Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2014, Article ID 576401, 11 pages
http://dx.doi.org/10.1155/2014/576401

Research Article

Altruistic Backoff: Collision Avoidance for Receiver-Initiated MAC Protocols for Wireless Sensor Networks

Xenofon Fafoutis, Charalampos Orfanidis, and Nicola Dragoni

Department of Applied Mathematics and Computer Science, Technical University of Denmark, 2800 Kongens Lyngby, Denmark

Correspondence should be addressed to Xenofon Fafoutis; xefa@dtu.dk

Received 28 October 2013; Revised 20 January 2014; Accepted 7 May 2014; Published 21 May 2014

Academic Editor: Christos Verikoukis

In receiver-initiated medium access control (MAC) protocols for wireless sensor networks, communication is initiated by the receiver node which transmits beacons indicating its availability to receive data. In the case of multiple senders having traffic for a given receiver, such beacons form points where collisions are likely to happen. In this paper, we present *altruistic backoff* (AB), a novel collision avoidance mechanism that aims to avoid collisions before the transmission of a beacon. As a result of an early backoff, senders spend less time in idle listening waiting for a beacon, thus saving significant amounts of energy. We present an implementation of AB for Texas Instruments' eZ430-rf2500 sensor nodes and we evaluate its performance with simulations and experiments.

1. Introduction

Wireless sensor networks (WSNs) consist of multiple embedded networked wireless devices that are characterized by resource and power constraints. The medium access control (MAC) protocol is responsible for the establishment of a communication link between wireless devices. Its primary role is to coordinate access to and transmission over a medium common to several nodes. Furthermore, it plays a key role in the design of energy-efficient WSNs, as it controls the active and sleeping states of a node, known as *duty cycling*. The energy consumption of a wireless sensor node is dominated by the power needs of its radio component [1]. As a result, duty cycling the radio plays a fundamental role towards the realization of energy-efficient wireless sensor networks.

For a communication link to be established, both the receiver and the sender need to be simultaneously in an active state. Here, an important distinction needs to be made. In the case of single-hop star topologies and assuming that the receiver has sufficient energy resources to be continuously in active mode, establishing the link does not constitute a particular challenge. A duty-cycling sender will always find the receiver available to receive traffic. Related work, in this scenario, primarily builds upon the IEEE 802.15.4 standard [2], such as DQ-MAC [3].

In multihop topologies, on the other hand, both the sender and the receiver are duty cycling. This poses a particular problem of finding a rendezvous point between a sender and receiver in which both of the nodes are in an active state and a communication link can be established. In the literature there are three fundamental approaches to address this issue. In protocols that follow a *synchronous* approach, like S-MAC [4], T-MAC [5], and DSMAC [6], to mention only a few, nodes organize the active and sleeping states to overlap with each other. The beacon-enable mode of IEEE 802.15.4 and its extensions (e.g., NCCARQ-WSN [7]) can be also classified as synchronous MAC protocols.

Asynchronous schemes do not require synchronization, as the nodes sleep and wake up independently of the others. This leads to the need of techniques on deciding a rendezvous point for nodes to communicate. There are two fundamental asynchronous techniques, namely, the *sender-initiated* and the *receiver-initiated*. The basic technique used in a *sender-initiated asynchronous MAC scheme* is called preamble sampling, where the sender transmits a preamble to indicate that there is a pending need for communication. The receiver wakes up occasionally into the active state to

listen to such a preamble transmission. Once the preamble is detected, the receiver replies with a positive acknowledgment to the sender when the preamble transmission stops. This establishes a communication link between the sender and receiver. Most notable examples of MAC protocols that are based on the sender-initiated paradigm are WiseMAC [8], B-MAC [9], and X-MAC [10]. A thorough survey of sender-initiated schemes can be found in [11].

This paper focuses on the latter asynchronous approach, the *receiver-initiated* approach. In *receiver-initiated asynchronous MAC protocols*, the sender listens to the channel waiting for small beacons transmitted by the receiver. The receiver transmits the beacons, which are used by the sender to synchronize with the receiver, in accordance to its duty cycle. The receiver-initiated paradigm was originally introduced by Lin et al. in 2004 (RICER [12]) and became popular with RI-MAC [13] in 2008.

Contribution and Outline of the Paper. The key idea behind the receiver-initiated paradigm is that beacons constitute indirect transmission timeslots. Therefore, when multiple nodes contend for the same beacon, a collision is inevitable. Unless there are specific conditions so that receivers can provide the network with much more beacons than the generated data packets, receiver-initiated protocols are particularly vulnerable to collisions. In this paper, we present *altruistic backoff* (AB), a novel energy-efficient collision avoidance mechanism for receiver-initiated MAC protocols. AB manages to decrease the energy wasted in idle listening by detecting and resolving inevitable collisions before the beacon transmission. Additionally, we implemented the protocol on Texas Instruments' eZ430-rf2500 sensor nodes [14] and we evaluate its performance by comparing it to *random backoff* through simulations and experiments.

The remainder of the paper is organized as follows. Section 2 presents the receiver-initiated paradigm of communication along with previous work on collision avoidance. Section 3 summarizes the proposed protocol, AB. Section 4 evaluates the protocol using simulations. Section 5 provides the implementation details and Section 6 presents the experimental results. Lastly, Section 7 concludes the paper.

2. Collision Avoidance in Receiver-Initiated MAC Protocols

In this section, we present the receiver-initiated paradigm of communication between duty-cycling nodes. Furthermore, we briefly survey how existing MAC protocols, that incorporate the receiver-initiated paradigm, address the challenge of collision avoidance. Lastly, we motivate the necessity of our proposed solution by contrasting it with the commonly used approach in terms of energy efficiency.

2.1. The Receiver-Initiated Paradigm. Receiver-initiated MAC protocols use beacons to establish a link between duty-cycling nodes, as sketched in Figure 1. In particular, a node is usually in a sleeping state, in which its radio is turned off. Occasionally, it interrupts its sleep to transmit a small frame,

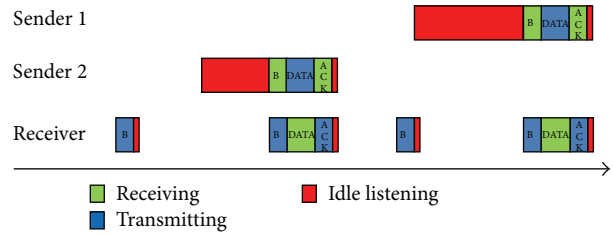

Figure 1: Receiver-initiated paradigm of communication. The sender is passively listening to the channel for a beacon (B) that initiates the communication. *Idle listening* indicates a source of energy consumption where the node is active, receiving, while the other side of the link is in sleeping mode.

called *beacon* (B), which indicates its availability to receive data. After the beacon transmission and for a predefined time, the node waits (with the radio tuned on) for a reply. In case of no reply, the node goes back to the sleeping state. A node with data to transmit interrupts its sleep and passively listens to the channel for a beacon that originates from the intended receiver. Upon reception of a beacon, data transmission follows, typically acknowledged by an additional control frame (ACK). The latter concludes the communication cycle and both nodes go to the sleeping state.

