - E_x = 0$. Put (9), (10), and (11) into (15); the largest duty cycle Φ_L^x satisfies

$$\Phi_L^x = \frac{\mathcal{A}(L_r^x - L_r^{\max}) + \Phi_{\max} L_r^{\max} \mathcal{B} + \Phi_{\max} \mathcal{C}}{L_r^x \mathcal{B} + \mathcal{C}}, \quad (19)$$

where

$$\mathcal{A} = w_s \left(t_{\text{pr}} + t_{\text{ak}} + t_p + \frac{t_{\text{duty}}}{2} \right) - w_r \left(t_p - t_{\text{pr}} + \frac{t_{\text{ak}}}{2} \right) - w_r \left(t_{\text{ak}} + t_p + \frac{t_{\text{pr}}}{2} \right), \quad (20)$$

$$\mathcal{B} = \left(\frac{t_{\text{duty}}}{4(t_{\text{pr}} + t_{\text{ak}})} + \frac{w_s t_{\text{duty}}}{2} \right), \quad (21)$$

$$\mathcal{C} = [(w_r - w_s) t_{\text{duty}} + \mathcal{B}]. \quad (22)$$

□

In every data period, each node will store its energy consumption of the whole period. Then in the next period, when the node generates packet, the energy consumption in the last period will be included in the header of the packet. When the node receives packet from its neighbor node, it will compare its energy consumption with the energy consumption stored in the packet header, that is, the energy consumed by its neighbor node. If the node consumes more energy than its neighbor node, then its own energy consumption will replace the energy consumption in packet header. And continue this process until the packet is transmitted to sink node; thus the sink node can get the biggest energy consumption of the whole flow. And the sink will calculate the average E2E delay of every flow at every period. We denote the calculated E2E delay of flow in period p_i as D_i^m and the delay in the next period as D_{i+1}^m . If D_i^m and D_{i+1}^m satisfy

$$D_i^m - D_{i+1}^m \geq D^{\text{th}}, \quad (23)$$

where D^{th} is the threshold delay for performing ACDC algorithm, then the algorithm should continue performing in period p_{i+2} . The equation shows that, after the adjustment of ACDC algorithm in period p_{i+1} , there are still a lot of nodes having enough energy left to enlarge the duty cycle. So in period p_{i+2} , the algorithm should continue performing (when $p_i = 0$, the ACDC algorithm will be performed unconditionally). However, if (23) does not set up, which means nodes do not have enough energy remained to enlarge the duty cycle, the algorithm should stop to avoid declining the network lifetime. The network will perform the less adjustment rounds with the increase of D^{th} , but energy unused will be more and the E2E delay will be longer, which will not be shorter than D^{th} .

The method of feedback to adjust the duty cycle is as follows: when the sink node receives the packet, it will store the largest energy consumption of the entire flow E_{\max} in an ACK packet. When the node ξ_x receives a ACK packet, it will compare its own energy consumption E_x with E_{\max} . If $E_x = E_{\max}$, it will not adjust its duty cycle. If $E_x < E_{\max}$, it will compute its largest duty cycle Φ_L^x first and then enlarges its duty cycle by (24).

- (1) *Step 1. Generate packet.*
- (2) Store period energy consumption into the header of packet and transmit it to awake neighbor node.
- (3) *Step 2. Receive packet from neighbor node.*
- (4) Compare its own energy consumption, E_x , with the energy consumption, E_{\max} , stored in the header of packet.
- (5) **If** $E_x > E_{\max}$ **then**
- (6) $E_{\max} = E_x$.
- (7) **End if**
- (8) Transmit the packet to next awake node.
- (9) *Step 3. Receive energy-notification ACK packet.*
- (10) Compute its residual energy and largest duty cycle, then enlarge its duty cycle.
- (11) $E_x^\Delta = E_{\max} - E_x$.
- (12) **If** $E_x^\Delta = 0$ **then**
- (13) Maintain the duty cycle Φ_x .
- (14) **End if**
- (15) **Else**
- (16) Compute Φ_L^x by Eq. (15).
- (17) $\Phi_x^{\text{adj}} = \Phi_x(1 - \chi^\Delta) + \Phi_L^x \chi^\Delta$ // Enlarge duty cycle.
- (18) **End else**

ALGORITHM 1: ACDC for reducing transmission delay.

Assume that the duty cycle adjusting rate in every round adjustment is χ^Δ ; then the duty cycle of the node after adjustment, denoted as Φ_x^{adj} , satisfies

$$\Phi_x^{\text{adj}} = \Phi_x + (\Phi_L^x - \Phi_x) \chi^\Delta = \Phi_x (1 - \chi^\Delta) + \Phi_L^x \chi^\Delta. \quad (24)$$

This strategy can control the duty cycle of node dynamically according to its energy cost. From (19) we know the energy consumption in far-sink area is smaller; thus the duty cycle of the nodes in far-sink area is smaller. So the value of $(\Phi_L - \Phi)$ is larger, which results in a larger increment of the duty cycle.

The ACDC algorithm for reducing transmission delay on the premise of not harming the lifetime of network is given in Algorithm 1.

4.3. Utility Analysis. In the subsection, we first propose the way to calculate the average delay. Then, we define the meaning of cost in this paper. At last, we put forward the formula to calculate the expected utility and illustrate the formula through an example.

Theorem 6. *The average delay for the node ξ_i to transfer a packet to its neighbor satisfies*

$$d_\psi = \frac{(1 - \Phi)^2}{2} \cdot \frac{T_{\text{duty}}}{\psi} + t_p + t_{\text{ack}} + t_{\text{pr}}, \quad (25)$$

where ψ is the number of the relay nodes of node ξ_i .

Proof. As shown in Figure 6, assume that the wake-up instants of the relay nodes are equally distributed over a period T_{duty} , which are denoted as $\{W_1, W_2, \dots, W_\psi\}$. Then the whole period is composed of ψ equal disjoint intervals. Thus, a node can transfer its packet to its neighbor within T_{duty}/ψ because there must be a neighbor in active state. When we calculate the average one-hop transmission delay, only one interval is needed to be taken into account. We

suppose that there are only two states in the communication system of node, turn on and turn off. The probability that a node begins its transmission when the relay node is active is Φ , and the delay is $t_p + t_{\text{ack}} + t_{\text{pr}}$. On the other hand, the probability to be in off stage is $(1 - \Phi)$ for a node. We also simply calculate d_ψ as the arithmetic average between the best and worst cases. The best case is when a node starts transmitting its packet exactly at the time the relay node starts performing its LPL operations, so that the packet can be transmitted exactly after $t_p + t_{\text{ack}} + t_{\text{pr}}$, while, in the worst case, the transmitting node needs to wait for the whole duration of the sleeping interval. In addition, in order to start the communication, the worst case takes into account that two transmissions of the preamble may be required in order to start the communication. In this case, the delay is $(1 - \Phi) \cdot (T_{\text{duty}}/\psi) + t_p + t_{\text{ack}} + t_{\text{pr}}$. At last, we take into account the delay related to the case with the receiver on and scaled by its probability of being on; formula (25) for d_ψ can be obtained. \square

Considering the total energy consumption of node ξ_i in one period is denoted as E_i and the initial energy of node ξ_i is denoted as Te_i , then the cost of node ξ_i can be calculated as

$$c_i = \tau \frac{E_i}{Te_i}, \quad (26)$$

where τ is the adjustment coefficient.

