SC-MAC: A Sender-Centric Asynchronous MAC Protocol for Burst Traffic in Wireless Sensor Networks

Ye Liu[†‡] Fulong Jiang[†] Hao Liu[†] Jianhui Wu[†]
[†]National ASIC System Engineering Research Center, Southeast University, Nanjing, China
[‡]Jiangsu Provincial Key Laboratory of Sensor Network Technology, Southeast University in Wuxi, China
{liuye, jfl, wjh}@seu.edu.cn, seuliuhao@gmail.com

***Abstract*—Event-driven applications in wireless sensor networks feature correlated traffic bursts: after a period of idle time with light traffic loads, multiple sensors that have detected the same event have to transmit large amounts of data simultaneously to sink node or cluster head. The demand for simultaneous data transmission often causes severe collision, which is one of the most significant sources of energy consumption in wireless sensor networks. In this paper, we propose SC-MAC (sender-centric MAC), a new asynchronous duty cycle MAC protocol designed for burst traffic loads. SC-MAC achieves collision-free environment while do not introduce extra overhead. In order to minimize delivery latency in tree structure or other multi-hop networks, SC-MAC also introduces a latency optimization mechanism. We show the performance of SC-MAC through *ns-2* simulation and compare it to PW-MAC, the state-of-the-art asynchronous MAC protocol. The simulation results show that SC-MAC significantly minimizes energy consumption and delivery latency.**

***Index Terms*—MAC; burst traffic; single-hop collaboration; energy efficiency**

I. Introduction

Wireless sensor networks have been widely studied in recent years, as they have a wide range of potential applications (e.g., environment monitoring, intrusion detection, medical systems, smart spaces). Such applications, according to data reporting method, can be categorized into event-driven and time-driven applications, along with query-driven [1]. In the time-driven case, sensor nodes periodically sense the environment and transmit the data of interest to the sink at constant periodic time intervals. In the event-driven case, sensor nodes do not transmit any data unless relevant events occur. In query-driven networks, sink sends a request of data gathering when needed.

In this paper, we focus our attention on event-driven applications, which is a general case in our daily life. It is the primary task in event-driven networks that reporting sensed data to the sink, where sensor nodes normally operate under light traffic loads and suddenly activate under burst or high traffic load such as due to random correlated-event traffic [2] or convergecast [3]. In such case, multiple nodes that sensed correlated-event will generate a large volume of data and send their reports to the sink node or to a node that does data aggregation in a short period of time. Because it triggers many nodes simultaneously, severe channel collision will be happened with high energy consumption and latency. Thus, to design an efficient MAC protocol for event-driven applications in wireless sensor networks, sensor nodes must be not only operated with low energy overhead during idle periods but also achieved low latency and high energy efficiency during burst traffic periods.

Most of the existing duty cycle MAC protocols are either synchronous or asynchronous. In synchronous protocols (e.g., DW-MAC [2], S-MAC [4], SCP [5]), sensor nodes wake up only during the common active time intervals to exchange packets. Although substantial energy can be saved, they require strict time synchronization and impose high overhead. In contrast, asynchronous protocols such as B-MAC [6], X-MAC [7], RI-MAC [8] and PW-MAC [9] maintain independent wake up schedule and thus eliminate the overhead for synchronization. These asynchronous MAC protocols are efficient either in light traffic loads or high traffic loads, but not well-suited for event-driven applications with burst traffic.

In B-MAC, prior to transmit DATA, a sender transmits a long preamble and receiver that detects the preamble will stay awake to receive the data. To reduce the length of preamble, X-MAC senders transmit "packetized preambles" and listen for a receiver-generated acknowledgment between packets. Both of them are high energy efficiency under light traffic loads, but their preamble transmissions occupies the wireless medium for a long time until DATA is delivered, making them less efficient in high traffic loads than RI-MAC and PW-MAC. In RI-MAC, receiver uses beacons to indicate that it is awake and ready to receive data. A node with pending DATA to send stays active silently. Upon receiving the beacon from intended receiver, sender starts its DATA transmission immediately. RI-MAC uses receiver-initiated data transmission mechanism to increase channel utilization. However, when a sender has a packet to send, it immediately wakes up to wait for the receiver, leading to a large sender duty cycle due to its idle listening until the receiver wakes up. PW-MAC uses predictive wakeup mechanism to reduce senders energy waste in RI-MAC, which achieve senders to wake up and turn on its radio right before the intended receiver wakes up. It also introduces prediction-based retransmission mechanism to achieve energy efficiency when wireless collisions occur. PW-MAC achieves high channel utilization and low energy consumption in high traffic loads. However, neither RI-MAC nor PW-MAC proposed an efficient method to solve severe channel collision during many-to-one communication in burst traffic loads, which lead to waste energy and increase end to end latency.

