On Multihop Broadcast over Adaptively Duty-Cycled Wireless Sensor Networks

Shouwen Lai[1] and Binoy Ravindran[1]

Virginia Tech, ECE Department, Blacksburg, VA 24060, USA,
{swlai,binoy}@vt.edu

Abstract. We consider the problem of multihop broadcast over adaptively duty-cycled wireless sensor networks (WSNs) where neighborhood nodes are not simultaneously awake. We present Hybrid-cast, an asynchronous and multihop broadcasting protocol, which can be applied to low duty-cycling or quorum-based duty-cycling schedule where nodes send out a beacon message at the beginning of wakeup slots. Hybrid-cast achieves better tradeoff between broadcast latency and broadcast count compared to previous broadcast solutions. It adopts opportunistic data delivery in order to reduce the broadcast latency. Meanwhile, it reduces redundant transmission via delivery deferring and online forwarder selection. We establish the upper bound of broadcast count and the broadcast latency for a given duty-cycling schedule. We evaluate Hybrid-cast through extensive simulations. The results validate the effectiveness and efficiency of our design.

1 Introduction

Multihop broadcast [17] is an important network service in WSNs, especially for applications such as code update, remote network configuration, route discovery, etc. Although the problem of broadcast has been well studied in always-on networks [12, 22] such as wireless ad hoc networks where neighbor connectivity is not a problem, broadcast is more difficult in duty-cycled WSNs where each node stays awake only for a fraction of time slots and neighborhood nodes are not simultaneously awake for receiving data. The problem becomes more difficult in asynchronous [24] and heterogenous duty-cycling [9] scenarios.

To support broadcast, synchronization of wakeup schedules is one promising approach adopted by many duty-cycling MAC protocols, such as S-MAC [23] and T-MAC [4]. Such protocols simplify broadcast communication by letting neighborhood nodes stay awake simultaneously. However, this approach results in high overhead for periodic clock synchronization when compared to the low frequency of broadcast service in WSNs. Since energy is critical to WSNs, energy-efficient asynchronous MAC protocols have become increasingly attractive for data communication, as proposed in B-MAC [14], RI-MAC [18], Disco [5], and quorum-based wakeup scheduling [24, 10].

However, previous asynchronous MAC protocols for duty-cycled WSNs mostly focus on unicast communication, and do not work well for broadcasting. One straightforward way to support one-hop broadcast in such cases is to deliver data multiple times for all neighbors, which results in redundant transmissions. With multihop broadcasting to an entire network, the problems are more amplified, as some neighbors attempt to forward the broadcast message while the original transmitting node still attempts to transmit it to other nodes of its neighbors, increasing collisions and wasting energy consumption for transmission.

There have been some efforts in the past to support multihop broadcasting in duty-cycled WSNs. Wang *et al.* [21] transformed the problem into a shortest-path problem with the assumption of duty-cycle awareness, which is not valid for asynchronously duty-cycled WSNs. DIP [16], ADB [17], and opportunistic flooding [6] were designed with a smart gossiping approach. Essentially, these protocols use unicast to replace broadcast for flooding, toward reducing the flooding latency in the entire network. However, they may lack efficiency in large-scale networks or on delivering large chunks of data to entire network because message cost and higher transmission energy consumption.

To overcome the disadvantages of replacement via pure unicast, we present Hybrid-cast, an asynchronous broadcast protocol for broadcasting with low latency and reduced message count. In Hybrid-cast, a node only forwards a message to neighbors who wake up and send out beacon messages. A node defers broadcasting by one or more time slot(s) after receiving the beacon message from the first awake neighbor in order to wait for more nodes that may potentially wake up, so that more nodes are accommodated in one broadcast. It also adopts online forwarder selection in order to reduce the transmission redundancy. Compared with previous protocols, Hybrid-cast can achieve less broadcast latency and smaller message count.

The rest of the paper is organized as follows: We discuss related works in Section 2. In Section 3, we state our models, assumptions, and preliminaries. In Section 4, we present the design of Hybrid-cast. We theoretically analyze the performance of Hybrid-cast in Section 5, and provide further discussions in Section 6. Simulation results are presented in Section 7. We conclude in Section 8.

2 Past and Related Works

We review past and related efforts on broadcast solutions for duty-cycled WSNs. Due to space constraints, we omit reviews for always-on multihop networks.

Gossip or opportunistic approach. Opportunistic unicast routing, like EXOR [1], was proposed to exploit wireless broadcast medium and multiple opportunistic paths for efficient message delivery. Regarding broadcasting, the main purpose of opportunistic approach aimed at ameliorating message implosion. Smart Gossip [8] adaptively determines the forwarding probability for received flooding messages at individual sensor nodes based on previous knowledge and network topology.

In Opportunistic Flooding [6] (abbreviated as OppFlooding), each node makes probabilistic forwarding decisions based on the delay distribution of next-hop nodes. Only opportunistic early packets are forwarded via the links outside of the energy-optimal tree to reduce flooding delays and the level of redundancy.

