Performance analysis of GTS allocation in Beacon Enabled IEEE 802.15.4

Pangun Park, Carlo Fischione, Karl Henrik Johansson

Abstract—Time-critical applications for wireless sensor networks (WSNs) are an important class of services supported by the standard IEEE 802.15.4. Control, actuation, and monitoring are all examples of applications where information must be delivered within some deadline. Understanding the delay in the packet delivery is fundamental to assess performance limitation for the standard. In this paper we analyze the guaranteed time slot (GTS) allocation mechanism used in IEEE 802.15.4 networks for time-critical applications. Specifically, we propose a Markov chain to model the stability, delay, and throughput of GTS allocation. We analyze the impact of the protocol parameters on these performance indexes. Monte Carlo simulations show that the theoretical analysis is quite accurate. Thus, our analysis can be used to design efficient GTS allocation for IEEE 802.15.4.

Keywords: IEEE 802.15.4 standard, GTS allocation, Wireless Sensor Network, MAC, Markov Chain.

I. INTRODUCTION

To support time-critical applications, IEEE 802.15.4 offers a guaranteed time slot (GTS) allocation mechanism at the network coordinator. The primary goal of GTS allocation is providing communication services to time critical data, i.e., make certain guarantees on eventual delivery and delivery times of packets to be transmitted by local devices to the network coordinator. Specifically, in IEEE 802.15.4, packets are transmitted on a superframe basis (see Fig. 1). Each superframe is divided into Contention Access Period (CAP), where nodes contend among each other to send packets, and a Contention Free Period (CFP), where nodes have GTSs to send packets without contention and thus with guaranteed transmission [1].

Most of the papers from the literature do not consider satisfactorily the CFP, where the GTS mechanism operates. Both CAP and CFP have been studied in [2] and [3], but the approach is mainly based on simulations. Some interesting algorithm are proposed to improve the performance of GTS allocation mechanism. To maximize the bandwidth utilization, the smaller slot size and offline message scheduling algorithm are proposed in [4], [5] and [6], respectively. In [7], the delay constraint and bandwidth utilization are considered to design the GTS scheduling algorithm. Huang [8] proposes an adaptive GTS allocation scheme by considering the low delay and fairness. An interesting theoretical performance evaluation of the GTS allocation has been proposed by Koubaa et al. [9], [10], [11] by using network calculus. These papers focus on the impact of the IEEE 802.15.4 standard parameters (the
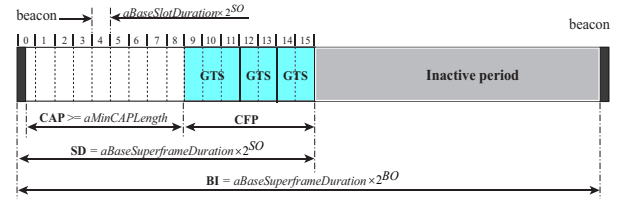
Fig. 1. Superframe structure in IEEE 802.15.4.

beacon and superframe orders [1]), the delay, throughput and energy consumption of GTS allocation. In [12], a round-robin scheduler is proposed to improve the bandwidth utilization based on network calculus approach. Network calculus, however, assumes a continuous flow model (whereas communication happens through low data rate packets in reality) and it analyzes the worst-case of traffic flows (which leads to severe under-utilization of time slots in actual environments). Consequently, the difference between the network flow model of the network calculus approach and the actual behavior may be quite large.

In this paper, we focus mainly on the uplink scenario, which is the most relevant for sensor networks application, and we develop a theoretical analysis of the impact of GTS allocation mechanism on stability, delay and throughput. More specifically, the original contributions of this paper are the following: We present a Markov chain model to analyze the performance of GTS allocation mechanism for IEEE 802.15.4 in terms of stability and delay in the slot assignment, and throughput guaranteed. The stability analysis, which gives the queue overflow probability, is quite useful for understanding the stability of the queue size of the network coordinator. Furthermore, we derive these performance measures as explicit function of the number of requests arriving during the superframes and protocol parameters. We believe that our investigation is the first one to provide such an accurate modelling and performance analysis. As a result, our theoretical analysis can be effectively used to support efficiently real-time applications by the optimization of the GTS allocation.

The remainder of this paper is organized as follows. In section II, we describe the IEEE 802.15.4 standard and the GTS allocation mechanism. In section III, we propose a Markov chain modelling of the GTS allocation. In section IV, we analyze the GTS stability, delay, and throughput. In section V, we validate the theoretical analysis by simulations, and give performance of GTS allocation. Section VI concludes the paper.

II. SYSTEM MODEL

In this section we give an overview of the key points of IEEE 802.15.4 that are needed for our analysis.

The IEEE 802.15.4 [1] standard specifies the physical layer and the MAC sub-layer for Low-Rate Wireless Personal Area Networks. The IEEE 802.15.4 supports beacon enabled and non-beacon enabled modes. The model selection is decided by the Personal Area Network coordinator (PANC). Fig. 1 shows a superframe structure of the beacon enabled mode. The PANC periodically sends the beacon frames in every beacon interval (BI) to identify its PAN and to synchronize devices that communicate with it. The PANC and devices can communicate during active period, called the superframe duration (SD), and enter the low-power mode during the inactive period. The structure of the superframe is defined by two parameters, the beacon order (BO) and the superframe order (SO), which determine the length of the superframe and its active period, respectively. The length of the superframe (BI) and the length of its active period (SD) are then defined as

$$\text{BI} = aBaseSuperframeDuration \times 2^{BO} , \qquad (1)$$

$$\text{SD} = aBaseSuperframeDuration \times 2^{SO} , \qquad (2)$$

where $0 \leq SO \leq BO \leq 14$. The *aBaseSuperframeDuration* and *aBaseSlotDuration* denote the minimum length of the superframe and the number of symbols forming a superframe slot, when BO is equal to 0, respectively. The active period is divided into 16 equally sized time slots. Each active period can be further divided into a CAP and an optional CFP, composed of guaranteed time slots. The slotted or unslotted CSMA/CA is used within CAP dependent on the beacon enabled and non-beacon enabled mode, respectively. The GTS allocation mechanism of IEEE 802.15.4 deals only with the beacon enabled mode.

Fig. 2(a) shows the data transmission during the CAP of beacon enabled mode. A slotted CSMA/CA mechanism is used to access the channel of non-time critical data frames and GTS requests during the CAP. In the CFP, the dedicated bandwidth is used for time critical data frames. Fig. 2(b) illustrates the GTS allocation mechanism within CFP of beacon enabled mode. The PANC is responsible for the GTS allocation and determines the length of the CFP in a superframe. To request the allocation of a new GTS, the device sends the GTS request command to the PANC. The PANC confirms its receipt by sending an acknowledgment frame within CAP. Upon receiving a GTS allocation request, the PANC checks whether there are sufficient resources and, if possible, allocates the requested GTS. The GTS capacity in a superframe satisfies the following requirements:

1) The maximum number of GTSs to be allocated to devices is seven, provided there is sufficient capacity in the superframe.
2) The minimum length of a CAP is *aMinCAPLength*.

Therefore the CFP length depends on the GTS requests and the current available capacity in the superframe.