Since the publication of RI-MAC [13], several MAC protocols that build on the receiver-initiated paradigm have been proposed. Such protocols mostly focus on optimizing the performance of the network and/or extending some features, such as mitigating the cost of beaconing (e.g., A-MAC [15]), mitigating the time a node awaits for a beacon (e.g., EE-RI-MAC [16] and PW-MAC [17]), dynamically adapting the duty cycles (e.g., ODMAC [18] and CyMAC [19]), adding broadcasting support (ADB [20] and YA-MAC [21]), and adding multichannel support (DCM [22] and EM-MAC [23]). Despite their differences, all these MAC protocols are based on the same receiver-initiated communication paradigm.

A comparison between the receiver-initiated paradigm and other communication paradigms for duty-cycled nodes is out of the scope of paper. For such comparison, we refer the reader to related works [12, 13, 24].

In the literature, there is a line of MAC protocols that extend the paradigm with techniques to predict the wake-up event of the receiver, such as WideMAC [25] and PW-MAC [17]. Moreover, there are several other extensions to the paradigm that are incompatible with such prediction techniques. For instance, in ODMAC [18], the wake-up events are unpredictable as they are scheduled based on the available energy that can be harvested from the environment. The main disadvantages of the wake-up prediction techniques are related to the requirement to deal with clock drifts in the microcontrollers of the sensor nodes and to the fact that they hinder the ability of sensor nodes to autonomously and individually adapt their duty cycles to various environmental parameters. In such dynamic conditions, the wake-up event of the receiver is continuously changing with respect to the available energy or the network conditions. Furthermore,

MAC protocols typically follow randomization techniques that aim to avoid unwanted synchronizations (for instance, two nodes continuously transmit their beacon simultaneously) [13]. In this paper, we consider the generic version of the receiver-initiated paradigm, in which the wake-up events of the receiver are unknown to the sender.

2.2. Collision Avoidance. Collision avoidance in wireless networks was introduced because collision detection mechanisms, traditionally used in wired networks, are impossible. Detecting a collision while it is happening is not possible in wireless networks, because the radio is not able to transmit and receive simultaneously. Collided transmissions can only be detected by the receiver after their completion. Therefore, in high throughput wireless networks with large data packets, such as IEEE 802.11 [26], collisions lead to a significant throughput degradation.

The solution to this problem was given by avoiding collisions through *Random Backoff* (RB). The idea is that the protocol defines a time interval (*timeslot*) and a contention window (CW). Before transmitting, each node selects a random number, chosen uniformly between zero and CW − 1, and it delays the data transmission by that amount of timeslots while listening to the channel for other transmissions. If the channel remains idle, the data transmission follows. If the channel gets occupied by another transmission, the node freezes the timeslot counter and backs off. When the channel becomes idle again the node unfreezes the timeslot counter and the process is repeated until the counter reaches zero. At this point, the data transmission follows. As a result, unless two transmitters select the same random number, the collision is avoided. The size of CW is associated with a performance tradeoff. If its value is too small, the probability of two nodes selecting the same random number gets high. On the other hand, if its value is too high, the transmitters waste a lot of time in idle listening, leading to protocol overhead and throughput degradation. IEEE 802.11 distributed coordination function (DCF) [26] solves this problem by adapting CW to the level of contention. This mechanism works as follows. CW is initialized with a small value, which is doubled every time a collision occurs (with a maximum limit) and gets back to its minimum value after a successful transmission. This mechanism is called *binary exponential backoff* and results to a low CW in low contention that can quickly increase in the case of traffic bursts.

Receiver-initiated MAC protocols for WSNs inherited the principle of RB from traditional wireless protocols. RI-MAC [13] adopts a variation of BEB. The experiments conducted by the authors of RI-MAC have shown that due to the presence of the capture effect [27] in FM radios, also called cochannel interference tolerance, such a contending scenario does not necessarily lead to collisions. This property shows that the traditional assumption that a packet collision always results in data corruption is false. For this reason, senders in RI-MAC immediately transmit the data upon receiving a base beacon, without any backoff. The receiver listens for a short period of time after transmitting the beacon, known as the dwell time. Dwell time is determined by the current backoff window size. Concurrently, it measures the channel power level and processes the bit pattern received. If a valid data frame header is not detected in time and the measured power level indicates that a transmission is in progress, then this condition is classified as a collision. If a collision occurs, the receiver performs a clear channel assessment (CCA), waiting for the channel to be free. Once a clear channel is determined, the receiver transmits a beacon with a backoff window specified, informing the senders of the failed transmission. The senders that are waiting for an ACK use the backoff window specified in the beacon to perform a random backoff. The senders listen to the channel, while waiting for the random period to expire, before retransmitting the data. If a transmission from another sender is detected, the sender withholds the transmission and waits for an ACK beacon, before resuming with a new random backoff. If a collision happens again, the receiver increments the backoff window using the BEB strategy, until the maximum window size is reached. After that, both senders and receivers accept a failed transmission and go back to sleep, retrying at a later point in time.

In addition to RI-MAC, other receiver-initiated MAC protocols adopt variations of RB, including RICER [12], RC-MAC [28], YA-MAC [21], DCM [22], EM-MAC [23], and IRDT [29]. Intermittent receiver-driven data transmission (IRDT) [29] also incorporates two additional collision avoidance mechanisms. The first is based on the frequency of beacon transmissions. The idea is that by increasing the beacons, the senders are stochastically distributed into more beacons and the collision probability decreases. However, this solution can work only if the receivers are capable of offering their energy resources for forwarding more traffic. The other collision avoidance mechanism is based on data aggregation. By aggregating multiple data packets into larger frames, the total amount of attempted transmissions decreases; thus, the probability of a collision decreases. However, this approach has a negative impact on the delay of each individual data packet. The authors define two methods of collision avoidance with data aggregation: static and dynamic. According to the static method, the protocol uses a constant buffer of n packets. The node keeps collecting packets from other nodes and packets locally generated into the buffer. When the buffer is full, it is transmitted as a single MAC frame. According to the dynamic method, a sender with a single packet to transmit normally waits for a beacon. While waiting, it periodically transmits its own beacons in order to collect packets from neighbors. When the beacon is received, the sender transmits a single frame with as many packets as it managed to collect during that time.