To resolve decision conflicts, the authors build a reduced flooding sender set to alleviate the hidden terminal problem. Within the same sender set, the solution uses a link-quality-based backoff method to resolve and prioritize simultaneous forwarding operations. The main problem of pure opportunistic flooding is the overhead in terms of transmission times.

Synchronized or duty-cycle awareness. Wang et al. [21] present a centralized algorithm, mathematically modeling the multihop broadcast problem as a shortest-path problem in a time-coverage graph, and also present two similar distributed algorithms. However, their work simplifies many aspects necessary for a complete MAC protocol, and may not be appropriate for real implementation. The work also assumes duty-cycle awareness, which makes it difficult to use it in asynchronous WSNs since duty-cycle awareness needs periodic time-synchronization due to clock drifting. RBS [20] proposes a broadcast service for duty-cycled sensor networks and shows its effectiveness in reducing broadcast count and energy costs.

All these works based on synchronization assume that there are usually multiple neighbors available at the same time to receive the multicast/flooding message sent by a sender. This is not true in low duty-cycled asynchronous networks.

Asynchronous solution. B-MAC [14] can support single-hop broadcast in the same way as it supports unicast, since the preamble transmission over an entire sleep period gives all of the transmitting node's neighbors a chance to detect the preamble and remain awake for the data packet. X-MAC [3] substantially improves B-MAC's performance for unicast, but broadcast support is not clearly discussed in that paper. X-MAC is not promising for broadcast since the transmitter has to continually trigger the neighbors to wake up.

ADB [17] avoids the problems faced by B-MAC and X-MAC by efficiently delivering information on the progress of each broadcast. It allows a node to go to sleep immediately when no more neighbors need to be reached. ADB is designed to be integrated with an unicast MAC that does not occupy the medium for a long time, in order to minimize latency before forwarding a broadcast. The effort in delivering a broadcast packet to a neighbor is adjusted based on link quality, rather than transmitting throughout a duty cycle or waiting throughout a duty cycle for neighbors to wake up. Basically, ADB belongs to the unicast replacement approach and it needs significant modification to existing MAC protocols for supporting broadcast.

3 Models and Preliminaries

3.1 Network Model and Assumptions

We model a multi-hop wireless sensor network as a directed graph $G(V, E)$, where V is the set of nodes, and E is the set of edges. If node v_j is within the transmission range of node v_i, then an edge (v_i, v_j) is in E. We assume bidirectional links. We use the term "connectivity" loosely in our model, in the sense that a topologically connected network in our context may not be connected at any time; instead, all nodes are reachable from a node within a finite amount of time by the underlying MAC protocol. We define the one-hop neighborhood of node n_i as $N(i)$.

We assume that time axes are arranged as consecutive short time slots, all slots have the same duration T_s, and each node n_i adopts a periodic wakeup schedule every L_i time slots. The wakeup schedule can be once every L_i slots or based on quorum schedules (i.e., cyclic quorum systems or grid quorum systems [11]). L_i is called cycle length for node n_i. We assume that beacon messages are sent out at the beginning of wakeup slots, as in [18, 10]. When a node wants to transmit messages, it will wait until beacons are received from neighbors.

We also make the following assumptions: (1) There is no time synchronization between nodes (thus the time slots in two nodes are not necessarily aligned); (2) The overhead of turning on and shutting down radio is negligibly small compared with the long duration of time slots (i.e., $50ms \sim 500ms$); (3) There is only one sink node in the network (but our solution can be easily extend to the scenario of multiple sink nodes).

3.2 Heterogenous Wakeup Scheduling

Heterogenous wakeup scheduling means that nodes adopt different wakeup schedules independently to reflect their remaining energy. How to configure this schedule (i.e., the value of L_i) has been described by past works such as [19], and is outside the scope of our work.

We consider two types of heterogenous wakeup scheduling approaches: low duty-cycling schedule and quorum duty-cycling schedule. Low duty-cycling means that a node wakes up one slot for every n_i (n_i is an integer) time slots. For example, in Figure 1(a), receiver 1 has a schedule of $[1, 0, 0]$, where 1 means wakeup slot, and receiver 2 has the schedule of $[1, 0, 0, 0]$. They do not always overlap on wakeup slots.

For quorum-based duty cycling, wakeup scheduling follows a quorum system [11] design. In quorum-based duty cycling, two neighbor nodes can hear each other at least once within limited time slots via the non-empty intersection property of quorums. We choose cyclic quorum system [10] in this paper. But our work can also be applied to other quorum systems.

We use the following definitions for briefly reviewing quorum systems (which are used for wakeup scheduling).

Let n denote a cycle length and $U = \{0, \cdots, n-1\}$.

