



Beacon Synchronization for GTS Collision Avoidance in an IEEE 802.15.4 Meshed Network

Jackson Francomme, Gilles Mercier, Thierry Val

► To cite this version:

Jackson Francomme, Gilles Mercier, Thierry Val. Beacon Synchronization for GTS Collision Avoidance in an IEEE 802.15.4 Meshed Network. 7th IFAC International Conference on Field-buses & Networks in Industrial & Embedded Systemes, Nov 2007, Toulouse, France. 8 p., 2007. <hal-00287108>

HAL Id: hal-00287108

<https://hal.archives-ouvertes.fr/hal-00287108>

Submitted on 11 Jun 2008

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.

BEACON SYNCHRONIZATION FOR GTS COLLISION AVOIDANCE IN AN IEEE802.15.4 MESHED NETWORK

Jackson Francomme, Gilles Mercier, Thierry Val

*Toulouse University - LATTIS EA4155 Research Lab.
1 pl. G. Brassens-BP 60073-31703 Blagnac cedex-France
{francomme; gilles.mercier; val}@iut-blagnac.fr*

Abstract: Industrial process control architectures are generally composed of nodes organized in a cluster-tree. Today, wired communications between nodes enable guaranteeing the constraint respect attached to determinism. Innovations in wireless technology allow using these new technologies instead of wired systems. IEEE 802.15.4 standard meet industrial local network needs, but it does not propose any mechanisms to avoid beacon and GTS (*Guaranteed Time Slot*) collisions in meshed network. This communication proposes a new synchronization method for beacons and GTSs in meshed networks using IEEE 802.15.4.

Keywords: IEEE802.15.4, WSN, Industrial Networks, Synchronization, Collision

1. INTRODUCTION

Communication architectures in industrial local networks are still cabled today (Francomme *et al.*, 2006b). A new generation of wireless local area network allows replacing cabled communication links with wireless ones. As far as industrial process control needs guarantees, especially in terms of message transmission delays, wireless medium access control proposes mechanisms which avoid main drawbacks, i.e. frame collision, excessive consumption. Complex wireless architectures are not yet fully adapted to some industrial requirements; that is particularly the case with wireless communications with spread networks applied to meshed architectures, in which message transmissions from one end of the network to the other can only be made by using a medium access control protocol based on contention, to avoid collisions, i.e. CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*). This does not allow a 100% guarantee of the transmission delay through the entire network, but statistically at least it bounds it. This external message flow

considered in a cluster must not call into question communications allowed in this cluster, which use a medium access mode without contention, i.e. in CFP (*Contention Free Period*); the IEEE 802.15.4, base of the ZigBee standard (ZigBee, 2006), does not bring any of the guarantees previously mentioned. Synchronization between coordinators is a crucial problem (Branislav, 2006) which can carry the different parts of the networks to use their whole functionalities, avoiding the major problems of frame collisions. Although the routing aspect of the data packets can justify a part of this study on the synchronization, it will not be developed in this article. Paragraph 2 summarizes the essential part of the IEEE 802.15.4 standard MAC Layer (*Medium Access Control*), which is necessary to point out the problem of beacon collisions with nearby clusters, i.e. with partial recovery area. The beacon collision avoidance is presented in paragraph 3; we will point out the standard weak points and present our solution to avoid beacon and GTS collisions.

2. THE IEEE 802.15.4 STANDARD

The IEEE 802.15.4 standard is convenient with wireless communication constraints in industrial area as far as it envisages mechanisms for a star topology allowing:

- collision-free exclusive use of the medium, when data transfers need deterministic transmission.
- shared-use medium with the CSMA/CA algorithm for communications which do not require urgency.
- deactivation, i.e. sleep state, for device having no tasks to carry out, i.e. message to read or to transmit, in order to save its energy, particularly those supplied by battery.

This standard only includes PHY and MAC layers of the communication model (IEEE, 2003). PHY layer uses the ISM frequency band (*Industrial Scientific Medical*). 2.4 GHz frequency band is the most common; it offers 16 different channels with a 250 Kbits/s maximum rate on each channel.

Two communication modes can be used:

- Unslotted mode (without beacon) for which all communications use CSMA/CA algorithm; this proposes a contention mechanism for the medium access to avoid collisions. This mode is purely asynchronous.
- Slotted mode (with beacon) which insures synchronization of network nodes; the standard indicates that this synchronization only concerns nodes in coordinator radio range (IEEE, 2003); so it only concerns the star topology. For more complex topologies, there is no synchronization mechanisms implemented, as far as we know, excepted in (Van de Bossche, 2006).

