IPP Hurray!

Technical Report

GTS Allocation Analysis in IEEE 802.15.4 for Real-Time Wireless Sensor Networks

Anis Koubaa

Mário Alves

Eduardo Tovar

GTS Allocation Analysis in IEEE 802.15.4 for Real-Time Wireless Sensor Networks

Anis KOUBAA, Mário ALVES, Eduardo TOVAR

IPP-HURRAY!

Polytechnic Institute of Porto (ISEP-IPP)

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8340509

E-mail: {akoubaa@dei ,mjf@, emt@dei }.isep.ipp.pt

http://www.hurray.isep.ipp.pt

Abstract

The IEEE 802.15.4 protocol proposes a flexible communication solution for Low-Rate Wireless Local Area Networks including sensor networks. It presents the advantage to fit different requirements of potential applications by adequately setting its parameters. When enabling its beacon mode, the protocol makes possible real-time guarantees by using its Guaranteed Time Slot (GTS) mechanism. This paper analyzes the performance of the GTS allocation mechanism in IEEE 802.15.4. The analysis gives a full understanding of the behavior of the GTS mechanism with regards to delay and throughput metrics. First, we propose two accurate models of service curves for a GTS allocation as a function of the IEEE 802.15.4 parameters. We then evaluate the delay bounds guaranteed by an allocation of a GTS using Network Calculus formalism. Finally, based on the analytic results, we analyze the impact of the IEEE 802.15.4 parameters on the throughput and delay bound guaranteed by a GTS allocation. The results of this work pave the way for an efficient dimensioning of an IEEE 802.15.4 cluster.

GTS Allocation Analysis in IEEE 802.15.4
for Real-Time Wireless Sensor Networks

Anis KOUBAA, Mário ALVES, Eduardo TOVAR

IPP-HURRAY! Research Group, Polytechnic Institute of Porto
Rua Dr. Antonio Bernardino de Almeida, 431, 4200-072 Porto, PORTUGAL
{akoubaa, emt}@dei.isep.ipp.pt, mjf@isep.ipp.pt

Abstract

The IEEE 802.15.4 protocol proposes a flexible communication solution for Low-Rate Wireless Personal Area Networks including sensor networks. It presents the advantage to fit different requirements of potential applications by adequately setting its parameters. When enabling its beacon mode, the protocol makes possible real-time guarantees by using its Guaranteed Time Slot (GTS) mechanism. This paper analyzes the performance of the GTS allocation mechanism in IEEE 802.15.4. The analysis gives a full understanding of the behavior of the GTS mechanism with regards to delay and throughput metrics. First, we propose two accurate models of service curves for a GTS allocation as a function of the IEEE 802.15.4 parameters. We then evaluate the delay bounds guaranteed by an allocation of a GTS using Network Calculus formalism. Finally, based on the analytic results, we analyze the impact of the IEEE 802.15.4 parameters on the throughput and delay bound guaranteed by a GTS allocation. The results of this work pave the way for an efficient dimensioning of an IEEE 802.15.4 cluster.

1. Introduction

With the emergence of new Wireless Sensor Network (WSN) applications under timing constraints, the provision of deterministic guarantees may be more crucial than saving energy during critical situations. The IEEE 802.15.4 protocol [1] is one potential candidate to achieve predictable real-time performance for Low-Rate Wireless Personal Area Networks (LR-WPAN).

The IEEE 802.15.4 Task Group (TG4) [1], together with the ZigBee Alliance [2], has developed an entire communication protocol stack for LR-WPAN. Although the IEEE 802.15.4 protocol was not exclusively designed for wireless sensor networks, it provides suitable mechanisms for WSN applications [3]. The physical layer of the IEEE 802.15.4 protocol seems particularly suitable for WSN applications, namely in terms of data-rate, energy-efficiency and robustness. More importantly, the Medium Access Control (MAC) sub-layer, with the provision of a superframe structure bounded by two signaling beacon frames (when in beacon-enabled mode), makes the IEEE 802.15.4 protocol sufficiently flexible and able to fulfill the needs of a large variety of applications. In fact, when an IEEE 802.15.4-compliant WPAN disables the generation of periodic beacon frames (non-beacon enabled mode), all nodes in the network compete to gain access to the medium using non-slotted CSMA/CA. The advantage of the non beacon-enabled mode, with regards to WSN application requirements, is that it easily allows scalability and self-organization. However, the non

beacon-enabled mode does not provide any guarantee to deliver data frames, within a certain deadline. For time-critical applications, timeliness guarantees may be achieved with the beacon-enabled mode. This mode offers the possibility of allocating/deallocating time slots in a superframe, called Guaranteed Time Slots (GTSs), and providing predictable minimum service guarantees. Having a minimum service guarantee, it is possible to predict a worst-case timing performance of the network.

This paper provides a methodology, based on the Network Calculus formalism [4], for evaluating the performance of real-time applications using the GTS mechanism in one IEEE 802.15.4 cluster. We propose two alternative models for the service curve provided by a GTS allocation, and derive the corresponding delay bounds. An expression of the duty cycle as a function of the delay bound is also derived. Additionally, analysis on the impact of the IEEE 802.15.4 parameters, namely the superframe order, on the maximum throughput and delay bound of a GTS allocation is also presented.

To our best knowledge, this is the first work evaluating the deterministic real-time guarantees of the IEEE 802.15.4 protocol. We hope that this work provides a first step towards the practical use of the IEEE 802.15.4 GTS mechanism, and therefore providing deterministic guarantees for real-time sensor network applications.

2. Related Work

From an allocation point of view, the concept of a GTS allocation is similar to a Time Division Multiplex Access (TDMA) time slot allocation. A reserved amount of bandwidth is periodically granted for a given data flow. The amount of bandwidth is determined by the duration of the time slot and its periodicity. However, the IEEE 802.15.4 GTS mechanism is more flexible than a classic TDMA since the GTS duration may be dynamically adjusted by setting adequately the parameters of the IEEE 802.15.4 superframe (see Section 3), whereas a TDMA time slot duration is, in general, fixed for a given network configuration.

