Efficient and Reliable MAC-Layer Broadcast for IEEE 802.15.4 Wireless Sensor Networks

Oana Iova, Fabrice Theoleyre, Mengchuan Zou, Jialiang Lu

▶ To cite this version:

Oana Iova, Fabrice Theoleyre, Mengchuan Zou, Jialiang Lu. Efficient and Reliable MAC-Layer Broadcast for IEEE 802.15.4 Wireless Sensor Networks. IFIP/IEEE Wireless and Mobile Networking Conference (WMNC), May 2014, Villamoura, Portugal. 10.1109/WMNC.2014.6878876 . hal-01206369

HAL Id: hal-01206369
https://hal.science/hal-01206369

Submitted on 24 Nov 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Efficient and Reliable MAC-Layer Broadcast for IEEE 802.15.4 Wireless Sensor Networks

Oana Iova, Fabrice Théoleyre
CNRS, ICUBE
University of Strasbourg, France
email: {otiova, theoleyre}@unistra.fr

Mengchuan Zou, Jia-liang Lu
Computer Science & Engineering
Jiaotong University, Shanghai, China
email: {ddeggzmc, jialiang.lu}@sjtu.edu.cn

***Abstract*—IEEE 802.15.4 represents a widely used MAC-layer standard for Wireless Sensor Networks. In multihop topologies, the protocol exploits a cluster-tree and organizes the transmissions by alternating sleeping and active periods in a superframe delimited by beacons. In this paper, we propose a new Contention Broadcast Only Period to limit beacon collisions and to reduce bandwidth wastage due to variable beacon durations. We adopt a CSMA-approach during the Contention Broadcast Only Period to efficiently deliver both beacon and broadcast packets. We also propose to use broadcast sequence numbers for a reliable MAC-layer broadcast delivery, for both cluster-tree and radio neighbors. Simulations with realistic conditions prove the relevance of this approach. We increase energy savings by reducing idle listening, and improve the MAC-layer broadcast reliability for both radio and cluster-tree delivery.**

***Index Terms*—IEEE 802.15.4; cluster-tree; broadcast; Broadcast-Only-Period; beacon collision avoidance**

I. Introduction & motivations

IEEE 802.15.4-2006 [1] was introduced at first in Personal Area Networks (PANs) and represents now one of the major standards for Wireless Sensor Networks (WSN). Medium access may be either asynchronous (beacon-less) or synchronous (with beacons). Without `beacons`, a node cannot sleep because it may receive a frame at any time. Thus, we focus here on the beacon-enabled mode: a node wakes up and transmits a `beacon` to notify its neighbors (i.e. children) that it is ready to receive frames. By appropriately scheduling the active and sleeping parts, IEEE 802.15.4-2006 limits the number of collisions while authorizing energy savings.

While the standard proposes three topologies (star, peer-to-peer, cluster-tree), only the last one allows the nodes to implement a low duty-cycle in multihop topologies. Indeed, the peer-to-peer mode works only with the beacon-less mode, and the star topology does not implement multihop transmissions.

The cluster-tree is constructed distributively with the IEEE 802.15.4 association procedure. At the beginning, only the PAN coordinator (gateway to e.g. a wired network) accepts new associations. Then, iteratively, each newly associated *coordinator* accepts neighbors to associate with the cluster-tree.

A `beacon` contains control information such as the superframe specification or descriptor for the dedicated timeslots. If the `beacons` are lost, this can lead to desynchronization among the nodes, e.g. a node will send the data packets while the coordinator is sleeping. Moreover, some nodes will not be able to associate at all to a coordinator and hence, to the topology.

This work was partially supported by the French National Research Agency (ANR) project IRIS under contract ANR-11-INFR-016.

978-1-4799-3060-9/14/$31.00 © 2014 IEEE

When it comes to how MAC-layer broadcast should be handled, IEEE 802.15.4-2006 makes just two specifications:

1) *any frame that is broadcast shall be sent with its Acknowledgment Request subfield set to zero* [1] else, all the acknowledgements would logically collide;
2) a broadcast message *shall be transmitted immediately following the beacon with the CSMA-CA algorithm* [1].

However, the standard does not explain *what a node should do* when it must broadcast a packet to all its neighbors. Since a child may sleep immediately after having received the `beacon`, the coordinator cannot send a broadcast packet during the Contention Access Period (CAP): some nodes will be deaf and will not receive the packets.

All the children must consequently stay awake during the whole active part of the superframe of its parent. However, we face the following problems:

- a MAC-layer broadcast packet may not be received because radio links are practically unreliable. Since some protocols (e.g. RPL [2]) depend on quite-reliable broadcast, this would cause convergence problems;
- since IEEE 802.15.4 exploits a tree, a network-wide broadcast (flooding) requires that each node forwards a broadcast packet. A single transmission failure may severely affect the global reliability, and amplify the well-known unreliability problem in radio networks [3];
- a node may turn-off its radio to save energy (e.g. during its backoff or when it is inactive). However, a coordinator may transmit a broadcast packet at any time.
- no discovery method is proposed.