Figure 1 presents the aspects of a superframe installed between two coordinator beacons; it shows three sections: the CAP section (*Contention Access Period*) where all communications respect the CSMA/CA algorithm for the medium access, the CFP section for the bandwidth to be reserved for specific uses, finally the inactive section when all the nodes of the network are in a sleeping mode in order to save some energy, i.e. batteries. The characteristics of these different sections depend on the parameters BO (*Beacon Order*) and SO (*Superframe Order*) given to the MAC layer (*Medium Access Control*), which will allow defining the superframe length, i.e. BI (*Beacon Interval*), then the active section SD (*Superframe Duration*) contained in BI; $BI = aBaseSuperframeDuration \cdot 2^{BO}$ [symbols] and $SD = aBaseSuperframeDuration \cdot 2^{SO}$ [symbols]. Each node will be active for $2^{-(BO-SO)}$ and asleep for $(1 - 2^{-(BO-SO)})$ with $0 \leq SO \leq BO \leq 14$; a weak duty cycle value (table 1) allows a relatively consequent energy saving (Francombe

et al., 2006a). An active section is divided into 16 equal length slots. The notion of slot is important for CFP communications; indeed, the coordinator will have the possibility to reserve an entire slot (GTS), even more, at the request of a node belonging to its star. In our context of sensor network, every node sends or receives a few data in every communication. Thus, we have evaluated the communication possibilities in one slot (table 2); it will more particularly concern the *slot 0* containing the coordinator beacon then the slots used for the GTS in CFP section.

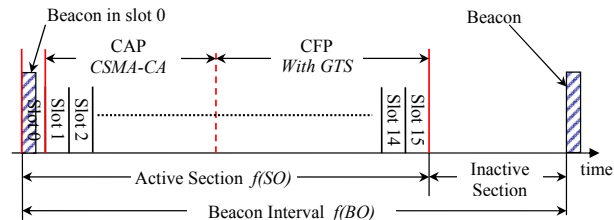


Fig. 1. Superframe time sections

Table 1. Duty-cycle evaluation (DC) according to BO and SO parameters.

BO-SO	0	1	2	3	4	5
	6	7	8	9	10	11
				12	13	14
DC(%)	100	50	25	12,5	6,25	3,12
	1,56	0,78	0,39	0,19	0,1	0,05
				0,025	0,012	0,006

We can notice that the set of nodes in a star topology communicates on a unique channel; thus, the bandwidth has to be shared between all their communications. A temporal multiplexing method is used for the medium access. Only CAP communications use the CSMA/CA algorithm, avoiding the simultaneous use of the medium by several nodes; the beacon and GTS transmission do not use any collision avoidance mechanism. Our contribution will consist in proposing as an addition to present solutions, a synchronization mechanism for all the superframe sections generated by the whole star coordinators in the PAN (*Personal Area Network*) (figure 2).

2.1 Network topologies for industrial process control

Two topologies can be considered:

- First, the cluster-tree topology which is a particular case of meshed network. Connectivity in this network is lower or limited to a certain number of connections; figure 2-a displays 4 clusters: CPAN cluster is not define to keep the figure simple; this topology corresponds to the tree structure obtained during node association to the network. In §2.2, we will present the address assignment mechanism which illustrates this tree structure.

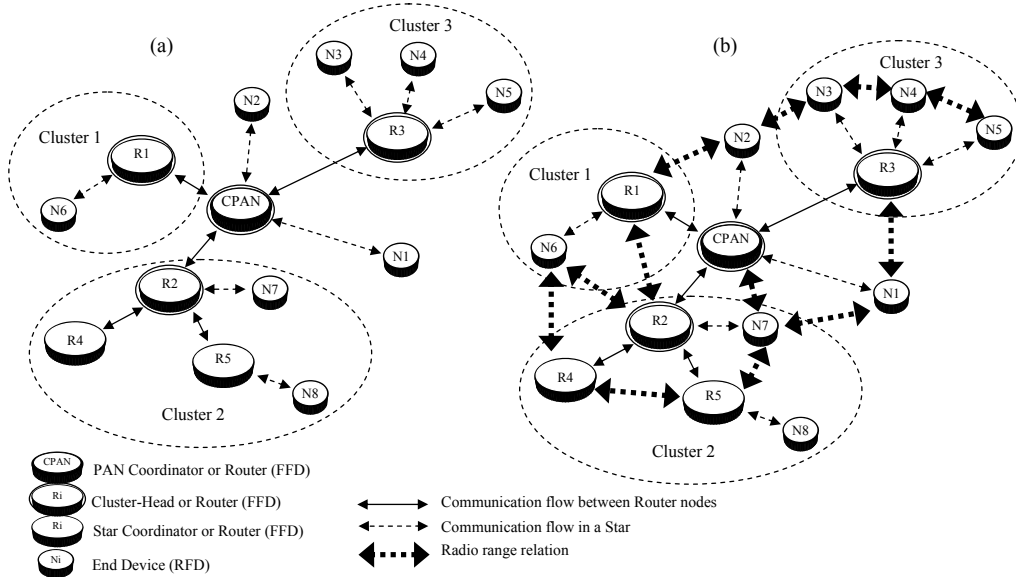


Fig. 2. (a) Cluster-tree topology - (b) Meshed topology with strong connectivity between routers

