Wireless Netw
DOI 10.1007/s11276-017-1473-6

Opportunistic broadcasting for low-power sensor networks with adaptive performance requirements

Lijie Xu[1,2] · Geng Yang[1] · Lei Wang[1] · Jia Xu[1] · Baowei Wang[3,4]

© Springer Science+Business Media New York 2017

Abstract To reduce the energy waste caused by idle listening, sensor nodes in wireless sensor networks (WSNs) usually work with low-duty-cycle mode. However, such mode brings many new challenges, especially for broadcasting applications. This paper proposes to exploit the broadcast nature of wireless media to further save energy for broadcasting in low-duty-cycle WSNs, by adopting a novel opportunistic broadcasting transmission model. The key idea is to allow nodes to defer their wake-up time slots to opportunistically overhear the broadcasting messages sent by their neighbors, improving the energy efficiency at the cost of the increase of average broadcasting delay. Instead of regarding delay or energy as the single optimization objective, in this paper, we present a broadcasting cost function, which provides an adaptive control on the tradeoff between delay and energy to cover various performance requirements. Our target is thus to find the optimal broadcasting schedule to minimize the broadcasting cost, based on the opportunistic broadcasting transmission model. To this end, we first model the target problem under the single-hop case as a dynamic programming problem and prove it is solvable in polynomial time, then extend it to the multi-hop case and come up with an efficient solution. Extensive simulation results reveal that our solution always has a better performance over the other solutions under whatever configurations.

Keywords Wireless sensor networks · Low-duty-cycle · Opportunistic broadcasting transmission model · Broadcasting schedule

1 Introduction

In recent years, the Internet of Things (IOT) has been developed rapidly with the progress of many emerging techniques, such as wireless sensor networks (WSNs) [1–4], multimedia processing [5–8], 4G communication technique [9] and cloud computing [10–14]. As the key technique of IOT, WSNs has made a great progress on the research areas of routing [15, 16], coverage [17, 18], topology control [19] and localization [20] etc.. More and more real WSNs applications have been deployed in various fields, they almost require that the system should run for a relatively long period. Also, it is usually hard to replace or recharge batteries for sensor nodes since many WSNs are deployed in a tough environment. This fact implies the importance to take energy efficiency as the first concern for

---

✉ Lijie Xu
ljxu@njupt.edu.cn

Geng Yang
yangg@njupt.edu.cn

Lei Wang
leiwang@njupt.edu.cn

Jia Xu
xujia@njupt.edu.cn

Baowei Wang
wbw.first@163.com

[1] School of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing 210023, People's Republic of China

[2] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, People's Republic of China

[3] School of Computer and Software, Nanjing University of Information Science & Technology, Nanjing 210044, People's Republic of China

[4] Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology, Nanjing 210044, People's Republic of China

Published online: 24 February 2017

WSNs. The existing work [21] has verified that idle listening is the major source of energy waste in sensor networks with low traffic. In order to greatly reduce the energy waste caused by idle listening, sensor nodes usually work with low-duty-cycle mode, namely each node has its own working schedule to alternate between working state and sleeping state. Low-duty-cycle mode makes each node sleep periodically so as to adapt to the low traffic characteristic of sensor networks, it thus facilitates the improvement of energy efficiency. However, it brings many new challenges regarding delay performance and energy performance. On the one hand, the difference of working schedules between neighboring nodes makes the network yield a notable increase on communication delay, which is usually called *sleep latency* [22]. On the other hand, it will have a significant effect on the energy efficiency for the broadcasting applications.

In WSNs, broadcasting is a frequently-used fundamental function. Many applications often require the sink node distribute the broadcasting message to the whole network in a multi-hop way. Besides, the single-hop broadcasting is also widely applied in many small-scale networks, or in many routing protocols where neighboring nodes usually require local information exchange. For low-duty-cycle WSNs, how to make an energy-efficient broadcasting is a challenging issue, this is because the fact that neighboring nodes usually have different working schedules disables the inherent broadcast characteristic of wireless media. This implies that any single-hop broadcast will be implemented with multiple unicasts, which is energy inefficient especially for large message broadcasting, such as *code update*. Actually, we can find that the broadcast nature of wireless media still offers opportunities to reduce the total energy consumption of broadcasting, even for low-duty-cycle networks. In this paper, we come up with a novel opportunistic broadcasting transmission model, which fully exploits the spatiotemporal correlation of wireless media to reduce the total energy consumption of broadcasting. The basic idea is to allow nodes to postpone their wake-up time slots to opportunistically overhear the messages sent by their neighbors, improving the energy efficiency at the cost of the increase of average broadcasting delay. We can find that by carefully adjusting the wake-up schedules of nodes, this model can provide a flexible control on the tradeoff between average broadcasting delay and total energy consumption of broadcasting.

Existing studies for broadcast problem in low-duty-cycle WSNs always regard delay or energy as the single optimization objective, these works are only available for the applications with single performance requirement, and thus the application scope is limited. For most of the real broadcasting applications, actually, both of delay and energy are important to the network performance and they should both be taken into consideration. Typically, performance requirements are application-specific, namely different broadcasting applications could have totally different performance requirements. For example, the applications with small and urgent messages would pay more attention on delay than energy, while the applications with large and non-urgent messages would pay more attention on energy than delay. Even for the same broadcasting application, we can find that performance requirements could also be different during different periods of the network lifetime. Therefore, it is of great significance to define a generalized optimization objective function that is available for applications with various performance requirements.

In this paper, we define a broadcasting cost function that characterizes the tradeoff between average broadcasting delay and total energy consumption of broadcasting, in which the parameter can be adaptively adjusted to satisfy various performance requirements. Our objective is thus to design an efficient broadcasting schedule for low-duty-cycle WSNs to minimize the broadcasting cost function, based on the proposed opportunistic broadcasting transmission model.

The main contributions of this work are summarized as follows:

- Instead of regarding delay or energy as the single optimization objective, we present a generalized optimization objective function that can provide an adaptive control on the tradeoff between delay and energy efficiency, so as to cover various performance requirements.
- We propose a novel opportunistic broadcasting transmission model for low-duty-cycle WSNs. This model allows some receivers to defer their wake-up time slots to opportunistically overhear the broadcasting messages sent by the sender for any single-hop case, and introduces an efficient pre-beacon scheme for multi-hop case.
- We model the target problem under the single-hop case as a dynamic programming problem and prove that it is solvable in polynomial time. Then, we extend it to the multi-hop case and propose an efficient solution.
- Extensive simulation results reveal that our solution always has a better performance over the other solutions under whatever configurations.

The rest of the paper is organized as follows: Sect. 2 summarizes the related work. Section 3 illustrates the network model and states the problem. Detailed description of our proposed algorithms are presented in Sect. 4. Followed by the simulation results in Sect. 5. Finally, we conclude our findings and state the future work in Sect. 6.

2 Related work

Over the past years, opportunistic routing [23] has received much attention. The key idea of opportunistic routing is to overcome the drawback of unreliable wireless transmission by taking advantage of the broadcast nature of the wireless medium, such that one transmission can be overheard by multiple neighbors. The existing works have proposed many efficient opportunistic routing schemes to improve the network performance. Biswas and Morris [24] proposed Extremely Opportunistic Routing (ExOR), a novel unicast routing technique for wireless multi-hop networks. ExOR forwards each packet through a sequence of nodes, deferring the choice of each node in the sequence until after the previous node has transmitted the packet on its radio. Simulation results showed that ExOR reduces the total number of transmissions by nearly a factor of two over the best possible pre-determined route. In [25], the authors proposed an analytical model to describe any routing procedures operating according to the opportunistic paradigm, and exploited such a model to derive a closed-form expression of the average number of data-link transmissions needed to successfully deliver a packet. In [26], the authors addressed the problem of the optimal candidate-set selection in the opportunistic routing paradigm, they provided an analytical framework to model both the optimal constrained and unconstrained candidate-set selection. and proposed two efficient algorithms. However, these existing works usually assumed that target network is always-awake. In this paper, we generally borrow the idea from opportunistic routing, and focus on how to exploit the broadcast nature of the wireless medium to make an opportunistic broadcasting for low-duty-cycle networks. Different from the traditional opportunistic routing schemes, our work makes a cross-layer design to joint sleep scheduling and opportunistic overhearing.

Recently, the broadcast problem for low-duty-cycle sensor networks has also been extensively studied by the research community. In terms of the optimization objective, most of the existing works on this topic can be classified into the following two categories.

(1) *Delay optimization* [27–31, 33–37]: Guo et al. [27] proposed an opportunistic flooding scheme for low-duty-cycle WSNs with unreliable links, this scheme improves the broadcasting delay by letting the senders make probabilistic forwarding decisions based on the delay distribution of next-hop nodes. Khiati and Djenouri [30] utilized clustering to realize the broadcasting for low-duty-cycle WSNs, the concurrent transmissions of the broadcasting packets between multiple clusters will result in a notable reduction on broadcasting delay. Lu and Whitehouse [31] designed an efficient broadcasting protocol for low-duty-cycle WSNs based on the capture effect in physical layer, this protocol greatly reduces the broadcasting delay by allowing the concurrent transmissions between multiple nodes. In [33], the authors analyzed the major factors that effect the broadcasting delay for low-duty-cycle WSNs and how these factors (duty cycle, link quality, etc.) effect the broadcasting delay, the analysis results can provide us a guideline to design better algorithms for broadcasting delay optimization.

(2) *Energy optimization* [32, 38–51]: In [38], the authors studied the minimum transmission broadcast problem in duty-cycled sensor networks, they first proved its NP-hardness, then proposed a centralized approximation algorithm with a logarithmic approximation ratio and a distributed approximation algorithm with a constant approximation ratio. In [39], the authors proposed two distributed connected-dominating-set based algorithms for the minimum transmission broadcast problem, and proved that it can achieve a lower constant approximation ratio than that proposed in [38]. However, both [38] and [39] assumed a strict sleep scheduling model where each working schedule period contains exactly one *active slot*. [40] considered a general model that allows each working schedule period to contain more than one *active slot* and proposed an efficient approximation algorithm. In [41], the authors investigated the minimum energy reliable broadcast problem for low-duty-cycle WSNs with unreliable links and adjustable transmission power, they proved that it is NP-hard and presented an efficient algorithm with a polylogarithmic approximation ratio. [43] considered the minimum energy broadcast problem for low-duty-cycle WSNs with a given delay constraint, the authors proposed an efficient heuristic solution and verified its high efficiency by extensive simulations. In our recent work [32], we utilized the broadcasting spatiotemporal locality to address the latency-optimal minimum energy broadcast problem for low-duty-cycle WSNs. However, the proposed broadcasting transmission model in [32] assumed that each forwarder cannot send the beacon packets until the broadcasting message is received, which is inefficient. In this paper, we further improve this model by introducing an efficient pre-beacon scheme, that is, our proposed model relaxes the assumption made in [32] and allows any node to send beacon packets to its next-hop nodes immediately after its reception on a beacon packet. Obviously, our proposed broadcasting transmission model in this paper is more efficient than that in [32]. Also, the target problem in this paper is totally different with that in [32], we mainly focus on the tradeoff between delay and energy for broadcasting, rather than regard energy as the single optimization objective. Compared with [32], the objective of this paper

is of greater significance, since it is totally unnecessary to require the broadcasting should be done within a bounded delay for many real applications, and more real broadcasting applications focus on the tradeoff between delay and energy.

Currently, very a few of the existing works considered the tradeoff between delay and energy as the optimization objective. In [52], the authors first defined a cost function that explicitly characterizes the tradeoff between the broadcasting delay and the total energy consumption for the duty-cycle-aware WSNs, and then proposed an efficient broadcasting schedule to minimize the cost function. By assigning different tradeoff parameters, the cost function covers the performance requirements from a broad spectrum of applications. However, it adopted the traditional broadcasting transmission model, namely any local single-hop broadcast is inefficiently implemented with multiple unicasts, which do not make full use of the spatiotemporal locality of broadcasting. Actually, the broadcast nature of wireless media offers opportunities to reduce the total energy consumption for broadcasting, even for low-duty-cycle WSNs. To the best of our knowledge, our work is the first to utilize the broadcasting spatiotemporal locality to address the tradeoff problem between delay and energy for broadcasting in low-duty-cycle WSNs.

3 Motivation

3.1 Network model and assumptions

Without loss of generality, we assume that all the sensor nodes are uniformly deployed in a square sensory field, in which the sink is located at the center, and each node has the same communication range. Also, it is assumed that time is divided into a lot of equal time slots, and each time slot is set long enough so that it can accommodate one potential broadcasting message transmission. Each time slot is either a *sleeping slot* where each node will turn all its function modules off except a timer to wake itself up, or an *active slot*, where each node will keep awake for a short duration at the beginning, which is called *listening interval*, to make the event sensing and channel listening.

