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

X X
Distributed Computer Systems Lab
RPTU Kaiserslautern-Landau
Kaiserslautern, Germany
X@rhrk.uni-kl.de

***Abstract*—This scientific report is about the work of Koubâa et al., titled "GTS Allocation Analysis in IEEE 802.15.4 for Real-Time Wireless Sensor Networks". It focuses on the analysis of the GTS (Guaranteed Time Slot) allocation in IEEE 802.15.4 for real-time wireless sensor networks. The study focuses on the beacon-enabled mode of the protocol, which provides synchronization and time guarantees necessary for real-time applications. The analysis employs Network Calculus, a mathematical framework for performance analysis in computer networks, to determine upper bounds on delay. The paper assumes a linear arrival curve for GTS traffic and derives the service curve for a GTS, which exhibits a stair function pattern. The authors approximate the stair function with a rate-latency service curve to facilitate the analysis. The delay bounds for GTS allocation are calculated based on the arrival curve and service curve. The paper also investigates the influence of the beacon and superframe order on GTS allocation. The results of the analysis provide insights into the delay performance of GTS allocation in IEEE 802.15.4 clusters. Additionally, the paper briefly discusses the author's personal experiences of working with AI in the context of the research.**

***Index Terms*—IEEE 802.15.4, Network Calculus, Delay Bound Analysis, CSMA/CA, Working With AI**

I. Introduction

In this report, we discuss the analysis of GTS (Guaranteed Time Slot) allocation in IEEE 802.15.4 for real-time wireless sensor networks. The work by Koubâa et al., titled "GTS Allocation Analysis in IEEE 802.15.4 for Real-Time Wireless Sensor Networks" serves as the primary reference for this report unless explicitly mentioned and referenced otherwise.

The IEEE 802.15.4 protocol is a widely adopted standard for low-power and low-rate wireless personal area networks (WPANs). It offers efficient and reliable communication for resource-constrained devices, making it a popular choice for Internet of Things (IoT) applications. The protocol operates in two main operational modes: non-beacon-enabled mode and beacon-enabled mode. In this report, we focus on the beacon-enabled mode, which provides synchronization and time guarantees necessary for real-time applications.

The beacon-enabled mode operates based on beacon intervals, which consist of superframes divided into time slots. The network coordinator periodically transmits beacon frames to synchronize all devices within the network. The superframe is divided into a Contention Access Period (CAP) and a Contention-Free Period (CFP). During the CAP, nodes contend for medium access using slotted CSMA/CA. In the CFP, time slots called GTSs are allocated to nodes by the PAN (Personal Area Network) coordinator for guaranteed transmission.

The goal of this report is to analyze the delay bounds of GTS allocation in IEEE 802.15.4 clusters. To achieve this, we employ the principles of Network Calculus, a mathematical framework for performance analysis in computer networks. Network Calculus allows us to determine upper bounds on delay and other performance metrics based on deterministic arrival and service models.

In the analysis conducted by Koubâa et al., a linear arrival curve is assumed for the GTS traffic, which is appropriate for periodic traffic patterns. The authors derive the service curve for a GTS, which exhibits a stair function pattern. To facilitate the analysis, they approximate the stair function with a rate-latency service curve. Using the arrival curve and service curve, the delay bounds for GTS allocation are calculated.

The report first gives a short insight into the theoretical background in Section II. Subsequently, the delay bound analysis of the GTS allocation is considered (Section III) and further the influence of the Beacon and Superframe order is investigated in Section IV. After that we introduce related work (Section V) and briefly describe the results of other studies. In section VI follows the conclusion and future work. The report ends with the personal experiences of working with AI, which are described in Section VII.

II. Theoretical Background

In this section we give the theoretical background that is important for understanding the further research. For this purpose, we first go into more detail about the IEEE 801.15.4 protocol (Subsection II-A) and then about the basics of the network caclulus (Subsection II-B).