In this paper, we present a new asynchronous duty cycling MAC protocol, called Sender-Centric MAC (SC-MAC). SC-MAC aims to avoid channel collision under burst traffic and tries to maximize the energy conservation. On the other hand, SC-MAC performs as well as PW-MAC in other traffic loads. The rest of this paper is organized as follows. In Section II, we present the detailed design of SC-MAC protocol, and Section III presents an evaluation of SC-MAC using *ns-2*. Finally, we conclude this paper in Section IV.

II. SC-MAC Design

In this section, we begin by giving the overview of SC-MAC and then describe the details of SC-MAC.

A. Overview

The goal of SC-MAC is reduce energy consumption caused by multiple contending nodes under burst traffic loads. Fig. 1 shows an overview of the operation of SC-MAC, in which nodes transmit pending DATA based on their own schedule, rather than transmit simultaneously at the receiver's wakeup time. In SC-MAC, each receiver periodically wakes up and broadcasts probe frame to check if there is any potential communication with it. Upon getting any hardware-ACKs (HACK) [10], the receiver concludes there are DATA frames intended for it. Then, it goes to sleep immediately and wakes up to receive DATA based on every intended sender's schedule in this backcast interval. Otherwise, if there is no incoming HACK after broadcasting a probe frame, as receiver does later in Fig. 1, it goes to sleep and does

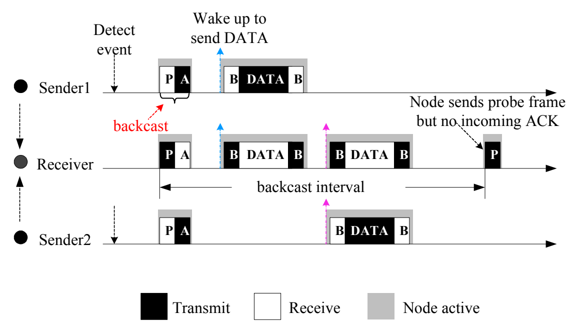


Fig. 1. Overview of SC-MAC.

not wake up until next probe frame generation. Senders in SC-MAC, when an interesting event was detected, wake up and turn on their radio right before the intended receiver wakes up. They acknowledge the probe frame using HACK to notice their intended receiver that there are incoming DATA frames for it. After that, senders go to sleep immediately and wakeup to transmit pending DATA according to their own schedule.

Due to multiple contending senders transmit DATA frames at different time in many-to-one communication, SC-MAC significantly reduces senders duty cycle compared to RI-MAC and PW-MAC. Binary exponential backoff (BEB) strategy used in RI-MAC and PW-MAC not only increase sender duty cycle, but also end-to-end delivery latency. In the extreme case, hard-coded maximum backoff limits will cause receivers to abort when contention is too high [11]. In order to reduce delivery latency further, we also introduce a latency optimization mechanism in Section II-D.

B. Backcast-Based Single-Hop Collaboration

Dutta et al. proposed an acknowledged anycast primitive, called backcast [10], in which a single radio frame transmission triggers zero or more acknowledgment frames that interfere non-destructively at the initiator. They did a range of experiments based on the IEEE 802.15.4-compliant CC2420 radio and the results show that a commodity radio can decode the superposition of at least a dozen identical acknowledgments with greater than 97% probability.