In the previous work [2], the cost usually refers to the energy consumption when transmitting packets. If a node has a higher cost, it means that the node costs more energy in one period. However, in the actual situation, the same energy consumption has different effects on the nodes with different initial energy. Thus, in this paper, the cost is related to the ratio of E_i to Te_i . If a node has a higher cost in the network, it means that the energy consumption ratio is higher, and the node will have a shorter lifetime.



FIGURE 2: Linear network topology.

We consider a nonretransmission delivery path on linear network, which can be shown as in Figure 2.

The expected utility from sink node ξ_s to the sink node ξ_d is denoted as u_e . Assuming that the probability of successful transmission is known, we can get the value of delay and cost through the previous analysis. We can propose the formula by calculating the utility values in different cases.

Theorem 7. *The delay between nodes ξ_i and ξ_{i+1} is denoted as $d_{i,i+1}$. The expected utility U_e for transferring the packet with an initial benefit α and a benefit coefficient λ on linear network can be calculated as*

$$u_e = \prod_{i=0}^{n-1} p_{i,i-1} \left(\alpha - \lambda \sum_{i=0}^{n-1} d_{i,i+1} \right) - \sum_{i=0}^{n-1} c_{i,i+1} \prod_{j=0}^{i-1} p_{j,j+1}. \quad (27)$$

Proof. We can obtain (27) by calculating all the possible delivery cases and summing the utility values according to the probability of each case.

If the packet is transmitted successfully, denoted as suc, which means the packet delivery in each hop is successful, then, the whole delay is the sum of the delay in each hop, that is, $\sum_{i=0}^{n-1} d_{i,i+1}$. In addition, the probability of the successful delivery p_{suc} , the benefit b_{suc} , and the total cost c_{suc} can be calculated as

$$\begin{aligned} p_{\text{suc}} &= \prod_{i=0}^{n-1} p_{i,i+1}; \\ b_{\text{suc}} &= \alpha - \lambda \sum_{i=0}^{n-1} d_{i,i+1}; \\ c_{\text{suc}} &= \sum_{i=0}^{n-1} c_{i,i+1}. \end{aligned} \quad (28)$$

If the packet transmission is failed at the hop “ m to $m+1$ ”, ($0 \leq m \leq n-1$), denoted as fail- m , the benefit will drop to zero, and the probability, benefit, and total cost satisfy

$$\begin{aligned} p_{\text{fail-}m} &= (1 - p_{k,k+1}) \prod_{i=0}^{k-1} p_{i,i+1}; \\ b_{\text{fail-}m} &= 0; \\ c_{\text{fail-}m} &= \sum_{i=0}^{k-1} c_{i,i+1}. \end{aligned} \quad (29)$$

The expected value of the utility u_e satisfies

$$u_e = p_{\text{suc}} (b_{\text{suc}} - c_{\text{suc}}) + \sum_{k=0}^{n-1} p_{\text{fail-}k} (b_{\text{fail-}k} - c_{\text{fail-}k}). \quad (30)$$

Putting (29) and (30) into (28), (27) can be obtained. \square

Theorem 8. *The expected utility values of a node ξ_m and its relay node ξ_z can be calculated as*

$$u_m(b_m) = p_{m,z} u_z(b_z) - c_{m,z}. \quad (31)$$

Proof. According to (23), the utility values of node ξ_m and node ξ_z are

$$u_m(b_m) = \prod_{i=m}^{n-1} p_{i,i-1} \left(b_m - \lambda \sum_{i=m}^{n-1} d_{i,i+1} \right) - \sum_{i=m}^{n-1} c_{i,i+1} \prod_{j=0}^{i-1} p_{j,j+1}. \quad (32)$$

$$u_z(b_z) = \prod_{i=z}^{n-1} p_{i,i-1} \left(b_m - \lambda \sum_{i=z}^{n-1} d_{i,i+1} \right) - \sum_{i=z}^{n-1} c_{i,i+1} \prod_{j=0}^{i-1} p_{j,j+1}; \quad (33)$$

Comparing $u_m(b_m)$ with $u_z(b_z)$, we derive the following formula:

$$u_m(b_m) = p_{m,n} u_n(b_n) - \prod_{i=m}^{n-1} p_{i,i+1} (b_m - b_n - \lambda \cdot t_{m,n}) - c_{m,n}. \quad (34)$$

Since node ξ_m is the neighbor of node ξ_z , according to (28), the benefits of node m and node ξ_z satisfy

$$b_m = b_m + \lambda \cdot t_{m,n}. \quad (35)$$

\square

Figure 3 gives an example of how to calculate the expected utility. We can also directly get the utility value of node ξ_s according to (27). But in reality, a node usually derives its own expected value after it receives the expected utility value of its neighbors.

Figure 3 shows an example to calculate the expected utility. First, according to (28), we can get the benefit of node ξ_r and node ξ_s . Then, according to (31), the utility of node ξ_r and node ξ_s can be calculated. Therefore, the utility of sending the packet from node ξ_s to node ξ_d is 36.4.

4.4. Relay Node Determination. The algorithm consists of three stages: the adjusting stage, the initialization stage, and the routing stage. In the adjusting stage, each node should use the ACDC algorithm we proposed before to enlarge its duty cycle to decrease the delay. In the initialization stage, each

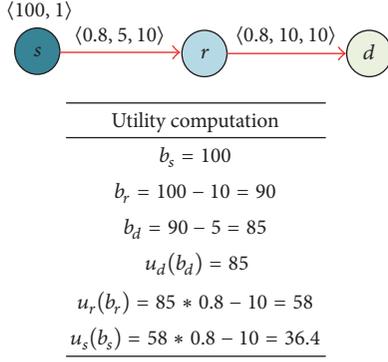


FIGURE 3: An example to calculate the expected utility.

node will calculate its largest expected utility and determine the next-hop relay according to the utility. In the routing stage, each node will transfer packets to its next-hop relay. Therefore, we can achieve the best expected utility. The way to determine the relay nodes is presented below.

First, each node will calculate the data load of each node first and then calculate the energy consumption of each node according to the data load. After computing the energy consumption of each node, each node will adjust its duty cycle to decrease the delay on the premise of not harming the lifetime of the network.

Then, each node will iteratively calculate its own utility values. From Theorem 8 we know that each node needs to know its neighbors' utility values to calculate its largest utility values. Therefore, it will send the requests to its neighbor nodes to ask for their utility values in the cycle.

After receiving the utility nodes from their neighbor nodes, each node will calculate the expected utility value of itself. The process continues until the iterative process converges. The process is beginning by the source node which will send the request $u_e = ?$ to its neighbor nodes.