A. IEEE 802.15.4 Protocol

The IEEE 802.15.4 protocol is a widely adopted standard for low-power and low-rate wireless personal area networks (WPANs). It offers efficient and reliable communication for resource-constrained devices, making it a popular choice for Internet of Things (IoT) applications. Operating in the 2.4 GHz band with a data rate of up to 250 kbps, it supports both star and peer-to-peer topologies. With its energy-saving mechanisms and features, the IEEE 802.15.4 protocol is well-suited for applications such as smart homes, industrial automation, and healthcare monitoring. Its widespread use and ability to address the challenges of low-power wireless communication have cemented its position as a fundamental building block in IoT and wireless sensor networks.

The IEEE 802.15.4 protocol operates in two main operational modes: non-beacon-enabled mode and beacon-enabled mode. These modes may be selected by a central node called *PAN coordinator*.

In the *non-beacon-enabled mode*, devices in the network operate independently without strict synchronization. Each device can transmit data at any time, resulting in a more decentralized and asynchronous communication approach. This mode is suitable for applications where devices have varying traffic patterns or low-duty cycle requirements, as it allows for flexible and energy-efficient communication.

In contrast, the *beacon-enabled mode* introduces a synchronized approach to communication. In this mode, the network coordinator periodically transmits beacon frames to synchronize all devices within the network. These beacons establish a common time reference and provide information about the network's parameters. Devices in the network then schedule their communication based on these beacons, resulting in a more synchronized and predictable communication pattern. This mode is beneficial for applications that require strict timing, periodic data exchanges, or time-sensitive operations.

In the paper of Koubâa et al. only the beacon-enabled mode was considered, since it has the capability to ensure timeliness guarantees for the network. In beacon-enabled mode, the *Beacon Interval* (BI) governs the timing of consecutive beacons. It consists of an active period, known as the *superframe*, and an optional inactive period. The superframe is divided into 16 equal time slots for frame transmissions, while the inactive period allows nodes to enter a sleep mode. [1]

The length of such a beacon interval and a superframe can be determined or calculated with the help of two parameters. The beacon interval is defined by the so-called *Beacon Order* (BO) as follows [1]:

$$\begin{aligned} BI =& aBaseSuperframeDuration \cdot 2^{BO}, \\ & \text{for } 0 \leq BO \leq 14. \end{aligned} \tag{1}$$

The Superframe Duration can be defined by the *Superframe Order* (SO) parameter:

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

In Equations (1) and (2), the respective minimum length of a superframe is given by *aBaseSuperframeDuration* (for $SO = 0$). The minimum length is fixed by the IEEE 802.15.4 standard to 3840 bits (960 symbols, each corresponding to 4 bits). We further assume a rate of 250kbps on the 2.4 GHz band. With this assumption, the duration of a minimum superframe corresponds to 15.36ms. [1]

The superframe can also be divided into two periods again. There is the *Contention Access Period* (CAP) and the *Contention-Free Period* (CFP), see Figure 1. During CAP, there is contention among the nodes for medium access via slotted CSMA/CA.

If a node wants to be allocated time for optinal CFP at the PAN coordinator, it submits a request to the coordinator. After checking the available resources, the time slots are then allocated. These time slots are called *Guaranteed Time Slots* (GTSs).

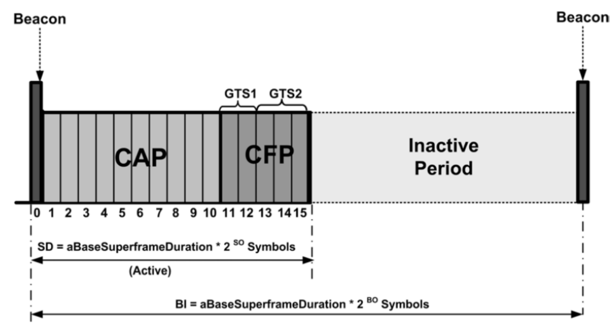

Fig. 1. Concepts: beacon interval and superframe [1]

B. Network Calculus: Delay Bound Analysis

Network Calculus is a mathematical framework that enables the analysis and prediction of performance guarantees in computer networks. It provides a systematic approach to determine upper bounds on delay, backlog, and other performance metrics based on deterministic arrival and service models. By employing mathematical techniques, deterministic network calculus allows for the evaluation and optimization of network performance, aiding in the design and provisioning of reliable and predictable communication systems.

