HAL
open science

Bandwidth and Energy Consumption Tradeoff for IEEE 802.15.4 in Multihop Topologies

Mohamed Nazim Abdeddaim, Benoit Darties, Fabrice Theoleyre

To cite this version:

Mohamed Nazim Abdeddaim, Benoit Darties, Fabrice Theoleyre. Bandwidth and Energy Consumption Tradeoff for IEEE 802.15.4 in Multihop Topologies. Wireless Sensor Networks: Technology & Applications, Intech, pp.00-00, 2012, 10.5772/48295 . hal-00923327

HAL Id: hal-00923327

https://hal.science/hal-00923327v1

Submitted on 2 Jan 2014

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.

Chapter 5

Bandwidth and Energy Consumption Tradeoff for IEEE 802.15.4 in Multihop Topologies

Abdeddaim Nazim, Darties Benoît and Theoleyre Fabrice

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/48295

1. Introduction

With the development of electronic technology and embedded systems, wireless sensor networks (WSN) are gaining growing interest over the past few years. A WSN is a collection of autonomous devices organized into a cooperative radio network . Each node is connected to one or several sensors, and may monitor various physical or environmental conditions, such as temperature, vibration, motion, radiation activity. WSN can vary from a few to thousands nodes equipped with transceivers for low range wireless communication.

WSN benefit from low-cost technology, flexible deployment and self-management features, making them a serious alternative to traditional networks in many domains. WSN application fields are numerous and highly anticipated in the near future: e.g. structure health monitoring (SHM), habitat monitoring, localization and tracking, environmental sensing, industry, healthcare monitoring. Over the past 10 years, numerous projects have lead to real-life deployments : one emblematic example for SHM is operated on the Golden Gate bridge [1], which is monitored thanks to a 46 hops-network of 64 nodes equipped with accelerometers and temperature sensors. Collected data help, among other things, to determine the response of the structure to ambient conditions, to measure the structural accelerations from wind load or strong shaking from a potential earthquake. WSN may also operate in various productivity domains including agriculture. Camalie Vineyards deployed a smart infrastructure [2], leading to significant productivity gain. WSN-based approaches are also particularly relevant in extreme environmental conditions, i.e. for monitoring volcanic activity. Others real-life experimentations may include habitat monitoring (e.g. Great Duck Island project), wildlife tracking (e.g. Zebranet Project), healthcare monitoring (e.g. Mobihealth), etc.

When wireless sensor networks are designed on embedded systems for long term operations, energy conservation is an essential requirement. To achieve low-cost, low-speed ubiquitous communication between devices, IEEE 802.15 working group has proposed the standard IEEE 802.15.4: this standard specifies the physical layer and media access control for

INTECH
open science | open minds

©2012 Theoleyre et al., licensee InTech. This is an open access chapter distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

low-rate energy-efficient transmissions, operating energy conservation though the use of sleep/wakeup scheduling protocols, thus minimizing the activity on sensor nodes. These features make IEEE 802.15.4 one of the main standards for low data rate wireless sensor networks. However, energy limitations have a negative impact on network capacities since transmissions are unavailable when nodes are in power-saving mode. In this chapter, we investigate the problem of increasing the capacity of IEEE 802.15.4 single-hop and multihop networks subjects to energy limitations. In the presented analysis, we assume classical scenarios where collected data must be routed to a unique collect point, i.e. a *convergecast* traffic pattern. The wide majority of IEEE 802.15.4 deployments also use a sink to collect data.

The chapter will be organized as follows. In Section 2, we analyze IEEE 802.15.4 mechanisms including node organization, MAC mechanisms, energy conservation, topology construction and node association. In Section 3, we detail how we should modify IEEE 802.15.4 to cope efficiently with multihop topologies, scheduling the transmissions. In Section 4, we quantify the impact of the cluster-tree algorithm on the network performances. We expose how the overall throughput can be improved with a novel cluster-tree construction algorithm defined formally as a Mixed Integer Linear Programming formulation. In Section 5, we quantify the impact of each parameter on the performances of IEEE 802.15.4. In particular, we present a self-configuration algorithm to dynamically adjust the Backoff Exponent so that the protocol always operates in optimal conditions. We finally conclude this chapter by discussing open challenges in this research area.

2. Background on IEEE 802.15.4

An IEEE 802.15.4 network comprises one PAN coordinator and a set of devices. Theses low-power devices are characterized by a limited transmission range and a limited quantity of energy. The IEEE has proposed the IEEE 802.15.4 to govern the medium access in this kind of networks, presented in [3]. The protocol uses a PAN coordinator, inter-connecting the WSN to e.g. the Internet. Although IEEE 802.15.4 was originally designed for single hop networks, the working group has proposed topologies to cope with multihop applications (Figure 1):

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;

peer-to-peer: 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.

2.1. MAC mechanisms

In these topologies, IEEE 802.15.4 may work either in *non-beacon* or in *beacon-enabled* mode. The MAC strategy impacts the duty-cycle and thus the capacity and energy consumption. In IEEE 802.15.4, one set of nodes (the *coordinators*) regulates the transmissions. Any exchange is initiated from the non-coordinator to the coordinator. In particular, the coordinator buffers all the packets destined to others. Then, each non-coordinator MUST periodically ask for the buffered packets to the coordinator by sending a `data-request` — at most every `macTransactionPersistenceTime` since packets are removed from the buffer of

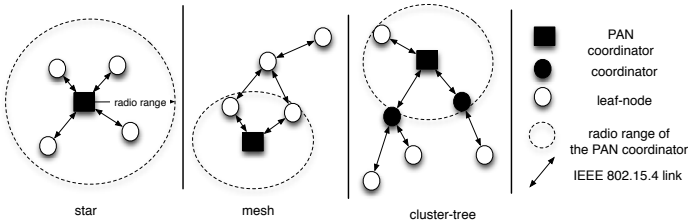

Figure 1. The different topologies proposed in IEEE 802.15.4

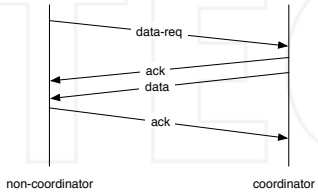

Figure 2. Indirect transmission (non-coordinator → coordinator)

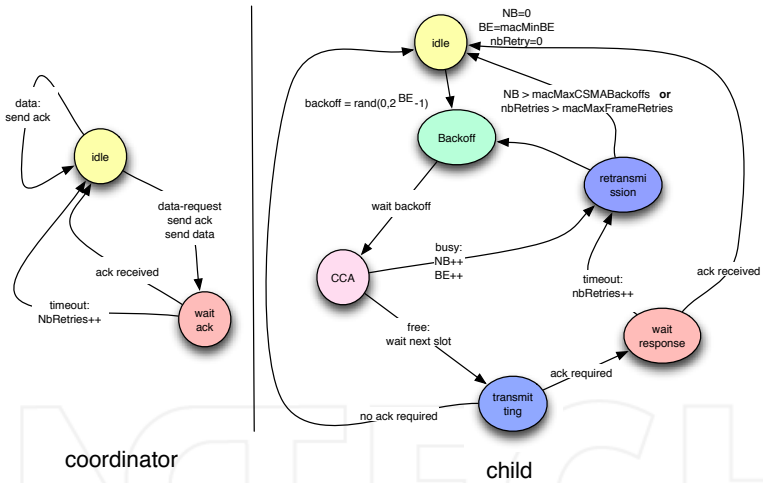

Figure 3. MAC procedure in the non-beacon mode

the coordinator after this timeout. IEEE 802.15.4 implements two *modes* for the medium access. In non-beacon mode, transmissions use a classical CSMA-CA scheme while the beacon-enabled mode introduces the concept of superframes to schedule the transmissions more appropriately and to assign dedicated timeslots.

2.1.1. Non-beacon mode

In non-beacon mode, a node just uses a classical CSMA-CA procedure to transmit its packets (Figure 3)

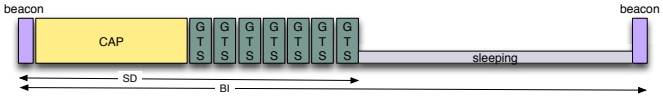

Figure 4. The superframe structure in IEEE 802.15.4

1. The node is in the idle state, and initializes the value of the backoff exponent (BE).
2. It chooses a random backoff comprised between 0 and $2^{BE} - 1$;
3. After this backoff, the node must trigger a Clear Channel Assessment (CCA): if the channel is free, the transmission begins. Else, the node triggers a new backoff after having incremented the Backoff Exponent (BE) and the number of backoff retries (NB). If NB exceeds the threshold `macMax CSMABackoffs`, the node comes back to the idle state;
4. After having transmitted the packet, the node comes back to the idle state if it does not need acknowledgement. Else, it waits for the ack;
5. If the ack is correctly received, the node considers the transmission successful and proceeds to the next packet. Else, it increases the number of retransmissions and tries a new retransmission if is does not exceed the threshold value `macMaxFrameRetries`.