Moreover, TDMA and the IEEE 802.15.4 MAC approaches differ in several aspects. Actually, the IEEE 802.15.4 presents several advantages as compared to TDMA for deployment in WSNs. Scalability is an important limitation of TDMA-based approaches since the number of nodes within one TDMA cluster must be kept as low as possible, which hinders its usage in WSNs. On the other hand, the IEEE 802.15.4 is expected to

manage up to 254 nodes into one cluster. Another drawback of TDMA is the lack of support for dynamic topology changes (e.g. a node failure, a new node enters the network, mobility), since the TDMA schedule has to be re-adapted to the new configuration of the network, every time the topology changes. Moreover, communications in TDMA-based networks are quite dependent from the cluster manager. If it fails, the underlying sensor nodes will be disconnected from the entire network. This is not the case for the IEEE 802.15.4, since the protocol is able to operate either with or without a central manager, and it is designed to be easily adapted to different network topologies [1].

The performance of the IEEE 802.15.4 protocol has been subject of a few research studies, however focusing more on the performance of its CSMA/CA protocol [5-7] or on its general characteristics using simulations [8]. In this paper, we aim at characterizing the service provided by the GTS mechanism and its guaranteed delay bound. The evaluation of the delay bound can be roughly classified into two approaches: the Worst-Case Schedulability Analysis (WCSA) derived from [12] and Network Calculus [4]. In this paper we consider the Network Calculus approach since it is independent from the traffic representation, and is more adapted to the computation of network delays [10].

We derive the service curves and delay bounds guaranteed by an allocation of a GTS as functions of the protocol parameters. The service curves and delay bounds derived in this paper are explicitly related to the parameters of the IEEE 802.15.4, thus providing a suitable model for the GTS allocation scheme. Due to the similarities between the GTS allocation and TDMA time slot allocation, the analytic study presented in this paper partially holds for the TDMA case. However, to our best knowledge, there has been no previous work that explicitly derives the service curve and delay bounds of a TDMA allocation using Network Calculus or even the WCSA approach. In [11], the authors derived an extended framework of Network Calculus, called Real-Time Calculus, intended to evaluate the timing performance of embedded systems and scheduling policies (e.g. TDMA bus), but they only present a framework that is very similar to Network Calculus theory.

The rest of the paper is organized as follows. Section 3 gives an overview of the IEEE 802.15.4 protocol. Section 4 provides two accurate service curve models for a GTS allocation and the corresponding delay bounds. In sections 5 and 6, we analyze the impact of the protocol parameters on maximum throughput and delay bound for a GTS allocation, respectively. The paper concludes with section 7.

3. Overview of the IEEE 802.15.4 Protocol

The IEEE 802.15.4 MAC protocol supports two operational modes that may be selected by a central node called *PAN coordinator*:

- the *non beacon-enabled mode*, in which the MAC is ruled by non-slotted CSMA/CA;
- the *beacon-enabled mode*, in which beacons are periodically sent by the PAN coordinator to identify its PAN and synchronize nodes that are associated with it.

In this paper, we only consider the beacon-enabled mode, since it has the ability of providing timeliness guarantees to the network. 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 frame transmissions are allowed. During the inactive period (if it exists), all nodes may enter in a sleep mode, thus saving energy. Fig. 1 illustrates the beacon interval and the superframe structures.
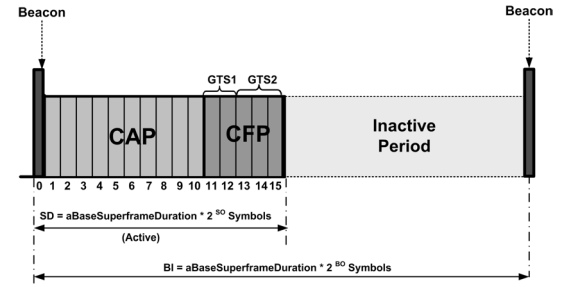

Fig. 1. Beacon interval and superframe concepts

The lengths of 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:

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

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

$$SD = aBaseSuperframeDuration * 2^{SO} , \quad for \ 0 \le SO \le BO \le 14 \tag{2}$$

In Eq. (1) and Eq. (2), *aBaseSuperframeDuration* denotes the minimum length of the superframe, corresponding to $SO = 0$. The IEEE 802.15.4 standard fixes this duration to 960 symbols (a symbol corresponds to 4 bits). This value corresponds to 15.36 ms, assuming a 250 kbps in the 2.4 GHz frequency band. In this paper, we will only consider the features of the 2.4 GHz frequency range, which is supported by the MICAz motes from Crossbow Tech. [9], for example.

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 request fails. The corresponding node then sends its data frames during the CAP. A detailed description of GTS management and of the slotted CSMA/CA mechanism is presented in [3].

4. Delay Bound Analysis of a GTS Allocation in an IEEE 802.15.4 Cluster

4.1 Problem Formulation

Let us consider an IEEE 802.15.4 cluster with a unique PAN coordinator, and a set of nodes within its radio coverage. The network operates in beacon-enabled mode, thus the PAN

coordinator periodically sends beacon frames. The Beacon Interval and the Superframe Duration are defined by Eq. (1) and Eq. (2), respectively. Let C be the total data rate of the output link. In our case, the data rate is fixed to $C = 250$ kbps, as previously mentioned in Section 3.