Assuming some basic knowledge of Network Calculus, we define here only the most necessary principles. In the Network Caculus, we assume, for a cumlative arrival function $R(t)$, the following two definitions [1]:

1) there exists an arrival curve $\alpha(t)$, s.t.

$$\forall s, 0 \leq s \leq t, R(t) - R(s) \leq \alpha(t - s),$$

2) there exists a minimum service curve $\beta(t)$, which describes a guaranteed minimum service for the arrivals in $R(t)$.

Examples of the above two definitions are as follows:

1) $\alpha(t) = b + r \cdot t$, with b being a burst event and r the rate of the arrivals (linear arrival curve),

2) $\beta_{R,T}(t) = R(t-T)^+$, with $R \geq r$ being the guaranteed bandwidth and T the maximum service latency (*rate-latency* service curve).

The delay bound D_{max} for a data flow with an assumed arrival curve $\alpha(t)$ and service curve $\beta(t)$ can be calculated as follows [1]:

$$D_{max} = \sup_{s\geq 0}\Big\{\inf\big\{\tau \geq 0 : \alpha(s) \leq \beta(s+\tau)\big\}\Big\}. \quad (3)$$

For the examples above (linear arrival curve and rate-latency service curve), the delay bound D_{max} looks like this (by applying Equation (3)) [1]:

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

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

In the paper [1], a delay bound analysis is done for one time slot GTS. For this, a linear arrival curve $\alpha(t) = b + r \cdot t$ (see definition above in Subsection II-B) is assumed, which is suitable for this problem according to [2] (since they could show that periodic traffic, with or without jitter, can be expressed as a linear arrival curve). This means that for the delay bound analysis "only" the appropriate service curve $\beta(t)$ has to be found (which in general is not a trivial problem).

It could be shown that we get a stair function as a "real" service curve for a GTS (compare Figure 2). For this we assume the following or rather the following is given:

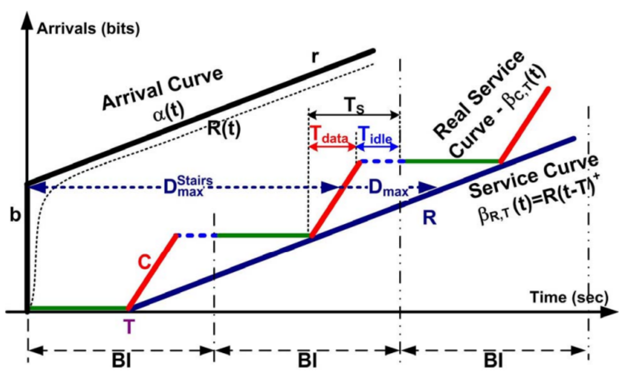

Fig. 2. GTS service curves ("real" and approximation) [1]

- let C be the total data rate of the output link ($C = 250$ kbps in our case),
- let Ts be the time slot duration in a superframe, then it holds that $Ts = \frac{SD}{16}$, with SD being the Superframe Duration from Equation (2),
- since the time for one time slot Ts is not used for data transmission only, we have to define that $Ts = T_{data} + T_{idle}$, with T_{data} being the time used for sending data and T_{idle} is the time for at least IFS (InterFrameSpacing),
- let BI be the Beacon Interval defined in Equation (1),
- let $DC = \frac{SD}{BI}$ be the so-called duty cycle which is the length of the superframe in relation to the beacon interval,
- let w_{idle} be the amount of unused bandwidth into a GTS (because of the idle time).

