Utility-based Routing in Wireless Sensor Networks

Xiaoguang Li and Jie Wu
Department of Computer Information and Sciences
Temple University
Philadelphia, PA 19122, USA
Email: {xiaoguang.li, jiewu}@temple.edu

Abstract *Wireless sensor networks* (WSNs) have been proposed for monitoring physical environments. The applications in WSNs have comprised a wide variety of scenarios. The design of routing protocols in WSNs becomes more complicated than the traditional network when we consider the energy cost, throughput, reliability, and delay as routing metrics. Selecting a particular routing protocol mainly depends on the capabilities of the nodes, and on the requirements of the application. In this chapter, we will briefly discuss the existing utility-based routing protocols for WSNs. We put them into several categories according to their utility properties, such as delay, cost, and packet delivery ratio. In addition, we will also cover the composition-based utility for wireless networks and its extensions in low duty-cycle WSNs.

1 Introduction

Over the last few years, *wireless sensor networks* (WSNs) have been used in many applications, such as military surveillance, infrastructure protection, and scientific exploration [1, 2, 3, 4]. The major task of WSNs is to monitor environmental changes and report unexpected events to the destination.

The special features of WSNs bring out new challenges. One of the features is the lifetime of a sensor node, which is constrained by the battery. Thus, to reduce the energy cost, the consideration of energy efficiency is often preferred in a WSN design. Moreover, these problems are complicated by the lossy links and collisions during communication among wireless sensor nodes. In practice, all of these utilities in WSNs are available in different forms with many individual peculiarities. Obvious trade-offs include accuracy, dependability, energy consumption, delay, reliability, and so on.

Unlike the prior works about WSNs, which mainly focus on the design of MAC protocols, we will briefly take an overview of algorithmic methods

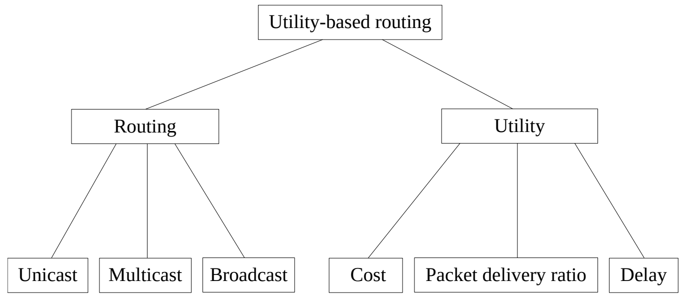

Fig. 1 Utility-based routing in WSNs.

which are related to the routing protocols. The routing protocol is designed to obtain a route for data transmission from the source to the destination. The route is selected based on the routing metric for different application requirements. In multi-hop networks, when a source node wants to send its packets to a destination, the intermediate node has to decide which neighbor an incoming packet should be forwarded to, so that it eventually reaches the destination.

As the routing protocol plays an important role in determining the path, a good application is dependent on the routing efficiency. Challenges in routing protocol design are very critical, based on different characteristics in WSNs. This chapter presents a survey on the routing designs of WSNs, based on a selected utility. This chapter aims at providing the basic knowledge on utility-based routing designs in WSNs. The readers are expected to acquire the recent studies and techniques on developing routing protocols in WSNs. We will first introduce the delay-based, packet-delivery-ratio-based, and energy-based utilities for routing protocols. Then, we will offer a special type of routing protocol based on composite utility.

2 Background

In this section, we will offer the background of utility-based routing. In addition, we will also introduce the other related issues regarding the routing design in WSNs.

2.1 Utility-based routing

Intuitively, the utility-based routing is composed of routing and utility, as shown in Figure 1. The routing designs are dominated by different forms of routing processes, such as unicast, multicast, and broadcast. The utilities are the routing metrics, such as cost, packet delivery ratio, and delay. Depending on the application of the sensor network, the utility-based routing can be continuous, event-driven, query-driven, or a hybrid. For the continuous model, the node sends data periodically. In the event-driven and query-driven models, the transmission is triggered when an event occurs or when a query is generated. However, the data delivery model mentioned above may coexist in the network. This is needed to accommodate different types of data delivery. There are two parts for utility-based routing: the routing protocols, and utilities, as shown in Figure 1. Many routing protocols have been proposed, based on the requirements of different applications and quality of service. In the real application, three transmission patterns are used for data delivery: unicast, multicast, and broadcast.

2.2 Objectives of different utilities

WSNs are widely used for environmental sensing and data processing, with extremely low energy and cost. The utilities of routing in WSNs are commonly discussed by delivery ratio, throughput, delay, saving cost (hop count, energy), and composition utility (combination of them). In this part, we will discuss the objectives of different metrics.

(1) Packet delivery ratio (PDR): The packet delivery ratio is estimated by sending a number of packets in a short period of time. The receiver will compute the percentage of received packets. Thus, the purpose of improving the delivery ratio is to reduce the delay and cost. Consequently, we can achieve both time and energy efficiency.

(2) Delay: During the transmission process, the packet is delayed at each node or during the data delivery. Especially in low duty-cycling, the sensor has to wait a certain time period to transmit the packet. Several issues need to be considered for the routing design, such as the expected end-to-end delay, packet delay, and sleep scheduling problems.

(3) Energy cost: The energy cost is also important in path-selection. As redundant packets may consume more energy, the metric can be designed by reducing any unexpected transmissions and real-time scheduling. In recent works, the expected transmission cost and real-time energy cost were proposed for selecting the path.

To represent the network topology, the weighted graph has been proposed. Given a weighted graph $G = (V, E, W)$, V is the set of vertices, E is the set of links, and W is the set of weights for the links. As shown in Figure 2, the weight of a link could be energy cost, delay, reliability or other conditions.

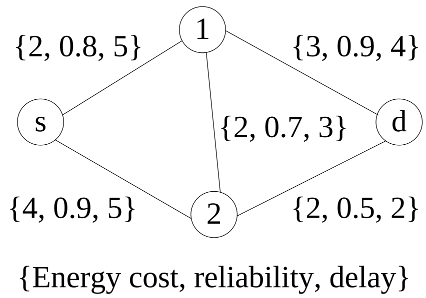

Fig. 2 An example of weighted graph.

2.3 Reporting model

Unlike wired networks, wireless channels are, by nature, error-prone [5]. Thus, neighboring nodes might not successfully receive messages. This means that transmission over a wireless connection is unreliable. Reliability is defined as the ability of the network to ensure reliable data transmission in a network structure that is continuously changing.

In low duty-cycle WSNs, many factors will affect the reliability of the link, such as *packet-error rate* (PER), buffer size, and duty-cycle. In wireless networks, packet errors are common, due to fading caused by the environment and interference from other wireless devices. Another problem is that if the nodes are randomly deployed, the inadequate locations and distances cause unreliability. Packet buffering in WSNs is limited, due to memory constraints. When a buffer is full, a packet must be dropped, which reduces reliability. To solve the reliability problem, several schemes have been proposed. Opportunistic and retransmission-based routing are two typical methods, as shown in Figure 3.

If the transmission fails, a retransmission may be needed for the delivery. For the first case, the sender can select another forwarder to transmit the packet, as shown in Figure 3a. This is called "opportunistic forwarding"[6]. In the other case, the node can simply retransmit the packet using the same link., as shown in Figure 3b. The transmission count, delivery ratio, or other utilities are provided to measure the metric.

Apart from different kinds of utilities, many works have been conducted about forwarding methods. In WSNs, the simplest forwarding method is *flooding*. Once a node receives the packet, the node will forward the packet to all of its neighbors. In this way, the packet is surely forwarded to the destination, as long as the network is connected. To avoid cyclic transmissions, the node should forward the packets to the neighbor it has not seen. To avoid useless propagation, packets usually have an expiration time, i.e, *time to live* (TTL) or maximum number of hops.

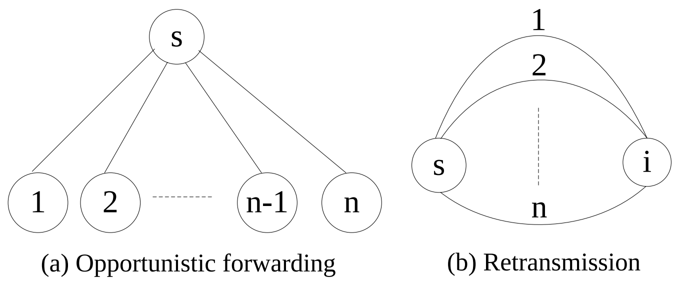

Fig. 3 Forwarding methods.

Probabilistic broadcast approaches, broadly called “gossip”, offer a simpler alternative to deterministic routing. With gossiping, nodes in the network are required to forward packets with a pre-specified probability, $p_{gossip} \leq 1$. The main idea is that the proper p_{gossip} will make the entire network receive the broadcast message with q very high probability. Recent research [7] has mentioned that the correct value of p_{gossip} is closely associated with the topology of the network.

2.4 Additional issues

In addition to the above utilities, other factors may also affect the network performance. In this subsection, we will offer other facts related to the routing design.

2.4.1 WSNs and low duty-cycle WSNs

Many applications in WSNs need a long time energy conservation due to limited energy supply. The special feature of the applications in WSNs is that the sensors are equipped with limited energy. Thus, it is desirable to turn off the radios when the sensors do not need to participate in the data delivery.

To resolve the conflicts, it is necessary to reduce the communication cost and duty-cycles. B-MAC [8] is one of them. B-MAC supports dynamic reconguration and provides bidirectional interfaces for system services to optimize performance, whether it be for throughput, latency, or power conservation. These sensors construct the *low duty-cycle WSNs* [9]. There are two states for the sensors: active or sleep. Usually, when a sensor node is in active mode, it can listen to the channel to receive the packets or transmit the packets. Various MAC protocols have been proposed in low duty-cycle WSNs. The

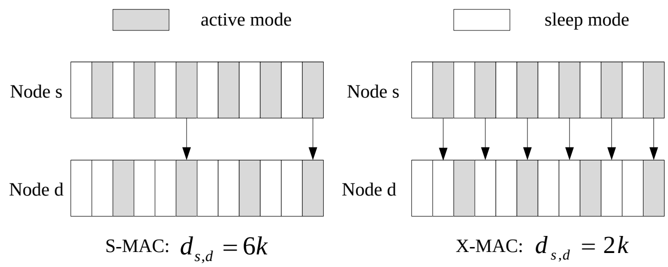

Fig. 4 Low duty-cycle WSNs

difference relies on the synchronization mechanism. Some MAC protocols require both of the nodes to be in active mode for data transmission, such as S-MAC [10]. The end-to-end delay is the least common multiple (LCM) of the two nodes. To lower the end-to-end delay latency, other MAC protocols were proposed. The sensors can still listen/overhear the packets by providing additional energy when they are in sleep mode. X-MAC [11] is one of them. With these MAC protocols, the end-to-end delay is the "wake up" period of the node. Figure 4 shows an example of two MAC protocols. Suppose nodes s and d are the neighboring nodes. The working schedules for nodes s and d are $2k$ and $3k$, respectively. The arrow lines show the schedule when two nodes can communicate with each other.