Self-adapting RI-MAC (SARI-MAC) [30] introduces a collision avoidance mechanism through time slot reservation. After the beacon transmission, a contention window period follows during which the nodes pick a uniformly random slot to request for a timeslot reservation. At the end of the contention window, the receiver sends back to all the contending nodes a report with the reservations. Nodes transmit their data in the reserved timeslot, which is long enough for a data packet and the respective acknowledgment. Opportunistic cooperation MAC (OC-MAC) [31] indirectly decreases collisions by allowing senders to opportunistically

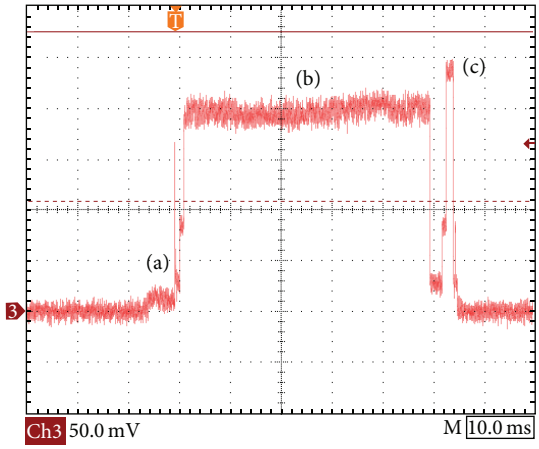

Figure 2: Consumption of a typical cycle based on a study of ODMAC [32]. The current drain is obtained by dividing the shown voltage by the shunt resistor's value (10 Ω). The activity cycle consists of the following actions: (a) sensing and packet generation, (b) waiting for a beacon from the receiver, and (c) transmitting the packet. Power consumption is dominated by the time the radio spends waiting for a beacon, that is, idle listening.

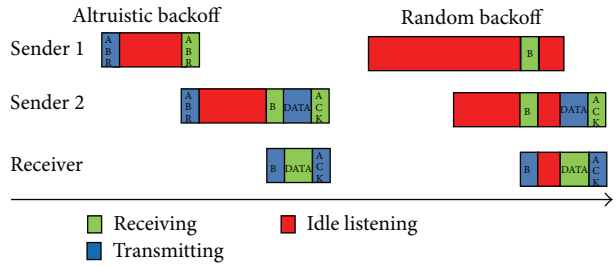

Figure 3: Altruistic versus random backoff. In RB the inevitable collision is resolved after the beacon transmission, while both nodes waste energy in idle listening waiting for it. AB uses control packets (ABR) to resolve the inevitable collision before the beacon allowing the nodes to back off earlier and save energy by decreasing the time they spend in idle listening.

forward traffic to neighbors that happen to be awake at the same time.

The widely adopted RB has its roots in avoiding long concurrent transmissions that decrease the network throughput. WSN are typically low-traffic networks with small frames that give priority to energy-efficiency rather than throughput. Our motivation for AB lies on our previous work on receiver-initiated MAC protocols, which indicates that during a transmission cycle most of the energy is consumed in idle listening, waiting for a beacon to establish a connection [32]. The cost of the data transmission itself is insignificant compared to idle listening (see Figure 2). RB implies that senders that contend for the same beacon will spend a vast amount of energy waiting for the beacon and the collision will be detected and resolved only after the beacon transmission. On the other hand, AB aims to detect the inevitable collision before the beacon transmission and allow the contending nodes to back off earlier and save energy.

3. Altruistic Backoff (AB)

Altruistic backoff is a collision avoidance mechanism that detects potential collisions and avoids them before the actual beacon transmission. Upon a wake-up event, it transmits a control packet, named ABR (*altruistic backoff request*), that identifies the beacon the node is waiting for. A node that is already waiting for the same beacon and receives this packet, altruistically backs off, offering the beacon to the node that wakes up last. At the low overhead of one extra control packet transmission per data packet transmission, collisions are mitigated and idle listening is significantly reduced. Figure 3 shows an example of AB collision avoidance compared to RB.

The presented backoff scheme does not suffer from fairness issues for two reasons. First, WSNs consist of cooperative nodes that do not have incentives to overutilize the channel. Furthermore, random channel access provides similar probabilities for all nodes to use the beacon. Essentially, the beacon and thus the channel are taken by the sender that wakes up last. Therefore, random channel access guarantees long-term fairness. In other words, as long as different senders have equal opportunities to wake up last, they have equal opportunities to take the beacon. Similarly to RB, long-term fairness can be compromised if nodes do not follow the protocol. In particular, if a sender continuously retransmits an ABR, it will always get the beacon. Generally, we do not consider this a problem, because WSNs are networks of cooperative nodes that do not have incentives to favor their performance against the performance of other nodes. However, this property is a security vulnerability that can lead to denial-of-service (DoS) attacks. Off-the-shelf security protocols, such as the receiver authentication protocol (RAP) [33], can be used to authenticate control packets in an energy-efficient manner and secure the protocol against such attacks.

Beyond being a security vulnerability, this property is used for quality of service (QoS) services through traffic differentiation. Traffic differentiation is valuable in case of applications that generate traffic of different urgency (e.g., alerts versus monitoring traffic). We define two types of data packets that correspond to two traffic classes, the high-priority class and best-effort class. The priority number that defines the priority class is included in the ABR. Upon the reception of an ABR, a node compares the priority number indicated in the ABR to the priority number of the local packet it has to transmit. If and only if the local packet belongs to the high-priority traffic class and the remote packet belongs to the best-effort traffic class, the node immediately transmits a new ABR to retake the beacon, as shown in Figure 4. As a result, the priority number guarantees that ABR retransmissions occur only when a node has a higher priority than the node that currently has the beacon.

Upon a backoff event, the time of a next transmission attempt can follow different policies with respect to the importance of the data. We can consider two extremes. On one hand, the sender might attempt to transmit immediately, as recommended for traffic of high priority. On the other

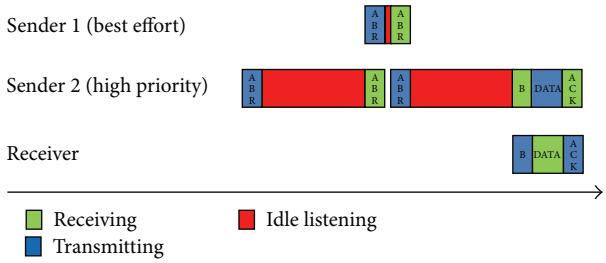

Figure 4: Traffic differentiation with AB. Nodes with traffic of high priority, upon being silenced by nodes with lower priority, immediately retransmits an ABR to retake the beacon.

hand, the sender might choose to buffer the packet and transmit it together with the following packet. We recommend this policy for best-effort traffic, as it is the policy that minimizes the energy consumption. Additionally, the sender might choose a solution between that compromises the advantages and the disadvantages of the two extremes. For the remainder of the paper and unless stated otherwise, we assume the use of the second policy.

AB is also able to support congestion control services with no additional overhead. Such feature has particular interest in adaptive protocols, such as ODMAC [18], that regulate the generated traffic to the energy resources of individual nodes. For example, consider a solar energy harvesting scenario where the receiver is placed in shadow (thus, unable to receive and forward much traffic) and the senders are placed in direct sunlight (thus, able to generate much traffic). In this scenario, the receiver would generate beacons at a low frequency and the senders would exchange many ABR frames, while contending for these few beacons. Senders may interpret frequent ABR frames, as a signal that the channel is congested and reduce the rate they generate data to avoid flooding the receiver.