In this paper, we assume that all the sensor nodes work with low-duty-cycle mode, where each node determines its own working schedule independently. For simplicity, we assume each node has a periodic working schedule with the period length L. Specifically, the working schedule of each node will alternate between one *active slot* and $L-1$ *sleeping slots*. For any node v_i, we use $t(v_i)$ to represent the scheduled active time slot in each working schedule period $(0 \le t(v_i) \le L-1)$. Figure 1 explicitly illustrates an example of the periodic working schedule where $L = 5$ and $t(\cdot) = 2$. Further, we use the undirected graph $G = (V, E)$ to represent the network topology, where V represents the set of $Num + 1$ nodes including one sink node v_0 and Num sensing nodes $\{v_1, \ldots, v_{Num}\}$, and E represents the set of all communication links. For any edge $(v_i, v_j) \in E$, we denote by $d(v_i, v_j)$ the point-to-point transmission delay (i.e., sleep latency) from node v_i to node v_j, and $d(v_i, v_j)$ can be calculated as follows:

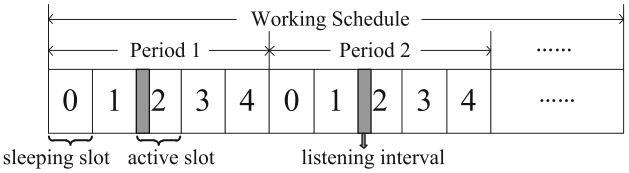

Fig. 1 An example of working schedule with $L = 5$

$$d(v_i, v_j) = \begin{cases} t(v_j) - t(v_i), & \text{if } t(v_j) > t(v_i); \\ t(v_j) - t(v_i) + L, & \text{otherwise.} \end{cases} \tag{1}$$

A summary of the primary notations in this paper is given in Table 1. Also, we have the following basic assumptions, that are commonly made in most of the research works regarding low-duty-cycle WSNs:

- Local time synchronization is achieved, and each node can forward its packets at any time while it can only receive the packets in active slots. Specifically, each node v_i will wake up at the beginning of any *active slot* and keep listening for a duration of *listening interval*, if any broadcasting packet with the target receiver ID v_i is received, it will keep receiving until all packets of the broadcasting message are received and then go to sleep; otherwise, it will go to sleep immediately. If any sender is expected to forward a broadcasting message to some receiver, it will set a timer to wake itself up at the beginning of the receiver's next *active slot* to complete the transmission, and then go to sleep.
- Each node is aware of the working schedules of all its neighboring nodes within two hops, this can be realized via local information exchange between neighboring nodes initially after the network is deployed.
- For simplicity, we do not consider the potential packet collision problem due to the fact that the low-duty-cycle mode inherently reduces the probability of collision to a great extent, which has been experimentally verified in [52].
- For better description, we assume the working schedules of any node and its neighbors are different from each other, which is usually true for low-duty-cycle WSNs. Actually, our solution proposed in this paper is still available for the general case where a few of the neighboring nodes could have the identical wake-up schedule. We will discuss this general case in Sect. 4.4.

Table 1 Summary of the primary notations

Notation	Meaning
L	The period length of each node's working schedule
η	The tradeoff factor in the broadcasting cost function
$t(v_i)$	The active slot in each working schedule period of node v_i
$d(v_i, v_j)$	Point-to-point transmission delay (sleep latency) from node v_i to node v_j
$S_f(v_i)$	The forwarding sequence of the forwarder v_i
$M(v_i)$	The forwarding decision of node v_i
$delay(v_i)$	End-to-end delay from the initiator to node v_i
$delay^*(v_i)$	The optimal end-to-end delay from the initiator to v_i
$G_s(v_i, R)$	The single-hop network topology with the sender v_i and the set of receivers R
$cost(v_i^j, v_i^k)$	The resulted broadcasting cost if $S_f(v_i) = <v_i^j, v_i^{j+1}, \ldots, \underline{v_i^k}>$, where only node v_i^k is the *InstantReceiver*
$D(v_i^j, v_i^k)$	The sum of the sleep latency from each node $v_i^m (j \le m \le k)$ to node v_i^k
$OPT(j)$	The optimal broadcasting cost for the single-hop network with one sender v_i and j receivers
$CPS(v_i)$	Candidate parents set of node v_i
$CCS(v_i)$	Candidate children set of node v_i
$CF(v_i)$	Competition factor of the candidate forwarder v_i
CFS	Candidate forwarders set

3.2 Broadcasting transmission model

Typically, energy efficiency and delay are the major performance metrics for evaluation of broadcasting algorithms. In this paper, we regard the total energy consumption of broadcasting and average broadcasting delay as the metric of energy efficiency and delay, respectively.

Note that, average broadcasting delay, which denotes the average of end-to-end (E2E) delay from the sink (the initiator) to all the sensor nodes in the network, is usually an important metric to evaluate the performance of broadcasting. For many broadcasting applications such as *configuration dissemination*, each node is expected to receive the broadcasting message as soon as possible to update the configuration, so that the new system requirement can be satisfied in a short period of time. For example, detection alarm system is a type of widely used applications for WSNs. Upon detecting that the reader (*e.g.*, temperature, humidity etc.) is above or below some threshold, the sensor node will report it to the sink quickly so that a prompt action can be taken. For such kind of applications, average broadcasting delay is a critical performance metric, since sometimes we need to change the system requirement (*e.g.*, to change the alarm threshold), which requires the sink distribute the updated alarm threshold to each node in the network as soon as possible, so as to reduce the chance of false positive or false negative as much as possible. Besides, the broadcasting applications are usually expected to have a low average broadcasting delay, to reduce the probability of collisions between broadcasting packets and data collecting packets.

Traditionally, all nodes in the network will receive the broadcasting message at their scheduled *active slots* which could lead to the minimum average broadcasting delay but, however, draw much more energy consumption since any single-hop broadcast is actually implemented by multiple unicasts. To improve energy efficiency for broadcasting, in this paper, we propose a novel and efficient opportunistic broadcasting transmission model. For any sender, this model defines two types of receivers: *DelayedReceivers* and *InstantReceivers*. The sender will forward the broadcasting message to each *InstantReceiver*, also, it will send a short beacon packet $Beacon(v_j)$, which only contains the ID of some *InstantReceiver* v_j, to any *DelayedReceiver*. Upon receiving $Beacon(v_j)$ from the sender, any *DelayedReceiver* will go to sleep immediately and set a timer to wake up itself at the next *active slot* of the *InstantReceiver* v_j, so that it can opportunistically overhear the broadcasting message which is sent from the sender to the *InstantReceiver* v_j. Note that, due to the assumption that each node knows the working schedules of all its neighboring nodes within two hops, the *DelayedReceiver* can be aware of the working schedule of the *InstantReceiver* v_j.

Figure 2 illustrates a simple broadcast example with one-hop case, where the number labeled within each square denotes the scheduled *active slot* (i.e., $t(v_0) = 3$, $t(v_1) = 5$, $t(v_2) = 8$, $t(v_3) = 1$) and the schedule period length L is set as 10. Figure 2(a) shows a traditional solution, in which the sender v_0 delivers the message to its neighbors one by one to implement the broadcasting (i.e., to set nodes v_1, v_2 and v_3 as the *InstantReceivers*). It has an average broadcasting delay of $(2 + 5 + 8)/3 = 5$ and requires a total energy

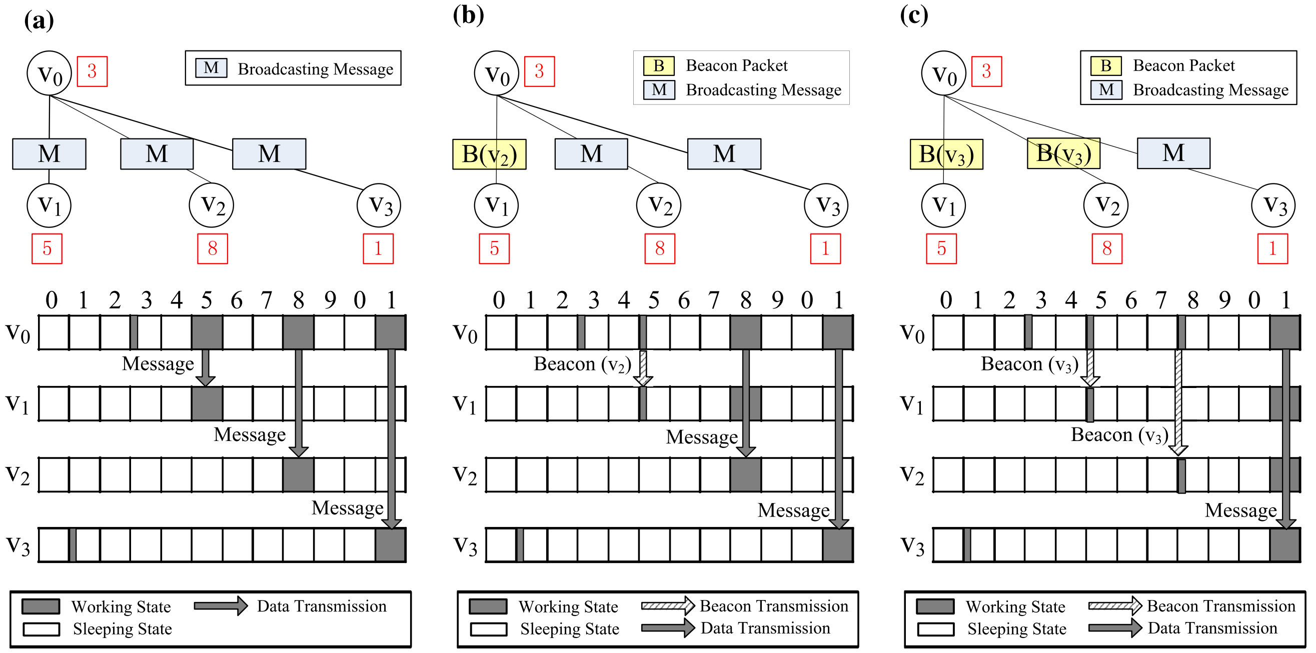

Fig. 2 Illustration of opportunistic broadcasting transmission model: single-hop case **a** broadcast without deferring, **b** broadcast with one *DelayedReceiver*, **c** broadcast with two *DelayedReceivers*

consumption of $E_{total} = 3 \times k \times e_s^d + 3 \times k \times e_r^d$ $(k \geq 1)$, where k denotes the number of data packets in a broadcasting message, e_s^d and e_r^d denote the energy consumption of sending and receiving a data packet, respectively. As shown in Fig. 2(b), if the sender v_0 delivers the beacon packet *Beacon*(v_2) to the *DelayedReceiver* v_1 and delivers the broadcasting message to the *InstantReceivers* $\{v_2, v_3\}$, node v_1 will defer its message receiving time by setting a timer to wake up itself at the next scheduled *active slot* of the *InstantReceiver* v_2 (i.e., time slot 8). By this way, the average broadcasting delay will be increased to $(5+5+8)/3 = 6$ and the total energy consumption will be $E'_{total} = e_s^b + e_r^b + 2 \times k \times e_s^d + 3 \times k \times e_r^d$, where e_s^b and e_r^b denote the energy consumption of sending and receiving a beacon packet, respectively. As shown in [53], it is usual that a data packet has a length of 133 bytes and a beacon packet has only a length of 19 bytes, which indicates that $e_s^b + e_r^b$ is far less than e_s^d in practice. So, total energy benefit of deferring the message receiving time of any receiver, i.e., $\Delta' = E_{total} - E'_{total} = k \times e_s^d - (e_s^b + e_r^b)$, must be greater than 0. For applications with large message broadcasting (*e.g.*, *code update*), especially, this benefit will be significant as $k \gg 1$. Figure 2(c) shows an example of broadcast with two *DelayedReceivers*, namely, the sender v_0 delivers the beacon packet *Beacon*(v_3) to the *DelayedReceivers* $\{v_1, v_2\}$, and delivers the broadcasting message to the *InstantReceiver* v_3. Obviously, it will have an average broadcasting delay of $(8+8+8)/3 = 8$ and a total energy consumption of $E''_{total} = 2 \times (e_s^b + e_r^b) + k \times e_s^d + 3 \times k \times e_r^d$. Compared with the traditional solution, total energy benefit of this way is $\Delta'' = E_{total} - E''_{total} = 2 \times k \times e_s^d - 2 \times (e_s^b + e_r^b) > \Delta' > 0$. According to the above observation, we can easily find that by deferring the message reception time of some receivers , the total energy consumption for broadcasting will be reduced, at the cost of the increase of average broadcasting delay. Specifically, the total energy benefit will increase as the number of *InstantReceivers* decreases, which implies the following conclusion:

Observation 1 *Based on the proposed opportunistic broadcasting transmission model, the total energy consumption for broadcasting can be essentially characterized by the number of the InstantReceivers* (*i.e., the total number of broadcasting message transmissions*).

By this model, we can find that more *DelayedReceivers* will result in more reduction on total energy consumption, however, bring longer average broadcasting delay. Thus, this model actually provides a flexible control on the tradeoff between average broadcasting delay and total energy consumption of broadcasting.

For multi-hop case, this model adopts an efficient pre-beacon scheme. The basic idea is to allow any node to send beacon packets to its next-hop nodes immediately after its reception on a beacon packet. By carefully designing the working schedules of all nodes in a multi-hop network, both the beacon packets and the broadcasting message can be transmitted in a timely way. According to Observation 1, note that, we will simply employ the number of *InstantReceivers* to characterize the total energy consumption of broadcasting for multi-hop case.

