A simple method for guaranteed deadline of periodic messages in 802.15.4 cluster cells for automation control applications

Jackson Francomme, Gilles Mercier, Thierry Val

A simple method for guaranteed deadline of periodic messages in 802.15.4 cluster cells for automation control applications

Jackson Francomme Gilles Mercier Thierry Val

ICARE EA3050 Research Laboratory BP 60073
1 place G. Brassens - 31703 BLAGNAC cedex - France
{ francomme; gilles.mercier; val@iut-blagnac.fr }

Abstract

We propose the implementation of a wireless sensor network applied to automation control applications, when the guarantee of a delay delivery for the complete reception of messages is necessary. The 802.15.4 *wireless standard network offers possibilities of management of the bandwidth. This paper presents the method to guarantee the deadline transmission of periodic messages. For external messages (crossing the network from clusters to its destination), this average latency is used as a parameter of the routing protocol decision balance between energy saving and delay transmission. In this last case, a QoS method all over the network has to be installed to maintain a bounded end to end transmission delay. This work is still in progress. A specific Matlab simulator has been developed, principles of this routing method are mentioned at the end of this paper.*

1. Introduction

Within the automation process control applications framework [1] [2] [3] [4], we mention the flexibility need rather than the true mobility. Wireless communication brings a fitting flexibility of manufacturing units and a connection facility. Thus, the sensors nodes[1] (sensors/actuators) [5] are attached in a permanent way to the same cell, actually clusters in the *Zigbee concept* [6] (Fig. 1).

The network organization is based on the cluster-tree structure protocol from the Zigbee[2] Alliance consortium developed at the *Zigbee network layer* [7]. Details on ZigBee architecture, cannot be fitted in this article, so we recommend you to refer on [8] [7] publications. A complete *PAN*[3] network consists of several cells (clusters) organized in star or bunch mode. The complete network coordination is assumed by a supervisor (Dedicated Device) and the communication continuity between clusters is assumed either by the coordinators of each cluster or the nodes belonging to both clusters (border node). The details of this network organization are exposed in [7] [8] [9].

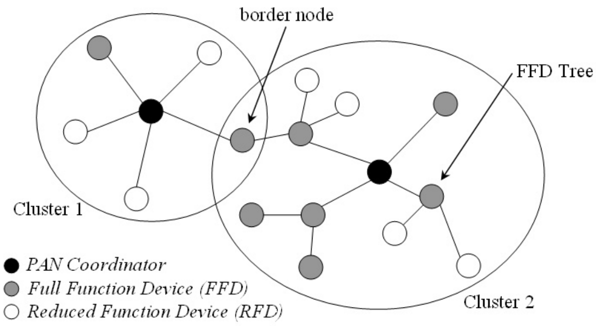

Figure 1. Star and Cluster-Tree (bunch) network organization

Our approach is based on a centralized node management for each cluster head (also called coordinator in the Zigbee Alliance terminology [9] [10]). The network coordinator will have multiple specific functions:

- Distribute temporal windows to ensure the deadline transmission of periodic messages.
- Sharing the bandwidth of the *CSMA-CA section* [10], between the nodes members of the cell and messages issued from external nodes of the cell for a destination not attached to this cluster. As the bandwidth for external traffic needs to be maintained as long as the route is established, a control of admission is necessary to accept or reject requests received to establish a route between a sender and its destination through the network, depending of the available bandwidth.

A concern which must remain present at all stages of this organization relates to the control, or at least the knowledge, of the energy consumption by the mobile nodes [5]. Thus, an optimal routing protocol will have

[1]*Sensor node*: Sensor and/or actuator usually attached to a RFD component (Reduced Function Device).

[2]The name *"ZigBee"* is derived from the erratic zigging patterns many bees make between flowers when collecting pollen. This is evocative of the invisible webs of connections existing in a fully wireless environment - wisegeek.com.

[3]*PAN*: Personal Area Network

to be developed. The cluster-head will have to provide the necessary parameters to the mobile node to choose the best path fitting either the delay transmission or the energy consumed to send a message, or both. Thus a specific metric including a weighting coefficient, will allow the most appropriate route among several parameters (*latency*, *delay*, *energy cost*). *Paragraph* 2 proposes a network organization within the sensors network framework for automation control applications. *Paragraph* 3 proposes a scheduling method based on the messages deadline guarantee, with a scenario as example. *Paragraph* 4 concludes on the arbitration principle, therefore with adjustment, of the bandwidth allocated to internal and external messages, particularly by considering the average waiting delay with no guaranteed deadline. The conclusion mentions a modification of the *standard Zigbee routing protocol AODV*[4] , but its development belongs to the overall work of the network concept. the AODV principle is presented in [11].

We have limited our approach to the $2.4GHz$ *ISM band*[5] use, which allows a maximum of 250kb/s rate.

A recent publication [12] offers an off-line scheduling analysis, similar to our approach, but differs in the sense that we have focused our concept on treating periodic messages applied to restricted applications: automation control process.