Definition 1. *A* quorum system $\mathcal{Q}$ *under* U *is a superset of non-empty subsets of* U*, each called a quorum, which satisfies the intersection property:* $\forall G, H \in \mathcal{Q} : G \cap H \neq \emptyset$. *If* $\forall G, H \in \mathcal{Q}$*,* $i \in \{0, 1, ...n-1\}$*:* $G \cap (H + i) \neq \emptyset$*, where* $H + i = \{(x+i) \ mod \ n : x \in H\}$. $\mathcal{Q}$ *is said to have the* rotation closure property

A cyclic quorum system (cqs) satisfies the rotation closure property, and is denoted as $C(A, n)$ where A is a quorum and n is the cycle length. For example, the cqs $\{\{1, 2, 4\}, \{2, 3, 5\} \cdots, \{7, 1, 3\}\}$ can be denoted as $C(\{1, 2, 4\}, 7)$. The wakeup schedule complying with $C(\{1, 2, 4\}, 7)$ are $[1, 1, 0, 1, 0, 0, 0]$ and its rotations as shown in Figure 1(b).

For two different cyclic quorum systems $C(A_1, n_1)$ and $C(A_2, n_2)$, if two quorums from them, respectively, have non-empty intersections even with drifting clocks, they can be used for heterogenous wakeup scheduling in WSNs as proved by in [10]. For example, given $C(\{1,2,4\},7)$ and $C(\{1,2,4,10\},13)$, two quorums from them, respectively, will have non-empty intersection for every 13 time slots. Therefore, two nodes that wake up with schedules complying with any two quorums from the two cyclic quorum systems can hear each other.

3.3 Problem Statement

Let us define the broadcast latency as the time between the beginning of a broadcast and the time at which every node receives the broadcast message. Also, let us define the broadcast count as the number of broadcasting via all nodes to ensure that the entire network receives the message. Our goal is to design a broadcast schedule, which can not only shorten the broadcast latency but also the broadcast count for flooding a message to the entire network. The protocol that we present, Hybrid-cast, is a heuristic solution to this problem.

4 The Hybrid-cast Protocol

4.1 Overview

In Hybrid-cast, a transmitter will stay awake for long enough time to hear the beacon message from its neighbors. Due to heterogenous wake-up scheduling, for low duty-cycling, the node will stay awake for L_m time slots, which is the largest cycle length of all neighbors. By doing this, it can hear beacons from all neighbors. For quorum duty-cycling, the transmitter will switch to the wakeup schedules which has the largest cycle length from all its neighbors.

Hybrid-cast adopts opportunistic forwarding with delivery deferring to shorten broadcast latency and broadcast count: the transmitter will forward the message within δ time after it hears the beacon messages from early-wakeup neighbors, rather than forwarding immediately after hearing the beacon messages. An illustration is given in Figure 1(a). Here, δ (i.e., $\delta = T_s$ for low duty-cycling) is called the deferring time. By deferring, the first-awake neighbor can still receive the broadcast message. Meanwhile, more neighbors which wake up during the deferred time period can receive the broadcast message, so that less number of broadcast is necessary for one-hop broadcasting.

To further reduce redundant transmissions, Hybrid-cast adopts online forwarder selection. "Online" means that a node selects the least relay node among its instant one-hop awake neighbors, rather than all one-hop neighbors, to cover its two hop neighbors, in order to reduce transmission redundancy and collision.

4.2 Wakeup Schedule Switching

Due to adaptive duty-cycling, neighbor discovery becomes more difficult. In order to hear the beacon message from all neighbors, a node must switch its wakeup schedule for staying awake for enough time slots.

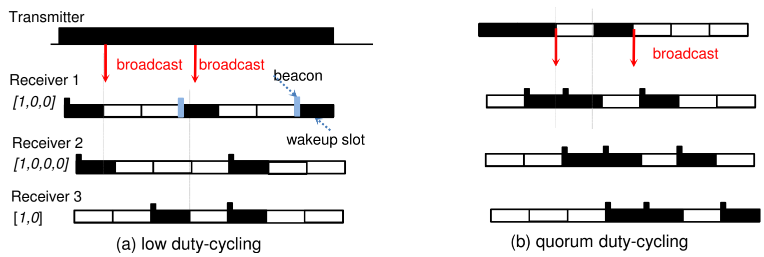

Fig. 1. Opportunistic broadcasting with delivery deferring (a) low duty-cycling case; (b) quorum duty-cycling case with wakeup schedules of [1,1,0,1,0,0,0] and its rotations which comply with (7,3,1) cqs design in [10].

For the case of low duty-cycling, in the idle state, a node n_i follows its own wakeup schedule. If the node needs to forward a broadcast message (i.e., the node is selected as a relay node), n_i should stay awake for at least L_m slots, where $L_m = \max_{n_j \in N(i)}\{L_j\}$. By doing this, n_i can hear beacon messages from all neighbors within the minimum necessary time slots.

For the case of quorum duty-cycling, n_i just switches to the schedule of the node which has the longest cycle length. Due to the non-empty intersection property [10], n_i can still hear all neighbors even when it does not stay awake in every time slot of a whole cycle length.