Here, we will take a simple example on a tree-like topology, which is shown in Fig. 3, to illustrate the opportunistic broadcasting transmission model for multi-hop case. Figure 3(a) shows a traditional solution without deferring. which can result in the minimum average broadcasting delay 5.4, but, draw the maximum of total energy consumption (the number of *InstantReceivers* is 9). According to the previous analysis for single-hop case, we can see that the energy would benefit from deferring the message reception time of some receivers. Intuitively, this model can be extended to multi-hop case by simply regarding a multi-hop broadcasting as multiple single-hop sessions. In our previous work [32], we adopted a simple and direct model for multi-hop case, which supposes that each forwarder should not send beacon packets to its next-hop neighbors until it has received the broadcasting message. Figure 3(b) explicitly illustrates an example of such intuitive solution without pre-beacon scheme. In this example, node v_2 defers its wake-up time slot to the next *active slot* of node v_4, to opportunistically overhear the broadcasting message which is sent from v_0 to v_4. Upon receiving the broadcasting message at time slot 7, the forwarder v_2 could send the beacon packet *Beacon*(v_6) to the *DelayedReceiver* v_7 and the *DelayedReceiver* v_5 in turn, and then forward the broadcasting message to the *InstantReceiver* v_6. Also, the forwarder v_5 would send the beacon packet *Beacon*(v_8) to the *DelayedReceiver* v_9 only after receiving the broadcasting message at time slot 6, and then forward the broadcasting message to the *InstantReceiver* v_8. For this solution, the network only has 3 *InstantReceivers*, but the average broadcasting delay will be increased to 14. However, this model is not so efficient, since the assumption that each forwarder can send the beacon packets to its next-hop neighbors only if

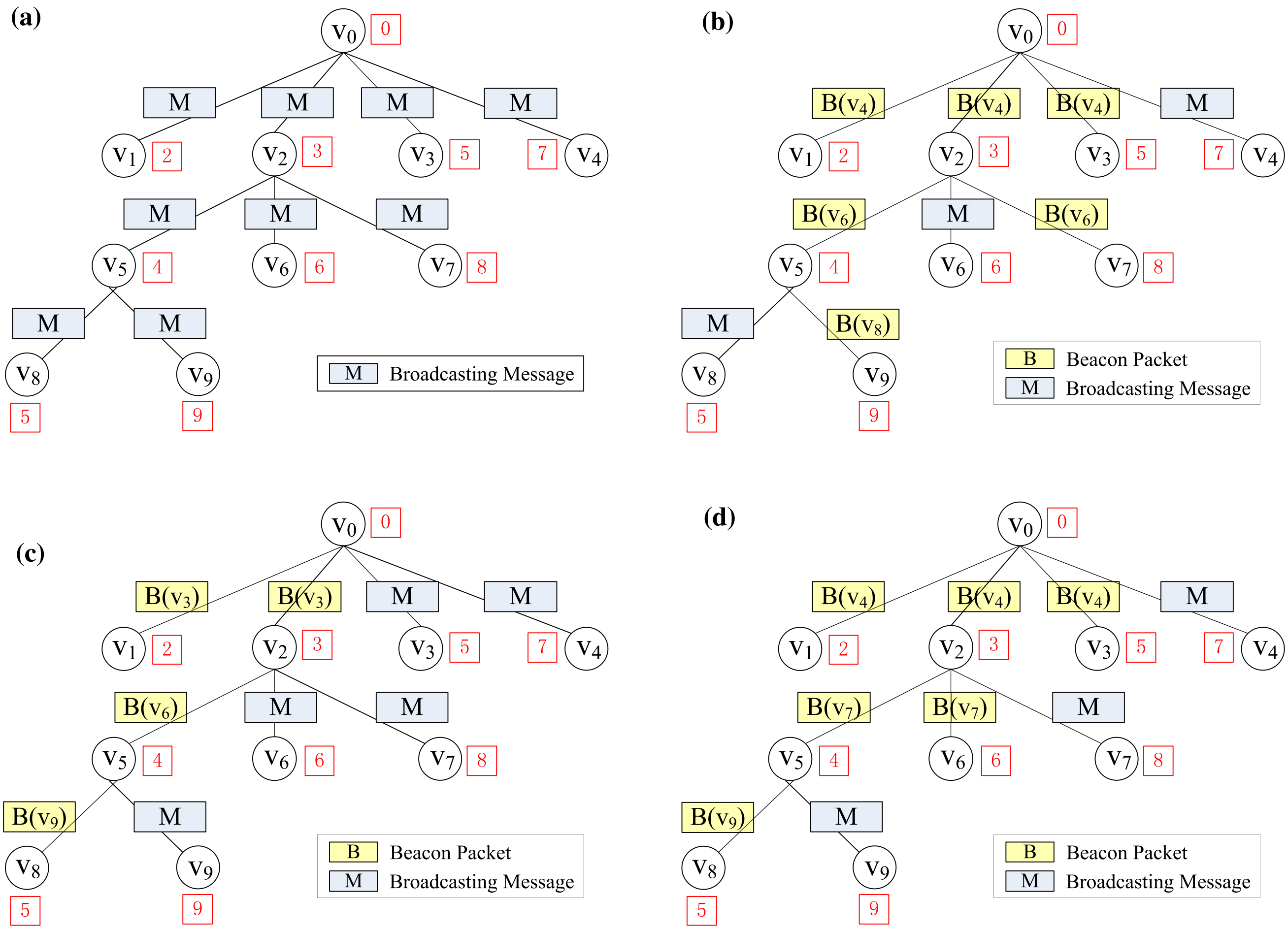

Fig. 3 Illustration of opportunistic broadcasting transmission model: multi-hop case **a** broadcast without deferring, **b** broadcast without pre-beacon scheme, **c** broadcast with pre-beacon scheme (4 *DelayedReceivers*), **d** broadcast with pre-beacon scheme (6 *DelayedReceivers*)

it has received the broadcasting message, would result in a relatively long average broadcasting delay.

In this paper, we propose a pre-beacon scheme to improve this broadcasting transmission model for multi-hop case. We allow any forwarder, who is the *DelayedReceiver*, to send the beacon packets to its next-hop neighbors between its beacon reception time and message reception time. Figure 3(c) illustrates an example of our proposed model with pre-beacon scheme. As shown in Fig. 3(c), the sink v_0 sends the beacon packet $Beacon(v_3)$ to node v_1 and node v_2, to let them postpone their message reception time to the next *active slot* of node v_3. Upon receiving the beacon packet, the forwarder v_2 will send a beacon packet $Beacon(v_6)$ to node v_5 at time slot 4, even v_2 has not received the broadcasting message at this moment. Also, the forwarder v_5 will send a beacon packet $Beacon(v_9)$ to node v_8 at time slot 5, which is between v_5's beacon reception time (time slot 4) and message reception time (time slot 6). Accordingly, we can find that by allowing any forwarder to send the beacon packets before its broadcasting message reception time, the broadcasting message could be forwarded in a more delay-efficient way. Note that, such model with pre-beacon scheme requires the working schedules of all nodes should be carefully designed, so that the message reception time must precede the message forwarding time for each forwarder. In this example, the average broadcasting delay is 6.7 and the number of *InstantReceivers* is 5. Figure 3(d) illustrates an extreme case of the our proposed model with pre-beacon scheme, which has the minimum number of *InstantReceivers* (i.e., 3) but the maximum average broadcasting delay (i.e., 7.8). As shown in Fig. 3(b) and (d), it is obvious that our proposed model with pre-beacon scheme will have a better average broadcasting delay than that without pre-beacon scheme, in the event that they have the same number of *InstantReceivers*.

3.3 Problem statement

Before the statement of our target problem, we first present the definition of *broadcasting schedule* for any low-duty-cycle sensor network, which is based on the opportunistic broadcasting transmission model.

Definition 1 (*Forwarding Sequence*) For any forwarder v_i of the broadcasting message, its *forwarding sequence* $S_f(v_i)$ is defined as a sequenced set of its receivers sorted based on the scheduled wake-up time, namely

$$S_f(v_i) = < [r_1^1, \ldots, r_1^{k_1}], \underline{r_1}, \ldots, [r_j^1, \ldots, r_j^{k_j}], \underline{r_j} > , \tag{2}$$

where $r_j^k (k = 1, \ldots, k_j)$ and the underlined r_j respectively denote the *DelayedReceivers* and *InstantReceivers* of node v_i. Specifically, the forwarder v_i will send the short control packet $Beacon(r_j)$ to each *DelayedReceiver* r_j^k and send the broadcasting message to each *InstantReceiver* r_j. Here, [] denotes an optional item.

Definition 2 (*Forwarding Decision*) Given a low-duty-cycle sensor network $G = (V, E)$ and the working schedules of all nodes, the *forwarding decision* of any node $v_i \in V$, saying $M(v_i)$, can be described as the following 2-tuple:

$$\begin{aligned} M(v_i) &= (\alpha, \beta), \\ \alpha \in \{0, \ldots, degree(v_i)\}, \beta &= \begin{cases} S_f(v_i), & \alpha > 0; \\ NULL, & \alpha = 0. \end{cases} \end{aligned} \tag{3}$$

In Eq. 3, $degree(v_i)$ denotes the node degree of v_i, and the variable α denotes node v_i's total forwarding number of the broadcasting message. If $M(v_i).\alpha > 0$, it implies v_i is the forwarder and $M(v_i).\beta$ will denote its *forwarding sequence* $S_f(v_i)$. If $M(v_i).\alpha = 0$, it implies v_i is not the forwarder and $M(v_i).\beta = NULL$, where $NULL$ denotes the omitted item. Specially, it must have $M(v_0).\alpha > 0$ for the sink v_0.

Definition 3 (*Broadcasting Schedule*) Given a low-duty-cycle sensor network $G = (V, E)$ and the working schedules of all nodes, a *broadcasting schedule M* in the network is defined as the set of all nodes' *forwarding decisions*:

$$M = \{M(v_i) | v_i \in V\}, \tag{4}$$

which is subject to the following constraints:

(1) Given an edge set $E' \subseteq E$ where any edge $(v_i, v_j) \in E'$ if and only if $M(v_i).\alpha > 0$ and $v_j \in M(v_i).\beta$ (or, $M(v_j).\alpha > 0$ and $v_i \in M(v_j).\beta$), then the undirected graph $G' = (V, E')$ must be a spanning subtree of G.

(2) For any forwarder v_i, the time that it receives the beacon packet (if v_i is the *DelayedReceiver*), or the time that it receives the broadcasting message (if v_i is the *InstantReceiver*), must precede the time that it forwards the beacon packets or the broadcasting message to each node in $M(v_i).\beta$.

(3) For any forwarder v_i, its message reception time must precede the time that it forwards the message to each *InstantReceiver* in $M(v_i).\beta$.

As stated in Sect. 1, in this paper, we mainly focus on a generalized optimization objective that is available for applications with various performance requirements. Given a sensor network $G = (V, E)$, specifically, we define a broadcasting cost function as follows:

$$cost(G) = DPI + \eta \times EPI \qquad (\eta \geq 0), \tag{5}$$

where DPI (Delay Performance Index) and EPI (Energy Performance Index) are variables that respectively characterize the delay performance and the energy performance, the parameter η denotes a tradeoff factor. By adaptively adjusting the non-negative parameter η, obviously, this broadcasting cost function can provide a flexible control over the tradeoff between delay and energy efficiency to satisfy various performance requirements.

In this paper, we will employ the average broadcasting delay to characterize DPI. Let $delay(v_i)$ and $delay^*(v_i)$ denote the real E2E delay and the theoretically optimal E2E delay from the initiator to any sensor node v_i, respectively, then the average broadcasting delay for any sensor network G can be represented as follows:

$$\begin{aligned}\widehat{delay} &= \frac{\sum\limits_{v_i \in V} delay(v_i)}{Num} \\ &= \frac{\sum\limits_{v_i \in V} delay^*(v_i) + \sum\limits_{v_i \in V} (delay(v_i) - delay^*(v_i))}{Num}\end{aligned} \tag{6}$$

where $delay(v_i) = delay^*(v_i) = 0$ if v_i is the initiator of the broadcasting, and Num is the total number of sensor nodes in G.

Seeing from Eq. 6, $\sum\limits_{v_i \in V} delay^*(v_i)$ and Num are basically fixed for any given network G, thus, the average broadcasting delay can be essentially characterized by

$$\Delta_{delay} = \sum_{v_i \in V} (delay(v_i) - delay^*(v_i)), \tag{7}$$

which denotes the sum of the incremental E2E delay for all nodes. Accordingly, we will employ Eq. 7 to characterize DPI. Based on Observation 1, also, we can essentially employ the total number of the broadcasting message transmissions, i.e., $\sum\limits_{v_i \in V} M(v_i).\alpha$, to characterize EPI.

In this paper, we combine the optimization on such a cost function with the opportunistic broadcasting transmission model, and our target is to address the following *Opportunistic Minimum Cost Broadcast Problem* (OMCB).

Problem 1 (*OMCB*) Given a low-duty-cycle sensor network $G = (V, E)$ and the working schedules of all nodes, how to find an efficient *broadcasting schedule* M in G, based on the opportunistic broadcasting transmission model, to minimize the following broadcasting cost function:

$$cost(G) = \Delta_{delay} + \eta \times \sum_{v_i \in V} M(v_i).\alpha \tag{8}$$

where $\Delta_{delay} = \sum\limits_{v_i \in V} (delay(v_i) - delay^*(v_i))$, and η denotes a non-negative parameter.

Note that, the tradeoff factor η is usually determined by the specific application with specific performance requirement in practice.

4 Opportunistic broadcasting algorithm

4.1 Overview

In this section, we will focus on the solution to our target problem. First, we address our target problem under the single-hop case. by adopting a dynamic programming approach, of which time complexity is in polynomial time. Then, we extend it to the multi-hop case, and come up with an efficient solution. Finally, we discuss how to extend our solution to the general case, where a few of the neighboring nodes could have the identical wake-up schedule.

4.2 The single-hop case

Here, we first consider how to solve our target problem under the single-hop case. The single-hop broadcasting is typically applied in many small-scale networks, or in many multi-hop routing protocols, where neighboring nodes usually require local information exchange. For any single-hop broadcasting, the sender is the only forwarder, which implies the *broadcasting schedule* only depends on the determination of the sender's *forwarding decision*.

