Real-Time Syst (2008) 39: 169–204
DOI 10.1007/s11241-007-9038-x

An implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks: theory and practice

**Anis Koubâa · Mário Alves · Eduardo Tovar ·
André Cunha**

Abstract Timeliness guarantee is an important feature of the recently standardized IEEE 802.15.4 protocol, turning it quite appealing for Wireless Sensor Network (WSN) applications under timing constraints. When operating in beacon-enabled mode, this protocol allows nodes with real-time requirements to allocate Guaranteed Time Slots (GTS) in the contention-free period. The protocol natively supports explicit GTS allocation, i.e. a node allocates a number of time slots in each superframe for exclusive use. The limitation of this explicit GTS allocation is that GTS resources may quickly disappear, since a maximum of seven GTSs can be allocated in each superframe, preventing other nodes to benefit from guaranteed service. Moreover, the GTS may be underutilized, resulting in wasted bandwidth. To overcome these limitations, this paper proposes **i-GAME**, an **i**mplicit **G**TS **A**llocation **Me**chanism in beacon-enabled IEEE 802.15.4 networks. The allocation is based on implicit GTS allocation requests, taking into account the traffic specifications and the delay requirements of the flows. The i-GAME approach enables the use of one GTS by multiple

---

This work was partially funded by FCT under CISTER research unit (UI608), by the PLURALITY project (CONC-REEQ/900/2001) and ARTIST2.

A. Koubâa (✉) · M. Alves · E. Tovar · A. Cunha
IPP-HURRAY! Research Group, Polytechnic Institute of Porto, Rua Dr. Antonio Bernardino de Almeida, 431, 4200-072 Porto, Portugal
e-mail: aska@isep.ipp.pt

M. Alves
e-mail: mjf@isep.ipp.pt

E. Tovar
e-mail: emt@isep.ipp.pt

A. Cunha
e-mail: arec@isep.ipp.pt

A. Koubâa
College of Computer Science and Information Systems, Al-Imam Muhammad Ibn Saud University, 11681 Riyadh, Saudi Arabia

nodes, still guaranteeing that all their (delay, bandwidth) requirements are satisfied. For that purpose, we propose an admission control algorithm that enables to decide whether to accept a new GTS allocation request or not, based not only on the remaining time slots, but also on the traffic specifications of the flows, their delay requirements and the available bandwidth resources. We show that our approach improves the bandwidth utilization as compared to the native explicit allocation mechanism defined in the IEEE 802.15.4 standard. We also present some practical considerations for the implementation of i-GAME, ensuring backward compatibility with the IEEE 801.5.4 standard with only minor add-ons. Finally, an experimental evaluation on a real system that validates our theoretical analysis and demonstrates the implementation of i-GAME is also presented.

Keywords IEEE 802.15.4 · Wireless sensor networks · Real-time · Guaranteed time slots · Performance evaluation · Network calculus

1 Introduction

The IEEE 802.15.4 protocol (IEEE-TG15.4 2003) has been recently adopted as a communication standard for Low-Rate Wireless Local Area Networks (LR-WPANs). It presents the advantage to be flexible enough for fitting different requirements of potential applications by adequately tuning its parameters. Even though the IEEE 802.15.4 protocol was not specifically designed for Wireless Sensor Networks (WSNs), it is potentially suitable for them. In fact, low data rate, low power consumption and low cost wireless networking are the key features of the IEEE 802.15.4 protocol, which typically fit the requirements of WSNs.

More specifically, the IEEE 802.15.4 Medium Access Control (MAC) protocol has the ability to provide very low duty cycles (down to 0.006%). This feature is particularly interesting for WSN applications, where energy consumption and network lifetime are main concerns. Additionally, the IEEE 802.15.4 protocol also provides real-time guarantees by using the Guaranteed-Time Slot (GTS) mechanism. This feature is quite attractive for time-sensitive WSNs. In fact, when operating in beacon-enabled mode, i.e. beacon frames are transmitted periodically by a central node called *PAN coordinator* for synchronizing the network, the IEEE 802.15.4 protocol allows the allocation/deallocation of GTSs in a superframe for nodes that require real-time guarantees. Hence, the GTS mechanism provides a minimum service guarantee for the corresponding nodes and enables the prediction of the worst-case performance for each node's application.

However, the GTS mechanism, as proposed in the standard (IEEE-TG15.4 2003), presents some limitations in terms of efficiency and deployment in WSNs with a large number of nodes. In fact, during each superframe (divided into sixteen time slots) only up to seven GTSs (1 up to 15 time slots per GTS) can be allocated, forming the *Contention-Free Period* (CFP) (see Fig. 1). The remaining time slots in the superframe compose the *Contention Access Period* (CAP) using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) as a MAC protocol.

Since each GTS is exclusively assigned to one node, the number of nodes involved in the CFP is limited to seven or less. This is because the IEEE 802.15.4 standard

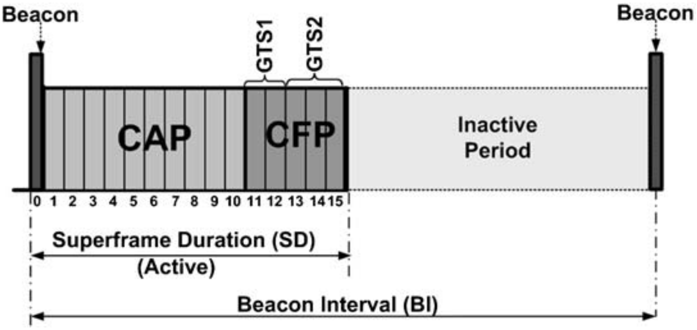


Fig. 1 Beacon interval and superframe structure

(IEEE-TG15.4 2003) assumes that a node performs an **explicit GTS allocation request** by asking the PAN coordinator for a certain number of time slots. A node is allowed to transmit during the CFP, if the number of available time slots in the superframe is higher than requested, and the minimum CAP length is not violated due to the allocation (IEEE-TG15.4 2003). Two negative impacts may result from this explicit allocation scheme.

1. The GTSs can be quickly consumed by a few number of nodes, preventing the others from having a guaranteed service.
2. A node with a low arrival rate that has been allocated a GTS, may use it only partially (when the amount of guaranteed bandwidth is higher than its arrival rate). This leads to underutilization of the GTS bandwidth resources. Due to the pre-fixed time slot duration in a superframe, it is practically impossible to balance the arrival rate of a node and its guaranteed GTS bandwidth. The amount of wasted bandwidth increases with the variance between the guaranteed bandwidth and the arrival rate. Note that this wasted bandwidth can be used by the CAP.

This paper proposes a simple and effective solution to overcome the previously described limitations of the explicit GTS allocation in the IEEE 802.15.4 protocol. Basically, the idea consists in sharing the same GTS between multiple nodes, instead of being exclusively dedicated to one node, if a certain schedule that satisfies the requirements of all requesting nodes exists. Sharing a GTS by several nodes means that the time slots of this GTS are dynamically allocated to different nodes in each superframe, according to a given schedule. In contrast, an explicit allocation statically devotes a GTS to only one node in all subsequent superframes. Hence, the GTS allocation mechanism proposed in this paper is based on the traffic specification of the requesting nodes, their delay requirements, and the available GTS resources. Instead of asking for a fixed number of time slots, a node that wants to have a guaranteed service sends its traffic specification and delay requirement to the PAN coordinator. Then, the latter runs an admission control algorithm based on this information and the amount of available GTS resources. The new allocation request will be accepted if there is a schedule that satisfies its requirements and those of all other previously accepted allocation requests; otherwise, the new allocation request is rejected. We refer to this as the *implicit GTS allocation mechanism* (i-GAME). We show that i-GAME has the advantage of accepting multiple flows sharing the same GTS, while still meeting their delay requirements. It also improves the utilization of the CFP by reducing the amount of wasted bandwidth of GTSs and maximizes the duration of the CAP, since the CFP length is reduced to a minimum.

Related work The performance of the explicit GTS allocation in IEEE 802.15.4 has been recently evaluated in Koubâa et al. (2006). That work proposes a delay bound analysis of an explicit GTS allocation. It also analyzes the impact of the beacon and superframe orders on the throughput, delay and power efficiency of a GTS allocation. In this paper, we extend the work in Koubâa et al. (2006) by considering implicit GTS allocations. We also prove the improvement as compared to the explicit GTS allocation approach, in terms of bandwidth utilization.

Basically, the problem that we are addressing in this paper can be regarded as analyzing the schedulability of a given number of flows sharing a certain number of time slots. Schedulability analysis of periodic task set using the GTS mechanism of the IEEE 802.15.4 protocol has been addressed in Yoo et al. (2005). In that paper, the authors have proposed a scheduling algorithm for periodic tasks, which generates the appropriate network parameters, namely the Beacon Order (BO), the Superframe Order (SO) and the GTS information that satisfy the timing constraints of the flows. In this paper, we consider more general flows by relaxing the periodicity constraint, and we propose a simple and efficient scheduling algorithm (i-GAME) with an admission control function to optimize the utilization of the GTSs.

On the other hand, the problem of multicycle scheduling has already been addressed by some works in the literature, but within completely different contexts and assumptions (Cavalieri et al. 2003; Raja and Noubir 1993; Feng and Mok 2002), as briefly outlined next. For instance, in Cavalieri et al. (2003), Raja and Noubir (1993), the authors have addressed multicycle polling scheduling in fieldbus networks. These papers have contributed to the schedulability analysis of a set of periodic tasks with deadlines equal to periods under Rate Monotonic (RM) and Earliest Deadline First (EDF) scheduling policies, where the nodes polled are different from one cycle to another. In both approaches, the idea consists in finding the minimum cycle, called primary cycle, which corresponds to the greatest common divisor of all task periods, and computing the number of time slots needed to transmit periodic traffic inside each cycle, if the task set is schedulable. The last step consists in executing tasks according to their priorities (using RM or EDF) in each primary cycle.

Our work differs from these approaches in two aspects. First, we don't consider periodic message arrivals, but we adopt a more general representation of the traffic using the (b, r)-curve model where b is the burst size of the flow and r is the average rate. This traffic model also incorporates the classical representation of the periodic arrival model with or without jitter (Koubâa and Song 2004). For that reason, our analysis is based on the Network Calculus theory. Second, the durations of the cycles in the previously referred approaches are fixed and related to the periods of the flows. This does not match with our case, since in the IEEE 802.15.4 protocol one cycle is represented by a Beacon Interval (BI) (see Fig. 1), whose duration depends on the beacon order parameter as it will be shown in Sect. 2. Moreover, since the IEEE 802.15.4 protocol does not allow more than seven GTS allocations, this may restrict the number of time slots in each cycle in contrast with the approaches in Cavalieri et al. (2003), Raja and Noubir (1993), where the number of time slots is only limited by the durations of the primary cycle and the time slot.

Contributions of this paper The contributions of this paper are the following.

- First, we present the motivation for an implicit GTS allocation mechanism for the IEEE 802.15.4 protocol, showing that the explicit allocation mechanism proposed by the standard lacks bandwidth efficiency, particularly for low rate WSN applications (Sect. 4). We also introduce the implicit allocation mechanism, i-GAME, through a practical example.
- Second, we evaluate the schedulability analysis of an implicit GTS allocation of k time slots shared by N nodes, where $k < N$, under round robin scheduling (Sect. 5). For that purpose we derive the service curve and the delay bound guaranteed by such an allocation, defined by the tuple (b, r, D) where b is the burst size, r is the arrival rate, and D is the delay requirement.
- Third, we present the i-GAME admission control mechanism, based on our analysis and we provide some guidelines for its implementation, with minor add-ons to the IEEE 802.15.4 standard protocol defined in IEEE-TG15.4 (2003) (Sect. 6).
- Finally, we present an experimental evaluation that demonstrates the practical feasibility of the i-GAME mechanism on a real-system implementation and provides a practical validation of the theoretical analysis (Sect. 7).

2 Background

2.1 Overview of the IEEE 802.15.4 MAC protocol

The IEEE 802.15.4 MAC protocol supports two operational modes that may be selected by a central node called *PAN coordinator*: (1) the *non beacon-enabled mode* where the MAC is ruled by non-slotted CSMA/CA; (2) the *beacon-enabled mode* where beacons are periodically sent by the PAN coordinator to identify its PAN and synchronize nodes that are associated with it. The most relevant MAC features are outlined next.

In this paper, we only consider the beacon-enabled mode, since it enables GTS allocations. In beacon-enabled mode, the *Beacon Interval* (BI) defines the time between two consecutive beacons, and includes an active period and, optionally, an inactive period. The active period, called *superframe*, is divided into 16 equally-sized time slots, during which data frame transmissions are allowed. During the inactive period (if it exists), all nodes may enter into a sleep mode, thus saving energy. Figure 1 illustrates the beacon interval and the superframe structure.

The Beacon Interval and the *Superframe Duration* (SD) are determined by two parameters, the *Beacon Order* (BO) and the *Superframe Order* (SO), respectively.

The Beacon Interval is defined as follows:

$$BI = aBaseSuperframeDuration \cdot 2^{BO}, \quad \text{for } 0 \le BO \le 14. \tag{1}$$

The Superframe Duration, which determines the length of the active period, is defined as follows:

$$SD = aBaseSuperframeDuration \cdot 2^{SO}, \quad \text{for } 0 \leq SO \leq BO \leq 14. \tag{2}$$

In (1) and (2), *aBaseSuperframeDuration* denotes the minimum duration of the superframe, corresponding to $SO = 0$. This value corresponds to 15.36 ms, assuming 250 kbps in the 2.4 GHz frequency band, which will be considered throughout the rest of this paper.

By default, the nodes compete for medium access using slotted CSMA/CA within the *Contention Access Period* (CAP) during *SD*. In case of a busy channel, a node computes its backoff period based on a random number of time slots. The IEEE 802.15.4 protocol also offers the possibility of having a *Contention-Free Period* (CFP) within the superframe (Fig. 1). The CFP, being optional, is activated upon request from a node to the PAN coordinator for allocating time slots depending on the node's requirements. Upon receiving this request, the PAN coordinator checks whether there are sufficient resources and, if possible, allocates the requested time slots. These time slots are called *Guaranteed Time Slots* (GTSs) and constitute the CFP. If the available resources are not sufficient, the GTS allocation request fails. The corresponding node then must send its data frames during the CAP. More details can be found in IEEE-TG15.4 (2003).

2.2 Delay bound analysis using network calculus

Network Calculus is a mathematical tool based on *min-plus* and *max-plus* algebras for designing and analyzing deterministic queuing systems (Leboudec and Thiran 2001). For a given data flow, the *input function* is the cumulative arrival function denoted by $R(t)$, which represents the number of bits that arrive during the interval $[0, t]$. We denote by $R^*(t)$ the *output function* of the flow, which represents the number of bits that leave the system during the interval $[0, t]$.