In this way, the energy can be consumed during: (1) Network setup: during this process, the sensors wake up, re-open network connections, and initialize the sensors. (2) Data processing: when the sensors are in active mode, they can transmit the packets. (3) Tear down: the sensors close the network connection, reset, and go into sleep mode. (4) Maintenance: The sensor nodes are in sleep mode. Very little energy will be consumed.

2.4.2 Topology

The topological deployment of WSNs is important. This could either be determined or random. For determined deployment, the sensors can be deployed along the road-side, or at a metro station, etc. The Sand [12] sensor network for target tracking and the CitySense [1] network for urban monitoring are the instances where optimal patterns can be provided [13]. However, lots of routing protocols are designed for random deployment.

Considering different deployments, it is important that a path exist from the source to destination. In other words, it is necessary to ensure the connectivity of the network. Typically, there is an inverse relationship between scalability and reliability in WSNs. As the number of nodes in the network increases, it is more difficult to ensure reliability. More dynamics in the environment will increase the number of control packets during the routing process. Moreover, the network cannot afford the large amount of overhead caused by the dynamics, which will result in less reliability. The two basic problems in real applications are bad paths and links to the sink nodes.

As we have discussed, the routing design in WSNs mainly focuses on the *packet delivery ratio* (PDR), latency, and energy efficiency [14]. In the following, we will provide some recent work that relates to these metrics, plus some new ones through a composition metric. The notation list is provided in Table 1. To make it consistent, we use p, d, and c for the reliability, delay, and cost calculations, respectively. Table 1 shows the notation list used in this paper.

Table 1 Notation list

Parameter	**Description**
$p_{i,j}$	the reliability of link (i, j)
p_i	the packet reception probability at node i
t_i	the time slot at node i
ω_i	the working schedule of node i
$d_{i,j}$	the delay between nodes i and j
$c_{i,j}$	the cost of link (i, j)
$h_{i,j}$	the hop count between nodes i and j
τ	the time span for each time unit
$d(P)$	the delay for a path P
$\rho_{i,j}$	the duty-cycle rate of link (i, j)

3 Single utility-based routing

In this section, we will provide an overview of the recent works related with single utility-based routing design. The “single utility” means that the routing metric is designed for the purpose of improving the packet delivery ratio, lowering the end-to-end delay, or saving energy costs.

3.1 Packet delivery ratio

The packet delivery ratio has several cases: the *expected delivery ratio* (EDR), the *expected transmission count* (ETX), and the *quality of forwarding* (QoF). In the following, we will introduce several related approaches.

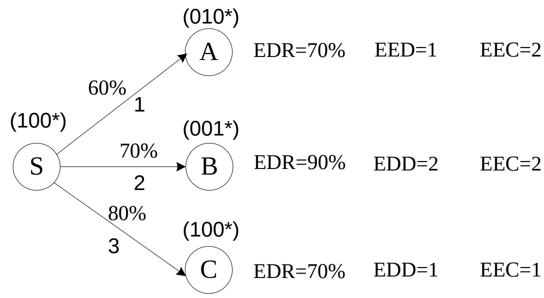

Fig. 5 An example of data forwarding in low duty-cycle WSNs.

3.1.1 Expected delivery ratio (EDR)

In opportunistic routing, each node s has n neighbors that construct the forwarding sequence, as shown in Figure 3a. For a given forwarding sequence, suppose that P_i is the overall probability that a packet is successfully delivered by the ith forwarder. It can be represented as follows [14]:

$$P_i = (\Pi_{j=1}^{i-1}(1 - p_j))p_i.$$

Therefore, the corresponding EDR for node s can be expressed as follows:

$$EDR = \sum_{i=1}^{n} P_i \cdot EDR_i. \tag{1}$$

Figure 5 shows an example of computing EDR. According to Eq. 1, EDR for node s is $0.6 \cdot 0.7 + (1-0.6) \cdot 0.7 \cdot 0.9 + (1-0.6) \cdot (1-0.7) \cdot 0.8 \cdot 0.7 = 0.74$, where $P_A = 0.6$, $P_B = (1-0.6) \cdot 0.7 = 0.28$ and $P_C = (1-0.6) \cdot (1-0.7) \cdot 0.8 = 0.096$.

3.1.2 Link correlation

The link correlation in low duty-cycle WSNs was first studied in [15]. In both indoor and outdoor experiments, they observed that if a packet is received by a sensor node with a low *packet reception ratio* (PRR), in most cases, this packet can also be received by the high PRR nodes. In order to quantify the relationship, conditional packet reception probability at node s was defined as $p_s(p_h|p_l)$, where p_h and p_l are for higher and lower link quality, respectively. p_h and p_l are the neighboring receivers of the sender s.

The work [15] was further extended to [16]. Traditionally, energy optimality in a designated flooding-tree is achieved by selecting parents with a smaller hop count and the best link quality. However, in this work, the authors studied the link correlation with which redundant transmission can be ignored. As an example, shown in Figure 6, node E wants to select the senders from A

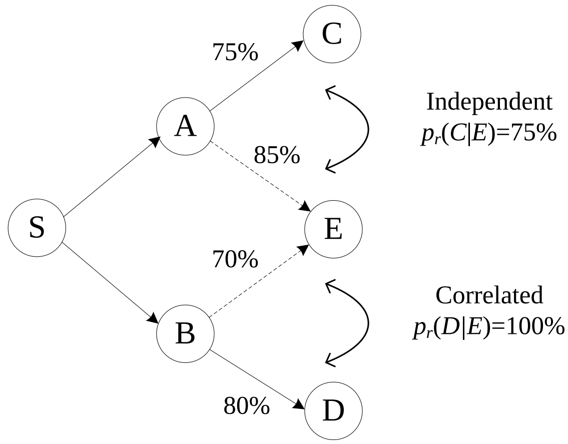

Fig. 6 An example of correlation.

and B. If we do not consider the link correlation, link AE should be selected for the transmission, since the quality of $AE(85\%)$ is higher than $BE(75\%)$. However, if link correlation is considered, the conclusion will not hold. Let p_1 and p_2 denote the link qualities for the two receivers, respectively. For k successful transmissions, the number of transmissions m needed for both of the receivers should satisfy the following equation:

$$\begin{aligned} p_r(m = k) &= p_r(m > k - 1) - p_r(m > k) \\ &= ((1 - p_1)^{k-1} + (1 - p_2)^{k-1} - p_{12}^{k-1}) \\ &- ((1 - p_1)^k + (1 - p_2)^k - p_{12}^k). \end{aligned}$$

Thus, the expected transmission $E(m)$ is:

$$E(m) = \sum_{k=1}^{+\infty} kP_r(m = k) = \frac{1}{p_1} + \frac{1}{p_2} - \frac{1}{1 - p_{12}}.$$

As shown in Figure 6, nodes E and C are independent. if node E chooses A as the forwarder, $p_{12} = (1-p_1)(1-p_2)$. $E(m) = 1/0.75 + 1/0.85 - (1/(1 - (1 - 0.75) \cdot (1 - 0.85))) = 1.47$. In the other case, since nodes D and E are correlated, if node E selects B as the forwarder, $p_{12} = 1 - p_E - p_D + p_E \cdot p_r(D/E) = 1 - 0.7 - 0.8 + 0.7 \cdot 1 = 0.2$. Here, $p_r(D/E)$ is the probability that node D can receive the packet if the packet can be received by node E. Then, $E(m) = 1/0.7 + 1/0.8 - 1/(1 - 0.2) = 1.43$.

The proposed correlated flooding includes two parts. One part is to collect the link quality information and partition the receivers into different groups. The senders will send a hello message to their neighbors. The receivers will record the information in a bitmap format (1: success, 0: failure). The distance

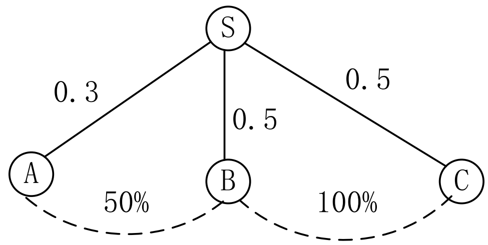

Fig. 7 An example of correlation.

between two correlated bitmaps is called the Hamming distance. It is defined as the number of different positions between the bitmaps. For example, the distance of [0111] and 0111 is 0, while the distance of [0111] and [1000] is 4.

The other part is the sender selection process. Each receiver may belong to multiple groups. In the sender selection process, the receiver will select the sender with the highest priority. If there are more than one, it will choose the one with the best link quality. The advantage of this method is that it makes use of the link correlation, and reduces the number of transmissions.

3.1.3 Expected transmission count (ETX)

Many routing metrics have been proposed to measure the link quality in wireless networks. The ETX [17] is one of the typical routing metrics. It can be represented as follows:

$$ETX = \frac{1}{p_f \cdot p_r}$$

where p_f is the probability that the packet can be successfully received. p_r is the reverse probability that the ACK can be successfully received. ETX selects paths with the minimum expected number of transmissions (including retransmissions) required to deliver a packet to its destination. For example, p_f and p_r equals to 0.7 and 0.8, respectively. Then, ETX of the link is $1/(0.7 \cdot 0.8) = 1.785$.

Basalamah et al. [18] proposed a new routing scheme which makes use of link correlation and opportunistic transmission scheme. The link correlation aware opportunistic routing was proposed to improve the performance exploiting the diversity gain. The motivation example can be explained as follows.

As shown in 7, the expected transmission count of the three links are $1/0.5 = 2$, $1/0.5 = 2$ and $1/0.3 = 3.33$ for nodes S to A, B, C, respectively. It is obvious that links (S, B) and (S, C) are better than link (S, C). Therefore, we have the expected transmission times:

$$ETX_{s,b,c} = \frac{1}{1 - \prod_i (1 - p_{s,i})}.$$

In this example, we have $ETX(s, b, c) = 1/(1 - 0.25) = 1.33$ for candidate set $\{B, C\}$. Likewise, we can also obtain a similar result by considering A and B as the candidate set. The result of $ETX_{s,a,b}$ is 1.176. However, this calculation is without link correlation. When we consider the link correlation, we could use the following equation:

$$ETX = \frac{1}{1 - Pr(\overline{E_{s,1}}, \overline{E_{s,2}}, \ldots, \overline{E_{s,n}})}$$

where $E_{s,i}$ is the event that the packet is successfully received by node i. Using this example, the links from node S to nodes B and C is 100% correlated. Thus, $Pr(\overline{E_{s,B}}, \overline{E_{s,C}})$ is 0.5. The ETX reduces to 2. In this situation, the selection of A and B is the best choice.