The data part of messages are issued from sensors and are only for traffic between the nodes of the same cell. The message length is bounded to 8 or 16 bytes.

Our scheduling method is a derivative from the *WorldFip allocation table method* [13], applied with restrictive limitations due to the 802.15.4 standard, and assumption that we may modify slightly the *MAC layer* software source, where the *GTS allocation* slots is done only once, and there is no *GTS request*.

Our contribution relates to the *GTS distribution*, therefore the *CSMA-CA* mechanisms are not explained. The publication [14] details the *CAP management*.

2. Network functionalities

2.1 Network organization

By the particularities of the automation process control applications, we retained the hierarchical architecture proposed by the **ZigBee Alliance**. The cluster management is assumed by the coordinator (cluster head), which is supposed not to be mobile, so that their topological implementation is known. We consider three different types of mobile nodes:

- *Type 1*: Nodes attached to this cluster will be able to obtain a *guaranteed message deadline* under particular condition. They are primarily data provided by sensors. Their messages are exclusively broadcasted inside the cluster to control a local application. Their production periodicity is known. The scheduling mechanism is explained later in §3.

- *Type 2*: Nodes attached to this cluster, but not requiring deadline guarantee, will have a minimal guaranteed bandwidth, therefore a known average latency, and managed by the coordinator. Both, nodes may move inside the cluster range, giving the requested flexibility to re-arrange the cell production organization; i.e.: machine tools production.

- *Type 3*: Mobile nodes moving from cluster to cluster inside the same *PAN*, are able to send messages to any members of the network. The cluster-head manages an *access list* of its *permanent members* and ensures an admission control of the external messages by establishing a negotiation protocol mainly based on latency of the message in transit. Thus, the mobile node will be able to choose its best adapted path to transmit a message to a destination. In a restricted use version of the ZigBee network, we propose in the next paragraph a method of messages transmission control according to the transmission guarantee requested or the acceptable average latency.

According to the fact that the producer/consumer(s) message couple is, either intended to be propagated and consumed inside the same cluster (*types 1* and *2*), or transmitted through a global network (*type 3*), via the coordinator node of each cluster to the terminal destination, the scheduling management of local and global messages will be located at two different levels from the network organization.

The architecture, adapted to our application, implies that the network backbone is made of a static part, of which one is mainly composed of the *PAN* network coordinators, and the other is moving part is mainly composed of *sensors/actuators nodes*. In regards to the nodes associated with the periodic messages, they can roam inside a cluster. All nodes inside a cluster are supposed to be heard by each other [7].

Assumptions :

1. Some nodes may belong simultaneously to *type 1* and *type 2*. Thus we will have to distinguish messages with or without guaranteed deadline, both for the destination inside the cluster.

2. Messages length of *type 1* mode, mainly issued from sensors, is supposed to be short as well as for others wired automation local networks (CAN, FIP, Profibus,...). For *types 2* and *3* nodes, there is no length limitation . In our feasibility automation study, we have made our simulations with a practical constant payload length of 8 or 16 bytes.

[4] AODV : Ad hoc On Demand Vector

[5] *ISM band*: The Industrial, Scientific, and Medical radio bands are defined by the ITU-R in 5.138 and 5.150 of the Radio Regulations.

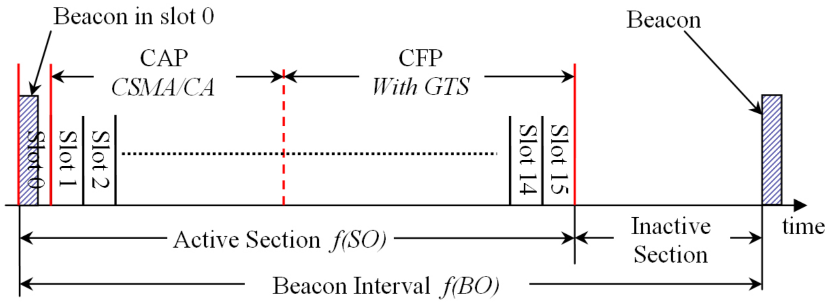

Figure 2. 802.15.4 superframe structure

2.2. Recall of the MAC 802.15.4 superframe

In a beaconless network (non slotted mode) [15] [8], arbitration access to the media is performed by a classic *CSMA-CA* (*Carrier Sense Multiple Access with Collision Avoidance mechanism*) [16]. Immediately after an *IFS* (*Inter Frame Space*) named T_{ack}, the destination acknowledges the reception without error. If not, the message will have to be retransmitted. This T_{ack} is facultative.

The superframe is divided in active and passive sections (inactive). In the passive section, the coordinator is in *"Sleep" mode*. The ratio active/passive section is defined by the *SO* (Section Order) and *BO* (Beacon Order) parameters, such as $0 \leq SO \leq BO \leq 14$. The non active section is useful when we must take care of the coordinator energy consumption [17], but will limit the bandwidth of the cluster.

The active section is also divided into 2 periods: *CFP*[6] and *CAP*[7] [15]. This section is divided into 16 slots (0 to 15); *slot 0* is reserved to the beacon.