To the best of our knowledge the MAC-layer broadcast problem has not yet been studied in IEEE 802.15.4-like networks. The focus was uniquely given to network-wide broadcast (i.e., flooding). The mechanisms we propose here may be adopted to any MAC protocol which exploit periodical beacons to maintain radio links.

The contribution of this paper is threefold:

1) we introduce a Contention Broadcast Only Period (CBOP) so that several coordinators may cohabit in the

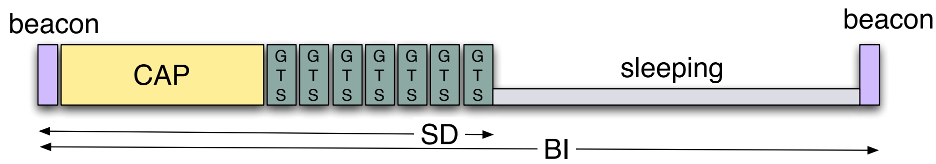

Fig. 1. Superframe structure of IEEE 802.15.4

same superframe while limiting bandwidth wastage. In particular, we limit the impact of a variable `beacon` length, and of a variable number of coordinators per superframe;

2) we implement reliable MAC-layer broadcast transmissions by introducing a broadcast sequence number;
3) we propose a method to implement a discovery MAC-layer broadcast that transmits a broadcast packet to non cluster-tree neighbors (e.g. non discovered nodes).

II. RELATED WORK

A. IEEE 802.15.4

The IEEE has proposed a standard to govern the medium access in this type of networks [1]. The protocol uses a PAN coordinator, inter-connecting the WSN to e.g. the Internet.

The protocol was designed to work with one of the following topologies (Figure 2):

- star: the PAN coordinator is in the radio range of all other nodes (i.e., each node forms a *branch of the star*). Single hop transmissions are in this case sufficient;
- mesh: a node may communicate with any neighbor, the structure being decentralized. A routing protocol may enable multihop communications, using P2P transmissions at the MAC layer;
- cluster-tree: a tree is constructed, rooted at the PAN coordinator. All the non leaf-nodes are designated as *coordinators* since they may forward the traffic to or from the root.

In these topologies, IEEE 802.15.4 may work either in *non-beacon* or in *beacon-enabled* mode. In the former mode, a node just uses a classical CSMA-CA procedure to transmit its packets. In star topologies, a node may sleep since its PAN coordinator buffers its packets: a node has just to periodically ask its PAN coordinator to send the buffered packets. However, in mesh and cluster-tree, the coordinators have to stay awake, limiting energy savings.

In *beacon-enabled* mode, IEEE 802.15.4 introduces the concept of superframes (Figure 1). Each coordinator sends periodically – every `Beacon Interval` (BI) – a `beacon`, piggybacking the control information. Then, transmissions from its children take place using a slotted CSMA-CA solution during the first part of the superframe (CAP) and with dedicated timeslots (GTS) in the second part. A GTS (Guaranteed Time Slot) has to be reserved a priori by a child with a request transmitted during the CAP (Contention Access Period). The whole active part of the superframe lasts for a `Superframe Duration` (SD). When a node has finished participating to the superframe, it may sleep until the next `beacon` reception/transmission.

The `Superframe Duration` (resp. `Beacon Interval`) are defined through the Beacon Order (respectively Superframe Order) values, according to the following relation:

$$SD = aBaseSuperFrameDuration * 2^{SO} \quad (1)$$

$$BI = aBaseSuperFrameDuration * 2^{BO} \quad (2)$$

By adjusting the BO and SO values, we can obtain a tradeoff between network capacity and energy savings. For instance, a duty cycle of 1% can be obtained if $BO - SO = 7$.

A node participates in two superframes: as a child for the superframe of its parent (designated as *outgoing*), and as a coordinator for its own superframe (designated as *incoming*). The standard specifies that both superframes are interspaced by `StartTime`.

However, the active periods must be carefully scheduled to avoid collisions among both `beacons` and `data` packets. In particular, if `StartTime` is a constant, the `beacons` of siblings will collide, since they are sent without a CSMA/CA mechanism. This will significantly decrease the performance of the network.

B. Enhancements

There exists two main approches to reduce the number of collisions. In the *Beacon Only Period* (BOP), nodes rely on a TDMA approach to send their `beacons`: at the beginning of each superframe a few slots are dedicated to `beacons` [4]. However, data packets can still collide during the Contention Access Period (CAP) [5]. Wong proposes to send the `beacon` during a computed slot in the CAP [6]. Since data packets and `beacons` are transmitted in the same frame, he proposes to delay the transmission of the data packets if their slot is the same. However, this can lead to collisions between `beacons` and data packets.