A node needs to know the largest cycle length of its neighbors before schedule switching. This can be achieved by either pre-setting the largest global cycle length or by dynamic neighbor information exchange protocols.

4.3 Opportunistic Forwarding with Deferring

Opportunistic forwarding means that a transmitter forwards data immediately to the neighbor which wake up earlier, for minimizing broadcast latency. Previous efforts on opportunistic flooding such as [6] use unicast for broadcasting. However, opportunistic forwarding via pure unicast suffers from large broadcast count.

In Hybrid-cast, broadcast deferring is adopted to minimize the one-hop broadcast count. By deferring, a transmitter will not broadcast messages immediately after receiving the beacon from the first-awake neighbor. In order to ensure that more neighbors receive the broadcast message, the transmitter defers the broadcasting by $\delta = 1$ *time slot*. By doing this, the first-awake neighbor can still receive the message, and the neighbors which wake up before the deferring time is due can also receive the broadcast message. Thus, deferring combines the advantages of opportunistic forwarding and the advantages of broadcasting over wireless radio.

As shown in Figure 1, suppose there are three neighbors for the transmitter. The transmitter only needs to broadcast two times (marked by the red arrow) to ensure that all neighbors will receive the message. This is more efficient than the pure opportunistic forwarding mechanism.

The only disadvantages of deferring is the additional latency (1 time slot for one-hop broadcasting) for flooding to the entire network. Therefore, deferring allows the tradeoff between the number of broadcast count and the broadcast latency to be exploited. We show in Section 5.2 that such additional latency is relatively small for the low duty-cycling case.

4.4 Online Forwarder Selection

In order to reduce the broadcast count or redundant transmission for multihop broadcasting, it is necessary to select as small number of relay nodes as possible. Many past efforts have formulated this problem as the Minimum Connecting Dominating Set (MCDS) problem [2]. However, we argue that a static MCDS cannot be applied for relay node selection in Hybrid-cast. First, to shorten the latency, it is necessary to select the relay nodes or forwarders along the direction of opportunistic forwarding, which results in online (or live) forwarder selection, rather than a static topology control as done in MCDS. Secondly, MCDS does not achieve minimum broadcast count in asynchronous duty-cycled WSNs due to multiple delivery for single hop broadcasting.

Algorithm 1: Algorithm for all node n_x:

set $N_{awake}(x)$;
$N^2_{reachable}(x) = \cup_{y \in N_{awake}(x)} N(y) - N_{awake}(x)$;
for $n_y \in N_{awake}(x)$ **do**
    if *node* $n_u \in N^2_{reachable}(x)$ *which is only reachable by* n_y **then**
        n_y is selected into O-$MPR(x)$;
        $N^2_{reachable}(x) = N^2_{reachable}(x) - n_u$;
while $N^2_{reachable}(x) \neq \emptyset$ **do**
    for $n_u \in N_{awake}(x)$ - *O-*$MPR(x)$ **do**
        $n_m = n_u$ which covers the most nodes in $N^2_{reachable}(x)$;
    n_m is selected into O-$MPR(x)$;
    $N^2_{reachable}(x) = N^2_{reachable}(x)$ - node set covered by n_m;

In Hybrid-cast, initially, each node maintains its one hop awake neighbors (defined as $N(x)$) and the set of two hop neighbors $N^2(x)$ based on any underlying neighbor discovery protocols. The sink node or a relay node n_x computes the least number of relay nodes among its one-hop awake neighbors (defined as $N_{awake}(x)$) to cover the reachable two hop neighbors (defined as $N^2_{reachable}(x)$).

$$N^2_{reachable}(x) = \cup_{y \in N_{awake}(x)} N(y) - N_{awake}(x) \qquad (1)$$

The main purpose of the online forwarder selection algorithm in a transmitter n_x is to compute $N^2_{reachable}(x)$ as shown in Equation 1, and to compute the minimum number of relays to cover $N^2_{reachable}(x)$.

We adopt a heuristic solution, which is similar to the minimum multipoint relays (MPR) algorithm in [15]. The MPR problem is NP-Complete as shown in [15]. Thus, the minimum online forwarder selection problem is also NP-Complete. We denote the online MPR set for the transmitter n_x as O-$MPR(x)$. We provide a heuristic algorithm for computing O-$MPR(x)$ as described in Algorithm 1. An illustration is given in Figure 2(a).

Let us define the delivery latency from node n_i to node n_j as the time between when the data is ready in n_i and time at which the broadcast data is received by the neighbors, and denote the latency as $\tau_{i,j}(t)$ at time t ($\tau_{i,j}(t)$ is varying at different time). We have the following property.