Third, node ξ_i will record the information of the packet in a routing table, which is denoted as $\langle b_i, u_i(b_i), r_i(b_i) \rangle$, and b_i is the remaining benefit, $r_i(b_i)$ is the relay node determined by node ξ_i , and $u_i(b_i)$ is the expected utility of node ξ_i . When node ξ_i receives a request from a neighbor node or itself, it will produce a record $\langle b_i, u_i(b_i) = -\infty, r_i(b_i) = ? \rangle$ in its routing table. Next, it calculates its expected utility. Then, the node will determine the relay node by Theorem 8. At last, each node will choose the largest utility as its own expected utility and the corresponding neighbor node will be the next-hop relay. The expected utility $u_i(b_i)$ and the next-hop relay $r_i(b_i)$ satisfy

$$r_i(b_i) = \arg \max p_{i,j} u_j(b_j) - c_{i,j}, \quad (36)$$

where $b_j = b_i - \lambda \cdot t_{i,j}$, and

$$u_i(b_i) = p_{i,r_i} u_{r_i}(b_{r_i}) - c_{i,r_i}. \quad (37)$$

At last, the expected utility of the sink node is the remaining benefit of the sink node, which means

$$u_s(b_s) = b_s. \quad (38)$$

Another core issue we discussed in this subsection is the retransmission times. For the problem, there are two different cases. One is the time slot being large enough; thus the retransmissions will happen in a single time slot. The other is only a single time slot being not large enough and the retransmissions may happen in multiple active time slots.

Theorem 9. Considering node ξ_i retransmits a packet l times to its neighbor node ξ_j in a single time slot. The expected utility, denoted by $u_i(b_i)|_l$, and the retransmission times l satisfy

$$u_i(b_i)|_l = \left[1 - (1 - p_{i,j})^l \right] u_j(b_j) - l c_{i,j}, \quad (39)$$

$$l = \left\lceil \frac{\ln c_{i,j} - \ln p_{i,j} u_j(b_j)}{\ln(1 - p_{i,j})} \right\rceil \quad (40)$$

$$\text{or } \left\lfloor \frac{\ln c_{i,j} - \ln p_{i,j} u_j(b_j)}{\ln(1 - p_{i,j})} \right\rfloor.$$

Proof. After l times retransmissions, the probability of successful transmission becomes $1 - (1 - p_{i,j})^l$. What is more, the cost is $l c_{i,j}$. Therefore, the expected utility after l times retransmissions can be calculated as

$$u_i(b_i)|_l = \left[1 - (1 - p_{i,j})^l \right] u_j(b_j) - l c_{i,j}. \quad (41)$$

According to (39), we can calculate the expected utility for the $l + 1$ retransmissions as l' :

$$u_i(b_i)|_{l+1} = \left[1 - (1 - p_{i,j})^{l+1} \right] u_j(b_j) - l c_{i,j}. \quad (42)$$

According to (41) and (42), we have

$$u_i(b_i)|_{l+1} - u_i(b_i)|_l = (1 - p_{i,j})^l p_{i,j} u_j(b_j) - c_{i,j}. \quad (43)$$

If l' is the optimal retransmission times, then it satisfies $(1 - p_{i,j})^l p_{i,j} u_j(b_j) - c_{i,j} = 0$; then we can get

$$l = \frac{\ln c_{i,j} - \ln p_{i,j} u_j(b_j)}{\ln(1 - p_{i,j})}. \quad (44)$$

Since l is a positive integer, the retransmission times can be calculated as

$$l^\Delta = \lceil l' \rceil \text{ or } \lfloor l' \rfloor. \quad (45)$$

□

Theorem 10. When node ξ_i retransmits a packet l times to its next-hop node ξ_j in multiple time slots, the expected utility, denoted by $u_i(b_i)|_l$, satisfies

$$u_i(b_i)|_l = \sum_{y=1}^l \left[(1 - p_{i,j})^{y-1} p_{i,j} \right] u_j \left(b_j - \lambda \left\lfloor \frac{y}{\mu} \right\rfloor T \right) - l c_{i,j}, \quad (46)$$

where μ is the most retransmission times which can be included in an active time slot.

```

(1) Compute  $l^\Delta$  by Eq. (45);
(2) Let  $l_{\text{low}} = 1$ , and  $l_{\text{high}} = l^*$ 
(3) while  $l_{\text{low}} \neq l_{\text{high}}$  do
(4)    $l_{\text{mid}} = \lfloor (l_{\text{low}} + l_{\text{high}})/2 \rfloor$ ;
(5)   Compute  $\Delta l(l_{\text{low}}) = u_i(b_i)|_{l_{\text{low}+1}} - u_i(b_i)|_{l_{\text{low}}}$  by using Eq. (48);
(6)   Compute  $\Delta l(l_{\text{mid}}) = u_i(b_i)|_{l_{\text{mid}+1}} - u_i(b_i)|_{l_{\text{mid}}}$  by using Eq. (48);
(7)   if  $\Delta l(l_{\text{low}})\Delta l(l_{\text{mid}}) < 0$  then
(8)      $l_{\text{high}} = l_{\text{mid}}$ ;
(9)   else
(10)     $l_{\text{low}} = l_{\text{mid}}$ ;
(11)  $l^* = l_{\text{low}}$ 

```

ALGORITHM 2: Determining the optimal retransmission times.

Proof. Compared to the retransmission in a single time slot, the retransmissions in multiple time slots will result in an extra delivery delay. The l times retransmission will occupy $\lfloor l/\mu \rfloor$ active time slots.

Assume that node ξ_i transfers packets to its next-hop node ξ_j with l retransmission times. Then, probability of successful transmission of y th ($0 \leq y \leq l$) retransmission is $(1-p_{i,j})^{y-1}p_{i,j}$. The retransmission will cause a corresponding delivery delay $\lfloor y/\mu \rfloor T$. Thus, the remaining benefit of node ξ_j will decrease $\lambda \lfloor y/\mu \rfloor T$, and the expected utility of node ξ_j becomes $u_j(b_j - \lambda \lfloor y/\mu \rfloor T)$. What is more, the cost of l times retransmissions is lc_{ij} . Therefore, the expected utility with l times retransmissions can be calculated as (40).

Based on (40), we can compute the optimal retransmission times l to get the maximum utility $u_i(b_i)|_l$. We denote the optimal retransmission times as l^* . We have the following theorem about l^* . \square

Theorem 11. *When node ξ_i retransmits a packet l times to its next-hop node ξ_j , the expected utility of node ξ_i will decrease with the increase of the retransmission times. What is more, the best retransmission times l^* are less than or equal to the retransmission times in single time slot, which means $l^* \leq l^\Delta$.*

Proof. According to (39), the expected utility $u_i(b_i)|_l$ for the $(l+1)$ retransmission times satisfies

$$u_i(b_i)|_{l+1} = \sum_{y=1}^l \left[(1-p_{i,j})^{y-1} p_{i,j} \right] u_j \left(b_j - \lambda \left\lfloor \frac{y}{\mu} \right\rfloor T \right) - (l+1)c_{ij}. \quad (47)$$

With (39) and (47), we have

$$u_i(b_i)|_{l+1} - u_i(b_i)|_l = \left[(1-p_{i,j})^l p_{i,j} \right] u_j \left(b_j - \lambda \left\lfloor \frac{y}{\mu} \right\rfloor T \right) - c_{ij}. \quad (48)$$

In (44), $u_j(b_j - \lambda \lfloor y/\mu \rfloor T)$ will decrease along with the increase of the retransmission times l . Therefore, there is a number which satisfies $u_i(b_i)|_{l''+1} - u_i(b_i)|_{l''} = 0$.