Furthermore, Network Calculus theory assumes that:

1. It exists an *arrival curve* $\alpha(t)$ that upper bounds $R(t)$ such that $\forall s$, $0 \leq s \leq t$, $R(t) - R(s) \leq \alpha(t - s)$. This inequality means that the amount of traffic that arrives to receive service in any interval $[s, t]$ never exceeds $\alpha(t - s)$. It is also said that $R(t)$ is constrained by $\alpha(t)$, or $R(t) \sim \alpha(t)$.
2. It exists a minimum *service curve* $\beta(t)$ guaranteed to $R(t)$. This means that the output flow during any given *busy period* $[t, t + \Delta]$ of the flow is at least equal to $\beta(\Delta)$, i.e. $R^*(t + \Delta) - R^*(t) \geq \beta(\Delta)$, where $\Delta > 0$ is the duration of any busy period.

The knowledge of the arrival and service curves enables the computation of the delay bound $D_{\max}$, which represents the worst-case response time of a message, and the backlog bound $Q_{\max}$, which is the maximum queue length of the flow.

The *delay bound*, $D_{\max}$, for a data flow with an arrival curve $\alpha(t)$ that receives the service curve $\beta(t)$ is the maximum horizontal distance between $\alpha(t)$ and $\beta(t)$ (see Fig. 2), and is expressed as follows:

$$D_{\max} = h(\alpha, \beta) = \sup_{s \geq 0}\{\inf\{\tau \geq 0 : \alpha(s) \leq \beta(s + \tau)\}\}. \tag{3}$$

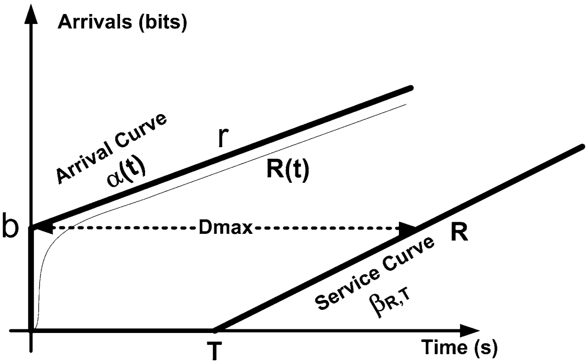


Fig. 2 Arrival curve, service curve and delay bound

Figure 2 presents an example of the delay and backlog bound computation for a linear arrival curve $\alpha(t) = b + r \cdot t$ that receives a *rate-latency* service curve $\beta_{R,T}(t) = R \cdot (t - T)^+$, where $R \geq r$ is the guaranteed bandwidth, T is the maximum latency of the service and $(x)^+ = \max(0, x)$.This service curve is typically used for servers that provide a bandwidth guarantee with a certain latency. The latency T refers to the deviation of the service (e.g. blocking factor of non-preemptive transmissions).

The delay bound $D_{\max}$ (presented in Fig. 2) guaranteed for the data flow with the arrival curve $\alpha(t) = b + r \cdot t$ (also called (b, r) curve) by the service curve $\beta_{R,T}(t) = R \cdot (t - T)^+$ is computed as follows Leboudec and Thiran (2001):

$$D_{\max} = \frac{b}{R} + T. \tag{4}$$

3 The explicit GTS allocation in IEEE 802.15.4

3.1 The explicit GTS allocation mechanism overview

In this section, we present a brief overview on the explicit GTS allocation protocol specified in the IEEE 802.15.4 standard. The IEEE 802.15.4 protocol offers the possibility of having a *Contention-Free Period* (CFP) within the superframe as shown in Fig. 1. The CFP is defined by a set of time guaranteed time slots requested by nodes that have timing requirements. The CFP, being optional, is activated upon request from a node to the PAN coordinator for allocating a certain number of time slots.

Figure 3 shows the GTS characteristics field format sent within an allocation request command frame (IEEE-TG15.4 2003) by a node to the PAN coordinator.

The node explicitly expresses the number of time slots that it wants to allocate in the *GTS Length* field. Note that the GTS length can be up to 15 time slots. The *GTS Direction* field specifies if the GTS is in receive-only mode (value = 1), i.e. data is transmitted from the PAN coordinator to the requesting node, or in transmit-only mode (value = 0), i.e. data is transmitted from the requesting node to the PAN coordinator. The *Characteristics Type* field refers to a GTS allocation if it is set to one or a GTS deallocation if it is set to zero.

Fig. 3 GTS characteristics field format in IEEE 802.15.4

bits	0-3	4	5	6-7
	GTS Length	GTS Direction	Characteristics Type	Reserved

Fig. 4 GTS Descriptor Field Format in IEEE 802.15.4

bits	0-15	16-19	20-23
	Node Address	GTS Starting Slot	GTS Length

Upon receiving this request, the PAN coordinator checks whether there are sufficient time slots available in the superframe for this request. If the number of available time slots in the superframe is smaller than the number requested, the GTS allocation request is rejected, otherwise it is accepted. The PAN coordinator must ensure that the CAP length remains always greater than *aMinCAPLength* equal to 7.04 ms (IEEE-TG15.4 2003). In the former case, the corresponding node may send its data frames during the CAP, but with no guarantee. If the GTS allocation request is accepted, the admitted node must keep track of beacon frames for checking which time slots have been allocated in the current superframe. This information is located in the GTS descriptor field (Fig. 4), which is embedded in each beacon frame. A beacon frame cannot have more than seven GTS descriptors, limiting the number of GTSs to seven.

The explicit GTS allocation adopted by the standard has the advantage of being simple. However, it may be not efficient enough in terms of bandwidth utilization for flows with low arrival rates, which is typically the case in wireless sensor networks, since the guaranteed bandwidth of a GTS can be much higher than the arrival rates, as explained in Sect. 4.2.

3.2 Delay bound analysis of a GTS allocation in an IEEE 802.15.4 cluster

The delay bound analysis of the explicit GTS allocation has been presented in Koubâa et al. (2006). The analysis was made for an IEEE 802.15.4 cluster, operating in beacon-enabled mode, with a unique PAN coordinator, and a set of nodes within its radio coverage. Data flows sent by these nodes and that corresponds to a GTS allocation are assumed to have a cumulative arrival function $R(t)$ upper bounded by the linear arrival curve $\alpha(t) = b + r \cdot t$ with b denoting the maximum burst size, and r being the average arrival rate. The objective was to compute the delay bound of a given flow bounded by an arrival curve $\alpha(t) = b + r \cdot t$ and allocating a GTS with n time slots.

To compute the delay bound, the service curve that corresponds to a GTS allocation of n time slots has been derived. Figure 5 shows the example of service curves and delay bounds for an allocation of 1, 2 and 3 time slots.

More formally, it has been shown that the service curve offered by a GTS allocation of n time slots is approximated by a rate-latency service curve $\beta_{R_n,T_n}(t) = R_n \cdot (t - T_n)$, where R_n is the guaranteed bandwidth of a GTS defined as:

$$R_n = n \cdot \left(\frac{T_{\text{data}}}{BI} \cdot C\right) \tag{5}$$

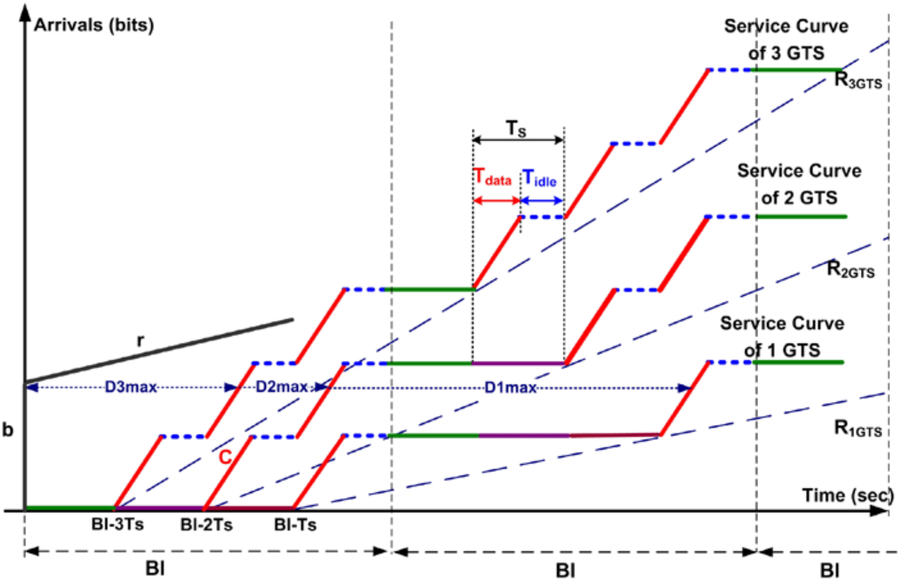


Fig. 5 The GTS service curve for *n* time slot allocation

and T_n is the latency of the service expressed as:

$$T_n = BI - n \cdot TS. \tag{6}$$

T_{data} defines the maximum duration used for data frame transmission inside a GTS, without taking the control overheads (inter-frame spacing (IFS) and acknowledgement) into account. C denotes the data rate equal to 250 kpbs.

As a result, it is shown that the delay bound guaranteed by the service curve $\beta_{R_n,T_n}(t)$ for a data flow bounded by a (b, r) curve is:

$$D_{n,\max} = \frac{b}{n \cdot ((T_{\text{data}} \cdot C)/BI)} + (BI - n \cdot Ts). \tag{7}$$

Another service curve in the form of a stair function was also derived in Koubâa et al. (2006). However, the analysis presented in that paper considers the rate-latency service curve $\beta_{R_n,T_n}(t)$ of one GTS.

4 i-GAME: an implicit GTS allocation mechanism

4.1 System model and assumptions

We consider an IEEE 802.15.4 cluster composed of a set of sensor nodes in the range of a particular node considered as the PAN coordinator. Note that this star topology may be particularly interesting for large-scale sensor networks when using clustering and/or two-tiered architectures (Koubâa et al. 2005). Moreover, the IEEE 802.15.4

supports cluster-tree topologies, which extend the star network by means of child coordinators (IEEE-TG15.4 2003) that synchronize the nodes out of the range of the PAN coordinator.

We assume that the PAN coordinator sets up the network with a superframe structure defined by the beacon order *BO* and the superframe order *SO*. The beacon interval (*BI*) and the superframe duration (*SD*) are computed using (1) and (2), respectively. We further assume in the analysis that the channel is error-free.

Each node i generates a flow F_i bounded by the arrival curve $\alpha_i(t) = b_i + r_i \cdot t$, where b_i is the maximum burst size, r_i is the average arrival rate and D_i denotes the delay requirement of flow F_i. We represent flow F_i by the tuple $F_{\mathrm{spec},i} = (b_i, r_i, D_i)$.

Let R_{TS} denote the guaranteed bandwidth per one time slot. Observe that R_{TS} can be computed using (5) for $n = 1$. For a GTS with length of k allocated time slots $(k < 15)$, we denote as R_{kTS} the bandwidth guaranteed by k time slots expressed as:

$$R_{kTS} = k \cdot R_{TS}. \tag{8}$$

The main problem addressed in this paper is how to **fairly share** the allocation of k time slots in the CFP between N requesting nodes, with respect to their flow specifications $F_{\mathrm{spec},i} = (b_i, r_i, D_i)$.

Intuitively, N flows are allowed to share a GTS allocation of k time slots, if two necessary conditions (C1) and (C2) hold:

$$\begin{array}{ll} \text{(C1)} & \sum_{i=1}^{N} r_i \le R_{kTS}, \\ \text{(C2)} & D_{i,\max} \le D_i, \quad \forall 1 \le i \le N. \end{array} \tag{9}$$

(C1) states that the sum of all arrival rates does not exceed the entire bandwidth of k time slots. (C2) states that the delay bound guaranteed by the allocation does not exceed the delay requirement, for each flow F_i.

4.2 Bandwidth utilization of explicit GTS allocations

This section defines the bandwidth utilization of a GTS allocation. It also presents the limitations of an explicit allocation in terms of bandwidth utilization efficiency.

Consider a flow $F_i = (b_i, r_i, D_i)$ that has an explicit GTS allocation of k_i time slots. Then, the bandwidth utilization of this GTS allocation is defined as:

$$U_{k_i TS} = r_i / R_{k_i TS} = r_i / (k_i \cdot R_{TS}). \tag{10}$$

Now, for a CFP of length k time slots, $k \le 15$, containing all allocated GTSs ($k = \sum_{i=1}^{N} k_i$) and corresponding to N allocating nodes, the average bandwidth utilization of the CFP is defined as:

$$U_{CFP} = U_{kTS} = \frac{1}{N} \sum_{i=1}^{N} U_{k_i TS} = \frac{1}{N \cdot R_{TS}} \sum_{i=1}^{N} \frac{r_i}{k_i}. \tag{11}$$

Observe that the minimum bandwidth that can be allocated is R_{TS} (it is not divisible). It is logical to assume that, with an explicit allocation, a node i that requests a

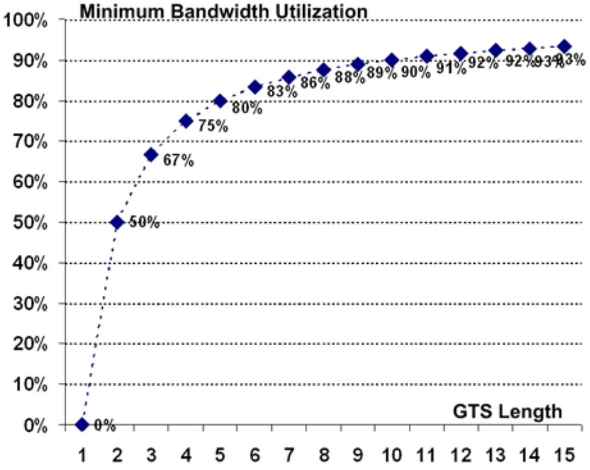


Fig. 6 Minimum utilization limits of an explicit allocation

GTS allocation of k_i time slots has an arrival rate r_i that satisfies:

$$(k_i - 1) \cdot R_{TS} < r_i \leq k_i \cdot R_{TS}. \tag{12}$$

From (8), (10) and (12), we obtain:

$$\frac{(k_i - 1)}{k_i} < U_{k_i TS} \leq 1. \tag{13}$$

Then, the minimum utilization limit is defined as:

$$U_{\min}^{k} = \frac{(k-1)}{k} \quad \forall k, 1 \leq k \leq 15. \tag{14}$$

Figure 6 presents the minimum utilization limits for different GTS length values, for one node.

From Fig. 6, it can be understood that the lowest utilizations can be experimented for GTSs with one time slot allocation. This is because the arrival rates of the flows can be low fractions of the indivisible R_{TS}, which triggers the motivation for sharing the time slot with other nodes, if the delay requirements of the flows can still be satisfied. This case is most likely to happen in sensor networks since their arrival rates may be particularly low.