A second solution relates to a variable $StartTime$: two nodes who have the same parent should not use the same $StartTime$ so that their superframes will not overlap. When superframes overlap, collisions between the data packets increase, since they send data packets in the same time frame (i.e., they have the same Contention Access Period). Finding the adequate $StartTime$ for all of them is equivalent to

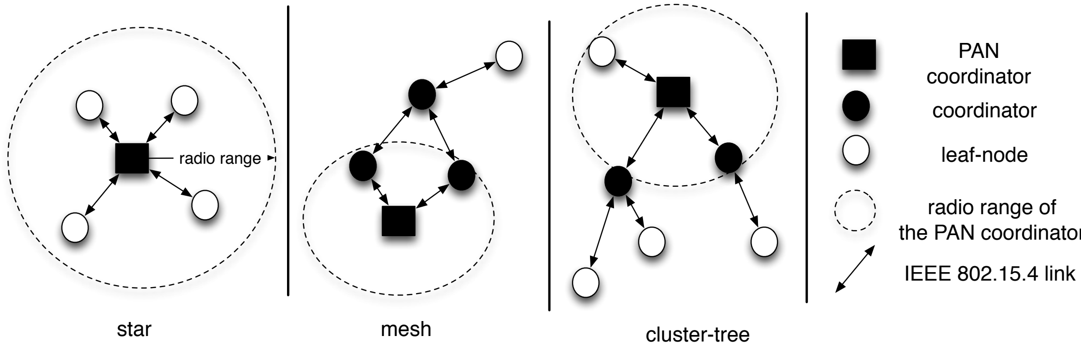

Fig. 2. The different topologies proposed in IEEE 802.15.4

scheduling the superframes with a TDMA approach. Several distributed solutions exist (e.g. [7]). However, bandwidth is wasted since the coordinators who do not have any children will waste a *slot* for their superframe. Thus, BOP and superframe scheduling may be implemented together to reduce both the collisions and the waste of bandwidth [8].

Otal *et al.* proposed to extend IEEE 802.15.4 by separating reservations and data transmissions to limit the number of collisions for star-based scenarios [9]. Our proposed broadcast solution (the use of beacon sequence number) may be adopted to this protocol to optimize the reliability.

C. Broadcast

As highlighted previously, IEEE 802.15.4-2006 does not clearly specify how to cope with MAC-layer broadcast. Most of the existing proposals focus rather on the network-wide broadcast (flooding). For example, Ding *et al.* limit the transmission redundancy in Zigbee [10] by constructing a tree. However, such solution only works with the non-beacon mode of IEEE 802.15.4 and cannot be applied to low duty-cycle protocols.

In the beacon-enabled mode, the broadcast mechanism must cope both with unreliable links and with a duty-cycle MAC. Consequently, most papers implement MAC-layer broadcast by duplicating the packets: a coordinator duplicates the broadcast packet into several unicast packets for each of its children and parent. Guo *et al.* reduce the network-wide broadcast delay by forwarding packets along non-optimal links of the flooding tree when the delay gain is appreciable [11]. However, this scheme relies on unicast transmissions and focuses only on the flooding case.

Wang *et al.* proposed a forwarding selection algorithm to reduce the flooding delay in low duty-cycle networks [12]. The algorithm uses a mix of MAC-layer unicast and broadcast to transmit the packets to the selected forwarders. The MAC-layer broadcast mechanism that we propose here would help implement this flooding optimization in IEEE 802.15.4.

III. Beacon Collision Avoidance (BCA)

Coordinators may share the same superframe either when they do not interfere with each other or when at most one of them has children. In this way, we can avoid collisions during the Contention Access Period (CAP).

A. Problem Statement

Coordinators sharing the same superframe must choose a different BOP slot. A coordinator chooses its BOP slot according to the uniform distribution. Let n_{slots} be the number of BOP slots, and n be the number of coordinators. Let $P[coll]$ represent the probability that at least one `beacon` collision happens. So we firstly compute the probability when no `beacon` collides. We consider that the coordinators are ranked (i.e., by their id). Let's assume the first coordinator has chosen a given random timeslot. No `beacon` collides if each k^{th} coordinator ($k \geq 2$) chooses a slot different from all other first $(k-1)$ coordinators. In other words, it has the choice among $n_{slots} - (k-1)$ slots. Consequently:

$$P[coll] = 1 - \prod_{k=2}^{n} \frac{n_{slots} - (k-1)}{n_{slots}} = 1 - \prod_{k=0}^{n-2} \frac{n_{slots} - (k+1)}{n_{slots}} \quad (3)$$

Thus, n_{slots} has to be chosen large enough to make this probability small.

Figure 3 illustrates this limit of the BOP method: the collision probability becomes quickly large when several coordinators compete for a BOP slot. Practically, we must maintain quite a large number of BOP slots, although a BOP slot consumes bandwidth: in a BOP slot must fit a whole `beacon`. A `beacon` may be long since it contains a list of pending frames (short and long addresses). Typically, a BOP slot duration is around $4ms$. If the SO value is small, the Beacon Only Period consumes most of the active part of the superframe. If we neglect the clock drifts, 4 BOP slots with $SO = 1$ last 36% of the active part of the superframe: there is not much space for data transmissions.

Since the duty-cycle is equal to $2^{-(BO-SO)}$, we should maintain a small SO to reduce the end-to-end delay while keeping a small duty-cycle.