4. Evaluation of AB through Simulations

In this section, we evaluate the proposed collision avoidance mechanism, AB, by comparing it with RB. The key difference between the two mechanisms lies in the way the collision is avoided. Having energy efficiency as our metric of interest, we focus the comparison on how much time the nodes spend on idle listening. In the case of AB, idle listening is the time a sender waits for a beacon. In the case of RB, idle listening is the time a sender waits for a beacon plus the number of timeslots it waits afterwards. We consider two variations of RB, namely, *constant backoff* (CB) and *binary exponential backoff* (BEB). In CB, the CW is fixed to a constant value (cw). In BEB, CW follows the binary exponential approach and cw represents the minimum contention window (CW_{min}).

We model and simulate the two methods as follows. We consider one single receiver that transmits beacons at a set frequency and a set of n nodes that are using these beacons to send their data. A round consists of the time between two beacon transmissions. Every round, each node has a probability to generate data that is equal to the ratio of the

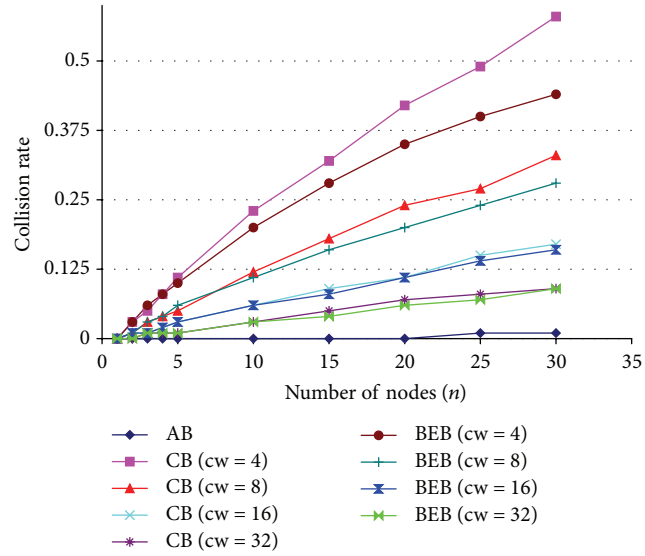

Figure 5: Collision rate for AB and random backoff with constant or binary exponential CW. In the case of BEB, cw represents the minimum contention window.

beaconing period of the receiver over its local sensing period. Nodes with data wake up at a random time during the round. The time from the beginning of a wake-up event until the reception of the following beacon or ABR is considered idle listening. In the case of AB, a collision happens when two nodes transmit the ABR at the same time frame. In the case of RB, a collision happens when two or more senders select the same and lowest random number. We set the duration of the timeslot at 100 μs and the maximum CW_{max} at 64. The simulations are conducted in MATLAB.

At the beginning we fix the beaconing period of the receiver (BP) to 4 seconds and the transmission attempt period of the receivers (SP) to 20 seconds. As a result, an average of 1/5 of the nodes in the network are contending for the channel in each round. Figure 5 shows the collision rate of the different schemes (calculated after 10000 rounds). BEB is preventing more collisions than CB for low contention windows (cw), but the difference decreases as the cw increases. This happens because as the cw increases, the probability of two or more nodes selecting the same random number decreases and, as a result, the need to double the contention window decreases. The same phenomenon appears when the number of nodes is low. AB appears more able to avoid collisions. This happens because of the random channel access. In other words, a collision can happen only if two or more nodes send an ABR simultaneously. Therefore, the time between two sequential beacons acts equivalently to a very large contention window. As a result, we expect that as we increase the contention window, the performance of BEB and CB will approach the performance of AB.

Figure 6 shows the average idle listening per transmission attempt on the same simulation. Notice that CB and BEB show a constant behavior that does not increase with neither the number of nodes nor with the contention window. The average idle listening is equal to half the period of beaconing

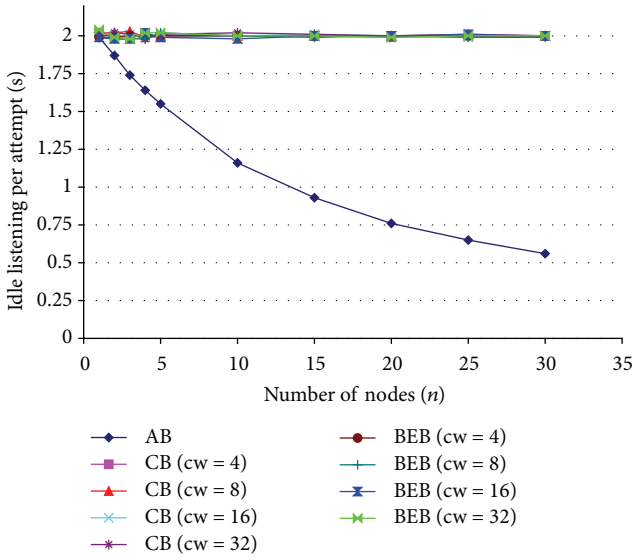

Figure 6: Average Idle listening per transmission attempt for AB and random backoff with constant or binary exponential CW. In the case of BEB, cw represents the minimum contention window.

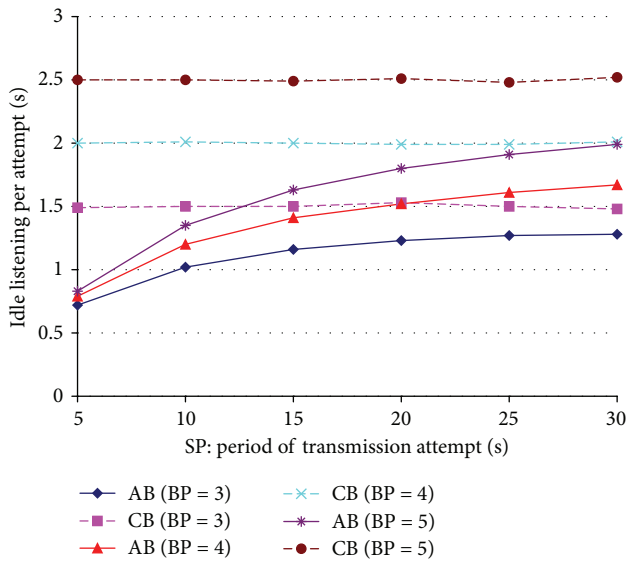

Figure 7: Average Idle listening per transmission attempt for AB and random backoff with constant CW. BP represents the beaconing period of the receiver in seconds. SP represents the period of a transmission attempt.

(BP/2). Intuitively, we expect the idle listening to increase as the contention window increases, as the number of timeslots is expected to be higher. However, the results indicate that the impact of increasing the contention window is insignificant. This behavior is explained by the size of the timeslot (100 μs) which is several orders of magnitude lower than the expected time a sender waits for a beacon. The figure shows that, in the case of AB, the average time the sender spends in idle listening decreases as the number of nodes increases. The more contention, the more ABR frames are transmitted and the faster contending nodes back off. Notice that the average idle listening for AB becomes half the period of beaconing (BP/2) when there is no contention ($n = 1$).