If there is sufficient bandwidth in the next superframe, the PANC determines a device list for GTS allocation based on a first-come-first-served (FCFS) fashion. Then, the PANC includes the GTS descriptor which is the device list that obtains GTSs in the following beacon to announce the allocation information. The PANC makes this decision within *aGTSDescPersistenceTime* superframes. Note that on receipt
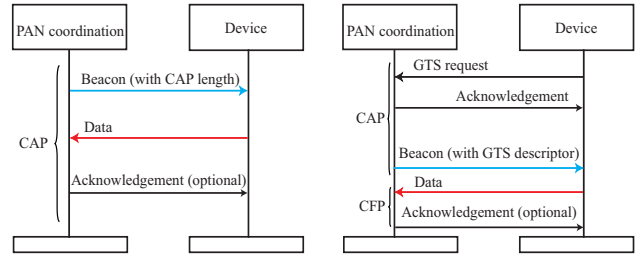
(a) Data transmission in CAP.   (b) Data transmission during CFP.

Fig. 2. Data transfers during the CAP and CFP of beacon enabled PAN coordinator.

of the acknowledgment to the GTS request command, the device continues to track beacons and wait for at most *aGTS-DescPersistenceTime* superframes. A device uses the dedicated bandwidth to transmit the packet within the CFP. In addition, a transmitting device ensures that its transaction is complete one interframe spacing (IFS) period before the end of its GTS. The MAC sublayer needs a finite amount of time to process data received by the PHY. Two successive frames transmitted from a device are separated by at least an IFS period. Depending on the length of the transmitted data frames, if the length of frame is less than a *aMaxSIFSFrameSize* then a Short Inter-Frame Spacing (SIFS) period of at least *macMinSIFSPeriod* symbols will follow. Otherwise Long Inter-Frame Spacing (LIFS) period of at least *macMinLIFSPeriod* symbols will be used.

In the following section, we propose an analytical modelling of the GTS allocation described above.

III. MODELLING OF GTS ALLOCATION

Consider the IEEE 802.15.4 standard with a star network and a set of N nodes within the PANC's radio coverage. Assume that the network operates in beacon enabled mode. Each device in the range of the PANC generates data packets to be sent to the PANC and informs the coordinator on the need of GTS resources by sending the request during CAP. Therefore, the PANC needs to allocate a number of GTSs by considering the received requests. These requests are stored in a queue of the PANC, and wait to be served in the next superframes, where the related GTS may be allocated. If too many requests arrive with respect to the PANC queue size, then we have a queue overflow. We consider only the transmit GTSs for the uplink traffic. Furthermore, we assume that all GTS transmissions are successful. Each device is allocated at most one GTS and the maximum number of GTSs Δ_u in a superframe is considered according to the IEEE 802.15.4 specifications [1].

The number of requests of GTSs depends on the number of time critical data packets that the devices want to send to the PANC. With contention-based transmissions during CAP, there will be loss of GTS requests due to data collisions in CAP. Hence, the performance of the GTS allocation mechanism in CFP is dependent on the number of requests that arrive successfully to the PANC during the CAP. Such a number is modelled by using a probability distribution, which abstracts the contention-based transmissions during CAP and possible collisions. We will analyze performance obtained by assuming different distributions (Poisson, Normal and Gamma). If there

are no sufficient GTS resources for the request in the current superframe, the GTS allocation will be delayed in the next superframe. For further details see [1].

The modelling of the GTS allocation is given in two steps. First, we derive the constraints on the number of time slots to allocate by considering the details of the GTS allocation mechanism of IEEE 802.15.4 specification. Then, we model the behavior of GTS allocation using Markov chain. Details follow in the sequel.

A. Number of Guaranteed Time Slots

In this subsection, we derive the number of GTSs (Δ_u) that can be allocated as a function of IEEE 802.15.4 protocol parameters (BO, SO) and we derive the number of time-critical data packets τ_n for each GTS request that can be served, and the maximum forward delay D_{FS} that a packet experiences to be transmitted in the GTS.

According to the standard [1], it is possible to transmit more than one data 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. Assume that each device requests to send the number of data frames τ_n with the packet forward delay D_{FS} including IFS and the transmission delay in symbol unit. The length of the packet forward delay is dependent on the size of the frame as follows

$$D_{\mathrm{FS}} = \begin{cases} D_{\mathrm{SIFS}} = \frac{L_{\mathrm{fr}} R_s}{R_b} + macMinSIFSPeriod \\ \qquad \text{if} \quad L_{\mathrm{fr}} \leq aMaxSIFSFrameSize\,, \\ D_{\mathrm{LIFS}} = \frac{L_{\mathrm{fr}} R_s}{R_b} + macMinLIFSPeriod \\ \qquad \text{otherwise}\,. \end{cases}$$
$$(3)$$

where L_{fr} is the frame length, R_s is the symbol rate, R_b is the bit rate. Let the duration of superframe slot be T_{SS}, then

$$T_{\mathrm{SS}} = \frac{T_{\mathrm{SD}}}{N_{\mathrm{SF}}} = T_{\mathrm{SFD}} \times 2^{SO-4}\,, \qquad (4)$$

where T_{SD} is the number of symbols forming a superframe, N_{SF} is the number of slots contained in a superframe, and T_{SFD} is the number of symbols forming a superframe when $SO = 0$. Consider that a single GTS may extend over a number of superframe slots θ_n. Since a given GTS needs to be greater than the total forward delay $\tau_n D_{\mathrm{FS}}$, it follows that

$$\theta_n \geq \theta_{\min} = \left\lceil \frac{\tau_n D_{\mathrm{FS}}}{T_{\mathrm{SS}}} \right\rceil\,, \qquad (5)$$

where $\theta_{\min}$ is the minimum number of superframe slots for a single GTS to serve the data frames $\tau_n D_{\mathrm{FS}}$. By considering the minimum length of a CAP, $T_{\mathrm{cap}} \geq T_{\min,\mathrm{cap}}$, the constraint of the maximum number of GTS that PANC can allocate in a superframe is given by

$$T_{\mathrm{SD}} - T_{\mathrm{cfp}} = T_{\mathrm{SD}} - \frac{T_{\mathrm{SD}}}{N_{\mathrm{SF}}} k\, \theta_{\min} \geq T_{\min,\mathrm{cap}}\,, \qquad (6)$$

where k is the number of waiting requests and T_{cfp} is the number of symbols forming a CFP. By considering that the maximum number of GTSs is 7 in any superframe, it results that the maximum number of GTSs to be allocated to devices

is

$$k \leq \Delta_u = \min \left(\left\lfloor \frac{N_{\mathrm{SF}} \left(1 - \frac{T_{\min,\mathrm{cap}}}{T_{\mathrm{SD}}} \right)}{\theta_{\min}} \right\rfloor, 7 \right)\,. \qquad (7)$$

Note that Δ_u is a function of IEEE 802.15.4 parameters, e.g., (BO, SO), and application constraints, e.g., the number of data packets τ_n and total maximum forward delay D_{FS}. In addition, a lower SO results on a decreasing of the maximum number of GTSs Δ_u.

B. Markov Chain Model

Here we develop a Markov model to study the behavior of the GTS allocation mechanism, see Fig. 3. Let t be a positive integer representing the time progress as expressed in superframe units. In other words, t is the superframe counter. Note that t corresponds to network time due to the beacon enabled mode. The superframe counter $t = 1$ correspond to the first superframe of the network without any waiting requests at the PANC.