Theorem 1. *Suppose node n_i has two neighbor n_j and n_k which are one hop away from each other. Then, at a time instant, t_i, we have the triangular property:*

$$\tau_{i,j}(t_i) \leq \tau_{i,k}(t_i) + \tau_{k,j}(t_i + \tau_{i,k}(t_i)) \tag{2}$$

Proof. Suppose at time t_i, the data arriving time slot at n_j is t_j, and the data arriving time at n_k is t_k.

If $t_k \leq t_j$, which means that the data arriving time at n_k is earlier than the data arriving time at n_j, $\tau_{i,k}(t_i + \tau_{k,j}(t_i + \tau_{i,k}(t_i)) = t_k - t_i + t_j - t_k = t_j - t_i = \tau_{i,j}(t_i)$. Otherwise, if $t_k > t_j$, which means that the data arriving time at n_k is later than the data arriving time at n_j, we have $\tau_{i,k}(t_i) + \tau_{k,j}(t_i + \tau_{i,k}(t_i)) = t_k - t_i + t_j^{'} - t_k > t_j - t_i + t_j^{'} - t_k > t_j - t_i > \tau_{i,j}(t_i)$. The theorem follows.

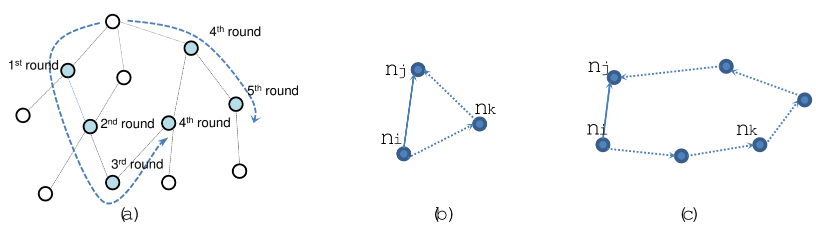

Fig. 2. Online forwarder selection and the triangular path condition.

Theorem 1, as illustrated in Figure 2(b), illustrates that node n_i will always broadcast data to its one-hop neighbor n_j directly, without through other nodes.

We also have the following property.

Lemma 1. Triangular Path Condition*: For a node n_i and its neighbor n_j, at any time, the one-hop broadcast latency $n_i \rightarrow n_j$ is always the minimum possible.*

We omit the proof for the triangular path condition since it is a simple extension from that of Theorem 2. An illustration is given in Figure 2(c). Note that the triangular path condition does not exist in static networks. The triangular path condition indicates that the one-hop direct broadcast always achieves the least latency, in adaptively duty-cycled WSNs.

5 Performance Analysis

We now analyze the performance of Hybrid-cast in terms of the broadcast count and the broadcast latency, in order to illustrate its design advantages.

5.1 Upper-Bound on One-Hop Broadcast Count

We consider two scenarios in analyzing the one-hop broadcast count. In the low duty-cycling scenario, the schedule for a node n_i is waking up once every L_i time slots. In the quorum duty-cycling scenario, the schedule for a node n_i is waking up q times for every L_i consecutive time slots, where q is the quorum size.

Lemma 2. *[low duty-cycling] In Hybrid-cast, for a node n_i, the broadcast count is at least one, and at most $\max\{\Delta, L_m\}$, where Δ is the node degree of n_i and L_m is the maximum cycle length of nodes in the neighborhood.*

Proof. If all nodes wake up within the same time slot, then after broadcast deferring, the transmitter can hear all neighbors, and one broadcast can cover all neighbors.

Otherwise, if $\Delta \geq L_m$, the transmitter can hear all neighbors via staying awake for L_m time slots. Therefore, the maximum broadcast count is L_m. If $\Delta < L_m$, the transmitter can hear neighbors for at most Δ times, and the maximum broadcast count is Δ. Thus, the maximum number of broadcast count is $\max\{\Delta, L_m\}$.

By the Lemma 2, the upper bound of broadcast count in Hybrid-cast is at most n (where n is the network size) in the ideal case.

Lemma 3. *[quorum duty-cycling] In Hybrid-cast, for node n_i, the broadcast count is at least one, and at most $\max\{\Delta, q_m\}$, where Δ is the node degree of n_i and q_m is the largest quorum size of the quorum systems adopted by nodes in the neighborhood.*

Proof. If all nodes wake up within one time slot, then after broadcast deferring, the transmitter can hear all neighbors, and one broadcast can cover all neighbors.

Otherwise, if $\Delta \geq q_m$, the transmitter can hear all neighbors via staying awake in time slots scheduled by the quorum design. Therefore the maximum broadcast count is q_m. If $\Delta < q$, the transmitter will hear neighbors for at most Δ times, and the maximum broadcast count is Δ. Thus, the maximum broadcast count is $\max\{\Delta, q_m\}$.