We define the management organization of the messages inside a cluster, by acting on parameters of the *MAC frame* and beacon definition in layer 2 of the *OSI model*. For messages of *type 3*, the management is made at layer 3, by the routing protocol. This point is discussed in §4.

One must consider the need, for each coordinator to proceed to a local cluster scheduling concerning messages of *type 1*. This scheduling will have to ensure the deadline of transmitted periodic messages and optimize the power consumption of the RFD[8] nodes (sensors) [18] [19] [20].

3 Scheduling feasibility

Within the scheduling framework, we mostly use the term of *task*. We will assimilate the concept of *task* to that of the *message* by integrating the *payload*, the *header*, the *control bytes*, and if necessary, the *acknowledgement* (optional with *ZigBee*).

The goal of this scheduling consists of checking that the feasibility criteria are gathered in response to a given scenario of messages distribution by respecting each of their deadlines. A task M_i (or message) is defined by four parameters $Mi\,\{C_i, T_i, D_i, J_i\}$: respectively its *duration*, its *periodicity*, its *deadline* or *transmission latency* and finally the *jitter* J_i corresponding to the moment of the data production by the sensor node and the arrival of the beacon (Fig. 3).

The duration C_i corresponds to the higher bound of an integer slots number, such as: $1 \leq C_i \leq 15$. It includes the transmission time of the initial message and according to whether it is acknowledged or not, increased by a T_{ack} space and an acknowledge frame. D_i is referenced to the beacon slot 0. As messages have a length of an entire message (upper bound), C_i and D_i will be an entire multiple of slots. The jitter is the lapse of time between, the absolute time, where the sensor gets the information, and the beginning of the next beacon. Therefore the jitter J_i is an external element of the *PAN* and cannot be a part of the scheduling. The activation of the sensor can be subsequent compared to the position of the beacon. In any event, to be able to transmit the message, the sensor must be available before the allocated window, whose position is defined in the *GTS field* of the beacon. As we are in beacon-tracking mode, the simplest way would be to activate the sensor synchronously in order to minimize the jitter and transmit a data with a time of minimal ageing. The jitter is minimized when it is equal or less than the awaking time of the radio transmitter/receiver.

1st feasibility condition:

We specify the development principle of the examination table which supports scheduling in the coordinator node.

The examination table contains the list of all periodic messages identifiers $M_i\ \{C_i, T_i, D_i, a_i\}$ constituted by a control scenario of processes known beforehand.

By considering a periodic traffic flow constituted of p messages of T_1, T_2,..., T_p periodicities, we will indicate two quantities *BI* and T_{MC}, respectively: the *Beacon Interval time* and the *macro-cycle period*, by analogy to the examination mechanism implemented in a *WorldFIP network* (micro and macro-cycle.) [21]. The values *BI* and

[6] CFP : Contention Free Period

[7] CAP :Contention Access Period

[8] RFD: Reduced Function Device . Typically sensors with a reduced stack protocol and limited application

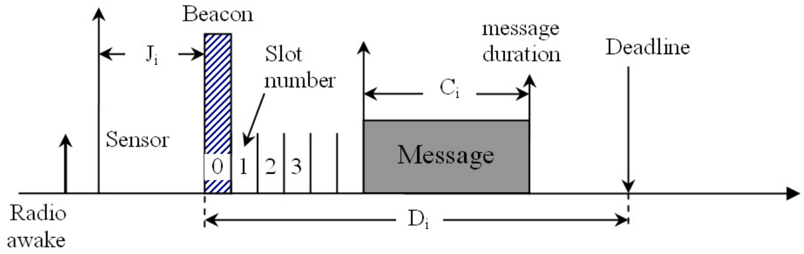

Figure 3. Message model within an 802.15.4 superframe

T_{MC} are defined in the following way:

$$With \quad \{1 \leq i \leq p\} \quad BI = GCD \quad T_i,^9 \qquad (1)$$

$$T_{MC} = LCM \quad T_i \quad (cond.1) \qquad (2)$$

The update of the *GTS windows* allocation parameters is made periodically before the emission of a new beacon.

The *GTS allocation* is performed by a specific scheduling method considering the deadline of each message for every Beacon Interval. We used an *EDF algorithm* to establish a *GTS allocation table*. In the 802.15.4 standard, a node wishing a *GTS* send a "*GTS-request*" in *CAP section* to the cluster head and waits for a *"GTS-confirm" response*, then up to 16 consecutive *GTS slots* are automatically allocated following the beacon after the "*GTS-confirm*". When the node has sent its message, it sends a "*GTS-deallocation*" to stop the remaining *GTS slot reservation*.

In our solution, an allocation table is established *off-line*, also there is no need to make a "*GTS-request*". The *GTS slot* is announced in the beacon.

This functionality diverges of the 802.15.4 as the *GTS* is only allocated once, instead of 15 consecutive superframe in the standard.