Let λ_i be the probability of i successful requests during CAP, given that the maximum number of requests is $L_{\max}$. If the arriving requests observe that the queue size of already waiting requests is over a threshold $B_{\max}$, then these arriving requests are dropped because of the time limitation N_{GPT}. Note that $B_{\max}$ is fixed and it is given by $\Delta_u (N_{\mathrm{GPT}} + 1)$ where Δ_u is the maximum number of GTS for each superframe and $N_{\mathrm{GPT}} = aGTSDescPersistenceTime$. After N_{GPT} superframes later, the GTS description of the beacon is removed, see details in [1]. When there are k requests, $\Delta_u \leq k \leq B_{\max}$, then some requests $k - \Delta_u$ will be delayed to obtain GTS in the next superframe.

Let $n(t)$ be a non-negative integer representing the number of waiting requests at the beginning of the current superframe t. Then $n(t) \in \{0, \cdots, B_{\max}, B_{\max}^*\}$, where we denoted by $B_{\max}^*$ that some of the new arrived requests are dropped due to the limited queue size $B_{\max}$. If the requests are over $B_{\max}$ after the new arrivals, then some of the new requests will be dropped on the base of a FCFS fashion. We assume that there is no reallocation of GTS and each request is allocated at one GTS. By this assumption, the pair $\{t, n(t)\}$ can be used to build the two-dimensional Markov chain reported in Fig. 3. The transition probabilities of the chain are described by the following equations:

$$P\{t+1, j | t, i\} = \lambda_j, \quad \text{for} \quad 0 \leq i < \Delta_u, j \neq B_{\max}^*, \qquad (8)$$

$$P\{t+1, j | t, i\} = \lambda_{j + \Delta_u - i}, \quad \text{for} \quad i \geq \Delta_u, j \geq i - \Delta_u,$$
$$j \neq B_{\max}^* \qquad (9)$$

$$P\{t+1, j | t, i\} = 0, \quad \text{for} \quad i \geq \Delta_u, j < i - \Delta_u, j \neq B_{\max}^*, \qquad (10)$$

$$P\{t+1, B_{\max}^* | t, i\} = \lambda_{B_{\max}}^*, \quad \text{for} \quad 0 \leq i < \Delta_u, \qquad (11)$$

$$P\{t+1, B_{\max}^* | t, i\} = \lambda_{B_{\max} + \Delta_u - i}^*, \quad \text{for} \quad i \geq \Delta_u, \qquad (12)$$

where $t \geq 1$ and $\lambda_k^* = 1 - \sum_{i=0}^{k} \lambda_i = \sum_{i=k+1}^{L_{\max}} \lambda_i$. Note that the maximum number of requests is $L_{\max}$. Eq. (8) shows the probability that j requests arrive when there are less than the maximum number of GTSs Δ_u requests at superframe t. Note that GTSs will be allocated to previously arrived requests
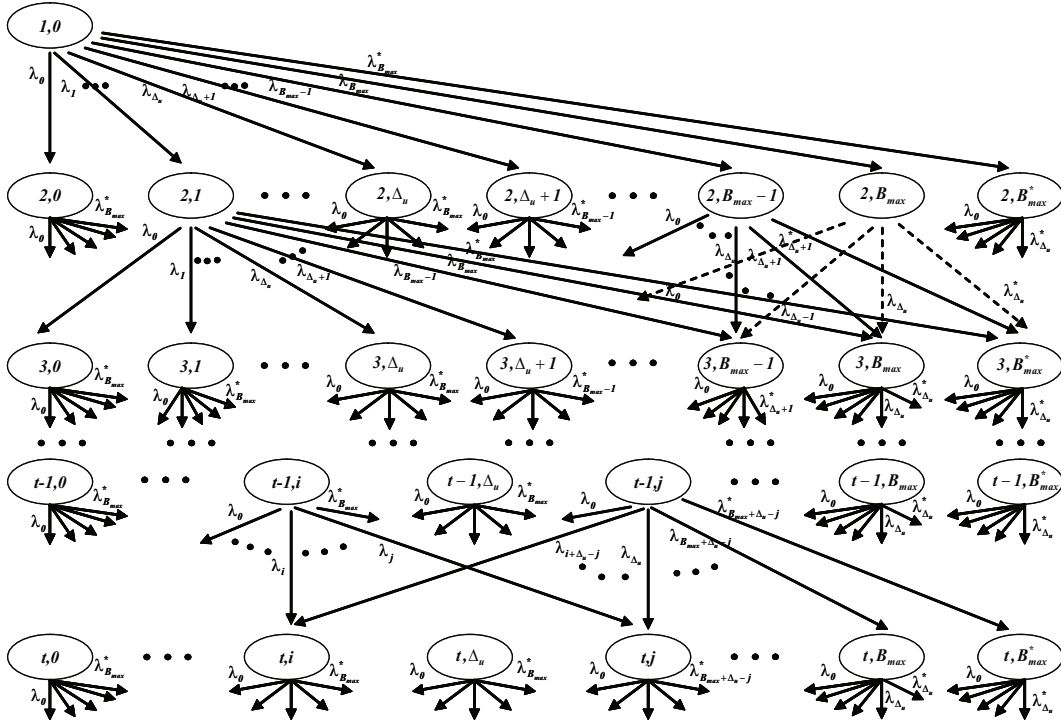
Fig. 3. Markov chain model for the GTS allocation of the CFP period.

before CAP of the superframe t i.e., all waiting requests at the beginning of the superframe $t+1$ arrive during CAP of the superframe t. Eq. (9) gives the probability that j requests are remained at the superframe $t+1$ when there are some requests $i \geq \Delta_u$ at the superframe t. The PANC allocates a maximum number of GTS Δ_u to serve the requests $i \geq \Delta_u$. Furthermore, since the number of waiting requests is larger than Δ_u, some requests will fail to get GTS at the superframe t. Hence, there are two different groups of requests at the superframe $t+1$, one group of requests arriving before the superframe t and another group of requests arriving at the superframe t. Let us call the first group as "*old requests*" and the second group as "*new requests*" throughout this paper. Eq. (10) shows the probability that there is less number of requests in the superframe $t+1$ than $i - \Delta_u$ if there are $i \geq \Delta_u$ requests at superframe t. Note that the transition probability is equal to 0 due to the maximum number of GTSs Δ_u. Eqs. (11) and (12) show the transition probabilities of the drop state $B_{\max}^*$ for the requests at superframe t.