3.1.4 Quality of forwarding (QoF)

In WSNs, some routing protocols are designed by allowing the retransmission strategy, as shown in Figure 3b. The *quality of forwarding* (QoF) [19] was proposed to measure the path quality. The authors considered two kinds of PDR: one is for physical links and the other is for virtual links (inside the node). p is denoted as the probability that the packets successfully go through the link. r is denoted as the most retransmission times. Thus, the *packet delivery ratio* (PDR) over a link is:

$$PDR = 1 - (1 - p)^{r+1}. \tag{2}$$

According to Eq. 2, the *expected transmission count* (ETC) was proposed using the following equation:

$$\begin{aligned} ETC &= \Big(\sum_{k=1}^{r+1} kp(1-p)^{k-1}\Big) + (r+1)(1-p)^{r+1} \\ &= \frac{1 - (1-p)^{r+1}}{p} \end{aligned} . \tag{3}$$

Here, $\sum_{k=1}^{r+1} kp(1-p)^{k-1}$ represents the expected transmission times that the packet passes the link. $(r+1)(1-p)^{r+1}$ represents the expected transmission times, where the packet will fail to to pass the link. Using Eqs. 2 and 3, the QoF is the ratio of the data delivery ratio to the actual transmission times:

$$QoF = \frac{PDR}{ETC}.$$

Thus, for a physical link, the $QoF = p$. Note that ETC is different from ETX in that it not only considers link quality, but also retransmission limit. When $r \to \infty$, ETX=ETC. To calculate the QoF of a path, the PDR of a node has also been integrated into the routing design. The path QoF considers both data delivery ratio and transmission cost. If the data delivery ratios of two paths are the same, QoF selects the path with lower transmission count. If the transmission count of two paths are the same, QoF favors the path with high data delivery ratio.

3.2 Delay

In this subsection, we will focus on the routing issues that relate to the *end-to-end* (E2E) delay. The E2E delay is one of the most fundamental issues for WSNs. Many applications in WSNs require an E2E delay guarantee for time sensitive data. For example, telemonitoring of human health status, vehicle anti-theft [20], and target tracking [4] are classified as the time-sensitive applications.

In low duty-cycle WSNs, due to the limitation of the energy budget, the sensors are scheduled to "sleep" or "active" states. When the sensors are in sleep mode, they cannot transmit the packets. The time spent on waiting for its neighbors to wake-up is called "sleeping latency". Thus, unlike the traditional wireless networks, the delay-based routing design also includes the sleeping latency, in addition to the transmission delay. The sleeping latency (in seconds), however, is much longer than the transmission delay and propagation delay (in milliseconds). Therefore, the E2E delays mainly dominate the sleeping latency.

In traditional wireless networks, the shortest path algorithm is used to find the optimal path in the weighted graph $G = (V, E)$. However, in the low duty-cycle WSNs, the graph changes over time, which is called a "time-dependant graph". $G = (V, E(t))$ [21] is used to represent the models, where V is a set of nodes, and $E(t)$ is a set of edges that appears at time t. Several works have been conducted using time-dependant graph models.

3.2.1 Sleep scheduling

In [22], the authors provided delay-efficient sleep scheduling for WSNs. They consider the case of a single wake up schedule, where each sensor can choose exactly one of k slots to wake up. They also prove that minimizing the E2E is, in general, NP-hard. The interesting part is the time slot assignment for each node. The time slot assignment is to assign a slot to a certain node, and schedule the node to wake up. Let ω be a slot assignment function and $d_{i,j}$ be the delay in transmitting the data from i to j. The delay on a path P

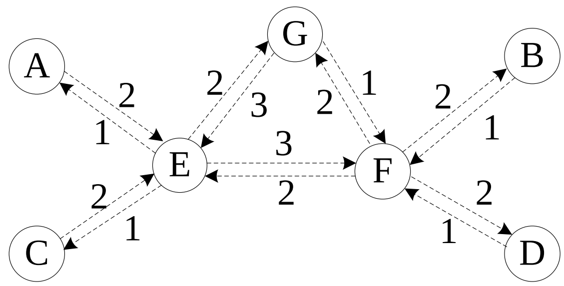

Fig. 8 An example of delay diameter.

under the slot assignment function ω is defined in Eq. 4:

$$d(P) = \sum_{(i,j)\in P} d_{i,j}. \tag{4}$$

They use two models: *all-to-all communication* and *weighted communication*. In the all-to-all communication, the delay diameter is defined as $D_{i,j}$, which is the shortest delay path between nodes i and j under the slot assignment function ω. The problem is to find an assignment function ω that minimizes the delay diameter. This is called *delay-efficient sleep scheduling* (DESS). Figure 8 shows an example of the delay diameter. Among all of the pairs: $(A,B),(A,C),(A,D),\cdots,(B,A),(B,C),\cdots$, the delay diameter $D_{A,B} = d_{A,E} + d_{E,F} + d_{F,B}$ is 7. The property is non-symmetric, since $d_{A,B} \neq d_{B,A}$. Compared to the delay parameter, the hop diameter is symmetric. For example, $h_{A,B} = h_{B,A} = 3$.

In the weighted communication model, they defined the *average delay diameter*, which is $\sum_{i,j\in V} w_{i,j} \cdot d_{i,j}$. Like DESS, they also offer *average delay efficient sleep scheduling* (ADESS). ADESS is used to find the slot assignment function ω that minimizes the average delay diameter. The main difference is that this method focuses on the fact that the communication between some pairs occurs more frequently than other pairs.

3.2.2 Pipeline scheduling

In order to reduce unnecessary forwarding interruption, a state-of the-art solution has been provided by using the technique of pipeline scheduling. The most recent work is presented in [23]. Cao et al. proposed a *Robust Multi-pipeline Scheduling* (RMS) algorithm to coordinate multiple parallel pipelines. Pipeline scheduling-based routing is one of the multipath routing designs. Multi-path routing is the routing technique of using multiple alternative paths through a network. The path could be node disjoint, edge disjoint,

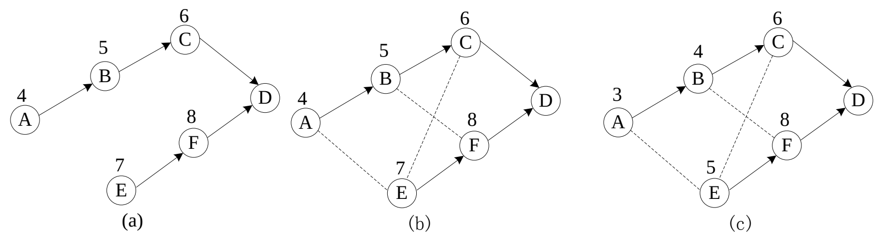

Fig. 9 An example of pipeline scheduling.

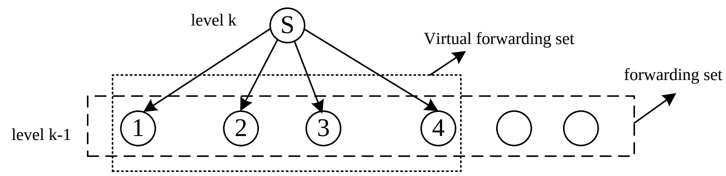

Fig. 10 An example of virtual forwarding set.

or overlapped. The specialty of pipeline scheduling is to lower the switching delay by coordinating the transmission time.

The advantage of the pipeline scheme is the decrease in the sleep latency. The single pipeline scheme is always fragile, due to unreliable links. Therefore, multiple pipelines are provided to dynamically switch one forwarder to another forwarder.

Examples are provided in Figure 9. We assume that the duty cycle is 100 seconds. In the original scheduling method, if the transmission from nodes A to B fails, the packet has to wait a longer time to be retransmitted, as shown in Figure 9a. However, if we use multiple pipelines, we can see that the packet can dynamically switch to another forwarder E. Figures 9b and 9c show the process. Note that if we reschedule the transmission time using pipeline, such that if one pipeline has failed, the node can dynamically switch to another one. In this way, the latency can be reduced, while using the same energy cost. In this scheme, the route is decided dynamically by the timely results. There are three steps for the algorithm.

The first is selection of the virtual forwarding set. The virtual forwarding set is constructed using the link quality. Suppose that the link quality is $\{q_{s1}, q_{s2}, \ldots, q_{sn}\}$. The ϕ is defined as the threshold, which is the one-hop delivery ratio. As shown in Figure 10, the first k attempts are $1-(1-q_{s1})(1-q_{s2})\cdots(1-q_{sk})$. Suppose that we have M links in the forwarding set, the following Eq. 16 needs to be satisfied:

$$1-\prod_{i=1}^{M}(1-q_{si}) < \phi. \tag{5}$$

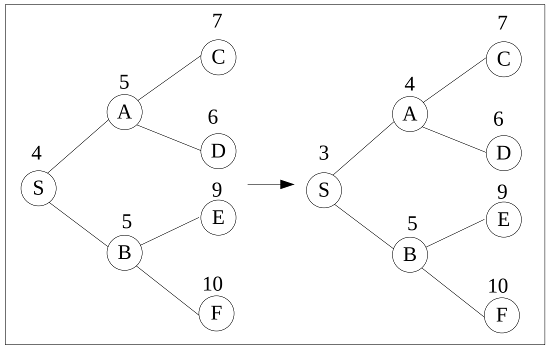

Fig. 11 An example of simultaneous wakeup time.

The second step is propagative scheduling. The basic purpose is to let each node decide its wakeup schedule, and minimize the expected delay of two consecutive levels. For example, node A has two parents, whose wakeup time is $\{t_1\}$ and $\{t_2\}$. Then, the candidate time of node A is $\{t_1 - 1\}$ and $\{t_2 - 1\}$. Therefore, we have modular delay as defined in Eq. 6:

$$|t_1 - t_2|_T = \begin{cases} t_1 - t_2, \ t_1 \geq t_2 \\ t_1 + T - t_2, \ t_1 < t_2. \end{cases} \tag{6}$$

The third step is to avoid the simultaneous wakeup time. Suppose that node A has two parents who wakeup at the same time slot 4. Therefore, node A can either choose nodes B or C to forward the packet. To avoid this, we can shift the wakeup time of node B to time slot 3. Figure 11 shows an example of this process.

3.2.3 Collaborative Scheduling

The collaborative scheduling was studied in [24]. The authors provided a collaborative scheme based on the concept of error interference. They designed a sensing probability bound to control tolerable sensing errors. The proposed scheme aims at achieving low energy cost. The error interference is defined as the difference between the ground truth environmental data and corresponding values generated by the predictor of sensor nodes. There are four stages in the proposed algorithms.

The first step is to detect the neighbors. In this stage, each node recognizes its neighboring nodes, and assigns a table for each neighbor to build the weight graph. The neighborhood formation is a dynamic stage, which will be refreshed after a defined period.

The second step is to generate the node-pair weighted graphs. Specially, the following approach to calculate data correlation $C(i, j)$ between two observation vectors by node N_i and node N_j:

$$C(i,j) = \frac{m\sum o_k^i o_k^j - \sum o_k^i o_k^j}{\sqrt{m\sum (o_k^i)^2 - \sum (o_k^i)^2}\sqrt{m\sum (o_k^j)^2 - \sum (o_k^j)^2}} \tag{7}$$

where $\{o_1^i, o_2^i, \ldots, o_k^i\}$ is the observation vector, obtained through discrete sampling.