The above results indicate that to study idle listening is sufficient to consider only one version random backoff. In Figure 7, we consider 5 contending senders and CB with fixed contention window (cw = 4). We variate the period of a transmission attempt (SP) and the period of beaconing (BP). The results show a similar constant behavior for CB, while the average idle listening of AB decreases as the traffic increases (SP decreases).

Figure 8 shows the distribution of successful transmissions over all the contending nodes, considering $n = 20$, BP = 4 s, and SP = 20 s, for the case of AB. We can observe that random channel access leads to equal probabilities for every node to be the last sender to wake up before the beacon. Therefore, AB provides long-term fairness for channel access.

Next, we demonstrate the ability of AB to differentiate traffic to provide QoS. We consider two classes of traffic, namely, *high priority* and *best effort*. Sensor nodes mark the data that they generate as *high priority* with a probability P. According to the protocol specification, the sensor node that wakes up last and has a data packet marked as *high priority* takes the beacon. If there is no sensor node with *high priority*

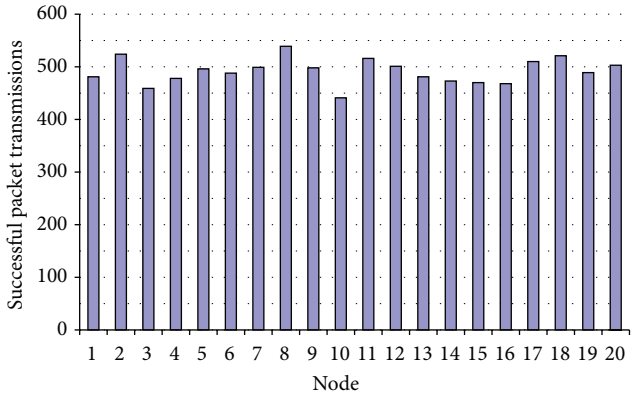

Figure 8: The distribution of successful transmissions indicates that AB provides long-term fairness, as contending nodes have equal probabilities to use the channel.

data packets contending for the medium, the sensor node that wakes up last and has a data packet marked as *best effort* takes the beacon. For the following simulation, we consider $P = 0.05$, BP = 1 s, and SP = 3 s. Figure 9 shows the average ratio of the amount of data packets that take a beacon over the total amount of generated packets, for each priority class (calculated after 10000 rounds). As the contention increases, a larger amount of *best effort* traffic backs off, while the *high priority* traffic is less affected. Essentially, AB sacrifices less important traffic to prioritize urgent traffic. The slight decreasing trend for the *high priority* traffic is attributed to the rounds that multiple nodes with *high priority* traffic contend with each other.

The results indicate that AB is long-term fair and scales well with high contention, as the ABR frames efficiently put

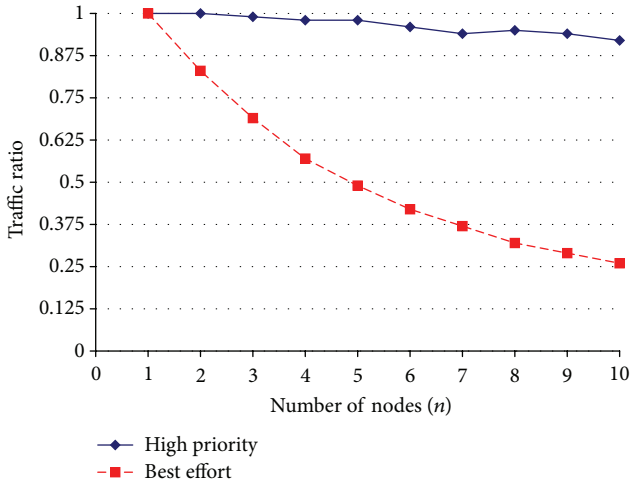

FIGURE 9: The average ratio of the amount of data packets that take a beacon over the total amount of generated packets for each priority class. As the contention increases, the protocol sacrifices *best effort* traffic for *high priority* traffic.

the contending nodes to sleep early and less energy is wasted in idle listening. Furthermore, AB is able to differentiate traffic to provide QoS.

5. Implementation of AB

We implemented AB as an extension to the implementation of on demand MAC (ODMAC) [32] for Texas Instruments' eZ430-rf2500 nodes [14]. The nodes consist of an MSP430 microcontroller (MCU) and a CC2500 radio, operating in the 2.4 GHz band. ODMAC is a receiver-initiated MAC protocol for energy harvesting-wireless sensor networks (EH-WSNs) that has been developed upon the principle of dynamically adapting the duty cycles to the amount of harvested energy. Nevertheless, the basic communication scheme of ODMAC follows the receiver-initiated paradigm of communication, as described in Section 2. We refer the reader to [32] for the details of implementation of the protocol. ODMAC is implemented as a finite state machine (FSM). Its functionality is mainly based upon two routines, namely, *send* and *receive*. Unless one of these two handlers is invoked, ODMAC is in sleeping state and the radio is off. The *send* routine generates and formats a packet around the payload (i.e., the result of a sensing operation). When the packet is ready, the radio is switched on into listening mode and the state machine awaits for an interrupt signaling the reception of an appropriate beacon. Should this happen, ODMAC continues its execution and the data packet is transmitted. At the end of a packet transmission, the radio is switched off.

AB extends the ODMAC *send* routine as follows. After the packet generation, an ABR frame that includes information about the intended receiver is generated. After a successful CCA the transmission of ABR follows. Then, the radio is switched to listening mode and the sender begins to listen for a beacon. Listening is interrupted either by the reception of the expected beacon or by the reception of an ABR that indicates interest for the same beacon. In the former case, data transmission follows normally. In the latter case, the routine returns and indicates a backoff. It should be noted that the *send* routine performs one attempt to transmit the packet. In case of backoff, the higher layer is free to decide at which point in the future will attempt again to transmit the same packet. The state machine in Figure 10 summarizes the operation of AB as part of the ODMAC protocol.

For the traffic differentiation services of AB, we extend the implementation by adding a priority bit in the header of ABR control packets. The priority bit indicates if the data packet is classified as *high priority* or *best effort*. When a sender that waits for a beacon receives another ABR packet, it compares its local priority bit with the received priority bit. If and only if the local data packet is classified as *high priority* and the received ABR indicates a *best effort* data packet, the sender retakes the channel by invoking the *send* routine again.

6. Experimental Results

In this section, we experimentally evaluate AB. For the purposes of a comparison with RB, we also implemented a simple variation of the protocol with constant contention window, CB, in the ODMAC protocol. We chose to implement the CB variation because our simulations (see Section 4) indicate that the variation of the protocol and the length of the contention window do not affect the idle listening overhead significantly. CB is implemented by adding a random delay between the reception of a beacon and the transmission of the data. In particular, we use a constant contention window (cw = 4) and a timeslot of 100 MCU cycles (≈100 μs).