- Secondly, the meshed topology in which strong connectivity exists between routing nodes (figure 2-b). Additional links between routing nodes are led by routers which are not on the same branch of the tree but are in radio range; it does not present any drawback, as far as we have a larger number of message routing path choices through the entire network. We have recently submitted a paper on this subject, in which we present a new route search technique considering several parameters, i.e. energy consumption, bounded delay, bandwidth reservation.

All the radio links between nodes are considered symmetric, i.e. there is a mutual reception between all pairs of nodes as indicated in figure 2.

2.2 Address assignment mechanism

The network tree structure is obtained from the association of the different nodes to the network. Each node wanting to join the personal area network emits an "Association Request" to the closest coordinator; this coordinator can be a router different from the PAN coordinator if this coordinator is not in radio range. An address assignment mechanism is proposed by the ZigBee standard (IEEE, 2003): the principle is based on a hierarchical addressing; address assignment takes into account the maximum number of children per coordinator node (C_m), the maximum number of children routers (R_m) of a relative parent as well as the depth of the network tree (L_m); thus, given addresses represent the route. A parent router uses C_m , R_m and L_m to calculate a parameter C_{skip} used to compute the address-set size of its child, according to the depth d of the given node: if $R_m = 1$ then $C_{skip} = 1 + C_m \cdot (L_m - d - 1)$, else:

$$C_{skip} = \frac{1 + C_m - R_m - C_m \cdot R_m^{L_m - d - 1}}{1 - R_m}$$

So, if a parent node at a depth d has an address A_{parent} , the n^{th} child router node is assigned to the address $A_{parent} + (n-1) \cdot C_{skip}(d) + 1$. Also, the n^{th} child terminal node is assigned to the address $A_{parent} + R_m \cdot C_{skip}(d) + n$; d range is from 1 to L_m .

In figure 2-a topology each parent is allowed 5 descendants ($C_m=5$), however only 3 of these can be routers ($R_m=3$) with an $L_m = 3$ depth, thus 66 different addresses are required. It is not difficult to check that among these 66 available addresses, only 14 are used. A research perspective would be to find a solution to avoid this address wasting when wide network branches do not have the same length .

This network is established by a unique PAN coordinator (*Dedicated Device*) of FFD type (*Full Function Device*), whose role is to create the network, but also to reply to specific association requests, among others; it also allows synchronizing all communications. Furthermore, this network is composed of various nodes of FFD type; among those some of them have a cluster coordinating function, i.e. cluster-head. Inside this cluster, we can find another type which coordinates nodes in a star topology, i.e. the star coordinator. The cluster terminology is more particularly linked to the application and not to the topology: it shows a set of nodes making a local application and includes several routing and end-devices. Thus, we will name star coordinator, a node authorizing other nodes to join the network. RFD (*Reduced Function Device*) type nodes, called end-devices, do not have any routing function.

A network of this type authorizes many communication flows:

- Internal, i.e. inside a star topology, in which the coordinator synchronizes communications by sending regular beacons. In the IEEE 802.15.4 standard, communications can be made in the CAP section by using the CSMA/CA algorithm or in CFP by using time slots reserved by the star coordinator.
- External, i.e. coming from outside of the star topology, where communications can only be made in CAP; the standard shows that it is not possible to reserve any bandwidth at the level of the entire network since beacon transmission is not allowed.

2.3 Content of the IEEE 802.15.4 frames

Standard IEEE 802.15.4 proposes 4 frame types: (1) beacon frame (2) data frame (3) control frame (4) acknowledgment frame. The different frame contents are not developed in this paper; details can be found in specification (IEEE, 2003). We will only say that in the sensors' network case that we can find in the industrial process control framework, frames exchanged between nodes are limited to a small size; the largest of these frames generally is the data frame (or beacon frame). In the worst case there are $(17+n)$ bytes (IEEE, 2003), n for the payload length in bytes. The longest physical frame will contain 133 bytes or 1064 bits, with a duration of 4.256 ms @ 250 Kbps.