We denote by $G_s(v_i, \{v_{j_1}, \ldots, v_{j_N}\})$ a star-like single-hop network topology with the sender (initiator) v_i and N receivers $\{v_{j_1}, \ldots, v_{j_N}\}$. For better description, we sort all the receivers according to the ascending order of the sleep latency from the sender to them, and the sorted receivers are re-marked as $\{v_i^1, v_i^2, \ldots, v_i^N\}$, where $d(v_i, v_i^j) < d(v_i, v_i^{j+1})$ $(1 \le j \le N-1)$. In this subsection, our target is to solve the following *Single-hop based Opportunistic Minimum Cost Broadcast Problem* (S-OMCB)

Problem 2 (*S-OMCB*) Given a single-hop network $G_s(v_i, \{v_i^1, \ldots, v_i^N\})$ where $d(v_i, v_i^j) < d(v_i, v_i^{j+1}) (1 \le j \le N-1)$, how to determine the optimal broadcasting schedule M in G_s, to minimize the following broadcasting cost function:

$$cost(G_s) = \Delta_{delay} + \eta \times M(v_i).\alpha \tag{9}$$

where $\Delta_{delay} = \sum\limits_{k=1}^{N} (delay(v_i^k) - delay^*(v_i^k))$.

Theorem 1 *Given a single-hop network* $G_s(v_i, \{v_i^1, \ldots, v_i^N\})$ *where* $d(v_i, v_i^j) < d(v_i, v_i^{j+1}) (1 \le j \le N-1)$, *if* v_i *sends a beacon packet* $Beacon(v_i^k)$ *to any DelayedReceiver* v_i^j *and sends the broadcasting message to the InstantReceiver* v_i^k $(1 \le j < k \le N)$, *then we have*

$$delay(v_i^j) - delay^*(v_i^j) = d(v_i^j, v_i^k) \tag{10}$$

Proof As v_i^j defers its message reception time to the *active slot* of v_i^k, it is obvious that $delay(v_i^j) = delay(v_i^k) = d(v_i, v_i^k)$ and $delay^*(v_i^j) = d(v_i, v_i^j)$. Since $j < k$ and $d(v_i, v_i^j) < d(v_i, v_i^k)$, we can find that the relationship between nodes' *active slots* must be one of the following three cases:

(1) If $0 \le t(v_i) < t(v_i^j) < t(v_i^k) \le L-1$, according to Eq. 1, we can have $delay(v_i^j) - delay^*(v_i^j) = d(v_i, v_i^k) - d(v_i, v_i^j) = t(v_i^k) - t(v_i) - (t(v_i^j) - t(v_i)) = t(v_i^k) - t(v_i^j) = d(v_i^j, v_i^k)$;

(2) If $0 \le t(v_i^j) < t(v_i^k) \le t(v_i) \le L-1$, according to Eq. 1, we can have $delay(v_i^j) - delay^*(v_i^j) = d(v_i, v_i^k) - d(v_i, v_i^j) = t(v_i^k) - t(v_i) + L - (t(v_i^j) - t(v_i) + L) = t(v_i^k) - t(v_i^j) = d(v_i^j, v_i^k)$;

(3) If $0 \le t(v_i^k) \le t(v_i) < t(v_i^j) \le L-1$, according to Eq. 1, we can have $delay(v_i^j) - delay^*(v_i^j) = d(v_i, v_i^k) - d(v_i, v_i^j) = t(v_i^k) - t(v_i) + L - (t(v_i^j) - t(v_i)) = t(v_i^k) - t(v_i^j) + L = d(v_i^j, v_i^k)$. Thus, the proof is completed.

□

According to Theorem 1, we can easily get the following corollary.

Corollary 1 *Given a single-hop network* $G_s(v_i, \{v_i^j, \ldots, v_i^k\})$ *where* $d(v_i, v_i^m) < d(v_i, v_i^{m+1}) (j \le m \le k-1)$, *we denote by* $cost(v_i^j, v_i^k)$ *the resulted broadcasting cost if* v_i *sends the beacon packet* $Beacon(v_i^k)$ *to the DelayedReceivers* $\{v_i^j, v_i^{j+1}, \ldots, v_i^{k-1}\}$ *(no beacon packet is sent if* $j = k$*) and sends the broadcasting message to the InstantReceiver* v_i^k, *then we have that*

$$\begin{aligned} cost(v_i^j, v_i^k) &= \sum_{m=j}^{k} (delay(v_i^m) - delay^*(v_i^m)) + \eta \\ &= D(v_i^j, v_i^k) + \eta \end{aligned} \tag{11}$$

where

$$D(v_i^j, v_i^k) = \begin{cases} \sum\limits_{m=j}^{k-1} d(v_i^m, v_i^k), & j < k; \\ 0, & j = k. \end{cases} \tag{12}$$

Let $OPT(k)$ denote the optimal broadcasting cost for the single-hop network $G_s(v_i, \{v_i^1, \ldots, v_i^k\})$ where $d(v_i, v_i^j) < d(v_i, v_i^{j+1}) (1 \le j \le k-1)$, we can have that

Theorem 2 *$OPT(k)$ has the property of optimal substructure.*

Proof We denote by $P(\{v_i^1, \ldots, v_i^k\})$ the problem that to find the optimal broadcasting schedule on a single-hop network topology $G = (V, E)$, which consists of the sender v_i and k re-marked receivers $\{v_i^1, \ldots, v_i^k\}$. Obviously, the last receiver v_i^k must be the *InstantReceiver* in the solution to $P(\{v_i^1, \ldots, v_i^k\})$. Suppose that M^* is the optimal broadcasting schedule of $P(\{v_i^1, \ldots, v_i^k\})$ and $M^*(v_i).\beta = < v_i^1, \ldots, \underline{v_i^j}, \ldots, \underline{v_i^k} >$, where any node $v_i^j (1 \le j < k)$ is assumed to be an *InstantReceiver* in the optimal solution. Let $cost(\{v_i^1, \ldots, v_i^j\})$ and $cost(\{v_i^{j+1}, \ldots, v_i^k\})$ respectively denote the resulted broadcasting cost for $P(\{v_i^1, \ldots, v_i^j\})$ and $P(\{v_i^{j+1}, \ldots, v_i^k\})$ when adopting the forwarding sequence in $M^*(v_i).\beta$, we can find that $OPT(k) = cost(\{v_i^1, \ldots, v_i^j\}) + cost(\{v_i^{j+1}, \ldots, v_i^k\})$. As $OPT(k)$ is the optimum to $P(\{v_i^1, \ldots, v_i^k\})$, then $cost(\{v_i^1, \ldots, v_i^j\})$ must equal to $OPT(j)$, which is the optimum to $P(\{v_i^1, \ldots, v_i^j\})$. This is because if $cost(\{v_i^1, \ldots, v_i^j\})$ does not equal to $OPT(j)$, there must exist another better solution to $P(\{v_i^1, \ldots, v_i^j\})$ so that M^* is not the optimal any more, which conflicts with our assumption. As $P(\{v_i^1, \ldots, v_i^j\}) (j < k)$ is the subproblem of $P(\{v_i^1, \ldots, v_i^k\})$, thus we can see that $OPT(k)$ has the property of optimal substructure. The proof is completed. □

According to Theorem 2, we can adopt a dynamic programming approach to solve the S-OMCB problem. Obviously, $OPT(k)$ must equal to the minimum of $\{cost(v_i^1, v_i^k), OPT(1) + cost(v_i^2, v_i^k), OPT(2) + cost(v_i^3, v_i^k), \ldots, OPT(k-1) + cost(v_i^k, v_i^k)\}$. In other words, we can have the following recurrence.

$$OPT(k) = \min_{1 \le j \le k} \{OPT(j-1) + D(v_i^j, v_i^k) + \eta\} \tag{13}$$

where $OPT(0) = 0$.

The S-OMCB problem thus turns to how to get $OPT(N)$ and the corresponding optimal broadcasting schedule. To solve this problem, we come up with an efficient *Opportunistic Single-hop Broadcasting Algorithm* (OSB-A), which is shown in Algorithm 1, and then we analyze its time complexity in Theorem 3.

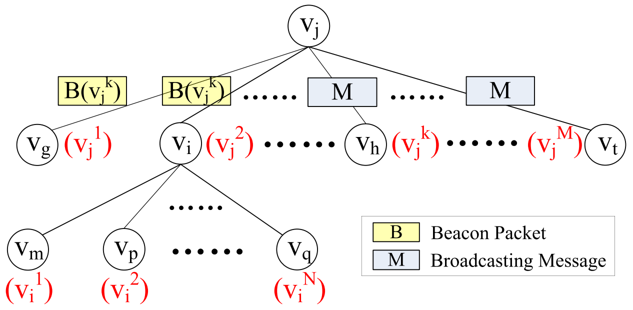

Fig. 4 Illustration of the constrained single-hop broadcasting in a multi-hop SPT

```
Algorithm 1: OSB-A
Input: The single-hop network $G_s(v_i,\{v_{j_1},\ldots,v_{j_N}\})$ and the working
       schedules of all nodes.
Output: The optimal broadcasting schedule $M$.
sort and re-mark all the receivers $\{v_{j_1},\ldots,v_{j_N}\}$ as $\{v_i^1,\ldots,v_i^N\}$, such
that $d(v_i,v_i^j) < d(v_i,v_i^{j+1})(1 \le j \le N-1)$ is satisfied;
$S_f(v_i) = <v_i^1,v_i^2,\ldots,v_i^N>$;
initially set all nodes in $S_f(v_i)$ as the DelayedReceivers;
$InstantReceiverCount = 0$;    $OPT[0] = 0$;
for $j = 1$ to $N$ do
    for $k = j$ to $N$ do
        compute $D(v_i^j,v_i^k)$ according to the calculation formula;
    end
end
for $k = 1$ to $N$ do
    $min\_cost = +\infty$;
    for $j = 0$ to $k-1$ do
        if $OPT[j] + D(v_i^{j+1},v_i^k) + \eta < min\_cost$ then
            $min\_cost = OPT[j] + D(v_i^{j+1},v_i^k) + \eta$;
            $s[k] = j$;
            /* $s[k] = j$ implies $OPT[k]$ is resulted from
            $OPT[j] + cost(v_i^{j+1},v_i^k)$. */
        end
    end
    $OPT[k] = min\_cost$;
end
$k = N$;
while $k \neq 0$ do
    set node $v_i^k$ in $S_f(v_i)$ as the InstantReceiver;
    $InstantReceiverCount = InstantReceiverCount + 1$;
    $k = s[k]$;
end
$M(v_i).\alpha = InstantReceiverCount$;
$M(v_i).\beta = S_f(v_i)$;
for $k = 1$ to $N$ do
    $M(v_i^k).\alpha = 0$;    $M(v_i^k).\beta = NULL$;
end
```

Theorem 3 *The time complexity of Algorithm* 1 is $O(N^3)$, *where* N *is the number of the receivers.*

Proof In Algorithm 1, we first need to compute all $D(v_i^j, v_i^k)$ $(j \le k)$ values. Note that, there are totally $O(N^2)$ pairs (v_i^j, v_i^k) where $j \le k$, and for each pair (v_i^j, v_i^k), we can employ Eq. 12 to compute $D(v_i^j, v_i^k)$ in $O(N)$ time. Thus, the total running time to compute all $D(v_i^j, v_i^k)(j \le k)$ values is $O(N^3)$. Next, this algorithm has N iterations to compute all $OPT[k](k \in \{1,\ldots,N\})$ values, and for each $OPT[k]$, it takes $O(N)$ time to determine the minimum in the Eq. 13. Thus, the total running time to compute all $OPT[k]$ $(k \in \{1, \ldots, N\})$ values is $O(N^2)$ once all $D(v_i^j, v_i^k)(j \le k)$ values have been determined. Also, we can easily find that it takes totally $O(N)$ time to determine the broadcasting schedule based on $OPT[k]$ and $s[k]$ $(k \in \{1,\ldots,N\})$. Obviously, the running time of Algorithm 1 is dominated by the $O(N^3)$ needed to compute all $D(v_i^j, v_i^k)(j \le k)$ values, thus the proof is completed. □

4.3 The multi-hop case

In the above subsection, we study the broadcasting problem for single-hop networks. However, the multi-hop broadcasting has a wider range of applications in practice. Here, we will extend OSB-A to the multi-hop case and propose an efficient *Opportunistic Multi-hop Broadcasting Algorithm* (OMB-A).

Before the statement of OMB-A, we first define a generalized version of the S-OMCB problem, which we call the *Constrained Single-hop based Opportunistic Minimum-Cost Broadcast Problem* (CS-OMCB).

Problem 3 (CS-OMCB) Given any local single-hop topology $G_s(v_i, \{v_i^1, \ldots, v_i^N\})$ in a multi-hop *Shortest Path Tree* (SPT) where $d(v_i, v_i^j) < d(v_i, v_i^{j+1})$ $(1 \le j \le N-1)$, if $delay(v_i) - delay^*(v_i)$ is determined, how to determine the optimal broadcasting schedule M in G_s, to minimize the following broadcasting cost function:

$$cost(G_s) = \Delta_{delay} + \eta \times M(v_i).\alpha \tag{14}$$

where $\Delta_{delay} = \sum_{k=1}^{N} (delay(v_i^k) - delay^*(v_i^k))$.

Essentially, the CS-OMCB problem is equivalent to the S-OMCB problem in a multi-hop context. Specially, it is found that the CS-OMCB problem will turn to be the S-OMCB problem if $delay(v_i) - delay^*(v_i) = 0$. Here, we will first propose the solution to the CS-OMCB problem, and then extend it to solve our target problem for multi-hop networks.