Each sensor node in the range of the PAN coordinator runs an application that generates a data flow. We consider that each data flow has 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 (Fig. 2).
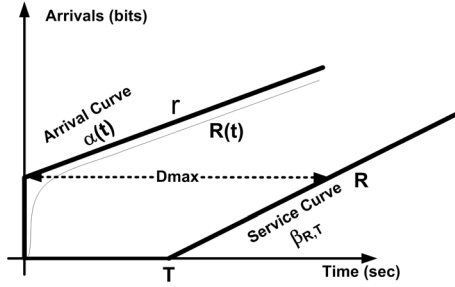

Fig. 2. Arrival curve, service curve and delay bound

This model is called a (b,r) model. In addition to its simplicity, the (b,r) model has the advantage to represent a higher bound of any kind of traffic, rather than dealing with unrealistic assumptions on the traffic type (e.g. assuming a Poisson arrival pattern). For instance, it has been shown [10], that a periodic traffic with or without jitter can be represented with a (b,r) curve. In general, it is possible to translate any traffic type to the (b,r) model, provided that the cumulative arrival function is upper bounded. The variance between the (b,r) curve and the realistic model may be large for highly variable data rate traffic. For a periodic traffic with or without jitter, however, the variance is not too pessimistic. Since WSN applications typically generate periodic traffic, the (b,r) model is considered to be acceptable in that context.

The first purpose of our analysis is to derive an expression for the delay bound of a data flow $R(t)$ bounded by a (b,r) curve, and that has allocated one GTS in the superframe (section 4.3). We then propose a general formula for a data flow that allocates n GTSs in the superframe (section 4.4). Section 4.2 gives a brief introduction to Network Calculus used in our analysis.

4.2 Delay Bound Analysis using Network Calculus

In Network Calculus theory [4], the delay bound analysis for a given data flow with a cumulative arrival function $R(t)$ assumes the following.

1. It exists an arrival curve $\alpha(t)$ that upper bounds $R(t)$ such that $\forall s, \ 0 \le s \le t, \ R(t) - R(s) \le \alpha(t-s)$. This inequality means that the amount of traffic that arrives to receive service in any interval $[s,t]$ never exceeds $\alpha(t-s)$.

2. It exists A minimum service curve $\beta(t)$ guaranteed to $R(t)$.

Then, the delay bound, D_{max}, for a data flow with an arrival curve $\alpha(t)$ that receives the service $\beta(t)$ is the maximum horizontal distance between $\alpha(t)$ and $\beta(t)$:

$$D_{\max} = h(\alpha,\beta) = \sup_{s \ge 0}\left\{\inf\left\{\tau \ge 0 : \alpha(s) \le \beta(s+\tau)\right\}\right\} \quad (3)$$

Fig. 2 illustrates an example of the delay bound for a linear arrival curve $\alpha(t) = b + r \cdot t$ that receives a *rate-latency* service curve $\beta_{R,T}(t) = R(t-T)^{+}$, where $R \ge r$ 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 $\alpha(t) = b + r \cdot t$ by the service curve $\beta_{R,T}(t) = R(t-T)^{+}$ is computed as [4]:

$$D_{\max} = \frac{b}{R} + T \quad (4)$$

4.3 Delay Bound Analysis for One Time Slot GTS

In this section, we derive the delay bound guaranteed for a data flow upper bounded by an arrival curve $\alpha(t) = b + r \cdot t$ and granted one GTS slot allocation in the superframe defined by the parameters BO and SO as defined in Eq. (1) and Eq. (2).

Based on the results of Network Calculus, our problem is reduced to finding a service curve $\beta(t)$ that corresponds to a GTS allocation. Then, it will be possible to derive the expression of the delay bound using Eq. (3) and Eq. (4).

4.3.1 The GTS Service Curve

Let us consider a one time slot GTS allocated to a data flow bounded by a (b,r) curve (Fig. 3).
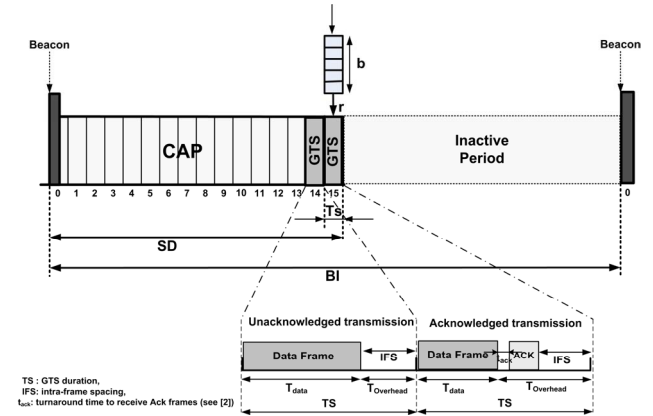

Fig. 3. The GTS service time and transmission modes

According to the standard [1], a node that has allocated a GTS can transmit a message **if and only if** the whole *transaction*, including data transmission, the Intra-Frame Spacing (IFS) and the acknowledgement (if requested), can be completed before the end of the GTS. Otherwise, it must wait until the next GTS. Fig. 3 also presents the cases of acknowledged and unacknowledged transactions. Hence, a given GTS allocation may impose a restriction on the frame length. Moreover, only a part of the GTS can be used for data transmission. The rest will be idle or used by a potential acknowledgement frame. The

impact of a given GTS allocation on the throughput will be discussed later on, in Section 5.

For the sake of simplicity and without loss of generality, we assume one data frame transmission in a GTS. Final results are still valid from multiple data frame transmissions inside a GTS.

Now, let Ts denote the Time Slot duration in the superframe. Then,

$$Ts = \frac{SD}{16} = abaseSuperframeDuration * 2^{(SO-4)} \qquad (5)$$

We define T_{data} and T_{idle} (Fig. 3) such that:

$$Ts = T_{data} + T_{idle} \qquad (6)$$

T_{data} defines the **maximum duration** used for data frame transmission inside a GTS. T_{idle} is the sum of *idle times* spent inside a GTS due to protocol overheads (IFS and/or Ack frames). The minimum value of T_{idle} comprises the time required for an IFS, T_{IFS}, and a potential acknowledgement in the case of an acknowledged transmission, T_{ack}. Hence, the following analysis holds for both acknowledged and unacknowledged transactions. The only difference is the setting of the value of T_{idle}. We can write:

$$T_{idle} = T_{Overhead} + T_{wasted}$$
$$where \ T_{Overhead} = T_{IFS} + T_{Ack} \cdot \amalg_{Ack} \qquad (7)$$

where $\amalg_{Ack} = 1$ for an acknowledged transaction and $\amalg_{Ack} = 0$ for an unacknowledged transaction. The value of *wasted time, T_{wasted}* is greater than zero if the length of a GTS is longer than the transaction time (transmission + IFS + Ack).