What is more, if the retransmission time $l < l''$, the value of $u_i(b_i)|_{l+1} - u_i(b_i)|_l$ will be larger than zero; on the contrary, $u_i(b_i)|_{l+1} - u_i(b_i)|_l$ will be smaller than zero if $l > l''$. That is to say, along with the increasing of the number of retransmission times, the expected utility value $u_i(b_i)$ increases first and decreases later. The expected utility $u_i(b_i)|_l$ is best when $l = l''$. Then, the maximum expected utility can be achieved when $l^* = \lfloor l'' \rfloor$ or $l^* = \lceil l'' \rceil$.

Moreover, comparing (39) and (44), we can get $u_j(b_j - \lambda \lfloor y/\mu \rfloor T) < u_j(b_j)$; that is, $l^* \leq l^\Delta$. \square

4.5. UADC Algorithm. We can get the optimal retransmission times l^* according to the above analysis about retransmission. First, we calculate the optimal retransmission times l^Δ in a single time slot. Then, we use binary search in the range of $[1, l^\Delta]$ to find the optimal retransmission l^* . What is more, $\Delta l(l) = u_i(b_i)|_{l+1} - u_i(b_i)|_l$. Based on Theorem 10, the expected utility $u_i(b_i)|_l$ is a convex function. Therefore, $[l_{\text{low}}, l_{\text{high}}]$ is always the range of the optimal retransmission times. The detailed process is given in Algorithm 2.

Now, we present the utility-based adaptive duty cycle routing algorithm, that is, UADC algorithm. First, in the beginning of each round, we adjust the duty cycle of each node to decrease the end-to-end delay according to ACDC algorithm. Then, we compute the expected utility in a single time slot. At last, the nodes will calculate the expected utility and its retransmission times. Each node will determine the next-hop relay according to the expected utility. What is more, the UADC algorithm is given in Algorithm 3.

5. The Experimental Results and Analysis

In this section, extensive experiments were conducted to evaluate the performance of UADC algorithm. First, we analyze the energy consumption and delay to evaluate the performance of ACDC algorithm, and we compare the UADC algorithm with SDC (same duty cycle) scheme. In SDC scheme, the duty cycle of each node is 0.5. At last, the utility is evaluated under different parameters. The parameter settings refer to Table 1, and the experimental results are presented as follows.

Initialization

- (1) **for** each node **do**
- (2) Compute the data load by Algorithm 1;
- (3) Compute the energy consumption by Eq. (8)
- (4) Adjust the duty cycle of each node by Algorithm 2;
- (5) **for** each time slot in T **do**
- (6) **if** node ξ_i is active **then**
- (7) **if** node ξ_i is sink node **then**
- (8) Produce a request $u_i(b_i) = ?$ to itself;
- (9) Receive $u_j(b_j)$ from node ξ_j .
- (10) Determine the optimal retransmission times $u_i(b_i)_{l+1}$ and calculate the expected utility of nodes by Eq. (35) and Algorithm 2.
- (11) Determine $r_i(b_i)$ by Eq. (38) and record it in the routing table.
- (12) **if** receive the request from node ξ_j and node ξ_j is active **then**
- (13) Send $u_i(b_i) = u_i(b_i)_{l^*}$ to node ξ_j .

Routing

- (13) **for** each time slot in T **do**
- (14) **if** $r_i(b_i) = j$ and the node ξ_j is in active state **then**
- (15) Send the corresponding packets to node ξ_j by l^* retransmissions.

ALGORITHM 3: The UADC algorithm.

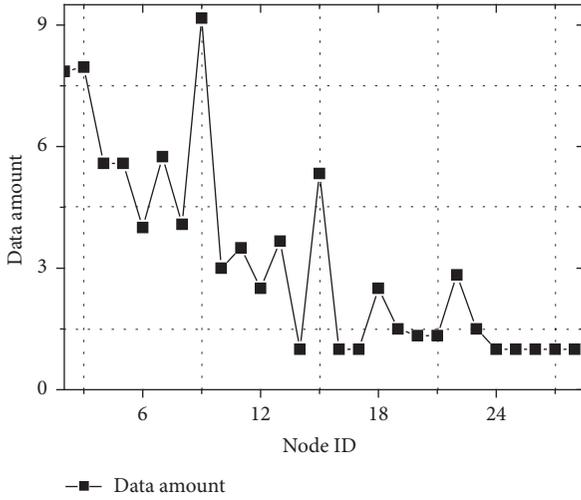


FIGURE 4: Data amount of nodes.

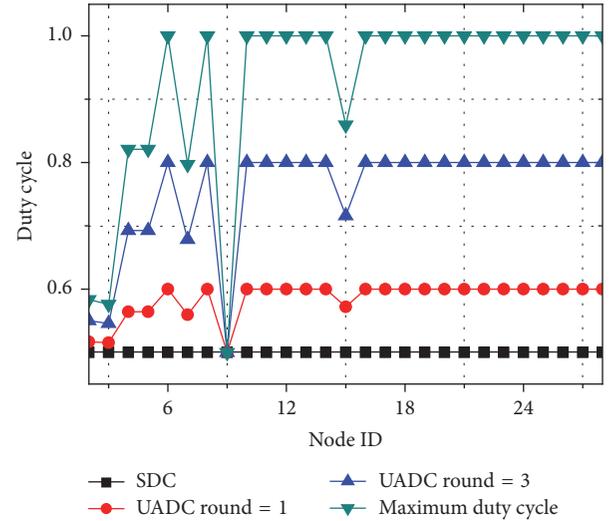


FIGURE 5: Duty cycle of nodes.

5.1. Energy and Delay Analysis. As is shown in Figure 4, the nodes which are closer to source node usually load more data, but there are some exceptions, like node ξ_8 . The reason for this phenomenon is that there are few nodes near the node; thus many nodes will choose node ξ_8 as relay node. According to the data amount of each node, we can calculate the maximum duty cycle of each node and adjust the duty cycle according to ACDC algorithm.

Through Figure 5, we know the duty cycle of the nodes will be increased after adjustment. The more data amount the node loads, the less duty cycle it improves. That is because the node which loads more data will cost more energy, under the premise of not harming the lifetime of the network, and the awake time of the nodes which load more data should be less.

Therefore, the duty cycle of the node which loads the most data remains unchanged.

Figure 6 shows that the energy consumption of the node which loads most data remains unchanged. And after adjustment, the energy consumption of other nodes increases, which improve the utilization of energy. The energy consumption increases but it cannot be larger than the energy consumption of node which loads the most data or the lifetime of the network will be decreased.

