



**HAL**  
open science

# Multi-Channel Cluster Tree for 802.15.4 Wireless Sensor Networks

Nazim Abdeddaim, Fabrice Theoleyre, Franck Rousseau, Andrzej Duda

► **To cite this version:**

Nazim Abdeddaim, Fabrice Theoleyre, Franck Rousseau, Andrzej Duda. Multi-Channel Cluster Tree for 802.15.4 Wireless Sensor Networks. PIMRC 2012 - 23rd IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Sep 2012, Sydney, Australia. pp.590 - 595, 10.1109/PIMRC.2012.6362853 . hal-00931042

**HAL Id: hal-00931042**

**<https://inria.hal.science/hal-00931042v1>**

Submitted on 2 Jun 2020

**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.

# Multi-Channel Cluster Tree for 802.15.4 Wireless Sensor Networks

Nazim Abdeddaim\*, Fabrice Theoleyre<sup>†</sup>, Franck Rousseau\*, and Andrzej Duda\*

\*Grenoble Institute of Technology, CNRS Grenoble Informatics Laboratory UMR 5217, Grenoble, France

<sup>†</sup>CNRS, LSIIT, University of Strasbourg, France

**Abstract**—We propose MCCT (Multi-Channel Cluster Tree), a cluster-tree construction protocol for nodes in IEEE 802.15.4 beacon-enabled mode. By multiplexing transmissions across orthogonal channels, we reduce collisions between control and data frames, which leads to better packet delivery rate and fairness. We propose a method for constructing a cluster-tree suitable for minimizing beacon collisions. The protocol builds on a neighbor discovery procedure that uses a dedicated control channel while still sticking to the superframe structure of IEEE 802.15.4. We also specify a channel assignment and superframe scheduling method that takes into account channel diversity. We evaluate the proposed protocol through simulation and compare with other proposals: standard 802.15.4 and a representative of distributed solutions to the superframe scheduling problem—MeshMAC. The simulation results show that MCCT significant improves packet delivery ratio, delay, and fairness. It also results in very good packet delivery ratio for increased network density.

## I. INTRODUCTION

The IEEE 802.15.4 standard for Low-Rate Wireless Personal Area Networks (LR-WPANs) [1] defines the physical and the MAC (Medium Access Control) layers and two operating modes. In the *non-beacon* mode, all nodes use CSMA-CA for channel access with contention, which implies that they should be always awake to avoid deafness. The *beacon-enabled* mode aims at saving energy: each coordinator sends a beacon to delimit its superframes and invites its children to send their frames during the Contention Access Period (CAP). Thus, nodes can achieve low energy consumption, because a node can safely turn its radio off during the rest of the superframe and wake up at the next beacon.

A IEEE 802.15.4 network can have a star, peer-to-peer, or cluster-tree topology. However, the standard does not specify the details of the cluster-tree construction algorithm leaving its implementation open. ZigBee [2] defines a protocol for cluster-tree construction based on three constraints: a maximum number of children and a maximum number of router children, and a maximum depth. A coordinator node transmits its beacons at instants defined by its own schedule. The ZigBee approach suffers from the problem of beacon collisions. Koubaa et al. proposed a centralized algorithm for scheduling superframes and beacons to prevent collisions [3]. MeshMac is a distributed solution to the superframe scheduling problem [4]. MeshMac divides a superframe into slots, each slot containing the active part of a superframe (CAP and CFP, Contention Free Period). A coordinator chooses a slot not used in its 2-neighborhood to schedule its superframe. The solution works if there is a sufficient number of slots for scheduling superframes in a 2-neighborhood, so it imposes a constraint on the BO (Beacon

Order) and SO (Superframe Order) parameters of the network (the beacon interval should be long enough for accommodating the required number of slots).

Beacon-Only Period (BOP) is another solution proposed by Jeon et al. [5]: a superframe starts with a period composed of several slots for sending beacons. Coordinators advertise in their beacons the slot they use and the slots of their neighbors. Based on this information, a coordinator can choose a free beacon slot in its 2-neighborhood to transmit its beacon without collision. Nevertheless, the superframes of different coordinators may overlap, which results in increased contention and collisions. An experimental study showed that this solution is only suitable for very low traffic scenarios and low network density [6].

