Plan the Access? Generic Hardware Independent Energy Consumption and Efficiency Model for Different LoRaWAN Channel Access Approaches

Frank Loh, *Member, IEEE*, Simon Raffeck, Stefan Geißler, and Tobias Hoßfeld, *Senior Member, IEEE*

***Abstract*—The growing popularity of LoRa-based solutions, such as weather, environmental, and climate monitoring, has resulted in larger LoRaWAN deployments and increased network load. However, the random channel access method used in LoRaWAN leads to higher message collision probabilities and potential data loss, diminishing the network's energy efficiency. This article addresses this issue by proposing a hardware-independent model that assesses energy consumption and efficiency in different LoRaWAN channel access approaches. Through simulation and analysis of real energy values, we provide insights and recommendations for optimizing the network performance and energy usage. This study fills a research gap and contributes to the development of more robust and energy-efficient LoRaWAN systems for various application areas.**

***Index Terms*—Channel access, energy consumption, energy efficiency, LoRaWAN, model.**

I. Introduction

THE RECENT opening of Amazon Sidewalk to the public is the next milestone in the success history of LoRaWAN. Amazon Sidewalk promises coverage by a widespread wireless network using Bluetooth, frequency shift keying, and LoRa as access technologies [1]. The Amazon Echo and Ring devices function as gateways and provide cloud connectivity to Internet of Things (IoT) devices [2]. This leads to an easy access to LoRaWAN for more than 90% of the U.S. population [2], [3].

This extended accessibility will even accelerate LoRaWAN adoption, while an annual growth rate of more than 40% is already predicted until 2032 [4]. The popularity is not surprising, as LoRa, the physical layer technology used in LoRaWAN, promises cheap sensors supporting transmissions across long distances with very little energy consumption [5]. For that reason, LoRaWAN is the ideal solution for various IoT use cases from simple weather monitoring to metering in Smart Cities, or environmental monitoring tasks.

Despite the benefits of a simple and cost-effective LoRaWAN, the current random channel access lacks interference protection, resulting in message collisions and data loss. To address this issue, research focuses on improved LoRaWAN channel access, specifically listen before talk and time scheduling [6], [7], [8]. However, additional functionality introduces complexity and impacts the energy consumption of end device. Existing literature quantifies LoRa transmission energy consumption [9], [10] or models transmission cycles with different access approaches [11]. Yet, these studies often focus on the LoRa transceiver alone and lack guidelines for selecting optimal channel access approaches based on transmission scenarios, energy consumption, and hardware independence.

To address this gap in the literature, we propose an energy model for LoRaWAN that quantifies the energy requirements for different tasks during data transmission. By normalizing the energy with the optimal LoRa transmission scenario, we can assess the energy required for a complete transmission across various channel access methods, including random access, listen before talk, and time scheduling. Without relying on specific hardware constraints, our model allows for comparisons of energy efficiency in different scenarios, parameter settings, network conditions, and loads. We validate our model through a lightweight simulation of LoRaWAN channel access, enabling numerical evaluations and informed conclusions about the most energy-efficient transmission option.

To this end, the contribution of this work is three-fold. First, we propose a general energy consumption and energy efficiency model, considering different channel access approaches, network and load situations, and expected collision probability in a LoRaWAN that is applicable for various hardware. Second, we present a way to model listen before talk for LoRaWAN, including the hidden node problem, listening for messages on the channel, and transmission back-offs, to avoid collisions. Lastly, we give suggestions for channel access selection in different network load situations with focus on high-energy efficiency based on our model and a comprehensive simulation study. With these contributions, we answer the following three research questions.

1) *RQ1:* Is it possible to model a LoRaWAN with different channel access approaches and the resulting energy consumption, usable for various underlying hardware?
2) *RQ2:* Which parameters and metrics influence the energy consumption and the energy efficiency of LoRaWAN channel access approaches?

Manuscript received 26 July 2023; revised 25 January 2024; accepted 14 February 2024. Date of publication 23 February 2024; date of current version 23 May 2024. This work was supported in part by the Open Access Publication Fund of the University of Wuerzburg and in part by the Bavarian Ministry of Economic Affairs, Regional Development and Energy under Grant DIK0507/02 "Greenfield." *(Corresponding author: Frank Loh.)*

The authors are with the Institute of Computer Science, Chair of Communication Networks, University of Würzburg, 97070 Würzburg, Germany (e-mail: frank.loh@uni-wuerzburg.de).