We would like to draw attention to the fact that a Beacon Only Period is very important to make the IEEE 802.15.4 network scalable. A conflict-free scheduling for superframes may even be impossible to obtain in large density cases or

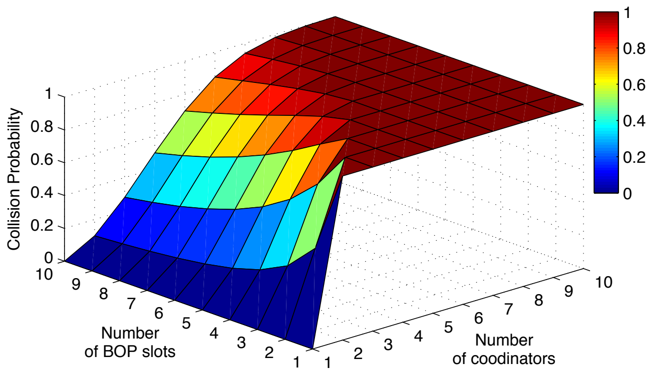

Fig. 3. Collision Probability during the Beacon Only Period

when the number of superframe slots is limited. However, reserving a whole superframe slot for one coordinator without children is clearly sub-optimal.

B. Our Solution: Contention Broadcast Only Period (CBOP)

We propose to replace the TDMA solution for the Beacon Only Period with a deterministic contention, which uses different IBS values (Inter-Beacon Space). While the first one fixes a priori the number of slots with a predefined length, we would rather control the Inter-Beacon-Space than handle several coordinators in a single superframe.

We adopt here an approach inspired from [13] where each node chooses a static mini-slot to transmit its frames. While the original approach focused more on single hop topologies, we propose to adopt this approach for multihop IEEE 802.15.4 networks, avoiding collisions among `beacons`.

A coordinator chooses an Inter-Beacon Space (IBS) value, constant for all its `beacons`:

- it chooses randomly one IBS value in the range $[0..b_{max}-1]$,where b_{max} is the maximum backoff period;
- when the superframe begins, a coordinator has to wait for `IBS_value` $*$ `aUnitBackoffPeriod` idle time before transmitting its `beacon`.

Two `beacons` collide only if both coordinators have chosen the same `IBS_value`. When this occurs, CBOP adopts the same algorithm to assign collision-free values as the BOP solution.

We can notice that the BOP slot duration is much longer than the IBS value ($4ms >> 0.3ms$). Thus, CBOP will save on average more bandwidth than the BOP strategy.

The Contention Access Period (CAP) starts b_{max} $*$ `aUnitBackoffPeriod` after the transmission of the last `beacon`. When a child senses the medium idle during the maximum `IBS_value`, it can safely consider that the `beacon` period is over.

C. Synchronization Requirements

Clock drifts have a significant impact on the performance of the BOP approach. In the classical version, a BOP slot must contain the `beacon` and a `guard_time`. The guard-time is obtained via the maximum clock-drift bound and the inter-beacon period [14].

In CBOP, we propose the following approach:

1) when a coordinator wakes-up at the beginning of its superframe, it must wait `IBS_value` $*$ `aUnitBackoffPeriod` $+$ `guard_time` before transmitting its `beacon`;
2) after the reception of a `beacon` from another coordinator, a node has to defer its `beacon` transmission, but waiting only for `IBS_value` $*$ `aUnitBackoffPeriod`.

For the synchronization, in the worst case CBOP wastes `guard_time` $* b_{max}$ when a coordinator is alone and has chosen the maximum BOP value. Since the classical BOP version wastes exactly `guard_time` $* n_{slots}$ in any case, we reduce on average the overhead due to synchronization.

D. Discussion on the Global Bandwidth Wastage

When we have the maximum number of coordinators ($= n_{slots}$) per superframe, CBOP will face the worst case. Lets consider that a `beacon` transmission lasts at most 4ms, and on average 1ms:

- BOP uses $n_{slots} \times (4ms + guard\_time)$ (each BOP slot must have a fixed duration to contain the longest `beacon`);
- CBOP uses:
  1) the guard time to deal with the clock drift, but only once, for the coordinator with $IBS\_value = 0$;
  2) $n_{slots} * 1ms$ for the transmission (one beacon per coordinator);
  3) $\left(\sum_{k=0}^{n_{slots}-1} k\right)$ `aUnitBackoffPeriod` for the backoff of each coordinator.

  Finally, the time dedicated to `beacons` is $guard\_time + 1ms \times n_{slots} + \frac{(n_{slots}-1)(n_{slots}-2)}{2} *$ `aUnitBackoffPeriod`.

Since the guard-time is actually longer than the `aUnitBackoffPeriod`, and the number of BOP slots is actually limited, CBOP performs better even in the worst case.

Besides, the CBOP method is more flexible since it can deal efficiently with `beacons` with a variable size (the number of pending addresses including in the `beacons` has a signifiant impact on the `beacon` size).

Let's consider an usual case where two coordinators share the same superframe (Figure 4). They both have 2 pending short destinations to include in their `beacons`. With 4 BOP slots, the BOP strategy must reserve at least $4*4 = 16ms$ for the BOP duration: a `beacon` with the maximum packet length must fit in each BOP slot. With CBOP, we must have two `beacons` ($\approx 2ms$ on average) and 3 IBS values ($< 2ms$). Clearly, more bandwidth is wasted for `beacons` in the BOP strategy.