This feasibility condition, imposes a relation between the T_i periods, the BI, and for its derivatives parameters defined in the 802.15.4 *MAC* sublayer. That is :

- number of slots in a superframe = 16, $BI = aSlotBaseDuration * aNumSuperframeSlots * 2^{B0}$ symbols.

- $macSuperframeOrder$, SO, and the superframe duration SD, are related as follows: for $0 \leq SO \leq BO \leq 14$, $SD = aBaseSuperframeDuration * 2^{SO}$ symbols.

2nd feasibility condition:

The *802.15.4 standard* introduces three main limits to be considered in the feasibility analysis:

1. The number of slots of a superframe is limited to (15 +1) including the beacon.

2. There is a maximum of 7 allocations of simultaneous *GTS windows* in the same *superframe*, for a nominal cluster capacity of *255 nodes*.

3. The 802.15.4 standard defines a minimum *CAP section* of *440 symbols*, that is for a *250 Kb/s*, roughly *3.5 ms*. Also in our *EDF* feasibility study we have introduce a virtual task having a periodicity equals to the *BI period*, with a capacity and a deadline of *3.5 ms*.

$$\sum C_i\{1 \leq i \leq k\} \leq 15slots, \quad (cond.2) \qquad (3)$$

$$k \leq 7 \; is \; respected$$

k is the number of messages in one specific *Beacon Interval*

3.1 Messages with identical periods

The case of messages with equal periods raises the difficulty only in the case where the **condition 2** cannot be respected.

In that case, it is possible to distribute the periodic load by spreading the messages with identical period on several consecutive superframes, which supposes a synchronization of sensors activation with the beacons.

3.2 Punctual non respect of deadlines

The distribution of the periodic messages workload, as explained above, gives a flexibility, which is extremely useful on the distributed placement of the messages within a macro-cycle, when we arrive at a deadlock situation in the *GTS windows* organization.

As the *GTS slots* are positioned starting from the end of the *Active Section*, it may sometimes appear that a message cannot respect its deadline specifically when there are few *GTS windows* inside a *CFP session* (Fig. 4).

As the parameter BI is specified within each beacon, if a punctual missed deadline appears in the following beacon interval, it is then possible to modify the value of the next active *beacon interval*.

This capacity to redefine the value of BI is an element of the *802.15.4 standard* flexibility. But this method requires the reloading of the individual timers of every node of the cluster; this is a sophisticated solution. We will see

[9] GCD: Greatest Common Divisor and LCM: Least Common Multiple.

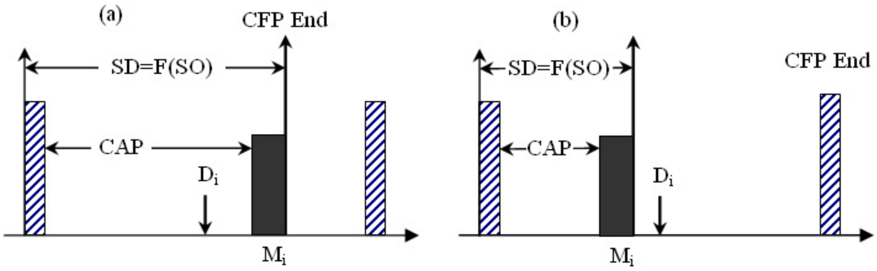

Figure 4. Missed Deadline (a) / Respected deadline (b)

later that an optimal ratio between the *CAP* and *CFP periods* within a macro-cycle is a more suitable solution.

3.3 Feasibility scheduling global test

We have mentioned that the *802.15.4 standard* offers flexibility for positioning the *GTS windows*, like dimensioning other various periods BI, SD, CAP and CFP.

On the basis of a given scenario and by observing the two conditions expressed in §3.1, we can establish in a dichotomist way (or with an adapted program) an aid for periodic messages placement.

Before starting such a step, it is appropriate to elaborate a global feasibility scheduling test, to ensure that all the deadlines on the whole macro-cycle are guaranteed.

If p represents the whole of the periodic messages having to be emitted within a *macro-cycle* T_{MC}, with an interval BI between beacons, the number of beacons inside a macro-cycle is equal to (Equ. 4):

$$N_m = \frac{T_{MC}}{BI} \tag{4}$$

The situation in the worst case of the periodic traffic will be met when the temporal space will be completely occupied by the transmission of the periodic messages; i.e. when the *CFP period* will be equal to BI and does not include any *CAP section*. More precisely, no traffic with contention or node activity will be possible.

The total time available in a *Beacon Interval BI* is equivalent to 15 slots out of the 16 of the interval. The total time for the transmission $T_e max$ in a macro-cycle is thus equal to (Equ. 7):

$$T_e max = \frac{15}{16}.T_{MC} \tag{5}$$

Thus the global condition of feasibility of the scheduled periodic traffic is expressed by (Equ. 6):

$$\delta = 1 - \frac{1}{T_e max}.\sum_{i=1}^{p} C_i \geq 0 \tag{6}$$

3.4 Deadline adjustment parameter

BI and T_{MC} are forced by relation [Equ. 2 (cond.1)]. In the previous relations the deadline is not taken in consideration. The value of δ characterizes the margin to assure the deadline objective. Also, if inside *Beacon Interval* one or more messages do not verify the deadline, one solution is to temporarily reduce the BI value, but we have seen in (§3.4) that this possibility is not suitable for more than one message. The second solution consists in reducing the Duty Cycle $DC = 2^{SO}/2^{BO}$ (Fig. 2); this leads to open the *CFP period* previously in the BI, but with the consequence of reducing the *CAP period*. In this last solution, DC is supposed to be fixed for the complete macro-cycle.