5.2 Delivery Latency

Lemma 4. *Suppose the depth of the network (i.e., maximum layers by breadth-first-search) is D_{max}. Then, the upper bound for delivery latency is $L_m * D_{max} * T_s$ in low duty-cycling mode, where L_m is the maximum cycle length of nodes in the network. The upper bound is $q_m * D_{max} * T_s$ for quorum duty-cycling mode, where q_m is the largest quorum size of the quorum systems adopted by all nodes in the network.*

Proof. Based on the Triangular Path Condition in Lemma 1, a node always broadcasts a message to its one-hop neighbors directly. Thus, for one hop broadcasting, the latency is at most $n_m * T_s$. After $n_m * T_s$, all nodes in the first layer will receive the broadcast message. Therefore, after $n_m * D_{max} * T_s$ time, all nodes in the network will receive the broadcast message.

6 Discussion

Note that we do not assume local synchronization or duty-cycle awareness, which is required by past works such as [6] and [21]. The assumption in Hybrid-cast is neighbor-awareness. Such awareness can be achieved by neighbor discovery protocols, or by quorum-based duty-cycling [10]. Each node will inform its neighbors after reconfiguration on the duty-cycling.

By adopting quorum duty-cycling, Hybrid-cast can be extended to mobile WSNs, because neighbor discovery is guaranteed within bounded time in quorum duty-cycling, as shown in [10].

Due to the problem of hidden terminal, it is possible that one node may receive broadcast messages from two nodes simultaneously, which leads to collision. For reliable broadcasting, if a node received the broadcast, it can set a mark field in the beacon message. By checking the beacon message from the neighbor, a transmitter can decide whether retransmission is necessary. We do not defer broadcast for retransmission. The transmitter could backoff a random period $0 \leq t \leq T_s$ in order to avoid collision.

We do not explicitly consider reliability issues in Hybrid-cast. However, the traditional ACK and NACK mechanisms for reliable data transmission can be applied to Hybrid-cast to support reliable broadcasting.

7 Simulation Results

We simulated Hybrid-cast using the OMNET++ simulator [13] and compared it against ADB [17] and opportunistic broadcasting [6] (denoted as OppFlooding).

Our experimental settings were consistent with the configurations in [6, 7]. We set the wireless loss rate as 0.1 and the duration of one time slot as 100 ms. The wireless communication range was set to 10m. We adopted the wireless loss model in [25], which considers the oscillation of radio. The size of the broadcast message packets was fixed as 512 bytes.

We examined the two main factors that affect the performance of our algorithms, including network size and duty-cycle setting. We generated a network with different number of nodes. For each network size, we randomly generated 10 topologies. Each data point reported in this section is the average of 10 topologies, with 10 runs on each topology. We varied the network size to understand its impact on the broadcast count and broadcast latency.

We measured the performance of the algorithms in a variety of duty cycle settings. For the low duty-cycling scenario, we varied the duration of the total periodic cycle length from $2T_s$ to $10T_s$ to generate heterogenous duty-cycling in a network for different nodes. For the quorum duty-cycling case, we choose the $(7, 3, 1)$, $(13, 4, 1)$, and $(21, 5, 1)$ difference sets for the heterogenous schedule settings. Since ADB, OppFlooding, and Hybrid-cast are independent of wakeup scheuling, we argue that the comparison is fair, even though ADB and OppFlooding do not explicitly support quorum duty-cycling.

7.1 Broadcast Count

We first measure the broadcast count which is the total number of broadcasting for flooding a message to the whole network. In this set of experiments, the network size was fixed by 200 nodes. The experimental results for different protocols are shown in Figure 3(a). In the low duty-cycling case, Hybrid-cast outperforms ADB by approximately 50%, because of the less number of unicasts involved, due to the protocol's deferring and online forwarder selection.

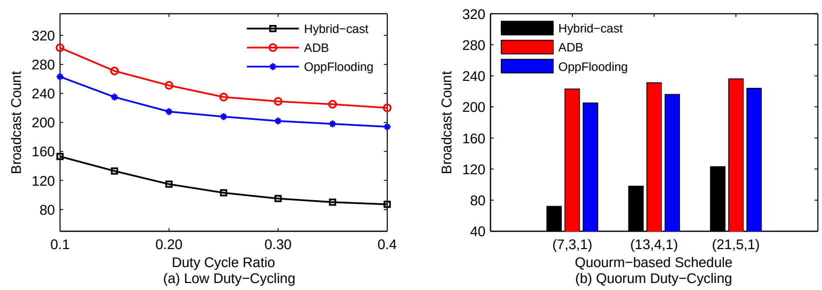

Fig. 3. Performance comparison on broadcast count

For the quorum-based duty cycle setting, all nodes in the network chose homogenous quorum schedules. The setting was varied simultaneously for all nodes in different set of experiments. As shown in Figure 3(b), Hybrid-cast performs better since broadcasting are aggregated within quorum slots in each cycle. For example, for the $(7, 3, 1)$ setting (i.e., a node will stay awake at the 1^{st}, 2^{nd}, and 4^{th} slot on every 7 consecutive slots), there are at most 3 broadcasts to ensure that all neighbor nodes receive the broadcast message. However, for ADB and OppFlooding, the average one-hop broadcast count was 5 or 6, given the average degree in the network that we configured. The results validate the performance analysis in Section 5.1.