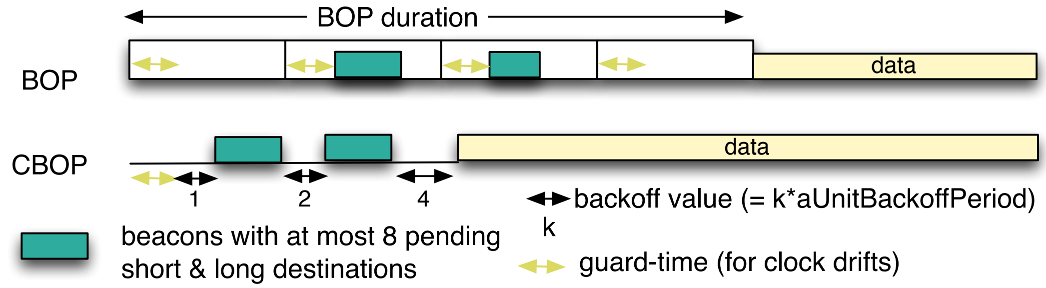

Fig. 4. Overhead for BOP and CBOP

IV. Reliable and Energy-Efficient Broadcast

As highlighted previously, IEEE 802.15.4 does not specify exactly how to deliver broadcast packets to all the radio neighbors. Bachir *et al.* make a distinction between neighboring broadcast (the packet has to be delivered to all the neighbors) and discovering broadcast (the packet aims at discovering new neighbors) [14].

In multihop networks, we must implement both types of MAC-layer broadcast. Indeed, we may require discovering broadcast (e.g. for the cluster-tree reconfiguration) and neighboring broadcast (e.g. for control traffic generated by the routing protocol).

A. Duplicated Broadcast

The easiest way to implement a reliable MAC-layer broadcast consists of two parts: duplicate the packet and send one copy per child in unicast [11]. This approach presents two limits:

- duplicates consume energy: a broadcast transmission is sometimes sufficient to cover several neighbors;
- this method does not work for *discovery*: a coordinator cannot enqueue unicast packets for unknown destinations.

We propose to use our Contention Broadcast Only Period to efficiently disseminate MAC-layer broadcast packets.

B. Broadcast Sequence Number

We will designate the nodes which track the `beacons` of a neighboring coordinator as *followers*. A *follower* may track broadcast packets from a coordinator (e.g. `hellos`) while not being associated with this coordinator. This means that non cluester-tree neighbors also can track and receive the broadcasted packets. Such *follower* MUST listen to the `beacons` of a neighboring coordinator at most every `macTransactionPersistenceTime`.

After having transmitted its `beacon`, the coordinator sends also the broadcast packets enqueued since its last `beacon`. This transmission is safe since another coordinator must sense an idle medium before transmitting its own `beacons`. Besides, we are not blocked by the fixed duration of a BOP slot anymore, which forbids to send a variable number of `beacons` and broadcast packets.

We can notice that a broadcast packet is transmitted only once, following the `beacon` after the packet has been generated/received. It is NOT transmitted after each `beacon`.

Since a neighbor may miss the broadcasted packet (it is sleeping or the packet was corrupted because of a lossy link), we must also implement a mechanism to guarantee the reliability. Consequently, a coordinator maintains also a broadcast sequence number (BSN) in the `beacon`: each time the coordinator has to send a broadcast to its neighbors, it increments the BSN value and enqueues the corresponding broadcast packet.

Broadcast reliability is achieved in the following way:

1) a coordinator piggybacks in its `beacons` its current BSN value;
2) the node compares the BSN included in the `beacon` and the BSN saved in its neighborhood table. If values differ, it generates a `Broadcast-Request` with the last received BSN (i.e., the *requested BSN*). It sends the packet according to the slotted CSMA-CA algorithm;
3) the coordinator acknowledges the `Broadcast-Request`. Then, it sends back-to-back the enqueued broadcast packets with a sequence number superior or equal to the requested BSN;
4) to exploit the broadcast nature of radio transmissions, a follower cannot switch its radio off after sending a `Broadcast-Request`. As soon as a broadcast packet is received in response, a follower updates accordingly the BSN associated with the source.

Although the broadcast transmissions are not acknowledged, we guarantee the reliability: the BSN value for one follower is only incremented when it receives the corresponding packet. Thus, `Broadcast-Request` will keep on being generated until the packet is correctly received. Besides, if the packet was dropped meantime by the transmitter, an empty data packet will be replied.

C. Fairness

If several coordinators share the same active part, unfairness may appear. Indeed, the coordinator with the smallest IBS may capture the medium for all its broadcast transmissions.

To avoid this scenario, a coordinator computes a fair use of the bandwidth: `Superframe Duration` divided by the maximum number of coordinators sharing an active part (b_{max}). A coordinator cannot transmit broadcast packets for a duration longer than $\frac{SD}{b_{max}}$.

We adopt here a pessimistic approach, considering the maximum number of contending coordinators. We may implement an adaptive approach, where each coordinator counts the number of contending coordinators in its active part. This parameter would be updated at the end of each active part.