An other simple solution is to place a *stuffing message* just after the message which does not respect its deadline. This method has the same effect that the previous solution and presents the advantage when one does not wish to modify the *Beacon Interval* parameters, but reduces temporarily the *CAP section*.

The different methods are illustrated in the following section.

3.5 Results

The scheduling feasibility is made *off-line*. We have built a series of programs from the relations [Equ.1] to [Equ.6]. To limit the diversity of the possible situations, we have considered practical cases. Within the framework of sensors data exchange, the size of the data is limited; we took into account only two maximum messages sizes, for the *CFP section*: useful 8 and 16 bytes. The other messages are transmitted in the *CAP section*.

An energy consumption study has been published [14]. Nevertheless we can intuitively understand that the energy consumption is minimized when the message length is equivalent to a multiple of a slot time. The messages duration being respectively $T_8 = 0,832\ ms$ and $T_{16} = 1,088 ms$, for useful data of 8 and 16 bytes. Therefore, possible slots time are $\{2T, 1T, 1/2T, 1/4T\}$.

We have started from a diversity of scenarios and have checked the feasibility of periodic tasks scheduling; i.e. to find the range of DC for which the deadlines are guaranteed. We have considered the time position of the end of the message (or the start point of the following message, or the end of *CFP section* when this message is the last to be sent). An example of scenario on 5 periodic messages of 8 or 16 bytes size with different periodicity illustrates the scheduling methodology [Table. 1]. The periodicity and deadlines are expressed in *milliseconds*.

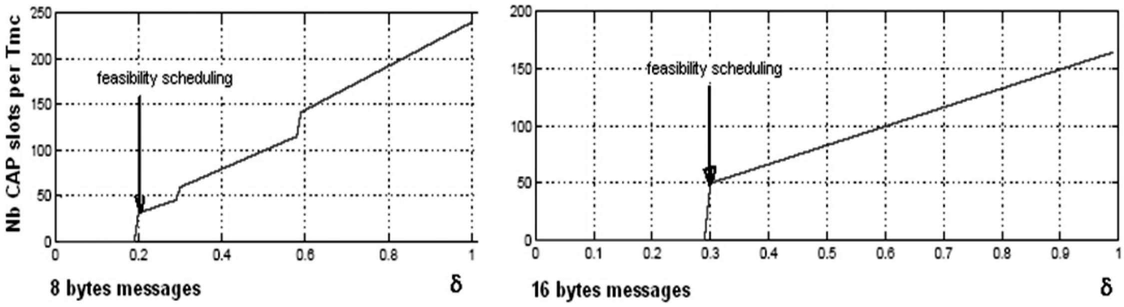

Figure 5. Feasibility analysis by graphs with no deadline consideration

Message number	1	2	3	4	5
Periodicity	15	30	60	75	150
Deadline	11	11	9	15	7

Table 1. Five messages of length *1 ms* with their deadline and periodicity

Graphs of the Fig. 5 are given for this scenario with a maximum length of 8 or 16 bytes, without deadline consideration. The feasibility scheduling threshold appears with $DC = 0,2$ for 8 bytes messages and with $DC = 0,3$ for 16 bytes messages (x-axis). These feasibility thresholds (computed from conditions 1 and 2), represent the minimum value of the micro-cycle and macro-cycle, to ensure the positioning of the *GTS slots* within all the *CFP section* included in a macro-cycle.

The y-axis value represents the number of slots allocated to the *CAP period* inside a macro-cycle (T_{MC}). The temporal positioning of the *GTS windows* is given for each *Beacon Interval*; within what we call a *micro-cycle*. If N_m is the number of *BI* inside a *macro-cycle* [Equ. 4], we can deduce the average value of the *CAP period.* This value is an essential element to determine the available bandwidth of the *CSMA-CA* communication, and therefore, the average latency of the messages transmitted or received in the section with contention, as well for the local data of the cluster and for the data in transit from cluster to cluster.

If we introduce the deadline now, a new set of programs scanning *DC* values missed deadlines appears.The first scheduling computation results consider the worst-case solution where all messages are activated in the same BI. The feasibility test for a given DC gives until 3 different messages missing their deadline. The solution to reduce this DC is to enlarge the CFP section. Another solution consists to spread the messages activation (if possible in the process)within the macro-cycle. These results are shown in Fig. 6 where an arbitration table of messages temporal positioning in the y-axis, with the slots position in a *Beacon Interval*. The *slot 0* is the *beacon*. In the x-axis, we have the messages periodicity distribution in the macro-cycle. In this example, $BI = 15\ ms$, $T_{MC} = N_m.BI$ ($N_m = 20$). The example has been established with a $DC = 11/15$. The scheduling used the *EDF algorithm* to position the messages in the *Beacon Interval*. The time positions of the deadline are placed on the left of the graph where we can see that on $BI = 11$ (165 ms), the message M5 does not respect its deadline, as shown in Fig. 6.