Figure 3 presents a simplified state diagram for the unslotted mode, separated for coordinators and children. As coordinator, the node has just to wait solicitation: the medium is pre-reserved by the other side. As child, it must choose a random backoff, trigger a CCA after the backoff is finished, and then transmit the frame if the medium is idle. The number of backoff retries and retransmissions are upper-bounded respectively by `macMaxCSMABackoffs` and `macMaxFrameRetries`. Besides, a node must ensure the `BE` values does not exceed `macMaxBE`. A child may wait for a response from the coordinator: either a simple `ack` if it transmitted a `data` packet or an `ack` followed by a `data` packet if it transmitted a `data-request`. As coordinator, the only response to wait can be an `ack`. IEEE 802.15.4 also proposes a Battery Life Extension (BLE) option for very energy-constrained devices. With this option enabled, a node will choose a minimal Backoff Exponent value after a successful transmission. Thus, it will gain medium access quicker than other nodes without BLE.

2.1.2. Beacon-enabled mode

In *beacon-enabled* mode, IEEE 802.15.4 introduces the concept of superframes (Figure 4). Each coordinators sends periodically – every `Beacon Interval` (BI) – a `beacon`, piggybacking control information. Just after having received the `beacon`, children may engage a transmission during the Contention Access Period (CAP) using slotted CSMA-CA. Children may in particular map a reservation during the CAP to obtain a dedicated Guaranteed Timeslot (GTS) placed just after the CAP. Finally, the children and the coordinator may sleep until the next beacon. The whole active part of the superframe lasts for a `Superframe Duration` (SD). When a node has finished to participate to the superframe, it may sleep until the next `beacon` reception/transmission. The Superframe Duration (resp. Beacon Interval) are defined through the Beacon Order (resp. Superframe Order) values, according to the following relation:

$$SD = aBaseSuperFrameDuration * 2^{SO} \tag{1}$$

$$BI = aBaseSuperFrameDuration * 2^{BO} \tag{2}$$

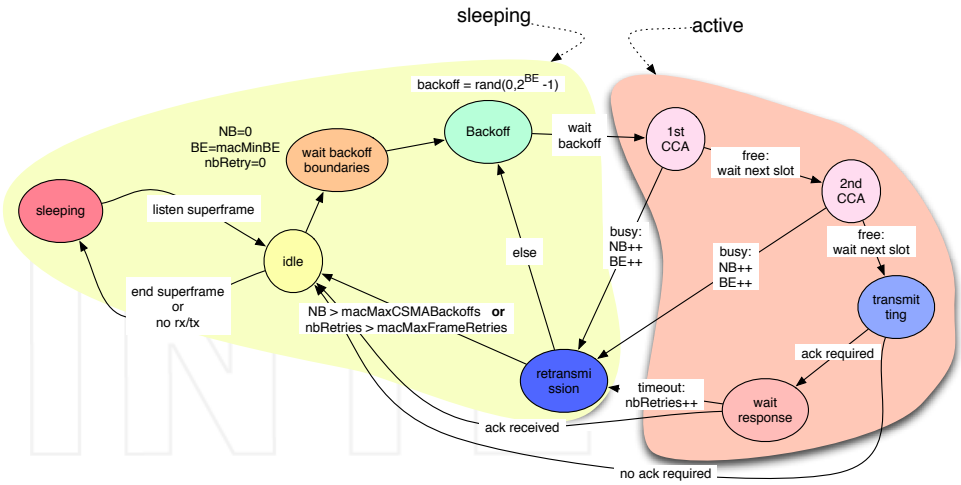

Figure 5. MAC procedure in the beacon-enabled mode for the child

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 $SO = BO - 7$ ($2^{-7} < 1\%$). The beacon-enabled mode is only available in the star and cluster-tree topologies since we must maintain a hierarchy of coordinators to schedule appropriately the `beacons`. Figure 5 illustrates a simplified state diagram of the child mode in slotted mode. The coordinators implement the same behavior as depicted in Figure 3. In particular, the node triggers two CCA before transmitting a packet (the `CW` parameter in the IEEE 802.15.4 standard). Besides, a child may sleep all other the time except during CCA, the transmission and while waiting for a response from the coordinator (a unique `ack` or a pair of `ack` and `data` packets).

2.2. Energy savings

2.2.1. Peer-to-peer topology

If no energy optimization technique is implemented in a peer-to-peer topology, a Full-Function Device (FFD) may receive a packet at any time from a neighbor. To save energy, the MAC layer must implement a synchronization mechanism so that two neighbors can agree on a sleeping schedule. The pair of nodes must be periodical awake to exchange packets. A Low-Power-Listening protocol with preambles may be implemented to be sure the receiver is awake. This technique was proposed to reduce the energy consumption in very low traffic conditions [4]. When a node wakes-up and senses a signal, it must stay awake to receive further a `data` packet. Bachir *et al.* [5] proposed to piggyback information in the preamble so that a node knows it is the destination of the next packet when it receives the preamble, technique known as *preamble sampling*. The nodes may also agree on their wake-up schedule, as proposed in S-MAC [6]. Nodes broadcast periodically `beacons` to publish their schedule and synchronize with their neighbors. This mechanism may be coupled with the preamble to deal with clock drifts and reduce the synchronization requirements, as in [7]. However, all these propositions are outside the scope of the standard and to the best of our knowledge were never evaluated with IEEE 802.15.4.

2.2.2. Star topology

In single hop networks (i.e. star topology), all transmissions are initiated by the *followers* (i.e. nodes which follow the PAN coordinator). Thus, a node may sleep safely, asking periodically for pending packets in the PAN coordinator with a `data-request`. Nodes may reduce their energy consumption: only the PAN coordinator has to stay awake continuously.

2.2.3. Cluster-trees

non-beacon mode In a cluster-tree, we may authorize some nodes to be categorized as Reduced-Function Devices (RFD). These nodes cannot relay packets because of energy constraints. Since they would constitute leaf nodes in the cluster-tree, they may sleep safely. However, all other nodes are Full-Function Devices (FFD) and cannot sleep: energy savings are very limited.

beacon-enabled mode As highlighted previously, all the transmissions are initiated by the children, i.e. the coordinator cannot start a transmission before a solicitation of its child. This feature permits to implement efficient power-saving mechanisms. As *follower* (i.e. a node which participates to a superframe without coordinating it), a node must wake-up to receive at least one `beacon` every `macTransactionPersistenceTime`. If packets are pending, it must retrieve them immediately by transmitting a `data-request`. As soon as a follower has neither pending packet nor packet in its own buffer to transmit to its parent, it may sleep. As coordinator, a node must stay awake during the whole active part of its superframe. For all these reasons, the beacon-enabled mode should be privileged since this constitutes the only way to optimize the energy consumption in multihop topologies.

2.3. Topology construction

Except for the peer-to-peer topology where a node can communicate with any neighbor, the cluster-tree and star topologies require a node to be associated before transmitting packets. An unassociated node must discover a neighboring associated coordinator and send an `association-request`. Then, the node sends a `data-request` to retrieve an `association-reply`. If the newly associated node is a Full-Function-Device (FFD), it may accept now the association requests. However, the standard does not specify what coordinator a node should choose to associate with. The properties of a cluster-tree when a node associates to the first available parent is studied in [8]. The impact of mobility on the topology formation process was also investigated in [9]: it creates some convergence problems. Zigbee [10] proposes to couple the association procedure with an address assignment scheme: addresses being hierarchical, routing is simplified. However, the limited pool of addresses could create the orphan problem, as highlighted in [11]. Thus, the cluster-tree must be well balanced. Only few attention has been given on determining the characteristics of the cluster-tree should own and what algorithms could obtain them.

2.3.1. Active versus passive discovery

To discover an already associated coordinator, we may implement two strategies:

active discovery: a node enters in *active scan* and sends a `beacon-request` on each operational channel. An already associated coordinator MUST reply with a `beacon` if the PAN work in the *non-beacon* mode, otherwise the coordinator ingnore the

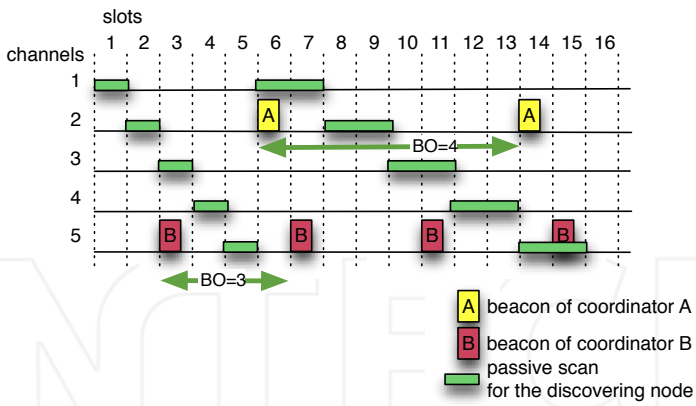

Figure 6. Passive Discovery in Multichannel IEEE 802.15.4

`beacon-request` and continue sending its periodic beacons. Upon reception of a `beacon`, the node can engage the association procedure .