The third one is to use the error bound to control the neighbors. The sensing node, can infer the prediction errors of correlated neighboring nodes by comparing its real-time sensing values with corresponding predicted values. Using the probability density function $\rho(x)$, each node i can evaluate the cumulative distribution function $PMF_i(e_i^m)$:

$$PMF_i(e_i^m) = \int_{-e_i^m}^{e_i^m} \rho(x)dx \tag{8}$$

where e_i^m is observation error, based on the difference between actual sensing data and prediction values that are generated by our prediction model. Then, the inferred error between nodes i and j can be expressed as follows:

$$e_{ij} = PMF_j^{-1}(PMF_i(t[k])). \tag{9}$$

Here, $t[k]$ is the variable for each step:

$$t[k] = \begin{cases} 2 \cdot t[k-1] & PMF_j(t[k-1]) < PMF_i(e_i^m) \\ \dfrac{t[k-1] + t[k-2]}{2} & PMF_j(t[k-1]) > PMF_i(e_i^m) \end{cases} \tag{10}$$

where $PMF^1()$ is the inverse function of PMF, and $t[0] = 0$, $t[1] = e_i^m$. It is an iteration process. It will not stop until $PMF_j(t[k1]) = PMF_i(e_i^m)$. The basic process is that each time we will compare $PMF_j(t[k-1])$ and $PMF_i(e_i^m)$. Then, we can decide the next $t[k]$ until $PMF_j(t[k1]) = PMF_i(e_i^m)$.

The fourth step is to determine whether the sensors switch on/off. The node can remain turned off if the inferred error is smaller than the error tolerance. The error tolerance is a specified bound (e.g. the tolerance threshold of errors).

3.2.4 Expected end-to-end delay

In [21], $\Gamma_i = (\omega_i, \tau)$ is used to represent the working schedule of node i. ω_i is an infinite binary string in which 1 denotes the active state, and 0 denotes the dormant state. τ denotes the time span for each bit. For example, the

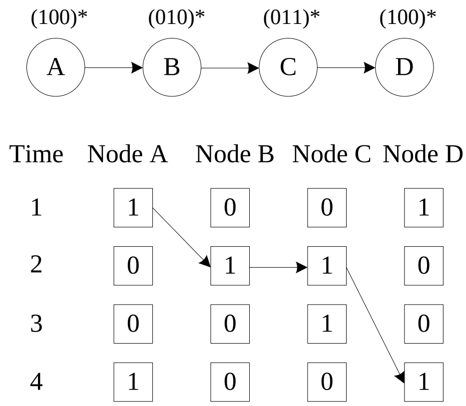

Fig. 12 Time-Expanded Network.

total time-span of the binary string (00101) with $\tau = 2$ seconds is 10 seconds, since there are totally five bits in the string. Thus, for a sender i and receiver j, if the working schedules are $\Gamma_i = ((10000)^*, \tau)$ and $\Gamma_j = ((00010)^*, \tau)$, respectively, the E2E should be $3\tau - 0\tau = 3\tau$. The main contribution is that they provide an E2E delay guarantee by adding extra active bits to nodes. For example, for the single link route $A \to B$, $\Gamma_A = ((010)^*, \tau)$ and $\Gamma_B = ((100)^*, \tau)$. By adding an extra active bit to node B, and by changing its working schedule from $(100)^*$ to $(101)^*$, the sleep latency can be reduced to τ.

The expected E2E was mentioned in [14]. It is formally defined as the expected data delivery delay from source node S to destination D over a multi-hop route. They proposed a time-expanded graph model to represent the low duty-cycle WSNs. In the ideal case, the E2E delay in this network is equal to $H \cdot \tau$, where H is the minimum number of hops between a source and destination. Figure 12 shows the process of the delivery from node A to node D. Since node B is in sleep mode, the packet can only be delivered at time 2. The transmission procedure is the same as for the following node. Thus, the E2E delay is 4, where node A sends the packet at time 1, and node D can receive it at time 4. The main concept of this paper is the forwarding sequence, which is maintained by each node. The forwarding sequences are constructed by the nodes' neighbors. During the transmission, the sink node will check the time associated with the first node in the sequence and forward the packet. If the transmission is successful, the forwarding is done. Otherwise, the node may check the second node in the sequence and forward the packet. If P'_i is the probability that the packet arrives at the ith forwarder, under this scenario, the routing metric of *expected E2E delay* (EED) is:

$$EED = \sum_{i}^{n} P'_i (EED_i + d_i) \tag{11}$$

where d_i is the waiting delay at node i. $P'_i = P_i \cdot EDR_i / EDR$ can be computed according to EDR and P_i, which is discussed in the subsection of packet delivery ratio. We use Figure 5 as an example to compute EED. From Figure 5, we have $d_A = 2$, $d_B = 4$, and $d_C = 6$. From the previous section, we can get $P'_A = 0.6 \cdot 0.7/0.74 = 0.57$, $P'_B = 0.28 \cdot 0.9/0.74 = 0.34$, and $P'_C = 0.096 \cdot 0.8/0.74 = 0.1$. According to Eq. 11, the EED of node S is $0.57 \cdot (2+1) + 0.34 \cdot (4+2) + 0.1 \cdot (6+1) = 4.45$.

3.2.5 End-to-end delay in the random walk model

The E2E delay was also studied in random walk models. In the *i.i.d.* (independent and identically distributed) random duty-cycling model [25], ρ is the duty cycle rate of each node. Each node is in the active state with a probability ρ, while it is dominated with $1-\rho$. For the link (i, j), the probability of nodes i and j being in the wake mode is ρ^2. It is shown that the per-hop latency is $d_i = \frac{1}{1-(1-\rho^2)^{n_i}}$, where n_i is the number of neighbors of node i. Note that, in this case, the working schedule is previously unknown. In the pseudo-random duty-cycle model [26], the node has the knowledge of the working schedule. Therefore, the per-hop latency is $d'_i = \frac{1}{1-(1-\rho)^{n_i}}$.

In [26], the authors studied several aspects of latency in random walk models: (1) Hitting time: the expected time for source s to hit the destination d. (2) Commute time: the expected round-trip time between source s and destination d. (3) Cover time: the expected time from the source s to all of the other nodes in the network. Under the stochastic routing framework, the authors in [27] studied the routing problem using the Markov chain. They developed centralized and distributed implementations in low duty-cycle WSNs.

3.2.6 Communication delay in low duty-cycle sensor networks

In [28], the authors introduced a novel solution of communication delay in low duty-cycle sensor networks. They provide sink-to-one and sink-to-many solutions, and their distributed implementation.

The network topology is denoted as $G(t) = (V, E(t))$, where V is a set of nodes and $E(t)$ is a set of directed links at time t. Each edge $e_{ij}(t)$ belongs to $E(t)$ if node i and node j are in each other's communication ranges. Node j is in active mode, so as to receive the packets at time t. $d_{ij}(t)$ is defined as the delay when node i sends the packet at time t, when node j is in active mode to receive the packet.

Figure 13 shows an example of the proposed solution. In this example. the original E2E delay from node s to node b is 6. After argumentation of active mode at time 4 for node b, the E2E delay can be reduced to 3. In addition, the original schedule of node b is time 7. After argumentation, the schedule

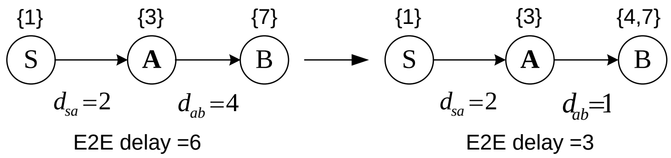

Fig. 13 Example of active argumentation.

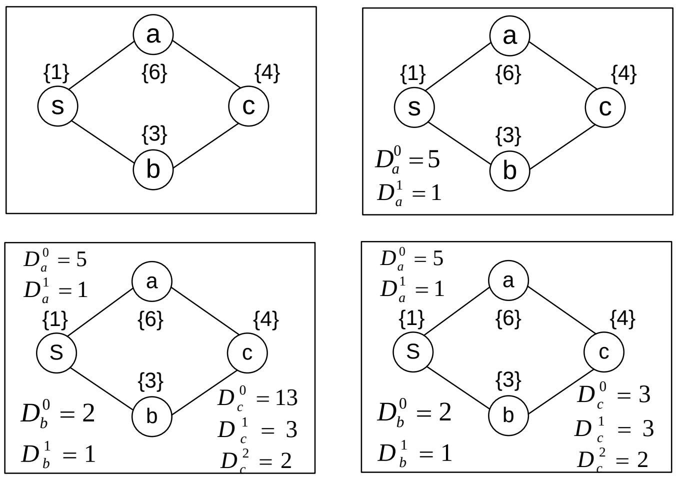

Fig. 14 Example of delay computation.

of node b is both time 4 and time 7. In the following, we will provide more details of the solution. The first is how to find the minimum delay for active instance augmentation. We define D_j^h as the minimal delay from the source to node j, with at most h active augmentation. The initial state is

$$D_j^h = \begin{cases} d_{sj}, \ h = 0 \\ 1, \ h = 1. \end{cases}$$

Based on this solution, we then offer the recursive solution. The main idea is that we could use intermediate node p to help the delivery. Then, we have

$$D_j^h = \min \begin{cases} D_j^h, \\ D_p^{h-1} + 1, \ h > 0 \\ D_p^h + d_{pj}(t). \end{cases}$$

An example can be presented in Figure 14. In this example, the initial states are $D_a^0 = 5$, $D_a^1 = 1$, $D_b^0 = 2$, and $D_b^1 = 1$. Then, for node c, we have $D_a^0 + d_{ac}(6) = 5 + 8 = 13$. Using the above equation, we have:

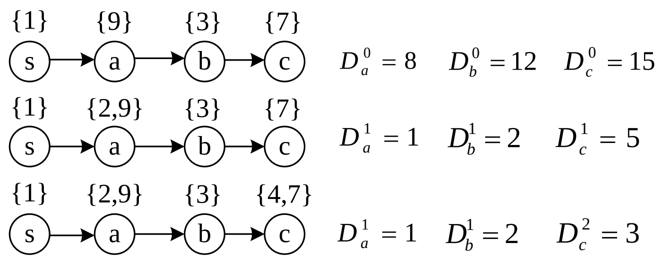

Fig. 15 Example of linear topology.

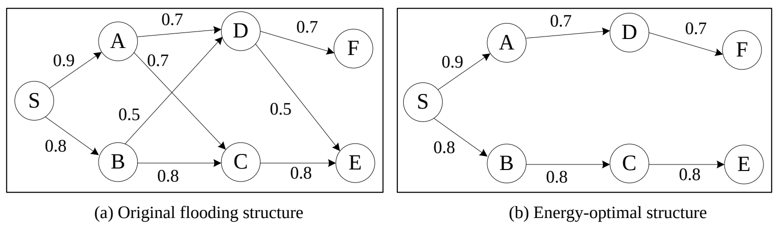

Fig. 16 DAG-based flooding structure.

$$D_c^1 = \min \begin{cases} D_a^0 + 1 = 5 + 1 = 6 \\ D_a^1 + d_{ac}(2) = 1 + 2 = 3 \end{cases} = 3.$$

Correspondingly, we have $D_c^2 = 1 + 1 = 2$. We also offer the example of linear topology shown in Figure 15. In this example, the initial states are $D_a^0 = 8$, $D_b^0 = 12$, and $D_c^0 = 15$. After the first round of argumentation, we have $D_a^1 = 1$. Then, D_b^1 is 2. $D_c^1 = 5$. The algorithms stops at the second argumentation, where we set $D_c^2 = 3$.