In SC-MAC, we use backcast primitive to achieve single-hop collaborative feedback, in which receivers periodically broadcast a poll message of the form "does there exist any nodes want to communicate with me?". On the other hand, all sensor nodes with pending DATA wakeup right before the intended receiver wakes up and turn on their radios to acknowledge it by HACK. After single-hop collaborative feedback, a sender and its intended receiver find rendezvous time for exchanging

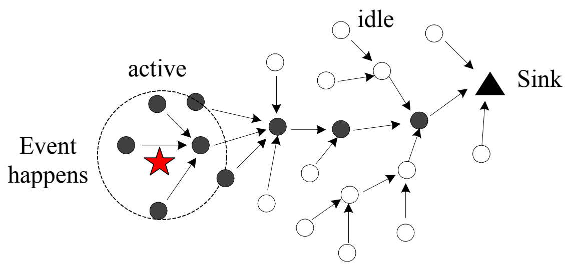


Fig. 2. Event-driven application example.

DATA.

Most energy-efficient duty-cycling MAC protocols use fixed or random interval to achieve periodic listen and sleep. In SC-MAC, receiver periodically broadcasts probe frame, denoted as "P" in Fig. 1, in a fixed interval. It also embeds next wakeup time and sleep interval in the probe frame. After receiving probe frame, sender stores them in a schedule table and can deduce all future backcast time of the intended receiver.

C. Rendezvous Time Determination

For the sake of creating collision-free environment in burst traffic while do not introduce complex operations and extra overhead, senders use pseudo-random wakeup-schedule to find a rendezvous time after transmitting HACK. Although these acknowledgments from contending senders collide at the receiver, this collision is nondestructive, allowing the intended receiver concludes that at least one sender will communicate with it. The receiver goes to sleep immediately and predicts the wakeup times of every potential sender, so that receiver can wake up to send beacon right after senders wake up, significantly reducing sender duty cycle. Although every node in PW-MAC also computes wakeup times using pseudo-random strategy, SC-MAC is different from PW-MAC essentially. In other words, pseudo-random wakeup-schedule in PW-MAC is a receiver-centric mechanism, in which receiver wakeup based on its own schedule and contending senders predict the same wakeup time of receiver and transmit DATA simultaneously, which cause collisions at the receiver.

We use linear congruential generator (LCG) to generate pseudo-random number, since it is simple and efficient in storage. LCG generates a pseudorandom number as follow:

$$X_{n+1} = (aX_n + c) \mod m. \tag{1}$$

Where, $m > 0$ is the modulus, a is the multiplier, c is the increment, and X_n is the current seed; the generated X_{n+1} becomes the next seed. The max period should be equal to backcast interval of the intended receiver, in order to senders wakeup at least once in each receivers backcast period.

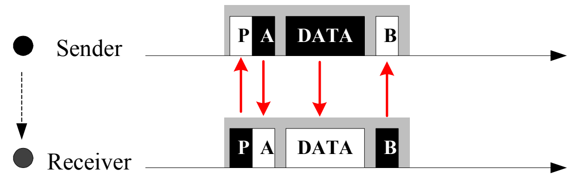


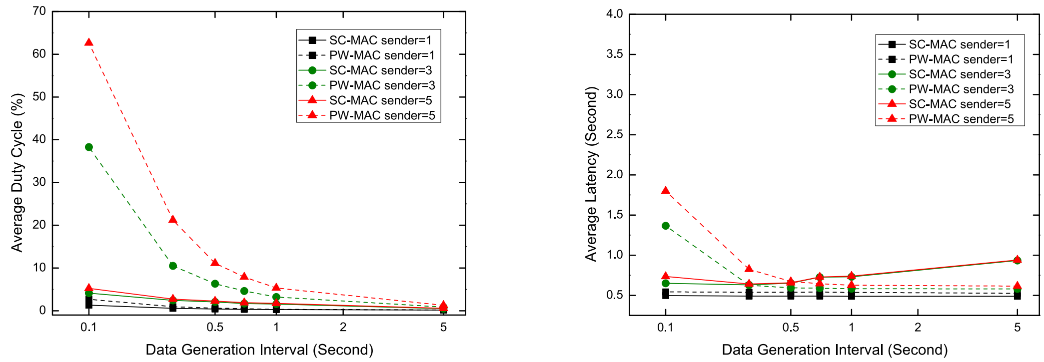
Fig. 3. Operation of SC-MAC with latency optimization mechanism.