Let us denote the probability that the process described by the Markov chain is in state k at time t as follows: $\pi_k^t = P\{t, k\}$ for $t \in \{1, \cdots, \infty\}$ and $k \in \{0, \cdots, B_{\max}, B_{\max}^*\}$. Using the transition probabilities given by Eqs. (8)–(12), we derive the state probabilities at superframe $t+1$ as

$$\pi_j^{t+1} = \sum_{i=0}^{\Delta_u - 1} \pi_i^t \lambda_j + \sum_{i=\Delta_u}^{j+\Delta_u} \pi_i^t \lambda_{j+\Delta_u - i},$$
$$\text{for} \quad j + \Delta_u < B_{\max}, j \neq B_{\max}^*, \tag{13}$$

$$\pi_j^{t+1} = \sum_{i=0}^{\Delta_u - 1} \pi_i^t \lambda_j + \sum_{i=\Delta_u}^{B_{\max}} \pi_i^t \lambda_{j+\Delta_u - i} + \pi_{B_{\max}^*}^t \lambda_{j+\Delta_u - B_{\max}},$$
$$\text{for} \quad j + \Delta_u \geq B_{\max}, j \neq B_{\max}^*, \tag{14}$$

$$\pi_{B_{\max}^*}^{t+1} = \sum_{i=0}^{\Delta_u - 1} \pi_i^t \lambda_{B_{\max}}^* + \sum_{i=\Delta_u}^{B_{\max}} \pi_i^t \lambda_{B_{\max} + \Delta_u - i}^* + \pi_{B_{\max}^*}^t \lambda_{\Delta_u}^*,$$
$$\text{for} \quad j = B_{\max}^*, \tag{15}$$

where $t \in \{1, \ldots, \infty\}$ and $i, j \in \{0, \ldots, B_{\max}, B_{\max}^*\}$. The state probabilities of Eqs. (13) and (14) are given by the sum of two parts, due to the number of waiting requests at the superframe t.

Denote by $\boldsymbol{\pi} = \lim_{t \to \infty} \boldsymbol{\pi}^t \in \mathbb{R}^{1 \times (B_{\max} + 2)}$ the stationary distribution and by $\boldsymbol{P} \in \mathbb{R}^{(B_{\max}+2) \times (B_{\max}+2)}$ the transition probability matrix [13]. For numerical calculations, it is convenient to use the following expression: $\boldsymbol{\pi} = \mathbf{e} \left(\boldsymbol{P} - \boldsymbol{I} + \boldsymbol{E} \right)^{-1}$, where $\boldsymbol{I} \in \mathbb{R}^{(B_{\max}+2) \times (B_{\max}+2)}$ is the identity matrix, $\boldsymbol{E} \in \mathbb{R}^{(B_{\max}+2) \times (B_{\max}+2)}$ is the matrix with all the elements equal to 1, and $\mathbf{e} \in \mathbb{R}^{1 \times (B_{\max}+2)}$ is the vector with all elements equal to 1. From the Markov chain of Fig. 3, we see that the transition probability matrix is

$$\boldsymbol{P} =$$

$$\begin{pmatrix}
\lambda_0 & \lambda_1 & \cdots & \lambda_{\Delta_u} & \lambda_{\Delta_u+1} & \cdots & \lambda_{B_{\max}} & \lambda_{B_{\max}}^* \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\lambda_0 & \lambda_1 & \cdots & \lambda_{\Delta_u} & \lambda_{\Delta_u+1} & \cdots & \lambda_{B_{\max}} & \lambda_{B_{\max}}^* \\
0 & \lambda_0 & \lambda_1 & \cdots & \lambda_{\Delta_u} & \cdots & \lambda_{B_{\max}-1} & \lambda_{B_{\max}-1}^* \\
\vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & \lambda_0 & \lambda_1 & \cdots & \lambda_{\Delta_u+1} & \lambda_{\Delta_u+1}^* \\
0 & 0 & \cdots & 0 & \lambda_0 & \cdots & \lambda_{\Delta_u} & \lambda_{\Delta_u}^* \\
0 & 0 & \cdots & 0 & \lambda_0 & \cdots & \lambda_{\Delta_u} & \lambda_{\Delta_u}^*
\end{pmatrix},$$
$$\tag{16}$$

where $\lambda_k^* = 1 - \sum_{i=0}^k \lambda_i = \sum_{i=k+1}^{L_{\max}} \lambda_i$.

IV. Performance Analysis of GTS Allocation

In this section, we build on the modelling of the GTS allocation developed in previous section by the Markov chain to analyze performance of GTS allocation in terms of stability (expected number of waiting and dropped requests, and queue overflow probability), delay of serving the requests, and throughput.

A. Stability Analysis

Here we give the expected number of GTSs requests by the devices waiting to be served, the expected number of GTSs requests that are dropped because of limited bandwidth, and the queue overflow probability.

Using the state probability π_k^t of the Markov chain derived in the previous section, we can compute the mean number of waiting requests of the PANC at the superframe t by

$$\mathbb{E}\left[r(t)\right] = \sum_{k=0}^{B_{\max}} k\,\pi_k^t + B_{\max}\,\pi_{B_{\max}^*}^t. \qquad (17)$$

Note that the number of requests is related to the delay of GTS allocation due to a FCFS fashion for the queue management.

From the Markov chain model, we see that the expected number of dropped requests is given by

$$\mathbb{E}\left[r_d(t)\right] = \sum_{i=0}^{\Delta_u-1} \pi_i^t \sum_{k=B_{\max}+1}^{L_{\max}} \lambda_k\,(k - B_{\max})$$
$$+ \sum_{i=\Delta_u}^{B_{\max}} \sum_{k=B_{\max}-i+\Delta_u+1}^{L_{\max}} \pi_i^t\,\lambda_k\,(k-(B_{\max}-i+\Delta_u))$$
$$+ \pi_{B_{\max}^*}^t \sum_{k=\Delta_u+1}^{L_{\max}} \lambda_k\,(k-\Delta_u). \qquad (18)$$

Let P_{over}^{t+1} denote the queue overflow probability of the requests in the superframe $t+1$. Then

$$P_{\text{over}}^{t+1} = \sum_{i=0}^{\Delta_u-1} \pi_i^t \sum_{k=B_{\max}+1}^{L_{\max}} \lambda_k + \sum_{i=\Delta_u}^{B_{\max}} \sum_{k=B_{\max}-i+\Delta_u+1}^{L_{\max}} \pi_i^t\,\lambda_k$$
$$+ \pi_{B_{\max}^*}^t \sum_{k=\Delta_u+1}^{L_{\max}} \lambda_k$$
$$= \sum_{i=0}^{\Delta_u-1} \pi_i^t\,\lambda_{B_{\max}}^* + \sum_{i=\Delta_u}^{B_{\max}} \pi_i^t\,\lambda_{B_{\max}-i+\Delta_u}^* + \pi_{B_{\max}^*}^t\,\lambda_{\Delta_u}^*. \qquad (19)$$

Finally, to analyze the stability of GTS allocation, we can use the stationary distribution given by Eq. (16) to derive the limit as t tends to infinity of the expected number of waiting requests, the expected number of dropped requests, and the queue overflow probability.

B. Delay Analysis

In this subsection, we analyze the expected delay of GTS allocation, namely the average delay between the arrival of a new request for GTSs by a device for a time-critical packet and its effective allocation in some of the next superframes.

The PANC determines a device list for GTS allocation in the next superframe based on a FCFS fashion. When *new requests* are received in a CAP, then the delay of GTS allocation can be estimated by observing the queue size of waiting requests. Note that PANC makes a preliminary decision whether it is able to serve a request or not. Assume that the arrival process of requests is uniformly distributed during CAP. Hence, the mean delay between the arrival time in CAP and the end of CAP at a superframe is half of CAP period.