If many nodes begin an active scan simultaneously, we may face to collisions. Besides, this method must be carefully implemented in beacon-enabled mode: a coordinator may sleep during the passive part of the superframe. Thus, one `beacon-request` has to be transmitted every Superframe Duration (SD), each channel being scanned during Beacon Interval (BI).

passive discovery: In beacon-enabled mode, a coordinator MUST transmit every Beacon Interval (BI) its `beacon` at the beginning of the active part of its superframe. Thus, a node may implement a *passive scan*: it has to stay at least BI on a channel to receive any `beacon` from already associated coordinators. If BI and channel are not known a priori, the node may assume the worst case, leading to very long discovery times. This strategy is impossible in non-beacon mode since an associated coordinator does not send any periodical packet.

The discovery process may be adapted for mobile nodes in beacon-enabled mode, as in [12] . They introduced a meeting channel: the manager in charge of mobile nodes send periodically their `beacons` on the meeting channel. Since the number of managers is limited and are geographically distributed, collisions may stay limited. Some focused on the discovery process for multichannel IEEE 802.15.4 (see [13]). The scan is optimized to discover first coordinators with a small Beacon-Interval (larger probability to be discovered first when a channel is scanned during a small duration). Figure 6 illustrates this behavior: the discovering node scans for small increasing durations each channel. Karowski *et al.* [14] extended this approach by reducing the redundancy in the discovery phase (a periodical slot has not to be scanned several times).

2.3.2. Required properties

Since Wireless Sensor Networks comprise a wide range of applications, we cannot focus on a particular topology: we may deploy a grid for surveillance applications, or pseudo-randomly

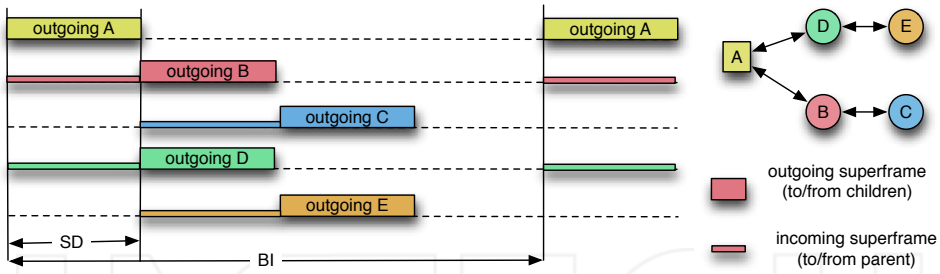

Figure 7. Scheduling of superframes in multihop topology.

for building automation. Moreover, the topology may be dynamic: we must be able to insert or remove nodes during the lifetime of the network. Thus, algorithms and protocols must distributed (decisions should use only local knowledge to decrease the overhead and improve the scalability) and self-configurable (the protocol must assign autonomously addresses and obtain its parameters values). In the same way, we must cope with different traffic patterns:

convergecast (Multipoint to Point) : all the packets are transmitted to one sink;

Point to Multipoint : information is flooded in the network;

Point to Point : direction communications may exist between e.g. a sensor and an actuator. For instance, Zigbee [10] proposes routing shortcut mechanisms to cope with this traffic pattern. On the contrary, RPL [15] uses only the DAG to route packets.

3. Cluster-tree scheduling in multihop topologies

Because we aim at minimizing the energy consumption in multihop topologies, we focus here on the cluster-tree topology in beacon-enabled mode. As highlighted in section 2.2, other modes do not permit to save energy easily.

3.1. Problem statement

Any device has to participate to the superframe of its parent, denoted *incoming superframe*. Additionally, a FFD has also to maintain its own superframe, called *outgoing superframe*. The standard mentions the outgoing and incoming superframes are interspaced by `StartTime`. Obviously, `StartTime` may be superior than Superframe Duration, else both superframes will overlap, creating collisions. Since cutting off its radio has an energy cost, we may choose `StartTime` $= SD$ so that the outgoing superframe follows directly the incoming superframe. Figure 7 illustrates a simple cluster-tree with this approach. Thus, all the coordinators with the same depth in the cluster-tree will start the active part of their superframes simultaneously. They will in particular send their `beacon` at the same time, creating a collision and making IEEE 802.15.4 inefficient. CSMA-CA performs very poorly in multihop topologies because of hidden nodes, unfairness and collisions, as demonstrated in [16]. Thus, we face to the same pathological cases in IEEE 802.15.4: collisions increase quickly with the number of nodes and the density, making the network unsuitable for large-scale applications. We clearly have to solve these problems to make IEEE 802.15.4 efficient in multihop topologies.

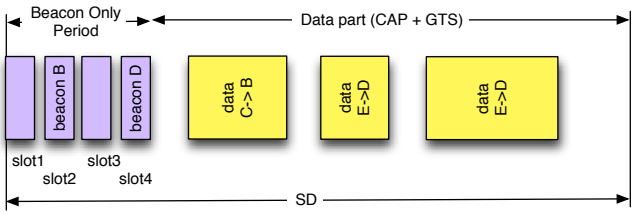

Figure 8. Beacon-Only Period to solve collisions among `beacons`

3.2. Beacon-only period

To reduce the number of collisions, we may reserve a *Beacon Only Period* (BOP) at the beginning of the superframe, as proposed by the IEEE 802.15.4b [17]. The BOP is divided into slots: each coordinator has to choose a BOP slot to transmit its `beacon`. Thus, several coordinators may share the same superframe without interference by choosing different BOP slots for their `beacons`. Let consider Figure 8 illustrating such approach (with the same topology as Figure 7). We only represented the superframe of the nodes B, C and D. We have in this case 4 dedicated timeslots used for the `beacons`: B chooses the second slot for its `beacons` while node D chooses the 4th one. An algorithm must be implemented to detect collisions of `beacons` during a BOP slot. Indeed, a node N may detect a collision but the transmitters may not if e.g. N is placed in the middle of both coordinators. While this technique limits collisions between `beacons`, `data` may keep on colliding. Indeed, the different coordinators share the same Contention Access Period, creating hidden terminals. Unfortunately, this scenario occurs frequently, as demonstrated in [18].

3.3. Superframe scheduling

A second solutions consists in scheduling the superframe so that nodes which participate simultaneously to the superframe of two different coordinators do not interfere with each other, as advocated by IEEE 802.15.4b [17]. In this way, the remove entirely the collisions among nodes which participate to different superframes. The standard advocates all the nodes to use the same Beacon Order (BO) and Superframe Order (SO) values. In this case, we can construct a schedule of superframes using a Time Division Multiple Access (TDMA) approach. Each slot may contain exactly one superframe: a node has consequently to find a slot not used by an interfering coordinator to transmit its superframes. Using different SO and the same BO is equivalent to reserving several consecutive slots for one coordinator. However, this implicates the scheduling problem by inserting new constraints. In particular, the consecutive slots must start at a multiple of SD. This scheduling problem has received strong attention in the past. A clustering approach could simplify the scheduling as in [19], in which the PAN coordinators find an accurate scheduling while controlling the power of each superframe. However, this requires a centralized approach, based on information complicated to obtain (i.e. interferences are not trivial to estimate). This scheduling has been extended for the mesh topologies in [20]. Thus, such solution does not permit to save energy. A greedy localized solution was proposed in [21]: a node picks a free slot in its 2-neighborhood to schedule its superframe. However, it only takes into account interferences among coordinators, and not with children, which occur frequently in a cluster-tree. The TDMA approach is often more efficient, as highlighted in [22]: it reduces obviously the collisions. However, a collision-free scheduling is practically complicated to achieve.

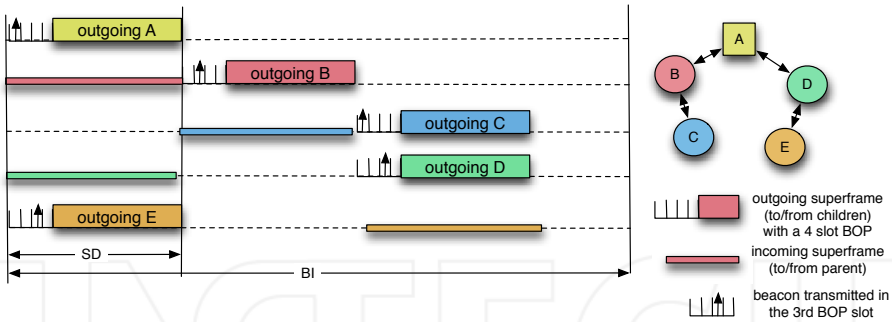

Figure 9. Beacon-Only-Period and superframe scheduling combination

3.4. Combining both techniques

Since we deal with constrained nodes, we aim at minimizing the energy consumption. Thus, we should use shortest paths routing to reduce the number of transmissions. This metric permits to avoid oscillations when creating the DAG while avoiding any loop. We chose to preserve a route length stretch factor of 1: the routes length through the DAG has to be the same as in the original graph. We will adopt the following notation to make a distinction between the different slots:

backoff slot : the units of time used by CSMA-CA algorithm;

GTS slot : the active part of the superframe is divided into slots (which are different from the backoff slots) as described in the standard. The first slots of the active parts are not used, while the last slots correspond to the GTS slots;

BOP slot : the Beacon-Only-Period is divided into slots, containing at most one `beacon`;

superframe slot : a slot containing the whole active part of a superframe, as explained previously.