4.3 Improving bandwidth utilization via implicit GTS allocations: i-GAME

According to condition (C1) in (9), N flows may share one GTS if the sum of their arrival rates is smaller or equal to the guaranteed bandwidth of the GTS. The main problem in this case is to find the adequate time slot allocation schedule in each beacon interval that respects a per-flow guaranteed bandwidth greater or equal to its arrival rate. The complexity of finding the adequate schedule depends on the number of GTS allocation requests and on the per-flow utilization of the GTS. A particular simple form of sharing the GTS is by using round robin scheduling, thus providing a fair share. However, round robin offers the same amount of guaranteed bandwidth to all flows without any differentiation. Hence, round robin is adequate when the arrival rate of each flow sharing the GTS is smaller than the bandwidth guaranteed by a fair share of the GTS. More formally, for a GTS allocation of k time slots fairly shared by N flows $F_{\text{spec},i} = (b_i, r_i, D_i)$, $i = 1 \ldots, N$, then:

$$r_i \le \frac{k \cdot R_{Ts}}{N}, \quad \forall i = 1, \ldots, N. \tag{15}$$

Note that the fair sharing of a GTS is effective when the arrival rates of the flows are similar. For instance, a flow with an arrival rate of 20 kbps cannot fairly share the same resource with a flow with an arrival rate of 1 kbps. Hence, we assume that (12) must hold for flows that are candidates for sharing the same GTS with other flows. This assumption is relevant for some WSN applications, where flows generated by sensor nodes have similar behaviors.

For the sake of simplicity and without loss of generality, in this paper we analyze flows with (at most) one time slot allocation. We make this assumption for two reasons.

1. It is common in WSNs that flows are generated at low rates. It has been shown in Koubâa et al. (2006) that the guaranteed bandwidth per one time slot allocation for a full duty cycle ($BO = SO$), is comprised between 9.38 kbps and 13.50 kbps, depending on the superframe order (SO). The traffic pattern of most WSN applications should have arrival rates much lower than these values, since in WSNs it is most likely to have a large number of nodes with low rates rather than a small number of nodes with high rates.
2. According to Fig. 6, the case of one time slot allocation is the most interesting for the i-GAME approach since the utilization (without i-GAME) can be very low (less than 50%), particularly for flows with low rates.

Note that the methodology presented next can be easily extended based on the same principles in order to merge flows requesting the same number of k time slots (satisfying (12), for $k > 1$) into one GTS with reduced size.

Based on the definition of implicit GTS allocation, the utilization of the GTS of k time slots shared by N flows $F_{\text{spec},i} = (b_i, r_i, D_i)$, $i = 1, \ldots, N$, is defined as:

$$U_{kTS}^{N} = \frac{1}{k \cdot R_{TS}} \sum_{i=1}^{N} r_i. \tag{16}$$

The bandwidth utilization of each flow in the GTS is defined as:

$$U_{i,kTS} = \frac{r_i}{k \cdot R_{TS}}. \tag{17}$$

In summary, this paper considers flows requesting an implicit GTS allocation with arrival rates $r_i \le R_{Ts}$, which corresponds to one time slot allocation in case of an explicit allocation. Our problem is then reduced to find a fair share of implicit GTS allocations into a CFP with a length of k time slots for N requesting nodes, where $k \le N$. Note that in this case, the CFP length (corresponding to implicit allocations) does not exceed seven time slots ($k < 7$) since only seven GTSs, each of one time slot length, can be allocated in a given superframe.

4.4 A practical intuition on the i-GAME approach

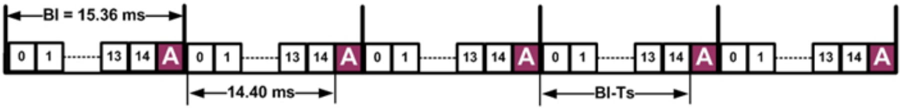
To give a practical intuition on the implicit GTS allocation approach, we present the following illustrative example.

Assume an IEEE 802.15.4 cluster where the PAN coordinator sets up the superframe structure with $BO = 0$ and $SO = 0$. This configuration corresponds to $BI = SD = 15.36$ ms, $Ts = 0.96$ ms and $R_{TS} = 9.38$ kbps (Koubâa et al. 2006).

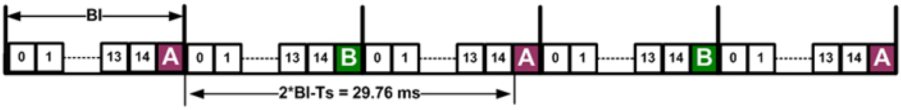
A first request Now, let a node A generate a flow F_A bounded by the arrival curve $\alpha_A(t) = 0.2 + 3 \cdot t$ kbits (a burst size with $b_{\mathrm{A}} = 200$ bits and an arrival rate of $r_A = 3$ kbps) and with a delay requirement $D_A = 150$ ms. Then, $F_{\mathrm{spec},A} = (200$ bits, 3 kbps, 150 ms). When node A requests a GTS allocation, it must send $F_{\mathrm{spec},A}$ to the PAN coordinator, which has to decide whether to accept the flow or not. Based on the results in Koubâa et al. (2006), with $BO = 0$ and $SO = 0$ ((5) and (6)), the service curve offered by one time slot allocation is $\beta_{1\mathrm{node},1TS}(t) = 9.38 \cdot (t - 14.40)^+$ kbits. Figs. 7a and 8a present the allocation of the GTS by node A and its service curve, respectively. Using (7), the PAN coordinator can compute the delay bound guaranteed by one time slot allocation based on $F_{\mathrm{spec},A}$. This delay bound is $D_{A,\max} = 35.72$ ms. Observe also that the guaranteed bandwidth by one time slot allocation (9.39 kbps) is higher than the arrival rate (3 kbps). As a result, both conditions (C1) and (C2) in (9) are satisfied; hence, the flow is accepted for one time slot allocation. The GTS will be partially used by node A with an utilization $r_A/R_{TS} = 32\%$ (see (16)).

A second request Assume that a second node B generating a flow F_B with a traffic specification $F_{\mathrm{spec},B} = (400$ bits, 2 kbps, 150 ms) wants to allocate a GTS. The traditional explicit mechanism would require the allocation of a new time slot exclusively for node B. This would lead to an additional wasted bandwidth, as for node A, since the arrival rate of r_B is lower than R_{TS}. We propose a different approach that is to share the previous GTS allocation with node A, if it would be possible to respect both $F_{\mathrm{spec},A}$ and $F_{\mathrm{spec},B}$. The problem is to determine the service curve offered by the same time slot for each flow.

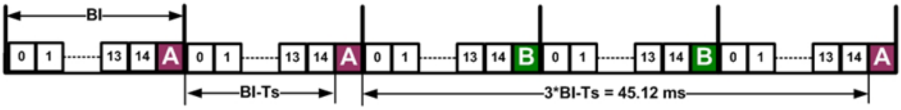
Assuming that the sharing of this time slot is based on round robin scheduling, the time slot alternates between both flows in each beacon interval (refer to Fig. 7b). Figure 8b shows the corresponding service curve for each flow. Since the time slot is shared between two nodes, the bandwidth guaranteed for each flow is equal to $R_{TS}/2$, and the latency is equal to $2 \cdot BI - Ts$ (see Figs. 7b and 8b). As a result, the service curve granted for each flow using round robin is $\beta_{2\mathrm{nodes},1TS}(t) = 4.69 \cdot (t - 29.76)^+$. Now, applying (7) to each flow F_A and F_B using the per-flow service curve $\beta_{2\mathrm{node},1TS}(t)$, we have: $D_{A,\max} = 72.40$ ms and $D_{B,\max} = 115.04$ ms, and thus condition (C2) is satisfied. Observe that the guaranteed rate of one time slot is higher than the sum of the arrival rates of both flows, i.e. condition (C1) is satisfied. As a result, both flows can be accepted to share the same GTS allocation under round robin scheduling. In this case, the utilization of the GTS is equal to $(r_A + r_B)/R_{TS} = 53\%$ (see (16)), obviously higher than that in the previous case.



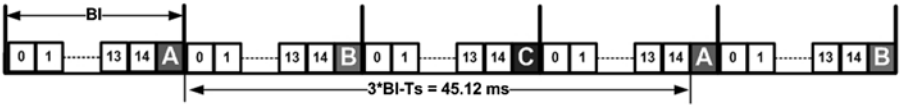
a One time slot allocation used by one node



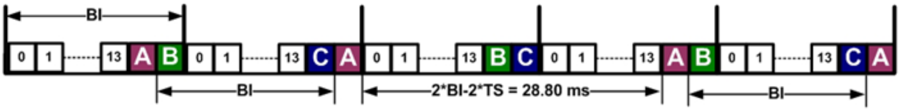
b One time slot allocation used by two nodes under round-robin scheduling



c One time slot allocation used by two nodes under a scheduling different from round robin



d One time slot allocation used by three nodes under round robin scheduling



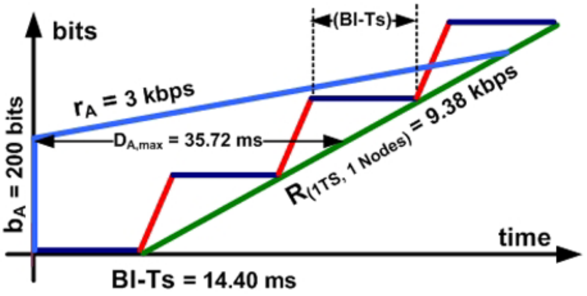
e Two time slot allocation used by three nodes under round robin scheduling

Fig. 7 Different implicit GTS allocations

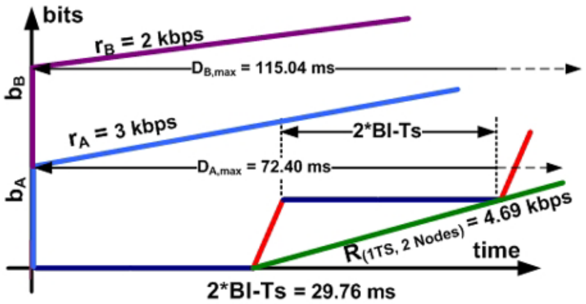
Observe in Fig. 7c that changing the scheduling policy results in a change of the service curve, even if the guaranteed bandwidth is the same. In Fig. 7c, the maximum latency is higher than the one with round robin scheduling.

A third request Now, assume that a third node C generating a flow F_C with a traffic specification $F_{\text{spec},C} = (500\text{ bits, }3\text{ kbps, }150\text{ ms})$ wants to allocate a GTS. Like in the previous requests, we compute the per-flow service curve for each node while sharing one time slot using round robin policy $\beta_{3\text{nodes},1TS}(t) = 3.12 \cdot (t - 45.12)^+$ (see Fig. 7d). The corresponding delay bounds for each of the three flows are: $D_{A,\max} = 109.22\text{ ms} \leq 150\text{ ms}$, $D_{B,\max} = 173.32\text{ ms} \geq 150\text{ ms}$, $D_{C,\max} = 205.4\text{ ms} \geq 150\text{ ms}$.

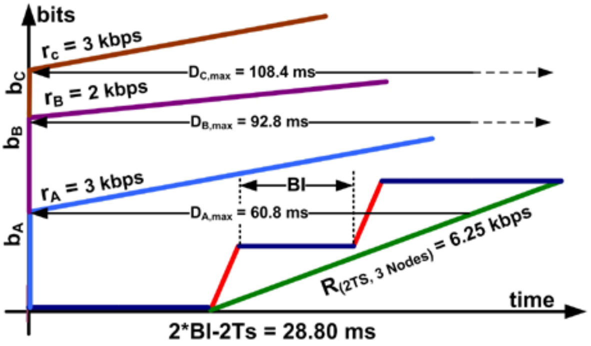
As a consequence, node C cannot be admitted to share the same time slot with A and B, even though the sum of all arrival rates is still lower than the guaranteed bandwidth $(3+2+3 < 9.38)$. Since there are still available resources in the superframe, it is possible to extend the CFP to two time slots and apply the same admission control algorithm to node C, but with a service curve $\beta_{3\text{nodes},2TS}(t) = 6.25 \cdot (t - 28.80)^+$ (see Figs. 7e, 8c). The corresponding delay bounds for each of the three flows are: $D_{A,\max} = 60.8\text{ ms} \leq 150\text{ ms}$; $D_{B,\max} = 92.8\text{ ms} \leq 150\text{ ms}$; $D_{C,\max} = 108.4\text{ ms} \leq 150\text{ ms}$. As a consequence, it is possible to meet the delay requirements of the three flows with only two time slots.



a Service curve of a one time slot allocation used by one node



b Service curve of a one time slot allocation used by two nodes under round robin scheduling



c Service curve of a two time slot allocation used by three nodes under round robin scheduling

Fig. 8 Service curves of implicit GTS allocations

Impact of the delay on the utilization In this latter case, the implicit allocation mechanism saves one time slot compared to an explicit GTS allocation. The bandwidth utilization of the CFP with implicit GTS allocation is then $(r_A + r_B + r_C)/R_{2TS} = 42\%$ (16), whereas in case of an explicit GTS allocation the bandwidth utilization is $(r_A + r_B + r_C)/(3 \cdot R_{TS}) = 28\%$ (11). The improvement in terms of utilization depends on the delay requirement. For more relaxed delay requirements, the improvement on utilization is more significant. For example, if the three flows had a delay requirement of 250 ms, it would be possible to allocate only one time slot, resulting in an utilization of $(r_A + r_B + r_C)/R_{1TS} = 85\%$.

What if the guaranteed bandwidth is lower than the arrival rate? Observe that in the previous scenarios, the guaranteed bandwidths offered by a shared GTS using round robin scheduling are higher than the arrival rates of the three flows. Now, imagine that node C has an arrival rate equal to 7 kbps. In this case, round robin is not sufficient for flow C since the guaranteed rate 6.25 kbps is lower than flow C's arrival rate. A first option is to extend the length of the CFP to have higher bandwidth and compute the corresponding service curve, while still applying round robin. This technique is simple, but it tends to an explicit allocation. Another technique consists in using weighted round robin, by assigning time slots proportionally to the arrival rates, and thus providing differentiated services inside one shared GTS with respect to the arrival rates. Each flow will then have its own service curve with respect to its arrival rate, and the corresponding delay bound would be compared to the delay requirement of the flow, as made previously. This technique is more efficient in terms of utilization, but introduces additional complexity to determine the weights, the schedule and then the corresponding service curves for each flow.

For the sake of simplicity, we consider in this paper the first alternative of extending the CFP length.

5 Schedulability analysis of an implicit GTS allocation under round-robin policy

5.1 Schedulability analysis

This section presents a generalization of the practical intuition presented in Sect. 4.4. Our purpose is to find a general expression of the service curve for N flows that share k time slots, where $k \le N$ using round robin scheduling, assuming that flows F_i have arrival rates $r_i \le R_{Ts}$ (the most relevant for WSN applications). Note that in this particular case, $k < 7$, since the maximum number of GTSs per superframe is limited to 7. Since we are considering a fair share of a GTS using round robin policy, $\beta_i(t)$ is equal to a rate-latency service curve $\beta_{R,T}(t)$ common to all flows sharing the same GTS. Two distinct cases need to be addressed.