7.2 Broadcast Latency

Figure 4(a) shows the broadcast latency (defined as the time from broadcast beginning to all nodes receiving the broadcast data). With deferring, Hybrid-cast has slightly higher latency than ADB and OppFlooding, by about 10%, when the duty cycle ratio is 0.4, and by about 5%, when the duty cycle ratio is 0.1. As shown in Figure 4(a), as the duty cycle ratio decreases, the disadvantages of Hybrid-cast become more negligible, since the broadcast latency is more dominated by neighbor discovery latency.

For the case of quorum duty-cycling, as shown in Figure 4(b), we observe a similar trend as that of low duty-cycling. The latencies for all three protocols tend to increase with larger quorum cycle. However, the latencies tend to converge to

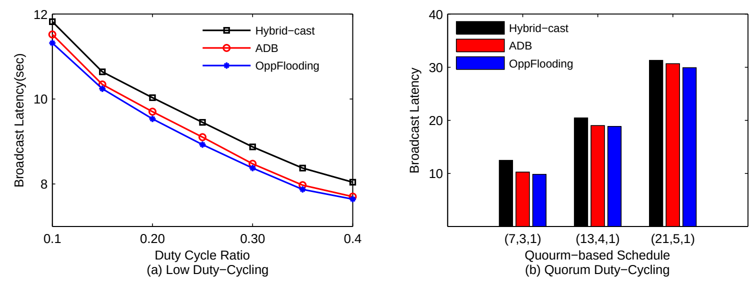

Fig. 4. Performance comparison on broadcast latency

the same value when the quorum cycle increases. This is because, the neighbor discovery latency is approximately linearly increasing with quorum cycle, as shown in [10] The results also validate the performance analysis in Section 5.2.

7.3 Impact of Network Size

We also evaluated the impact of network size and heterogenous duty-cycling on message count and broadcast latency. For the low duty-cycling case, each node randomly selected a duty cycle ratio in the range 0.1 to 0.4. For the quorum duty-cycling case, we chose the $(7, 3, 1)$, $(13, 4, 1)$, and $(21, 5, 1)$ difference sets for the schedules of all nodes (the non-empty intersection property among these sets was proved in [10]). In the simulation experiments, we varied the network size from 200 nodes to 1600 nodes.

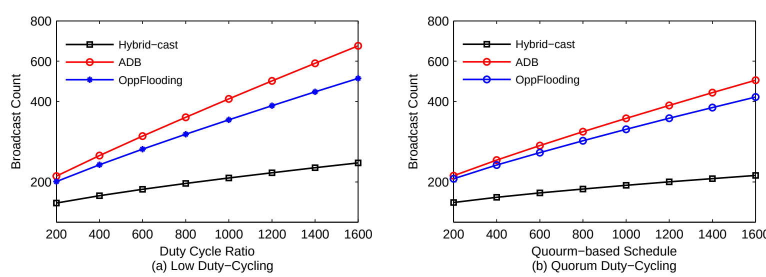

Fig. 5. Broadcast count with different network size.

As shown in Figure 5, as the network size increases, the message count of Hybrid-cast and the other two solutions exhibit an increasing trend. This is because, more relay nodes will be selected in larger networks. The same trend exists for broadcast latency as shown in Figure 6, as there are more hops along the breadth-first-search tree. This is consistent with the analysis in Section 5.2.

We also evaluated the impact of network size for quorum-based duty cycle setting. We observed similar trends for the broadcast count and broadcast la-

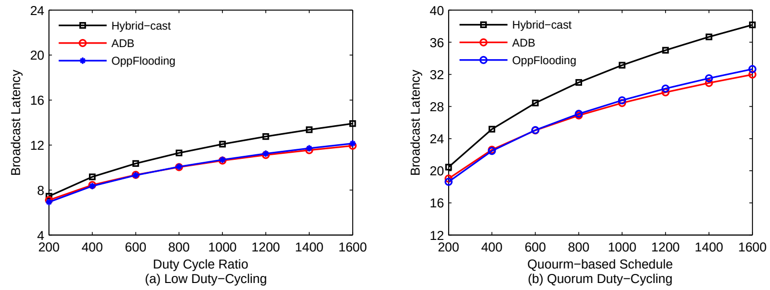

Fig. 6. Broadcast latency with different network size.

tency as that in the low duty-cycling setting. The performance comparisons thus illustrate the performance tradeoff achieved by Hybrid-cast.

8 Conclusions

In this paper, we designed an asynchronous broadcasting protocol, Hybrid-cast, for WSNs with adaptively low duty-cycling or quorum-based duty-cycling schedules. The main difficulty of this problem is that, sensor nodes are not time-synchronized and do not stay awake simultaneously. Hybrid-cast broadcasts messages to the neighbors who wake up early, in order to shorten the broadcast latency. Previous solutions often use multiple unicasts for broadcasting, which incurs high overhead. To overcome the disadvantages of such multiple unicasts, Hybrid-cast defers broadcasting to ensure that the number of awake neighbors is as large as possible. We also selected the minimum relay points online in order to reduce broadcast count and collisions.