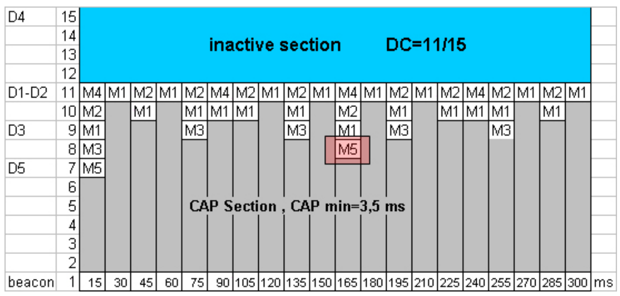

Figure 6. Arbitration table scheduling with DC = 0.73

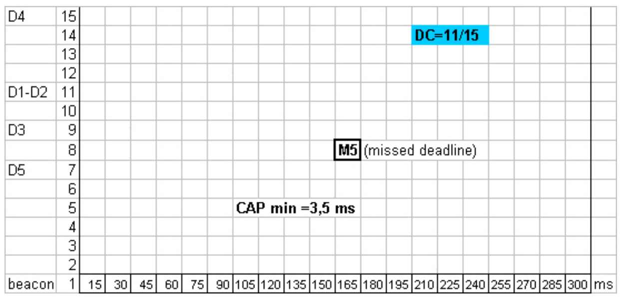

Figure 7. Sort of the missed deadline on a given scenario

We have developed others sets of programs to sort, missed deadlines message for a given *DC ratio*. Figure 7 presents the unique message $Nb5$ on *slot 11* missing its deadline.

We explained that there were 2 ways to solve the problem of the missing deadline. The first one is to insert a *stuffing message* after the missed message, without modifying the *Duty Cycle DC*, the second one consists to find the most appropriate DC (i.e. with the largest *CAP section*) where all the deadline are guaranteed. Figure 8-a shows the first solution with a maintain $DC = 11/15$, figure 8-b exposes a new arbitration table for the second

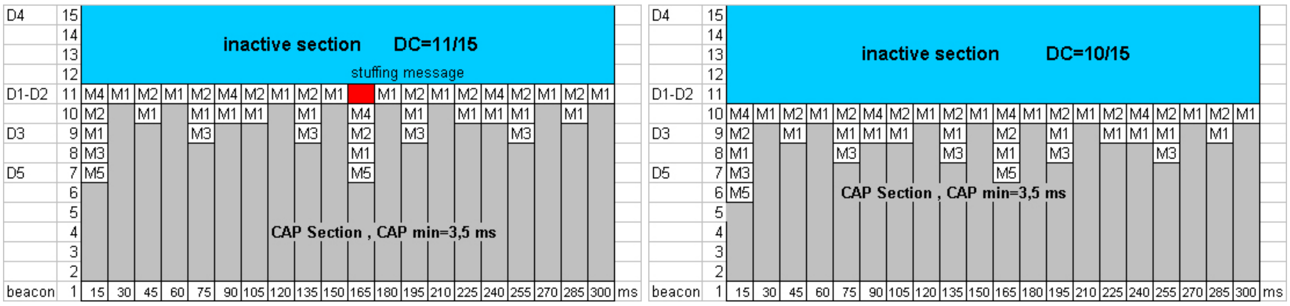

Figure 8. Insertion of a stuffing message (a) / Modifying the DC to respect all deadlines (b)

solution with a $DC = 10/15$.

4 Conclusion and work in progress

This paper has detailed a scheduling mechanism to place the messages to be sent inside the same cluster with deadlines transmission constraints. When their periodicity and deadline are known, a table is built to allow the coordinator to distribute the necessary *GTS* in the *CFP section*. The *GTS time position* and the concerned node are mentioned in every beacon. An *EDF algorithm* is performed *off-line* to ensure the respect of all the deadlines, in the *beacon interval*.

This subject is a part of an overall work, and does not concern the other messages (periodic without deadline guarantee and sporadic).We have extended our approach for the other messages, in order to manage the *collision section* (*CAP*) in terms of latency and consumption.

A recent work has established a method to compute T_α as the average waiting transmission delay in the *CAP section* [14].

$$T_\alpha = T_m + R * T_{CCA} + x \quad (7)$$

Where T_m is the message length, R is the *average number of back-off* in a network with *n nodes*, T_{CCA} is the *clear channel access period* and x is a function of parameters R and *back-off period*. Schartz [22] proposed an efficient iterative algorithm that we have implemented, in our simulation, to obtain the *x* value.