Some authors proposed to take advantage of multiple channels in a cluster-tree. TMCP (Tree Based Multi-channel Protocol) is a centralized solution proposed to minimize the interference in the tree by creating several subtrees rooted at the PAN coordinator: each subtree operates on an orthogonal channel [7]. However, nodes in the same subtree keep on colliding, because TMCP only reduces collisions among the 1-hop neighbors of the PAN coordinator.

The goal of the present paper is to propose MCCT (Multi-Channel Cluster Tree), a cluster-tree construction protocol for nodes in beacon-enabled mode that avoids beacon collisions, takes advantage of multiple channels, and results in a network with low energy consumption. MCCT organizes the association process of nodes wanting to join the tree and determines channels as well as superframe instants to use. It decides with which coordinator a node associates based on the information on the number of associated children and channels used in the coordinator neighborhood. The association implies that the node will use the channel chosen by the coordinator for further communications. If the node has sufficient resources (i.e., if it is a Full Function Device), it becomes a coordinator and chooses a different channel from its parent to avoid beacon collisions. As a coordinator, it can accept associations from other nodes wanting to join the network while limiting the number of children to reduce contention. The use of multiple channels eliminates beacon collisions and results in increased network capacity, which becomes important in networks with higher density.

To avoid scanning all channels before the association with a coordinator, we have decided to use a common control channel for all cluster-tree construction and maintenance operations. To join the network, a node only scans the control channel

for a control message from coordinators with an invitation to associate.

The number of children associated with one coordinator needs to be limited, because otherwise the contention for channel access after a beacon results in a significant number of collisions [8]. To lower the contention, coordinators in our protocol advertise the number of children, so nodes trying to associate will choose coordinators with a lower number of children.

Even if several previous papers had similar objectives [7], [9], [10], [11], [12], [13], [14], [15], our protocol solves all pending issues: channel allocation integrated with cluster-tree construction, elimination of beacon collisions, reduction of intra-cluster contention, and improvement of network capacity. Unlike many proposals [7], [9], [10], [12], [14], [13], [15], our approach closely sticks to the standard 802.15.4 protocol with beacon-enabled mode only requiring one new control frame.

We evaluate the proposed protocol through simulation and compare with other proposals: standard 802.15.4 and a representative of distributed solutions to the superframe scheduling problem—MeshMAC. The simulation results show that MCCT significantly improves packet delivery ratio, delay, and fairness. It also results in very good packet delivery ratio for increased network density.

## II. 802.15.4 BACKGROUND

We briefly review the support of 802.15.4 for cluster-tree construction.

In the cluster-tree topology, the PAN coordinator is the root of the network. It serves as a gateway and represents the first coordinator in the cluster-tree. All other nodes are unassociated at the beginning and they broadcast a discovery frame (active scanning) or wait for *beacons* (passive scanning) to join the network. Passive scanning is the only available discovery mechanism in the *beacon-enabled* mode.

When a node discovers a neighbor, it may choose it as a parent coordinator and associate with by exchanging control frames. After association, the node may become a coordinator itself: it periodically sends its *beacons* to invite other nodes to associate.

Beacons sent by coordinators also indicate the start of the data exchange period: they contain the list of destination addresses for frames stored at a coordinator. During the Contention Access Period, a child node either retrieves frames by transmitting a *data-request* frame if its address was present in the pending destination list, or transmits its *data* frames to the coordinator. To avoid collisions, all children nodes use the slotted CSMA-CA method to access the medium. Note that the coordinator never initiates a transmission, but only replies to solicitations from children nodes. They have to explicitly request their frames from a coordinator, which enables switching off their radio and saving energy without deafness.

A node may reserve a Guaranteed Time Slot for periodic transmissions during the Contention Free Period by trans-

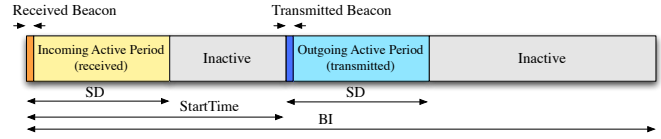


Fig. 1. Incoming and Outgoing superframe structure in IEEE 802.15.4

mitting a request during the Contention Access Period. If accepted, the slot is dedicated to the transmissions of the node.