A frame of the data flow that has allocated the GTS uses the entire capacity of the output link for a time period of T_{data}, every Beacon Interval *BI*.

a. Real service curve of a GTS

First, we need to define the maximum latency T that a burst may wait for a service. This latency occurs for a burst that arrives just after the end of the GTS. It results that the maximum latency T (Fig. 4) can be computed as:
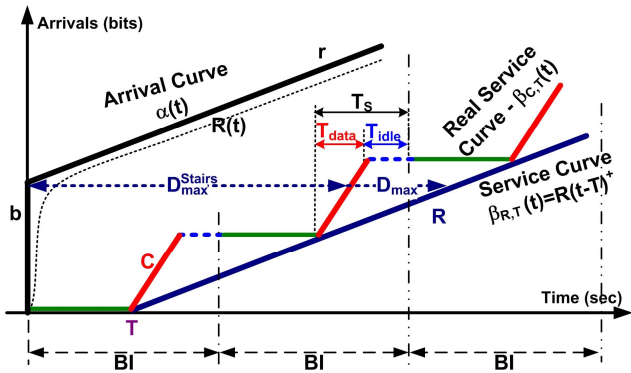
$$T = BI - Ts \qquad (8)$$

Fig. 4. The GTS service curves

In our analysis, we define the kth *superframe* as the superframe contained in the time interval $\left[(k-1)BI, \ k \cdot BI \right]$.

In the first superframe, the service curve received by the data flow; that is, the minimum number of bits that must be transmitted during the GTS is:

$$\beta_{C,T}^{1}(t) = \begin{cases} C\left(t - (BI - Ts)\right)^{+} & \forall t, \ 0 \leq t \leq BI - T_{idle} \\ 0 & Otherwise \end{cases}$$

where $(x)^{+} = \max(0, x)$. In the second superframe, the service curve received by the data flow is:

$$\beta_{C,T}^{2}(t) = \begin{cases} C \cdot T_{data} + C\left(t - (2 \cdot BI - Ts)\right)^{+} \\ \qquad \forall t, \ BI \leq t \leq 2 \cdot BI - T_{idle} \\ 0 \qquad Otherwise \end{cases}$$

By recurrence, the general expression that gives the service curve received in the kth superframe is:

$$\beta_{C,T}^{k}(t) = \begin{cases} (k-1) \cdot C \cdot T_{data} + C\left(t - (k \cdot BI - Ts)\right)^{+} \\ \forall t, \ (k-1) \cdot BI \leq t \leq k \cdot BI - T_{idle} \\ 0 \qquad Otherwise \end{cases} \qquad (9)$$

As a result, the overall service curve $\beta_{C,T}^{stair}(t)$ is defined as:

$$\beta_{C,T}^{stair}(t) = \sum_{k} \beta_{C,T}^{k}(t) \qquad \forall t \qquad (10)$$

$\beta_{C,T}^{stair}(t)$ is a stair function, which is the sum of all rate-latency service curves of each superframe, and represents an overall service curve for the data flow (Fig. 4).

b. Approximation to a rate-latency service curve

Hereafter, we show that the stair service curve $\beta_{C,T}^{stair}(t)$ can be simplified to a basic rate-latency service curve $\beta_{R,T}(t)$, where R is the guaranteed bandwidth of one time slot GTS.

In fact, R is the amount of bits sent in each beacon interval divided by BI, and is expressed as follows:

$$R = \frac{T_{data} \cdot C}{BI} = \left(\frac{Ts - T_{idle}}{BI}\right) \cdot C \qquad (11)$$

We denote DC as the duty cycle, which is defined as:

$$DC = \left(\frac{SD}{BI}\right) = 2^{IO} \leq 1 \quad where \quad IO = SO - BO \qquad (12)$$

IO represents the *Inactivity Order*. Then, applying Eqs. (5) (11) and (12), the guaranteed bandwidth can be rewritten as follows:

$$R = \frac{2^{IO}C}{16} - \frac{T_{idle}}{BI}C = \lambda \cdot DC \cdot C - w_{idle} \qquad (13)$$
$$where \ w_{idle} = w_{overhead} + w_{wasted}$$

with $\lambda = 1/16$, and w_{idle} representing the amount of *unused bandwidth* into a GTS due to protocol overheads (IFS and/or Ack frame) and wasted time inside the GTS.

Eq. (13) represents the guaranteed bandwidth as a function of the duty cycle. As a consequence, since R is the guaranteed bandwidth of one time slot GTS and $T = BI - Ts$ is the

maximum latency, then $\beta_{R,T}(t) = R(t-T)^+$ is a rate-latency service curve that corresponds to one time slot allocation.

Observe that in Fig. 4, $\beta_{R,T}(t)$ is a good approximation of $\beta_{C,T}^{stair}(t)$. In summary:

A GTS allocated in a superframe defined by a beacon order BO and a superframe order SO provides a minimum service curve $\beta_{C,T}^{stair}(t)$ expressed in Eq. (10), which may be approximated by a rate-latency service curve $\beta_{R,T}(t) = R(t-T)^+$ with an guaranteed bandwidth R expressed in Eq.(13) and a latency T expressed in Eq. (8).