When the notation is not ambiguous, we will use the term *slot* alone. We integrated both the **Beacon-Only Period** and the **superframe scheduling** techniques as depicted in Figure 9. This organization has the following assets:

1. we avoid wasting bandwidth for coordinators without children;
2. the scheduling permits to maintain several parents, creating redundant paths toward the sink;
3. we reduce energy consumption by maximizing the number of nodes without children.

3.4.1. Neighborhood discovery

To implement a distributed scheduling solution, a coordinator should know the list of slots used by interfering coordinators. We consider two coordinators interfere with each other if their superframe collide. For instance, a child of the first coordinator may interfere with the child of the other one. Since determining the exact set of interfering coordinators is complicated to obtain, we should just estimate it. IEEE 802.15.4 [17] proposed to exchange

this information along the cluster-tree although the radio and cluster-topologies may very different. We consider rather the 2-radio neighborhood constitutes a better approximation, collisions among 3 or 4-neighbors being solved separately as exceptions.

A coordinator has finally to piggyback in its `beacons` its depth in the cluster-tree (to break conflicts), a leaf flag (i.e. has the coordinator at least one child), the current BOP and superframe slots and a list of 1-neighbors with their short address and their BOP and superframe slots.

To have the radio and not the cluster-tree topology, a coordinator always wakes-up at the beginning of the slots used by its 1-neighbors. It will then go sleeping immediately after having received the corresponding `beacon`. To track changes, a coordinator must periodically stay awake to receive all the `beacons` and update its 2-neighborhood table.

3.4.2. BOP scheduling

The Beacon-Only-Period aims just at solving collisions between coordinators sharing the same superframe. Thus, a new coordinator just picks a random BOP slot not used by any 2-neighbor using the same superframe. We use consequently the approach proposed in [22].

3.4.3. Superframe scheduling

After having collected the slots used by the superframes of its 2-neighboring coordinators, a new coordinator *C* chooses a superframe slot as following:

1. *C* sorts the superframe slots according to the number of 2-neighbors using them;
2. *C* removes the slot of its parent: it cannot maintain a superframe simultaneously without a severe impact on performance;
3. *C* picks randomly one of the least loaded superframe slots.

Obviously, if a superframe slot is not used at all in the 2-neighborhood, it will pick it. However, collisions may even appear because of e.g. inconsistent simultaneous decisions. We can make the distinction between the following cases:

- two coordinators are interfering. In particular, at least one coordinator detects a carrier sense when the other transmits a packet. If they use a different BOP slot, one coordinator will detect an interfering `beacon`: it reapplies the precedent rules forbidding the current slot.

  If they use the same BOP slot, they cannot detect a carrier sense. However, children will not be able to finalize the association since they did not receive the `beacon`. Thus, a coordinator without child (no `association-request` was received) just chooses another superframe slot forbidding the current slot. This change is safe since it does not have any follower.
- an unassociated node suffers from the hidden terminal problem, making a pair of coordinators colliding. With a high probability, this node has another neighboring coordinator with which it can associate. If no candidate exists, the unassociated node will *simulate* a `beacon` transmission to create a collision in the interfering coordinators: at least one of them will choose another superframe slot, solving the collision problem.

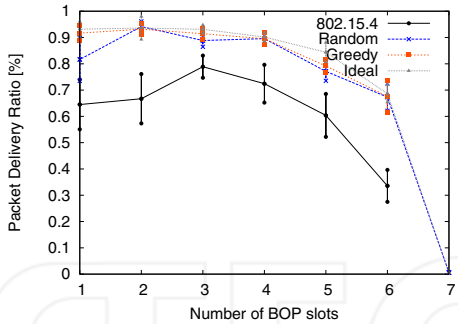

Figure 10. Tuning the Beacon-Only-Period

A coordinator decides to change its slot at the beginning of the active part of its superframe. If the slot has changed, it broadcasts its `beacon` as usually with the new superframe slot and goes sleeping. The possible receivers will be able to detect the source has changed its superframe slot and will update their neighborhood table. They will then *follow* this coordinator (i.e. listen to its new slot to receive its `beacons`).

3.5. Performance evaluation

We used the WSNet simulator [23], with the beacon-enabled IEEE 802.15.4 module [24]. We compared here different scheduling algorithms: original IEEE 802.15.4 (constant `StartTime`), random scheduling (any slot different except the slot of my parent), greedy (the algorithm presented here), ideal (centralized coloring).

We first illustrated the impact of the BOP duration (Figure 10). With only a few BOP slots, collisions between coordinators are more frequent, with a negative impact on the packet delivery ratio. With no surprise, the original IEEE 802.15.4 algorithm performs badly, whatever the BOP duration is. Finally, we can remark having many BOP slots reduces the collision probability but consumes bandwidth: almost all the active part of the superframe is dedicated to `beacons`. We also measured the collision ratio, i.e. ratio of coordinators which have at least one interfering coordinator in the same superframe. We can verify IEEE 802.15.4 presents the largest collision ratio. The greedy scheduling limits collisions, even with low BO values: even when the number of superframe slots is limited, the collision probability remains acceptable. Our greedy solution performs quite close to the ideal scheduling algorithm, which uses centralized a priori knowledge.

4. Optimizing capacity & energy consumption in IEEE 802.15.4 cluster-trees

4.1. Problem statement

Because of the organization into superframes, we have a kind of distributed TDMA scheduling. In particular, all the nodes must transmit their packets to their coordinator (parent) during the active part of the superframe. Since all the active parts are contained into slots of the same size (i.e. Superframe Duration), we may encounter bottlenecks if the cluster-tree is not well balanced. Indeed, too many children in a superframe means all these nodes must share the same radio bandwidth. Moreover, IEEE 802.15.4 is not particularly robust and the number of collisions quickly increases when many nodes participate to the

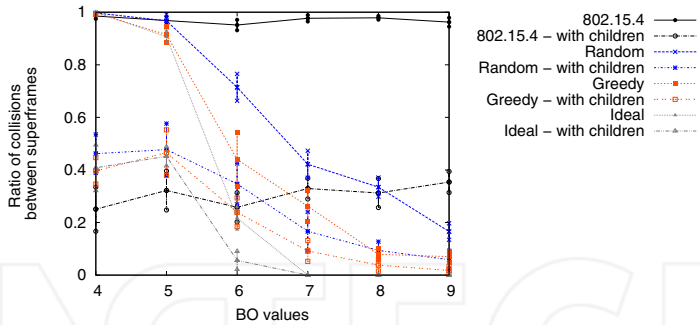

Figure 11. Collisions between superframes

same superframes, leading to collisions, wasting bandwidth. We propose here to formulate this problem as a Mixed Integer Linear Problem (MILP). We will translate all the cluster-tree relations, bandwidth sharing, and IEEE 802.15.4 mechanisms into linear constraints. If we define a linear objective, we face to a classical optimization problem. Our MILP formulation uses the radio and interference topology as an input. Thus, it provides an upper-bound, i.e. what we would obtain with an optimal centralized solution. In particular, we highlighted there exists a large gap to fill: we must still propose a distributed algorithm constructing an efficient cluster-tree.

The standard does not specify which parent to choose when creating the cluster-tree: it just specifies how to associate with. Surprisingly, only few articles investigated the impact of this choice on the cluster-tree topology. Cuomo et al. studied in [8] the average number of children and the height of the tree while Claudios et al. investigated in [25] the dynamic properties of the structure.We propose to investigate how to define formally the cluster-tree structure to create a linear optimization problem.

4.2. Model & assumptions

We expose here all our notations and the models we use : in a heterogeneous IEEE 802.15.4 network, we model the network with a symmetric graph $G = (V, E)$, V being the set of devices and E the set of available links between them. Unidirectional links cannot be exploited in IEEE 802.15.4 and are simply dismissed from the topology. We denote by $N(u)$ the neighborhood of u in G. We associate to G a classical link-interference (or conflict) graph G_c, where each vertex is a radio link, and they are neighbor in G_c if they interfere with each other.

4.2.1. Bandwidth and collisions

As the proposed model must be clearly independent from the PHY layer, we consider the radio bandwidth is equal to 1 unit, and is distributed between interfering transmitters. Collisions may decrease available bandwidth. Even if we assume that `beacons` do not collide during the BOP, CSMA-CA slotted mode does not permit to avoid entirely collisions. This is particularly true when the medium has to forward a large number of packets from different transmitters. A simple and efficient model has been proposed in [26]. Considering collisions are more frequent when the number of transmitters increases, authors demonstrated the percentage of radio bandwidth wasted by collisions is roughly proportional to the number

of nodes. Interpolating these results, we estimate the bandwidth wasted due to collisions BW_{coll} follows Equation 3 with a coefficient of determination of 0.97:

$$BW_{coll} = nb_stations * 0.018675 + 0.0104 \tag{3}$$

where *nb_stations* is the number of devices in contention. Note that a superframe should contain at most 53 stations, otherwise all available bandwidth may be wasted by collisions.