Digital Object Identifier 10.1109/JIOT.2024.3369325

2327-4662 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

3) *RQ3:* What is the best channel access approach for LoRaWAN from an energy efficiency point of view, dependent on the load in the network?

The remainder is structured as follows. Section II presents background information and summarizes related work. Section III introduces our energy model and introduces details for each channel access approach. Afterwards, we present our simulation to validate the model and define validation and evaluation scenarios in Section IV and evaluate the scenarios in Section V. Finally, Section VI concludes.

II. Background and Related Work

This section covers an introduction to data transmission with LoRa in a LoRaWAN, different channel access approaches, and several metrics to describe the quality of a LoRaWAN. A special focus on energy consumption and energy efficiency. At the end, related work is discussed.

A. LoRa and LoRaWAN

LoRa is used as physical layer modulation technique in LoRaWAN. Its key parameters, required to understand the remainder of this work, are the spreading factor, the bandwidth, the LoRa message size, and the time on air to transmit messages. With the spreading factor (SF) that determines the number of raw bits a symbol carries, and the bandwidth (BW), the symbol rate R can be expressed as $R = [\mathrm{BW}/2^{\mathrm{SF}}]$. A spreading factor in LoRa is in the range of SF 7 to SF 12, and the typically used bandwidth for LoRa in Europe is 125 kHz.

Messages in LoRa contain a preamble and 4.25 symbols for synchronization, followed by an optional header, and the actual payload [7]. The typical preamble length is eight symbols and 20 bits for the header. Furthermore, a LoRa message contains a cyclic redundancy check, an optional data rate optimization, and a coding rate for the forward error correction code. The coding rate can be between (4/5) and (4/8). In this work, we use an eight symbol preamble, an enabled header, enabled cyclic redundancy check, and no low-data rate optimization. The used coding rate is (4/8), where each symbol is coded by two symbols. Details about the payload follow in the methodology section. Further information about LoRa's physical layer and messages is available, e.g., in [12] and [13]. Last, to understand LoRa transmissions and the time a LoRa transmission blocks a frequency channel, the time on air needs to be investigated. It is calculated by multiplying the message length with the time to transmit a single symbol T_s with $T_s = 1/R$. Thus, the main factors influencing the time on air are the payload length and the spreading factor.

B. LoRaWAN Channel Access

LoRaWAN uses the EU 863 – 870 frequency range for message transmission [14]. There, eight frequencies are usable for LoRa uplink messages with an additional one for frequency shift keying. One channel is reserved for downlink data. In this work, we assume full independence of the channels and thus, no message collision of data transmitted in different channels. For that reason, we only analyze a single one and assume extrapolation to the other channels is valid. Similarly, it is done in literature studying message transmission, channel access, or LoRaWAN data transmission quality in general [7], [13], [15].

Today's LoRaWAN uses an ALOHA like version of random access [12] for channel access. The theoretical maximal utilization is 18.4% [16], however, the actual performance is better because of the very robust physical layer in LoRaWAN [17]. Nevertheless, alternatives are studied recently. With listen before talk, the frequency of listening on the channel in advance to a LoRa transmission and data is only transmitted if it is free. A different alternative is slotted ALOHA, where a time division multiple access approach is added to the pure ALOHA channel access. There, messages can only be transmitted in predefined time slots and sensors have to wait for the next available slot with their transmission. However, this comes with additional synchronization overhead and has been shown to work only if the time on air of all LoRa messages in the network is similar [18]. The last introduced approach is a complete time scheduling of LoRa messages. This increases the overhead in comparison to slotted ALOHA since each sensor has a predefined slot for its transmission and can not simply select the next slot. However, in theory, this approach can avoid message collision completely and increase the error-free throughput significantly [7].

C. LoRaWAN Quality Metrics

If we assume coverage of all sensors in a deployment, a good LoRaWAN has a small collision probability to avoid message loss, a low-energy consumption, and a high-energy efficiency for message transmission. For that reason, these quality influencing metrics are discussed in the following.

Collision Probability: The collision probability in this work is defined as the percentage of message transmissions that overlap in time with other messages. According to literature, messages transmitted with different spreading factors [17] or with the same spreading factor [19] can be decoded and are not always lost. However, message recovery can not be guaranteed in both cases. That is, why we study the worst case and assume that all overlapping messages, independent on the used channel access approach, collide and are lost.