4.3.2 Delay Bound

a. Case of a rate-latency service curve

Based on the results of Eq. (4) and if we consider the approximation of the service curve $\beta_{R,T}(t)$, the delay bound experienced by a data flow with an arrival curve $\alpha(t) = b + r \cdot t$, which has allocated one time slot GTS, can be computed as follows:

$$D_{\max} = \frac{b}{\lambda \cdot DC \cdot C - w_{idle}} + (BI - Ts) \tag{14}$$

b. Case of a stair service curve

A more accurate delay bound is obtained by considering the stair service curve $\beta_{C,T}^{stair}(t)$. The delay bound is represented by the maximum horizontal deviation $h\left(\alpha, \beta_{C,T}^{stair}\right)$ between the arrival curve $\alpha(t) = b + r \cdot t$ and $\beta_{C,T}^{stair}(t)$. The maximum horizontal deviation is reached at the angular point defined by the intersection between the slope of $\alpha(t)$ and the y-axis, so at the burst size b (Fig. 4).

Let us consider a burst size $b \leq C \cdot T_{data}$. Then, the delay bound is the maximum horizontal deviation between the angular point and the first stair of $\beta_{C,T}^{stair}(t)$; that is, $\beta_{C,T}^1(t)$. In this case, the delay bound $D_{\max}^{stair}$ is:

$$D_{\max}^{stair} = \frac{b}{C} + (BI - Ts) \qquad \text{if } b \leq C \cdot T_{data}$$

Now, if $C \cdot T_{data} < b \leq 2 \cdot C \cdot T_{data}$ then, the delay bound is the maximum horizontal deviation between the angular point and the second stair function of $\beta_{C,T}^{stair}(t)$; that is, $\beta_{C,T}^2(t)$. This example is shown in Fig. 4. In that case, the delay bound $D_{\max}^{stair}$ is:

$$D_{\max}^{stair} = \frac{b - C \cdot T_{data}}{C} + 2 \cdot BI - Ts$$
$$= \frac{b}{C} + 2 \cdot BI - Ts - T_{data}$$
$$\text{if } C \cdot T_{data} < b \leq 2 \cdot C \cdot T_{data}$$

In general, when $k \cdot C \cdot T_{data} < b \leq (k+1)C \cdot T_{data}$, we show that the delay bound of a data flow with an arrival curve $\alpha(t) = b + r \cdot t$, which has allocated one time slot GTS is:

$$D_{\max}^{stair} = \frac{b}{C} + (k+1) \cdot BI - Ts - k \cdot T_{data}$$
$$\text{if } k \cdot C \cdot T_{data} < b \leq (k+1) \cdot C \cdot T_{data} \tag{15}$$

4.4 General Case of n Time Slot GTS

The reasoning for a GTS with n allocated time slots is similar to the previous one except that the service will not be continuous due to the mandatory idle period inside each slot for IFS and acknowledgement processing. This causes a slight change in the construction of the stair service curve for an allocation of n time slot when $2 \leq n \leq 7$ (no more than 7 GTSs can be allocated in a superframe). Fig. 5 shows an example for allocations of 1, 2 and 3 GTSs inside a superframe.
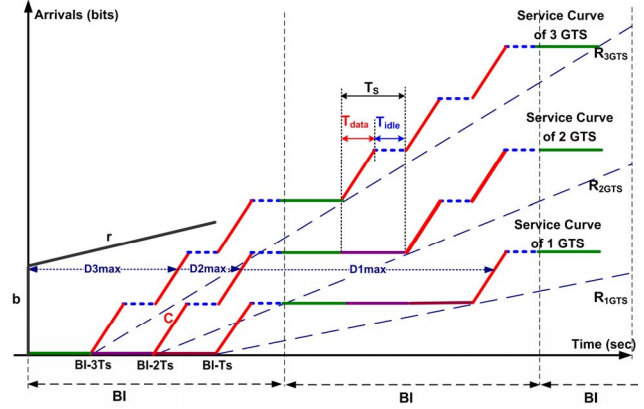

Fig. 5. The GTS service curve for *n* time slot allocation

As a consequence, the guaranteed bandwidth of a GTS with n time slots is:

$$R_n = n \cdot (\lambda \cdot DC \cdot C - w_{idle}) \tag{16}$$

and the latency T is then:

$$T_n = BI - n \cdot TS \tag{17}$$

So, the delay bound of a data flow with (b, r) curve by the service curve $\beta_{R,T}(t)$ becomes:

$$D_{n,\max} = \frac{b}{n(\lambda \cdot DC \cdot C - w_{idle})} + (BI - n \cdot Ts) \tag{18}$$

A more precise bound is obtained with the $\beta_{C,T}^{stair}(t)$ service curve as follows:

$$D_{n,\max}^{stair} = \left(\frac{b}{C} + (k+1) \cdot BI - n \cdot (Ts + k \cdot T_{data}) + m \cdot T_{idle} \right)$$
$$\text{where } m = \left\lfloor \frac{b - k \cdot (n \cdot T_{data}) \cdot C}{T_{data} \cdot C} \right\rfloor \tag{19}$$
$$\text{if } k \cdot C \cdot (n \cdot T_{data}) < b \leq (k+1) \cdot C \cdot (n \cdot T_{data})$$

5. Impacts of the Beacon and Superframe Orders On the Maximum Throughput of a GTS

5.1 Problem Statement

According to [1], it is possible to transmit more than one frame inside one allocated GTS. The only restriction is that the transactions (including IFS and acknowledgement if required) must complete before the end of the GTS. However, when the superframe order is incremented by one, the time slot duration is doubled. From a given superframe order, it is not possible to use

the entire duration of the time slot for data frame transmissions. This leads to an under utilization of the GTS. This is due to two factors.

1. The maximum burst size of a data flow is bounded by the maximum memory size of a node. It is not possible to store a bulk of data larger than the capacity of the memory.
2. The arrival of data frames from higher layers (application) is most often with low rate.

More formally, for a data flow with an arrival curve $\alpha(t) = b + r \cdot t$ the burst size b cannot exceed the memory size of the sensor node, and the average arrival rate of the data flow r, is assumed to be low.

In practice, memory sizes of the sensor nodes do not exceed a few kbits. For example, the TelosA mote [13] from mote*iv* company only provides 2 kbits for Random Access Memory (RAM) (we do not consider external flash memories), whereas the MICAz mote [9] provides up to 32 kbits.