Coordinators transmit *beacons* every Beacon Interval (BI) while a superframe lasts for a Superframe Duration (SD) (cf. Fig. 1). The Beacon Order (BO) and the Superframe Order (SO) values determine the BI and SD values.

To support low delay forwarding over multiple hops, the 802.15.4 working group defined the Outgoing (maintained by a coordinator) and the Incoming superframes (maintained by the parent node) interspaced by *STARTTIME* (cf. Fig. 1). Nodes may sleep during the inactive parts of the superframe. However, two coordinators with the same parent and the same *STARTTIME* value, will transmit their superframes simultaneously and a collision will occur. Several authors proposed solutions to reduce the number of collisions [3], [5], [4].

## III. RELATED WORK ON MULTIPLE CHANNEL MAC

We have briefly reviewed the most important related work on the cluster-tree construction in the introduction. Below, we refer to some proposals for taking into account multiple channels.

MMSN (Multi-Frequency Media Access Control for Wireless Sensor Networks) [9] was the first multichannel MAC protocol for WSN. Its operation is composed of two phases: frequency assignment and media access. In the frequency assignment phase, each node obtains a channel for data reception using four different frequency assignment strategies. The media access phase begins after channel assignment—nodes are synchronized and use time slots and different channels for data transmissions. As the protocol assigns channels after constructing the topology, inviting new nodes to join the network requires a new phase of frequency assignment.

The authors of MCMAC (Multi-channel MAC) proposed explicit channel reservation before each data exchange through a negotiation on a common control channel [11]. However, the explicit channel reservation results in important overhead and the control channel may become a bottleneck due to its high utilization.

The MC-LMAC (Multi-Channel Lightweight MAC) protocol [15] integrates both TDMA for timeslot selection and FDMA for contention free parallel transmissions on different channels. Each timeslot is divided into two parts: a control period and data transmission. During the control period, a node switches to the control channel waiting for a notification from a potential sender. If the node receives such solicitation, it switches to the communication channel of the sender to receive the data packet. The drawback of MC-LMAC is the overhead resulting from control messages sent before each data transfer.

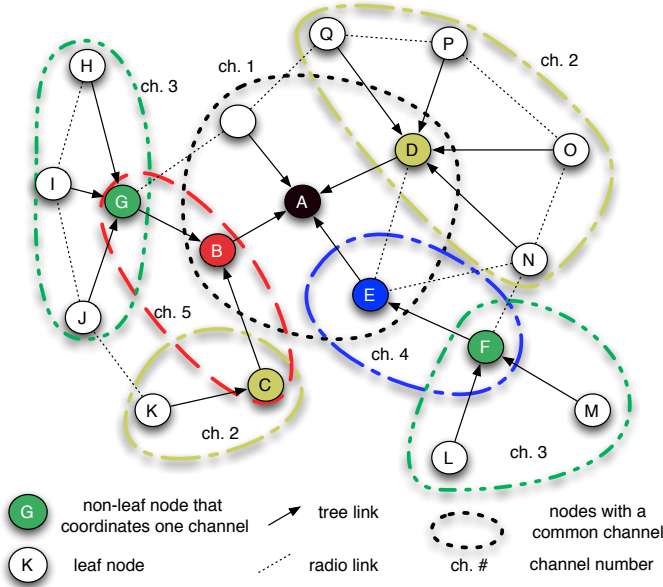


Fig. 2. Multichannel cluster-tree

Y-MAC also mixes TDMA and FDMA [13] by assigning one receiving timeslot for each node on a base channel. For traffic bursts, Y-MAC uses other channels defined in a hopping sequence specific to the receiver. Y-MAC suffers from the same flexibility and scalability problems as other TDMA based approaches due to fixed slot allocation.

#### IV. MCCT – MULTI-CHANNEL CLUSTER TREE

We propose a scheme for constructing a multichannel cluster-tree so that nodes can operate in parallel, i.e. superframes may overlap in time on separate channels.

Each coordinator chooses one channel to communicate with its children. Hence, a node must transmit upward frames in the cluster-tree on the channel chosen by its parent. Interfering coordinators should choose orthogonal channels to avoid beacon collisions.