We have developed canonical relations to establish the different costs of energy consumption in function of the different states of a radio transmitter (tracking mode, transmitting or receiving in *CAP* and *CFP section*, etc). Theses costs, associated with the average *waiting delay* allow a sensor node to manage its own *local battery life time*. A different method published by Gang Lu & al [23] gave similar results with a *NS2 simulation*.

Moreover, the cluster head knows the average *CAP section* within a macro-cycle. Then, it is able to reserve a bandwidth for the local messages (*type 2*). On a negotiated session, when receiving a request to establish a route (for *type 3* messages), it can either reject the request or propose a latency delay for this route. If the proposed route is confirmed, the cluster head is charged to maintain this latency as long as the route is valid. We are currently evaluating a modification of the current *AODV routing protocol* in introducing a *multi routes choice* among different parameters: $latency$, $energy$, $distance$ or a mixed balance of these parameters. AODV is the standard routing protocol implemented in a Zigbee infrastructure. This method is simple to implement, but not optimal, in the sense that the retained route is only based on a minimum "instantaneous" propagation delay from the source to the destination. Therefore, this delay is not guaranteed during the route validity and may deeply increase, due to the saturation of one or more cluster-head. Also, the energy consumption is not considering.

Our solution is based on the AODV principle, but is taking into account *multi routes possibilities* between a minimum and a maximum propagation delay, including an energy consideration. We are proposing a multicriteria metric based on energy and propagation delay to class the different possibles routes. A simulator has been developped under Matlab.

The figure 9 shows the simulation results of a routing algorithm between two mobile nodes. The network is a *square of 300 x 300 m* size. *Mobile 1* is attached to the *cluster 3*, and *mobile 2* is attached to *cluster 19*. The figure 9-a represents the mesh with all the possible routes between these 2 mobiles. The figure 9-b, displays the 2 most efficient routes considering only the energy parameters (the bolded line is the first route, the other line is the second order choice). Also, if the first route fails for any reason, it is more suitable to take the second route than to perform a new routing request. A mixed balanced metric has been used , and performs a weighting between delay or energy saving. All these concepts are to be published shortly.

References

[1] J.P. Thomesse (LORIA, Nancy), "Les réseaux de terrain : des réseaux temps réel", *Ecole d'Eté Temps Réel ETR'2005 - Nancy*, vol. ISBN: 2 905267-47-X, pp. 261–268, Septembre 2005.

[2] A. Willig, K. Matheus, and A. Wolisz, "Wireless Technology in Industrial Networks", *Proceedings of the IEEE*, vol. Vol. 93, no. No. 6, pp. 1130–1151, June 2005.

[3] J. Heidemann and R. Govindan, "Embedded Sensor Networks", *USC - Information and computer Sciences Institute*, 2004.

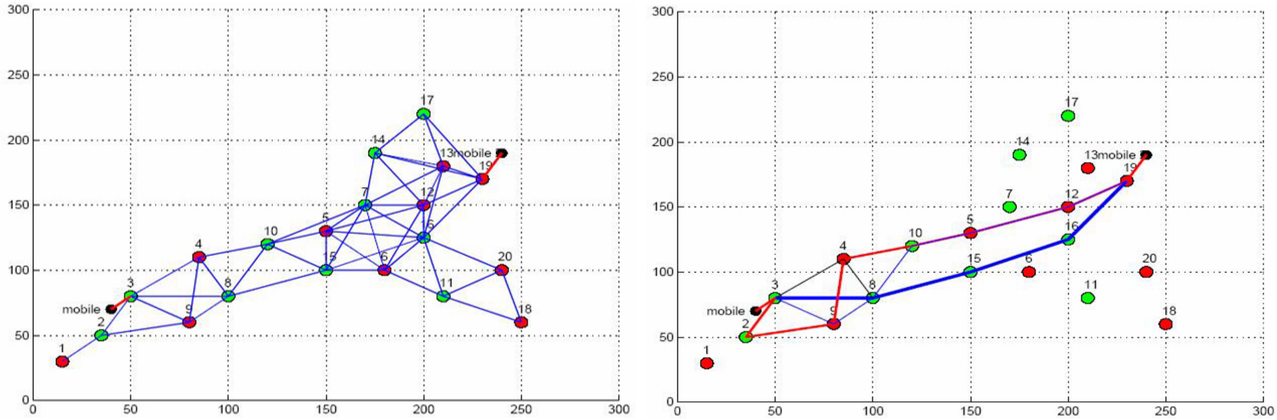

Figure 9. Modified EAODV routing : All routes (a) - Selected route (b)

[4] K. Romer and F. Mattern, "ESF/PESC Exploratory Workshop on Wireless Sensor Networks - Scientific Report", *Institute for Pervasive Computing Department of Computer Science ETH Zurich - Switzerland*, , no. 1-2, April 2004.

[5] D. Simplot-Ryl (INRIA Futurs, Lille), "Some Real-time issues in Wireless Sensor Networks", *Ecole d'Eté Temps Réel ETR'2005 - Nancy*, vol. ISBN: 2 905267-47-X, pp. 255–259, September 2005.