In SC-MAC, if a new node joins networks and has pending DATA, it stays active silently while waiting for the probe frame from the intended receiver. Upon receiving the probe frame, the new node stores intended receivers next backcast time and sleep interval, also consider this probe frame as beacon frame. It embeds parameters of its pseudo-random number generator in the pending DATA packet and transmits to the receiver. If multiple new nodes happen to send DATA frames at the same time to a receiver, such transmission collision is resolved using the collision resolution mechanism of PW-MAC, with which the receiver detects the collision and notifies the senders to retransmit their packets after increasing their random backoff windows. After successfully transmission, the intended receiver stores the schedules of all its known neighbors in a table. So senders and the receivers can predict future wakeup times each other.

D. Delivery Latency Optimization Mechanism

As shown in Fig. 2, collision often occurs near the "interesting" event and most of forwarding nodes communicate with their intended receiver without collision. So, when only a sensor node actives, it should transmit DATA immediately rather than find rendezvous time after backcast. We introduce an optimization mechanism to reduce delivery latency. Fig. 3 shows the optimized SC-MAC communication. The sender, upon receiving probe frame from the intended receiver, generates an auto-ack (HACK) and transmits a DATA frame immediately after the HACK. After successfully receiving DATA frames, the receiver goes to sleep and does not predict the wakeup times of all its active neighbors. If receiver failed to receive DATA frames due to collision or channel interference, it goes to sleep and wakes up to receive DATA based on every intended sender's schedule. Because of not receiving ACK beacon for the DATA packet sent, Senders then switch to sleeping state and wake up at their own schedule to retransmit DATA packets, thereby avoiding a large number of collisions.

The duty cycle in PW-MAC is controlled by sleep interval, which is set to a random value between $0.5 \times L$



(a) Average Duty Cycle (b) Average Delivery Latency

Fig. 4. Performance of SC-MAC and PW-MAC in star networks.

and $1.5 \times L$ (suppose a sleep interval of L is used in some WSN). In SC-MAC, the backcast interval is set to a fixed interval of $1 \times L$ so that having the same duty cycle as PW-MAC in idle period. What has caught our attention is the average delivery latency per hop is $0.75 \times L$ in PW-MAC and $0.5 \times L$ in SC-MAC. It is obvious that SC-MAC can reduce $0.25 \times L$ delivery latency per hop compared with PW-MAC.

E. Broadcast and Routing Support

Multihop broadcast-based communication is an important network service in sensor network applications and may be used in routing and resource discovery or in network-wide queries and information dissemination. Since acting as both sender and receiver, forward nodes with SC-MAC periodically wake up based on its own schedule like receiver nodes do. However, sender nodes with SC-MAC wake up only when the intended node transmits a probe packet. This makes it difficult for nodes to perform routing. So a broadcast probe interval should be allocated to sender nodes. Other nodes could either use their backcast interval as broadcast probe interval or set up a new probe interval. All nodes should inform their one-hop neighbors this time schedule in order to achieve broadcast communication from uplink or downlink. Thus, SC-MAC can easily support broadcast DATA frame transmission, either by transmitting the DATA as a unicast transmission to each neighbor of the sender or transmitting the DATA like ADB(Asynchronous Duty-cycle Broadcasting) [12] does.

III. EVALUATION

We evaluated SC-MAC using version 2.29 of the *ns-2* simulator and simulated the Two Ray Ground radio propagation model and a single omni-directional antenna at each sensor node. Table I summarizes the key simulation parameters. We compared SC-MAC and PW-MAC in three types of networks: star networks, chain networks and random networks.