Let us consider the example presented in Fig. 2. Node A is the PAN coordinator. Non-leaf nodes (B, C, D, E, F, and G) choose a channel to communicate with their children. We call a *cluster* the set of nodes composed of a coordinator and its children that use the same *cluster channel* for their transmissions, e.g. nodes D, N, O, P, Q operating on channel 2. Non-interfering clusters may choose the same channel (e.g. clusters of G and F).

##### A. Neighbor discovery

We propose to use a dedicated *shared control channel* for all cluster-tree construction and maintenance operations. To invite nodes to join the network, coordinators send `hello` frames once per BI on the control channel at random instants during the inactive period of a superframe (represented as `schedule_rand()` in Fig. 3). `hello` is a new type of control frames that contains all the information on the neighborhood of a given coordinator including the used channels (list of

neighboring ids, their superframe slot, their depth and the channel they used). A unassociated node listens to the control channel to find possible parents and associate with one of them according to the standard IEEE 802.15.4 association procedure.

The introduction of `hellos` decouples the association process from wake up synchronization by means of beacons that keep their role defined in IEEE 802.15.4. However, a `hello` frame also contains the scheduling information on how neighbor nodes use superframes to avoid beacon collisions.

Even if dedicating a channel to control traffic wastes 6% of the radio bandwidth (there are 16 channels in IEEE 802.15.4), it is necessary from the point of view of the association delay and the energy consumption during the discovery phase. Neighbor discovery on multiple channels would require scanning 16 channels for at least one BI, which may take considerable time, for instance, for  $BO = 14$ , a node has to listen during 96 minutes on the average to find a parent, which may drain a large part of the initially available energy. Moreover, random scheduling of `hellos` on the control channel prevents their collisions while bounding the association delay to one BI (collisions are still possible, however they are very rare due to long inactivity periods).

Fig. 3 illustrates the neighbor discovery process. Node A is the PAN coordinator and sends a `hello` on channel 0, the dedicated control channel. Node B receives the `hello` frame and associates with A in the next superframe. Then, B starts acting as a coordinator and schedules a `hello` at a random instant during its inactive period. Finally, node C receives a `hello` from B and associates with the network.

##### B. Cluster-tree construction

The construction of the cluster-tree begins with the PAN coordinator periodically transmitting `hellos` with depth 0 on the dedicated control channel. Nodes that want to join the network start in the *unassociated state*—they listen to the control channel to find a parent to associate with and to join the cluster-tree. They construct a *neighborhood table* based on the received `hellos`. When a node reaches the end of its scanning period that has the maximal duration of BI, it chooses a coordinator. Then, it executes the standard IEEE 802.15.4 association procedure on the cluster channel of the chosen coordinator during the active part of the superframe and becomes associated. With respect to other nodes, it may also become a coordinator: choose its own cluster channel for the communication with its future children based on the information on channels used by other coordinators (cf. the detailed algorithm in Section IV-D) and start sending `hellos`. We introduce two types of coordinators:

- *active coordinators* are non-leaf nodes in the cluster-tree that forward traffic: they send beacons on their cluster channel and are active during their superframe ready to receive traffic from their children;
- *passive coordinators* are leaf nodes: they do not send beacons and they are only active at the beginning of their superframe waiting for association requests from potential children.

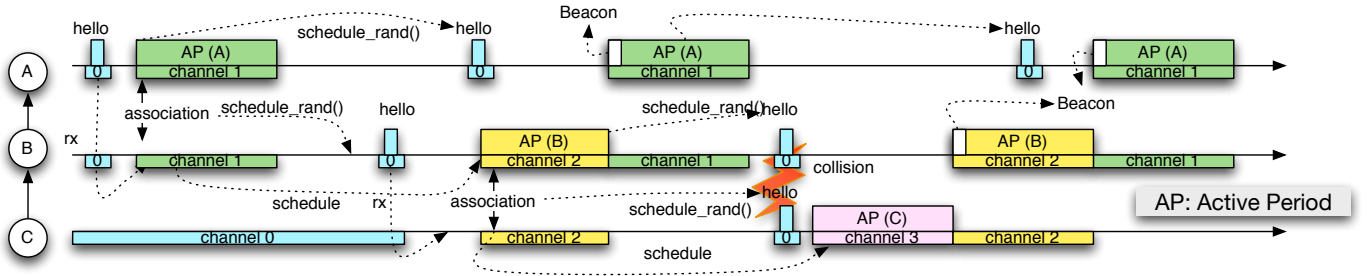


Fig. 3. Example of the neighbor discovery process

The distinction leads to energy savings, because passive coordinators do not have to stay awake during the superframe active period  $SD$ . Upon association, a Full Function Device becomes a passive coordinator, starts transmitting `hello`s on the control channel and scheduling its superframes, but without sending `beacons`. A Reduced Function Device may only become a leaf.