Figure 7 shows the remaining energy of nodes after several rounds. As we can see, the more the adjust times are, the less the remaining energy is, which proves the view that our algorithm can improve the utilization of energy. And when the remaining energy of the node is less, the energy

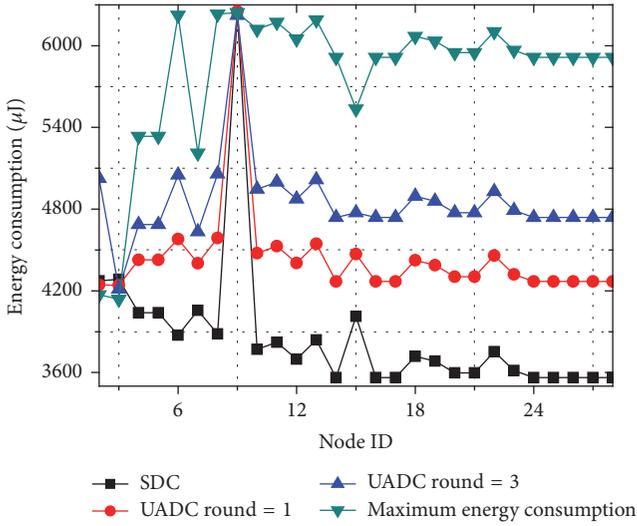


FIGURE 6: Energy consumption of nodes.

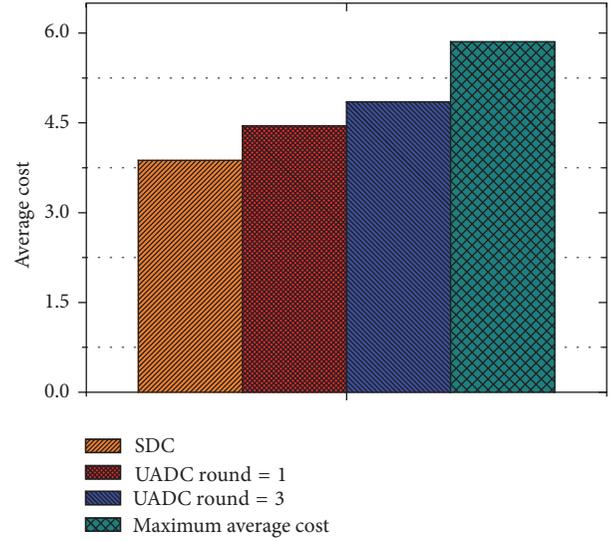


FIGURE 8: Average cost.

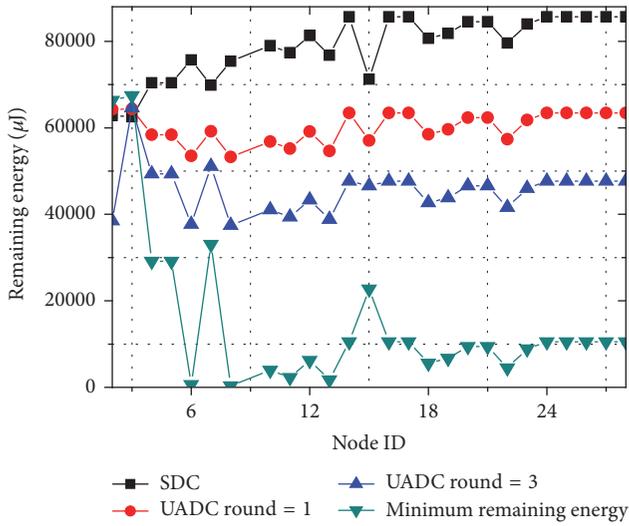


FIGURE 7: Remaining energy of nodes.

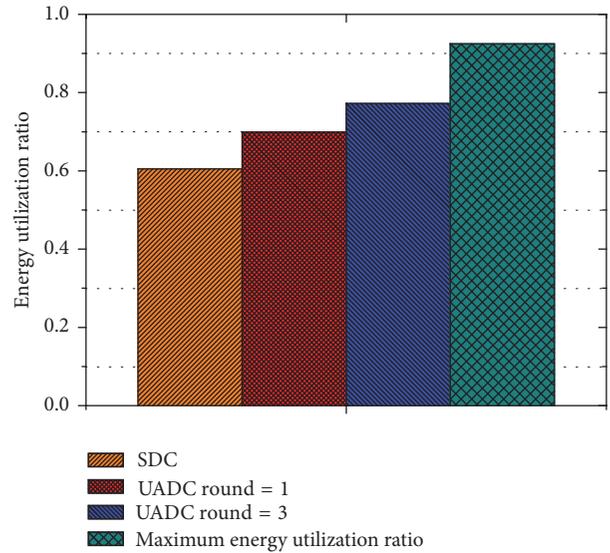


FIGURE 9: Energy utilization ratio.

consumption is more, thus the duty cycle of the node is higher.

According to the energy consumption and (26), we get the average cost of nodes. Through Figure 8, we can find that the average cost increases due to the increase of energy consumption. What is more, comparing Figure 8 with Figure 12, we find that UADC algorithm decreases the average delay but increases cost when transmitting packages.

Figure 9 shows the energy utilization ratio; as we can see, through UADC algorithm, we can improve the energy utilization ratio of more than 32.3% at most. That is because UADC algorithm takes advantage of the energy of every node better than SDC algorithm. Thus UADC can extend the network lifetime effectively.

As is shown in Figure 10, the average energy consumption increases with the increase of the number of adjustments.

Compared with Figure 13, we can conclude that the ACDC algorithm decreases the transmission delay by increasing the energy consumption of nodes. Because the duty cycle of node ξ_9 remains unchanged, the energy consumption and delay of the node also remain unchanged. For other nodes, the duty cycle increases so the energy consumption increases; thus the transmission delay decreases.

Figure 11 shows the delay of nodes in our simulation. We can see that the delay of nodes can be decreased after the adjustment of ACDC algorithm, and the delay of node ξ_9 remains with the same value because it loads the most data amount. The average delay is shown in Figure 12; through UADC algorithm, the average delay can be reduced by 77%, which greatly demonstrates the efficiency of UADC to improve the delay performance of network.

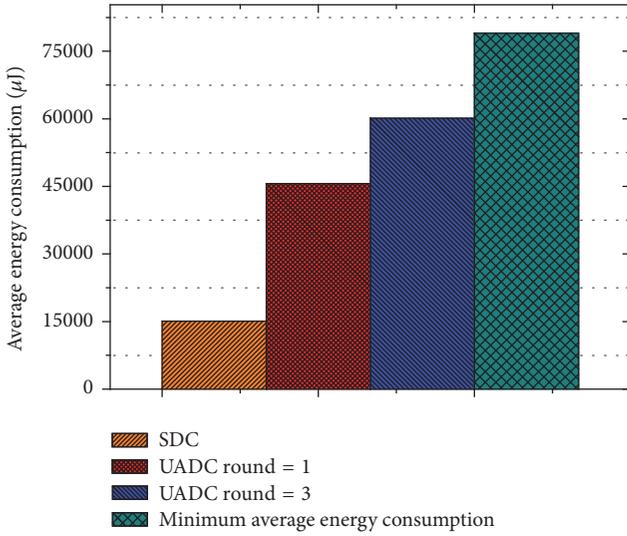


FIGURE 10: Average energy consumption.