For this, in the paper is derived that for k superframes in a row the service curve is given by:

$$\beta^k_{C,T}(t) = \begin{cases} (k-1)\cdot C \cdot T_{data} + C(t-(k\cdot BI - Ts))^+ \\ \quad \forall t, (k-1)\cdot BI \leq t \leq k \cdot BI - T_{idle} \\ \\ 0 \qquad \text{Otherwise} \end{cases} \quad (5)$$

The overall service curve is given by:

$$\beta^{stair}_{C,T}(t) = \sum_k \beta^k_{C,T}(t) \quad ,\forall t \quad (6)$$

This "real" service curve was then replaced by an approximation of it. For this purpose, the stair function is approximated by a rate-latency service curve (see Figure 2).

This means that the service curve for a GTS for a derived (this was formally shown in the paper [1]) bandwidth $R = \frac{1}{16}\cdot DC \cdot C - w_{idle}$ and latency $T = BI - Ts$ can be expressed by $\beta_{R,T} = R(t-T)^+$ (see Subsection II-B).

Now that both the arrival curve and the service cruve are known, the delay bound can be calculated by applying Equation (3). For the approximation, Equation (4) yields the following delay bound:

$$D_{max} = \frac{b}{\frac{1}{16}\cdot DC \cdot C - w_{idle}} + (BI - Ts).$$

For the "real" stair service curve, the paper specifies the following delay bound:

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

At the end it is additionally generalized to n Time Slots GTS which gives the following Delay Bound:

$$D^{stair}_{n,\max} = \Bigg(\frac{b}{C} + (k+1)\cdot BI - \\ n\cdot(Ts + k \cdot T_{data}) + m \cdot T_{idle}\Bigg)$$

$$\text{, where } m = \left\lfloor \frac{b - k\cdot(n \cdot T_{data})\cdot C}{T_{data}\cdot C} \right\rfloor$$
$$\text{if } k \cdot C \cdot (n \cdot T_{data}) < b \leq (k+1)\cdot C \cdot (n \cdot T_{data}).$$

IV. The Influence of Beacon and Superframe Orders

In this section we deal with the results of the influence of the Beacon (BO) and Superframe Order (SO) on the maximum GTS throuput (Subsection IV-A) and on the delay bound (Subsection IV-B).

A. Maximum GTS Throuput

We now consider the results of the maximum throughput evaluation. The *maximum throughput* of a GTS is defined as the upper limit of data transmission capacity (bandwidth) that can be utilized in a GTS [1].

According to the paper [1] the maximum throughput can be specified as follows (using the definitions from above):

$$Th_{max} = \frac{T_{data}}{BI} \cdot C. \tag{7}$$

Since BI and C are known, T_{data} must now be calculated in order to calculate Th_{max}.

In the detailed derivation in the paper [1], we get the following result for T_{data}:

$$Th_{data} = \min\Bigg(\frac{b + r \cdot Ts}{C}, \max\begin{pmatrix} Ts - (N_{LIFS} - 1) \cdot LIFS - \Delta(IFS), \\ Ts - N_{SIFS} \cdot SIFS \end{pmatrix}\Bigg) \tag{8}$$

This results in the following maximum throughput Th_{max} (according tho Equation (7)):

$$Th_{max} = \min\Bigg(\frac{b + r \cdot Ts}{BI}, \max\begin{pmatrix} Ts - (N_{LIFS} - 1) \cdot LIFS - \Delta(IFS), \\ Ts - N_{SIFS} \cdot SIFS \end{pmatrix} \frac{C}{BI}\Bigg) \tag{9}$$

The numerical results for the maximum throughput of a GTS allocation (as a function of the average arrival rate and of the burst size), which are presented in the paper can be recreated by applying Equation (9). For this we need to set the values for the variables used. For this they used the following settings:

- Link Capacity:
  $C = 250.000$ bit/s,
- Base Superframe Duration:
  $aBaseSuperframeDuration = 3840$ bit,
- Length SIFS: 48 bit,
- Length LIFS: 160 bit,
- Maximum Frame Length:
  $aMaxPHYPacketSize = 1016$ bit,
- Maximum Frame Size using SIFS:
  $aMaxSIFSFrameSize = 144$ bit