To measure the idle listening time interval, we use the internal timer unit, which is set to use the low frequency oscillator (12 KHz) that remains active when the MCU goes into low power (i.e., sleeping) modes. Because of the size of its counter register (16 bits), the timer is able to measure time intervals up to approximately 5.5 seconds. Each node is set to keep the sum of all the time it spent in idle listening since reset and reports the value in every data packet. In addition to that, a sequence number of all the data transmission attempts are also reported. Using the two aforementioned values, we can estimate the average time a node spent in idle listening per data transmission attempt.

For the experiments presented in this section, we use the following test bed. We use a single-hop star topology with a set of senders contending to transmit to a single receiver. The contending senders are placed physically close to each other and to the receiver, in order to mitigate any packet losses due to channel errors. The receiver is connected to a laptop, through which we collect all the received packets. The receiver node is set to transmit beacons but never generate data of its own. A set of sender nodes are configured to periodically transmit data to the receiver. We use ODMAC's randomization feature [32] to randomize the period of data transmission attempts and enforce random channel access. In particular, after each data transmission, the wake-up interrupts are randomized over the whole space of the register. The node then calls the *send* routine once every *sm* wake-up

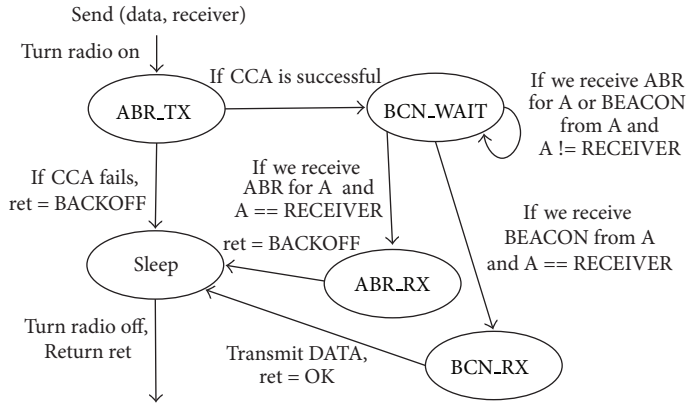

FIGURE 10: The finite state machine that specifies the operation of AB as part of the ODMAC protocol.

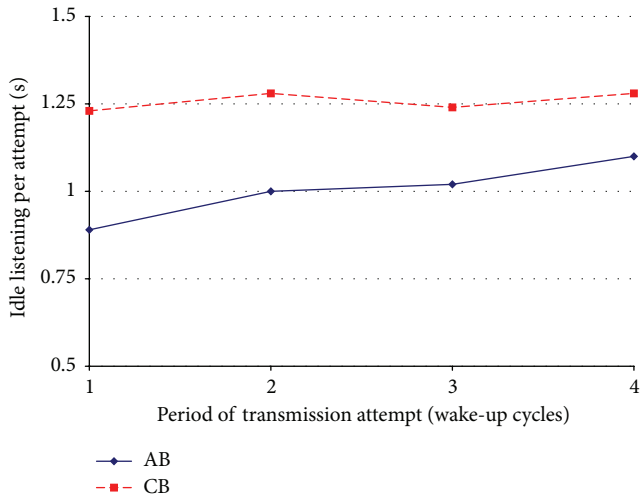

FIGURE 11: Average idle listening per transmission attempt for ab and random backoff with constant CW. Wake-up interrupts are uniformly randomized after each transmission to enforce random channel access.

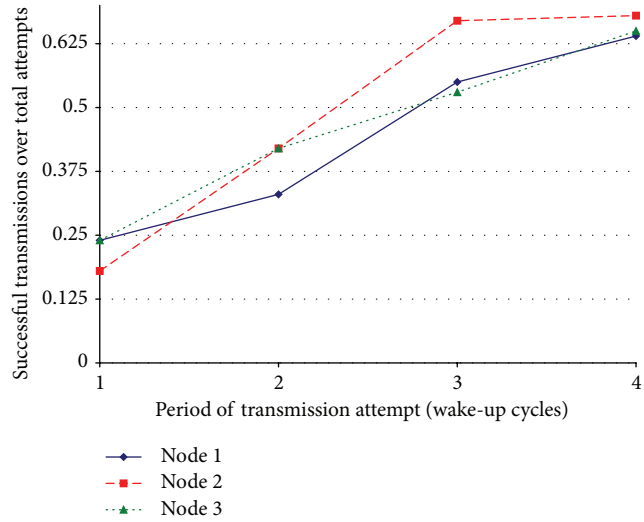

FIGURE 12: The ratio of successful transmissions over the total number of transmission attempts indicates that AB is long-term fair.

interrupts, where *sm* is a configurable parameter that controls the average period of data transmission attempts.

In the experiment shown in Figure 11, we set the beaconing period of the receiver to 4 seconds and we used 3 contending senders. In the x-axis we variate the period of a transmission attempt for all the senders in wake-up interrupts, that is, the *sm* parameter. The duration of each experiment was 1 hour. The results indicate a similar trend to the respective simulation experiment, shown in Figure 7, which verifies the energy consumption improvements of AB. CB follows a similar constant behavior. AB, on the other hand, is spending less time in idle listening as the traffic increases. Figure 12 shows the ratio of successful transmissions over the total number of transmission attempts for the same experiments for AB. The results demonstrate the long-term fairness of the protocol, as the nodes appear to have equal opportunities to take the channel. We can notice that none of the senders is led to starvation and the number of times they took the channel is at the same order of magnitude between the three nodes. The relative difference between the senders is attributed to the duration of the experiment (1 hour). We expect longer experiments to smooth such differences out.

In the next experiment, we fix the period of transmission attempts to 2 wake-up cycles and we variate the number of contending nodes from 1, that is, no contention, to 4. Figure 13 shows the average time each node spends on idle listening per transmission attempt for the two protocols. The duration of each experiment was 1 hour. The results follow a similar trend to the respective simulation experiment, shown in Figure 6. In particular, when there is no contention, the two protocols have similar performance. For the case of CB, the average time spent in idle listening remains constant, being dominated by the time the node waits for a beacon. In the case of AB, on the other hand, idle decreases as the contention increases.

Next, we evaluate the long-term fairness of AB in the scenario of contending senders with different traffic generation

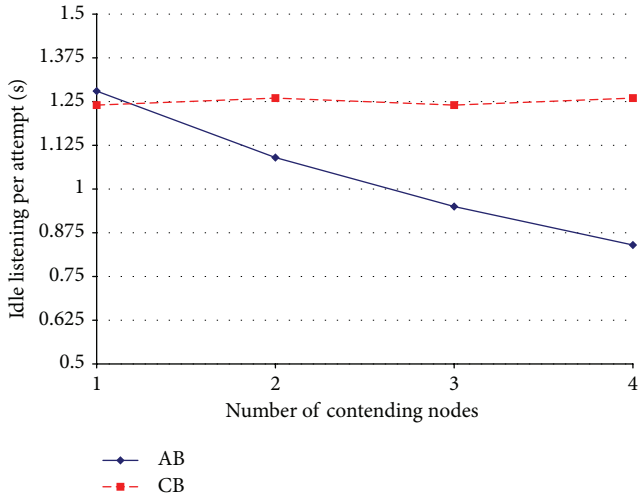

FIGURE 13: Average idle listening per transmission attempt for ab and random backoff with constant CW for different numbers of contending nodes.