As soon as a node associates with a passive coordinator, the coordinator changes its state to active, i.e. it starts sending beacons and forwarding traffic on behalf of its children. Node *B* in Fig. 3 acts as a coordinator after its association: it schedules both its next superframes and `hello`s.

### C. Choice of a coordinator

When joining the network, a node chooses a coordinator with the smallest non-null number of children, which prevents creating a new active coordinator when not needed, saves active channels, and maximizes the number of leaves in the tree.

In case of several coordinators having the same number of children, a node chooses the closest coordinator to the root. If there are several coordinators with the same number of children and the same depth in the tree, the node randomly chooses one of the coordinators.

MCCT maximizes the number of leaves, because unassociated nodes must choose a parent that already has children nodes. Moreover, as nodes choose the parent with the smallest number of children, they tend to construct a balanced tree.

However, we need to limit the number of children associated with one coordinator, because excessive contention between nodes using the same cluster channel may lead to collisions and frame losses. So, we propose to constrain the node association to a parent that has less than *threshold* children. If all potential parents have too many children ( $> threshold$ ), a node may choose a coordinator without children, i.e. a passive coordinator. If there is no passive coordinator in the neighborhood, the node can choose an active coordinator with a number of children greater than *threshold*: connectivity must be guaranteed.

To determine a suitable value for the *threshold* parameter, we have run simulations of IEEE 802.15.4 in a star topology on a single channel varying the number of children. Fig. 4 presents the Packet Delivery Ratio (PDR) in function of the number of contending nodes. for  $BO = 13$  and  $SO = 6$  (other parameter values have consistently led to similar results).

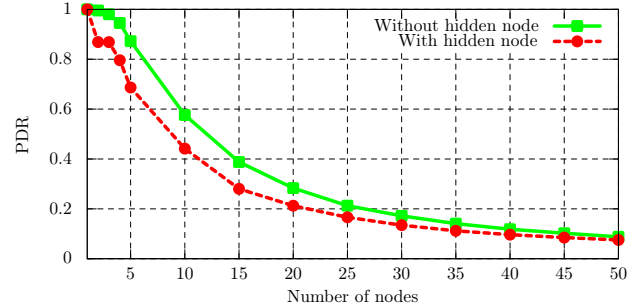


Fig. 4. 802.15.4 channel contention: PDR in IEEE 802.15.4 star topology ( $BO = 13$ ,  $SO = 6$ )

We can observe that PDR quickly drops with the increasing number of contending nodes (a similar evidence was already reported elsewhere [8]). Thus, a coordinator should limit the number of its children. The results show that to maintain an acceptable packet delivery ratio, e.g. larger than 80%, a coordinator should maintain at most 5 active children assuming the case without hidden nodes (to take into account hidden nodes, we can decrease the *threshold* parameter to at most 4).

### D. Channel and superframe slot assignment

To avoid the collision and deafness problems arising in multihop networks, two nodes in the same interference region should operate on distinct channels. At the same time, to avoid deafness, a sender has to be sure that the receiver is actually listening to the right channel before sending a frame.

Thus, each coordinator needs to choose a cluster channel not used in its neighborhood for its superframes. All children nodes are tuned to the cluster channel of their parent during the active period of the parent superframe. To avoid collisions, the active periods of the coordinator and its parent should not overlap. We divide a superframe into *superframe slots* to define non-overlapping periods: they are active periods of  $SD$  duration placed at multiples of  $SD$ .  $STARTTIME$  serves as an offset to define the start of a given slot, for instance  $STARTTIME = 0$  for the slot 0 and  $STARTTIME = SD$  for slot 1. There are  $2^{BO-SO}$  slots to schedule superframes. If a node chooses a different slot from its parent, collisions may only arise among coordinators with the same depth modulo  $2^{BO-SO}$ .