3.2.7 Opportunistic flooding and expected packet delay

Flooding has been investigated extensively in wireless networks. However, there are several challenges when using low duty-cycle WSNs. Firstly, the nodes stay asleep most of time and wake up asynchronously. A broadcasting packet is rarely received by multiple nodes simultaneously. Secondly, the sender may have to wait for a certain period of time until its receiver becomes active. Finally, the wireless link is unreliable. The transmission may be repeated due to the low link quality.

Opportunistic flooding in low duty-cycle WSNs was proposed in [29]. The main objective is to reduce redundant transmissions, while achieving fast dissemination. As shown in Figure 16a, the flooding structure of the network is a *directed acyclic graph* (DAG) of N vertices. Figure 16b offers the solution of the structure, which is called an "energy-optimal tree." The energy optimal

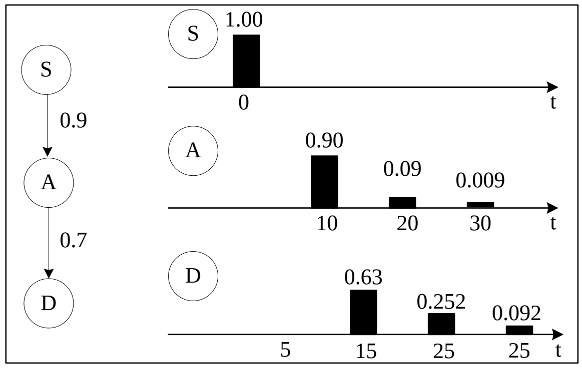

Fig. 17 The *pmf* computation.

tree is built based on a smaller hop count and the best link quality. The proposed opportunistic flooding consists of three parts: the *probability mass function* (pmf), decision making process, and decision conflict resolution.

The pmf is denoted as a set of tuples $\{(t_h(i), p_h(i))\}$, where $p_h(i)$ is the probability of receiving the packet at time $t_h(i)$. For the source node, it will always awaken. Therefore, the delay is 0 and the probability is 100%. The pmf of the source is (0,100%). Then, the probability that it receives the flooding packet at its jth active time is:

$$p_{h+1}(j) = \sum_{i:t_h(i)<t_{h+1}(j)} p_h(i)p(1-p)^{n_{ij}}.$$

Figure 17 shows an example to compute pmf. The probability that node A receives the packet at time 10 is 0.9. At time 20, the probability is $0.9 \cdot (1 - 0.9) = 0.09$. For the node D, the probability is the sum of two cases: (i) node A receives the packet at time 10, or (ii) the probability that node A receives the packet at time 20. Then, pmf is $0.9 \cdot (1 - 0.7) \cdot 0.7 + 0.09 \cdot 0.7 = 0.252$. Similarly, all of the nodes in the network will compute their pmf.

In the decision-making process, the p-quantile delay (D_p) is a threshold delay. A node computes the *expected packet delay* (EPD) and makes a forwarding decision, based on the comparison between the EPD and D_p. The EPD will be introduced in the following section. If we have the transmission from A to B, the EPD can be computed by using the following equation:

$$EPD = \sum_{j:t_{h+1}(j)>t_l(i)} t_{h+1}(j)p(1-p)^{n_{ij}}$$

where p is the link quality. A node with hop count h is denoted as a level-h node. Then, n_{ij} is the level-$(h+1)$ node's active time units between $t_h(i)$ and $t_{h+1}(j)$. EPD is the sum of all the series. A simple way to obtain the EPD is to use the expected transmissions $\lceil \frac{1}{p} \rceil$ and get the time slot of the $\lceil \frac{1}{p} \rceil$th

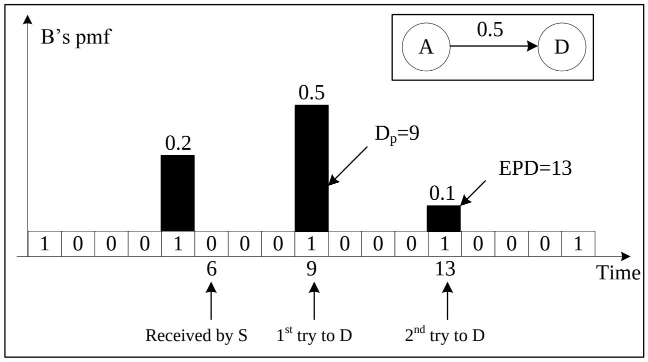

Fig. 18 An example of Decision Making

transmission. For example, as shown in Figure 17, since $p = 0.5$, we have the expected transmission 2. Then, we have $EPD = 13$, as the 2rd try is at time 13. D_p can be computed by using the discrete quartile function:

$$F^{-1}(p) = \min\{x \in R : P_r(t \leq x) \geq p\}.$$

Figure 18 shows an example of the decision making process. Thus, $D_p = F^{-1}(0.2 + 0.5) = 9$. Since EPD=13> $D_p = 9$, the second try is redundant.

The motivation for the decision conflict resolution is the hidden terminal problem. Since links are unreliable, the more nodes in the same set will make it possible for packets to be sent at the same time. It will be likely that a transmission is not sensed by all of the other nodes, leading to a collision. The link quality threshold h_{th} was proposed for the selection process. All of the links have a higher link quality than h_{th}. During the selection process, the selected candidate will follow the order of the link quality. To resolve the conflict in the sender set, the backoff function was designed to select the higher link quality function. This means that the duration of the backoff depends on the link quality between the sender and the receiver. When multiple nodes want to send the packet to the same node, the node with shortest backoff time $d_{backoff}$ can forward the packet first. Suppose that the bound of backoff is $D_{backoff}$ and the maximum size of sender set is n. Each node could compute its own backoff by using the following equation:

$$d_{backoff} = (\lfloor n(1-p) \rfloor) \frac{D_{backoff}}{n} + d_{rand}$$

where d_{rand} is a random time period, and $d_{backoff}$ is the backoff time for each node. The basic idea is that each node computes its own backoff time $d_{backoff}$ and transmit the packets after $d_{backoff}$.

3.3 Energy cost

The energy cost is an important part of WSNs. While developing the routing protocols, it is crucial to ensure the power efficiency. The power consumption can be put into two categories: data transmission and data processing.

3.3.1 Real-time power-aware routing

The *real-time power-aware routing* (RPAR) scheme was proposed in [30]. This routing protocol is based on the tradeoff between transmission power and communication delay. This work focuses on the real-time applications in which meeting deadlines is more important than throughput. The goal is to increase the number of packets that meet deadlines, while minimizing the energy consumption.

The *delivery velocity* was proposed as the distance that a packet travels, divided by its packet delay. The slack is the time remaining until the deadline expires. It can be updated at each hop. With the source s and destination d, the required velocity of a packet is:

$$v_{req}(s,d) = \frac{dis(s,d)}{slack_{rec} - (t_{head} - t_{rec})}$$

where $dis(s,d)$ is the Euclidean distance of s and d. t_{head} is the time when the packet becomes the head of the transmission queue. t_{rec} is the time to receive the packet. The proposed RPAR protocol uses the velocity assignment policy to map a packet's deadline. They also provide a delay estimator for different forwarding choices:

$$d_p = (d_{cont} + d_{tran}) \cdot ETX(p)$$

where $ETX(p)$ is the expected number of transmissions from node s to neighbor i at power p. d_{cont} and d_{tran} are the contention delay and transmission delay, respectively. Based on the velocity policy and delay estimator, RPAR computes the energy cost of all of the eligible choices. Then, it will forward the packet using the most energy-efficient choice:

$$C(s,d) = C(p) \cdot ETX(p) \cdot \frac{dis(s,d)}{dis(s,d) - dis(i,d)}.$$

3.3.2 Expected energy consumption

As shown in Figure 3, we set $|S|$ as the forwarding sequence. The EEC [14] was defined as the energy consumption required to deliver a packet from node s to sink node d. For each node i, EEC_i is the expected energy cost, and c_i

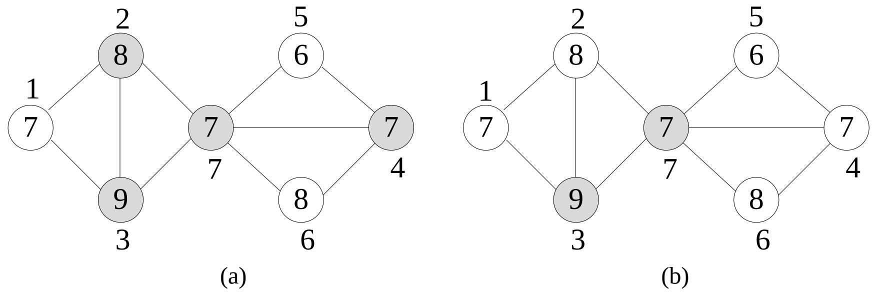

Fig. 19 An example for power-aware broadcasting.

is the transmission cost. If P'_i is the probability that the packet arrives at the ith forwarder, then the EEC should be:

$$EEC = \sum_{i=1}^{n} P'_i \cdot (c_i + EEC_i) \tag{12}$$

where P'_i is the same as was discussed in the subsection about the packet delivery ratio. As an example, shown in Figure 5, since $P'_A = 0.57$, $P'_B = 0.34$, and $P'_C = 0.1$, EEC of node s is $0.57 \cdot (1+2) + 0.34 \cdot (2+2) + 0.1 \cdot (3+1) = 3.47$.

So far, we have introduced the three routing metrics, proposed in [14]: expected delivery radio (EDR), expected E2E delay (EED), and expected energy consumption (EEC). Among all of the metrics, the basic idea is that we can select the *optimal* subsequence from the forwarding sequence, so as to achieve the optimal solution. The optimal sequence is in terms of the maximum EDR, minimum E2E delay, or minimum EEC.

3.3.3 Power-aware broadcasting

As we mentioned in the previous section, a straightforward broadcasting scheme is flooding. However, this approach will result in redundant transmissions and more energy consumption. In [31], Rule 1 and Rule 2 are provided to select gateways based on the node priority ($id(v)$). Suppose that $N(v)$ represents the neighbor set of node v, and $N[v] = N(v) \cup \{v\}$ is a closed neighbor set of v; in Rule 1, if $N[v] \subseteq N[u]$ in G and $id(v) < id(u)$, then unmark v. In Rule 2, if $N(v) \subseteq N(u) \cup N(w)$ in G and $id(v) = \min\{id(v), id(u), id(w)\}$, then unmark v. To prolong the lifetime of the sensor nodes, the authors in [32] proposed the use of connected dominating sets (CDS) to reduce the number of transmissions, as well as conserve energy.