3. BEACON COLLISION AVOIDANCE

3.1 Cluster synchronization and coexistence

Each coordinator providing the management of its satellite nodes (*children*) must guarantee a defect-free synchronization as well as respecting allocated time sections, i.e. GTS; this is no more possible when two star coordinators are in a close neighborhood. Indeed, each of them sends its beacons and provides the GTS allocation within its star, but nothing guarantees that communications will be made at different times, i.e. nearby star transmissions can be mutually disrupted, creating collisions. Thus, it will be necessary to have an exclusion mechanism for the medium access, guaranteeing that multiplexed communications which do not have any contention mechanism (beacon and GTS), produce collisions. Superframe temporal multiplexing and beacon-only-period propositions submitted to the IEEE 802.15 TG4b (IEEE, 2007) (Shao *et al.*, 2004) (Lee *et al.*, 2004) do not provide the answer to our problem because routing nodes must be inevitably active at the same time to be able to participate to the message routing (Francomme *et al.*, 2006b). Consequently, every router node should be in its CAP section to

be able to receive frames from the emitter which is outside its star association, and forward them in its neighborhood if the message is not for it or for one of its children. However, nearby communications should not disrupt the beacon transmission nor the communications in CFP section. The first problem has been well identified by the Workgroup TG4b (IEEE, 2007) and, an algorithm allowing direct and indirect collision avoidance has been suggested (Koubaa *et al.*, 2007). A direct collision occurs when two nodes (N1 and N7 on figure 2), in radio range, simultaneously emit data on the wireless medium; an indirect collision occurs when one node (N6 for example) simultaneously receives communications from two of its close neighbors which are not in radio range (R1 and R4). Beacon synchronization method is based on beacon organization in a *beacon-only-period*, at the beginning of the superframe; this *beacon-only-period* is composed of numerous slots called CFTS (*Contention-Free Time Slot*) reserved by the PAN coordinator for every star coordinator, i.e. supervisor (figure 3). The first difficulty lies in one-hop neighborhood detection for direct collisions, but also the two-hop neighborhood detection for indirect collisions. The second difficulty lies in the CFTS organization to keep hierarchical synchronization, i.e. a child coordinator will not be able to use a CFTS which comes before the one its parent uses to inform it about the position of the CFTS reserved by the supervisor.

3.2 Our proposal

We consider the fact that the mesh topology does not allow the IEEE 802.15.4 the use of the beacon mode. Therefore, it is excluded to guarantee a transmission delay between an unspecified emitter and receiver, whatever topology is used. A beacon organization mechanism has been proposed in (Koubaa *et al.*, 2007) for a mesh network, avoiding collisions between coordinator beacons in nearby stars. As far as we know, no method has been proposed to avoid collisions within CFP section for the nearby coordinators. Our method completes the one proposed in (Koubaa *et al.*, 2007) with one rule which allows to check whether the PAN supervisor location provides an efficient synchronization for the numerous clusters, i.e. synchronization is as fast as possible to reach the farthest router node with a minimum of hops; also, we propose an algorithm allowing to organize the GTS in CFP sections of nearby star coordinators, and avoiding simultaneous transmissions on the medium for communications not using an arbitration mechanism for the medium access.

A second observation can be made on the use rate in some slots of the superframe; it is especially the case of *slot 0* used by the coordinator beacon