On the other hand, the duration of one time slot for a superframe order $SO = 10$ is approximately equal to one second, which corresponds to send at most a bulk of 208 *aMaxPHYPacketSize* (= 1016 bits [1]) bit-long unacknowledged frames (~ 211 kbits) with a long IFS. However, due to sensor node resource limitations (processing and memory) it is unpractical to assume that a sensor node has the ability to generate 211 kbits of data during one second, due to the limitation of the arrival curve. For instance, a sensor node that generates traffic with a maximum burst size of 32 kbits, and an average arrival rate of 10 kbps cannot produce more than (32 kbits + (10 kbps · 1 sec)) = 42 kbits during one second. Hence, the amount of effectively used bandwidth does not exceed 42 kbps, which corresponds to a throughput of around 20% considering the previous scenario.

The purpose of this section is to evaluate the maximum throughput that may be achieved by a GTS for different values of the superframe order *SO*. We show how the burst size and the arrival rate affect the throughput for high superframe orders.

5.2 Maximum Throughput Evaluation

In what follows, we define the *maximum throughput* of a GTS as the maximum amount of bandwidth that can be used for data transmission in a GTS.

Similarly to the guaranteed bandwidth of a GTS (Eq. 13), the maximum throughput can be expressed as follows:

$$Th_{\max} = \lambda \cdot DC \cdot C - w_{idle} = \frac{T_{data}}{BI} \cdot C$$

The only difference between the guaranteed bandwidth and the maximum throughput is the value of T_{data}. In fact, the computation of the guaranteed bandwidth, independently from any arrival curve, assumes the usage of the entire capacity of a GTS. However, the maximum throughput depends from the arrival pattern, since the actual maximum utilization of a GTS depends from the burst size and the arrival rate.

Hence, for a given superframe order the maximum throughput is related to the maximum time *effectively used* for data transmission inside a GTS.

Consecutive data transmissions are separated by IFS. Depending on the length of the transmitted data frames, the IFS value is equal to a *SIFS* (*Short Inter-Frame Spacing* = 48 bits [1]) for data frame lengths lower than *aMaxSIFSFrameSize* = 144 bits [1]. Otherwise, the IFS is equal to *LIFS* is the (*Long Inter-Frame*

Spacing LIFS = 160 bits [1]) for frame lengths greater than *aMaxSIFSFrameSize* and smaller than *aMaxPHYPacketSize*. The value of the used IFS also impacts the throughput.

Depending on the burst size, two cases are possible.

a. The burst size is higher than the GTS duration

In this case, the GTS is fully used to transmit the burst (or a portion of it) at least during a time slot, as shown in Fig. 6.
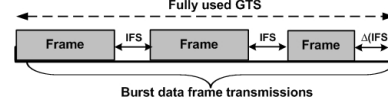

Fig. 6. Fully used GTS

The maximum throughput in this case depends on the maximum number of data frames, denoted as *N*, that may be sent during the GTS. This number depends on whether the frames are sent with a SIFS or LIFS intra-frame spacings. Hence, we consider two cases for evaluating the maximum throughput for multiple transmissions:

1. Multiple frames are transmitted with the maximum frame length, *aMaxPHYPacketSize*, therefore separated by LIFS periods. Then, the maximum number of frames that can be transmitted is:

$$N_{LIFS} = \left\lfloor \frac{Ts}{aMaxPHYPacketSize + LIFS} \right\rfloor + 1 \qquad (20)$$

Since the GTS is fully used in data transmission without wasted time, then the total duration of data transmission for "long" frames can be computed as:

$$T_{data}^{LIFS} = Ts - (N_{LIFS} - 1) \cdot LIFS - \Delta(IFS) \qquad (21)$$

and $\Delta(IFS)$ = SIFS bits if the last data transmission is lower than *aMaxSIFSFrameSize*; otherwise, it is equal to *LIFS*.

2. Multiple frames are transmitted with the maximum frame length equal to *aMaxSIFSFrameSize*, therefore separated by SIFS periods. Then, we have:

$$N_{SIFS} = \left\lfloor \frac{Ts}{aMaxSIFSFrameSize + SIFS} \right\rfloor + 1 \qquad (22)$$

Since the GTS is fully used for data transmission, i.e. without wasted time, then the total duration of data transmission for "short" frames can be computed as::

$$T_{data}^{SIFS} = Ts - N \cdot SIFS \qquad (23)$$

The maximum time used for transmission in a fully used GTS is then:

$$T_{data}^{full} = \max\left(T_{data}^{LIFS}, T_{data}^{SIFS}\right)$$
$$= \max \begin{pmatrix} Ts - (N_{LIFS} - 1) \cdot LIFS - \Delta(IFS), \\ Ts - N_{SIFS} \cdot SIFS \end{pmatrix} \qquad (24)$$

Hence, the maximum throughput, in the case of a fully used GTS, is defined as:

$$Th_{\max}^{full} = \max \begin{pmatrix} Ts - (N_{LIFS} - 1) \cdot LIFS - \Delta(IFS), \\ Ts - N_{SIFS} \cdot SIFS \end{pmatrix} \frac{C}{BI} \qquad (25)$$

Note also that $Th_{\max}^{full}$ represents the guaranteed bandwidth R for a GTS allocation.

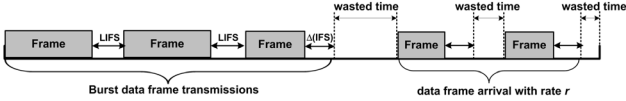In this case, the GTS is partially used, as shown in Fig. 7.

Fig. 7. Partially used GTS

The maximum throughput in this case depends on the maximum number of data frames, denoted as N, that may be sent during the GTS. Since the arrival is bounded by the curve $\alpha(t) = b + r \cdot t$, no more than $\alpha(t) = b + r \cdot Ts$ can arrive and be transmitted during the GTS.