4.2.2. Superframes

An efficient way to model the schedule of coordinator's superframes would consist in dividing time into n_{slot} superframes slots. Each active part of a superframe is included in exactly one superframe slot numbered from 0 to $n_{slot} - 1$. Assuming all the nodes have the same beacon period BI and superframe duration SD values (e.g. BO and SO are imposed by the PAN), we have:

$$n_{slot} = \frac{BI}{SD} = 2^{BO-SO}. \tag{4}$$

IEEE 802.15.4 specifies the superframe of one node and its parent must be interspaced by `StartTime`. If we consider `StartTime` is a constant, we should choose `StartTime`=SD. In other words, if a node uses the superframe slot k to transmit its superframe, its child will use the superframe slot $k + 1$. Moreover, we assume that the PAN coordinator will use slot 1. Thus the depth of a node in the cluster-tree gives deterministically its superframe slot. Since the number of timeslots is bounded, we have a kind of modulo: a node with a large depth in the cluster-tree will re-use the superframe slot of the PAN coordinator. If we relax this constraint and and we consider `StartTime` is not anymore a constant (as in Section 3), we have rather to schedule the superframes to limit interference.

4.3. MILP formulation

We present a MILP formulation for the construction of a cluster-tree topology. Input data can be summarized as a collection of devices and the set of all available links represented with a graph $G = (V, E)$ and the corresponding link-interference graph G_c. A *milp* formulation consists in defining a problem using a set of unaffected variables, a set of linear constraints between them, and an objective function. Then a *milpsolver* produces a valid variables affectation respecting all the constraint while maximizing / minimizing the objective function. The domain definition and the type of each variable (continuous or integer) must be given as an input to the solver.

4.3.1. Variables

The following part describes all the variables used in our formulation, and their meaning in the resulting affectation. For clarity we have regrouped variables according to their use.

Role definition: we define for each node $u \in V$ a binary variable $\mathbf{root}(\mathbf{u}) \in \{0, 1\}$ such that $root(u) = 1$ if and only if u is the network PAN coordinator, and $root(u) = 0$ in all other cases. Any FFD may become coordinator as soon as it is associated to the cluster-tree. Inversely, a RFD is only a leaf of the tree, and cannot accept children. We model this different by using variables $\mathbf{coord}(\mathbf{u}) \in \{0, 1\}$, for each node $u \in V$. $coord(u) = 1$ iif u is a FFD.

Superframe Scheduling: in order to assign each coordinator a superframe slot, we introduce integer variables $\mathbf{slot}(\mathbf{u}) \in \{0, n_{slot} - 1\}$ for each node $u \in V$ such that $slot(u) = k$ if and only if node u uses the superframe slot number k. When defining the constraints, we need

also the additional variables $\mathbf{superframe(u,k)} \in \{0,1\}$ for each node $u \in V$ and each integer $k \in [0, n_{slot}-1]$. $superframe(u,k) = 1$ iif coordinator u is assigned to superframe slot k, or $superframe(u,k) = 0$ otherwise. There is a natural equivalence between $slot(u)$ and $superframe(u,k)$. Indeed, $slot(u) = k$ iif $superframe(u,k) = 1$. We also note that nodes acting as end-devices won't be assigned a superframe slot, thus we state $superframe(u,k) = 0$ for each RFD and each $k \in [0, n_{slot}-1]$ by reducing the definition domain for concerned variables.

Link activity: our main objective consists in determining the subset of links from E restricted to the cluster-tree topology. Only these links may be used to forward the traffic. For each link $(u,v) \in E$ we use a binary variable $\mathbf{child(u,v)} \in \{0,1\}$ such that $child(u,v) = 1$ iif u is a child of node v, and $child(u,v) = 0$ otherwise. A link may be used during the superframe of the corresponding coordinator. We use the binary variables $\mathbf{act(u,v,k)}$ for each radio link $(u,v) \in E$ and each integer $k \in [0, n_{slot}-1]$. $act(u,v,k) = 1$ iif the link (u,v) is active during slot k, i.e. v is a coordinator using timeslot k and u is a child of v. Else, we have $act(u,v,k) = 0$.

Data flow: We define continuous variables $\mathbf{f(u)} \in [0,1]$ as the quantity of data generated by the node u to the PAN coordinator. Then, for each $link(u,v) \in E$ and each integer $k \in [0, n_{slot}-1]$, we note $\mathbf{f(u,v,k)} \in [0,1]$ the traffic from node u to node v using superframe slot k. We also use the continuous variables $\mathbf{f(u,v)} \in [0,1]$ for each link $(u,v) \in E$. $f(u,v)$ is the normalized quantity of bandwidth consumed by the radio link from u to v. It represents the fraction of time during which u transmits its packets to v.

4.3.2. Constraints

We now translate the IEEE 802.15.4 structure and mechanisms into linear constraints.

Tree structure: any device excepted the PAN coordinator has to be associated to exactly one parent node. As $child(u,v)$ are binary integer variables, constraint 5 guarantees the parent unicity. Besides, the PAN coordinator has no parent since it is the root of the cluster-tree (constraint 6). We guarantee the cluster-tree is connected by stating $f(u) > \epsilon$ for each $u \in V - \{PC\}$.

$$\forall u \in V - \{PC\}, \quad \sum_{v \in N(u)} child(u,v) = 1 \tag{5}$$

$$\sum_{v \in N(PC)} child(PC,v) = 0 \tag{6}$$

A node with at least one child must be a coordinator (constraint 7). Reciprocally, a node without child does not maintain a superframe (constraint 8).

$$\forall u \in V, \forall v \in N(u) \quad child(v,u) \leq coord(u) \tag{7}$$

$$\forall u \in V, \quad coord(u) \leq \sum_{v \in N(u)} child(v,u) \tag{8}$$

Superframe scheduling: the following constraint imposes that at most one superframe slot is assigned to any coordinator, and that no slot is assigned to a leaf node:

$$\forall u \in V, \quad \sum_{k \in [0, n_{slot}-1]} superframe(u,k) \leq coord(u) \tag{9}$$

We can also establish a link between variables superframe(u,k) and slot(u) with equation 10:

$$\forall u \in V, \sum_{k \in [0, n_{slot}-1]} superframe(u,k) = slot(u) \tag{10}$$

Each coordinator must be assigned a slot immediately consecutive to it parent, excepted if the parent uses the largest slot number. Else, we have a modulo and the slot number of the child will be 0. We propose to split this problem into 3 constraints:

1. if u is a child of v, the difference between both slots cannot be superior to 2:

$$\forall (u,v) \in E, slot(u) < slot(v) + 2 + (1 - child(u,v)) \times n_{slot} \tag{11}$$

   The part $(1 - child(u,v)) \times n_{slot}$ simply inhibits this constraint when u is not the child of v.

2. in the same way, if u is a child of v, the slot number of u must be at least equal to the slot number of v plus 1, except if u is a leaf (the second line inhibits the constraint in such situation), or if the maximum slot number is assigned to v (third line). Fourth line inhibits this constraint when u is not the child of v.

$$\begin{aligned} \forall (u,v) \in E, slot(u) \geq slot(v) + 1 \\ - (1 - coord(u)) \times n_{slot} \\ - superframe(v, n_{slot} - 1) \times n_{slot} \\ - (1 - child(u,v)) \times n_{slot} \end{aligned} \tag{12}$$

3. Finally, if a node v uses the largest slot $(n_{slot} - 1)$, then its child u uses the slot 0:

$$\begin{aligned} \forall (u,v) \in E, \quad slot(u) \leq 2(n_{slot} - 1) - (n_{slot} - 1) \times child(u,v) \\ - (n_{slot} - 1) \times superframe(v, n_{slot} - 1) \end{aligned} \tag{13}$$

Bandwidth sharing: a node u use the superframe slot of it parent v to exchange packets. Thus, the link (u,v) has be active during the slot k:

$$\forall k \in [0..n_{slot} - 1], \forall (u,v) \in E, f(u,v,k) \leq superframe(v,k) \tag{14}$$

$$\forall (u,v) \in E, \forall k \in [0..n_{slot} - 1], f(u,v,k) \leq act(u,v,k) \tag{15}$$

Then the overall bandwidth from u to v is given as follows :

$$\forall (u,v) \in E, \quad \sum_k f(u,v,k) = f(u,v) \tag{16}$$

Since we consider the superframe slots have the same size, the bandwidth dedicated to one superframe is the radio bandwidth (1 unit) divided by the number of slots. Besides, bandwidth is shared among interfering nodes for their data packets and the corresponding collisions. If we consider the conflict graph, interfering links are neighbors. Thus, we have just to reference all the cliques (interfering links) and verify the bandwidth sharing constraint holds for each clique independently:

$$\forall k \in [0..n_{slot} - 1], \forall c \in \mathcal{C}, \sum_{(u,v) \in c} f(u,v,k) \leq \frac{1}{|n_{slot}|} \left(1 - 0.018675 \sum act(u,v,k) - 0.0104\right) \tag{17}$$

where $\mathcal{C}$ is the set of all the cliques in the interference-link graph G_c.