Suppose that j requests arrive at the superframe t and observe i requests already waiting in the queue. We can calculate the number of successful requests that get a GTS and delayed requests that fail to obtain a GTS at the next superframe, as explained in the following. Based on FCFS fashion, j *new requests* are able to obtain GTS after i *old requests* arriving before the current superframe t. We first compute the number of superframe to allocate GTS for *old requests* (see Fig. 4). Notice that i *old requests* require to use $q_i + 1$ superframes, plus r_i remainders that obtain the GTSs later, after $q_i + 1$ superframes have been served. Hence, i *old requests* require $q_i + 2$ superframes to obtain GTS if there are r_i remainders. Note that r_i remainders will share the CFP along with the *new requests* at the superframe $t + q_i + 1$, as represented in Fig. 4. It follows that

$$q_i = \left\lfloor \frac{i-\Delta_u}{\Delta_u} \right\rfloor, \qquad r_i = \text{rem}\left(\frac{i-\Delta_u}{\Delta_u}\right), \qquad (20)$$

where q_i is an integer quotient and the function **rem** returns an integer remainder. The nominator $i-\Delta_u$ of Eq. (20) takes into account the current GTS allocation mechanism of superframe t. Note that the *new requests* are not able to get GTS at the current superframe t since beacon includes information of GTS allocation at each superframe.

By considering the waiting time of *old requests*, it is possible to calculate the delay of j *new requests*. The quotient q_i of *old requests* can be considered as a delay offset in the computation of the delay of the *new requests*. Note that we consider r_i remainders by summing it with j *new requests* due to the shared CFP with r_i remained requests (see Fig. 4). By a similar approach used to derive Eq. (20), the quotient and remainder of *new requests* are given by

$$q_j = \left\lfloor \frac{j+r_i}{\Delta_u} \right\rfloor, \qquad r_j = \text{rem}\left(\frac{j+r_i}{\Delta_u}\right). \qquad (21)$$

The *new requests* require $q_j + 1$ superframes to complete GTS allocation. By looking at Fig. 4, r_j waiting requests will remain waiting to obtain GTS at the superframe $t+q_i+q_j+1$.

To analyze the delay, we define the beacon interval as

$$T_{\text{BI}}^{t,t+1} = T_{\text{cfp}}^t + T_{\text{sp}}^t + T_{\text{cap}}^{t+1}. \qquad (22)$$

where T_{cfp}^t is a CFP at superframe t, T_{sp}^t is a inactive period at superframe t, and T_{cap}^{t+1} is a CAP at superframe $t+1$. Let $D_{i,j,t}$ denote the expected delay of j *new requests* when there are i *old requests* at the superframe t. We distinguish the number of successful requests that obtain GTS out of j *new requests* for a different superframe. Note that j *new requests* of the superframe t will obtain GTS from the superframe $t + q_i + 1$ to the superframe $t + q_i + q_j + 1$. From Fig. 4, we see that the delay of allocating GTS is the sum of three components:

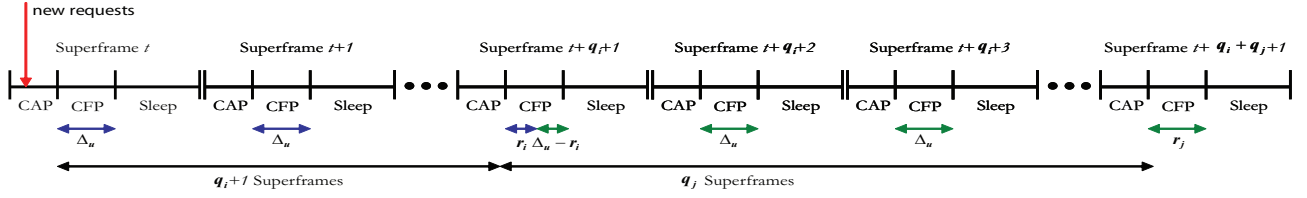- Arrival delay of CAP: Since the arrival time of *new*

Fig. 4. Waiting time of j *new requests* that observe i *old requests* at superframe t.

requests is uniformly distributed in CAP period, the arrival delay must be based on FCFS fashion.

- Offset delay of *old* and *new requests* : The *new requests* need to wait a number of superframes before obtaining GTS since there are *old* and *new requests* arriving before the current request.
- Service delay of CFP: If the *new requests* are able to obtain GTS at the current superframe, then the superframe slot size T_{SS} and the minimum number of superframe slots θ_{min} need to be accounted for.

Hence, it is possible to describe the delay of j *new requests* by considering the superframe time to obtain GTS, as given by the following claim:

Claim 1: The superframe time delay to obtain GTS is

$$d_g(n) =$$
$$\begin{cases} \sum_{k=t}^{t+q_i} T_{BI}^{k,k+1} + \frac{T_{cap}^t}{2} + \theta_{min} T_{SS}^{t+q_i+1}(r_i + \frac{j+1}{2}) \\ \quad \text{for} \quad j < r_j \, (q_j = 0) \\ \sum_{k=t}^{t+q_i} T_{BI}^{k,k+1} + T_{cap}^t \left(1 - \frac{\Delta_u - r_i}{2j}\right) \\ \quad + \theta_{min} T_{SS}^{t+q_i+1} \left(r_i + \frac{\Delta_u - r_i + 1}{2}\right) \\ \quad \text{for} \quad j \geq r_j \, (q_j > 0), n = t + q_i \\ \sum_{k=t}^{t+q_i} T_{BI}^{k,k+1} + \sum_{l=t+q_i+1}^{n} T_{cap}^t \left(1 - \frac{\Delta_u - r_i}{j}\right) \\ \quad - \frac{(n - (t+q_i+1))\Delta_u}{j} - \frac{\Delta_u}{2j}\right) + T_{BI}^{l,l+1} + \frac{\theta_{min} T_{SS}^{n+1}(\Delta_u + 1)}{2} \\ \quad \text{for} \quad j \geq r_j \, (q_j > 0), t + q_i + 1 \leq n \leq t + q_i + q_j - 1 \\ \sum_{k=t}^{t+q_i} T_{BI}^{k,k+1} + \sum_{l=t+q_i}^{t+q_i+q_j} T_{cap}^t \left(1 - \frac{\Delta_u - r_i}{j} - \frac{(q_j-1)\Delta_u}{j}\right) \\ \quad - \frac{r_j}{2j}\right) + T_{BI}^{l,l+1} + \frac{\theta_{min} T_{SS}^{t+q_i+q_j+1}(r_j + 1)}{2} \\ \quad \text{for} \quad j \geq r_j \, (q_j > 0), n = t + q_i + q_j. \end{cases}$$
$$(23)$$

where q_i, q_j is the quotient and r_i, r_j is the remainder of *old* and *new requests*, respectively.

Proof: If the *new requests* are $j < r_j$ (i.e., $q_j = 0$), then the average of arrival delay is half of T_{cap}. Furthermore, the average of service delay is the half of the service time of $j + 1$ *new requests* with a single GTS duration $\theta_{min} T_{SS}^{t+q_i+1}$ for each request. Note that r_i remainders are considered as an offset before the GTS allocation of j *new requests* . The offset $\sum_{k=t}^{q_i+t} T_{BI}^{k,k+1}$ takes into account the delay of i *old requests* .