D. IEEE 802.15.4e

Recently, an amendment to the standard was proposed: IEEE 802.15.4e. The TSCH mode presented in this amendment permits to implement a fast channel hopping approach. Here, timeslots are allocated to avoid interference or collisions. A *dedicated link* is assigned to a single radio link while a *shared link* may be used by several receivers (without acknowledgment) and/or several transmitters. CSMA-CA or ALOHA is required to solve conflicts between interfering transmitters in a shared link.

Broadcast in these conditions faces the same problems. First, the unreliability caused by the fact that some of the receivers may not receive the broadcast packets. Second, the inefficiency caused by individually acknowledging each broadcast packet by each receiver.

Thus, our broadcast algorithm may be used in the same way for shared links with several receivers:

- a transmitter sends a `beacon` or any packet piggybacking its current BSN;
- all its neighbors may wake-up to receive this BSN value;
- a neighbor may ask during a different timeslot the missing broadcast packets to the source;
- the source finally delivers the required broadcast packets in its next timeslot dedicated to broadcast.

The method presented here is sufficiently generic to cope with various situations.

V. Performance Evaluation

We have used WSNet, an event-driven simulator for large scale wireless sensor networks (http://wsnet.gforge.inria.fr) to implement the beacon-enabled mode of IEEE 802.15.4. The simulator has been thoroughly evaluated [15]. Each coordinator selects greedily and distributively a superframe slot to limit collisions [16]. The cluster-tree is constructed distributively, coordinators blacklisting `beacons` with a too small RSSI to avoid choosing bad links in the cluster-tree.

We have considered random circular topologies, where the PAN coordinator is located at the center of the simulated area and the other sensors are placed randomly on a disk (on average, a node has 9 neighbors). We consider only Full-Function-Devices (FFD) i.e., any node joining the cluster-tree acts as coordinator. By default, the network comprises 50 nodes. We ran 10 simulations for each set of parameters and inserted the 95% confidence interval in the graphs.

At the PHY layer, we used the path-loss shadowing model, calibrated with the scenario FB6 (indoor real deployment) presented in [17] (shadowing, path loss= 1.97, standard deviation= 2.0, $Pr(2m) = -61.4dBm$). We used $BO = 8$, $SO = 1$ (duty-cycle $\simeq 1\%$) and 4 BOP slots (n_{slots}).

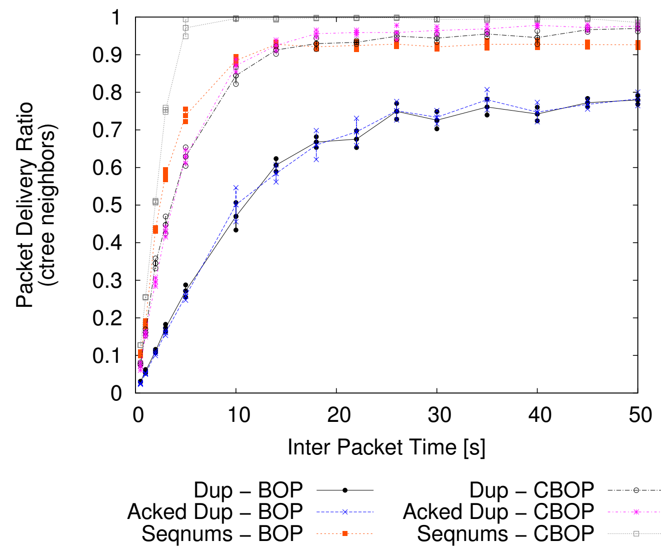

Fig. 5. PDR with an increasing traffic for the different algorithms

We compared the following solutions:

- `Dup`: a broadcast packet is duplicated into several unicast packets and sent to the parents and children;
- `Acked Dup`: the broadcast packets are duplicated and each unicast copy must be acknowledged by the destination;
- `Seqnums`: we implemented the sequence number piggybacked on `beacons` and the `Broadcast-Requests`.

We also compared the original Beacon-Only Period Solution (BOP) — TDMA solution in which one slot is dedicated to each `beacon`— and our Contention Broadcast Only Period mechanism (CBOP).

We first measured in Figure 5 the packet delivery ratio for packets broadcasted to the cluster-tree neighbors (parents and children). We can observe that our Contention Broadcast Only Period efficiently disseminates broadcasts while minimizing the bandwidth dedicated to the CBOP (waste of bandwidth due to IBS is limited). On the contrary, the BOP solution creates many collisions among `beacons`, explaining the lower packet delivery ratio. We can also notice that our broadcast solution based on sequence numbers is more efficient compared to duplicating broadcast packets. Moreover, our CBOP mechanism is much more robust to larger traffic. With CBOP, we may operate at a lower duty-cycle, increasing energy savings.

We also measured the overhead (Figure 6). The seqnum solution efficiently reduces control traffic: broadcast transmissions during the CBOP are often sufficient to *cover* all the neighbors. Some additional `Broadcast-Requests` are seldom required for unreliable links.