Flow conservation: Only links between children and parents are available for data flow transmission :

$$\forall (u,v) \in E, \quad f(u,v) \leq child(u,v) \tag{18}$$

Beside, we use the classical flow conservation constraints. Each node sends to its parent both the data packets it generated toward the PAN coordinator and the data packets forwarded from its children:

$$\forall u \in V - \{PC\}, \sum_{p \in N(u)} f(u,p) = \sum_{c \in N(u)} f(c,u) + f(u) \tag{19}$$

Since the PAN coordinator is the single destination (converge case), the following constraint holds:

$$\sum_{u \in N(PC)} f(u,PC) = \sum_{u \in V} f(u) \tag{20}$$

4.3.3. Objective

Our MILP formulation can be associated to any objective defined with linear function. We may maximize the network fair capacity:

$$Obj: \quad max\,(min\,(f(u))) \tag{21}$$

We may also maximize the global network throughput, without guaranteeing fairness, or aim at maximizing the network lifetime by defining linear constraints for the energy consumption and then minimize the maximum energy consumed by each node.

4.4. Numerical results

We illustrate here the performances of the greedy algorithm (a node chooses the first parent to associate with) compared to the upper bound obtained through the MILP formulation (denoted as *optimal* in the graphs). We distribute randomly the nodes in a disk. We consider here 4 timeslots for the superframe scheduling, and use cplex v.12 to solve the MILP formulation. We also plot the 95% confidence intervals. We measure the aggregated network capacity while maintaining the density constant (8 neighbors)(Figure 12). Clearly, associating to the first available parent is sub-optimal: bottlenecks quickly appear, degrading the network capacity. On the contrary, the optimal solution leads to a constant global throughput whatever the number of nodes is (i.e. collisions are very seldom). While this formulation uses the a priori knowledge of the radio and interference topology, we highlighted here the fact that existing cluster-tree construction algorithms are not efficient to optimize the network throughput and the energy consumption.

5. Self-configuring IEEE 802.15.4 version

5.1. Problem statement

IEEE 802.15.4 uses a backoff mechanism to solve contention and retransmissions to cope with the half-duplex and unreliability properties to a radio transmitter. It must cope with any topology while maximizing the throughput and minimizing the energy consumption. However, parameter values greatly impact the performances of the protocol (as highlighted in simulations we will include here). Consequently, we must propose a self-configuration

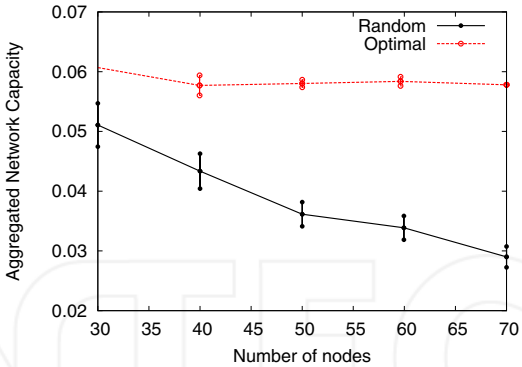

Figure 12. Network capacity vs. nb. of nodes (avg. degree of 8)

method to find always the best parameters values to operate always in optimal conditions. We will present here how we can obtain such property. To optimize the throughput and to reduce the energy consumption, we must reduce the collisions since retransmissions mean bandwidth and energy wastage, and also reduce the backoff times since we waste bandwidth. Obviously, the optimal solution will find the tradeoff between these two antagonist objectives. An analytical model has been presented in [27] to find the optimal parameters values in IEEE 802.15.4. They propose to solve an optimization problem concerning the delay, power-consumption and reliability. The nodes must estimate continuously the busy channel probability and the channel access probability. While IEEE 802.15.4 suggests default values for the parameters (e.g. `BE`, `NB`), they do not lead to optimal performances. Thus, a method has been presented in [18] to maintain a targeted packet delivery radio (PDR). A too large PDR means we waste bandwidth because of large backoff values while a small PDR would not be acceptable for some applications. We consider maximizing the throughput is more efficient: we do not need to drop packets when the traffic is light.

5.2. Defining good parameter values

We propose here to evaluate the impact of each parameter individually.

Contention Window (`CW`): `CW` represents the number of times a node has to trigger a CCA before transmitting its frame. CW should be equal to 2: a node will avoid a collision when the medium is free and the source is waiting for the `acknowledgement`. Thus, the optimal `CW` is its default value.

`macMaxFrameRetries`: `macMaxFrameRetries` denotes the number of times a frame is retransmitted when the `ack` is not received. Anastasi et al. [18] highlighted the fact that packets are very seldom dropped because of too many retransmissions ($<$ 2% when `macMaxFrameRetries`=2).

To optimize the throughput, we may adopt an adaptive approach, depending on the traffic load. When the queue contains many frames to transmit, it should use a small `macMaxFrameRetries`: even if a packet is dropped, others have to be transmitted. On the contrary, a small queue length means we may authorize more retransmissions to *fill* the

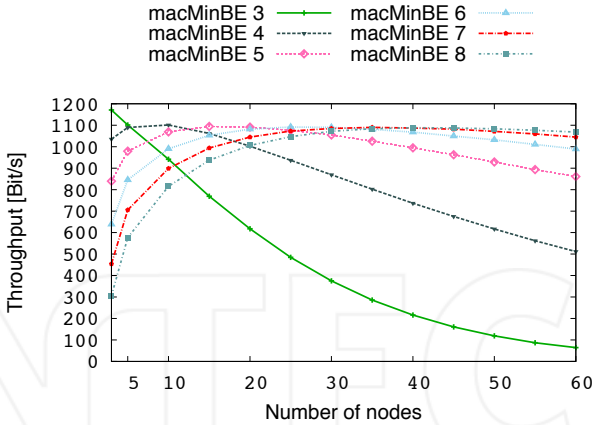

Figure 13. The throughput in saturating mode for IEEE 802.15.4

medium. The standard defining a maximum number of retransmissions of 4, the value may be obtained as following:

$$macMaxFrameRetries = (q_{max} - q(t)) * 4 \tag{22}$$

with q_{max} (resp. $q(t)$) the maximum (resp. current) queue length.

Backoff Exponent (BE): The backoff exponent (BE) defines the range of the random value for the backoff time ($backoff \in [0..2^{BE} - 1]$). Obviously, the backoff exponent has an impact on the throughput. However, its optimal value depends on the number of simultaneous transmitters (cf. Figure 13), which is practically complicated to estimate. We propose in the next section an adaptive approach to always operate close to the optimal BE value.

macMaxCSMABackoffs: `macMaxCSMABackoffs` denotes the maximum number of times a node can choose a new backoff value when it senses the medium busy after a CCA. Figure 14) illustrates the impact of `macMaxCSMABackoffs` on the throughput for different traffic loads. A large `macMaxCSMABackoff` value increases the throughput. Indeed, a CCA does not *consume* bandwidth: dropping a packet after a busy CCA is quite aggressive. Obviously, `macMaxCSMABackoff` does increase the throughput only in unsaturated mode: we have to *fill* the whole superframe. We propose the following simple method. When the node has still packets to transmit although the active part has elapsed, it decreases the `macMaxCSMABackoff` value. Inversely, a node which has an empty buffer before the end increases the `macMaxCSMABackoff` value.

5.3. Self-adaptation mechanism for the Backoff Exponent value

To increase the networks lifetime, the nodes must switch their radio off most of the time. The bandwidth is consequently reduced in the same proportion. Besides, all the nodes which participate to a superframe wake-up synchronously and try to send the frames they have buffered since the last beacon of their parent. Thus, even a low traffic may create an high collision rate at the beginning of the active part of the superframe. We must take care of efficiently sharing the bandwidth and reducing the collision rate. We proposed in the previous

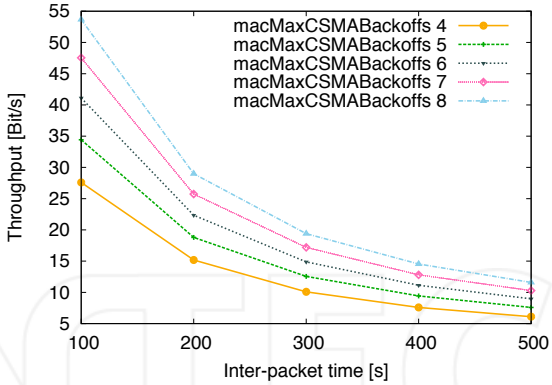

Figure 14. Impact of macMaxCSMABackoff on the throughput

section methods to find the optimal values for most of the parameters. We focus here on the Backoff Exponent, which is practically the most complicated to adjust dynamically.

Our objective consists in limiting bandwidth wastage. A too long backoff leads to medium under-utilization: the nodes wait a too long time, wasting bandwidth. Inversely, a collision *consumes* bandwidth. However, to estimate the sum of bandwidth dedicated to collisions and backoffs is a complicated task. Instead of minimizing bandwidth wastage, we propose rather to maximize the throughput, the antagonist metric. Indeed, we have highlighted the best throughput is stable in Figure 13, i.e. whatever the parameters values, we may obtain at most a throughput of 1,100 b/s. Thus, we *just* have to find the BE value which offers this throughput.

5.3.1. Dynamic exploration

We propose a dynamic exploration algorithm to find the most appropriate BE value. By appropriately testing some *good* BE values, we try to locally optimize the throughput. Indeed, we can remark for a fixed number of nodes, the throughput is monotonically increasing or decreasing when the BE value increases/decreases. We test continuously different values, oscillating around the optimal throughput. Besides, we can remark in Figure 13 the second best BE value leads to a throughput very close to the optimal throughput. Thus, oscillating between both values wastes a limited amount of bandwidth. While the throughput is increasing, we keep on exploring the same direction (either increasing or decreasing the BE value). As soon as the throughput is sub-optimal, we inverse the direction.