If j *new requests* are more than r_j (i.e., $q_j > 0$), then it is possible to categorize the delay into three groups (initial, intermediate, end delay). The different group of delays are also combined with the arrival, offset, and service delay components. The initial delay is the delay of first allocated GTSs $\Delta_u - r_i$ out of j *new requests* at the superframe $t+q_i+1$, see Fig. 4. The average of arrival delay of the initial delay group considers the ratio $1 - \frac{\Delta_u - r_i}{2j}$ since the arrival time of j *new requests* are uniformly distributed during CAP. With a similar approach of $j < r_j$, the average of the service delay

considers r_i remainders and $\Delta_u - r_i$ *new requests* .

The intermediate delay is the delay between the first and last allocated GTSs $(q_j - 1)\Delta_u$ out of j *new requests* from the superframe $t + q_i + 2$ to $t + q_i + q_j$, see Fig. 4. Hence, the number of allocated GTSs is the maximum number of GTSs Δ_u out of j *new requests* . The average of arrival delay of the intermediate delay group considers the ratio $1 - \frac{\Delta_u - r_i}{j} - \frac{(n - (t+q_i+1))\Delta_u}{j} - \frac{\Delta_u}{2j}$ since other requests of the number $\Delta_u - r_i$ of the initial delay group and the number $(n - (t+q_i+1))\Delta_u$ of the intermediate delay group obtain GTSs before Δ_u *new requests* out of j. Note that the number of allocated GTS is dependent on the time progress n (superframe unit). With a similar approach used in the case $j < r_j$, we see that the average of the service delay considers Δ_u *new requests* .

The end delay is the delay of last allocated GTSs r_j out of j *new requests* at the superframe $t + q_i + q_j + 1$, see Fig. 4. The average of arrival delay of the end delay group considers the ratio $1 - \frac{\Delta_u - r_i}{j} - \frac{(q_j-1)\Delta_u}{j} - \frac{r_j}{2j}$ since other requests of the number $\Delta_u - r_i$ of initial delay group and the number $(q_j - 1)\Delta_u$ of intermediate delay group obtain GTSs before r_j remainders out of j *new requests* . With a similar approach of the case $j < r_j$, we see that the average of the service delay considers r_j remainders. ∎

By considering Eq. (23), the expected delay $D_{i,j,t}$ of j *new requests* that oberserve a queue size of i waiting requests at superframe t is

$$D_{i,j,t} = \begin{cases} \sum_{k=t}^{t+q_i} T_{BI}^{k,k+1} + \frac{T_{cap}^t}{2} + \theta_{min} T_{SS}^{t+q_i+1}(r_i + \frac{j+1}{2}) \\ \quad \text{for} \quad j < r_j \\ \frac{\Delta_u - r_i}{j} d_g(t + q_i) + \sum_{n=t+q_i+1}^{t+q_i+q_j-1} \frac{\Delta_u}{j} d_g(n) \\ \quad \times u(q_j - 2) + \frac{r_j}{j} d_g(t + q_i + q_j) \quad \text{for} \quad j \geq r_j. \end{cases}$$
$$(24)$$

where $u(t)$ is the unit step function.

Hence, the following claim holds:

Claim 2: The expected delay experienced by j *new requests* arriving at superframe t is

$$\mathbb{E}[D(t)] = \sum_{i=0}^{\Delta_u - 1} \sum_{j=1}^{L_{max}} \pi_i^t \frac{\lambda_j}{\sum_{k=1}^{L_{max}} \lambda_k} \{D_{i,j,t}\, u(B_{max} - j)$$

$$+ D_{i,B_{max},t}\, u(j - (B_{max} + 1))\}$$

$$+ \sum_{i=\Delta_u}^{B_{max}} \sum_{j=1}^{L_{max}} \pi_i^t \frac{\lambda_j}{\sum_{k=1}^{L_{max}} \lambda_k} \{D_{i,j,t}\, u(B_{max} - i + \Delta_u - j)$$

$$+D_{i,B_{\max}-i+\Delta_u,t}\,u(j-(B_{\max}-i+\Delta_u+1))\}$$

$$+\pi^t_{B^*_{\max}}\sum_{j=1}^{L_{\max}}\frac{\lambda_j}{\sum_{k=1}^{L_{\max}}\lambda_k}\{D_{B_{\max},j,t}\,u(\Delta_u-j)$$

$$+D_{B_{\max},\Delta_u,t}\,u(j-(\Delta_u+1))\}\,,\qquad(25)$$

where $D_{i,j,t}$ is the estimated delay by using the Eq. (24) and $L_{\max}$ is the maximum number of requests.

Proof: The first term of Eq. (25) presents the average delay of *new requests* when the *old requests* $i \leq \Delta_u - 1$. By considering the Markov chain, if there are more than $B_{\max}$ *new requests* , then the extra requests out of $B_{\max}$ arrivals will be dropped. The unit function is used to take into account the limited queue size $B_{\max}$. The second term considers the average delay when the *old requests* $\Delta_u \leq i \leq B_{\max}$. Note that the Markov chain represents the feasible number of *new requests* without the dropped requests. With similar approach of first term, it is possible to consider the maximum number of *new requests* , $B_{\max} - i + \Delta_u$, without dropped requests when i *old requests* wait in the queue. The third term computes the average delay at the dropped state $B^*_{\max}$, see the Markov chain. At the dropped state $B^*_{\max}$, Δ_u *new requests* are considered since the queue size of *old requests* is $B_{\max}$. ∎

The average delay mainly depends on the traffic pattern λ_i of the number of GTS requests and protocol parameters (BO, SO) of $\Delta_u, D_{i,j,t}$. It is possible to consider the average delay constraint by using the specific queue size $B_{\max}$.

C. Throughput

Here we characterize the GTS throughput, namely, the average amount of packets that can be transmitted during a GTS.

Let $P_s(t)$ be the probability that a GTS allocation is successful at the superframe t. Then

$$P_s(t) = 1 - P_{\text{drop}}(t) \qquad (26)$$

$$= 1 - \frac{\mathbb{E}\left[r_d(t)\right]}{\sum_{k=1}^{L_{\max}}\lambda_k k\left(\sum_{i=0}^{B_{\max}}\pi^t_i + \pi^t_{B^*_{\max}}\right)}\,,$$

where $P_{\text{drop}}(t)$ is the drop probability due to the limited queue size $B_{\max}$ at the superframe t. Note that $P_{\text{drop}}(t)$ is the ratio between the mean number of dropped requests, given by Eq. (18), and the mean number of total requests at the superframe t. As a number of requests increase, it increases the length of waiting queue and results on higher dropped probability. If we assume that the frame size D_{FS} is smaller than T_{cfp}, then the normalized system throughput $\text{S}(t)$ of the superframe t is given by the ratio of the average length of successfully allocated payload in a GTS time slot to the average length of a GTS time slot, namely

$$\text{S}(t) = \frac{P_{\text{S}}(t)\,L_{\text{pl}}\,\tau_n}{\theta_{\min}\,T_{\text{SS}}}\,,\qquad(27)$$

where L_{pl} is the length of payload of each data packet, τ_n is the number of data packets, T_{SS} is the length of superframe slot, and $\theta_{\min}$ is the minimum number of superframe slots, which are given by (4) and (5), respectively. The normalized system throughput $\text{S}(t)$ depends on the traffic pattern since the drop probability $P_{\text{drop}}(t)$ is related to the number of

data packets τ_n, the frame size L_{fr}, the mean and variance of requests. Hence, the system throughput is related to the time effectively used for data transmission within a GTS. It is possible to derive the optimal protocol parameters (BO, SO) which maximize the throughput of GTS usage.