The verification and reconstruction of the numerical results revealed identical results to those of the paper (see Figure 3 and Figure 4 below, which were created by us):

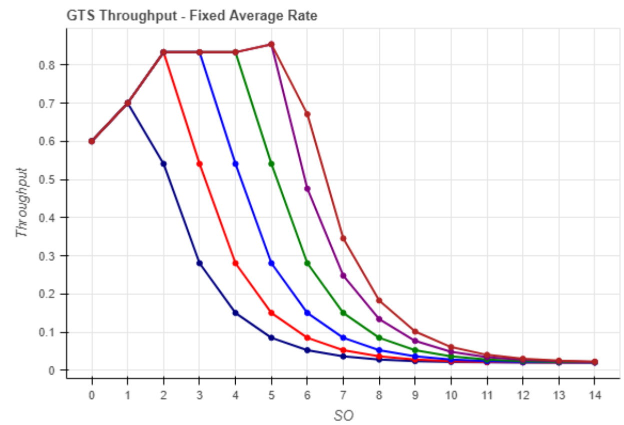

Fig. 3. Maximum throughput: GTS Allocation as a function of the average arrival rate

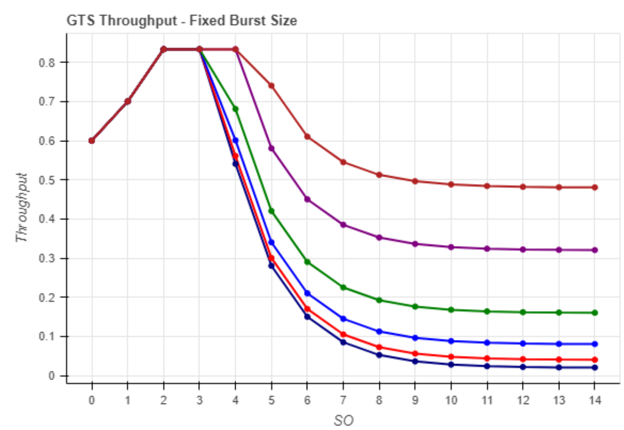

Fig. 4. Maximum throughput: GTS Allocation as a function of the burst size

B. Delay Bound

In this part we will now take a look at the influence of the Beacon and Superframe Order on the Delay Bound. The problem, that is focused on in this Subsection, involves finding the optimal arrangement of the superframe structure (SO) to minimize the delay constraint, given a specific duty cycle. For a duty cycle of 100% and an average arrival rate of 5 kbps we get the following results in Figure 5 for the delay bound (as a function of the burst size).

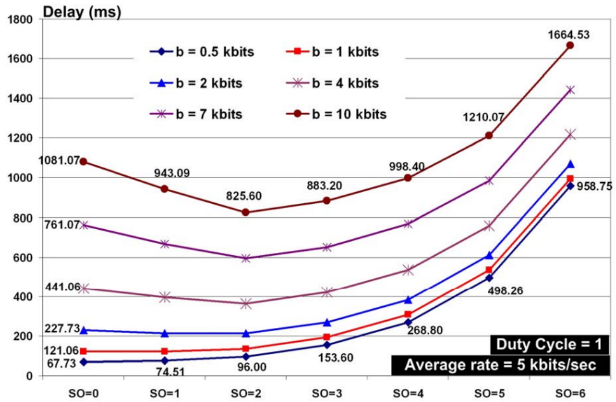

Fig. 5. Impact of SO on the delay bound [1]

In paper [1] it was described for Figure 5 that the lowest delay bound in the superframe order depends on the burst size. For small burst sizes (0.5 kbits, 1 kbits), the delay bound increases with the superframe order. In this case, the latency impact is more significant than the guaranteed bandwidth on the delay bound, making superframe order $SO = 0$ the best choice for the lowest delay bound.