On the other hand, in general, for high values of r (close to the link capacity C), the amount of bits carried by $\alpha(t)$ may be higher than what can actually be transmitted during the GTS when it is fully used, which is equivalent to the previous case. Hence, T_{data} is always lower than T_{data}^{full}.

Therefore, in general, we have:

$$T_{data} = \min \left(\begin{array}{l} \dfrac{b + r \cdot Ts}{C}, \\ \max \begin{pmatrix} Ts - (N_{LIFS} - 1) \cdot LIFS - \Delta(IFS), \\ Ts - N_{SIFS} \cdot SIFS \end{pmatrix} \end{array} \right) \quad (26)$$

As a result, the maximum throughput is:

$$Th_{max} = \min \left(\begin{array}{l} \dfrac{b + r \cdot Ts}{BI}, \\ \max \begin{pmatrix} Ts - (N_{LIFS} - 1) \cdot LIFS - \Delta(IFS), \\ Ts - N_{SIFS} \cdot SIFS \end{pmatrix} \dfrac{C}{BI} \end{array} \right) \quad (27)$$

Eq. (27) is a general formula that holds for both cases (*a* and *b*).

5.3 Numerical results

Now, we evaluate the maximum throughput for a duty cycle equal to one ($BO = SO$). Fig. 8 and Fig. 9 present the maximum throughput of one GTS for different superframe orders with a duty cycle equal to one, as a function of the average arrival rate r of data flow and the burst size, respectively. The results are obtained using Eq. (27).

To identify the impact of the average arrival rate on the maximum throughout, Fig. 8 is plotted for a constant burst size equal to 2 kbits. Idem, to identify the impact of the burst size on the maximum throughout, Fig. 9 is plotted for a constant average arrival rate equal to 5 kbps considering unacknowledged data frame transmissions.

It can be understood from both figures that the GTS allocation mechanism presents a bad behavior in terms of maximum throughput for superframe orders higher than 3. This is due to a high amount of wasted bandwidth in large GTSs.

In Fig. 8, for a given burst size (2 kbits), the throughput performance for high superframe orders is improved with higher arrival rate. It is possible to reach a maximum throughput with an arrival rate close to 250 kbps. However, in wireless sensor networks, it is more likely that the average arrival rate is low, leading to a poor utilization of a GTS.
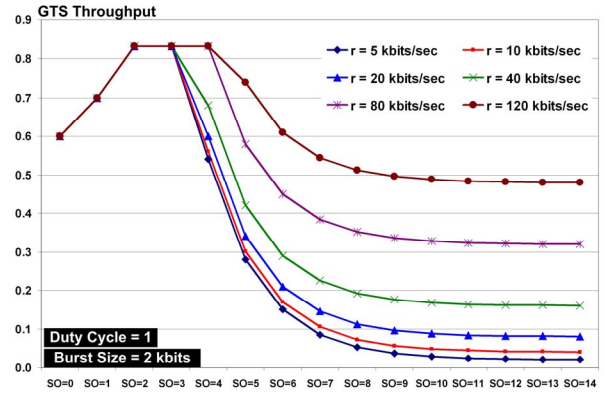

Fig. 8. Maximum throughput of a GTS allocation as a function of the average arrival rate

In Fig. 9, for a relatively low arrival rate (5 kbps), the throughput performance is improved when the burst size gets higher. In contrast with the impact of average arrival rate, the impact of increasing the burst size on the throughput is very limited for superframe orders higher than 10. However, the impact of increasing the burst size is more obvious for low superframe order (from 2 to 6).
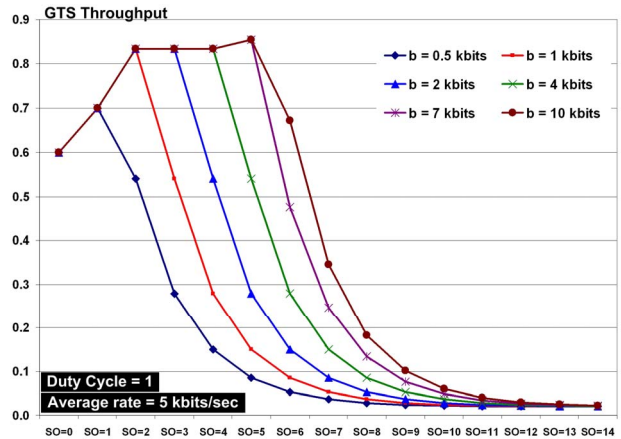

Fig. 9. Maximum throughput of a GTS allocation as a function of the burst size

For wireless sensor networks with low data rates and low burst sizes, a first conclusion we may draw is that high superframe orders are not suitable to ensure efficient usage of GTS allocation in terms of throughput. High superframe orders will result on increased amount of wasted bandwidth of an allocated GTS, which leads to underutilization of the shared wireless medium.

6. Impact of The Beacon and Superframe Orders On the Delay Bound

In this section, we investigate the impact of the beacon and the superframe orders on the delay bound. The problem is to determine the best configuration of the superframe structure (SO) that reduces the delay bound for a given duty cycle.

Referring to Eq. (14), the delay bound typically depends on the guaranteed bandwidth R and also on the beacon interval (service latency). Fig. 10 illustrates the guaranteed bandwidth per one GTS allocation (Eq. (25)).
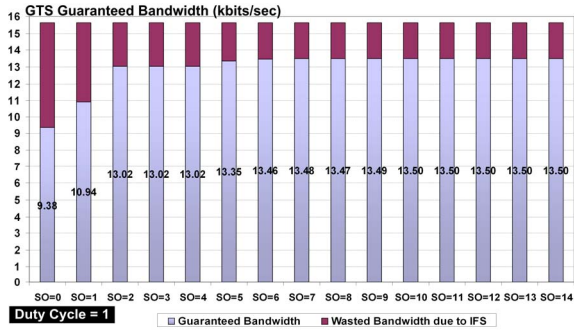
Fig. 10. Guaranteed bandwidth (kbps) per one GTS allocation

We observe that the guaranteed bandwidth is slightly above 13 kbps, except for low superframe orders. The guaranteed bandwidth for $SO = 0$ and $SO = 1$ is relatively low compared to the others. This is due to the impact of IFS, since the time slot durations are too small to send high amount of data.