V. NUMERICAL RESULTS

Here we present extensive Monte Carlo simulations of the GTS allocation to validate our theoretical results, which we then use for a performance analysis. The simulations are based on the specifications of the IEEE 802.15.4 [1], with beacon order set equal to superframe order, namely BO=SO. In the simulations, the number of GTS requests for each superframe follows some different probability density functions. In particular, the simulation experiments are obtained with Poisson, Normal, and Gamma distribution. We considered the Gamma distribution because many other distributions can be approximated by it. Furthermore, we investigated the effects of the protocol parameters (BO, SO) in terms of throughput and delay. Details follows in the sequel.

A. Validation

We validated the average number of requests, average delay of GTS allocation, the average number of dropped requests and the queue overflow probability as the time progress. Recall that the average number of dropped requests and the queue overflow probability are the important metrics for the stability of queue management.

Figs. 5(a), 5(b) compare the average number of waiting requests given by Eq. (17) and the average delay of requests given by Eq. (25) with simulation results for different distributions of the number of GTS requests, respectively. The analytical model follows well the simulation results. By comparing Fig. 5(a) and Fig. 5(b), we confirm the strong dependence between the average number of requests and the average delay due to a FCFS fashion of the queue management. As a number of GTS requests increase during the CAP, the length of waiting queue and average delay increase.

Figs. 6(a), 6(b) report the average number of dropped requests given by Eq. (18) and the overflow probability as obtained by Eq. (19) with simulations, respectively. Observe that our analytical model matches well the average overflow probability for every distribution of the number of GTS requests. Therefore, we conclude that we can apply the average number of dropped requests and the overflow probability to analyze the stability of the queue management, as we see next. In addition, since a number of GTS requests depend on the random access scheme of CAP, it will be interesting to investigate the impact of reliability in CAP to the stability issue of queue management of GTS allocation.

B. Effect of Beacon and Superframe Order

In this section, we investigate the impact of beacon order on the average throughput and delay when the Poisson distribution is assumed for the number of GTS requests. We remark here that similar behaviors as those investigated by using the Poisson distribution are observed by adopting the Normal and Gamma distributions.
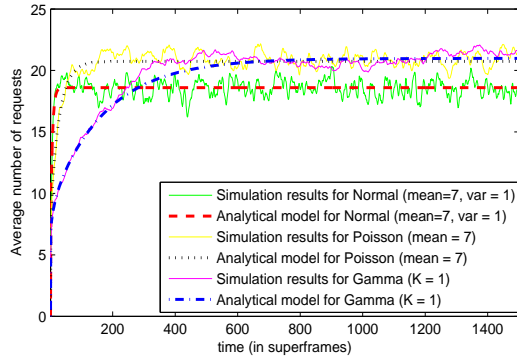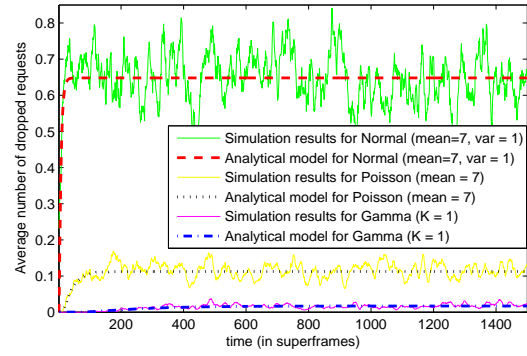
(a) Average number of requests
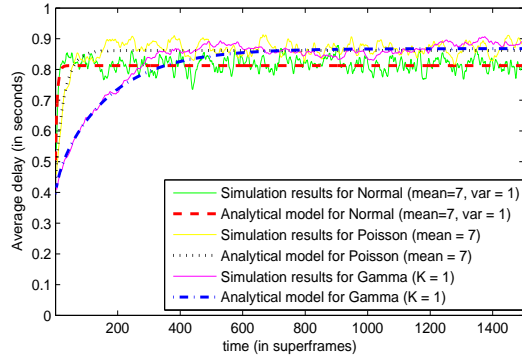
(b) Average delay of GTS allocation

Fig. 5. Average number of waiting requests (a) and average delay of requests (b) as obtained by simulations and Eqs. (17) and (25), respectively. The length of the payload is $L_{\mathrm{pl}} = 40$ bytes, the number of data packets is $\tau_n = 3$, the beacon order is BO $= 4$, the mean number of requests is 7, the variance of requests is 1 for the Normal distribution, the mean number of requests is 7 for the Poisson distribution, and the shape and scale of the Gamma distribution is equal to 1 and 7, respectively.

(a) Average number of dropped requests

(b) Queue overflow probability

Fig. 6. Stability of the queue management as obtained by simulations and analytical model of Eqs. (18) and (19). The length of payload is $L_{\mathrm{pl}} = 40$ bytes, the number of data packets is $\tau_n = 3$, the beacon order is BO$= 4$, the mean number of request is 7, the variance of requests is 1 for the Normal distribution, the mean number of requests is 7 for the Poisson distribution, and the shape and scale for the Gamma distribution are 1 and 7, respectively.
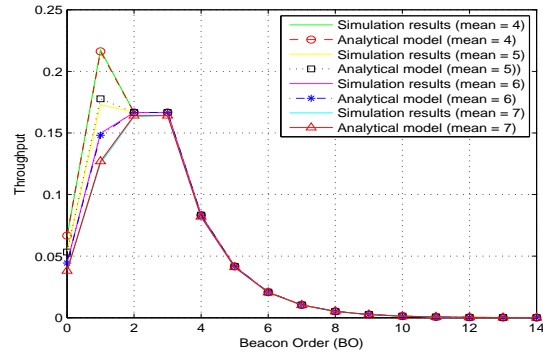
We evaluated the average throughput of one GTS given in Eq. (27) for different beacon order values, as a function of the mean of the Poisson distribution, whereas the length of data payload L_{pl}, and the number of data packets τ_n are fixed.

Fig. 7 shows the effects of beacon order on the throughput as a function of the mean number of requests of the Poisson distribution, and for a fixed length of payload ($L_{\mathrm{pl}} = 40$ bytes), a fixed number of data packets ($\tau_n = 2$). Observe that the GTS allocation mechanism presents a worse behavior in terms of throughput for BO>3. Indeed, high beacon orders result on increased amount of wasted bandwidth of an allocated GTS. Furthermore, if BO$\leq$1 and the mean number of requests is greater than 6, then a decreasing throughput occurs due to the increasing dropped requests. Note that an increasing number of requests increases the length of waiting queue and results on higher dropped probability, see Eq. (26). In addtion, it will be very interesting extension to define the global throughput by considering the data transmission both CAP and CFP.