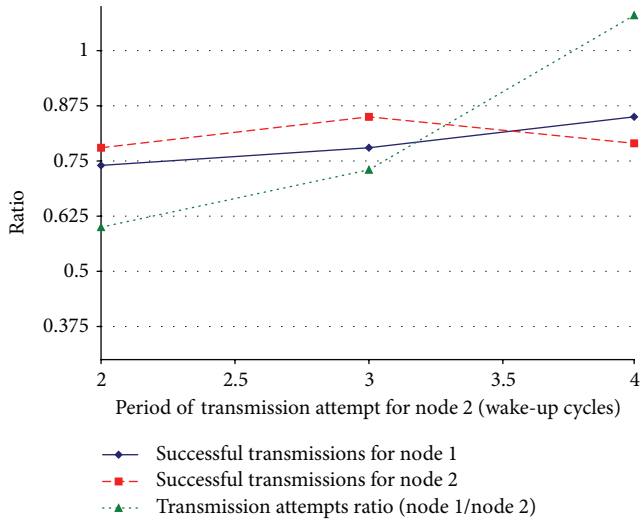

FIGURE 14: The ratio of successful transmissions over the total number of transmission attempts for node 1 and node 2. The period of transmission attempts for node 1 is fixed to 4 wake-up cycles. The triangle-line shows the ratio of the packets generated by node 1 over node 2.

frequencies. Such scenario has interest in cases of nodes with different forwarding duties or different power resources (e.g., energy harvesting sensor nodes have access to different levels of ambient energy). The experiment is designed as follows. We use 2 nodes and fix the period of transmission attempts of the first node to 4 wake-up interrupts, while varying the period of the second node from 2 to 4. The duration of each experiment is 2 hours. Figure 14 shows the results of the experiment. The triangle-line shows the ratio of the packets generated by node 1 over node 2, which increases as the period of transmission attempt of node 2 increases. Note that, when the nodes have equal periods, the ratio is close to 1. We observe that, despite the fact that the two nodes attempt to

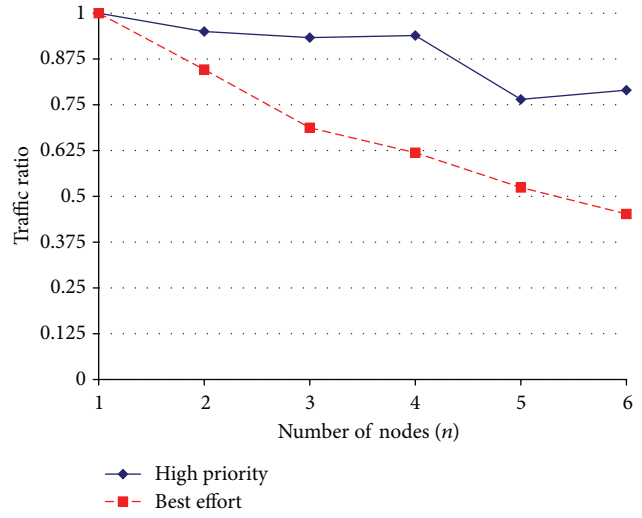

FIGURE 15: The average ratio of the amount of data packets that take a beacon over the total amount of generated packets for each priority class. As the contention increases, the protocol sacrifices *best effort* traffic for *high priority* traffic.

use the channel at different frequencies, they maintain equal opportunities to obtain the beacon. The ratio of success full packet transmissions over the total amount of transmission attempts shows a constant behavior.

In the next experiment, we experimentally evaluate traffic differentiation by replicating the simulation shown in Figure 9. The beaconing period of the receiver is set to 1 second and the period of transmission attempts of the senders is randomized with an average of approximately 3 seconds. Moreover, nodes generate *high priority* data packets with a probability of $P = 0.05$. Figure 15 shows the average ratio of the amount of data packets that take a beacon over the total amount of generated packets, for each priority class. The duration of each experiment is 1 hour. Due to hardware constraints, the experiment was conducted with up to 6 contending nodes. The results validate the simulations and show that as the contention increases, a larger amount of *best effort* traffic backs off, giving priority to the *high priority* traffic.

In the last figure, we validate the simulations by comparing their estimations to the results obtained through the experimental evaluation. In particular, we configure the simulator to the exact same configuration that is used in the test-bed experiment presented in Figure 13. In the experiment the period of transmission attempts of the senders is set to 2 wake-up cycles that are uniformly randomized over the whole space of the register, leading to an average period of approximately 5.5 seconds. Thus, in the simulator we set period of transmission attempts to 5.5 seconds. The beaconing period of the receiver is set to 3 seconds. Figure 16 plots the ratio of the average idle listening per transmission attempt of AB over CB as obtained from the simulation and the test-bed experiment. Observe that both simulations and test-bed experiments give close results, while the behavior of the protocol follows the same trend. The difference indicates that, in the experiments, random access is not as uniformly

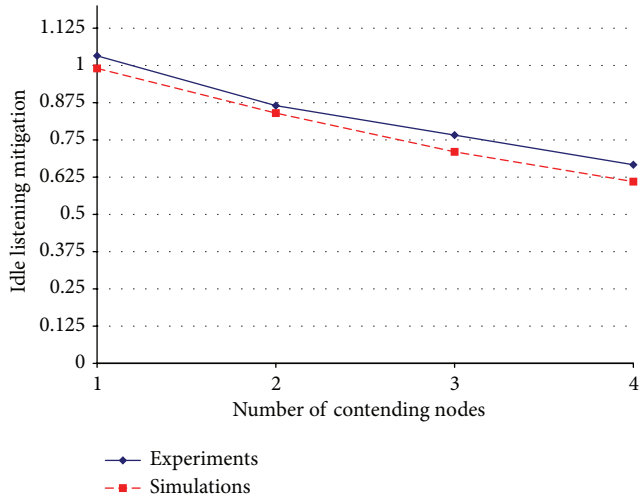

FIGURE 16: The ratio of the average idle listening per transmission attempt of AB over CB as obtained from the simulations and the test-bed experiments.

distributed throughout the interval between two beacons, as assumed in the simulations.

The results of the experiments verify the trends that are suggested by the simulations, presented in Section 4. AB scales well with both high contention and high traffic and provides equal opportunities for the contending nodes to access the channel. Detecting the inevitable collisions before the beacon transmission allows the nodes to resolve the collision before significant amount of energy is wasted in idle listening while waiting for the beacon.

7. Conclusion

In this paper, we have focused on receiver-initiated MAC protocols in wireless sensor networks. Such protocols initiate the data exchange with a beacon that is transmitted by the receiver and states its availability to receive traffic. Beacons nullify the benefits of random channel access, as they constitute points of potential collisions even in situations of sparse traffic. We have proposed AB, a collision avoidance mechanism that exploits random channel access to avoid collisions while decreasing the time nodes waste energy in idle listening. Simulations and experiments indicate that AB is long-term fair and scales well with increasing levels of contention. Furthermore, AB provides QoS by prioritizing traffic of different urgencies. AB is compared to the commonly used collision avoidance mechanism, namely, random backoff, and the results demonstrate the energy savings that can be achieved with AB. Finally, we have discussed an implementation of the proposed collision avoidance mechanism for Texas Instruments' eZ430-rf2500 sensor nodes [14], incorporated in the ODMAC protocol [32].