For larger burst sizes (2, 4, 7, and 10 kbits), the relationship between the delay bound and the superframe order becomes non-monotonic. In these cases, the delay bound does not consistently increase or decrease with the superframe order. Instead, lower delays are observed with superframe orders higher than 0.

To illustrate this, consider a burst size of 10 kbits. The lowest delay bound is achieved with a superframe order of $SO = 2$. Surprisingly, the delay bounds for superframe orders $SO = 3$ and $SO = 4$ are even lower than that guaranteed with $SO = 0$. This phenomenon can be attributed to the influence of the guaranteed bandwidth on the delay bound.

When the burst size is relatively high, the impact of the guaranteed bandwidth (R) outweighs the effect of latency ($BI - Ts$), particularly for the first three superframe orders. The lower guaranteed bandwidth for $SO = 0$ results in a higher delay bound compared to $SO \in \{1, 2, 3, 4\}$ when the burst size is 10 kbits.

In summary, the choice of the optimal superframe order depends on the specific burst size and application requirements. For WSN applications with small burst sizes and low data rates, superframe order $SO = 0$ is generally the most suitable option, as it offers reduced latency for real-time guarantees. However, when the burst size is relatively high (more than 1 kbits), superframe order $SO = 2$ provides better timeliness guarantees due to its higher throughput. [1]

V. Related Work

As we have seen in the previous sections, we were able to specify the worst case delay bound for GTS in IEEE 802.15.4 and to study it. Since this is only one part of the superframe and the CAP is not covered by this analysis, we now look at the performance results of the CSMA/CA protocol used in the CAP. But before this the CSMA/CA mechanism is briefly explained.

CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) is a fundamental mechanism utilized in wireless networks, particularly in IEEE 802.11-based Wi-Fi networks, to regulate access to a shared transmission medium. Its primary objective is to prevent collisions that occur when multiple devices attempt to transmit simultaneously, causing data corruption and loss. Unlike CSMA/CD, which focuses on collision detection, CSMA/CA emphasizes collision avoidance.

The CSMA/CA mechanism involves a series of steps to ensure efficient and collision-free data transmission:

- Carrier Sense: Before initiating a transmission, a device employing CSMA/CA senses the wireless medium to detect ongoing transmissions. If the medium is idle, the device proceeds; otherwise, it waits for the medium to become available.
- Interframe Space (IFS): Upon identifying an idle medium, the device waits for a predefined interframe space. The duration varies depending on the frame's priority, with higher priority frames assigned shorter IFS durations.
- Random Backoff: Following the IFS, the device enters a random backoff period. It selects a random duration from a specific range and waits before attempting to transmit. Randomization helps avoid collisions caused by simultaneous completion of multiple devices' backoff periods.
- Carrier Sense During Backoff: While in the backoff period, the device continuously monitors the medium. If the medium becomes busy, indicating another device has started transmitting, the backoff timer freezes until the medium becomes idle again. The timer resumes counting down once the medium is clear.
- Transmission: When the backoff timer reaches zero without medium activity, the device assumes it can transmit. It initiates the data transmission and awaits an acknowledgment from the receiving device.
- Acknowledgment: After data transmission, the receiving device sends an acknowledgment (ACK) frame to confirm successful reception. If no ACK is received within a specified period, the sender assumes a collision occurred and triggers the retransmission process.

By managing access to the shared medium and introducing randomization in the backoff process, CSMA/CA mitigates collisions, resulting in improved efficiency and reliability in wireless network communications.

Besides using the 2.4GHz band, there is also the possibility to use two other frequencies. These are the 868 MHz frequency band on the one hand and the 915 MHz frequency band on the other. An analysis of the performance of CSMA/CA with regard to the use of different frequency bands was presented in the paper ”Evaluation of Slotted CSMA/CA of IEEE 802.15.4” by Alvi, Ahmad Naseem, et al. [3].

For this purpose, the reliability, propagability of channel access failure, throughput analysis and transmission failure probability were investigated for the different frequency bands.

It could be shown that the 2.4 GHz prevailed in all investigations and provided the better results in each case [3].