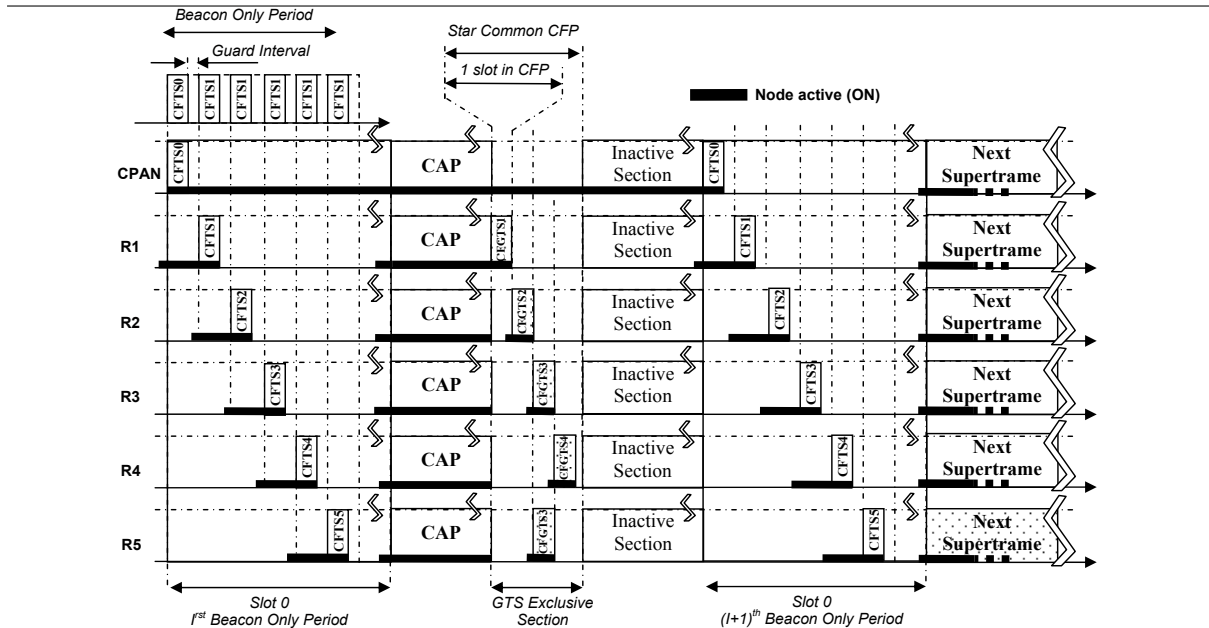


Fig. 3. Beacon and GTS synchronization in a mesh network

transmission. Indeed, the standard does not envisage the transmission of more than one beacon in *slot 0*; this frame with a relatively weak size uses a weak part of this first slot in the superframe; thus, we propose to use the remainder of *slot 0* for the beacon synchronization mechanism. This mechanism will not modify the current data rate because it uses a free part of the superframe. Table 2 gives a precise idea of the communication capacity for every slot, mainly for *slot 0*, and for the slots in the CFP section, in which there are theoretically no collisions. For a $SO \geq 5$ (for example), we can notice that *slot 0* (for example) has enough space to contain several beacons associated to their guard interval, i.e. the delay necessary to reach the reception state from the transmission one. So, our approach is justified. We can make the same observation for the slots used by GTS; data frames received or transmitted by a network node have a relatively weak size in regard to the duration of a GTS. Thus, we propose an exploitation of these periods both for the beacon synchronization and for an efficient GTS section use. To sum up, we are looking for an optimization of the medium in multiplexed time sections, in order to maximize the data rate without decreasing the standard reliability. With this method, we can notice that beacon synchronization only uses one slot for all network nodes; Interesting, isn't it?

T_{Slot} shows the duration of one slot in the superframe; this value depends on the superframe active section (SD). N_{TBO} shows the number of elementary backoff periods (T_{BO}) contained in one slot; all the communications in a slot are synchronized with the backoff period. $N_{Beacon-S0}$ is the possible number of beacon frames that *slot 0* can theoretically contain. In

Table 2. Slot communication capacity evaluation according to SO

SO	0	1	2	3	4
	5	6	7	8	9
	10	11	12	13	14
$T_{Slot}(ms)$	0,96	1,92	3,84	7,68	15,36
	30,72	61,44	122,9	245,7	491,5
	983	1966	3932	7864	15728
N_{TBO}	3	6	12	24	48
	96	192	384	768	1536
	3072	6144	12288	24576	49152
$N_{Beacon-S0}$	1	2	4	8	16
	32	64	128	256	512
	1024	2048	4096	8192	16384

the 2.4GHz frequency band, the data rate is equal to 62,5 Ksymbols, $T_{symbols} = 16\mu s$ and $aUnitBackoffPeriod(T_{BO}) = 20$ symbols. We can then deduce the slot duration: $aBaseSuperframeDuration = aBaseSlotDuration * aNumSuperframeSlots$ symbols with $aBaseSlotDuration = 60$ symbols and $aNumSuperframeSlots = 16$ (IEEE, 2003).

3.2.1. Collision-free time slot principle

The principle for collision-free time slots is described in a first research (Koubaa *et al.*, 2007); it consists in the determination of a maximum number of slots; a specific mechanism of the graph theory has been used, i.e. colored graphs (Kumar, 1992); this tool is widely used for the resolution of communication issues in wireless networks (Ramanathan, 1992). Each color obtained on the graph represents a CFTS placed in the *beacon-only-period*. We propose a more exhaustive exploitation of the mechanism used in this approach. This method depends on the neighborhood knowledge of each node in the network. When a node joins the network, it sends the exhaustive list of

neighbors it receives frames from, to the supervisor; they can be router nodes as well as end devices. It has been shown that direct and indirect collision avoidance requires a detailed knowledge in the network architecture (Koubaa *et al.*, 2007); that is why 2-hop neighborhood relations should be considered. Rules allowing *beacon-only-period* building is the following:

- **Rule 1:** Coordinator CFTS should be different from its neighbor CFTS, thus its parent's.
- **Rule 2:** Coordinator R_i CFTS must be different from the CFTS of its neighbor parent.
- **Rule 3:** given a CFTS set organized in an increasing order from index 0 to $n-1$, coordinator R_i CFTS index will not be higher than the one given to its parent.