They propose saving energy by only allowing dominating nodes to retransmit the packets. In addition, they also provide the activity scheduling method to dynamically select the dominating nodes. The selected gateway nodes are

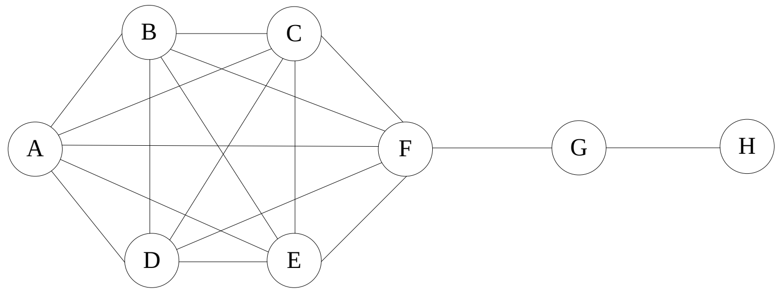

Fig. 20 An example representing the need for smart gossip.

based on marking process [33], where Rule 1 and Rule 2 are based on energy levels. Figure 19 shows an example of the procedure, where each node is assigned with a value inside each node representing energy level. Figure 19a shows the graph with marking process, and Figure 19b presents the results after Rule 1 is based on energy level. That is, energy level is used as the primary priority, and node id is used as the secondary priority when energy levels are the same. The nodes in the pink color are the selected gateways. In this case, node 2 is removed as it is covered by node 3, which has a higher energy level. Nodes 4 and 7 have the same energy level, but node 4 is removed, for it has a smaller id.

3.3.4 Gossip

In WSNs, broadcast services should minimize energy consumption by reducing redundant transmissions. Flooding is considered the simplest method of broadcasting. However, this will lead to collisions and redundant packet receptions.

Smart gossip [7] was proposed to adapt transmission probabilities based on the underlying network topology. To obtain the p_{gossip}, the *average reception percentage* was proposed to measure the reliability, which is the reception percentage averaged over all nodes in the network. To evaluate the overhead, the *average forwarding percentage* was also proposed, which is the forwarding percentage averaged over all nodes in the network. It has been proven that the average forwarding percentage is also the measure of the average energy consumed at a node while transmitting gossip messages.

Figure 20 shows an example that represents the need for smart gossip. If node F can identify that node G depends only on F to receive the gossip, and nodes B, C, D, and E can identify that they are never required to forward the gossip, then we achieve the efficiency. If a node Y has k parents, then it suffices for each parent to use a gossip probability (p_{gossip}), which ensures the probability that at least one of them will transmit is greater than the per-hop reception probability (p_r). This idea can be presented as follows:

$$(1 - p_{gossip})^k < 1 - p_r$$

where $(1 - p_{gossip})^k$ is the probability that all k parents choose not to transmit. This has to be less than $(1 - p_r)$ to meet the application reliability requirement.

4 Composite Utility-based Metric

Although we have sorted the recent works into different categories, routing design may be involved in several factors. These factors may be related to each other. In this subsection, we offer additional discussions of them. We call the utility with multiple purposes a composite utility-based metric.

One of the composite utility-based routing schemes is presented in [34], which is also called "utility-based routing". The concept of utility in this work is different from our previous discussion. This utility-based routing scheme is a special routing approach that is based on the network topology, as well as the importance of the packet to be delivered [35, 34]. The composite utility model here is in terms of the expected benefit (of the routing source successfully forwarding a packet to the destination) minus the expected cost incurred by forwarding nodes. Unlike wired networks, wireless connections are unreliable, due to interference and coverage issues. With the utility-based routing metric, the more valuable packet will be delivered through a more reliable route at the expense of a higher transmission energy cost[34]; This is a common phenomenon in wireless communication. Utility-based routing is used to reflect the trade-off between a highly reliable route (which is usually more costly) and a less reliable route (which is usually less costly) based on the value of the packet. A simple analogy that relates to utility-based routing is the postal service: a high-value package (e.g., one that contains a passport for a visa application) usually uses registered mail for reliability at a higher premium cost. An ordinary package is usually mailed through a regular service.

There are two aspects to be considered in the routing design of wireless networks. Firstly, unlike the wired network, wireless connections are unreliable due to the interference or coverage issues. Another problem that cannot be neglected is the cost of transmission, due to limited energy supplies. In [34], the authors have the following assumptions. The reliability is the packet-error-rate associated with the link (i, j). The cost is the energy cost for transmitting a packet from sender i to receiver j. This consists of the transmission power, alone. In the following, we will offer the overview of several works related to utility-based routing. We will start with ad-hoc networks. Then, we will focus on multi-hop networks and opportunistic routing. At last, we will discuss the utility-based approach in low duty-cycle WSNs.

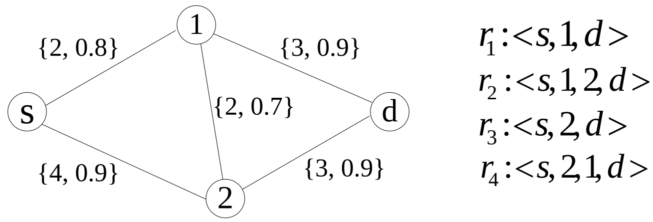

Fig. 21 An example of utility-based routing.

Table 2 Utility in a simple wireless network

	u_d	u_1	u_2	U
r_1	20/30	15/24		10/17.2
r_2	20/30	8.5/14.8	15/24	4.8/9.8
r_3	20/30		15/24	9.5/17.6
r_4	20/30	15/24	8.5/14.8	3.7/9.3

4.1 Composite utility-based routing in ad-hoc networks

The utility-based routing in ad-hoc networks was proposed in [36]. The basic idea of utility can be presented as follows. To illustrate the model, we use a single link route from s to d. The source node s wants to transmit a packet to the destination d. If this transmission is successful, we will offer a benefit v for the route. During this transmission, each link will incur a cost. The expected utility value can be derived from the benefit and expected cost. The idea can be presented in the following:

$$u_{s,d} = p_{s,d} \cdot (v - c_{s,d}) + (1 - p_{s,d}) \cdot (0 - c_{s,d}) = p_{s,d}v - c_{s,d} \qquad (13)$$

where $u_{s,d}$ is the utility for the transmission. $p_{s,d}$ and $c_{s,d}$ are the probability and cost of link (s, d), respectively. Thus, for a path $< 1, 2, \cdots, n >$, the expected utility U is:

$$U = (\prod_{i=1}^{n-1} p_{i,i+1}) \cdot v - \sum_{i=1}^{n-1} c_{i,i+1} \prod_{j=1}^{i-1} p_{j,j+1} = P_R \cdot v - C_R \qquad (14)$$

where P_R is the path reliability, and C_R is the path cost. When comparing Eq. 13 with Eq. 14, the expected utility has a similar form: stability times benefit minus cost. The stability P_R is the multiplication of reliability for each node on the path. The expected cost of each hop is dependant on the successful delivery of previous transmissions. For example, as shown in Figure 21, there are four routes. For r_1, we have $U = 0.8 \cdot 0.9 \cdot 20 - 2 - 3 \cdot 0.8 = 10$. In a backward manner, we can view node 1 as the virtual source. By applying

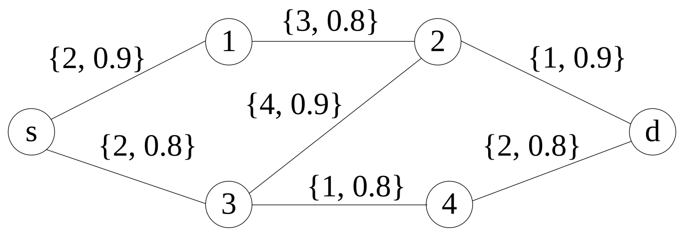

Fig. 22 An example of utility-based routing in multi-hop networks.

Eq. 13, we have $u_1 = 0.9 \cdot 20 - 3 = 15$. For the link $(s, 1)$, nodes s and 1 can be viewed as the source and destination. The utility U is $15 \cdot 0.8 - 2 = 10$. From the results, we know that the route with maximum utility is not the same when considering different benefits for the delivery. When comparing routes r_1 and r_3, r_1 is better than r_3 when we offer the benefit 20. However, route r_3 is the best if the benefit is 30. This means that the packets with more benefit will select a more reliable route, despite the cost being higher.

4.2 Composite utility-based routing using opportunistic routing

Utility-based routing in multi-hop networks mainly focuses on the *opportunistic routing* (OR). The opportunistic routing was proposed in [6]. The main features of OR routing is the rule of the selection of a relay set for each node, and the rule regarding relay prioritization. OR routing would achieve higher throughput, since each of the source's transmissions is likely to be received by at least one relay.

In [35], the authors explored the optimality of utility-based routing through OR without allowing retransmissions. By integrating the idea proposed in [36], OR routing with the utility-based model can be presented as follows:

$$opu_i = \sum_{j=i+1}^{i+k} (opu_j \cdot p_{i,j} \cdot \prod_{l=i+1}^{j-1} (1 - p_{i,l})) - c \tag{15}$$

where opu_i denotes the node i's *residual expected network utilities* (RENU). Note that the relays are prioritized in order from $i+1$ to $i+k$, with $i+1$ as the highest priority.

Figure 22 shows an example of multi-hop networks. The calculation starts from the node d. Suppose that the benefit is 20. For the node 2, the utility $u_2 = p_{2,d} \cdot v - c_{2,d} = 17$. The procedure is the same as for node 4. Thus, $u_4 = 14$. However, node 3 has two receivers. By using Eq. 15, we have $u_4 = 17 \cdot 0.9 + 14 \cdot 0.8 \cdot (1 - 0.9) - 4 = 11.4$. At last, we can get the utility $opu = 10.45$.

Table 3 Utility in a simple wireless network.

	u_d	u_1	u_2	u_3	u_4	u_s
u	20	10.6	17	11.3	14	9.54
opu	20	10.6	17	11.4	14	10.45

4.3 Composite utility-based routing in low duty-cycle WSNs

In low duty-cycle WSNs, the sensors are scheduled to be either in active or sleep mode to achieve low-energy consumption [21, 11]. When the sensors are in sleep mode, they cannot transmit the packets. The utility-based routing in low duty-cycle WSNs is more challenging when we consider the periodic delay of the active and sleep schedule on each node. In addition to the cost and reliability, the delay and deadline are also important for the routing design in low duty-cycle WSNs.

Although energy can be saved by putting the sensor into sleep mode, the cost still exists, due to the set-up and tear-down operations during the transition from active to sleep mode, and vice versa [37]. In fact, energy efficiency is highly related to time efficiency. It can be assumed that time and energy efficiency are equal during the normal operation mode. The time efficiency can be obtained by the ratio of the time spent on the data transmission over the total time for the ideal working schedule. In the randomized duty-cycle scheme, the Markov chain method was used for the analysis [37]. This provides the result of the expected time efficiency. The lower bound and upper bound on it are also provided. The intuitive idea is that the duty-cycle will cause additional delay, but the energy can be saved by putting the nodes into sleep mode.

From our discussion, we know that PDR is highly dependant on reliability. In the definition of PDR, it is measured by a given time period. Then, PDR and delay have an inverse relationship with each other. Higher PDR means a lower delay, and vise versa. As we have discussed in the previous section, the time efficiency and energy efficiency are equal. Thus, it can be derived that the energy cost is also lower. In other words, if the reliability is lower, additional delay and cost will be incurred. In this subsection, we will introduce the timed-based method. Since low duty-cycle WSNs are time-dependent networks, we use the concept of “journey” to represent a path. We set $J = (R, T)$ to be a journey where $R = \{0(= s), 1, \cdots, n(= d)\}$ and $T = \{t_0, \cdots, t_n\}$, such that $t_i \leq t_{i+1}\ \forall i \in \{0, \cdots, n-1\}$. Here, t_i is the contact time at node i. In general, there are several possible journeys for a given path. Unlike single-utility, multiple factors can be jointly involved in the routing metric design.