5.3.2. Stabilization

A coordinator reaching the optimal throughput should maintain the same BE. Since we have to cope with statistical variations, we consider the throughput has not changed if its difference with the previously measured throughput is inferior to a threshold (Δ= 1 packet/Superframe Duration). To avoid freezing the self-adaptation, a node MUST initiate a new exploration, even if the throughput does not change for a long time. When BE stays unchanged for k successive superframes (3 in our simulations), the coordinator MUST explore another BE value in the same direction as the previous exploration.

Algorithm 1: Updating the BE value

```
1  /* Initialization                                                    */
2  if isCoordinator then
3      myMacMinBE ←3 ;
4      direction ←+1 ;
5      oldTroughput ←0;
6  /* Children read the macMinBE directly in the beacons                */
7  if ! isCoordinator then
8      receiveBeacon;
9      macMinBE ←readValueFromBeacon;
10     return ;
11 /* the coordinator must continuously update its BE                   */
12 if isCoordinator then
13     changeBE = BEUnchangedDuring (k * SD) ;
14     /* the throughput is varying                                     */
15     changeBE |= (newThroughput < oldTroughput- Δ) ;
16     changeBE |= (newThroughput > oldTroughput + Δ) ;
17     if changeBE then
18         /* the throughput has decreased:  let change the direction   */
19         if newThroughput < oldTroughput- Δ then
20             direction ←( - direction);
21         myMacMinBE ←myMacMinBE + direction;
22     /* the coordinator sends the BE in beacons                       */
23     sendBeacon(myMacMinBE);
24     oldTroughput ←newThroughput;
```

5.3.3. Consistency

Nodes participating to the same superframe should not use different BE values. Else, IEEE 802.15.4 would become unfair: nodes with a lower BE would monopolize medium access. We propose consequently the BE is announced in the beacons from the coordinator.

The final algorithm is described in Algorithm 1. We use default values on the beginning (lines 2-5). Children extract the BE from beacons and stop any computation (lines 6-10). A coordinator has to continuously update its BE: when the throughput is decreasing, the node changes its direction for the exploration (lines 19-21). When the throughput has changed significantly (more than Δ) or stayed unchanged for a too long time, the BE is increased/decreased, depending on the direction (line 22). Finally, the coordinator saves the current throughput value and piggybacks the new BE in its beacons.

5.4. Performance evaluation

We have evaluated Algorithm 1 to find dynamically the best BE value with the original version of IEEE 802.15.4 with a static BE value. We simulate the protocol with the same parameters as in section 3.5. Figure 15 shows the performance of our protocol (SAV 802.15.4) in saturated mode (i.e. a node has always packets to transmit). We observe that our protocol provides almost optimal throughput, whatever the number of nodes. We also verified our self-adaptation algorithm reacts well to multihop topologies (Figure 16). We can verify the self-adaptation algorithm succeeds to oscillate around the optimal BE and maximizes the throughput.

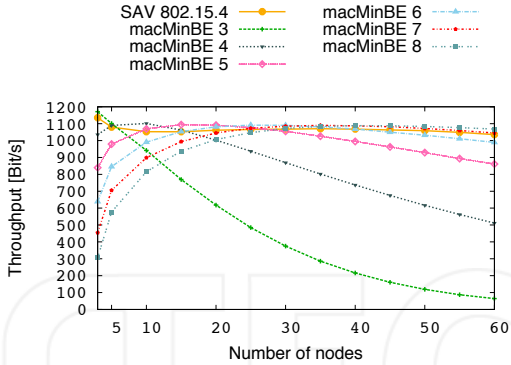

Figure 15. The throughput in saturating mode for IEEE 802.15.4

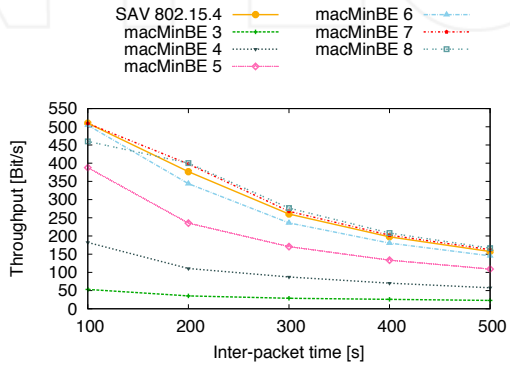

Figure 16. The throughput in non saturating mode of multihop IEEE 802.15.4

6. Open challenges

6.1. Topology formation

In section 4, we have presented a method to evaluate the capacity and energy consumption inherent to any cluster-tree. By providing a tight upper-bound, we can estimate the gap to fill before obtaining an optimal cluster-tree concerning e.g. energy, bandwidth. However, how to reach this objective in a distributed way is still an open problem. Yu *et al.* [28] proposed to minimize the number of clusters: aggregation is simplified, and more wireless routers can sleep. However, we could exploit with benefit the low duty-cycle property of IEEE 802.15.4 to save energy. What strategy is the most efficient in these scenarios? We have also to jointly optimize the scheduling and the cluster-tree hierarchy in a convergecast scenario to minimize the end-to-end delay or at least upper-bounding it. To construct a list of interfering nodes, Tseng *et al.* [29] proposed to use a larger transmission power for neighborhood discovery. However, such process will break the other `data` transmissions impacting negatively the performances. Besides, a trade-off between over and under-estimation is practically complicated to achieve. Moreover, many collisions may occur before converging to a legal scheduling.

6.2. Real-time traffic

IEEE 802.15.4 provides Guaranteed-Time-Slots (GTS) for real time traffic: these slots have to be pre-reserved during the Contention Access Period. In the last version of the standard, the number of GTS can be extended to cope with the particular requirements of the scenario. Park *et al.* [30] studied how to minimize energy consumption for real-time traffic in IEEE 802.15.4 networks. A multihop flow must reserve the GTS along all the path toward the PAN coordinator. Besides, a flow may not require one GTS during each superframe: it depends on the Beacon Order (BO) value and on the data rate of the flow. In that case, the coordinators must multiplex different real-time flows in the same GTS: this corresponds to a bin-packing problem, without knowing a priori the next real-time traffic requests. Finally, some bursty sensitive traffic may be generated because of e.g. alarms. These transmissions should be protected while minimizing the end-to-end delay.

6.3. Metric

A node has to choose the best parent in the cluster-tree. Obviously, it has to define what metric is the most accurate to represent the aptitude of a node to serve as parent. This property depends on selfish (offered bandwidth, reliability, delay) and global metrics (energy consumption toward the PAN coordinator, load-balancing). Thus, a node must take a decision after having captured all these criteria.

6.4. Stability

Testbed often highlighted real conditions lead to instabilities. Bezahaf *et al.* [31] demonstrated routes are very unstable in wireless mesh networks with DSDV. In the same way, Silva *et al.* [32] highlighted the signal is also very time-varying in Wireless Sensor Networks. We must absolutely prevent such oscillations in the cluster-tree. A node may re-associate to another parent because it improves the local and global performances of IEEE 802.15.4. However, it should avoid a domino effect, creating a wave of re-associations.

6.5. Experimental validation

IEEE 802.15.4 has already been extensively studied by simulations: slotted IEEE 802.15.4 was implemented for ns2 in [33], for opnet in [34, 35] (with or without GTS respectively). An analytical framework to model the IEEE 802.15.4 behavior is presented in [36]. Pollin *et al.* [26] presented a Markov Chain for both the saturated and unsaturated traffic scenarios. Chen *et al.* [37] focused on the industrial low rate WPAN and provided an omnet++ implementation. However, only a few real implementations exist. For instance, contiki does not support yet beacon-enabled IEEE 802.15.4. TinyOS has two main implementations:

- open-ZB propose an open source implementation of the beacon-enabled mode of IEEE 802.15.4 with GTS. It also includes the Zigbee features;
- tkn154 includes the full slotted IEEE 802.15.4 except a few features like PAN id conflict notification.

IEEE 802.15.4 has been evaluated with a reactive routing protocol (AODV) on topologies of 6 nodes in [38]. The authors measured only the end-to-end delay for the packets of a flow. The beacon-enabled and non-beacon modes were also compared in a star topology in [39]. More complex scenarios should be evaluated to definitively prove the accuracy of IEEE

802.15.4. In particular, is IEEE 802.15.4 scalable? What are its performances with dozens of hundreds of nodes? Similarly, how does a IEEE 802.15.4 network perform in harsh conditions: outdoor and/or high interference environments, high traffic conditions with large densities, co-existence with other WLAN or WPAN, etc.

7. Conclusion

IEEE 802.15.4 is emerging as the standard for Low-Power Wireless Personal Area Networks. We have presented here how we should modify IEEE 802.15.4 to cope with multihop environments: we must be able to forward traffic while limiting energy consumption. Slots should be carefully scheduled in a distributive way to forward traffic without collisions since it wastes energy. Moreover, a node should choose appropriately its parent to balance the load and energy: the cluster-tree should avoid the creation of bottlenecks. Finally, we have presented a very simple algorithm to find dynamically the best parameters values in IEEE 802.15.4. This enhanced version of IEEE 802.15.4, only modifying slightly the standard optimizes greatly the performances, enabling new types of multihop applications.