This is also the case in the analysis of the throughput (see results in Figure 6), where the throughput for Figure 6 in [3] was defined as follows: ”Throughput is defined as time required for successful transmission of packets from source to destination Throughput against applied load is calculated by simply multiplying the probability of successful transmitted packets with the applied load of 10 nodes.” (Alvi, Ahmad Naseem, et al., 2012, p.6).

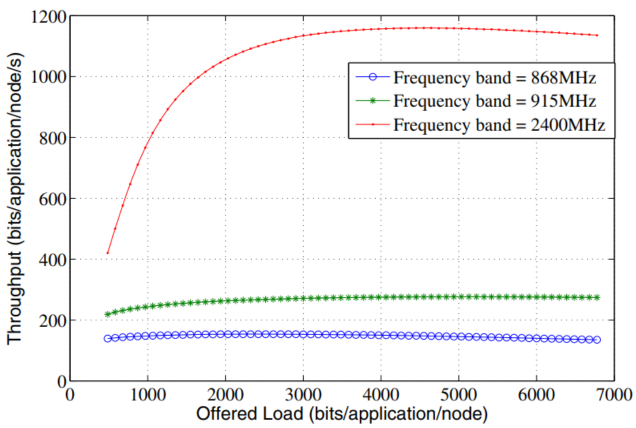

Fig. 6. Analysis of the throughput of different frequency bands [3]

Further interesting research of CSMA/CA for IEEE 802.15.4 has been done in the following report ”A Comprehensive Simulation Study of Slotted CSMA/CA for IEEE 802.15.4 Wireless Sensor Networks” by Koubaa, Anis, Mário Alves, and Eduardo Tovar [4]. Here, the approach to investigate CSMA/CA with the help of simulations was used. In [4] it could be shown that for lower superframe orders the throughput in the CAP becomes lower.

VI. Conclusion and Future Work

The paper [1] proposes a methodology to analyze the Guaranteed Time Slot (GTS) mechanism in the IEEE 802.15.4 protocol. The authors introduce two accurate models for the service curve of a GTS allocation and derive the corresponding delay bounds. They also present an expression linking the duty cycle to the delay. The study investigates the impact of beacon order and superframe order on maximum throughput and delay bounds.

In summary, for WSN applications with low arrival rates and small burst sizes, using low superframe orders is preferable for achieving low delay bounds. However, this approach sacrifices efficient utilization of the GTS capacity due to shorter durations and the influence of IFS. The authors hope that this work represents a first step towards effectively utilizing the GTS mechanism in IEEE 802.15.4-compliant networks.

One option for future work would be to perform the investigation using the stochastic network calculus, which might allow us to get even tighter bounds with a small violation probability.

VII. Personal Experiences Writing Scientific Texts with the Help of AI

This complete section was written by Chat GPT. Only this one prompt was used for it:

” *Write a big section for a scientific text.*

This section is about personal experiences writing scientific texts with the help of AI. Only use the following experiences that were made:

- *it was possible to let chatGPT write the whole introduction and abstract section, fot this we asked about ”write an abstract for the scientifiy paper given by this latex document: [latex code]” and ”for the given latex document write me a introduction section: [latex code]”. Die Ergebnisse dazu sind sehr beeindruckend!*
- *for basic definitions it is great to let the AI write the whole text (e.g. ”give me a short overview of [...]”),*
- *if you want to summarize sections and use other formulations and / or synonyms (e.g. ”summarize and reformulate [...]”), the help of the AI is also very good (only worked well for pure texts without mathematical formulas),*
- *working with references was not good at all, because you got references when using Chat GPT via Microsoft Edge, which partly did not exist at all and / or were incorrect,*
- *However, the work with the AI was overall positive, because it relieves you of the work of finding good formulations and helps you not to make spelling mistakes. In addition, the answers can always be adapted to the wishes, which makes them more and more individual (which we consider positive).*

At the end (but before the conclusion), one more paragraph may be written about others' experiences of working in a scientific context with ChatGPT. ”.