We can find that the proposed OSB-A can NOT be directly applied to solve the CS-OMCB problem, since for the forwarder v_i in G_s, the constrained condition that $delay(v_i) - delay^*(v_i) > 0$ could make the broadcasting schedule resulted by OSB-A NOT satisfy the constraint (3) of Definition 3. Figure 4 takes a simple example, where the re-marked notation of each receiver is shown within the pair of brackets, to illustrate the CS-OMCB problem. In Fig. 4, $\{v_j^1, \ldots, v_j^M\}$ are M re-marked receivers of the forwarder v_j where $d(v_j, v_j^k) < d(v_j, v_j^{k+1})$ $(1 \le k \le M-1)$, and $\{v_i^1, \ldots, v_i^N\}$ are N re-marked receivers of the forwarder v_i where $d(v_i, v_i^k) < d(v_i, v_i^{k+1})$ $(1 \le k \le N-1)$. We suppose that the forwarder v_j has already made the *forwarding decision*, in which the receiver v_j^2 (i.e., node v_i) defers its message reception time to the *active slot* of the receiver v_j^k. For the CS-OMCB problem, we mainly focus on how to determine the minimum-cost *forwarding sequence* of the forwarder v_i, given that the $delay(v_i) - delay^*(v_i)$ value is known. To satisfy the constraint (3) of Definition 3, we have to carefully design the forwarder v_i's *forwarding sequence*, of which determination is dependent on the $delay(v_i) - delay^*(v_i)$ value. Next, we will discuss the solution to the CS-OMCB problem, which is called the *Constrained Opportunistic Single-hop Broadcasting Algorithm* (COSB-A), in terms of the following three potential cases.

(1) *Case 1*: $delay(v_i) - delay^*(v_i) < d(v_i, v_i^1)$
If $delay(v_i) - delay^*(v_i) < d(v_i, v_i^1)$, it implies the determination of $M(v_i)$ must be independent of the $delay(v_i) - delay^*(v_i)$ value. For this case, we can find that the CS-OMCB problem will turn to be the S-OMCB problem, and thus we can directly use OSB-A to solve it.

(2) *Case 2*: $delay(v_i) - delay^*(v_i) \geq d(v_i, v_i^N)$
If $delay(v_i) - delay^*(v_i) \geq d(v_i, v_i^N)$, it implies any feasible solution M to the S-OMCB problem must NOT be a feasible solution to the CS-OMCB problem, this is because the condition $d(v_i, v_i^N) \leq delay(v_i) - delay^*(v_i)$ must make M NOT satisfy the constraint (3) of Definition 3. For this case, we can easily find the optimal solution, that is, to let the forwarder v_i send $Beacon(v_i^1)$ to all the *DelayedReceivers* $\{v_i^1, \ldots, v_i^N\}$ and then send the broadcasting message to the *InstantReceiver* v_i^1 at the next *active slot* of v_i^1, so that all the receivers will defer their message reception time to the *active slot* of node v_i^1's next round working schedule period. Note that, the receiver v_i^1 is both the *DelayedReceiver* and the *InstantReceiver* in this case.

(3) *Case 3*: $d(v_i, v_i^1) \leq delay(v_i) - delay^*(v_i) < d(v_i, v_i^N)$
If $d(v_i, v_i^1) \leq delay(v_i) - delay^*(v_i) < d(v_i, v_i^N)$, it implies any receiver v_i^k with $d(v_i, v_i^k) \leq delay(v_i) - delay^*(v_i)$ must be a *DelayedReceiver* in $S_f(v_i)$, in other words, the sleep latency from the forwarder v_i to the first *InstantReceiver* in $S_f(v_i)$ must be larger than $delay(v_i) - delay^*(v_i)$. Here, we let $DS(v_i)$ denote a set of the predetermined *DelayedReceivers* in $S_f(v_i)$, namely

$$DS(v_i) = \{v_i^k | d(v_i, v_i^k) \leq delay(v_i) - delay^*(v_i)\} \tag{15}$$

For this case, our target is thus to find a minimum-cost broadcasting schedule, given the constraint that all nodes in $DS(v_i)$ must be the *DelayedReceivers*. For the solution to our target problem, note that all nodes in $DS(v_i)$ have been determined as the *DelayedReceivers*, thus we only need to focus on how to determine v_i's *forwarding sequence* in $\{v_i^{K+1}, \ldots, v_i^N\}$ where $K = |DS(v_i)|$.

Obviously, the CS-OMCB problem for this case can be transformed into the following equivalent problem: *Given any local single-hop topology* $G_s(v_i, \{v_i^1, \ldots, v_i^N\})$ *in a multi-hop SPT where* $d(v_i, v_i^j) < d(v_i, v_i^{j+1}) (1 \leq j \leq N-1)$, *and its subgraph* $G_s'(v_i, \{v_i^{K+1}, \ldots, v_i^N\})$ *where* $K = |DS(v_i)|$, *how to determine the optimal broadcasting schedule M in the subgraph* G_s', *to minimize the following broadcasting cost function* $cost(G_s)$:

$$cost(G_s) = cost(G_s') + \sum_{v \in DS(v_i)} d(v, v_i^*) \tag{16}$$

where v_i^* *denotes the first InstantReceiver in* $M(v_i).\beta$, *and*

$$cost(G_s') = \sum_{j=K+1}^{N} (delay(v_i^j) - delay^*(v_i^j)) + \eta \times M(v_i).\alpha.$$

For this case, we redefine $OPT(k)$ as the optimal broadcasting cost for the above-mentioned transformed equivalent problem where $N = K + k$. Accordingly, our target is to get $OPT(N-K)$ and the corresponding optimal broadcasting schedule, based on the following recurrence:

$$OPT(k) = \min_{1 \leq j \leq k} \{OPT(j-1) + D(v_i^{K+j}, v_i^{K+k}) + \eta\} \tag{17}$$

where $OPT(0) = 0$, and

$$D(v_i^{K+j}, v_i^{K+k}) = \begin{cases} \sum_{m=1}^{k-1} d(v_i^{K+m}, v_i^{K+k}) + \sum_{v \in DS(v_i)} d(v, v_i^{K+k}), & 1 = j < k; \\ \sum_{m=j}^{k-1} d(v_i^{K+m}, v_i^{K+k}), & 1 < j < k; \\ \sum_{v \in DS(v_i)} d(v, v_i^{K+1}), & 1 = j = k; \\ 0, & 1 < j = k. \end{cases} \tag{18}$$

Therefore, we can solve the CS-OMCB problem for this case, by directly adopting OSB-A on G_s' to get $OPT(N-K)$ and the corresponding optimal broadcasting schedule based on Eq. 17. The detailed statement of COSB-A is shown in Algorithm 2.

Algorithm 2: COSB-A

```
Input: The local single-hop network topology $G_s(v_i, \{v_{j_1}, \ldots, v_{j_N}\})$,
       the working schedules of all nodes and $delay(v_i) - delay^*(v_i)$.
Output: The optimal broadcasting schedule $M$.
sort and re-mark all the receivers $\{v_{j_1}, \ldots, v_{j_N}\}$ as $\{v_i^1, \ldots, v_i^N\}$, such
that $d(v_i, v_i^j) < d(v_i, v_i^{j+1})(1 \le j \le N-1)$ is satisfied;
$\Delta delay(v_i) = delay(v_i) - delay^*(v_i)$;
if $\Delta delay(v_i) < d(v_i, v_i^1)$ then
    set Equation 12 as the calculation formula of $D(v_i^j, v_i^k)$;
    call OSB-A on $G'$;
end
if $\Delta delay(v_i) \ge d(v_i, v_i^N)$ then
    $S_f(v_i) =< v_i^1, v_i^2, \ldots, v_i^N, \underline{v_i^1} >$;
    $M(v_i).\alpha = 1$;   $M(v_i).\beta = S_f(v_i)$;
    for $k = 1$ to $N$ do
        $M(v_i^k).\alpha = 0$;   $M(v_i^k).\beta = NULL$;
    end
end
if $\Delta delay(v_i) \ge d(v_i, v_i^1)$ && $\Delta delay(v_i) < d(v_i, v_i^N)$ then
    get $DS(v_i)$ according to Equation 15;   $K = |DS(v_i)|$;
    set Equation 18 as the calculation formula of $D(v_i^j, v_i^k)$;
    call OSB-A on $G'_s(v_i, \{v_i^{K+1}, \ldots, v_i^N\})$;
    $S'_f(v_i) =< v_i^1, \ldots, v_i^K >$;
    set all nodes in $S'_f(v_i)$ as the DelayedReceivers;
    update $M(v_i).\beta$ by adding $S'_f(v_i)$ to the beginning of $M(v_i).\beta$;
    for $k = 1$ to $K$ do
        $M(v_i^k).\alpha = 0$;   $M(v_i^k).\beta = NULL$;
    end
end
```

Given a multi-hop sensor network $G = (V, E)$, we can easily find the *Shortest Path Fat Tree* (SPFT) in G by adopting a simple Bellman-Ford-like algorithm. For any node v_i in SPFT, we use $CPS(v_i)$ and $CCS(v_i)$ to denote its *Candidate Parents Set* and *Candidate Children Set*, respectively. Specifically, if any node $v_j \in CCS(v_i)$, it implies node v_i must exist in some shortest E2E delay path from the sink to node v_j. Also, if any node $v_j \in CPS(v_i)$, it implies node v_j must exist in some shortest E2E delay path from the sink to node v_i. Obviously, a *Shortest Path Tree* (SPT) will be constructed if any sensing node v_i in SPFT selects any node $v_j \in CPS(v_i)$ as its parent.

Based on the SPFT, we will propose an efficient *Opportunistic Multi-hop Broadcasting Algorithm* (OMB-A), which is the extension of COSB-A, to solve the OMCB problem for multi-hop networks. The basic idea of OMB-A can be described as follows:

Initially, we mark the sink v_0 with *covered state*, mark all the sensing nodes with *uncovered states*, define $delay(v_0) - delay^*(v_0) = 0$ and $CFS = \{v_0\}$ where CFS denotes the *Candidate Forwarders Set*. For any candidate forwarder $v_i \in CFS$, we define the following *Competition Factor* (CF):

$$CF(v_i) = \frac{cost^*(G_s(v_i, CCS(v_i)))}{|CCS(v_i)|} \tag{19}$$

where $cost^*(G_s(v_i, CCS(v_i)))$ denotes the resulted optimal broadcasting cost if COSB-A is adopted on $G_s(v_i, CCS(v_i))$. Obviously, the forwarder with the least CF value will be preferred, since the greedy strategy that locally takes less cost to cover more sensing nodes each time could intuitively result in a lower total broadcasting cost of the network. Thus, we will select the forwarder $v_* \in CFS$, which has the least CF value in CFS, as the competition winner, and then execute COSB-A on the local single-hop topology $G_s(v_*, CCS(v_*))$. For any node $v_i \in CCS(v_*)$, we will mark it with *covered state* and add it into CFS, then remove it from $CCS(v_j)$ for any $v_j \in CPS(v_i)$. Afterwards, we remove any node v with $CCS(v) = \emptyset$ from CFS. The above process is repeated until $CFS = \emptyset$ (i.e., all nodes in V are marked with *covered states*). Note that, any sensing node v_i will be aware of the $delay(v_i) - delay^*(v_i)$ value once it is marked with *covered state*. Algorithm 3 shows the detailed process of OMB-A.

Algorithm 3: OMB-A

```
Input: The multi-hop network topology $G = (V, E)$ and the working
       schedules of all nodes.
Output: The optimal broadcasting schedule $M$.
construct the SPFT on $G$, then get $CPS(v)$ and $CCS(v)$ for any $v \in V$;
mark the sink $v_0$ with covered state;
mark all the sensing nodes with uncovered states;
$CFS = \{v_0\}$;
while $CFS \neq \emptyset$ do
    compute $CF(v)$ according to Equation 19 for any node $v \in CFS$;
    $v_* = \arg\min_{v \in CFS} CF(v)$;
    compute $delay(v_*) - delay^*(v_*)$;
    call COSB-A on $G_s(v_*, CCS(v_*))$;
    for each node $v_i \in CCS(v_*)$ do
        mark $v_i$ with covered state;
        $CFS = CFS \bigcup \{v_i\}$;
        for each node $v_j \in CPS(v_i)$ do
            $CCS(v_j) = CCS(v_j) - \{v_i\}$;
        end
    end
    for each node $v \in CFS$ do
        if $CCS(v) = \emptyset$ then
            $CFS = CFS - \{v\}$;
        end
    end
end
```

4.4 Discussion

Note that we assumed the working schedules of neighboring nodes are different from each other, which is commonly seen in low-duty-cycle WSNs. However, our solution can be also extended to the generalized case, where a few of the neighboring nodes could have the identical wake-up schedule, by simply regarding the set of neighbors having identical wake-up time slot as one *virtual node*.

For example, given a single-hop topology $G_s(v_i, \{v_{j_1}, \ldots, v_{j_6}\})$ in which $d(v_i, v_{j_1}) < d(v_i, v_{j_2}) = d(v_i, v_{j_3}) = d(v_i, v_{j_4}) < d(v_i, v_{j_5}) = d(v_i, v_{j_6})$, the initial *forwarding sequence* of the sender v_i can be represented as follows.

$$S_f(v_i) = < v_i^1, v_i^2, v_i^3 > \tag{20}$$

where $v_i^1 = \{v_{j_1}\}$, $v_i^2 = \{v_{j_2}, v_{j_3}, v_{j_4}\}$, $v_i^3 = \{v_{j_5}, v_{j_6}\}$. Here, each v_i^j $(j \in \{1, 2, 3\})$ denotes a *virtual node*, we can define that $t(v_i^1) = t(v_{j_1})$, $t(v_i^2) = t(v_{j_2}) = t(v_{j_3}) = t(v_{j_4})$ and $t(v_i^3) = t(v_{j_5}) = t(v_{j_6})$. Further, a *virtual node* is called the *DelayedReceiver* (*InstantReceiver*) if and only if all sensor nodes in this *virtual node* are the *DelayedReceivers* (*InstantReceivers*). Note that, any *InstantReceiver virtual node* in $S_f(v_i)$ represents one broadcasting message transmission, and for $1 < j < k$, $D(v_i^j, v_i^k)$ will be represented by

$$D(v_i^j, v_i^k) = \sum_{m=j}^{k-1} \sum_{v \in v_i^m} d(v, v_i^k) = \sum_{m=j}^{k-1} (|v_i^m| \times d(v_i^m, v_i^k)) \tag{21}$$

where $|v_i^m|$ denotes the number of sensor nodes in *virtual node* v_i^m.