Rules 1 and 2 are taken into account in the neighborhood matrix building. By applying the Welch and Powell algorithm (Ramanathan, 1992), these rules allow determining the number of slots required for the *beacon-only-period* to avoid collisions; in our example (figure 2), 5 CFTSs are required. In order to take into account rule 3 which corrects the synchronization defect for real-time applications, it is necessary to classify the coordinators according to their place in the hierarchy starting from the supervisor which distributes synchronization information. It seems that a more detailed architecture study would allow us to obtain a new principle to get an optimal (strategic) location for the network supervisor in order to decrease the delay necessary for the synchronization of all the nodes in the network. Thus, we propose a new algorithm to allocate the CFTS in figure 4:

-
- Building of the 2-hop neighbourhood table
 - Classifying router nodes according to their rank, i.e. their depth in the network: parents are at the top of the list.
 - For each rank, classifying router nodes according to their number of neighbors
 - Allocating the first CFTS to the first router in the list
 - For k ranging from 1 to m , m : greatest rank in the network,
 - o For i ranging from 1 to n , n : number of routers in the rank
 - For j ranging from $i+1$ to n
 - If routers i,j do not have common neighbor
 - o Then allocate the same CFTS to router i and j
 - o Exclude router j in the next tests - EndIf
 - Incrementing j to reach the next router - EndFor
 - Incrementing i to treat the next CFTS - EndFor
 - o Incrementing k to compute the next rank - EndFor
 - Transmitting the CFTS allocation table
-

Fig. 4. General algorithm for CFTSs allocation

When using once again certain graph theory elements, i.e. rising the 1-hop neighbor table at a power u , we can easily determine the number of possible routes to reach a receiver from an emitter. By scanning the neighborhood matrix rising to the power u , we can notice that for increasing values of u , only one line of the matrix contains

all its elements different from 0 ; then, we can reach all the nodes of the network from the location shown in the matrix; it is the best location for the supervisor. It is a first important point which enables preparing node dissemination allowing reaching every network node in a minimum number of jumps. If we continue to increase the matrix power, for each exponent, we obtain a new group (rank) of nodes reachable with a number of hops equal to the exponent u . They actually represent the hierarchy we are looking for. Thus, we can classify the different nodes, not only in regard to their degree, i.e. number of connection with neighbors, but also according to their rank in the network, i.e. the number of jumps separating them from the supervisor. With our example, we obtain two router classes: the first one where nodes are reachable in one hop from the supervisor, composed of R1, R2, R3, and the second one where nodes are reachable in two jumps, composed of R4 and R5. The new distribution gives a 6-CFTS set (figure 5). This trivial example described in figure 3 allows understanding the mechanism.

Groupe		CP	R1	R2	R3	R4	R5	N1	N2	N3	N4	N5	N6	N7	N8	N° CFTS	
-	CP	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	CFTS ₀
1	R1	1	0	1	1	1	1	1	1	1	0	0	1	1	0	0	CFTS ₁
1	R2	1	1	0	1	1	1	1	1	0	0	0	1	1	1	0	CFTS ₂
1	R3	1	1	1	0	0	0	1	1	1	1	0	1	0	1	0	CFTS ₃
2	R5	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	CFTS ₄
2	R4	1	1	1	0	0	1	0	0	0	0	0	1	1	1	1	CFTS ₅

Fig. 5. Coordinator 2-hop neighborhood matrix and CFTS hierarchical assignment

This approach validated through a simulation with a more complex network distribution, composed of 25 nodes uniformly distributed on a rectangular area; we obtain a 10-CFTS set for a non-hierarchical association, and 14 CFTSs for hierarchical consideration.

3.2.2. Collision-free GTS slots

Now, we will define the CFGTS notion (*Contention-Free GTS*) which allows the star coordinators to use a GTS in their CFP section with collision avoidance, i.e. no conflict with neighbor transmission. The set of CFGTSs will be reserved by the supervisor in a particular time section we will name *GTS-Only-Period*. The supervisor builds a square matrix representing the 1-hop neighborhood of all the network nodes from which we define the number of CFGTSs for all the star coordinators in the network.

It is important to notice that a GTS is used by the star coordinator and one of its descendant. So, the CFGTS reservation depends on two nodes: the parent P_i and the descendant D_i . The rule allowing to build the *GTS-Only-Period* in the common CFP is the following:

- **Rule 1:** The CFGTS of the couple of nodes P_i/D_i should be different from the neighbors' CFGTSs.

The hierarchical aspect is not significant because nodes emitting frames in CFP are already synchronized, from the beginning of the superframe with its parent beacon reception; GTS communications are completely asynchronous from a cluster to another one.