Fig. 11 shows the delay bound of a GTS allocation, for a 100% duty cycle and an average arrival rate of 5 kbps, as a function of the burst size.
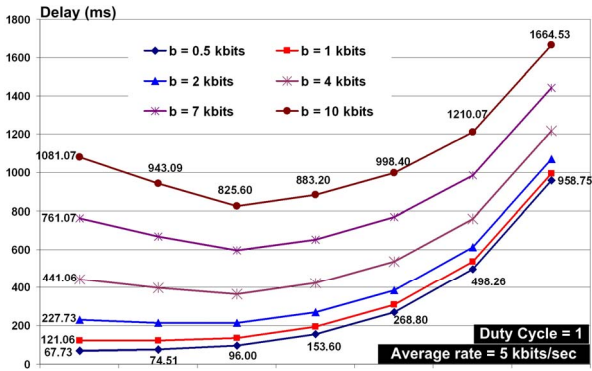

Fig. 11. Delay bound (ms)

Observe that the superframe order that provides the lowest delay bound depends from the burst size. For low burst size values (0.5 kbits, 1 kbits), the delay bound increases with the superframe order. In this case, the impact of the latency (BI-Ts) is more important than the impact of the guaranteed bandwidth on the delay bound, which turns the superframe order $SO = 0$ the most suitable for providing the lowest delay bound.

However, for higher values of burst size (2, 4, 7, and 10 kbits), the delay bound as a function of the superframe order is no longer monotonic. Actually, lower delays are achieved for superframe orders higher than 0. For example, a burst size of 10 kbits, the lowest delay bound is guaranteed with a superframe order $SO = 2$. Moreover, note that in this case, the delay bounds for superframe orders $SO = 3$ and $SO = 4$ are lower than that guaranteed with $SO = 0$. This behavior is explained by the impact of the guaranteed bandwidth on the delay bound. In fact, when the burst size is relatively high, the impact of the guaranteed bandwidth R is more important than the latency ($BI - Ts$) especially for the first three superframe orders. The low guaranteed bandwidth for $SO = 0$ explains well a higher delay bound than in $SO \in \{1, 2, 3, 4\}$ for a burst size equal to 10 kbits.

In summary, for common WSN applications with low burst sizes and low data rates, it is more likely that superframe order $SO = 0$ is the most suitable for providing real-time guarantees

thanks to its reduced latency. This is at the cost of lower throughput and guaranteed bandwidth, but this is not important for low arrival curves. However, if the burst size is relatively high (more than 1 kbits), the superframe order $SO = 2$ is better for providing timeliness guarantees thanks to its higher throughput.

7 Conclusions

In this paper, we have proposed a methodology for analyzing the Guaranteed Time Slot mechanism provided by the IEEE 802.15.4 protocol. Using Network Calculus formalism, we have proposed two accurate models for the service curve provided by a GTS allocation and derived the delay bounds guaranteed by such an allocation. An expression of the duty cycle as a function of the delay was also presented. Based on those results, the impact of the beacon order and the superframe order on the maximum throughput and delay bound was analyzed.

In summary, for WSN applications typically with low arrival rates and low burst size, using low superframe orders is more convenient for providing low delay bounds. However, they lack of efficient utilization of the GTS capacity due to their short duration and to the impact of IFS. We hope that this work provides a first step towards the efficient use of the GTS mechanism in IEEE 802.15.4-compliant networks.

8 Acknowledgments

References

[1]   IEEE 802.15.4 Standard-2003, "Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)", IEEE-SA Standards Board, 2003.

[2]   ZigBee Alliance, ZigBee Specification – June 2005, http://www.caba.org/standard/zigbee.html

[3]   A. Koubâa, M. Alves, and E. Tovar, "IEEE 802.15.4 for Wireless Sensor Networks: A Technical Overview", IPP-HURRAY Technical Report (TR-050702), Jul. 2005.

[4]   J-Y. Leboudec, and P. Thiran, *A Theory of Deterministic Queuing Systems for the Internet*, LNCS, Vol. 2050, P. 274 , ISBN: 3-540-42184-X, 2001.

[5]   J. Mišic, and V. B. Mišic, "Duty Cycle Management in Sensor Networks Based on 802.15.4 Beacon Enabled MAC", Ad Hoc and Sensor Wireless Networks Journal, Old City Publishing, 1(3):207-233, 2005.

[6]   J. Mišic and V. B. Mišic, "Access Delay and Throughput for Uplink Transmissions in IEEE 802.15.4 PAN", Elsevier Computer Communications Journal, 28(10):1152-1166, June 2005.

[7]   J. Mišic, S. Shafi, and V. B. Mišic, "The Impact of MAC Parameters on the Performance of 802.15.4 PAN", Elsevier Ad hoc Networks Journal, 3(5):509–528, 2005.

[8]   J. Zheng, and M. L. Lee, "A Comprehensive Performance Study of IEEE 802.15.4", IEEE Press Book, 2004.

[9]   MICAz DataSheet , http://www.xbow.com/

[10]   A. Koubâa, Y. Q. Song, "Evaluation and Improvement of Response Time Bounds for Real-Time Applications under Non-Preemptive Fixed Priority Scheduling", International Journal of Production and Research (IJPR), (Taylor & Francis), Vol. 42, No. 14, pp. 2899-2913, July 2004.

[11]   S. Chakraborty, S. Kunzli, and L. Thiele, "A General Framework for Analyzing System Properties in Platform-based Embedded System Designs", Design, Automation and Test in Europe Conference and Exhibition, Page(s):190 – 195, 2003.

[12]   C. L. Liu, and J. W Layland, "Scheduling Algorithms for Multiprogramming in Hard Real-Time Environment". Journal of ACM, 20, 46–61, 1973.

[13]   Telos mote specification sheet, http://www.moteiv.com/