Fig. 8(a) illustrates the throughput as achieved by the analytical model and simulations as a function of the payload length L_{pl}. The good configuration of the beacon order that maximizes the throughput is around BO=2, 3 for a given duty cycle. However, the maximum throughput value depends on the length of payload. Note that such a dependance is

Fig. 7. Effect of beacon order on the throughput of GTS allocation as a function of the mean of the Poisson distribution $(4, 5, 6, 7)$. The length of the payload is $L_{\mathrm{pl}} = 40$ bytes, and the number of data packets is $\tau_n = 2$.

very limited for BO>10 due to wasted bandwidth. Fig. 8(b) shows the throughput as obtained by the analytical model and simulations as a function of the number of data packets τ_n. The analysis predicts quite well the simulation results. The best configuration of the beacon order that maximizes the throughput is a function of the number of data packets.

Fig. 9 shows the average delay of a GTS allocation, for a given length of payload and number of data packets as a function of the mean of the Poisson distribution. It is clear that the average delay depends on the mean number of requests.
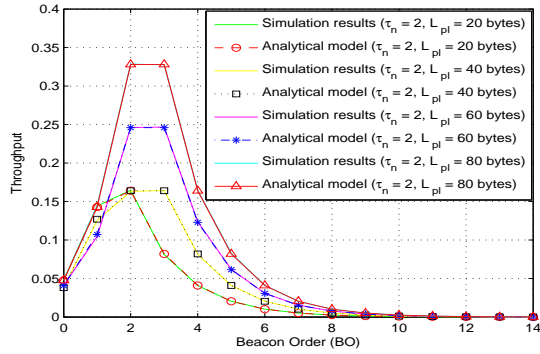
(a) Throughput as a function of L_{pl} $(20, 40, 60, 80$ bytes$)$, the number of data packets $\tau_n = 2$.
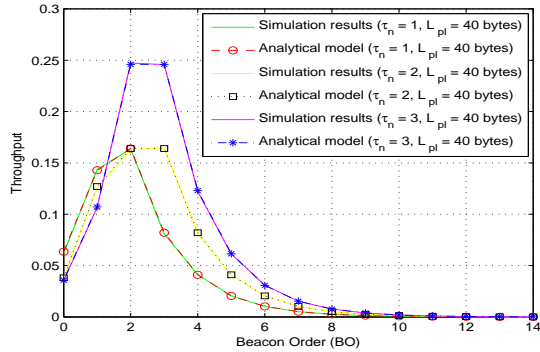
(b) Throughput as a function of τ_n $(1, 2, 3)$, the length of payload $(L_{pl} = 40$ bytes$)$.

Fig. 8. Effect of the beacon order on the throughput of GTS allocation. The mean number of requests of Poisson distribution is 7.
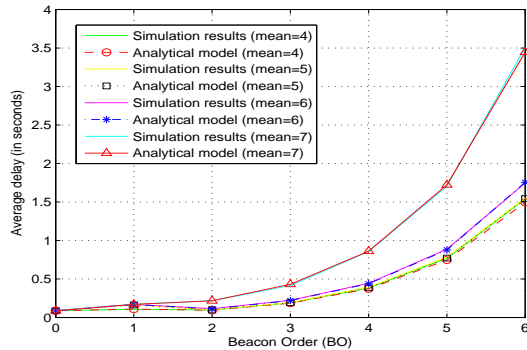

Fig. 9. Effect of beacon order on the average delay of GTS allocation as a function of the mean of the Poisson distribution for the number of GTS requests $(4, 5, 6, 7)$, for a fixed length of the payload $(L_{pl} = 40$ bytes$)$ and if the number of data packets $\tau_n = 2$.

Notice that increasing the mean number of requests increases also the mean delay because of the longer waiting time, see Fig. 5. When the maximum number of requests is greater than 7, the average delay is monotonically increasing, i.e., we see an unstable queue size because the maximum number of GTSs per superframe is 7. The monotonic relation between average delay and BO does not hold for BO = 2, 3 of the mean number of GTS requests (5, 6). This behavior is explained by looking at the maximum number of GTSs Δ_u given by Eq. (7). Note that a lower BO results on a decreasing of the maximum number of GTSs Δ_u. By observing the traffic pattern of

the number of GTS requests, it is possible to optimize the protocol parameters (BO, SO) which guarantee the average delay constraint.

VI. CONCLUSIONS

In this paper, we presented an analytical model based on a Markov chain to compute the performance of the GTS allocation mechanism in IEEE 802.15.4 standard. Monte Carlo simulations validated the analysis. Our theoretical analysis gives accurate numerical results, which are different from the ones obtained in [10] by using the network calculus.

We evaluated the stability of the queue size at the network coordinator, the delay to serve a GTS request, and the achieved throughput for different traffic patterns and protocol parameters. We derived the dependance of the average delay and queue size as a function of the number of requests. Furthermore, we analyzed the achieved throughput as a function of the amount of data packets to forward for each request. We observed that lower beacon order gives lower delay but ensures a worse throughput because of the higher drop probability. By contrast, higher beacon order increases significantly the average delay and degrades the throughput due to wasted bandwidth.

Future work includes the model extension by considering the random access scheme of CAP and the modification of GTS allocation instead of current inefficient FCFS fashion.

REFERENCES

[1] *IEEE Std 802.15.4-2996, September, Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)*, IEEE, 2006. [Online]. Available: http://www.ieee802.org/15

[2] G. Lu, B. Krishnamachari, and C. Raghavendra, "Performance evaluation of the IEEE 802.15.4 MAC for low-rate low-power wireless networks," in *IEEE IPCCC*, Apr 2004.

[3] N. Timmons and W. Scanlon, "Analysis of the performance of IEEE 802.15.4 for medical sensor body area networking," in *IEEE SECON*, Oct 2004.

[4] L. Cheng, X. Zhang, and A. G. Bourgeois, "GTS allocation scheme revisited," *Electronics Letters*, 2007.

[5] J. K. Song, J. D. Ryoo, S. C. Kim, J. W. Kim, H. Y. Kim, and P. S. Mah, "A dynamic GTS allocation algorithm in IEEE 802.15.4 for QoS guaranteed real-time applications," in *IEEE ISCE*, 2007, pp. 1–6.

[6] S. E. Yoo, D. Y. Kim, M. L. Pham, Y. M. Doh, E. C. Choi, and J. D. Huh, "Scheduling support for guaranteed time services in IEEE 802.15.4 low rate WPAN," in *IEEE RTCSA*, 2005, pp. 400–406.

[7] C. Na, Y. Yang, and A. Mishra, "An optimal GTS scheduling algorithm for time-sensitive transactions in IEEE 802.15.4 networks," *Comput. Netw.*, pp. 2543–2557, 2008.

[8] Y. Huang, A. Pang, and H. Hung, "An adaptive GTS allocation scheme for IEEE 802.15.4," *IEEE Trans. Parallel Distrib. Syst.*, pp. 641–651, 2008.

[9] A. Koubaa and 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)*, pp. 2899–2913, July 2004.

[10] A. Koubaa, M. Alves, and E. Tovar, "GTS allocation analysis in IEEE 802.15.4 for real-time wireless sensor networks," in *IEEE IPDPS*, 2006, pp. 25–29.

[11] ——, "Energy and delay trade-off of the GTS allocation mechanism in IEEE 802.15.4 for wireless sensor networks," *International Journal of Communication Systems*, 2006.

[12] ——, "i-GAME: An implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks," in *IEEE ECRTS*, 2006.

[13] D. Gross and C. M. Harris, *Fundamentals of Queueing theory*. Wesley inter-science, 1998.