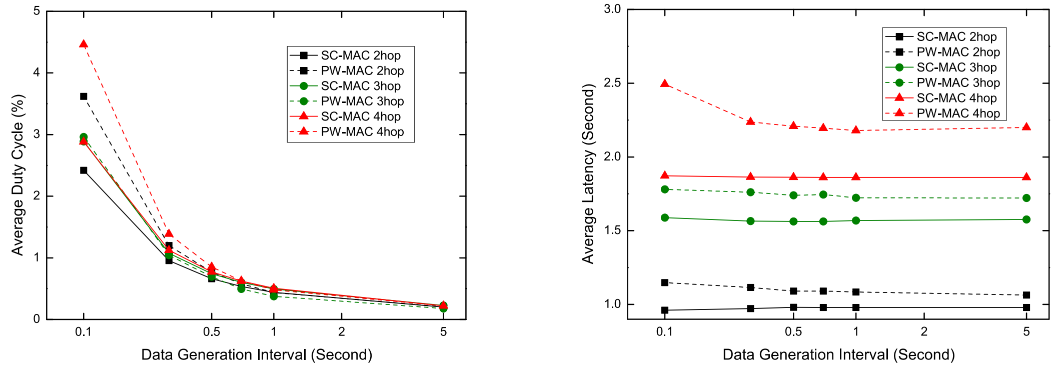
A. Simulation Results in Star Networks

We first evaluated SC-MAC and PW-MAC in star scenario where multiple sensor nodes happen to detect the same event, which can lead to the collisions. We varied the traffic loads by varying the number of contending senders and data generation interval. The simulation lasted 500 seconds and each source node starts to generate packet from 100s to 400s.

The results for star scenario are shown in Fig. 4. Fig. 4(a) shows the duty cycle achieved by SC-MAC and PW-MAC. When there is one sender, both of them achieve high performance. However, with increasing the number of contending senders, the duty cycles with PW-MAC increase quickly to almost 65% when there are 5 contending senders, whereas those with SC-MAC remain under 6% with increasing contending senders. The reason for low duty cycle of SC-MAC is that it greatly reduces the packet collision probability. Although senders in PW-MAC can transmit DATA packets as soon as possible in light traffic loads, as shown in Fig. 4(b), backoffs can degrade the MAC protocols

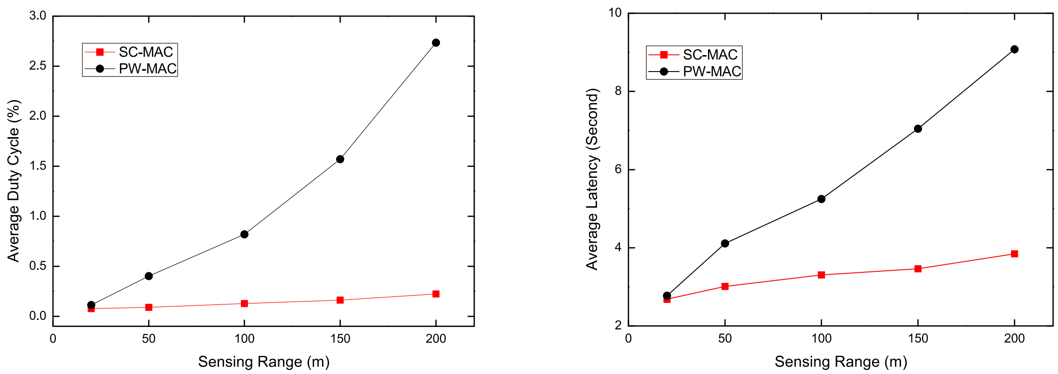
TABLE I
SIMULATION PARAMETERS.

Bandwidth	250 Kbps
SIFS	192 μs
Slot time	320 μs
CCA check delay	128 μs
Tx range	250 m
Carrier sensing range	550 m
Backoff window	0-255
Retry limit	5
Queue size	50 packets



(a) Average Duty Cycle (b) Average Delivery Latency

Fig. 5. Performance of SC-MAC and PW-MAC in chain networks.



(a) Average Duty Cycle (b) Average Delivery Latency

Fig. 6. Performance of SC-MAC and PW-MAC in random networks.

performance when contention is very high, while senders in SC-MAC can transmit queued packets immediately without backoff time under heavy channel contention, which significantly reducing delivery latency.

B. Simulation Results in Chain Networks

This section presents the evaluation of SC-MAC and PW-MAC in multihop networks. There is a single data packet flow with flow length of 1, 2, 3 or 4 hops, respectively. The first node is the source and the last node is the sink. Other traffic parameters is same as parameters in star networks.