We first provide the timed benefit function. This is used in the timed-utility model. In our timed-utility model, we measure the packet value, cost,

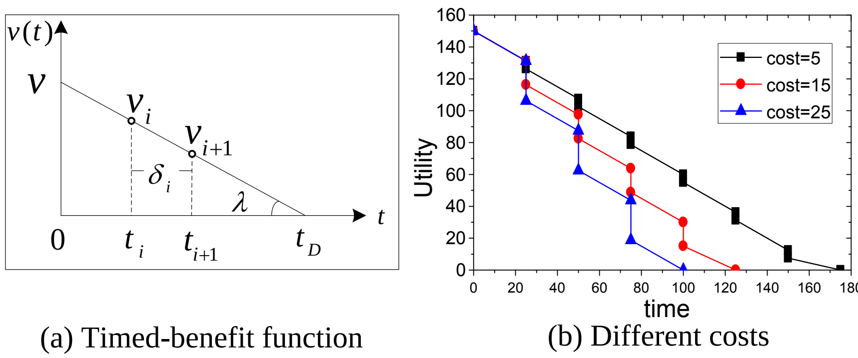

Fig. 23 An example of timed-utility.

and delay to make forwarding decisions. The timed-benefit function is to integrate the delay and deadline into the composite utility.

We use a single link route from s to d as an example. $v(t)$ is defined as the function of TB where the benefit linearly decreases over time, as shown in Figure 23. t is denoted as the contact time slot for the transmission. The deadline t_D is set as the timeliness of a packet when the benefit is reduced to zero. Suppose that c is the cost for the transmission; the utility u is $u = v(t) - c$. We use $\lambda = v/t_D$ as the slope. Then, $u = v - \lambda \cdot t - c$. Correspondingly, for a journey J, the utility u_i for each node i could be presented as follows:

$$u_{i+1} = u_i - \lambda \cdot \delta_i - c_{i,i+1} \tag{16}$$

where $c_{i,i+1}$ is the communication cost of link $(i, i+1)$ and δ_i is the delay of link $(i, i+1)$. Figure 23b shows the results with different costs. There are two types of decay. The first one is related with $v(t)$, and the other one is for the communication cost. According to Eq. 14, for a journey J, we have the expected utility:

$$U = (\prod_{i=0}^{n-1} p_{i,i+1}) \cdot (v - \lambda \cdot (\sum_{i=0}^{n-1} \delta_i)) - \sum_{i=0}^{n-1} c_{i,i+1} \prod_{j=0}^{i-1} p_{j,j+1} \tag{17}$$

By applying the result, the forward solution is forwarded for each hop. It has been divided into two parts:

$$v'_{i+1} = p_{i,i+1} \cdot (v'_i - \lambda \cdot (\prod_{j=0}^{i-1} p_{j,j+1}) \cdot \delta_i). \tag{18}$$

Further, we define a notation u_i as the expected utility. The expected utility value is where node i is treated as the virtual destination. Then, we have:

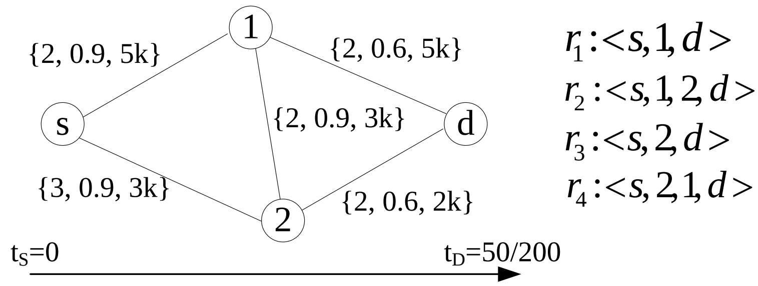

Fig. 24 An example for the TU model.

$$u_{i+1} = v'_{i+1} - \sum_{j=0}^{i} (\prod_{k=0}^{j-1} p_{k,k+1}) c_{j,j+1}. \tag{19}$$

To explain our function, we use $r_1 :< s, 1, d >$ with $t_D = 50$ and $v = 50$. As shown in Figure 24, we have $\delta_s = 5$, $\delta_1 = 5$ and $\lambda = 1$. According to Eq. 17, we obtain the utility $U_{r_1} = 0.9 \cdot 0.6 \cdot (50 - 10) - 2 - 2 \cdot 0.9 = 17.8$. Alternatively, we can also use the step-by-step formula, i.e., Eq. 18 and 19, to obtain the utility. Since we have $t_D = 50$ and $u_s = 50$, then the expected remaining benefit of node 1 is $v'_1 = p_{s,1} \cdot (v - \lambda \cdot \delta_s) = 0.9 \cdot (50 - 1 \cdot 5) = 40.5$, and the expected utility is $u_1 = v'_1 - c_{s,1} = 40.5 - 2 = 38.5$. Further, the expected remaining benefit of node d is $v'_d = p_{1,d} \cdot (v'_1 - \lambda \cdot p_{s,1} \cdot \delta_1) = 0.6 \cdot (40.5 - 0.9 \cdot 5) = 21.6$. Then, we can get the expected utility of d, $u_d = v'_d - c_{s,1} - c_{1,d} \cdot p_{1,d} = 21.6 - 1.8 - 2 = 17.8$.

4.4 Composite utility-based broadcast in low duty-cycle WSNs

The composite utility-based broadcast in low duty-cycle WSNs was proposed in [38]. The composite utility refers to reliability, cost, and delay. Given the node s to broadcast a message starting from time t_0, the forwarding schedule can be represented as:

$$S = (1, t_1), (2, t_2), \ldots, (i, t_i), \ldots, (m, t_m)$$

where $(t_0 \le t_1 \le \cdots \le t_m)$. (i, t_i) denotes the i-th forwarding, where node i forwards the packet at time t_i. We set S_i as the set of nodes that receives the broadcast message in the i-th forwarding. Then, $| \bigcup_{i=0}^{m} S_i |= n$, where n is the number of nodes. The function $f(|S|, t_m - t_0)$ is proposed to deal with the trade-off between the total message forwarding ($|S|$) and the total latency ($t_m - t_0$). This paper focuses on a common linear combination, $f(|S|, t_m - t_0) = \alpha|S| + \beta(t_m - t_0)$. This function covers the demands for different applications. For example, if the broadcast message is about an emergency event and of small size, a small α with a large β will ensure that the message is quickly delivered to the whole network, though possibly with higher forwarding costs. On the other hand, if it is not an emergency message, a large α with a small β will work well to save forwarding costs.

Recent work in [39] deals with energy and delay constrained for WSNs. They provide a solution by integrating the reliability, cost, and delay constraints. The main idea is that they use the multi path scheme to solve the problem.

5 Additional discussions

5.1 The other utility models

In this part, we offer the *expected retransmissions* (ExpR). From the aspect of reliability, we know that many applications need an end-to-end delivery guarantee. The following equation is used to obtain the expected retransmissions of link $(i, i+1)$:

$$r_i = 1/p_i.$$

Besides this, we also offer the *delay ratio* (DelayR) based method, which is:

$$r_i = \delta_{i-1}/\delta_i$$

where $1 \leq r \leq 1/p$. In DelayR, we allow partial retransmissions without extra delay. Then, we have provided three extensions: MaxU, ExpR, and DelayR.

5.2 The timed-benefit with different indices

In this part, we will offer the extension with different indices. The proposed benefit function will linearly decrease as the time increases. However, in other cases, the delay distribution may vary upon different cases. In the following, we will provide the extension of timed-benefit function. In the extension of timed-benefit function, the benefit is also zero when t arrives at deadline t_D. Thus, we have:

$$v(t) = v \cdot (1 - t/t_D)^k$$

where k is the index. It can be varied according to different cases. Figure 25 shows an example of timed-benefit function with different k. Since $0 < 1 - t/t_D \leq 1$, it is a convex function when $k > 1$. It follows the concave function if $0 < k < 1$.

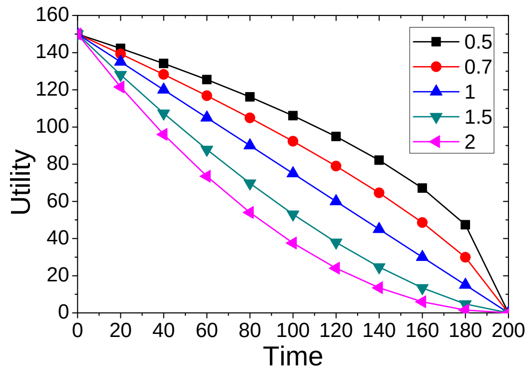

Fig. 25 The extension of timed-benefit function.

5.3 Comparisons

So far, we have introduced single utility-based routing and composite utility-based routing. The two kinds of utility-based routing are designed to achieve different objectives. Therefore, the comparisons among them become very interesting. Here, we settle them into several parts.

(1) Single utility vs Composite utility: A naive way is to analyze the joint performance (throughput, packet loss rate) or single performance (delay, cost, packet delivery ratio). For example, the paths selected from single utility, such as minimum E2E cost, minimum E2E delay, or maximum E2E reliability, can be compared with the paths chosen from composite utility.

(2) Utility under space view or time domain: As we have mentioned in previous chapters, node deployment serves an important role in the selection of a path. Several nodes which are close to each other can construct a group, called a community. Different communities have their own utility value. The problem is how to build communities and compare them. Moreover, under different time domains, the variety of the communities is another challenge.

(3) Single copy vs Multiple copies: The utility models we have mentioned can be extended using multiple copies. The utility value is updated, since multiple nodes hold the packets. The E2E cost, E2E delay, and E2E reliability needed to be reconsidered. The composite utility model can be renewed, as well.

5.4 Future work

In this part, we will discuss future work related to routing in WSNs. Currently, lots of research is being conducted in the design of routing protocols. We provide Table 4 for the classification of recent works. As shown in Table

4, works are seldom designed for multicast transmission. Furthermore, the speciality of low duty-cycle WSNs requires a more feasible routing design. Thus, we provide the future works in several directions.

A promising direction lies in multicast routing. Unlike unicast and broadcast, multicast routing facilitates the packets to be forwarded to a group of destinations. In this situation, there are several paths for the different destinations. Reducing the redundant packets and selecting a feasible path becomes more challenging; especially in low duty-cycle WSNs, where many applications require the packets to be delivered before the deadline. This means that the end-to-end delay is very important for the selection of the route.