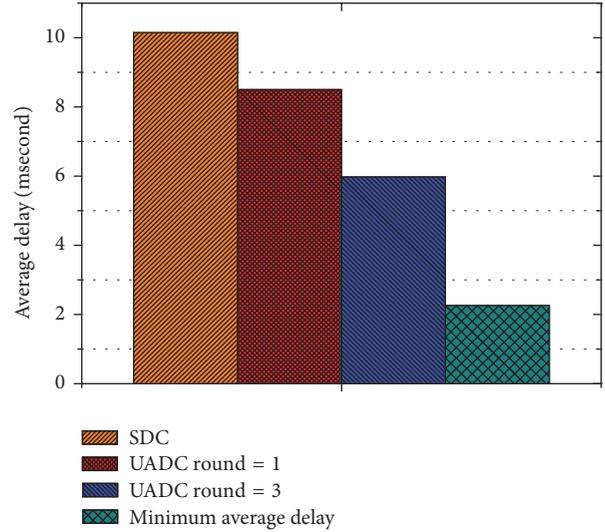


FIGURE 12: Average delay.

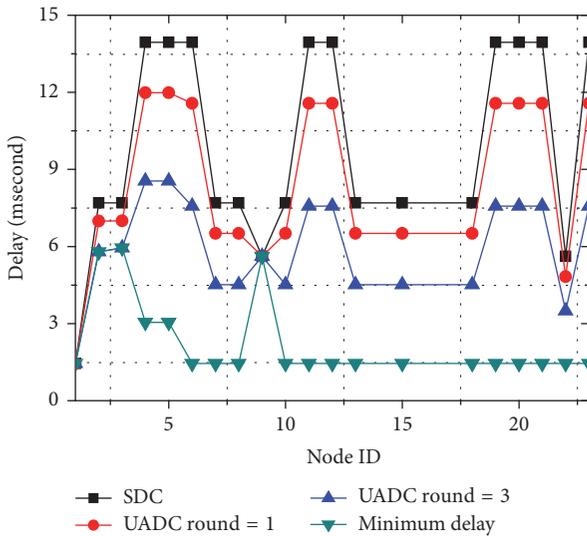


FIGURE 11: Delay of nodes.

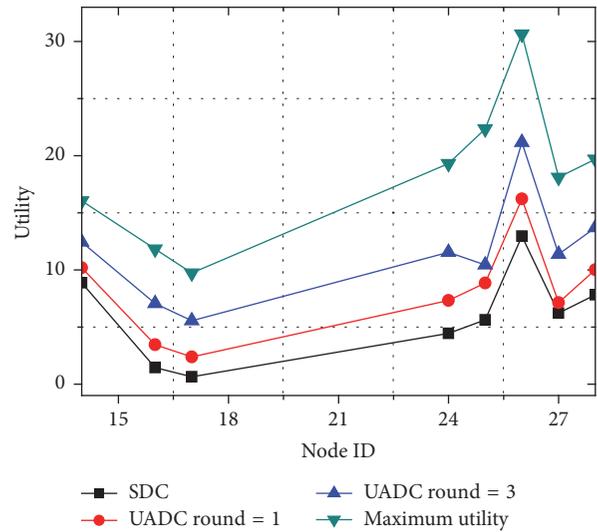


FIGURE 13: Utility of nodes.

5.2. *Utility Analysis.* After adjusting the duty cycle of nodes by ACDC algorithm, the transmission delay decreases; then the nodes will adopt UADC algorithm to determine the relay nodes to transmit packet and determine the routing. In this subsection, we evaluate the performance of UADC algorithm on utility.

Figure 13 shows the utility of several paths. The eight nodes are the initial nodes. As we can see, compared with SDC scheme, the utility of nodes increases much after adjusting the duty cycle of nodes. The average utility refers to the average value of utilities of all the packet deliveries. The results are shown in Figure 14, UADC algorithm increases the utility by 310.1 percent on average. Through the previous analysis, it is proved that the main reason that the utility increases is the decreasing of delay.

Then, we evaluate the performances on average utility under different parameters. As is shown in Figure 15, the average utility increases with the increase of the initial benefit. Similarly, Figure 16 shows that the average utility decreases with the increase of the initial benefit. What is more, with the increase of benefit decay coefficient, the average utility of UADC increases more than SDC. And with the increase of benefit decay coefficient, the average utility of UADC decreases less than SDC. Altogether these prove the effectiveness and practicability of UADC.

At last, we change both the initial benefit and the benefit decay coefficient at the same time to record the change of average utility of the UADC algorithm, and we compare it with SDC scheme. As we can see from Figure 17, the utility increases with the increase of initial benefit and the decrease of benefit decay coefficient. For SDC scheme, the

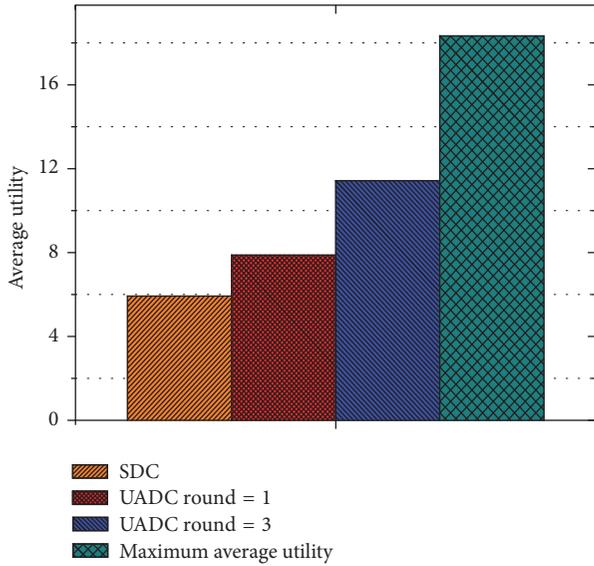


FIGURE 14: Average utility.

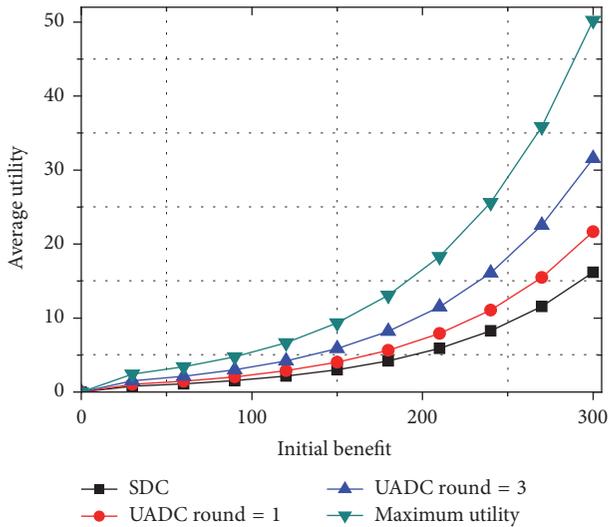


FIGURE 15: Average utility of different initial benefit.

conclusion is still right, but based on Figure 18, we conclude that compared with UADC algorithm, the rising rate of utility is slower and the declining rate is faster.