With a simple scenario applied to our example, in which 6 GTSs (out of 13 possibilities) are required, this method gives 4 CFGTSs (or 9 CFGTSs for 13 GTSs requests)(figure 6). The standard shows that up to 7 GTSs can be used within a superframe. If more than 7 CFGTSs are required, then it should be necessary to consider additional choice criteria like urgency or maximum of GTS served. An application of this approach is depicted in (figure 3) and gives a global solution for beacon and GTS collision avoidance. We can notice that in a star topology, the superframe emitted by a coordinator, contains GTS slots not assigned for internal communications to avoid collisions; we can point out that most of the time in a wireless sensor network, a sensor node is in a sleep mode to save energy, i.e. its battery. In every network star, our model seems to deprive nodes of a significant part of the superframe active section; it is not really a problem because the whole traffic in a star topology remains low, as it is waiting for a wireless sensor network.

	CP	R1	R2	R3	R5	N1	N3	N5	N6	N7	N8		
R5,R2	1	1	1	0	1	0	0	0	1	1	1	CFGTS1	1 _B
R2,N7	1	1	1	0	1	1	0	0	1	1	0	CFGTS2	2 _J
CP,N1	1	1	1	1	0	1	0	0	0	0	1	CFGTS3	3 _V
R3,N3	1	0	0	1	0	1	1	1	0	0	0	CFGTS1	1 _B
R1,N6	1	1	1	0	0	0	0	0	1	0	0	CFGTS4	4 _R
R5,N8	0	0	1	0	1	0	0	0	0	1	1	CFGTS3	3 _V

Fig. 6. Node 1-hop neighborhood matrix and CFGTS assignment

In order to allow the GTS reservation to a star coordinator on any slot of the CFP, it will be necessary to create a new entity which permits GTS shifting to the correct location in the CFP; we could create a *virtual GTS* (VGTS) for which it is not necessary to have a context reservation in memory; these VGTS are located on all slots reserved for the CFGTSs of other coordinators. The general algorithm for the CFGTSs is given in figure 7.

3.3 Limits in the CFTS and CFGTS solution

This new method allows avoiding beacon and GTS collisions between every cluster in the network. In order to enable a suitable synchronization, it is necessary for every superframe to have the same active section length (SD). Furthermore, as the

-
- Building the 1-hop neighbourhood table
 - Classifying couple of nodes P_i/D_i according to number of neighbours requiring a GTS
 - Assigning the first CFGTS to the first couple in the list
 - For i ranging from 1 to n , n : the number of couples
 - o For j ranging from $i+1$ to n
 - If couples i,j are not direct neighbour
 - Then assign to the couple j the same CFGTS as the couple i
 - Exclude the couple j in next tests - EndIf
 - Incrementing j to treat the next couple - EndFor
 - o Incrementing i to compute the next CFGTS - EndFor
 - Transmitting the CFGTS allocation table
-

Fig. 7. General algorithm for CFGTS allocation

CFP section of every coordinator is common, it requires a limited GTS density. The superframe active section only proposes 16 slots, the first being assigned to the beacon transmission, the second guaranteeing a minimum CAP section ($aMinCAPLength = 440$ symbols then $7,04$ ms @ 2.4 GHz). In the best cases, 14 slots remain for 7 CFGTSs, per superframe.

The synchronization is representative of the topology; the distribution of CFTSs and CFGTSs will be different if the topology changes. Thus, for each FFD node having a cluster-head function or being a star coordinator, it is necessary to systematically listen to beacons from the supervisor, i.e. with the same BI (beacon Interval), in order to propagate CFTS and CFGTS information towards the deepest nodes, then allowing collision-free working in the whole network, within *slot 0* and the common CFP section.

Thus, we can distinguish two activity levels: (1) one concerning the supervisor beacon tracking and general network synchronization with a superframe duration leading to synchronization; this duration is identical to all coordinators. (2) other concerning the star internal operation, in which the superframe duration can be larger than the one previously defined; however, they should be multiples. We find this information in every coordinator beacon.

3.4 Wide networks

When the network is composed of an important number of nodes distributed on a large area, the hop number required to reach a particular node may be important, and the GTS mechanism may not be efficient since the reservation request transmission and the response frame will take too much time. The cluster-head mentioned in §2.1 should then be given a specific function. Thus, it would obtain a supervision delegation, but only for the cluster that it controls. It could also be able to answer GTS reservation requests from its children. Moreover, it would have an automatic CFGTS reservation (there are so few CFGTSs

in a superframe) from the supervisor, then it would pass it on to one of its children; like the supervisor, it would analyze its cluster topology and would deduce a potential use of its unique GTS, a priori. Then, if two nodes respect the rules previously presented and providing they do not cause any collision, they can use the same CFGTS. Otherwise, only one of them can use it. If the star internal traffic requires a GTS composed of several slots, this cluster-head should send a demand to the supervisor in order to avoid conflicts with the CFGTSs of other nearby clusters. The implementation of this new technique requires a node identification, as described hereafter: (1)the PAN supervisor: CPAN (2)the cluster-head: R1 to R3 (3)the coordinator of a star association which can be a cluster-head: R1 to R5 (4)end-devices: N1 to N8.