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] A. Bachir, M. Dohler, T. Watteyne, and K. K. Leung, "MAC essentials for wireless sensor networks," *IEEE Communications Surveys and Tutorials*, vol. 12, no. 2, pp. 222–248, 2010.

[2] IEEE, "IEEE std. 802.15.4-2003: wireless medium access control (mac) and physical layer (phy) specifications for low rate wireless personal area networks (lr-wsns)," 2003.

[3] B. Otal, L. Alonso, and C. Verikoukis, "Design and analysis of an energy-saving distributed mac mechanism for wireless body sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, article 10, 2010.

[4] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications (INFOCOM '02)*, vol. 3, pp. 1567–1576, June 2002.

[5] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, vol. 5–7, pp. 171–180, ACM, November 2003.

[6] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '04)*, vol. 3, pp. 1534–1539, IEEE, March 2004.

[7] A. Antonopoulos and C. Verikoukis, "Network-coding-based cooperative ARQ medium access control protocol for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 601321, 9 pages, 2012.

[8] A. El-Hoiydi and J.-D. Decotignie, "WiseMAC: an ultra low power MAC protocol for multi-hopWireless sensor networks," in *Proceedings of the 1st International Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS '04)*, pp. 18–31, 2004.

[9] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd ACM International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 95–107, ACM, November 2004.

[10] 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 ACM International Conference on Embedded Networked Sensor Systems (SenSys '06)*, pp. 307–320, ACM, November 2006.

[11] C. Cano, B. Bellalta, A. Sfairopoulou, and M. Oliver, "Low energy operation in WSNs: a survey of preamble sampling MAC protocols," *Computer Networks*, vol. 55, no. 15, pp. 3351–3363, 2011.

[12] E.-Y. A. Lin, J. M. Rabaey, and A. Wolisz, "Power-efficient Rendez-vous schemes for dense wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '04)*, vol. 7, pp. 3769–3776, June 2004.

[13] 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 Networked Sensor Systems (SenSys '08)*, pp. 1–14, ACM, November 2008.

[14] Texas Instruments, "eZ430-RF2500 Development Tool," SLAU227E, 2009, http://www.ti.com/lit/ug/slau227e/slau227e.pdf.

[15] P. Dutta, S. Dawson-Haggerty, Y. Chen, C.-J. M. Liang, and A. Terzis, "A-MAC: a versatile and efficient receiver-initiated

link layer for low-power wireless," *ACM Transactions on Sensor Networks*, vol. 8, no. 4, article 30, 2012.

[16] Y.-T. Yong, C.-O. Chow, J. Kanesan, and H. Ishii, "EE-RI-MAC: an energy-efficient receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," *International Journal of Physical Sciences*, vol. 6, no. 11, pp. 2633–2643, 2011.

[17] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "PW-MAC: an energy-efficient predictive-wakeup MAC protocol for wireless sensor networks," in *Proceedings of the 30th IEEE Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '11)*, pp. 1305–1313, IEEE, April 2011.

[18] X. Fafoutis and N. Dragoni, "ODMAC: an on-demand mac protocol for energy harvesting—wireless sensor networks," in *Proceedings of the 8th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN '11)*, pp. 49–56, ACM, November 2011.

[19] Y. Peng, Z. Li, D. Qiao, and W. Zhang, "Delay-bounded MAC with minimal idle listening for sensor networks," in *Proceedings of the 30th IEEE Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '11)*, pp. 1314–1322, April 2011.

[20] 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 (SenSys '09)*, pp. 43–56, ACM, November 2009.

[21] P. Yadav and J. A. McCann, "YA-MCA: handling unified unicast and broadcast traffic in multi-hop wireless sensor networks," in *Proceedings of the 7th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS '11)*, pp. 1–9, IEEE, June 2011.

[22] J. Li, D. Zhang, and L. Guo, "DCM: a duty cycle based multi-channel MAC protocol for wireless sensor networks," in *Proceedings of the IET International Conference on Wireless Sensor Network (IET-WSN '10)*, pp. 233–238, November 2010.

[23] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "EM-MAC: a dynamic multichannel energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 12th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '11)*, p. 23, ACM, May 2011.

[24] X. Fafoutis and N. Dragoni, "Analytical comparison of MAC schemes for energy harvesting—wireless sensor networks," in *Proceedings of the 9th International Conference on Networked Sensing Systems (INSS '12)*, IEEE, June 2012.

[25] J. Rousselot, A. El-Hoiydi, and J.-D. Decotignie, "WideMac: a low power and routing friendly MAC protocol for ultra wideband sensor networks," in *Proceedings of the IEEE International Conference on Ultra-Wideband (ICUWB '08)*, vol. 3, pp. 105–108, September 2008.

[26] IEEE, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Standards Association Std., 2012.

[27] K. Whitehouse, A. Woo, F. Jiang, J. Polastre, and D. Culler, "Exploiting the capture effect for collision detection and recovery," in *Proceedings of the 2nd IEEE Workshop on Embedded Networked Sensors (EmNetS-II '05)*, pp. 45–52, IEEE, May 2005.

[28] P. Huang, C. Wang, L. Xiao, and H. Chen, "RC-MAC: a receiver-centric medium access control protocol for wireless sensor networks," in *Proceedings of the IEEE 18th International Workshop on Quality of Service (IWQoS '10)*, pp. 1–9, June 2010.

[29] D. Kominami, M. Sugano, M. Murata, and T. Hatauchi, "Energy-efficient receiver-driven wireless mesh sensor networks," *Sensors*, vol. 11, no. 1, pp. 111–137, 2011.

[30] Q. Lampin, D. Barthel, I. Augé-Blum, and F. Valois, "SARI-MAC: the self adapting receiver initiated MAC protocol for wireless sensor networks," in *Proceedings of the IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '12)*, pp. 12–18, IEEE, October 2012.

[31] X. Wang, X. Zhang, G. Chen, and Q. Zhang, "Opportunistic cooperation in low duty cycle wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '10)*, pp. 1–5, IEEE, May 2010.

[32] X. Fafoutis, A. D. Mauro, and N. Dragoni, "Sustainable medium access control: implementation and evaluation of ODMAC," in *Proceedings of the IEEE International Conference on Communications Workshops (ICC '13)*, IEEE, 2013.

[33] A. D. Mauro, X. Fafoutis, S. M. Mödersheim, and N. Dragoni, "Detecting and preventing beacon replay attacks in receiver-initiated MAC protocols for energy efficient WSNs," in *Proceedings of the 18th Nordic Conference on Secure IT Systems (NordSec '13)*, 2013.