The choice of the channel and superframe slot is based on the information contained in `hello`s: coordinators in-

clude a map of channels and superframe slots so that nodes can construct a list of channels and slots used in their 2-neighborhood. In this way, two interfering nodes belonging to different branches of the cluster tree are aware of their respective superframe slots and thus can avoid collisions.

We aim at a channel allocation scheme with a low complexity. The number of available channels makes a greedy approach possible—coordinators can apply a randomized algorithm to select their cluster channel:

- 1) they sort channels in the descending order of the number of interferers during the superframe slot of the node (interferers are the nodes that use the same channel and the same slot)
- 2) they randomly select one of the least used channels.

To minimize the end-to-end delay for upward traffic, a node will choose to maintain a superframe finishing when the superframe of its parent begins. Fig. 3 illustrates how nodes *A*, *B*, and *C* choose their respective superframe slots so that the upward traffic benefit from short delays.

#### E. Synchronization requirements

The multi-channel operation requires the same synchronization constraints as the original IEEE 802.15.4—superframes are maintained in the same way as in IEEE 802.15.4 and can safely use *beacons* to cope with clock drifts. We can also add a guard time before starting the superframe by taking advantage of synchronization based on *beacons*.

#### F. Characteristics

The algorithm converges to the state in which each node has a single parent and is a part of a cluster-tree rooted at the PAN coordinator. The resulting cluster-tree presents the following characteristics:

- Limited collisions: interfering nodes with overlapping superframes (e.g. siblings) will use orthogonal channels.
- Energy-efficiency: we maximize the number of leaves and minimize the number of active coordinators, which is efficient for saving energy.
- Low route stretch factor: minimizing the number of forwarding nodes results in shorter routes and fewer hops to reach destinations, hence saving energy and bandwidth.

#### G. Network maintenance

During its operation, a node needs to monitor connectivity towards the PAN coordinator. For this purpose, we use the sequence number in *beacons*: the PAN coordinator increments the sequence number of IEEE 802.15.4 frames in every *beacon* and coordinators include the sequence number of the last received *beacon* in their own *beacons* and *hellos*. A node detects the loss of connectivity when the sequence number remains unchanged for several BI (typically 3 in our simulations). In this case, it changes its state to unassociated and if it is an active coordinator, stops transmitting *beacons*. Its children detect that their parent is unassociated and in turn become unassociated. The same happens to all nodes in the

TABLE I  
SIMULATION PARAMETERS

<b>Number of nodes</b>	60 (random)
<b>Avg. No. of neighbors</b>	9
<b>Traffic type, rate</b>	CBR, 0.5 pkt/min
<b>802.15.4 parameters</b>	SO = 1, BO ∈ [2..14]

sub-tree so they restart the association procedure as described above.

Notice that since the detection may take several BI at each level of the sub-tree, a disconnected coordinator may try to quickly associate with another parent if possible, which protects from the destruction of the entire sub-tree and results in a fast recovery. Sequence numbers in *beacons* avoid the creation of loops in the cluster-tree during recovery.

## V. PERFORMANCE EVALUATION

We have compared MCCT with the standard IEEE 802.15.4 and MeshMAC [4] in terms of packet delivery ratio, delay, and fairness. In the standard IEEE 802.15.4, a node waits for a beacon and associates with the sender.

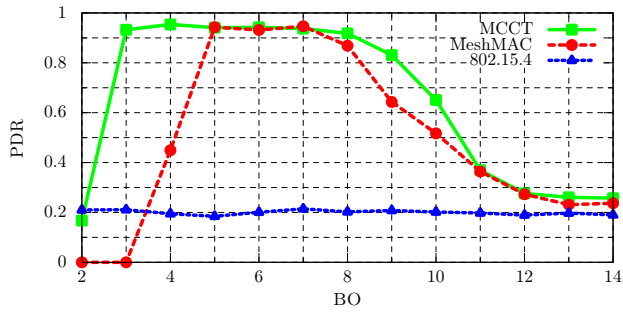
We have implemented the beacon-enabled mode of IEEE 802.15.4 with both the original superframe precedence and greedy superframe scheduling of MeshMAC [4] as well as MCCT in WSNNet [16]. WSNNet is an efficient event-driven simulator dedicated to Wireless Sensor Networks. It is modular and extensible, and was extensively evaluated [17].