Through the previous analysis, it is proved that the UADC is able to increase energy efficiency, reduce transmission delay, and keep long lifetime at the same time, which prove the validity of our proposed algorithm.

6. Conclusion

In this paper, we present an energy efficient adaptive duty cycle routing scheme for Internet of Things, which takes reliability, benefit, duty cycle, energy consumption, and delay into consideration at the same time. To begin with, we analyze

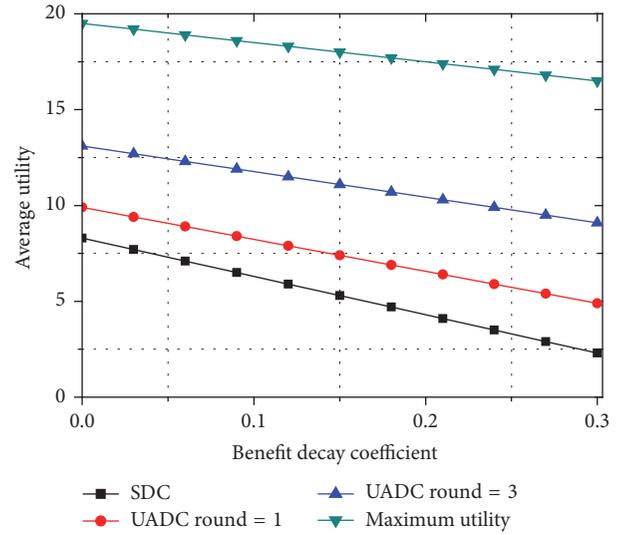


FIGURE 16: Average utility of different benefit decay coefficient.

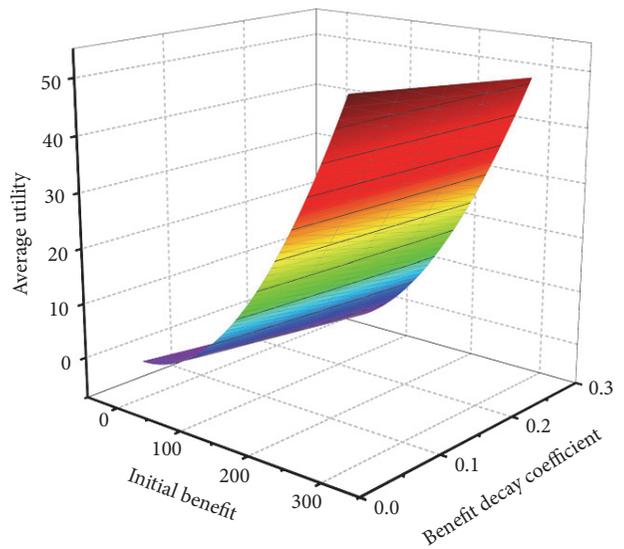


FIGURE 17: Average utility of UADC.

the energy consumption of each node according to the data amount it loads. Then, based on the energy consumption of each node, we analyze the duty cycle each node can adopt and we propose ACDC algorithm to adjust the duty cycle. Thus we can get the transmission delay between nodes according to the duty cycle. Next, we present the way to calculate the expected utility and choose the relay nodes. What is more, we analyze the retransmission times to maximize the utility. Based on the previous analysis, we present the utility-based adaptive duty cycle routing algorithm. According to the simulation results, it is indicated that the scheme can improve the utility on the premise of not harming the lifetime of network.

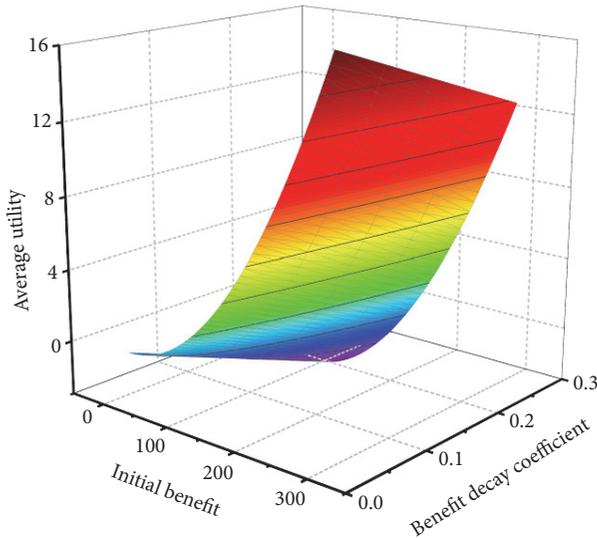


FIGURE 18: Average utility of SDC.

Notations

- σ_{LPL} : The power required when performing the LPL operations
 σ_r : The power used for receiving a packet
 σ_t : The power used for transmitting an alert packet
 σ_s : The power used when node is in sleep stage
 L_t : The transmitting packets load of a node
 L_r : The receiving packets load of a node
 λ : Data generation rate
 d_ψ : The average delay of the node with ψ relay nodes
 c : The cost of the node
 u : The utility of the node.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

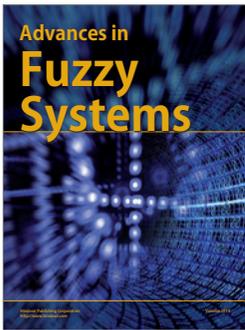
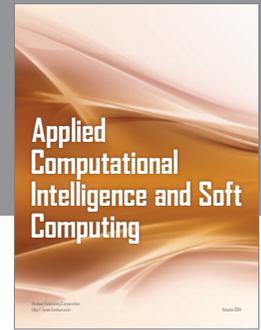
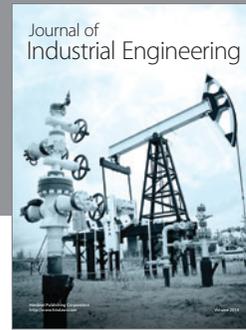
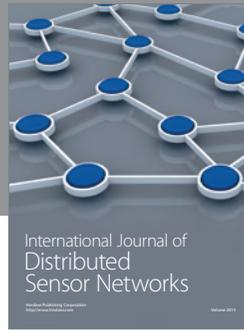
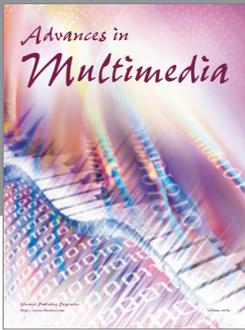
Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61379110, 61073104, 61572528, 61272494, and 61572526), the National Basic Research Program of China (973 Program) (2014CB046305), and Central South University undergraduate free exploration project (201510533283).