We evaluated the impact of the CBOP algorithm on the energy consumption. In particular, Figure 7 reports the average sleeping time for each solution. With BOP, a node has to wait

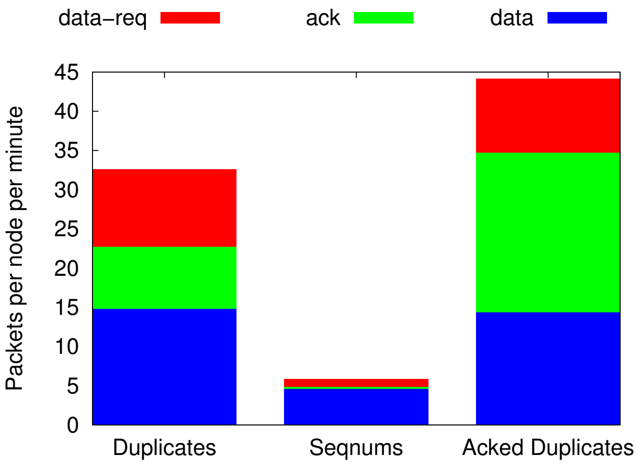

Fig. 6. Overhead — 1 broadcast packet every 20s, CBOP algorithm

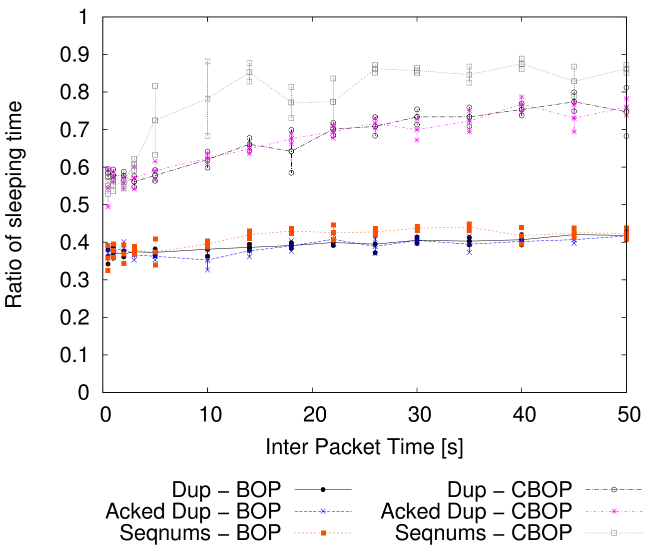

Fig. 7. Energy — duty-cycle ratio

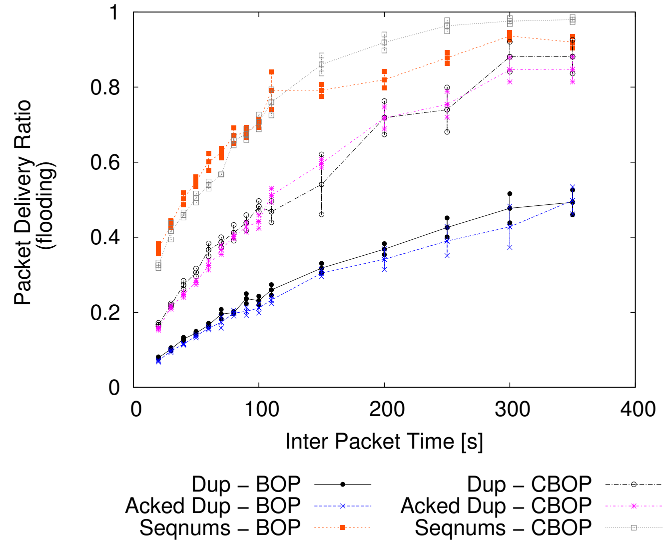

Fig. 8. Coverage for a flooding

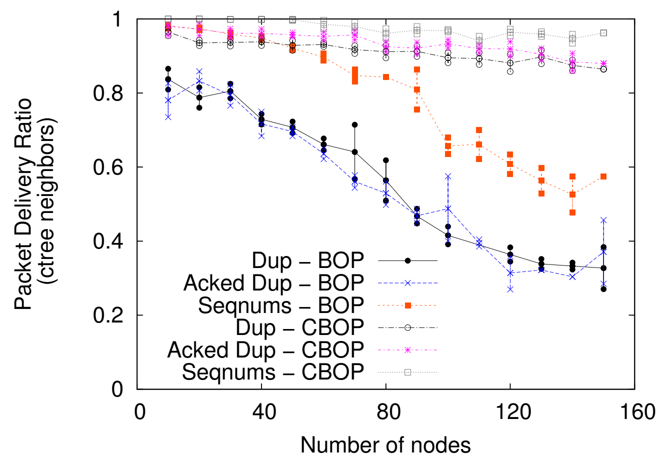

Fig. 9. Scalability – 1 packet every 7 seconds

for the whole BOP duration for all the superframe it follows. Thus, a node sleeps just around 40% of the time. On the contrary, CBOP limits idle listening by reducing the period dedicated to broadcast and `beacons`, making a node sleep longer (around 80% of the time).

Next, we evaluated the impact of the broadcast algorithm on the flooding reliability: each node which receives a broadcast packet has to forward it (Figure 8). By exploiting efficiently the redundancy of the flooding structure, CBOP with sequence numbers achieves the best reliability. For small inter packet times, the load is too important to be forwarded efficiently, reducing the packet delivery ratio. When packets are forwarded only to cluster-tree neighbors, the reliability decreases: this structure is too weak to guarantee a correct delivery. In the same way, creating duplicated packets increases the number of collisions, impacting negatively the reliability.