The average duty cycles and delivery latency are shown in Fig. 5(a) and Fig. 5(b), respectively. Both of them show very similar duty cycles in chain networks as there are no collision occur. Due to forwarding node send a large volume of data to its downstream node in high traffic loads, the upstream node in PW-MAC may happen to stay awake until receiving the pending beacon from the forwarding node, whereas the upstream node in SC-MAC goes to sleep if it does not receive the probe frame of the forwarding nodes and wakeup at the next backcast time of the forwarding node. So PW-MAC results in a much higher duty cycle than does SC-MAC in high traffic loads. On the other hand, SC-MAC experience lower latency than PW-MAC. This lower latency is mainly because optimization mechanism used in SC-MAC.

C. Simulation Results in Random Networks

In this set of simulations, we compared SC-MAC and PW-MAC in random networks, in which 50 nodes randomly located in a $1000m \times 1000m$ area, and the sink node is at bottom-left corner. RCE model, with sensing range from 20-meter to 200-meter, is used to generate 100 events, one every 20 seconds.

The results for these simulations are shown in Fig. 6. Fig. 6(a) shows the average duty cycles in the RCE model as the sensing range increases and Fig. 6(b) shows end-to-end latency of packets. For the same reasons as discussed above, SC-MAC outperforms PW-MAC in each of both metrics. For example, under the highest burst traffic loads when sensing range is 200m, SC-MAC reduces average duty cycles by 80% compared to PW-MAC and reduces end-to-end latency by 57%.

IV. Conclusion

This paper presents sender-centric MAC (SC-MAC), a new asynchronous duty cycle MAC protocol for wireless sensor networks. SC-MAC can efficiently handle simultaneous any-to-one transmissions by backcast-based collaboration. We evaluated SC-MAC through *ns-2* simulation. Compared to PW-MAC, SC-MAC achieves greater power efficiency and lower delivery latency under a wide range of traffic loads, especially in burst traffic loads. In the future, we aim to evaluate this MAC through measurements of an implementation in TinyOS on MICAz motes.

Acknowledgment

This work was partly supported by the National Science Foundation of China (Grant No.61001104) and Natural Science Foundation of Jiangsu. (Grant No.BK2010166, No.BK2010167 and No.BK2011018).

References

[1] J. Al-Karaki and A. Kamal, "Routing techniques in wireless sensor networks: a survey," *Wireless Communications, IEEE*, vol. 11, no. 6, pp. 6 – 28, dec. 2004.

[2] Y. Sun, S. Du, O. Gurewitz, and D. B. Johnson, "Dw-mac: a low latency, energy efficient demand-wakeup mac protocol for wireless sensor networks," in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, ser. MobiHoc '08, 2008, pp. 53–62.

[3] H. Zhang, A. Arora, Y.-r. Choi, and M. G. Gouda, "Reliable bursty convergecast in wireless sensor networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, ser. MobiHoc '05, 2005, pp. 266–276.

[4] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2002, pp. 1567 – 1576 vol.3.

[5] W. Ye, F. Silva, and J. Heidemann, "Ultra-low duty cycle mac with scheduled channel polling," in *Proceedings of the 4th international conference on Embedded networked sensor systems*, ser. SenSys '06, 2006, pp. 321–334.

[6] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, ser. SenSys '04, 2004, pp. 95–107.

[7] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-mac: a short preamble mac protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems*, ser. SenSys '06, 2006, pp. 307–320.

[8] Y. Sun, O. Gurewitz, and D. B. Johnson, "Ri-mac: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*, ser. SenSys '08, 2008, pp. 1–14.

[9] L. Tang, Y. Sun, O. Gurewitz, and D. Johnson, "Pw-mac: An energy-efficient predictive-wakeup mac protocol for wireless sensor networks," in *INFOCOM, 2011 Proceedings IEEE*, april 2011, pp. 1305 –1313.

[10] P. Dutta, R. Mus?loiu-e, I. Stoica, and A. Terzis, "Wireless ack collisions not considered harmful," in *In HotNets-VII: The Seventh Workshop on Hot Topics in Networks*, 2008.

[11] D. Carlson and A. Terzis, "Flip-mac: A density-adaptive contention-reduction protocol for efficient any-to-one communication," in *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, june 2011, pp. 1 –8.

[12] Y. Sun, O. Gurewitz, S. Du, L. Tang, and D. B. Johnson, "Adb: an efficient multihop broadcast protocol based on asynchronous duty-cycling in wireless sensor networks," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '09, 2009, pp. 43–56.