1. **Case of $k = N$.** This case is equivalent to an explicit allocation. Each node has its own time slot since round robin is deployed. The delay bound is then computed based on (7) and compared to D_i.
2. **Case of $k < N$.** this case is more interesting because the number of nodes is higher than the allocated time slots, as presented in the example (Sect. 4.4). Obviously, it can be understood from the motivating example that the guaranteed rate for each flow is:

$$R = \frac{R_{kTS}}{N} = \frac{k}{N} \cdot R_{TS}. \tag{18}$$

The main problem is to compute the service latency related to the service curve. Observe, through the examples of Fig. 7, that the latency can be expressed as:

$$T = p \cdot BI + q \cdot Ts, \tag{19}$$

where $p \in \mathbb{N}$ denotes the number of beacon intervals that contributes to the service latency, and $q \in \mathbb{Z}^-$ represents the number of time slots to be subtracted from the latency. For $k \leq 7$ and $N \geq 1$, we have:

$$\begin{aligned} p &= \left\lceil \frac{N}{k} \right\rceil > 0 \quad \text{and} \\ q &= N - p \cdot k - 1 < 0. \end{aligned} \tag{20}$$

Equation (20) can be verified with the examples in Fig. 7 as well as with all other combinations of N and k. As a result, the service curve corresponding to a fair share of k time slots between N nodes for $k < N$ under round robin scheduling is:

$$\beta_{R,T}(t) = \frac{k}{N} \cdot R_{TS} \cdot (t - (p \cdot BI + q \cdot Ts))^+ \tag{21}$$

with p and q defined in (20).

So, for a flow F_i with $r_i \leq R$, the corresponding delay bound guaranteed by the fair share, based on (4), is:

$$D_{i,\max} = N \cdot \frac{b_i}{k \cdot R_{TS}} + (p \cdot BI + q \cdot Ts). \tag{22}$$

Observe that (21) and (22) are general expressions that are also valid in case of $k = N$. In this case, $p = 1$ and $q = -1$ and (21) and (22) are equivalent to (5), (6) and (7) (with $n = 1$), respectively.

In summary, a set of N flows (F_i, $i = 1, \ldots, N$) sharing a number of k time slots where $k \leq N$ are schedulable under round robin, if, for each flow with $F_{\text{spec},i} = (b_i, r_i, D_i)$ we have $r_i \leq k \cdot R_{TS}/N$ ((15) implies condition (C1) in (9)), and $D_{i,\max} \leq D_i$ (equivalent to condition (C2) in (9)) where $D_{i,\max}$ is obtained from (22).

Figure 9 presents an example of delay bound analysis expressed by (22).

Note that, by analogy to the results in Koubâa et al. (2006), it is possible to compute a more accurate delay bound estimation based on the stair service curve of the GTS allocation for a given flow. In the particular case where *the burst size is smaller than the duration of a time slot* ($b_i \leq Ts$), i.e. all the frames forming the burst can be sent in during one time slot, a more accurate delay bound is expressed as follows (see

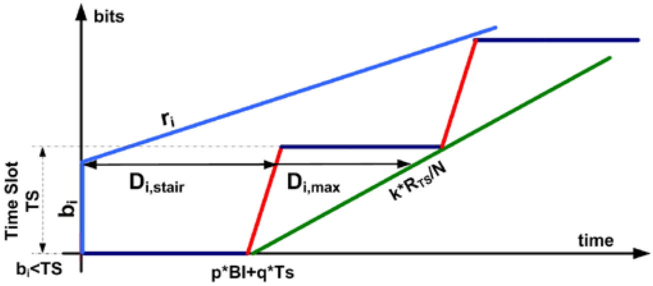


Fig. 9 Delay bound analysis of an implicit GTS allocation

Fig. 9):

$$D_{i,\mathrm{stair}} = \frac{b_i}{C} + (p \cdot BI + q \cdot Ts), \tag{23}$$

where C denotes the data rate equal to 250 kbps.

5.2 Numerical evaluation

We propose to illustrate the advantage of i-GAME in improving the bandwidth utilization efficiency as compared to the explicit GTS allocation mechanism. Consider a set of 14 flows F_i, $\forall 1 \le i \le 14$ with the arrival rates as presented in Table 1.

Since the guaranteed delay bound typically depends on the burst size, we assume (without loss of generality) that all flows have the same burst size $b_i = 200$ bits, $\forall 1 \le i \le 14$.

We consider a PAN with the same parameters as defined in Sect. 4.4 ($BO = SO = 0$).

Consider also the following three cases:

1. Explicit GTS allocations for 7 flows (F1 to F7)
2. Implicit GTS allocations for 7 flows (F1 to F7)
3. Implicit GTS allocations for 14 flows (F1 to F14)

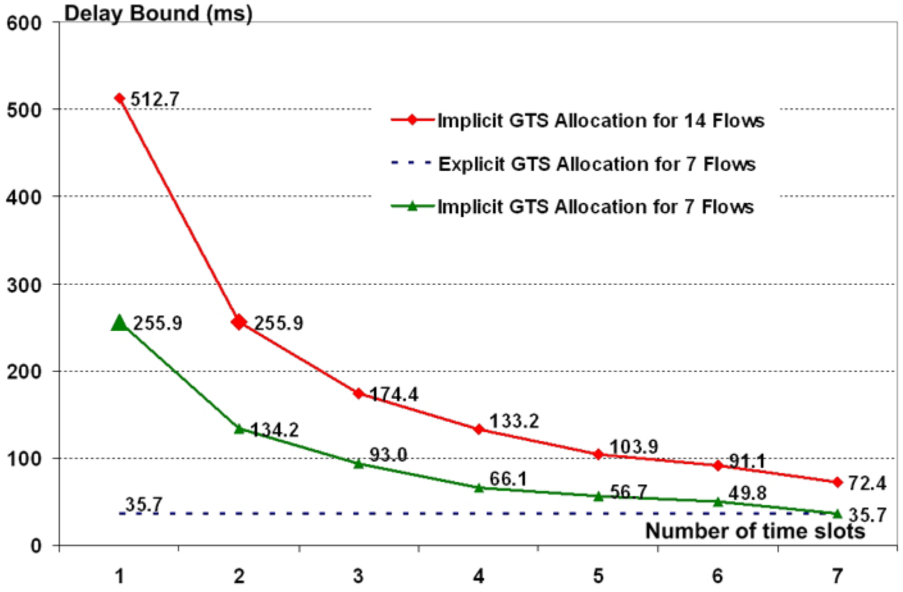
Note that condition (C1) is satisfied in all cases ($\sum_{i=1}^{7} r_i = 6.25$ kbps, $\sum_{i=1}^{14} r_i = 9.1$ kbps). Figures 10a and 10b present the bandwidth utilization and the guaranteed delay bound, respectively, as a function of the number of allocated time slots for the implicit GTS allocations. The bandwidth utilization of the explicit allocation (9.5%) is obtained from (11) and represented by a dotted line for a comparison purpose. The bandwidth utilization of implicit allocations is obtained from (16).

It can be understood from Fig. 10a that the i-GAME approach significantly improves the bandwidth utilization compared to the explicit allocation. However, the degree of improvement depends on the delay requirements of the flows as it can be observed in Fig. 10b.

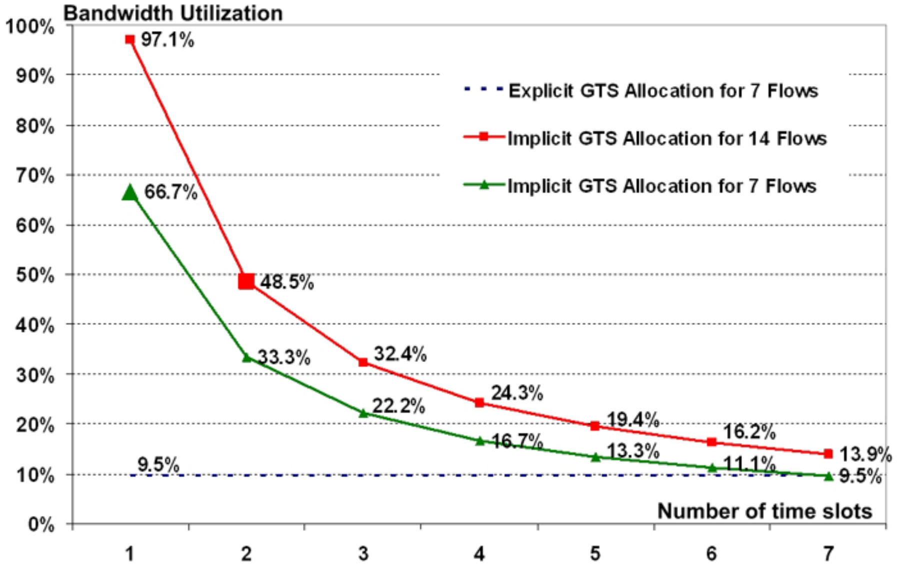
For example, assume that all flows have a delay requirement $D_i = 300$ ms. It is possible to meet this requirement for the seven flows (F1 to F7) with only one allocated time slot ($D_{i,\max} = 255.9$ ms), since $r_i \le R_{TS}/7$, $\forall 1 \le i \le 7$. In this case, the bandwidth utilization is 66.7%, which is much higher than 9.5% in case of the

Table 1 Arrival rates of the flows F1–F14

Flow	Arrival rates r_i (kbits/sec)
F1, F12	0.5
F2, F5, F6	1
F3, F4, F8	1.25
F7, F9, F11	0.25
F10	0.1
F13	0.3
F14	0.2



a Bandwidth utilization improvement with i-GAME



b Delay bounds guaranteed by the i-GAME approach

Fig. 10 Comparative evaluation of i-GAME and the explicit GTS allocation

explicit GTS allocation. It is also possible to meet this delay requirement for the fourteen flows with only two allocated time slots resulting in a utilization of 48.5%. In this case, all the flows take advantage of a guaranteed service with only two allocated time slots, which is not possible using the explicit GTS allocation mechanism.

Moreover, by using the implicit GTS allocation, the length of the CFP is significantly reduced, thus increasing the CAP period.

6 i-GAME implementation approach

6.1 The i-GAME admission control algorithm

This section presents an admission control algorithm for the implicit GTS allocation mechanism (i-GAME) presented in Sects. 3 and 4. We define the admission control algorithm in case of $r_i \leq R_{TS}, \forall i = 1, \ldots, N$ under round robin scheduling (results of Sect. 5). We assume that flows with $r_i \geq R_{TS}$ explicitly request a number of time slots based on their arrival rates, using (12).

Algorithm 1 presents the i-GAME *management algorithm* for implicit GTS allocations under round robin scheduling.

Algorithm 2 presents the *admission control function* used to decide whether to accept or not a node requesting an implicit allocation of a GTS based on its $F_{\text{spec}} = (b, r, D)$.

Algorithm 1. i-GAME Management Algorithm

```
1   type Flow = $(id, b, r, D)$ //traffic specification and delay requirement
2   type FlowSetType = ($F_i$, where $F_i$ requests a time slot in the CFP)
3   int $N = 0$; // the number of flow sharing a GTS
4   int $k = 1$; // the number of shared time slot
5   FlowSetType FlowSet; Flow $F$;
6   On (arrival of a new flow $F$) do {
7       $N = N + 1$;
8         if (admission_control ($k, N$, FlowSet, $F$) == false) {
9             if ($k == 7$) { //the maximum number of GTSs is reached
10                reject_request($F$);
11                $N = N - 1$; break;
12            }
13            else { // $k < 7$
14                $k = k + 1$; //increase the length of the CFP
15                goto line 8;
16            }
17        }
18        else {
19           accept_request($F$); //accept the new flow to share the GTS
20           FlowSet_Add(FlowSet, $F$); //add the new flow to the GTSset
21        }
```

Fig. 11 The i-GAME management algorithm

Algorithm 2. i-GAME Admission Control Function

```
$R_{TS}$ = guaranteed bandwidth by one time slot
$Ts$ = time slot duration
boolean admission_control (int $k$, int $N$, FlowSetType FlowSet, Flow $F$)
{
   boolean adm_crt = true;
   if ($k$<=$N$) {
         $p = \mathbf{ceil}(N/k)$;
         $q = N - p * k - 1$;
         for (int $i = 1, i{+}{+}; i <= N$)
            if (($Di < ((bi/(k * R_{TS}/N)) + (p * Bi - q * Ts)))$ or
                     $(ri > k * R_{TS}/N)$) adm_crt = false;
   } else //the case ($k > N$) is considered as explicit allocation
         adm_crt = false;
}
```

Fig. 12 The i-GAME admission control function

GTS management algorithm When a new implicit GTS allocation request initiated by a flow $F = (b, r, D)$ is received by the PAN coordinator, the admission control algorithm increments N, i.e. the number of flows sharing the same GTS. Then, the admission control function is called taking as inputs the number of allocated time slots (k), the number of flows sharing this GTS of k time slots (N), the set of flows sharing the GTS (*FlowSet*), and the new flow F requesting the GTS allocation.

If the admission control function returns *false*, then the PAN coordinator tries to extend the CFP length by adding a new time slot, if the maximum length of seven time slots has not been reached (since, in this case, each node allocates at most one time slot in a superframe); otherwise the new request is rejected. If a new time slot can be added to the CFP, then k is incremented by one and the admission control algorithm is called again with the new k value.

If the admission control returns *true*, the request of the new flow F will be accepted and the latter is added to the *FlowSet* list.

The admission control function This function returns a Boolean value stating whether to accept or not the new flow requesting a GTS. As we have mentioned before, we assume that flows requesting implicit allocation satisfies $r_i \le R_{TS}$, $\forall i = 1, \ldots, N$ (for the validity of (16) to (22)). This decision is based on the shared GTS length (k), the number of flows sharing the GTS (N), the specification of the new flow F and the existing flows in the *FlowSet*. The *adm_crt* flag is set to *true* at the start of the algorithm, and will be set to *false* if the delay requirement cannot be met or if the guaranteed bandwidth is higher than the arrival rate. The delay requirement of each flow will be compared to the guaranteed delay expressed in (22), which is shown to be valid for both cases $k < N$ and $k = N$. Actually, the case $k > N$ is not considered in the i-GAME admission control function, since it is considered as an explicit GTS allocation request, as we have previously mentioned. Note that the delay bound condition in line 10 of Algorithm 2 can be evaluated using (23) to obtain more accurate delay bound, in the case of a burst size smaller than a time slot duration. This improvement resulting by applying this equation is illustrated in the experimental evaluation section.

6.2 i-GAME implementation guidelines

This section presents some practical considerations for the implementation of the i-GAME mechanism in IEEE 802.15.4. An interesting feature of i-GAME is that its implementation only requires minor add-ons to the standard protocol, i.e. it does not impose any changes to the existing protocol.

The idea consists in using the reserved 6th bit in the GTS characteristics frame, embedded in a GTS allocation request command field (compare Figs. 13 and 3). This bit is referred to as *Allocation Type*.

The *Allocation Type* bit set to 0 corresponds to an explicit GTS allocation. In this case, the allocation process will follow the standard recommendations. If it is set to 1, it refers to the i-GAME implicit allocation mechanism proposed in this paper. In this case, to keep the IEEE 802.15.4 with no changes, the flow specification information $F_{\text{spec}} = (b, r, D)$ should be embedded in the higher layer packets, as presented in Fig. 14.