The use of AI, particularly ChatGPT, has significantly impacted the process of writing scientific texts in our experiences. One notable achievement was the ability to generate the introduction and abstract sections entirely with the help of ChatGPT. By simply providing the AI with the LaTeX code of the paper, we requested it to ”write an abstract for the scientific paper given by this LaTeX document” or ”write an introduction section for the given LaTeX document.” The results were astonishingly impressive. The AI produced coherent and well-structured abstracts and introductions, relieving us of the burden of crafting these crucial sections from scratch. This feature alone has saved us considerable time and effort, allowing us to focus on other aspects of our research.

Another valuable aspect of AI assistance lies in the generation of basic definitions. When we needed a concise overview of a particular concept or term, we simply asked the AI to ”give us a short overview of [the concept].” In response, ChatGPT provided accurate and succinct explanations, helping us to clarify complex ideas or terms in a concise manner. This capability proved to be particularly useful when introducing technical concepts to readers unfamiliar with the subject matter.

Additionally, the AI's ability to summarize and reformulate sections of text has been highly beneficial. By requesting the AI to "summarize and reformulate" specific sections, we were able to obtain alternative phrasings and concise summaries that enhanced the overall clarity and readability of our scientific texts. This feature has proven effective for non-technical sections of the paper but was less successful when dealing with mathematical formulas or equations.

However, one limitation we encountered while working with AI was the handling of references. When using ChatGPT via Microsoft Edge, the references provided were often incorrect or nonexistent. This issue hindered the process of including accurate and relevant citations in our scientific texts. As a result, we had to resort to traditional methods of obtaining references manually, which somewhat diminished the overall efficiency of the AI assistance.

Nevertheless, despite this limitation, our overall experience with AI in scientific writing has been overwhelmingly positive. The AI technology not only alleviates the laborious task of finding optimal formulations but also assists in avoiding spelling mistakes. Furthermore, the answers generated by ChatGPT can be easily customized and adapted to meet individual preferences, adding a personal touch to the text. This flexibility allows us to fine-tune the responses to align with our specific requirements, making the AI's assistance more tailored and individualized—an aspect we consider highly beneficial.

In addition to our own experiences, other researchers have reported positive outcomes when employing AI tools in scientific contexts. The ability to quickly generate coherent and precise scientific texts has been praised for streamlining the writing process, enabling researchers to focus more on the core aspects of their work. The convenience and efficiency brought about by AI in scientific writing have the potential to revolutionize the way researchers communicate their findings, ultimately accelerating the pace of scientific progress.

In conclusion, the integration of AI, particularly ChatGPT, has greatly facilitated the process of writing scientific texts. From generating abstracts and introductions to providing concise overviews and alternative formulations, the AI assistance has proven to be a valuable resource. While limitations remain, such as challenges with references, the overall positive impact of AI on scientific writing cannot be understated. As researchers continue to explore and harness the potential of AI tools, we can anticipate further advancements in the field of scientific communication.

REFERENCES

[1] Koubâa, Anis, Mário Alves, and Eduardo Tovar. "GTS allocation analysis in IEEE 802.15. 4 for real-time wireless sensor networks." Proceedings 20th IEEE international parallel & distributed processing symposium. IEEE, 2006.

[2] Koubaa, Anis, and Y-Q. Song*. "Evaluation and improvement of response time bounds for real-time applications under non-pre-emptive fixed priority scheduling." International Journal of Production Research 42.14 (2004): 2899-2913.

[3] Alvi, Ahmad Naseem, et al. "Evaluation of slotted CSMA/CA of IEEE 802.15. 4." 2012 seventh international conference on broadband, wireless computing, communication and applications. IEEE, 2012.

[4] Koubaa, Anis, Mário Alves, and Eduardo Tovar. "A comprehensive simulation study of slotted CSMA/CA for IEEE 802.15. 4 wireless sensor networks." 2006 IEEE international workshop on factory communication systems. IEEE, 2006.