We have considered an optimistic scenario for MeshMAC in which a node knows *a priori* its 2-neighbors when scheduling superframes.

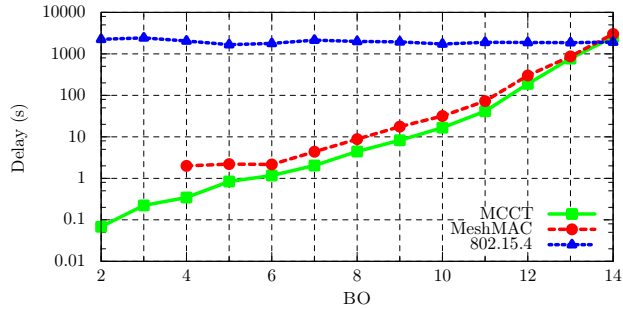
We have simulated a topology with random placement of 60 nodes in a disk and average degree (number of neighbors) of 9. Nodes generate constant low intensity convergecast traffic. Table I summarizes the main simulation parameters.

Fig. 5 shows the results. MCCT significantly improves the PDR for a large range of BO values. For BO = 4, we must schedule at most 8 non overlapping superframes ( $2^{BO-SO} = 2^{4-1}$ ). Frequency diversity improves efficiency, because nodes schedule superframes without any collision using orthogonal channels. End-to-end delay is also reduced: less collisions imply less retransmissions (note that the figure presents delay in a logarithmic scale). The Jain index presents the fairness in terms of the throughput of each node arriving to the sink. The figure shows that MCCT benefits from improved fairness for low BO values.

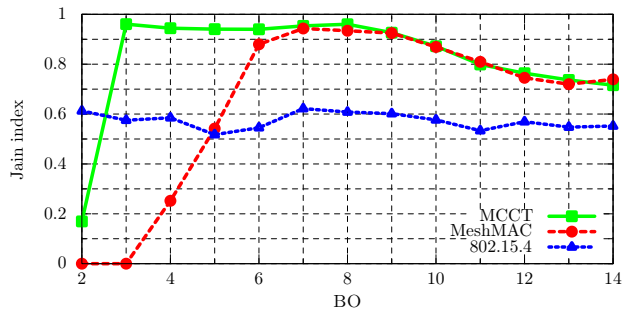
We have also simulated the impact of the number of neighbors on the Packet Delivery Ratio (cf. Fig. 6). We have fixed BO = 7 and SO = 2, because these values provide a duty-cycle of 3 % and 32 slots to schedule superframes. For small and medium densities, both protocols achieve a good delivery ratio: all the coordinators find a collision-free scheduling. However, the packet delivery ratio drops more quickly with MeshMAC: the number of slots is insufficient to schedule properly the superframes. On the contrary, MCCT uses 15 channels and 32 timeslots in each channel thus reducing the number of collisions. The packet delivery ratio remains between 97 and 77% even with 35 neighbors.



(a) Packet Delivery Ratio



(b) Delay in seconds



(c) Jain Index

Fig. 5. Performance in random circular topologies of 60 nodes

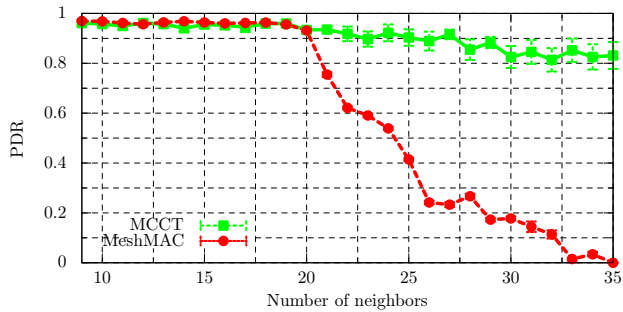


Fig. 6. Packet Delivery Ratio in function of the average number of neighbors

## VI. CONCLUSION AND PERSPECTIVES

We have proposed MCCT (Multi-Channel Cluster Tree), a cluster-tree construction protocol for nodes in beacon-enabled

mode with three main features: i) a neighbor discovery procedure that efficiently works in a multi-channel setup and keeps the IEEE 802.15.4 superframe structure, ii) a distributed localized protocol for constructing a cluster-tree that minimizes the number of coordinators and balances the tree, iii) an efficient channel and superframe slot selection method. Our simulation results show a significant improvement of packet delivery ratio and delay over the standard IEEE 802.15.4 and MeshMAC. We plan to evaluate the proposed scheme on an experimental testbed to validate its behavior in realistic conditions.