The admission control algorithm should be implemented at a higher layer (e.g. Network Layer) and should return the decision to the MAC sublayer (Fig. 15).

Hence, upon reception of an implicit GTS allocation request (*Allocation Type* $=$ 1), the MAC sublayer of the PAN coordinator should forward the Flow Specification Field (shown in Fig. 14) to the higher layer for processing by the admission control module. The *burst size* and the *arrival rate* fields should be expressed by four bits each (16 classes for each field). The *Delay Requirement* field is expressed by five bits (32 classes). Using this frame format, the PAN coordinator should define a fixed range for each value (class) of the corresponding field. These patterns should be known in advance by all nodes associated to the PAN before initiating an implicit allocation. The specification of these classes and ranges is out of the scope of this paper.

When the flow specification is received by the admission control module, it evaluates the acceptance of the new flow based on Algorithm 2. The decision should be notified to the MAC sublayer through the service access point. In case of acceptance,

bits	0-3	4	5	6	7
	GTS Length	GTS Direction	Characteristics Type	Allocation Type	Reserved

Fig. 13 The GTS characteristics extension field format for implicit request allocation

Fig. 14 The flow specification field format for i-GAME

bits	0-3	4-7	8-12	12-15
	Burst size	Arrival Rate	Delay Requirement	Reserved

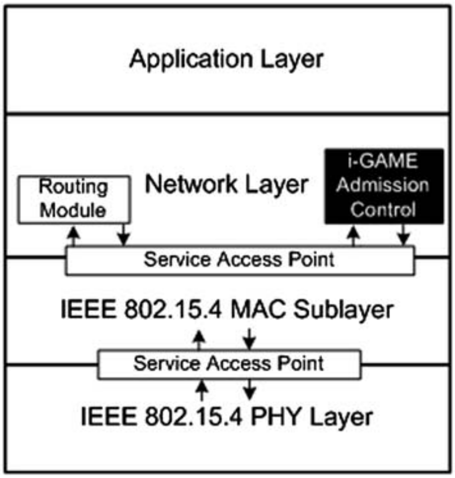


Fig. 15 The protocol layer architecture for i-GAME

the MAC sublayer allocates the time slots in the CFP in round robin order to all accepted nodes. For that purpose, the MAC sublayer should establish a certain order to allocate the time slots according to round robin scheduling. Each beacon frame of a new beacon interval should indicate which nodes are allowed to use the GTS in the current superframe, with respect to the established order.

7 Experimental evaluation

In this experimental evaluation section, we demonstrate the practical feasibility of the i-GAME approach in a real system implementation and we validate the analytical results of this paper.

7.1 The implementation platform

In the context of the ART-WiSe framework, we have developed a complete implementation of the IEEE 802.15.4 protocol using NesC (Gay et al. 2003), an extension to the C programming language for embedded systems, under the TinyOS (Hill 2003) operating system. This implementation is running in MICAz motes (Crossbow 2004), a sensor network platform offering a low-power microcontroller of 128 kbytes of program flash memory and equipped with a CC2420 transceiver (Chipcon 2004) for radio communications. The CC2420 radio transceiver is compliant with the IEEE 802.15.4 physical layer specification and operates at the 2.4 GHz frequency band, with a data rate of 250 kbps. Nonetheless, the IEEE 802.15.4 MAC layer is not provided in MICAz motes, which triggers the need for its implementation. The compiled code of the protocol (Physical, MAC and Network (NWK) layers) occupies around 24 kilobytes of code memory and 3 kilobytes of data memory. Figures 16 presents the architecture of our IEEE 802.15.4 implementation. The gray modules in the figure represent the TinyOS interfaces and modules that we have implemented. The hardware drivers of the CC2420 radio transceiver are already provided by TinyOS. Note

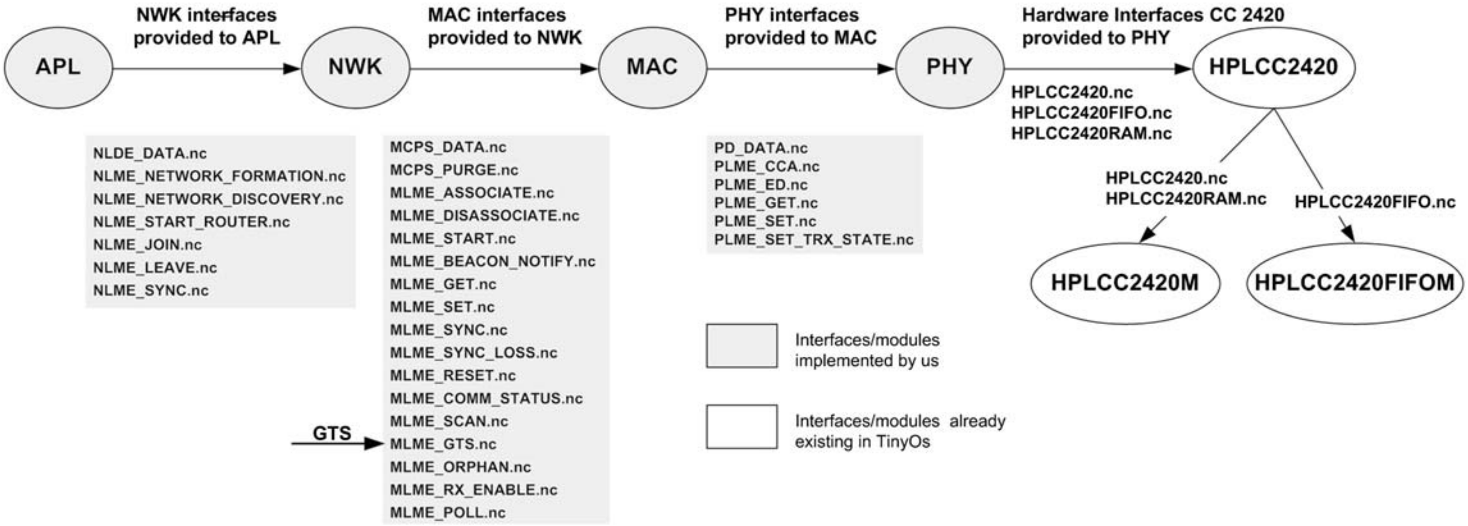


Fig. 16 The protocol stack software structure in TinyOS

that a part of the Zigbee Network layer has been implemented for supporting Zigbee cluster-tree networks.

The explicit GTS allocation mechanism has been implemented in the MAC layer module as an interface between the MAC and the Network layers, according to the IEEE 802.15.4 standard specification (IEEE-TG15.4 2003). We have also implemented the i-GAME admission control mechanism in the Network layer of a PAN Coordinator, based on the implementation guidelines provided in Sect. 6 with only few add-ons to the MAC layer. The i-GAME mechanism implementation does not affect the non-coordinator nodes since they only require setting the *Allocation Type* bit in the GTS request to the appropriate value: 1—*implicit*, 0—*explicit*. A detailed standard-like description of the interfaces added to the Network layer and the enhancements to the MAC layer for supporting the i-GAME mechanism is presented in Cunha et al. (2006).

A first constraint in the implementation of the beacon-enabled mode is related to the TinyOS management of the hardware timer of the MICAz motes, which prevents from having the exact durations of the beacon interval, the superframe duration and the time slot *in milliseconds*, as specified by the standard. The major problem is that the MICAz mote hardware timer operates at a frequency of 7.3728 MHz. In order to achieve a clock tick granularity that best fits our requirements (i.e. approximating the hardware clock tick with the theoretical duration of one symbol (4 bits) in IEEE 802.15.4), we have divided the clock frequency by 256 (scale of 4) resulting in frequency of 28.8 kHz, which approximately corresponds to 34,72 µs. This clock tick value approximately corresponds to the duration of two symbols (8 bits), which is theoretically equal to 32 µs according to the standard specification (IEEE-TG15.4 2003). This inaccuracy, resulting from the hardware timer constraint, leads to a cumulative effect on the discrepancy between the theoretical and experimental values of the beacon interval, the superframe duration and the time slot duration (in milliseconds) for high superframe and beacon orders. For instance, the superframe duration with $SO = 3$ is equal to 7680 symbols, which theoretically corresponds to 122.88 ms, but experimentally corresponds to 133.36 ms, based on the MICAz clock granularity.

Secondly, achieving highly accurate synchronization between the PAN Coordinator and the other nodes with small superframe orders, namely $SO = \{0, 1, 2\}$, is very challenging. The reason is that the time slot duration corresponding to these superframe orders is too small (less than 4 ms) to tolerate an inaccuracy of hundreds of micro seconds in the GTS parts. In fact, such an inaccuracy is due to the beacon processing procedure, which is time consuming and unpredictable. Even if we consider the time reference as the first bit detected of the beacon frame (by reading the start of frame delimiter (SFD) value), the timer evolution in the node is not very accurate since some parts of the beacon frame processing function are atomic, thus leading to some unpredictable loss of synchronization accuracy of hundred of microseconds during this processing time. For high superframe orders ($SO > 2$), this inaccuracy has a lower impact on the synchronization of the nodes with their PAN Coordinator. For that reason, we consider in this paper an experimental test bed with $BO = SO = 3$.

Concerning the implementation of the proposed algorithms (Figs. 11 and 12), the admission control function is a time consuming procedure due to the operations involved in the verification of the acceptance of a new flow. Therefore, this function was

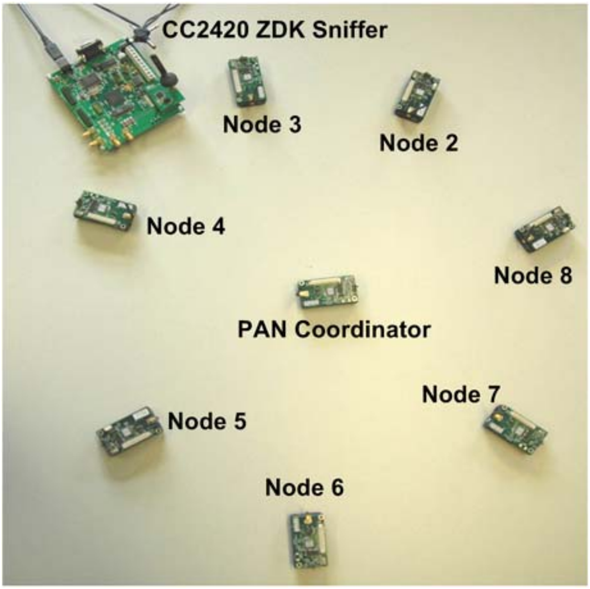


Fig. 17 Experimental testbed—8 MICAz motes including one PAN coordinator

implemented in a non-atomic way, such that it does not jeopardize the global behavior of the device by allowing its preemption by asynchronous events (e.g. hardware interrupts and timers).

We are working towards optimizing the beacon processing function in order to achieve better synchronization.

7.2 Experimental test bed

We set up an IEEE 802.15.4 WPAN in a beacon-enabled mode with one PAN Coordinator and 7 nodes distributed within the transmission range of the PAN Coordinator (Fig. 17). The superframe structure is configured with a beacon order $BO = 3$ and a superframe order $SO = 3$ for the reasons explained in the previous section. Each node runs an application that generates periodic traffic with a period $P = 200$ ms and a frame size $L = 120$ bits (104 bits of MAC header + 16 bits of data payload). The arrival curve $\alpha(t) = b + r \cdot t$ corresponding to this periodic data flow is defined by the average rate $r = L/P = 0.6$ kbps and the burst size $b = L = 120$ bits (Koubâa and Song 2004). In this scenario, we have experimentally evaluated the bandwidth guaranteed by one time slot, which was found to be equal to $R_{TS} = 2.70$ kbps. This small value is due to the acknowledgment overhead in each data frame transmission.

Since in Sect. 6 we have defined 16 classes to map the *burst size* and the *average rate*, and 32 classes to map the *delay bound*, we have configured the mapping table in the PAN Coordinator and in the nodes as presented in Table 2.

Hence, based on the specification of the Flow Specification frame format (Fig. 14), a node using the i-GAME mechanism for requesting an implicit GTS allocation should configure his request by choosing the corresponding classes. Thus, assuming for instance a delay bound of 300 ms, nodes that send implicit GTS allocations configure their Flow Specification Field to the value $(0 \times 1, 0 \times 0, 0 \times 0)$,

Table 2 Class mapping configuration

Class	Burst Size (bits)	Arrival Rate (kbps)	Delay Bound (ms)
0×0	80	0.6	300
0×1	120	1.2	500
0×2	160	2.4	700
0×3	200	4.8	900
Default	1016	9.6	2000

which will be interpreted by the PAN Coordinator as the following flow specification $F_{\text{spec},i} = (b_i \leq 120 \text{ bit}, r_i \leq 0.6 \text{ kbps}, D_i \leq 300 \text{ ms})$.

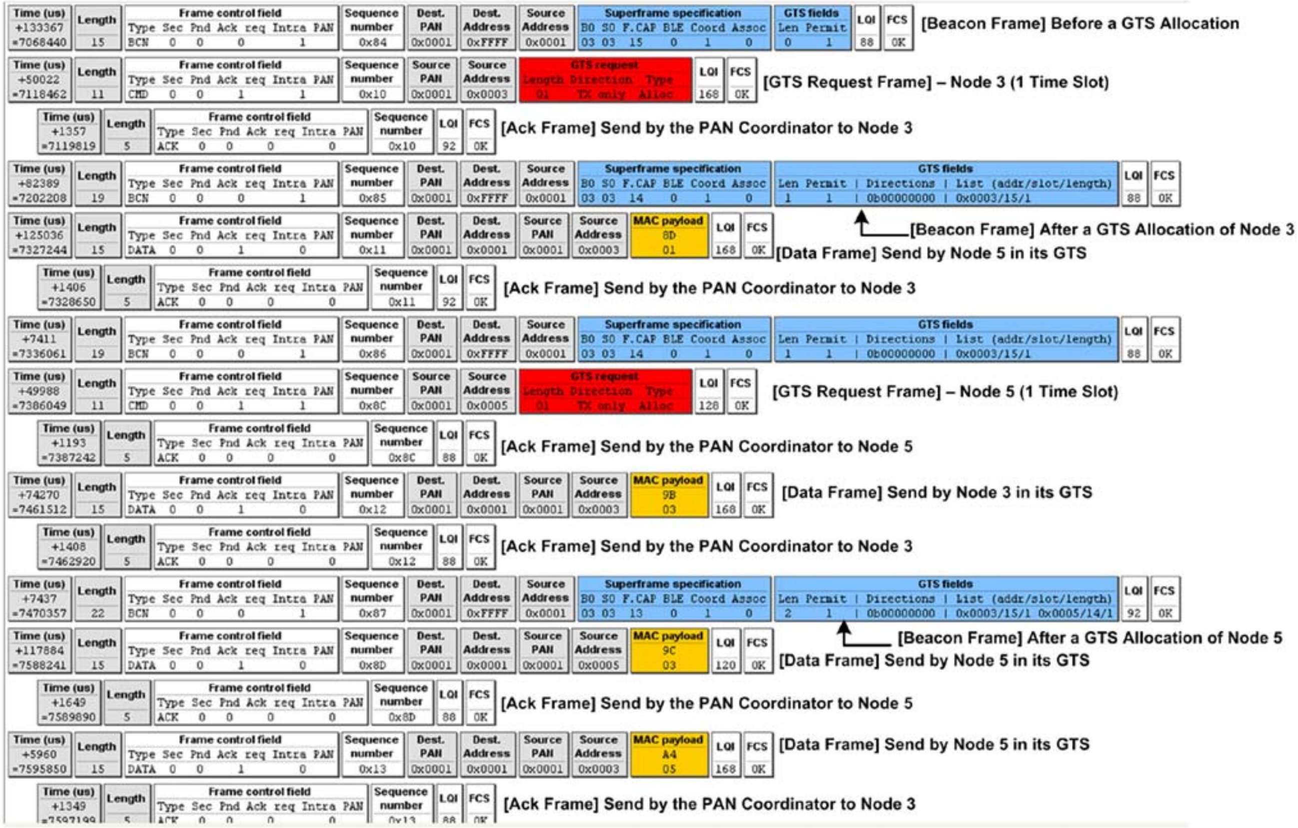
Note that the configuration mapping table will depend on the application requirements and should be adapted either manually by the designer or automatically according to a user-defined specification. Nevertheless, note that this table will apply to all nodes in the WPAN. Since the burst size $b = 120$ bits is lower than the duration (in bits) of one time slot, we have used (23) to evaluate the delay bound in the admission control function of the PAN Coordinator.