We mathematically established the upper bound of broadcast count and broadcast latency for a given duty-cycling schedule. We compared the performance of Hybrid-cast with ADB and OppFlooding protocols. Our simulation results validated the effectiveness and efficiency of our design.

References

1. Sanjit Biswas and Robert Morris. Exor: opportunistic multi-hop routing for wireless networks. *SIGCOMM Comput. Commun. Rev.*, 35(4):133–144, 2005.
2. Jeremy Blum, Min Ding, Andrew Thaeler, and Xiuzhen Cheng. Connected dominating set in sensor networks and manets. *Handbook of Combinatorial Optimization*, pages 329–369, 2005.
3. Michael Buettner, Gary V. Yee, Eric Anderson, and Richard Han. X-mac: a short preamble mac protocol for duty-cycled wireless sensor networks. In *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 307–320, 2006.
4. T.V. Dam and K. Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks. In *ACM SenSys*, 2003.
5. Prabal Dutta and David Culler. Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications. In *ACM SenSys*, pages 71–84, 2008.

6. Shuo Guo, Yu Gu, Bo Jiang, and Tian He. Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links. In *ACM conference on Mobile computing and networking (MobiCom)*, pages 133–144, 2009.
7. Raja J., Pierre B., and Cristina V. Adaptive low power listening for wireless sensor networks. *IEEE Transactions on Mobile Computing*, 6(8):988–1004, 2007.
8. Pradeep Kyasanur, Romit Roy Choudhury, and Indranil Gupta. Smart gossip: An adaptive gossip-based broadcasting service for sensor networks. In *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS)*, pages 91–100, Oct 2006.
9. S. Lai and B. Ravindran. On distributed time-dependent shortest paths over duty-cycled wireless sensor networks. In *IEEE International Conference on Computer Communications (INFOCOM)*, 2010.
10. S. Lai, B. Zhang, B. Ravindran, and H. Cho. Cqs-pair: Cyclic quorum system pair for wakeup scheduling in wireless sensor networks. In *International Conference on Principles of Distributed Systems (OPODIS)*, volume 5401, pages 295–310. Springer, 2008.
11. W.S. Luk and T.T. Huang. Two new quorum based algorithms for distributed mutual exclusion. In *Proceedings of the International Conference on Distributed Computing Systems (ICDCS)*, pages 100 – 106, 1997.
12. Sze-Yao Ni, Yu-Chee Tseng, Yuh-Shyan Chen, and Jang-Ping Sheu. The broadcast storm problem in a mobile ad hoc network. In *5th annual ACM/IEEE international conference on Mobile computing and networking (MobiCom)*, pages 151–162, 1999.
13. OMNET++. http://www.omnetpp.org/.
14. J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *SenSys*, pages 95–107, 2004.
15. A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying for flooding broadcast messages in mobile wireless networks. In *35th Annual Hawaii International Conference on System Science*, pages 3866–3875, 2002.
16. Fred Stann, John Heidemann, Rajesh Shroff, and Muhammad Zaki Murtaza. Rbp: robust broadcast propagation in wireless networks. In *ACM SenSys*, pages 85–98, 2006.
17. Yanjun Sun, Omer Gurewitz, Shu Du, Lei Tang, and David B. Johnson. Adb: an efficient multihop broadcast protocol based on asynchronous duty-cycling in wireless sensor networks. In *ACM SenSys*, pages 43–56, 2009.
18. Yanjun Sun, Omer Gurewitz, and David B. Johnson. Ri-mac: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks. In *ACM Sensys*, pages 1–14, 2008.
19. C.M. Vigorito, D. Ganesan, and A.G. Barto. Adaptive control of duty cycling in energy-harvesting wireless sensor networks. In *IEEE SECON'07*, pages 21–30, June 2007.
20. Feng Wang and Jiangchuan Liu. Rbs: A reliable broadcast service for large-scale low duty-cycled wireless sensor networks. In *IEEE International Conference on Communications (ICC)*, pages 2416–2420, May 2008.
21. Feng Wang and Jiangchuan Liu. Duty-cycle-aware broadcast in wireless sensor networks. In *IEEE International Conference on Computer Communications (INFOCOM)*, pages 468–476, 2009.
22. Brad Williams and Tracy Camp. Comparison of broadcasting techniques for mobile ad hoc networks. In *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing (MobiHoc)*, pages 194–205, 2002.
23. W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12:493–506, 2004.
24. R. Zheng, J. C. Hou, and L. Sha. Asynchronous wakeup for ad hoc networks. In *MobiHoc*, pages 35–45, 2003.
25. M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. In *2004*, pages 517–526, 2004.