Finally, we evaluated the scalability of these solutions (Figure 9). BOP is not scalable: more nodes mean more collisions among `beacons` and among data packets. Thus, the broadcast reliability quickly decreases when we increase the number of nodes. While CBOP achieves an almost perfect delivery for small networks, duplicating packets often creates more collisions: several children may send an IEEE 802.15.4 `data-request` command simultaneously to retrieve packets buffered at a coordinator. This well-know phenomenon in IEEE 802.15.4 impacts the packet delivery ratio when duplicating broadcast packets.

VI. Conclusion & Perspectives

We proposed here to modify the superframe structure by introducing a Contention Broadcast Only Period: each competing coordinator chooses distributively a fixed Inter-Beacon-Space to send its `beacons` and broadcast packets in its superframe. By removing the BOP slot, we reduce the bandwidth wasted by `beacons`: we can safely reduce the duty-cycle while maintaining the same capacity. Besides, we also proposed to use broadcast sequence numbers to guarantee a certain reliability in lossy networks. Simulations with a realistic shadowing PHY model prove our solution efficiently disseminates broadcast packets while limiting the overhead.

In the future, we plan to explore the impact of real-testbed deployments on our broadcast strategy. Furthermore, we aim at investigating what would be the optimal cluster-tree to implement efficiently both unicast and broadcast transmissions. We also must study how self-pruning techniques may be incorporated to this mechanism for reliable flooding.

References

[1] IEEE 802.15 WPAN task group 4b (tg4b). http://www.ieee802.org/15/pub/TG4b.html, September 2006.

[2] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, JP. Vasseur, and R. Alexander. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550, Internet Engineering Task Force, 2012.

[3] F. Kuhn, N. Lynch, C. Newport, R. Oshman, and A. Richa. Broadcasting in unreliable radio networks. In *PODC*, pages 336–345, Zurich, Switzerland, 2010. ACM.

[4] A. Koubaa, A. Cunha, and M. Alves. A time division beacon scheduling mechanism for IEEE 802.15.4/zigbee cluster-tree wireless sensor networks. In *ECRTS*, pages 125–135, Washington, DC, USA, 2007. IEEE Computer Society.

[5] A. Koubâa, B. Nefzi, M. Alves, and Song Y.-Q. Improving the IEEE 802.15.4 slotted CSMA/CA MAC for time-critical events in wireless sensor networks. Technical report, Polytechnic Institute of Porto, July 2006.

[6] C.-M. Wong. A novel beacon frame scheduling algorithm based on cluster-tree ieee 802.15.4 wireless sensor networks. In *Communication Systems (ICCS), 2012 IEEE International Conference on*, pages 285–289, November 2012.

[7] W.-Y. Lee, W.-I. Hwang, Y.-A. Jeon, and S.-S. Choi. Distributed fast beacon scheduling for mesh networks. In *MASS*, Valencia, Spain, October 2011. IEEE.

[8] B. Pavkovic, W.-J. Hwang, and F. Theoleyre. Cluster-directed acyclic graph formation for IEEE 80215.4 in multihop topologies. In *NTMS*, Istanbul, Turkey, May 2012. IFIP/IEEE.

[9] B. Otal, L. Alonso, and C. Verikoukis. Energy-efficiency analysis of a distributed queuing medium access control protocol for biomedical wireless sensor networks in saturation conditions. *Sensors*, 11:1277–1296, 2011.

[10] G. Ding, Z. Sahinoglu, P. Orlik, J. Zhang, and B. Bhargava. Tree-based data broadcast in IEEE 802.15.4 and ZigBee networks. *IEEE Transactions on Mobile Computing*, 5(11):1561 –1574, November 2006.

[11] S. Guo, Y. Gu, B. Jiang, and T. He. Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links. In *MobiCom*, pages 133–144. ACM, 2009.

[12] F. Wang and J. Liu. On reliable broadcast in low duty-cycle wireless sensor networks. *IEEE Transactions on Mobile Computing*, 11(5):767 –779, May 2012.

[13] J. Lee and J.C. Walrand. Design and analysis of an asynchronous zero collision mac protocol. arXiv:0806.3542, 2008.

[14] A. Bachir, F. Theoleyre, A. Duda, and K. K. Leung. Energy-efficient broadcasts in wireless sensor networks with multiple virtual channels. In *WCNC*, Sydney, Australia, 2010. IEEE.

[15] E. Ben Hamida, G. Chelius, and J-M. Gorce. On the complexity of an accurate and precise performance evaluation of wireless networks using simulations. In *MSWiM*, Vancouver, Canada, 2008. ACM.

[16] B. Pavkovic, F. Theoleyre, and A. Duda. IEEE 802.15.4 and RPL cross-optimization for reliable opportunistic routing in wsn. In *MSWiM*, Miami, USA, October 2011. ACM.

[17] Y. Chen and A. Terzis. On the implications of the log-normal path loss model: an efficient method to deploy and move sensor motes. In *SenSys*, Seattle, Washington, 2011. ACM.