Acknowledgement

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

Author details

Abdeddaim Nazim
Grenoble Informatics Laboratory, France

Darties Benoît
LE2I - University of Burgundy, France

Theoleyre Fabrice
CNRS - University of Strasbourg, France

8. References

[1] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon. Health monitoring of civil infrastructures using wireless sensor networks. In *international conference on Information processing in sensor networks (IPSN)*, Cambridge, USA, 2007. ACM.

[2] T. Ulrich. Wireless network monitors h2o: System saves resources, increases yield in cabernet vineyard. *Wines and Vines Magazine*, July 2008.

[3] IEEE 802.15.4-2006 standard. `http://www.ieee802.org/15/pub/TG4.html`, 2006.

[4] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *International conference on Embedded networked sensor systems (Sensys)*, pages 95–107. ACM, Baltimore, MD, USA 2004.

[5] A. Bachir, M. Heusse, A. Duda, and K.K. Leung. Preamble sampling MAC protocols with persistent receivers in wireless sensor networks. *IEEE Transactions on Wireless Communications*, 8(3):1091 –1095, March 2009.

[6] Wei Ye, J. Heidemann, and D. Estrin. An energy-efficient mac protocol for wireless sensor networks. In *INFOCOM*, volume 3, pages 1567 – 1576. IEEE, 2002.

[7] W. Ye and J. Heidemann. Ultra-low duty cycle MAC with scheduled channel polling. In *Conference on Embedded Networked Sensor Systems (SenSys)*, pages 321–334, USA, 2006.

[8] F. Cuomo, E. Cipollone, and A. Abbagnale. Performance analysis of IEEE 802.15.4 wireless sensor networks: An insight into the topology formation process. *Computer Networks*, 53(18):3057–3075, December 2009.

[9] Anna Abbagnale, Emanuele Cipollone, and Francesca Cuomo. A case study for evaluating IEEE 802.15.4 wireless sensor network formation with mobile sinks. In *International Conference on Communications (ICC)*, Dresden, Germany, June 2009. IEEE.

[10] Zigbee. `http://www.zigbee.org/`, 2005.

[11] Meng-Shiuan Pan, Chia-Hung Tsai, and Yu-Chee Tseng. The orphan problem in zigbee wireless networks. *IEEE Transactions on Mobile Computing*, 8(11):1573–1584, November 2009.

[12] N. Hadid, A. Guitton, and M. Misson. Exploiting a meeting channel to interconnect 802.15.4-compliant mobile entities: discovery and association phases. In *International Symposium on Computers and Communications (ISCC)*, pages 94–99. IEEE, june 2010.

[13] Andreas Willig, Niels Karowski, and Jan-Hinrich Hauer. Passive discovery of IEEE 802.15.4-based body sensor networks. *Ad Hoc Networks*, 8(7):742–754, September 2010.

[14] N. Karowski, A. C. Viana, and A. Wolisz. Optimized asynchronous multi-channel neighbor discovery. In *INFOCOM*, Shanghai, China, May 2011. IEEE.

[15] 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, 2012.

[16] C. Chaudet, D. Dhoutaut, and I.G. Lassous. Performance issues with IEEE 802.11 in ad hoc networking. *IEEE Communications Magazine*, 43(7):110 – 116, july 2005.

[17] IEEE 802.15 WPAN task group 4b (tg4b). `http://www.ieee802.org/15/pub/TG4b.html`, September 2004.

[18] G. Anastasi, M. Conti, and M. Di Francesco. The mac unreliability problem in IEEE 802.15.4 wireless sensor networks. In *International conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM)*, pages 196–203, Tenerife, Spain, October 2009. ACM.

[19] Jaejoon Cho and Sunshin An. An adaptive beacon scheduling mechanism using power control in cluster-tree wpans. *Wireless Personal Communications*, 50(2):143–160, 2009.

[20] R. Burda and C. Wietfeld. A distributed and autonomous beacon scheduling algorithm for IEEE 802.15.4/ZigBee networks. In *Internatonal Conference on Mobile Adhoc and Sensor Systems (MASS)*, Pisa, Italy, October 2007. IEEE.

[21] P. Selvan Muthukumaran, R. de Paz, R. Špinar, and D. Pesch. Meshmac: Enabling mesh networking over IEEE802.15.4 through distributed beacon scheduling. In *International Conference on Ad Hoc Networks (AdHocNets)*, Niagara Falls, Ontario, Canada, September 2009.

[22] B.C. Villaverde, R. De Paz Alberola, S. Rea, and D. Pesch. Experimental evaluation of beacon scheduling mechanisms for multihop IEEE 802.15.4 wireless sensor networks. In *SENSORCOMM*, pages 226–231. IARIA, July 2010.

[23] G. Chelius, A. Fraboulet, and E. Fleury. Worldsens: development and prototyping tools for application specific wireless sensors networks. In *International Conference on Information Processing in Sensor Networks (IPSN)*, Cambridge, Massachusetts, USA, 2007. ACM.

[24] N. Abddedaim and F. Theoleyre. Implementation of a wsnet module to simulate the IEEE 802.15.4 beacon-enabled mode in multihop topologies. Research Report 00590853, HAL, May 2011.

[25] F.J. Claudios, R. Radeke, D. Marandin, P. Todorova, and S. Tomic. Performance study of reconfiguration algorithms in cluster-tree topologies for wireless sensor networks. In *International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, France, 2007.

[26] S. Pollin, M. Ergen, S. Ergen, B. Bougard, L. Der Perre, I. Moerman, A. Bahai, P. Varaiya, and F. Catthoor. Performance analysis of slotted carrier sense IEEE 802.15.4 medium access layer. *IEEE Transactions on Wireless Communications*, 7(9):3359 –3371, September 2008.

[27] Pangun Park, Carlo Fischione, and Karl Henrik Johansson. Adaptive IEEE 802.15.4 protocol for energy efficient, reliable and timely communications. In *International Conference on Information Processing in Sensor Networks (IPSN)*, pages 327–338, Stockholm, Sweden, 2010. ACM/IEEE.

[28] C. W. Yu, C. Chang, and J. Chou. Minimizing the number of clusters in IEEE 802.15.4 wireless sensor networks. In *International Workshop on Wireless & Mobile Networks (WiMoN)*, 2009.

[29] Yu-Chee Tseng and Meng-Shiuan Pan. Quick convergecast in ZigBee/IEEE 802.15.4 tree-based wireless sensor networks. In *International workshop on Mobility management and wireless access (MobiWac)*, pages 60–66, Terromolinos, Spain, 2006. ACM.

[30] Myung-Gon Park, Kang-Wook Kim, and Chang-Gun Lee. Holistic optimization of real-time IEEE 802.15.4/ZigBee networks. In *International Conference on Advanced Information Networking and Applications (AINA)*, pages 443 –450. IEEE, march 2011.

[31] M. Bezahaf, L. Iannone, M. Dias de Amorim, and S. Fdida. Insights into the routing stability of a multi-hop wireless testbed. In *International Conference on Ad Hoc Networks (AdHocNets)*, Victoria, BC, Canada, 2010.

[32] A. R. Silva and M. C. Vuran. Development of a testbed for wireless underground sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2010:9:1–9:12, April 2010.

[33] Jianliang Zheng and Myung J. Lee. *Sensor Network Operations*, chapter A Comprehensive Performance Study of IEEE 802.15.4, pages 218–237. IEEE Press, 2006.

[34] Anis Koubaa, Mario Alves, and Eduardo Tovar. A comprehensive simulation study of slotted CSMA/CA for IEEE 802.15.4 wireless sensor networks. In *International Workshop on Factory Communication Systems (WFCS)*, Torino, Italy, June 2006. IEEE.

[35] P. Jurcik, A. Koubaa, M. Alves, E. Tovar, and Z. Hanzalek. A simulation model for the IEEE 802.15.4 protocol: Delay/throughput evaluation of the GTS mechanism. In *MASCOTS*, oct. 2007.

[36] Tae Ok Kim, Jin Soo Park, Kyung Jae Kim, and Bong Dae Choi. Analytical model of IEEE 802.15.4 non-beacon mode with download traffic by the piggyback method. In *Network and Parallel Computing (NPC)*, pages 435–444, Dalian, China, 2007.

[37] Feng Chen, Nan Wang, R. German, and F. Dressler. Performance evaluation of IEEE 802.15.4 LR-WPAN for industrial applications. In *Conference on Wireless on Demand Network Systems and Services (WONS)*, pages 89 –96, Garmisch-Partenkirchen, Germany, jan. 2008. IEEE.

[38] C. Gomez, P. Salvatella, O. Alonso, and J. Paradells. Adapting AODV for IEEE 802.15.4 mesh sensor networks: theoretical discussion and performance evaluation in a real environment. In *International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Niagara-Falls, USA, June 2006. IEEE.

[39] C. Suh, Z. Mir, and Y. Ko. Design and implementation of enhanced IEEE 802.15.4 for supporting multimedia service in wireless sensor networks. *Computer Networks*, 52(13), September 2008.