## ACKNOWLEDGEMENTS

This work was partially supported by the European Commission project CALIPSO under contract 288879, the French National Research Agency (ANR) projects ARESA2 under contract ANR-09-VERS-017, and IRIS ANR-11-INFR-016.

## REFERENCES

- [1] IEEE 802.15.4-2006 standard, <http://www.ieee802.org/15/pub/TG4.html>.
- [2] Zigbee-Alliance, "ZigBee specification", <http://www.zigbee.org>.
- [3] A. Koubâa, A. Cunha, M. Alves, and E. Tovar. TDBS: A Time Division Beacon Scheduling Mechanism for ZigBee Cluster-Tree Wireless Sensor Networks. *Real-Time Systems*, 40(3):321–354, 2008.
- [4] P.S. Muthukumar, R. de Paz, R. Špinar, and D. Pesch. MeshMAC: Enabling Mesh Networking over IEEE 802.15.4 through Distributed Beacon Scheduling. In *AdHocNets*, 2009.
- [5] H. Jeon and Y. Kim. BOP (Beacon-Only Period) and Beacon Scheduling for MEU (Mesh-Enabled USN) Devices. In *ICACT*, February 2007.
- [6] Berta Carballido Villaverde, Rodolfo De Paz Alberola, Susan Rea, and Dirk Pesch. Experimental Evaluation of Beacon Scheduling Mechanisms for Multihop IEEE 802.15.4 Wireless Sensor Networks. In *SENSORCOMM*, 2010.
- [7] Yafeng Wu, John A. Stankovic, Tian He, and Shan Lin. Realistic and Efficient Multi-Channel Communications in Wireless Sensor Networks. In *INFOCOM*, pages 1193–1201. IEEE, 2008.
- [8] G. Anastasi, M. Conti, and M. Di Francesco. The MAC Unreliability Problem in IEEE 802.15.4 Wireless Sensor Networks. In *MSWiM*. ACM, 2009.
- [9] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic, and T. F. Abdelzaher. MMSN: Multi-Frequency Media Access Control for Wireless Sensor Networks. In *INFOCOM*, april 2006.
- [10] K.R. Chowdhury, N. Nandiraju, D. Cavalcanti, and D.P. Agrawal. CMAC - A multi-channel energy efficient MAC for wireless sensor networks. In *WCNC*. IEEE, april 2006.
- [11] Xun Chen, Peng Han, Qiu-Sheng He, Shi liang Tu, and Zhang long Chen. A Multi-Channel MAC Protocol for Wireless Sensor Networks. In *CIT*. IEEE, 2006.
- [12] M.D. Jovanovic and G.L. Djordjevic. TFMAC: Multi-channel MAC Protocol for Wireless Sensor Networks. In *TELSIKS*. IEEE, sept. 2007.
- [13] Youngmin Kim, Hyejeong Shin, and Hojung Cha. Y-MAC: An Energy-Efficient Multi-channel MAC Protocol for Dense Wireless Sensor Networks. In *IPSN*, pages 53–63. IEEE/ACM, 2008.
- [14] Joris Borms, Kris Steenhaut, and Bart Lemmens. Low-Overhead Dynamic Multi-channel MAC for Wireless Sensor Networks. In *EWSN*, Coimbra, Portugal, 2010.
- [15] Ozlem Durmaz Incel, Lodewijk van Hoesel, Pierre Jansen, and Paul Havinga. MC-LMAC: A multi-channel MAC protocol for wireless sensor networks. *Ad Hoc Netw.*, 9(1):73–94, January 2011.
- [16] A. Fraboulet, G. Chelius, and E. Fleury. Worldsens: Development and Prototyping Tools for Application Specific Wireless Sensors Networks. In *Proceedings of IPSN, SPOTS track*, 2007.
- [17] Elyes Ben Hamida, Guillaume Chelius, and Jean-Marie Gorce. On the Complexity of an Accurate and Precise Performance Evaluation of Wireless Networks using Simulations. In *MSWiM*, Vancouver, British Columbia, Canada, October 2008. ACM.