In what follows, we report the observations and the results of explicit and implicit GTS allocations corresponding to this network scenario. We used the CC2420 ZDK Development Kit (Chipcon 2005) and its sniffer application to intercept and visualize the traffic generated by the nodes in the WPAN. The MAC address of the PAN Coordinator is set to 0×1 and the other MAC addresses are set to the indices of the nodes (refer to Fig. 17).

7.3 Experimental analysis

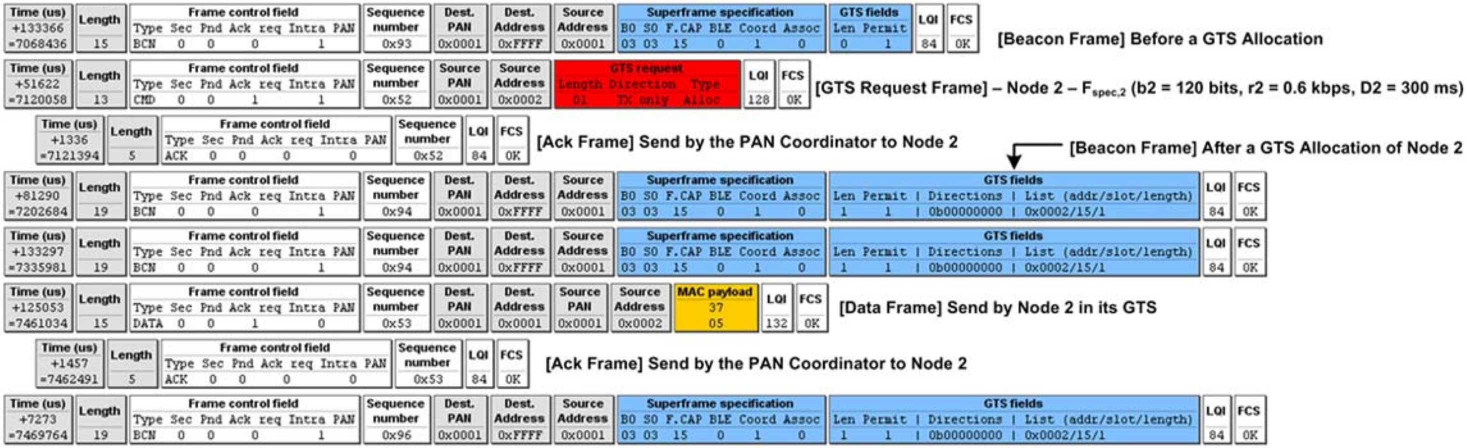
Figure 18 shows a sequence of two explicit GTS allocations made by two nodes. In this example, observe that one time slot per node has been allocated. We run the network for a significant time span to compute the maximum delay experienced by the flows allocating GTSs. Each time a frame experiences a delay higher than all the previous delays, this value is updated and printed in the MAC payload of each data packet as it can be seen in Fig. 18. Note that the delays observed in this figure do not correspond to the final measured maximum delays since it only presents the first packet that has been sent. The effective experimental maximum delay bound that we have measured is equal to 124.59 ms, which is very close to the theoretical delay bound of 125.51 ms, when applying the delay bound analysis of an explicit GTS allocation (Koubâa et al. 2006). The good accuracy of the theoretical bound is mainly a result of the good approximation of the arrival curve of the periodic traffic and the stair service curve of the GTS allocation mechanism.

Figure 19 presents the sequence of implicit GTS allocations using the i-GAME mechanism. In Fig. 19a, a first request made by node 2 is shown. With one flow there is no difference to the explicit allocation (only in the frame format sent by the node). Unfortunately, the CC2420 ZDK sniffer does not show the payload of the implicit GTS allocation since it is build to show just the standard packet format field. However, we have verified by looking at the sequence of bits sent (in another visualization mode) that the required fields of the flow specification (Fig. 14) are



```
Time (us) | Length | Frame control field             | Sequence | Dest.  | Dest.   | Source  | Superframe specification    | GTS fields | LQI | FCS
+133367   |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc | Len Permit |     |
=7068440  | 15     | BCN  0   0       0         1    | 0x84     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0     | 0   1      | 88  | OK
[Beacon Frame] Before a GTS Allocation

Time (us) | Length | Frame control field             | Sequence | Source | Source  | GTS request                 | LQI | FCS
+50022    |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | Length Direction Type       |     |
=7118462  | 11     | CMD  0   0       1         1    | 0x10     | 0x0001 | 0x0003  | 01     TX only   Alloc      | 168 | OK
[GTS Request Frame] – Node 3 (1 Time Slot)

    Time (us) | Length | Frame control field             | Sequence | LQI | FCS
    +1357     |        | Type Sec Pnd Ack req Intra PAN  | number   |     |
    =7119819  | 5      | ACK  0   0       0         0    | 0x10     | 92  | OK
    [Ack Frame] Send by the PAN Coordinator to Node 3

Time (us) | Length | Frame control field             | Sequence | Dest.  | Dest.   | Source  | Superframe specification    | GTS fields                                           | LQI | FCS
+82389    |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc | Len Permit | Directions | List (addr/slot/length) |     |
=7202208  | 19     | BCN  0   0       0         1    | 0x85     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 14    0   1     0     | 1   1      | 0b00000000 | 0x0003/15/1             | 88  | OK
                                                                                                                        [Beacon Frame] After a GTS Allocation of Node 3

Time (us) | Length | Frame control field             | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+125036   |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | PAN    | Address | 8D          |     |
=7327244  | 15     | DATA 0   0       1         0    | 0x11     | 0x0001 | 0x0001  | 0x0001 | 0x0003  | 01          | 168 | OK
[Data Frame] Send by Node 5 in its GTS

    Time (us) | Length | Frame control field             | Sequence | LQI | FCS
    +1406     |        | Type Sec Pnd Ack req Intra PAN  | number   |     |
    =7328650  | 5      | ACK  0   0       0         0    | 0x11     | 92  | OK
    [Ack Frame] Send by the PAN Coordinator to Node 3

Time (us) | Length | Frame control field             | Sequence | Dest.  | Dest.   | Source  | Superframe specification    | GTS fields                                           | LQI | FCS
+7411     |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc | Len Permit | Directions | List (addr/slot/length) |     |
=7336061  | 19     | BCN  0   0       0         1    | 0x86     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 14    0   1     0     | 1   1      | 0b00000000 | 0x0003/15/1             | 88  | OK

Time (us) | Length | Frame control field             | Sequence | Source | Source  | GTS request                 | LQI | FCS
+49988    |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | Length Direction Type       |     |
=7386049  | 11     | CMD  0   0       1         1    | 0x8C     | 0x0001 | 0x0005  | 01     TX only   Alloc      | 128 | OK
[GTS Request Frame] – Node 5 (1 Time Slot)

    Time (us) | Length | Frame control field             | Sequence | LQI | FCS
    +1193     |        | Type Sec Pnd Ack req Intra PAN  | number   |     |
    =7387242  | 5      | ACK  0   0       0         0    | 0x8C     | 88  | OK
    [Ack Frame] Send by the PAN Coordinator to Node 5

Time (us) | Length | Frame control field             | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+74270    |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | PAN    | Address | 9B          |     |
=7461512  | 15     | DATA 0   0       1         0    | 0x12     | 0x0001 | 0x0001  | 0x0001 | 0x0003  | 03          | 168 | OK
[Data Frame] Send by Node 3 in its GTS

    Time (us) | Length | Frame control field             | Sequence | LQI | FCS
    +1408     |        | Type Sec Pnd Ack req Intra PAN  | number   |     |
    =7462920  | 5      | ACK  0   0       0         0    | 0x12     | 88  | OK
    [Ack Frame] Send by the PAN Coordinator to Node 3

Time (us) | Length | Frame control field             | Sequence | Dest.  | Dest.   | Source  | Superframe specification    | GTS fields                                                       | LQI | FCS
+7437     |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc | Len Permit | Directions | List (addr/slot/length)             |     |
=7470357  | 22     | BCN  0   0       0         1    | 0x87     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 13    0   1     0     | 2   1      | 0b00000000 | 0x0003/15/1 0x0005/14/1             | 92  | OK
                                                                                                                  [Beacon Frame] After a GTS Allocation of Node 5

Time (us) | Length | Frame control field             | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+117884   |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | PAN    | Address | 9C          |     |
=7588241  | 15     | DATA 0   0       1         0    | 0x8D     | 0x0001 | 0x0001  | 0x0001 | 0x0005  | 03          | 120 | OK
[Data Frame] Send by Node 5 in its GTS

    Time (us) | Length | Frame control field             | Sequence | LQI | FCS
    +1649     |        | Type Sec Pnd Ack req Intra PAN  | number   |     |
    =7589890  | 5      | ACK  0   0       0         0    | 0x8D     | 88  | OK
    [Ack Frame] Send by the PAN Coordinator to Node 5

Time (us) | Length | Frame control field             | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+5960     |        | Type Sec Pnd Ack req Intra PAN  | number   | PAN    | Address | PAN    | Address | A4          |     |
=7595850  | 15     | DATA 0   0       1         0    | 0x13     | 0x0001 | 0x0001  | 0x0001 | 0x0003  | 05          | 168 | OK
[Data Frame] Send by Node 5 in its GTS

    Time (us) | Length | Frame control field             | Sequence | LQI | FCS
    +1349     |        | Type Sec Pnd Ack req Intra PAN  | number   |     |
    =7597199  | 5      | ACK  0   0       0         0    | 0x13     | 88  | OK
    [Ack Frame] Send by the PAN Coordinator to Node 3
```

Fig. 18 A scenario of an explicit GTS allocation of two nodes



```
Time (us) | Length | Frame control field                  | Sequence | Dest.  | Dest.   | Source  | Superframe specification        | GTS fields | LQI | FCS
+133366   |        | Type Sec Pnd Ack req Intra PAN       | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc     | Len Permit |     |
=7068436  | 15     | BCN  0   0       0       1           | 0x93     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0         | 0   1      | 84  | OK
```

[Beacon Frame] Before a GTS Allocation

```
Time (us) | Length | Frame control field                  | Sequence | Source | Source  | GTS request                  | LQI | FCS
+51622    |        | Type Sec Pnd Ack req Intra PAN       | number   | PAN    | Address | Length Direction Type        |     |
=7120058  | 13     | CMD  0   0       1       1           | 0x52     | 0x0001 | 0x0002  | 01     TX only   Alloc       | 128 | OK
```

[GTS Request Frame] – Node 2 – $F_{spec,2}$ (b2 = 120 bits, r2 = 0.6 kbps, D2 = 300 ms)

```
Time (us) | Length | Frame control field                  | Sequence | LQI | FCS
+1336     |        | Type Sec Pnd Ack req Intra PAN       | number   |     |
=7121394  | 5      | ACK  0   0       0       0           | 0x52     | 84  | OK
```

[Ack Frame] Send by the PAN Coordinator to Node 2

[Beacon Frame] After a GTS Allocation of Node 2

```
Time (us) | Length | Frame control field                  | Sequence | Dest.  | Dest.   | Source  | Superframe specification        | GTS fields                                                | LQI | FCS
+81290    |        | Type Sec Pnd Ack req Intra PAN       | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc     | Len Permit | Directions | List (addr/slot/length)       |     |
=7202684  | 19     | BCN  0   0       0       1           | 0x94     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0         | 1   1      | 0b00000000 | 0x0002/15/1                   | 84  | OK
```

```
Time (us) | Length | Frame control field                  | Sequence | Dest.  | Dest.   | Source  | Superframe specification        | GTS fields                                                | LQI | FCS
+133297   |        | Type Sec Pnd Ack req Intra PAN       | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc     | Len Permit | Directions | List (addr/slot/length)       |     |
=7335981  | 19     | BCN  0   0       0       1           | 0x94     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0         | 1   1      | 0b00000000 | 0x0002/15/1                   | 84  | OK
```

```
Time (us) | Length | Frame control field                  | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+125053   |        | Type Sec Pnd Ack req Intra PAN       | number   | PAN    | Address | PAN    | Address | 37          |     |
=7461034  | 15     | DATA 0   0       1       0           | 0x53     | 0x0001 | 0x0001  | 0x0001 | 0x0002  | 05          | 132 | OK
```