As we have mentioned before, the utility-based routing approach is designed for a single pair that consists of a source and a destination. In the unicast, we offer the reward if the packet arrives at the destination and satisfies certain requirements. This work can be extended by the consideration of several destinations, or all other nodes. A simple idea is to offer a reward when all of the destinations receive the packet. A more complicated way is that the reward could be divided by the destinations. Moreover, the optimal solution was proposed using a backward method. This method might not be suitable for multiple destinations. However, we still hope to find a feasible solution for this problem. The extensions could also be retransmission strategies. As shown in Figure 3, there are two methods used upon the failure of a transmission. For the second method, the number of retransmissions can be investigated by achieving some goals: delay, cost, reliability, or a combination of them.

6 Conclusion

In this chapter, we studied the routing issues in WSNs. We covered several topics in the single utility design for different kinds of routing protocols. We discussed utility-based routing, which considers the value of packets. Composition-based methods were also introduced. For example, delay and cost can jointly be considered in the routing design. In the future, we believe that new routing protocols can be provided for mobility control, as well as other factors involved in our discussion, for different kinds of applications.

References

1. R. Murty, A. Gosain, M. Tierney, A. Brody, A. Fahad, J. Bers, and M. Welsh, "Citysense: A vision for an urban-scale wireless networking testbed," in *Proc. of the IEEE International Conference on Technologies for Homeland Security*, 2008.
2. J. Redi, S. Kolek, K. Manning, C. Partridge, R. Rosales-Hain, R. Ramanathan, and I. Castineyra, "JAVeLEN-An ultra-low energy ad hoc wireless network," *Ad Hoc Net-*

Table 4 Routing protocols

Routing	Latency	Energy cost	Reliability	Communication pattern	Types of WSNs
[21]	√			unicast	low duty-cycle WSNs
[29]			√	broadcast	low duty-cycle WSNs
[17]			√	unicast	WSNs
[35]		√	√	unicast	WSNs
[34]		√	√	unicast	WSNs
[40]		√	√	unicast	low duty-cycle WSNs
[36]		√	√	unicast	WSNs
[41]		√	√	unicast	WSNs
[42]			√	unicast	WSNs
[6]			√	unicast	WSNs
[14]			√	unicast	low duty-cycle WSNs
[38]	√			broadcast	low duty-cycle WSNs
[22]	√			unicast	WSNs
[5]			√	broadcast	low duty-cycle WSNs
[19]			√	unicast	WSNs
[30]		√		unicast	WSNs
[7]		√	√	broadcast	WSNs
[16]	√	√	√	broadcast	low duty-cycle WSNs
[15]			√	broadcast	WSNs
[43]	√		√	broadcast	low duty-cycle WSNs
[37]		√		unicast	WSNs
[26]	√			unicast	low duty-cycle WSNs
[3]	√	√		multicast	low duty-cycle WSNs
[44]		√		multicast	WSNs
[27]	√			unicast	low duty-cycle WSNs
[45]		√		unicast	low duty-cycle WSNs
[46]			√	unicast	low duty-cycle WSNs
[25]	√			unicast	low duty-cycle WSNs
[2]		√		unicast	WSNs
[32]		√		broadcast	WSNs
[31]		√		broadcast	WSNs
[33]		√		broadcast	WSNs
[18]			√	unicast	WSNs
[23]	√			unicast	low duty-cycle WSNs
[39]	√	√		unicast	WSNs
[24]	√			unicast	WSNs
[28]	√			unicast	low duty-cycle WSNs

works, vol. 6, pp. 108–126, 2008.
3. A. Ruzzelli, G. O'Hare, and R. Jurdak, "MERLIN: Cross-layer integration of MAC and routing for low duty-cycle sensor networks," *Ad Hoc Networks*, vol. 6, pp. 1238–1257, 2008.
4. Z. Zhong, T. Zhu, D. Wang, and T. He, "Tracking with unreliable node sequences," in *Proc. of IEEE Infocom*, 2009.
5. F. Wang and J. Liu, "On reliable broadcast in low duty-cycle wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 11, pp. 767–779, 2011.
6. S. Biswas and R. Morris, "ExOR: Opportunistic routing in multi-hop wireless networks," in *Proc. of ACM SIGCOMM*, 2005.
7. P. Kyasanur, R. Choudhury, and I. Gupta, "Smart gossip: An adaptive gossip-based broadcasting service for sensor networks," in *Proc. of IEEE MASS*, 2006.
8. J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. of the 2nd international conference on Embedded networked sensor systems (Sensys)*, 2004.
9. X. H. Mohd Riduan Ahmad, Eryk Dutkiewicz, "A survey of low duty cycle MAC protocols in wireless sensor networks," *Book Chapter in Wireless Sensor Network*, 2009.
10. W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networ," in *Proc. of IEEE Infocom*, 2002.
11. M. Buettner, G. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proc. of ACM Sensys*, 2006.
12. A. Arora, P. Dutta, *et al.*, "A line in the sand: A wireless sensor network for target detection, classification, and tracking," *Computer Networks*, vol. 46, pp. 605–634, 2004.
13. X. Bai, C. Zhang, D. Xuan, J. Teng, and W. Jia, "Low-connectivity and full-coverage three dimensional networks," in *Proc. of ACM Mobihoc*, 2009.
14. Y. Gu and T. He, "Data forwarding in extremely low duty-cycle sensor networks with unreliable communication links," in *Proc. of ACM Sensys*, 2007.
15. T. Zhu, Z. Zhong, T. He, and Z. Zhang, "Exploring link correlation for efficient flooding in wireless sensor networks," in *Proc. of the 7th USENIX Conference on Networked Systems Design and Implementation*, 2010.
16. S. Guo, S. Kim, T. Zhu, Y. Gu, and T. He, "Correlated flooding in low-duty-cycle wireless sensor networks," in *Proc. of IEEE ICNP*, 2011.
17. D. Aguayo and J. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proc. of ACM Mobicom*, 2003.
18. A. Basalamah, S. Kim, S. Guo, T. He, A. Tobe, Y.Basalamah, S. Kim, S. Guo, T. He, and Y. Tobe, "Link correlation aware opportunistic routing," in *Proc. of IEEE Infocom*, 2012.
19. J. Wang, Y. Liu, M. Li, W. Dong, and Y. He, "Qof: Towards comprehensive path quality measurement in wireless sensor networks," in *Proc. of IEEE Infocom*, 2011.
20. H. Song, S. Zhu, and G. Cao, "Svats: A sensor-network-based vehicle anti-theft system," in *Proc. of IEEE Infocom*, 2008.
21. Y. Gu, T. He, M. Lin, and J. Xu, "Spatiotemporal delay control for low-duty-cycle sensor networks," in *Proc. of IEEE Real-Time Systems Symposium (RTSS)*, 2009.
22. G. Lu, N. Sadagopan, B. Krishnamachari, and A. Goel, "Delay efficient sleep scheduling in wireless sensor networks," in *Proc. of IEEE Infocom*, 2005.
23. Y. Cao, S. Guo, and T. He, "Robust multi-pipeline scheduling in low-duty-cycle wireless sensor networks," in *Proc. of IEEE Infocom*, 2012.
24. Q. Zhang, Y. Gu, L. Gu, Q. Cao, and T. He, "Collaborative scheduling in highly dynamic environments using error inference," in *Mobile Ad-hoc and Sensor Networks (MSN), 2011 Seventh International Conference on*, pp. 105–114, IEEE, 2011.
25. P. Basu and C. Chau, "Opportunistic forwarding in wireless networks with duty cycling," in *Proc. of the3rd ACM Workshop on Challenged Networks*, 2008.

26. C. Chau and P. Basu, “Analysis of latency of stateless opportunistic forwarding in intermittently connected networks,” *IEEE/ACM Transactions on Networking (TON)*, vol. 19, pp. 1111–1124, 2011.
27. D. Kim and M. Liu, “Optimal stochastic routing in low duty-cycled wireless sensor networks,” in *Proc. of the 4th Annual International Conference on Wireless Internet*, 2008.
28. Y. Gu and T. He, “Bounding communication delay in energy harvesting sensor networks,” in *Distributed Computing Systems (ICDCS), 2010 IEEE 30th International Conference on*, pp. 837–847, IEEE, 2010.
29. S. Guo, Y. Gu, B. Jiang, and T. He, “Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links,” in *Proc. of ACM Mobicom*, 2009.
30. O. Chipara, Z. He, G. Xing, Q. Chen, X. Wang, C. Lu, J. Stankovic, and T. Abdelzaher, “Real-time power-aware routing in sensor networks,” in *Proc. of IEEE IWQoS*, 2006.
31. J. Wu, “Extended dominating-set-based routing in ad hoc wireless networks with unidirectional links,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, pp. 866–881, 2002.
32. J. Wu, B. Wu, and I. Stojmenovic, “Power-aware broadcasting and activity scheduling in ad hoc wireless networks using connected dominating sets,” *Wireless Communications and Mobile Computing*, vol. 3, pp. 425–438, 2003.
33. J. Wu and H. Li, “On calculating connected dominating set for efficient routing in ad hoc wireless networks,” in *Proc. of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications*, 1999.
34. M. Lu and J. Wu, “Social welfare based routing in ad hoc networks,” in *Proc. of IEEE ICPP*, 2006.
35. J. Wu, M. Lu, and F. Li, “Utility-based opportunistic routing in multi-hop wireless networks,” in *Proc. of IEEE ICDCS*, 2008.
36. M. Lu, F. Li, and J. Wu, “Efficient opportunistic routing in utility-based ad hoc networks,” *IEEE Transactions on Reliability*, vol. 58, pp. 485–495, 2009.
37. G. Ghidini and S. Das, “An energy-efficient markov chain-based randomized duty cycling scheme for wireless sensor networks,” in *Proc. of IEEE ICDCS*, 2011.
38. F. Wang and J. Liu, “Duty-cycle-aware broadcast in wireless sensor networks,” in *Proc. of IEEE Infocom*, 2009.
39. S. Bai, W. Zhang, G. Xue, J. Tang, and C. Wang, “Dear: Delay-bounded energy-constrained adaptive routing in wireless sensor networks,” in *Proc. of IEEE Infocom*, 2012.
40. Y. Gu and T. He, “Dynamic switching-based data forwarding for low-duty-cycle wireless sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 10, pp. 1741–1754, 2010.
41. M. Lu and J. Wu, “Utility-based data-gathering in wireless sensor networks with unstable links,” *Journal of Distributed Computing and Networking*, pp. 13–24, 2008.
42. Q. Cao, T. He, L. Fang, T. Abdelzaher, J. Stankovic, and S. Son, “Efficiency centric communication model for wireless sensor networks,” in *Proc. of IEEE Infocom*, 2006.
43. S. Lai and B. Ravindran, “On multihop broadcast over adaptively duty-cycled wireless sensor networks,” *Distributed Computing in Sensor Systems*, pp. 158–171, 2010.
44. C. Feng and W. Heinzelman, “Rbmulticast: Receiver based multicast for wireless sensor networks,” in *Proc. of IEEE WCNC*, 2009.
45. W. Pak, K. Cho, and S. Bahk, “Energy efficient routing protocol for wireless sensor networks with ultra low duty cycle,” in *Proc. of IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, 2009.
46. F. Yang and I. Augé-Blum, “Delivery ratio-maximized wakeup scheduling for ultra-low duty-cycled WSNs under real-time constraints,” *Computer Networks*, vol. 55, pp. 497–513, 2011.