5 Performance evaluation

In this section, we evaluate the performance of our solution via simulations.

In our setting, we consider that *Num* sensor nodes are uniformly distributed in a 100 m×100 m sensory field, where the sink node is located at the center, i.e., (50, 50 m). For simplicity, we assume that each period of any node's working schedule consists of one *active slot* and $L - 1$ *sleeping slots*, and each node randomly and independently determines its own working schedule. Here, we assume that all nodes have the same communication range r_c and the *disk communication model* is adopted, i.e., any node v_i can deliver a packet to any node v_j if and only if node v_j is located within the communication range of node v_i. Further, we adopt the following classic energy consumption model which is commonly used in many existing literature:

$$e_s(l) = l \cdot E_{elec} + l \cdot \varepsilon_{amp} r_c^2, \quad e_r(l) = l \cdot E_{elec}, \tag{22}$$

where $E_{elec} = 50nJ/bit$, $\varepsilon_{amp} = 100pJ/bit/m^2$, l denotes the packet length, $e_s(l)$ and $e_r(l)$ denote the energy consumed by sending a packet and receiving a packet, respectively. As the same with the literature [53], we define that each data packet and each beacon packet have a length of 133 bytes and 19 bytes, respectively. Unless otherwise stated, we set $Num = 800$, $L = 100$, $r_c = 20$ m, $\eta = 100$. In this paper, we develop a simulator using Java, to evaluate the performance of our solution, and all the simulation results are generated by averaging over 20 times.

Here, we take the following five heuristic approaches as the baselines to evaluate the performance of our solution.

- **SPT-based delay-first**: This approach adopts a delay-first strategy where no deferring strategy is employed by each node. It first constructs a SPT over the network topology, then the sink node broadcasts the message directly along with the SPT. In this approach, all sensing nodes are the *InstantReceivers*.
- **SPT-based energy-first**: This approach adopts an energy-first strategy where each forwarder only sets exactly one of its receivers as the *InstantReceiver*. It first constructs a SPT over the network topology, then for any local single-hop topology $G_s(v_i, \{v_i^1, \ldots, v_i^N\})$ on SPT where $d(v_i, v_i^j) < d(v_i, v_i^{j+1})$ $(1 \leq j \leq N - 1)$, if $delay(v_i) - delay^*(v_i) \geq d(v_i, v_i^N)$, then the *forwarding sequence* of the forwarder v_i will be determined as $S_f(v_i) = < v_i^1, v_i^2, \ldots, v_i^N, \underline{v_i^1} >$; otherwise, the *forwarding sequence* of the forwarder v_i will be determined as $S_f(v_i) = < v_i^1, v_i^2, \ldots, \underline{v_i^N} >$. In other words, only the last receiver in the *forwarding sequence* of the forwarder will be determined as the *InstantReceiver* and the others will be determined as the *DelayedReceivers*. The broadcasting schedule of the network will be found by successively determining the *forwarding sequence* of each forwarder on SPT in a top-down order.
- **SPT-based COSB-A**: This approach first constructs a SPT over the network topology, then for any local single-hop topology on SPT, the forwarder will determine its *forwarding sequence* by adopting COSB-A. The broadcasting schedule of the network will be found by successively determining the *forwarding sequence* of each forwarder on SPT in a top-down order.
- **Unstructured delay-first**: This approach is similar to OMB-A, the difference is that the forwarder for any local single-hop broadcasting will adopt the delay-first strategy, instead of COSB-A. Specifically, it will redefine $cost^*(G_s(v_i, CCS(v_i)))$ in OMB-A as the resulted broadcasting cost on $G_s(v_i, CCS(v_i))$ if v_i directly sends the broadcasting message to each node in $CCS(v_i)$. Once the competition winner v_* is determined in each round, v_* will directly send the broadcasting message to each node in $CCS(v_*)$, i.e., initially determine its *forwarding sequence* $S_f(v_*)$ based on $CCS(v_*)$ and then mark all nodes in $S_f(v_*)$ with the *InstantReceivers*.
- **Unstructured energy-first**: This approach is similar to OMB-A, the difference is that the forwarder for any

local single-hop broadcasting will adopt the energy-first strategy, instead of COSB-A. Specifically, it will redefine $cost^*(G_s(v_i, CCS(v_i)))$ in OMB-A as the resulted broadcasting cost on $G_s(v_i, CCS(v_i))$ if v_i only sets exactly one of its receivers in $CCS(v_i)$ as the *InstantReceiver*. Suppose that $CCS(v_i)$ consists of $\{v_i^1, \ldots, v_i^N\}$ where $d(v_i, v_i^j) < d(v_i, v_i^{j+1})$ $(1 \le j \le N-1)$. Once the competition winner v_* is determined in each round, v_* will initially determine its *forwarding sequence* $S_f(v_*)$ based on $CCS(v_*)$ and $delay(v_*) - delay^*(v_*)$, if it is found that $delay(v_*) - delay^*(v_*) \ge d(v_*, v_*^N)$, then the *forwarding sequence* of the forwarder v_* will be determined as $S_f(v_*) = <v_*^1, v_*^2, \ldots, v_*^N, \underline{v_*^1}>$; otherwise, the *forwarding sequence* of the forwarder v_* will be determined as $S_f(v_*) = <v_*^1, v_*^2, \ldots, \underline{v_*^N}>$. In other words, only the last receiver in the *forwarding sequence* of the forwarder will be determined as the *InstantReceiver* and the others will be determined as the *DelayedReceivers*.

First, we will compare our proposed OMB-A with the above-mentioned baselines under various performance requirements. Figure 5 and Table 2 show the performance comparison between various solutions in terms of the broadcasting cost, when the tradeoff factor η varies between 0 and 1. We can easily find that if $0 < \eta < 1$, our proposed OMB-A always gains nearly the same performance with the *unstructured delay-first* solution and outperforms the other solutions, this is because the tradeoff factor η is so small that the delay performance will dominate the broadcasting cost, it implies the COSB-A adopted at each round of competition winner selection in OMB-A will be approximately equivalent to the delay-first strategy. For the solutions with the same single-hop broadcast scheduling strategy, we can see that the unstructured solution always outperform the SPT-based one, since the unstructured solutions have a better flexibility on the forwarders selection compared with the SPT-based ones, where the forwarders are fixed and determined by the construction of SPT. Essentially, our proposed OMB-A can be regarded as the unstructured COSB-A. Table 2 shows the comparison result between OMB-A and the solutions with the energy-first strategy for the cases with $0 < \eta < 1$. Obviously, the solutions with the energy-first strategy must exhibit much worse performance than the other solutions, in that the broadcasting cost is totally dominated by delay performance for the cases with $0 < \eta < 1$.

As shown in Fig. 6 where the tradeoff factor η varies between 1 and 20, we can find that our proposed OMB-A still has a better performance than the other solutions. When $1 \le \eta \le 5$, our proposed OMB-A has nearly the same performance with the *unstructured delay-first* solution,

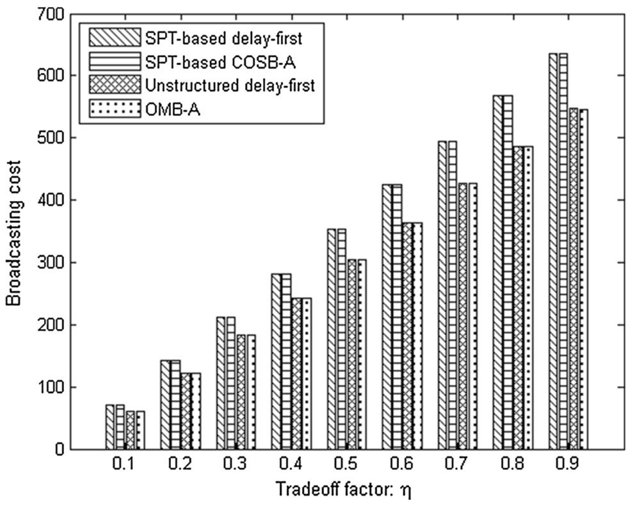

Fig. 5 Performance comparison when η varies between 0 and 1

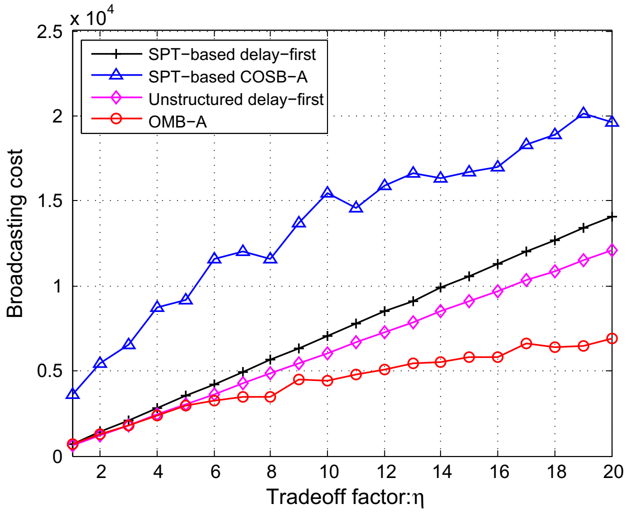

Fig. 6 Performance comparison when η varies between 1 and 20

Table 2 Comparison on the broadcasting cost when η varies between 0 and 1

	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.3$	$\eta = 0.4$	$\eta = 0.5$	$\eta = 0.6$	$\eta = 0.7$	$\eta = 0.8$	$\eta = 0.9$
SPT-based energy-first	48962.1	49283.9	49100.6	48859.8	49210.6	49850.4	48255.9	47847.9	48300.3
Unstructured energy-first	38467.3	43080.1	42327.2	40988.1	43475.7	42543.4	43084.4	40780.8	42594.7
OMB-A	60.7	121.7	182.7	241.9	303.6	362.8	426.7	486.4	546.2

which implies the broadcasting cost is still dominated by delay performance even for the cases with $1 \leq \eta \leq 5$. When $\eta > 5$, our proposed OMB-A outperforms the *unstructured delay-first* solution. Specifically, our proposed OMB-A will have a larger performance advantage over the *unstructured delay-first* solution as the tradeoff factor η increases, this is because the increase of η makes the broadcasting cost no longer be dominated by delay performance but have more dependence on energy performance, which implies the COSB-A adopted in OMB-A, that considers both delay and energy performance, must have a better performance than the delay-first strategy that totally neglects energy performance. Also, Table 3 shows the comparison result between OMB-A and the solutions with the energy-first strategy for the cases with $1 < \eta < 20$, we can find that OMB-A still gains a much better performance than the solutions with the energy-first strategy even for the cases with $1 < \eta < 20$, where the broadcasting cost is not totally dominated by delay performance but has relatively less dependence on energy performance.

Figure 7 shows the comparison result between various solutions when the tradeoff factor η varies between 20 and 400. We can see that even for the cases with large η, our proposed OMB-A still has the best performance over all the solutions. As η increases, the performance of OMB-A will gradually approach that of the *unstructured energy-first* solution, and also the performance of the *SPT-based COSB-A* solution will gradually approach that of the *SPT-based energy-first* solution, since the increase of η will make energy performance gradually dominate the broadcasting cost. When η is large enough such that the broadcasting cost is totally dominated by energy performance, our proposed OMB-A will eventually have almost the same performance with the *unstructured energy-first* solution. According to Fig. 7, we can easily find that for the SPT-based solutions, the cut-off point is approximately 100, i.e., when $\eta < 100$, the *SPT-based delay-first* solution will have a better performance that the *SPT-based energy-first* solution, and when $\eta \geq 100$, the former will have a worse performance that the latter. For the unstructured solutions, we can also find the cut-off point is approximately 80. In Fig. 7, we notice that the *SPT-based COSB-A* solution exhibits a better performance than the *SPT-based delay-first* solution, this is different from both the case in Fig. 5 where they have nearly the same performance, and the case in Fig. 6 where the *SPT-based COSB-A* solution has a worse performance than the *SPT-based delay-first* solution. This is because when η is so small ($0 < \eta < 1$) that the broadcasting cost is mainly dominated by delay performance, the COSB-A adopted for each local single-hop topology in SPT is approximately equivalent to the delay-first strategy. As η increases ($1 \leq \eta \leq 20$), the broadcasting cost is not totally dominated by delay performance but has relatively less dependence on energy performance, the delay-first strategy would still be the approximately optimal strategy for most, but not all, of the local single-hop topologies in SPT, this implies for all the local single-hop topologies in SPT, the delay-first strategy would result in a better performance than the COSB-A, the latter could make the broadcast problems for many local single-hop topologies where the delay-first strategy is the approximately optimal strategy become the constrained broadcast problems. When η is large ($\eta > 20$), the broadcasting cost has more dependence on energy performance, the delay-first strategy would NOT be the approximately optimal strategy for most of the local single-hop topologies in SPT, this implies the COSB-A that considers both delay and energy performance could have a better performance than the delay-first strategy that totally neglects energy performance.