[Data Frame] Send by Node 2 in its GTS

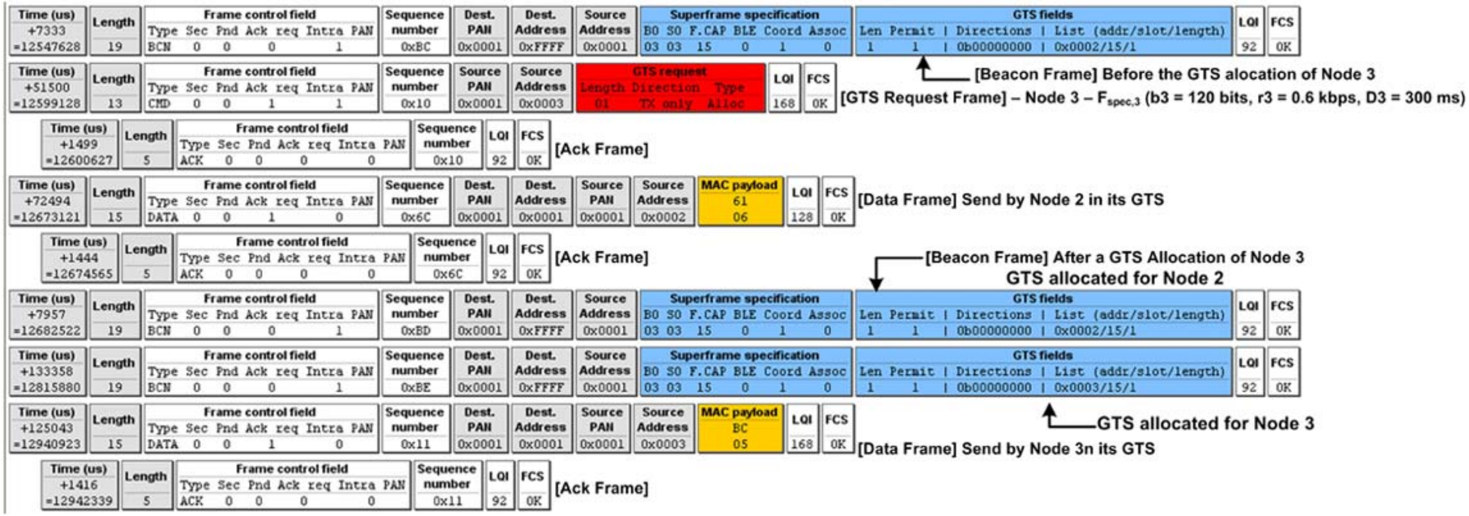
```
Time (us) | Length | Frame control field                  | Sequence | LQI | FCS
+1457     |        | Type Sec Pnd Ack req Intra PAN       | number   |     |
=7462491  | 5      | ACK  0   0       0       0           | 0x53     | 84  | OK
```

[Ack Frame] Send by the PAN Coordinator to Node 2

```
Time (us) | Length | Frame control field                  | Sequence | Dest.  | Dest.   | Source  | Superframe specification        | GTS fields                                                | LQI | FCS
+7273     |        | Type Sec Pnd Ack req Intra PAN       | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc     | Len Permit | Directions | List (addr/slot/length)       |     |
=7469764  | 19     | BCN  0   0       0       1           | 0x96     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0         | 1   1      | 0b00000000 | 0x0002/15/1                   | 84  | OK
```

a The first implicit allocation request, sent by node 2

Fig. 19 A scenario of an implicit GTS allocation of three nodes using i-GAME



```
Time (us) | Length | Frame control field                  | Sequence number | Dest. PAN | Dest. Address | Source Address | Superframe specification           | GTS fields                                                     | LQI | FCS
+7333     | 19     | Type Sec Pnd Ack req Intra PAN       | 0xBC            | 0x0001    | 0xFFFF        | 0x0001         | BO SO F.CAP BLE Coord Assoc        | Len Permit | Directions | List (addr/slot/length)            | 92  | OK
=12547628 |        | BCN  0   0   0       1               |                 |           |               |                | 03 03 15    0   1     0            | 1   1      | 0b00000000 | 0x0002/15/1                        |     |

Time (us) | Length | Frame control field                  | Sequence number | Source PAN | Source Address | GTS request                  | LQI | FCS
+51500    | 13     | Type Sec Pnd Ack req Intra PAN       | 0x10            | 0x0001     | 0x0003         | Length Direction Type        | 168 | OK
=12599128 |        | CMD  0   0   1       1               |                 |            |                | 01     TX only   Alloc       |     |

Time (us) | Length | Frame control field                  | Sequence number | LQI | FCS
+1499     | 5      | Type Sec Pnd Ack req Intra PAN       | 0x10            | 92  | OK
=12600627 |        | ACK  0   0   0       0               |                 |     |

Time (us) | Length | Frame control field                  | Sequence number | Dest. PAN | Dest. Address | Source PAN | Source Address | MAC payload | LQI | FCS
+72494    | 15     | Type Sec Pnd Ack req Intra PAN       | 0x6C            | 0x0001    | 0x0001        | 0x0001     | 0x0002         | 61          | 128 | OK
=12673121 |        | DATA 0   0   1       0               |                 |           |               |            |                | 06          |     |

Time (us) | Length | Frame control field                  | Sequence number | LQI | FCS
+1444     | 5      | Type Sec Pnd Ack req Intra PAN       | 0x6C            | 92  | OK
=12674565 |        | ACK  0   0   0       0               |                 |     |

Time (us) | Length | Frame control field                  | Sequence number | Dest. PAN | Dest. Address | Source Address | Superframe specification           | GTS fields                                                     | LQI | FCS
+7957     | 19     | Type Sec Pnd Ack req Intra PAN       | 0xBD            | 0x0001    | 0xFFFF        | 0x0001         | BO SO F.CAP BLE Coord Assoc        | Len Permit | Directions | List (addr/slot/length)            | 92  | OK
=12682522 |        | BCN  0   0   0       1               |                 |           |               |                | 03 03 15    0   1     0            | 1   1      | 0b00000000 | 0x0002/15/1                        |     |

Time (us) | Length | Frame control field                  | Sequence number | Dest. PAN | Dest. Address | Source Address | Superframe specification           | GTS fields                                                     | LQI | FCS
+133358   | 19     | Type Sec Pnd Ack req Intra PAN       | 0xBE            | 0x0001    | 0xFFFF        | 0x0001         | BO SO F.CAP BLE Coord Assoc        | Len Permit | Directions | List (addr/slot/length)            | 92  | OK
=12815880 |        | BCN  0   0   0       1               |                 |           |               |                | 03 03 15    0   1     0            | 1   1      | 0b00000000 | 0x0003/15/1                        |     |

Time (us) | Length | Frame control field                  | Sequence number | Dest. PAN | Dest. Address | Source PAN | Source Address | MAC payload | LQI | FCS
+125043   | 15     | Type Sec Pnd Ack req Intra PAN       | 0x11            | 0x0001    | 0x0001        | 0x0001     | 0x0003         | BC          | 168 | OK
=12940923 |        | DATA 0   0   1       0               |                 |           |               |            |                | 05          |     |

Time (us) | Length | Frame control field                  | Sequence number | LQI | FCS
+1416     | 5      | Type Sec Pnd Ack req Intra PAN       | 0x11            | 92  | OK
=12942339 |        | ACK  0   0   0       0               |                 |     |
```

[Beacon Frame] Before the GTS alocation of Node 3

[GTS Request Frame] – Node 3 – $F_{spec,3}$ ($b3 = 120$ bits, $r3 = 0.6$ kbps, $D3 = 300$ ms)

[Ack Frame]

[Data Frame] Send by Node 2 in its GTS

[Ack Frame]

[Beacon Frame] After a GTS Allocation of Node 3
GTS allocated for Node 2

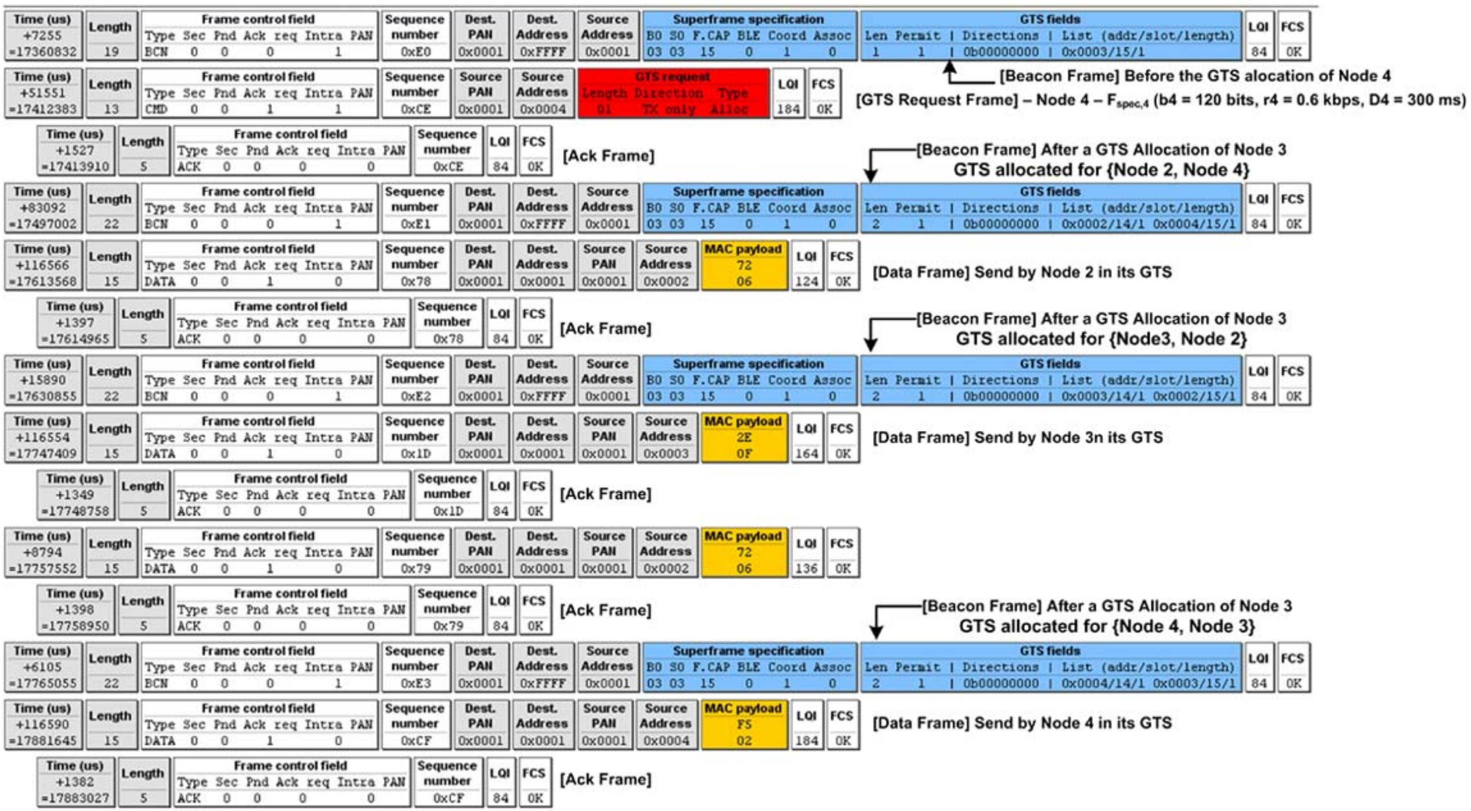
GTS allocated for Node 3

[Data Frame] Send by Node 3n its GTS

[Ack Frame]

b The second implicit allocation request, sent by node 3

Fig. 19 (*Continued*)



```
Time (us) | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source  | Superframe specification         | GTS fields                                                        | LQI | FCS
+7255     |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc      | Len Permit | Directions | List (addr/slot/length)              |     |
=17360832 | 19     | BCN  0   0   0       1             | 0xE0     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0          | 1   1      | 0b00000000 | 0x0003/15/1                          | 84  | OK
```

[Beacon Frame] Before the GTS alocation of Node 4

```
Time (us) | Length | Frame control field                | Sequence | Source | Source  | GTS request                   | LQI | FCS
+51551    |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Length Direction Type         |     |
=17412383 | 13     | CMD  0   0   1       1             | 0xCE     | 0x0001 | 0x0004  | 01     TX only   Alloc        | 184 | OK
```

[GTS Request Frame] – Node 4 – $F_{spec,4}$ (b4 = 120 bits, r4 = 0.6 kbps, D4 = 300 ms)

```
Time (us) | Length | Frame control field                | Sequence | LQI | FCS
+1527     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
=17413910 | 5      | ACK  0   0   0       0             | 0xCE     | 84  | OK
```

[Ack Frame]

[Beacon Frame] After a GTS Allocation of Node 3
GTS allocated for {Node 2, Node 4}

```
Time (us) | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source  | Superframe specification         | GTS fields                                                        | LQI | FCS
+83092    |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc      | Len Permit | Directions | List (addr/slot/length)              |     |
=17497002 | 22     | BCN  0   0   0       1             | 0xE1     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0          | 2   1      | 0b00000000 | 0x0002/14/1 0x0004/15/1              | 84  | OK
```

```
Time (us) | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+116566   |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | PAN    | Address | 72          |     |
=17613568 | 15     | DATA 0   0   1       0             | 0x78     | 0x0001 | 0x0001  | 0x0001 | 0x0002  | 06          | 124 | OK
```

[Data Frame] Send by Node 2 in its GTS

```
Time (us) | Length | Frame control field                | Sequence | LQI | FCS
+1397     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
=17614965 | 5      | ACK  0   0   0       0             | 0x78     | 84  | OK
```

[Ack Frame]

[Beacon Frame] After a GTS Allocation of Node 3
GTS allocated for {Node3, Node 2}

```
Time (us) | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source  | Superframe specification         | GTS fields                                                        | LQI | FCS
+15890    |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc      | Len Permit | Directions | List (addr/slot/length)              |     |
=17630855 | 22     | BCN  0   0   0       1             | 0xE2     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0          | 2   1      | 0b00000000 | 0x0003/14/1 0x0002/15/1              | 84  | OK
```

```
Time (us) | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+116554   |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | PAN    | Address | 2E          |     |
=17747409 | 15     | DATA 0   0   1       0             | 0x1D     | 0x0001 | 0x0001  | 0x0001 | 0x0003  | 0F          | 164 | OK
```

[Data Frame] Send by Node 3n its GTS

```
Time (us) | Length | Frame control field                | Sequence | LQI | FCS
+1349     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
=17748758 | 5      | ACK  0   0   0       0             | 0x1D     | 84  | OK
```

[Ack Frame]

```
Time (us) | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+8794     |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | PAN    | Address | 72          |     |
=17757552 | 15     | DATA 0   0   1       0             | 0x79     | 0x0001 | 0x0001  | 0x0001 | 0x0002  | 06          | 136 | OK
```

```
Time (us) | Length | Frame control field                | Sequence | LQI | FCS
+1398     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
=17758950 | 5      | ACK  0   0   0       0             | 0x79     | 84  | OK
```

[Ack Frame]

[Beacon Frame] After a GTS Allocation of Node 3
GTS allocated for {Node 4, Node 3}

```
Time (us) | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source  | Superframe specification         | GTS fields                                                        | LQI | FCS
+6105     |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc      | Len Permit | Directions | List (addr/slot/length)              |     |
=17765055 | 22     | BCN  0   0   0       1             | 0xE3     | 0x0001 | 0xFFFF  | 0x0001  | 03 03 15    0   1     0          | 2   1      | 0b00000000 | 0x0004/14/1 0x0003/15/1              | 84  | OK
```

```
Time (us) | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+116590   |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | PAN    | Address | F5          |     |
=17881645 | 15     | DATA 0   0   1       0             | 0xCF     | 0x0001 | 0x0001  | 0x0001 | 0x0004  | 02          | 184 | OK
```

[Data Frame] Send by Node 4 in its GTS

```
Time (us) | Length | Frame control field                | Sequence | LQI | FCS
+1382     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
=17883027 | 5      | ACK  0   0   0       0             | 0xCF     | 84  | OK
```

[Ack Frame]

c The third implicit allocation request, sent by node 4

Fig. 19 (*Continued*)

correctly configured. From Figs. 18 and 19, we observe that the length of an implicit GTS allocation frame (13 bytes, in Fig. 19) is two bytes more than the length of an explicit GTS allocation frame (11 bytes, in Fig. 18), which corresponds to the 16 additional bits for the flow specification field.