References

- [1] T. Han and N. Ansari, "Provisioning green energy for base stations in heterogeneous networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 7, pp. 5439–5448, 2016.
- [2] N. Zhao, F. R. Yu, and V. C. M. Leung, "Wireless energy harvesting in interference alignment networks," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 72–78, 2015.
- [3] N. Zhang, N. Cheng, A. T. Gamage, K. Zhang, J. W. Mark, and X. Shen, "Cloud assisted HetNets toward 5G wireless networks," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 59–65, 2015.
- [4] T. Han and N. Ansari, "A traffic load balancing framework for software-defined radio access networks powered by hybrid energy sources," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 1038–1051, 2016.
- [5] F. Ye, Y. Liang, H. Zhang, X. Zhang, and Y. Qian, "Design and analysis of a wireless monitoring network for transmission lines in smart grid," *Wireless Communications and Mobile Computing*, vol. 16, no. 10, pp. 1209–1220, 2016.
- [6] N. Ansari, T. Han, and M. Taheri, "Gate: greening at the edges," *IEEE Wireless Communications*, vol. 23, no. 2, pp. 62–69, 2016.
- [7] T. Han and N. Ansari, "Powering mobile networks with green energy," *IEEE Wireless Communications*, vol. 21, no. 1, pp. 90–96, 2014.
- [8] N. Zhang, H. Liang, N. Cheng, Y. Tang, J. W. Mark, and X. S. Shen, "Dynamic spectrum access in multi-channel cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 11, pp. 2053–2064, 2014.
- [9] Z. Chen, A. Liu, Z. Li, Y. Choi, and J. Li, "Distributed duty cycle control for delay improvement in wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 10, no. 3, pp. 559–578, 2017.
- [10] X. Liu, M. Dong, K. Ota, L. T. Yang, and A. Liu, "Trace malicious source to guarantee cyber security for mass monitor critical infrastructure," *Journal of Computer and System Sciences*, 2016.
- [11] M. Xiao, J. Wu, and L. Huang, "Time-sensitive utility-based single-copy routing in low-duty-cycle wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 5, pp. 1452–1465, 2015.
- [12] F. Ye, Y. Qian, and R. Q. Hu, "Energy efficient self-sustaining wireless neighborhood area network design for smart grid," *IEEE Transactions on Smart Grid*, vol. 6, no. 1, pp. 220–229, 2015.
- [13] N. Zhang, H. Zhou, K. Zheng, N. Cheng, J. W. Mark, and X. S. Shen, "Cooperative heterogeneous framework for spectrum harvesting in cognitive cellular network," *IEEE Communications Magazine*, vol. 53, no. 5, pp. 60–67, 2015.
- [14] A. Liu, Q. Zhang, Z. Li, Y. Choi, J. Li, and N. Komuro, "A green and reliable communication modeling for industrial internet of things," *Computers & Electrical Engineering*, 2016.
- [15] S. He, J. Chen, X. Li, X. S. Shen, and Y. Sun, "Mobility and intruder prior information improving the barrier coverage of sparse sensor networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 6, pp. 1268–1282, 2014.
- [16] X. Liu, A. Liu, and C. Huang, "Adaptive information dissemination control to provide diffdelay for the internet of things," *Sensors*, vol. 17, no. 1, article 138, 2017.
- [17] M. Tahir and R. Farrell, "A cross-layer framework for optimal delay-margin, network lifetime and utility tradeoff in wireless visual sensor networks," *Ad Hoc Networks*, vol. 11, no. 2, pp. 701–711, 2013.
- [18] R. Zhang, O. Berder, J.-M. Gorce, and O. Sentieys, "Energy-delay tradeoff in wireless multihop networks with unreliable links," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1306–1321, 2012.
- [19] Y. Chen, C.-N. Chuah, and Q. Zhao, "Network configuration for optimal utilization efficiency of wireless sensor networks," *Ad Hoc Networks*, vol. 6, no. 1, pp. 92–107, 2008.

- [20] M. Dong, K. Ota, L. T. Yang, A. Liu, and M. Guo, "LSCD: a low-storage clone detection protocol for cyber-physical systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 5, pp. 712–723, 2016.
- [21] S. He, D. Shin, J. Zhang, J. Chen, and Y. Sun, "Full-view area coverage in camera sensor networks: dimension reduction and near-optimal solutions," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7448–7461, 2016.
- [22] Q. Xu, Z. Su, and S. Guo, "A game theoretical incentive scheme for relay selection services in mobile social networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 6692–6702, 2016.
- [23] H. Li, D. Liu, Y. Dai, and T. H. Luan, "Engineering searchable encryption of mobile cloud networks: when QoE meets QoP," *IEEE Wireless Communications*, vol. 22, no. 4, pp. 74–80, 2015.
- [24] Z. Su, Q. Xu, M. Fei, and M. Dong, "Game theoretic resource allocation in media cloud with mobile social users," *IEEE Transactions on Multimedia*, vol. 18, no. 8, pp. 1650–1660, 2016.
- [25] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 3, pp. 312–325, 2016.
- [26] Y. Liu, A. Liu, and Z. Chen, "Analysis and improvement of send-and-wait automatic repeat-reQuest protocols for wireless sensor networks," *Wireless Personal Communications*, vol. 81, no. 3, pp. 923–959, 2015.
- [27] H. Li, X. Lin, H. Yang, X. Liang, R. Lu, and X. Shen, "EPPDR: an efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2053–2064, 2014.
- [28] T. Li, A. Liu, and C. Huang, "A similarity scenario-based recommendation model with small disturbances for unknown items in social networks," *IEEE Access*, vol. 4, pp. 9251–9272, 2016.
- [29] Z. Chen, A. Liu, Z. Li, Y. Choi, H. Sekiya, and J. Li, "Energy-efficient broadcasting scheme for smart industrial wireless sensor networks," *Mobile Information Systems*, vol. 2017, Article ID 7538190, 17 pages, 2017.
- [30] Y. Xu, A. Liu, and C. Huang, "Delay-aware program codes dissemination scheme in internet of everything," *Mobile Information Systems*, vol. 2016, Article ID 2436074, 18 pages, 2016.
- [31] Y. Hu, M. Dong, K. Ota, A. Liu, and M. Guo, "Mobile target detection in wireless sensor networks with adjustable sensing frequency," *IEEE Systems Journal*, vol. 10, no. 3, pp. 1160–1171, 2016.
- [32] M. Ettus, "System capacity, latency, and power consumption in multihop-routed SS-CDMA wireless networks," in *Proceedings of the Radio and Wireless Conference (RAWCON '08)*, pp. 55–58, Colorado Springs, Colo, USA, 2008.
- [33] Q. Zhang and A. Liu, "An unequal redundancy level-based mechanism for reliable data collection in wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, article 258, 2016.
- [34] Y. Gu and T. He, "Dynamic switching-based data forwarding for low-duty-cycle wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 12, pp. 1741–1754, 2011.
- [35] Y. Liu, M. Dong, K. Ota, and A. Liu, "ActiveTrust: secure and trustable routing in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 9, pp. 2013–2027, 2016.
- [36] S. Vasuhi and V. Vaidehi, "Target tracking using interactive multiple model for wireless sensor network," *Information Fusion*, vol. 27, pp. 41–53, 2016.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