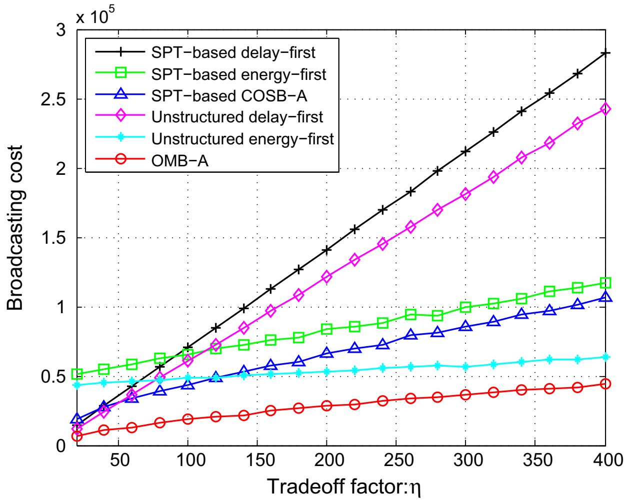

Fig. 7 Performance comparison when η varies between 20 and 400

Next, we will investigate the impact of the other factors (*e.g.*, *Num*, *L* and r_c) on the performance. The comparison

Table 3 Comparison on the broadcasting cost when η varies between 1 and 20

	$\eta = 2$	$\eta = 4$	$\eta = 6$	$\eta = 8$	$\eta = 10$	$\eta = 12$	$\eta = 14$	$\eta = 16$	$\eta = 18$
SPT-based energy-first	48468.9	49300.8	51351	49381.5	51967.4	49591.6	52057.7	51875.2	51527.3
Unstructured energy-first	40728.6	40633.9	42794.9	43454.9	43610.8	43520.4	43486.5	42826.7	42857.2
OMB-A	1292	2384.5	3237.4	3496.1	4457.1	5110.6	5544.3	5790.1	6368.3

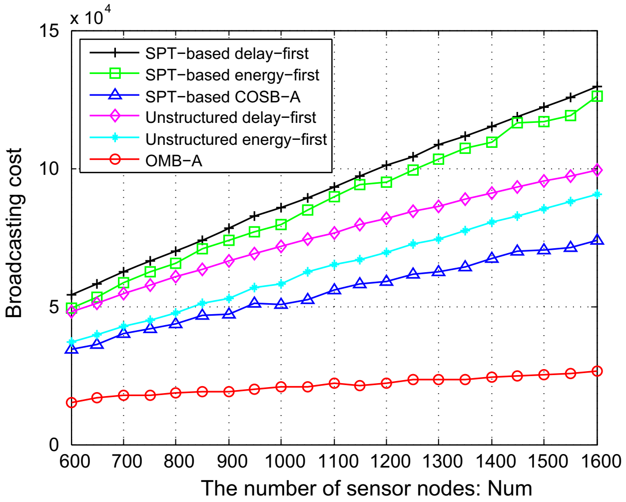

Fig. 8 Performance comparison when *Num* varies between 600 and 1600

result between various solutions under different number of sensor nodes is shown in Fig. 8 where $\eta = 100$. We can find that our proposed OMB-A has a significant performance advantage over the other solutions no matter how *Num* varies, and such advantage will get larger as the network density increases. Compared with the other solutions, note that, our proposed OMB-A will NOT experience a notable increase on the broadcasting cost as *Num* increases.

Figure 9 exhibits the impact of duty cycle on the performance. For all the solutions, we find that the broadcasting cost will decrease as L decreases (i.e., the duty cycle increases), this is because the case with lower L could make more *virtual nodes* that consist of multiple sensor nodes with the identical working schedule be generated. Obviously, if any of these *virtual nodes* is marked with the *InstantReceiver*, the forwarder would only need to send one message to cover multiple receivers and the number of the *DelayedReceivers* would be reduced, the broadcasting cost could thus be reduced. Also, the lower L results in a smaller value range of *sleep latency*, which would further reduce the broadcasting cost. As shown in Fig. 9, our proposed OMB-A always has a better performance than the other solutions under whatever duty cycle, and also it has a larger performance advantage over the other solutions as L increases (i.e., the duty cycle decreases), that is, our solution will perform better for low-duty-cycle networks. For the solutions with the delay-first strategy, we can find that the broadcasting cost will gradually turn to be stable as L increases, since for any local single-hop topology, the broadcasting cost resulted from the delay-first strategy mainly depends on the number of *virtual nodes*, which will converge to the number of all the receivers as L increases. For the solutions with the energy-first strategy, we can find that the broadcasting cost is almost growing linearly as L increases, since for any local single-hop topology, the broadcasting cost resulted from the energy-first strategy mainly depends on the average of point-to-point sleep latency, which will grow linearly as L increases.

Further, we compare the performance between various solutions when the communication range r_c varies. The comparison result is shown in Fig. 10. Generally speaking, all the solutions will gain a lower broadcasting cost as r_c increases, this is because the increase of the communication range could reduce the number of local single-hop broadcasting in the network and also have more possibility that multiple receivers have the identical working schedule in each local single-hop topology. In Fig. 10, we can find that our proposed OMB-A always performs better than the other solutions no matter how r_c varies. Note that, the

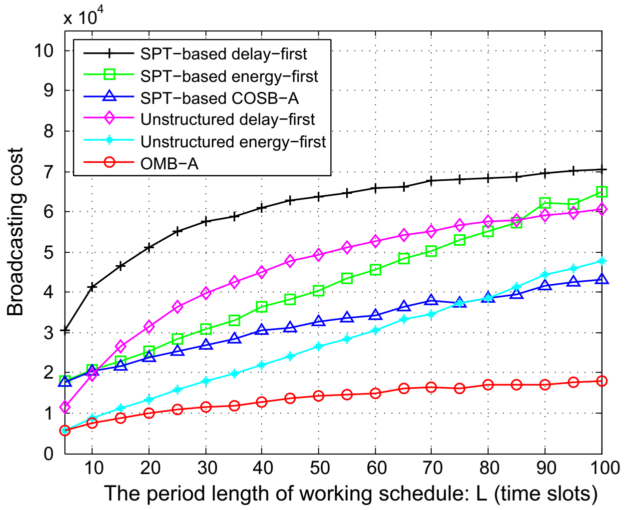

Fig. 9 Performance comparison when L varies between 5 and 100

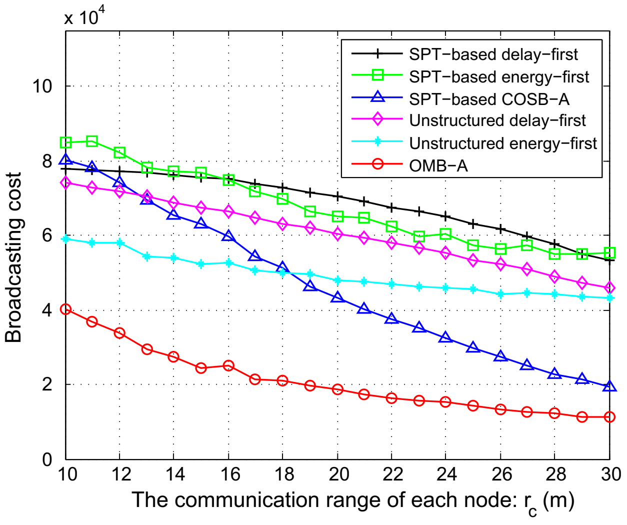

Fig. 10 Performance comparison when r_c varies between 10 and 30

increase of the communication range will make the performance of the *SPT-based COSB-A* gradually approach that of the OMB-A. Specially, when r_c is large enough such that the sink can directly reach each node by single hop, our proposed OMB-A and the *SPT-based COSB-A* solution will both become the theoretically optimal solution.

Accordingly, we can conclude that for the opportunistic minimum-cost broadcast problem, our proposed OMB-A always outperforms the other solutions under whatever configurations.

In this paper, we utilize Eq. 8 to define the broadcasting cost function. In Eq. 8, specifically, we simply employ the sum of the incremental E2E delay for all nodes (i.e., Δ_{delay}) and the total number of the broadcasting message transmissions (i.e., $\sum_{v_i \in V} M(v_i).\alpha$) to define DPI and EPI, respectively. Here, we further validate the effectiveness of such definition. Figures 11 and 12 show the relationship between the tradeoff factor η and broadcasting performance (total energy consumption and average broadcasting delay) when our proposed OMB-A is adopted. We can find that when η increases, the total energy consumption will generally decrease and the average broadcasting delay will generally increase. This implies our proposed simplified definition of broadcasting cost function (Eq. 8) can characterize the tradeoff between total energy consumption and average broadcasting delay. Therefore, any value of the tradeoff factor η in Eq. 8 can essentially characterize a certain broadcasting performance requirement. In other words, any tradeoff relationship between Δ_{delay} and $\sum_{v_i \in V} M(v_i).\alpha$ can be actually regarded as another tradeoff relationship between average broadcasting delay and total energy consumption for broadcasting. This implies that in practice, we can just set the value of η in Eq. 8 to generally and indirectly characterize a certain tradeoff relationship between average broadcasting delay and total energy consumption for broadcasting.

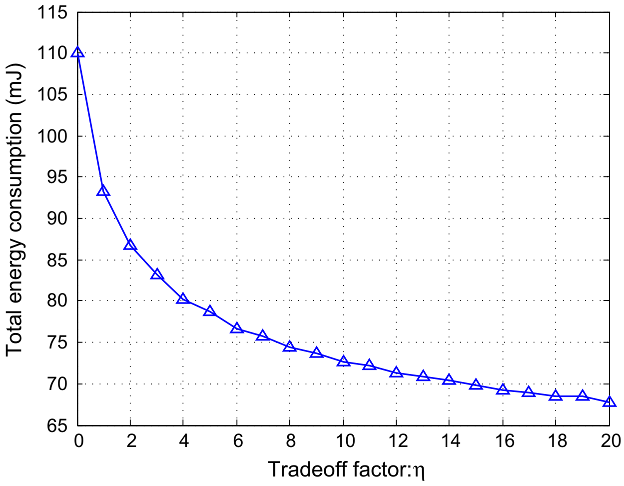

Fig. 11 Total energy consumption when η varies between 0 and 20

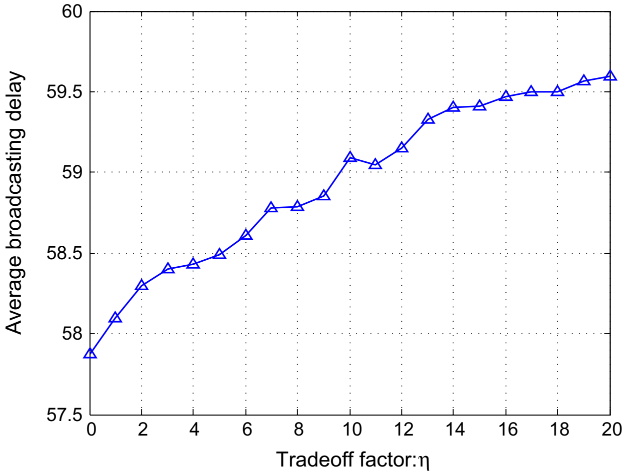

Fig. 12 Average broadcasting delay when η varies between 0 and 20

6 Conclusion and future work

In this paper, we present a novel opportunistic broadcasting transmission model, and consider how to make full use of such model to address the broadcast problem for low-duty-cycle WSNs. Different from the traditional studies that typically regard delay or energy as the single optimization objective, we define a generalized broadcasting cost function, which can provide an adaptive control on the tradeoff between average broadcasting delay and total energy consumption of broadcasting to meet various performance requirements. Our target is to find an efficient broadcasting schedule to minimize such broadcasting cost function, so that the specific performance requirement is achieved. Based on the opportunistic broadcasting transmission model, we first address our target problem under the single-hop case, by adopting a dynamic programming approach that can be completed in polynomial time. Then, we extend it to the multi-hop case, and come up with an efficient *Opportunistic Multi-hop Broadcasting Algorithm* (OMB-A). Extensive simulation results have verified the high efficiency of our proposed OMB-A compared with the other solutions.

To the best of our knowledge, this paper is the first work that utilizing the broadcasting spatiotemporal locality to address the tradeoff problem between delay and energy for broadcasting in low-duty-cycle sensor networks. However, it still leaves something to be desired, for example, how to extend it to the networks with unreliable links. In practice, many WSNs are usually deployed in a tough environment with lossy links. The combination of low-duty-cycle operation and the unreliability of links will further exacerbate the inefficiency of broadcasting, which implies our target problem under the unreliable networks has become a challenging issue. For low-duty-cycle WSNs with unreliable links, thus, how to carefully joint the opportunistic broadcasting transmission model and link correlation to design efficient broadcasting algorithms will be the main concern of our future work.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61502251, 61572263, 61472193), China Postdoctoral Science Foundation Funded Project (No. 2016M601859), Natural Science Foundation of Jiangsu Province (No. BK20141429), NUPTSF (Grant No. NY214169), PAPD and CICAEET.