A second implicit GTS allocation request is performed by Node 3, as shown in Fig. 19b. Observe that contrarily to the scenario in Fig. 18, the PAN Coordinator still uses the same one-time slot GTS alternatively for both nodes (using round robin scheduling). This is because the admission control test was successful to accept Node 3 for sharing one time slot with Node 2 while their requirements are still satisfied. In fact, based on this configuration, the delay bound computed by the admission control mechanism is equal to 258.82 ms, which is lower than the delay requirement of 300 ms.

The sequence corresponding to a third implicit GTS allocation request made by Node 4 is presented in Fig. 19c. Observe that after this request the PAN Coordinator increments the number of allocated time slots. In the case of an allocation of one time slot by these three nodes, the delay bound returned by the admission control mechanism is equal to 391.16 ms, which is greater than the one required. When incrementing the GTS length by one, the delay bound guaranteed for the three nodes is then equal to 258.82 ms, which satisfies the 300 ms delay requirement. Observe that the round robin scheduling algorithm is well maintained by the PAN Coordinator in each new superframe.

To observe the impact of the delay requirement on the improvement of the GTS efficiency, we have run the experimental test bed for three additional scenarios, in which nodes choose the other delay classes (0×1, 0×2, and 0×3) for their delay requirements according to Table 2. The results are plotted in Fig. 20 and perfectly confirmed by the analytical formulations.

Observe that relaxing the delay bound of 7 nodes requesting GTS enables to save, in the case of 900 ms of delay requirement, up to 5 time slots ($\sim$42 ms) as compared to explicit allocation, while still satisfying the delay bounds. This (saved) time can be used by the contention-access period, thus improving the utilization of the network.

Figures 21 shows the comparison between the delay bound obtained by experiments and those theoretically obtained by (22) and (23). We have considered the scenario where nodes requesting implicit allocations have a 300 ms delay requirement. The equivalent number of time slots allocated is shown in Fig. 20. The delay bound based on (22) are computed using the experimentally evaluated bandwidth of a time slot $R_{TS} = 2.70$ kbps.

It can be observed that the experimental delay bounds perfectly match the theoretical stair delay bounds provided by (23). This is quite important to achieve better utilization of the GTS. Contrarily, (22) (linear curve) leads to more pessimistic bounds that prevents from maximizing the number of nodes sharing a GTS. In fact, if we had used (22) in the admission control mechanism of the PAN Coordinator, it could have resulted that two nodes could not share the same GTS (based on Fig. 20) since the evaluated delay bound is 347 ms, which is greater than the 300 ms delay requirement. However, the actual delay bound, as determined by (23), does not exceed 259 ms. This definitely shows the importance of avoiding a too pessimistic admission control algorithm in practice, especially with resource constrained networks, such as WSNs (Cunha et al. 2006).

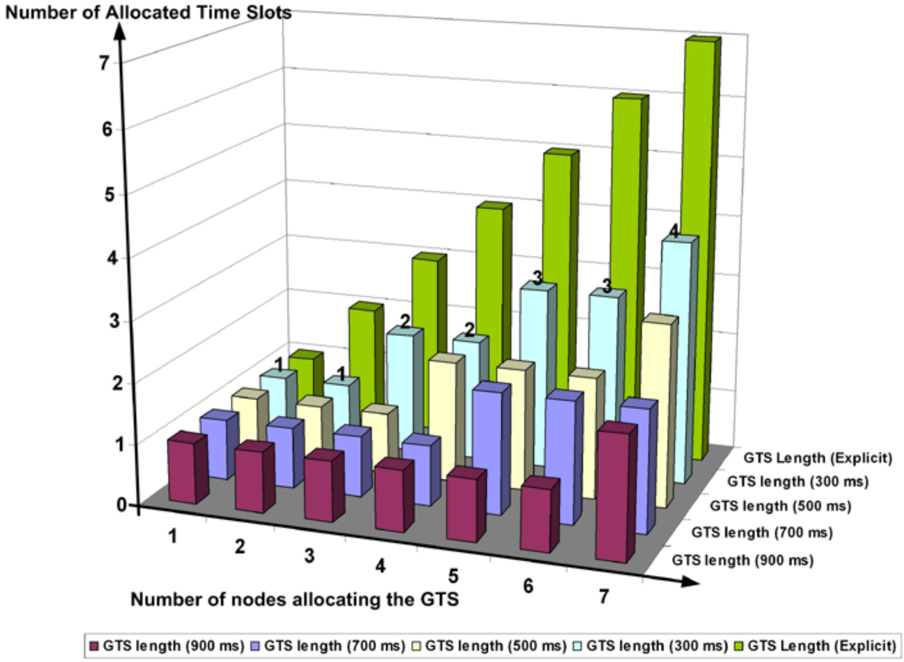


Fig. 20 Number of nodes allocating a GTS with i-GAME versus the GTS length

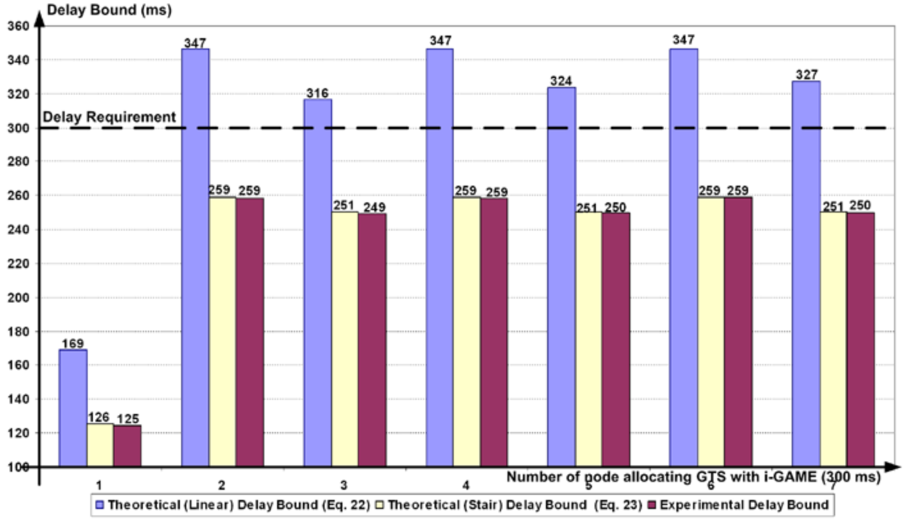


Fig. 21 Experimental delay bounds versus theoretical delay bounds

8 Conclusions

While the IEEE 802.15.4 protocol is a standard for low-rate wireless personal area networks, it still leaves room for potential improvements. This paper contributes on the definition of i-GAME, a new approach to allocate GTS in the IEEE 802.15.4 protocol for WSNs. This proposal is motivated by the bandwidth utilization inefficiency of the explicit GTS allocation mechanism supported by the IEEE 802.15.4 protocol for flows with low rates. i-GAME overcomes this problem by allowing to share the same GTS between multiple flows based on their traffic specifications and delay requirements. We have provided the schedulability analysis of i-GAME for computing the delay bound of a set of nodes sharing a given number of time slots. This theoretical analysis has been experimentally validated by a test bed that firstly demonstrates the practical feasibility of i-GAME, which has been implemented in a real platform, and secondly confirms the improvement resulting from using the implicit GTS allocation mechanism over the classical explicit GTS allocation in terms of bandwidth utilization.

We also showed that the implementation of i-GAME only requires minor add-ons to the IEEE 802.15.4 protocol and ensures backward compatibility with the standard, making our approach easily implementable in Commercial-Off-The-Shell (COTS) platforms. The i-GAME admission control mechanism is build on top of the MAC layer.

In this paper, we have considered the case of low-rate data flows, requiring at most one time slot GTS allocation. Even though this assumption is realistic, some WSN applications may operate at higher rates. Currently, we are working toward extending i-GAME for flows with different characteristics, envisaging the same objectives of simplicity and backward compatibility with the standard. To do so, we need to improve the scheduling policy instead of using round robin, which only requires changing the scheduling table, while keeping the admission control management procedure as proposed by i-GAME.

References

Cavalieri S, Monforte S, Corsaro A, Scapellato G (2003) Multicycle polling scheduling algorithms for fieldbus networks. J Real-Time Syst 25:157–185

Chipcon (2004) CC2420 transceiver datasheet, http://www.chipcon.com

Chipcon, (2005) User Manual Rev. 01—CC2420 ZDK Development Kit, http://www.chipcon.com

Crossbow (2004) MICAz datasheet, http://www.xbow.com

Cunha A, Koubâa A, Alves M (July 2006) Implementation of the i-GAME mechanism in IEEE 802.15.4 WPANs. IPP-HURRAY technical report, TR060702

Feng X, Mok A (2002) A model of hierarchical real-time virtual resources. In: IEEE Real Time System Symposium. IEEE Computer Society, Austin

Gay D, Levis P, von Behren R, Welsh M, Brewer E, Culler D (2003) The nesC language: a holistic approach to networked embedded systems. In: Proceedings of the programming language design and implementation

Hill J (2003) System architecture for wireless sensor networks. In: Computer Science Department. University of California, Berkeley

IEEE-TG15.4 (2003) Part 15.4: wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (LR-WPANs). IEEE standard for information technology

Koubâa A, Alves M, Tovar E (2005) A two-tiered architecture for real-time communications in large-scale wireless sensor networks: research challenges. In: 17th Euromicro conference on real-time systems (ECRTS'05), WiP session, Palma de Mallorca

Koubâa A, Alves M, Tovar E (2006) GTS allocation analysis in IEEE 802.15.4 for real-time wireless sensor networks. In: 14th international workshop on parallel and distributed real-time systems (WPDRTS 2006). IEEE, Rhodes Island

Koubâa A, Song YQ (2004) Evaluation and improvement of response time bounds for real-time applications under non-pre-emptive fixed priority scheduling. Int J Prod Res 42:2899–2913

Leboudec J-Y, Thiran P (2001) A theory of deterministic queuing systems for the Internet. Lecture notes in computer science (LNCS), vol 2050

Raja P, Noubir G (1993) Static and dynamic polling mechanisms for fieldbus networks. ACM SIGOPS Oper Syst Rev 27:34–45

Yoo S-E, Kim D, Pham M-L, Doh Y, Choi E, Huh J-D (2005) Scheduling support for guaranteed time services in IEEE 802.15.4 low rate WPAN. In: 11th IEEE international conference on embedded and real-time computing systems and applications (RTCSA'05). IEEE Computer Society, Hong Kong

Anis Koubâa was born in 1977 and is currently an Assistant Professor at Al-Imam Muhammad Ibn Saud University (Riyadh, Saudi Arabia) in the College of Computer Science and Information Systems and a Research Associate at the CISTER/IPP-HURRAY Research Group (Porto, Portugal). He is actively working on the design of real-time and reliable architectures for wireless sensor networks. He has driven the research efforts in the context of the ART-WiSe and OPEN-ZB frameworks that have contributed to the release of an open source implementation of the IEEE 802.15.4/ZigBee standard protocol stack. He received his Engineering degree in Telecommunications (2000) from the Higher School of Telecommunications in Tunis (Tunisia), MSc (2001) and PhD (2004) degrees in Computer Science from the National Polytechnic Institute of Lorraine (INPL) in Nancy (France) and associated to the INRIA/LORIA research laboratory. His main research activities focus on wireless ad-hoc and sensor networks, real-time distributed systems and networks, quality of service, WPANs/WLANs integration. His is actively participating as a reviewer or a program committee member in some reputed international journals, conferences and workshops dealing with real-time networks, quality of service, wireless communications and related issues.

Mário Alves was born in 1968 and has a Degree (1991), a MSc (1995) and a PhD (2003) in Electrical and Computer Engineering at the University of Porto. He is a Full Professor at the School of Engineering of the Polytechnic Institute of Porto (ISEP/IPP) and a researcher of the IPP-HURRAY/CISTER Research Unit. He has participated in several national and international projects (e.g. ESPRIT, IST), served in the reviewing, PC and organization of several world-reputed conferences (e.g. ECRTS) and has published tens of scientific paper at top-level conferences and journals in his expertise area. His PhD work addressed the design of real-time hybrid (wired/wireless) factory-floor networks, in line with the RFieldbus IST Project (http://www.hurray.isep.ipp.pt/RFieldbus). Currently, his research interests are devoted to wireless sensor networks, namely on the design of an architecture for supporting real-time and reliable communications in large-scale embedded computing applications with critical requirements. He is leading the ART-WiSe (http/www.hurray.isep.ipp.pt/ART-WiSe) and Open-ZB (http://www.open-ZB.net) research frameworks.

Eduardo Tovar was born in 1967 and has received the Licentiate, MSc and PhD degrees in electrical and computer engineering from the University of Porto, Porto, Portugal, in 1990, 1995 and 1999, respectively. Currently he his Professor of Industrial Computer Engineering in the Computer Engineering Department at the Polytechnic Institute of Porto (ISEP-IPP), where he is also engaged in research on real-time distributed systems, wireless sensor networks, multiprocessor systems, cyber-physical systems and industrial communication systems. He heads the CISTER/IPP-HURRAY Research Unit (UI 608), a top ranked ("Excellent") unit of the FCT Portuguese network of research units. Since 1991 he authored or co-authored more than 100 scientific and technical papers in the area of real-time computing systems, wireless sensor networks, distributed embedded systems and industrial computer engineering. Eduardo Tovar has been consistently participating in top-rated scientific events as member of the Program Committee, as Program Chair or as General Chair. Examples are: IEEE RTSS (Real Time Systems Symposium); IEEE RTAS (Real-Time and Embedded Technology and Applications Symposium); IEEE SDRS (Symposium on Distributed Reliable Systems); IEEE ICDCS (International Conference on Distributed Computing Systems); ACM EMSOFT (Annual ACM Conference on Embedded Software); Euromicro ECRTS (Euromicro Conference on Real-Time Systems); IEEE ETFA (Emerging Technologies on Factory Automation) or IEEE WFCS (Workshop on Factory Communication Systems). Notably, he has been Program Chair for ECRTS 2005 and WDES 2006, and is currently Program Co-chair of OPODIS 2007.

André Cunha was born in 1981 in the city of Porto, Portugal. He has a degree in Computer Science from the School of Engineering of the Polytechnic Institute of Porto (2005) and an MSc degree in Computer Networks and Services from the Faculty of Engineering University of Porto (2007). He is a full time research assistant of the IPP-HURRAY!/CISTER Research Unit since February 2005. His research interests are in the thematic areas of wireless sensor network, critical systems, reliable and real-time communications, large-scale embedded distributed systems and ubiquitous computing. Currently, he is working in the ART-WiSe (http/www.hurray.isep.ipp.pt/art-wise) and Open-ZB (http://www.open-zb.net) research frameworks.