[6] Working Group for WPANs, "Cluster Tree Network - IEEE P802.15", 2001.

[7] ZigBee Alliance, (1997), LAN-MAN Standards Committee of the IEEE Computer Society, Wireless LAN medium access control (MAC) and physical layer (PHY) specification, IEEE Std 802.11 [Online]. Available: http://www.caba.org/standard/zigbee.html

[8] S. Coleri-Ergen, "ZigBee/IEEE 802.15.4 Summary", *Email: csinem@eecs.berkeley.edu*, September 2004.

[9] E. Wiklund, "Concepts of Wireless Communications - White Paper", *Xcipo Innovation AB*.

[10] A. Koubaa, M. Alves, and E. Tovar, "A Real-Time Sensor Network Architecture with Power-Efficient Resource Reservation in IEEE 802.15.4", *Polytechnic Institute of Porto (ISEP-IPP)*, vol. TR-050801, August 2005.

[11] M. Frodigh, P. Johansson, and P. Larsson, "Wireless ad hoc networking The art of networking without a network", *Ericsson*, , no. Review No. 4, 2000.

[12] S. Yoo, D. Kim, and M.-L. Pham, "Scheduling Support for Guaranteed Time Services in IEEE 802.15.4 Low Rate WPAN", *Proceedings 11th IEEE International Conference RTCSA'05*, vol. 400–406, August 2005.

[13] F. Cottet, J. Delacroix, C. Kaiser, and Z. Mammeri, *Ordonnancement temps réel Ordonnancement centralisé*, Edition Hermes 2000 - ISBN : 2-7462-0099-6, 1989.

[14] S. Chebira, "Ph D Thesis. Définition d'une stratégie de gestion locale d'un réseau sans fil à contraintes d'échéances strictes et économie d'énergie", *University of Paris XII- Créteil*, March 2006.

[15] Standard for Information Technology-Part 15.4:, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Low Rate Wireless Personal Area Networks (LR-WPANS)", *Specifications 802.15.4-IEEE*, 2003.

[16] A. Koubaa, M. Alves, and E. Tovar, "IEEE 802.15.4 for Wireless Sensor Networks: A Technical Overview", *Polytechnic Institute of Porto (ISEP-IPP)*, vol. TR-050702, July 2005.

[17] S. Jeannenot, P. Richard, and F. Ridouard, "Ordonnancement Temps Réel avec profils variables de consommation d´énergie", *proc. Real-Time Embedded Systems*, 2004.

[18] M. Hoffmann, N. Krommenacker, and V. Lecuire (CRAN, Nancy), "Validation d'ordonnancement de messages periodiques sur réseau sans fil 802.11b", vol. 13th International Conference on Real-Time Systems RTS'2005, Paris, France, no. 5-6, april 2005.

[19] S. Martin, P. Minet, and L. George (ECE-INRIA, Paris), "Improving non preemptive fixed priority scheduling with dynamic priority as second criterion", vol. 13th International Conference on Real-Time Systems, RTS05, Paris, France, April 2005.

[20] T. Val and G. Juanole (LAAS-ICARE, Toulouse), "QdS dans les réseaux sans fil", *Ecole d'Eté Temps Réel ETR'2005 - Nancy*, vol. ISBN: 2 905267-47-X, pp. 285–294, Septembre 2005.

[21] F. Cottet, J. Delacroix, C. Kaiser, and Z. Mammeri, *Ordonnancement Temps Réel*, Hermes Sciences ISBN 2-7462-0099-6, Erewhon, NC, 2000.

[22] M. Schartz, "Telecommunication Networks: Protocols, Modeling and Analysis", *Addison-Wesley*, vol. Ch. 8, 1988.

[23] G.Lu, B. Chrishnamashari, and C. Raghanavandra, "Performance evaluation of the IEEE 802.15.4 MAC for Low-Rate, Low-Power Wireless Network", *EWCN conference*, April 2004.

[24] C. Perkins and E. Royer, "Ad-hoc on-demand distance-vector (AODV) routing", *In Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999.

[25] "Internet Engineering TaskForce", *MANET Working Group Hakim Badis INTERNET-DRAFT Khaldoun A. Agha (http://www.ietf.org)*, October 2004.

[26] Z. Mammeri (IRIT, Toulouse), "Qualité de Service dans les réseaux : problématique, solutions et challenges", *Ecole d'Eté Temps Réel ETR'2005 - Nancy*, vol. ISBN: 2 905267-47-X, pp. 207–223, Septembre 2005.

[27] "IEEE 802.15 WPAN Task Group 4 (TG4)", *http://www.ieee802.org/15/pub/TG4.html*.

[28] B. Gaujal and N. Navet, "A New EDF Feasibility Test : RR-5125", *Rapport de recherche de l'INRIA - Lorraine , Equipe : TRIO*, Février 2004.

[29] Y.-H. Lu, L. Benini, and G. D. Micheli, "Power-Aware Operating Systems for Interactive Systems", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 10, no. 2, pp. 119–134, April 2002.