References

1. Liu, Y., He, Y., Li, M., Wang, J., Liu, K., & Li, X. (2013). Does wireless sensor network scale? A measurement study on GreenOrbs. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, *24*(10), 1983–1993.
2. Liu, Y., Mao, S., He, Y., Liu, K., & Wang, J. (2013). CitySee: Not only a wireless sensor network. *IEEE Network*, *27*, 42–47.
3. Li, M., & Liu, Y. (2009). Underground coal mine monitoring with wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, *5*(2), 1–29. Article No. 10.
4. Li, M., Yang, Z., & Liu, Y. (2013). Sea depth measurement with restricted floating sensors. *ACM Transactions on Embedded Computing Systems (TECS)*, *13*(1), 1–21. Article No. 1.
5. Pan, Z., Lei, J., Zhang, Y., Sun, X., & Kwong, S. (2016). Fast motion estimation based on content property for low-complexity H.265/HEVC encoder. *IEEE Transactions on Broadcasting*, *62*(3), 675–684.
6. Pan, Z., Jin, P., Lei, J., Zhang, Y., Sun, X., & Kwong, S. (2016). Fast reference frame selection based on content similarity for low complexity HEVC encoder. *Journal of Visual Communication and Image Representation, 40*(Part B), 516–524.
7. Pan, Z., Zhang, Y., & Kwong, S. (2015). Efficient motion and disparity estimation optimization for low complexity multiview video coding. *IEEE Transactions on Broadcasting*, *61*(2), 166–176.
8. Chen, Y., Hao, C., Wu, W., & Wu, E. (2016). Robust dense reconstruction by range merging based on confidence estimation. *Science China Information Sciences*, *59*(9), 1–11.
9. Huang, J., Qian, F., Gerber, A., Mao, Z. M., Sen, S., & Spatscheck, O. (2012). A close examination of performance and power characteristics of 4G LTE networks. In *Proceedings of the 10th ACM international conference on mobile systems, applications, and services* (pp. 225–238).
10. Fu, Z., Ren, K., Shu, J., Sun, X., & Huang, F. (2016). Enabling personalized search over encrypted outsourced data with efficiency improvement. *IEEE Transactions on Parallel and Distributed Systems*, *27*(9), 2546–2559.
11. Xia, Z., Wang, X., Sun, X., & Wang, Q. (2015). A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data. *IEEE Transactions on Parallel and Distributed Systems*, *27*(2), 340–352.
12. Kong, Y., Zhang, M., & Ye, D. (2016). A belief propagation-based method for task allocation in open and dynamic cloud environments. *Knowledge-based Systems*, *115*, 123–132.
13. Liu, Q., Cai, W., Shen, J., Fu, Z., Liu, X., & Linge, N. (2016). A speculative approach to spatial-temporal efficiency with multi-objective optimization in a heterogeneous cloud environment. *Security and Communication Networks*, *9*(17), 4002–4012.
14. Fu, Z., Sun, X., Liu, Q., Zhou, L., & Shu, J. (2015). Achieving efficient cloud search services: Multi-keyword ranked search over encrypted cloud data supporting parallel computing. *IEICE Transactions on Communications*, *E98–B*(1), 190–200.
15. Shen, J., Tan, H., Wang, J., Wang, J., & Lee, S. (2015). A novel routing protocol providing good transmission reliability in underwater sensor networks. *Journal of Internet Technology*, *16*(1), 171–178.
16. Xiao, F., Xie, X., Jiang, Z., Sun, L., & Wang, R. (2016). Utility-aware data transmission scheme for delay tolerant networks. *Peer-to-Peer Networking and Applications*, *9*(5), 936–944.
17. Zhang, Y., Sun, X., & Wang, B. (2016). Efficient algorithm for k-barrier coverage based on integer linear programming. *China Communications*, *13*(7), 16–23.
18. Xiao, F., Yang, X., Yang, M., Sun, L., Wang, R., & Yang, P. (2016). Surface coverage algorithm in directional sensor networks for 3D complex terrains. *Tsinghua Science and Technology*, *21*(4), 397–406.
19. Xie, S., & Wang, Y. (2014). Construction of tree network with limited delivery latency in homogeneous wireless sensor networks. *Wireless Personal Communications*, *78*(1), 231–246.
20. Xiao, F., Sha, C., Chen, L., & Sun, L. (2015). Noise-tolerant localization from incomplete range measurements for wireless sensor networks. In *Proceedings of IEEE conference on computer communications (INFOCOM)* (pp. 2794–2802).
21. Stemm, M., & Katz, R. H. (1997). Measuring and reducing energy consumption of network interfaces in hand-held devices. *IEICE Transactions on Communications*, *E80–B*(8), 1125–1131.
22. Gu, Y., & He, T. (2007). Data forwarding in extremely low duty-cycle sensor networks with unreliable communication links. In *Proceedings of the 5th ACM international conference on embedded networked sensor systems (SenSys)* (pp. 321–334).
23. Liu, H., Zhang, B., Mouftah, H. T., Shen, X., & Ma, J. (2009). Opportunistic routing for wireless ad hoc and sensor networks: Present and future directions. *IEEE Communications Magazine*, *47*(12), 103–109.
24. Biswas, S., & Morris, R. (2004). Opportunistic routing in multi-hop wireless networks. *ACM SIGCOMM Computer Communication Review*, *34*(1), 69–74.
25. Cacciapuoti, A. S., Caleffi, M., & Paura, L. (2009). A theoretical model for opportunistic routing in ad hoc networks. In *IEEE International conference on ultra modern telecommunications & workshops* (pp. 1–7).
26. Cacciapuoti, A. S., Caleffi, M., & Paura, L. (2010). Optimal constrained candidate selection for opportunistic routing. In *IEEE global telecommunications conference (GLOBECOM)* (pp. 1–5).
27. Guo, S., Gu, Y., Jiang, B., & He, T. (2009). Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links. In *Proceedings of the ACM annual international conference on mobile computing and networking (MobiCom)* (pp. 133–144).
28. Jiao, X., Lou, W., Ma, J., Cao, J., Wang, X., & Zhou, X. (2010). Duty-cycle-aware minimum latency broadcast scheduling in multi-hop wireless networks. In *Proceedings of the 30th IEEE international conference on distributed computing systems (ICDCS)* (pp. 754–763).
29. Zhao, D., Chin, K. W., & Raad, R. (2014). Minimizing broadcast latency and redundancy in asynchronous wireless sensor networks. *Wireless Networks*, *20*(3), 345–360.
30. Khiati, M., & Djenouri, D. (2012). Cluster-based fast broadcast in duty-cycled wireless sensor networks. In *Proceedings of the 11th IEEE international symposium on network computing and applications (NCA)* (pp. 249–252).

31. Lu, J., & Whitehouse, K. (2009). Flash flooding: Exploiting the capture effect for rapid flooding in wireless sensor networks. In *Proceedings of IEEE conference on computer communications (INFOCOM)* (pp. 2491–2499).
32. Xu, L., Chen, G., Cao, J., Lin, S., Dai, H., Wu, X., et al. (2015). Optimizing energy efficiency for minimum latency broadcast in low-duty-cycle sensor networks. *ACM Transactions on Sensor Networks (TOSN), 11(4)*, *57*, 1–31.
33. Li, Z., Li, M., Liu, J., & Tang, S. (2011). Understanding the flooding in low-duty-cycle wireless sensor networks. In *Proceedings of IEEE International Conference on Parallel Processing (ICPP)* (pp. 673–682).
34. Jiang, Z., Wu, D., Guo, M., Wu, J., Kline, R., & Wang, X. (2012). Minimum latency broadcasting with conflict awareness in wireless sensor networks. In *Proceedings of the 41st IEEE international conference on parallel processing (ICPP)* (pp. 490–499).
35. Zhao, D., Chin, K. W., & Raad, R. (2014). Approximation algorithms for broadcasting in duty cycled wireless sensor networks. *Wireless Networks*, *20*(8), 2219–2236.
36. Jiao, X., Lou, W., Wang, X., Ma, J., Cao, J., & Zhou, X. (2013). On interference-aware gossiping in uncoordinated duty-cycled multi-hop wireless networks. *Ad Hoc Networks*, *11*(4), 1319–1330.
37. Du, W., Liando, J. C., Zhang, H., & Li, M. (2015). When pipelines meet fountain: fast data dissemination in wireless sensor networks. In *Proceedings of the 13th ACM international conference on embedded networked sensor systems (SenSys)* (pp. 365–378).
38. Hong, J., Cao, J., Li, W., Lu, S., & Chen, D. (2010). Minimum-transmission broadcast in uncoordinated duty-cycled wireless ad hoc networks. *IEEE Transactions on Vehicular Technology (TVT)*, *59*(1), 307–318.
39. Tang, B., Ye, B., Hong, J., & You, K. (2011). Distributed low redundancy broadcast for uncoordinated duty-cycled WANETs. In *Proceedings of IEEE global telecommunications conference (GLOBECOM)* (pp. 1–5) .
40. Han, K., Liu, Y., & Luo, J. (2013). Duty-cycle-aware minimum-energy multicasting in wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, *21*(3), 910–923.
41. Han, K., Xiang, L., Luo, J., Xiao, M., & Huang, L. (2013). Energy-efficient reliable data dissemination in duty-cycled wireless sensor networks. In *Proceedings of the 14th ACM international symposium on mobile ad hoc networking and computing (Mobihoc) (pp. 287–292).*
42. Niu, J., Cheng, L., Gu, Y., Jun, J., & Zhang, Q. (2013). Minimum-delay and energy-efficient flooding tree in asynchronous low-duty-cycle wireless sensor networks. In *Proceedings of IEEE wireless communications and networking conference (WCNC)* (pp. 1261–1266) .
43. Zhang, X., Jia, X., Jin, J., & Dan, K. S. (2015). Delay-constrained efficient broadcasting in duty-unaware asynchronous wireless sensor networks. In *Proceedings of IEEE wireless communications and networking conference (WCNC)* (pp. 1841–1846).
44. Cheng, L., Gu, Y., He, T., & Niu, J. (2013). Dynamic switching-based reliable flooding in low-duty-cycle wireless sensor networks. In *Proceedings of IEEE conference on computer communications (INFOCOM)* (pp. 1393–1401).
45. Guo, S., Kim, S. M., Zhu, T., Gu, Y., & He, T. (2011). Correlated flooding in low-duty-cycle wireless sensor networks. In *Proceedings of the 19th IEEE international conference on network protocols (ICNP)* (pp. 383–392).
46. Sun, Y., Gurewitz, O., Du, S., Tang, L., & Johnson, D. B. (2009). ADB: An efficient multihop broadcast protocol based on asynchronous duty-cycling in wireless sensor networks. In *Proceedings of the 7th ACM international conference on embedded networked sensor systems (SenSys)* (pp. 43–56).
47. Han, K., Luo, J., Xiang, L., Xiao, M., & Huang, L. (2015). Achieving energy efficiency and reliability for data dissemination in duty-cycled WSNs. *IEEE/ACM Transactions on Networking (TON)*, *23*(4), 1041–1052.
48. Zhao, Z., Dong, W., Bu, J., Gu, Y., & Chen, C. (2015). Link-correlation-aware data dissemination in wireless sensor networks. *IEEE Transactions on Industrial Electronics*, *62*(9), 5747–5757.
49. Chen, Q., Cheng, S., Gao, H., Li, J., & Cai, Z. (2015). Energy-efficient algorithm for multicasting in duty-cycled sensor networks. *Sensors*, *15*(12), 31224–31243.
50. Duc, T. L., Le, D. T., Zalyubovskiy, V. V., Kim, D. S., & Choo, H. (2016). Level-based approach for minimum-transmission broadcast in duty-cycled wireless sensor networks. *Pervasive and Mobile Computing*, *27*(C), 116–132.
51. Xu, L., Zhu, X., Dai, H., Wu, X., & Chen, G. (2016). Towards energy-fairness for broadcast scheduling with minimum delay in low-duty-cycle sensor networks. *Computer Communications (COMCOM)*, *75*, 81–96.
52. Wang, F., & Liu, J. (2012). On reliable broadcast in low duty-cycle wireless sensor networks. *IEEE Transactions on Mobile Computing (TMC)*, *11*(5), 767–779.
53. Wang, Q., Hempstead, M., & Yang, W. (2006). A realistic power consumption model for wireless sensor network devices. In *Proceedings of the 3rd annual IEEE communications society on sensor and ad hoc communications and networks (SECON)* (pp. 286–295).

Lijie Xu is currently an assistant professor of the School of Computer Science and Technology at Nanjing University of Posts and Telecommunications, Jiangsu, China. He received the Ph.D. degree in the Department of Computer Science and Technology from Nanjing University, Jiangsu, China, in 2014. He was a research assistant in the Department of Computing at the Hong Kong Polytechnic University from Nov. 2011 to May 2012. His research interests are mainly in the areas of wireless sensor networks, ad-hoc networks, mobile and distributed computing.

Geng Yang received the Ph.D. degree in Computational Mathematics from Laval University in 1994 and was a post-doctoral research fellow in Center for Research on Computation & its Applications at University of Montreal from 1994 to 1996 in Canada respectively. His research interest includes information & network security, privacy preserving, and parallel & distributed computing. Professor Yang is a member of the IEEE Computer Society.

Lei Wang received his bachelor degree from Anhui University of China in 2008. He was a Master-Doctor combined program graduate student in computer science of Nanjing University of Science & Technology (NUST), China since 2008. He was a visiting student in Michigan State University during 2013-2014. Currently, he is an assistant professor in School of Computer, Nanjing University of Posts and Telecommunications, China. His research interests are network coding, wireless sensor network, multipath routing, distributed storage system.

Jia Xu received the M.S. degree in School of Information and Engineering from Yangzhou University, Jiangsu, China, in 2006 and the Ph.D. degree in School of Computer Science and Engineering from Nanjing University of Science and Technology, Jiangsu, China, in 2010. He is currently a professor in Nanjing University of Posts and Telecommunications. He was a visiting scholar in the Department of Electrical Engineering and Computer Science at Colorado School of Mines from Nov. 2014 to May 2015. His main research interests include crowdsourcing, opportunistic networks and wireless sensor networks.

Baowei Wang is currently working as an Assistant Professor in the School of Computer and Software at Nanjing University of Information Science & Technology, Nanjing, China. He received his B.S. and Ph.D. degrees in Computer Science from Hunan University, Changsha, China, in 2005 and 2011, respectively. His research interests include wireless sensor networks, data security and privacy, and mobile computing.