4. CONCLUSION

This article deals with a new technique which allows synchronizing numerous clusters in a mesh network to avoid beacon collision and GTS collision, but also to ensure that the set of network coordinators are available in their CAP section to participate to the message routing. The CFP section is used for internal messages in the star topology, in which it is necessary to guarantee deadlines for addressee delay delivering. The CAP section is used for internal message transmission, in which we can accept an average transmission delay. For external message using the CAP section, we will guarantee a bounded delay in the case it is admitted within the star traffic. A perspective of this work will be to propose a mechanism allowing an optimal use of the GTS within a star association. Based on observations in §3.2, a GTS is distinctly under-used by a node which has a few data to transmit, in most cases. We will propose the exploitation of the remaining GTS by other nodes of the star association, which do not have any effective assigned GTS. For example, inside its beacon payload, the coordinator could send a GTS expectation list, which has not yet receive any confirmation. Once a node has finished using its GTS, it could allow the first node of the waiting list to take advantage of its remaining GTS, if it is absolutely certain that it can end all transactions before the end of this GTS. An illustration of the mechanism is given in figure 3. The GTS collision-free technique proposed remains valid as long as there is a weak star internal traffic and reasonable network size; the synchronization requiring that the *slot 0* of the superframe is common for all nodes in the network, as well as the CFP. The CFTS number can be relatively large and depends on the *slot 0* size, and also on the superframe active section. The number of CFGTS is limited to 7 and can include up to 14 slots; the GTS

requests in a star association must remain weak if we expect an important number of nodes to cohabit. Our example with 14 nodes has shown that the standard let us sufficient possibilities to use more than one CFGTS per star coordinator. Topology changes require an update of the CFTS and CFGTS distribution; thus, it is necessary to ensure a synchronization which follows these topology evolutions as well as possible. It requires choosing a synchronization period smaller than the delay between two topology variations. All communications being synchronized on the supervisor beacon reception, collisions can be avoided; beacon and GTS synchronization information is available in the beacon payload. In order to avoid an excessive waiting time for a mobile network node having changed of geographical location and of attachment point to the network, this synchronization update will be faster than these changes.

5. REFERENCES

- Branislav Kusy** (2006). Spatiotemporal Coordination In Wireless Sensor Networks.
- Freescale Semiconductors** (2006). User's Guide: 802.15.4 MAC PHY Software. *802154MPSUG V1.3*.
- IEEE Computer Society** (2003). Std 802.15.4 - 2003, part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs).
- IEEE 802.15** (2007). *WPANTM* Task Group 4b (TG4b). <http://www.ieee802.org/15>
- Francomme, J.**, G. Mercier and T. Val (2006a). Contrôle de Processus Industriels Sans Fil Embarqué à Economie d'Énergie et Contraintes Temporelles - Gestion de l'énergie avec le std. IEEE 802.15.4. *8ème Congrès EDSYS*. Albi(France).
- Francomme, J.**, G. Mercier and T. Val (2006b). A simple method for guaranteed deadline of periodic messages in 802.15.4 cluster cells for control automation applications. *11th IEEE International Conference on Emerging Technologies and Factory Automation*, pp. 270-277. Prague(Czech Republic).
- Koubaa, A.**, M. Alves, M. Attia, A. V. Nieuwenhuyse (2007). Collision-Free Beacon Scheduling Mechanisms for IEEE 802.15.4/Zigbee Cluster-Tree Wireless Sensor Networks. *7th International Workshop on Applications and Services in Wireless Networks*. Santander(Spain).
- Kumar, V.** (1992). Algorithms for Constraint-Satisfaction Problems: A Survey. *AI Magazine*, **Vol 13**, pp. 32-44.
- Lee, M.**, J. Zheng, Y. Liu, H.-R. Shao, H. Dai and J. Zhang, H. Jeon (2004). Combined Beacon Scheduling, Proposal to IEEE 802.15.4b.
- Van den Bossche, A.**, T. Val and E. Campo (2006). Proposition of a full deterministic medium access method for wireless network in a robotic application. *63rd IEEE Conference on Vehicular Technology*, pp. 279-283.
- Shao, H.-R.**, J. Zhang, H. Dai (2004). Enhancements to IEEE 802.15.4. *Proposal to IEEE 802.15.4b Task Group*.
- Ramanathan, S.** (1992). Scheduling algorithms for Multihop radio networks. PhD Thesis.
- ZigBee Alliance** (2006). ZigBee Specification from the web: <http://www.zigbee.org/>