Energy Consumption: Different channel access approaches can have an impact on the collision probability in LoRaWAN [7], but also influence the energy consumption. It is shown in [11] that almost all the energy in the process of a LoRa transmission with random access is consumed in the following states: transceiver wake up, actual message transmission, open an optional receive window, optional reception of data, and optional data processing. Alternative channel access approaches like listen before talk or slotted ALOHA have waiting and listening states, where additional energy is consumed. Thus, there is a tradeoff between reducing the collision probability and an increasing energy consumption due to synchronization, listening on the channel, data reception, and data processing. For that reason, a pure concentration on the total consumed energy per device can draw wrong conclusions with regard to LoRaWAN quality.

Energy Efficiency: A potentially better metric to benchmark different approaches is the energy efficiency. It compares the required energy to transmit data with the best case transmission [20], without collisions, message overhead, or any overhead like message resynchronization from complex channel access approaches.

D. Modeling Energy Consumption

The energy model presented in this work extends our generic channel access model [11]. There, we identify different relevant states st_i a sensor can assume during a LoRa transmission: wake up transceiver st_1, the actual data transmission st_2, wait for transmission start st_3, listening on the channel st_4, open receive window st_5, data reception st_6, and data processing st_7. These states during message transmission are required for all available device classes in LoRaWAN. The only difference between the classes is the way, devices open receive windows and consequently, the time devices can remain in sleep mode. While class A devices only open receive windows after data is transmitted, class B devices regularly open windows to receive potential information from the gateway. In contrast, class C devices always keep a receive window open to be online for potentially arriving gateway messages at the cost of higher energy consumption during normal operation. However, as this behavior does not affect the message transmission itself and class B device can also open receive windows directly after a transmitted message, it does not influence the energy investigation and modeling conducted in this work. Consequently, we assume the usage of class A devices, having the most limitations with regard to availability for gateway messages and the least energy consumption. The energy consumption E_{total} of a LoRa message transmission cycle from transceiver wake up to the return back into sleep mode is obtained by summing up the required energy E_i in each of these states. It is dependent on the channel access approach and can be computed by multiplying the required power P_i in each respective state i with the time T_i the sensors remain in the state. However, we observe that not all of these states are relevant to quantify differences in energy consumption. The wake-up transceiver state st_1 is the same for all transmissions, independent on the channel access approach. Therefore, it only needs to be considered if the complete transmission cycle is evaluated to determine, for example, the battery lifetime. It can, however, be omitted when the channel access approaches are compared. Furthermore, the model can be improved if the different states are simplified. This means, the process of opening a receive window can be abstracted to a wait and a listen operation. Likewise, a listen operation is equal to a reception operation if the antenna is powered in the same way and acknowledgments or other data can be received there. An additional reception process can be added again if more data must be received. However, large update transmissions are not expected with LoRa, since sensors have often only limited processing capabilities and the gateways also need to comply with the duty cycle. Details about the resulting energy consumption equations are available in [11].

E. Related Work

Improving a LoRaWAN by in-depth collision probability studies, energy consumption analysis, and alternatives to random channel access are recently discussed by researchers. Especially channel access studies date back to 2016, when Bankov et al. [12] investigated it first. Since then, different researchers analyzed the performance of random access [21] in a LoRaWAN or studied alternatives beyond ALOHA [6]. Especially slotted ALOHA is considered as an alternative medium access control approach [22], [23], [24]. However, it is proven to only work efficiently if the time on air for all messages in the network is similar [18]. An alternative is listen before talk, coming with additional overhead for channel sensing and a benefit in reduced collision probability [7], [8]. In contrast, a time scheduled approach for all messages promises the avoidance of all collision [7]. However, synchronization can be a challenge, as LoRa devices often use cheap hardware only. Nevertheless, all these approaches have one thing in common: there is either no energy consumption and energy efficiency investigation available in literature or the studies rely on very specific hardware that makes comparison rather complex.

From an energy consumption point of view, literature already studied the energy performance for LoRaWAN in general [9] or for LoRaWAN for Industry 4.0 applications in particular [25]. Furthermore, energy consumption values for different sensors transmitting with LoRa are measured [9], [10], and energy models exist for LoRaWAN [10], [11]. While a hardware independent energy model exists for IoT in general [26], to the best of our knowledge there is none available yet for LoRaWAN. Maudet et al. [27] proposed a redefined energy model for LoRaWAN that is validated using empirical measurements, taking the number of nodes and the resulting collision probability into account. Singh et al. [28] presented an empirical measurement study on the energy consumption of LoRaWAN among others. Using these results, they analyze the battery lifetime in general and for real life use case. Lastly, in [29], an ns-3 simulation for LoRaWAN is implemented using real life measurement results to investigate the power consumption of the different states in a LoRa transmission. To this end, we aim to eliminate the gap of a missing hardware independent energy model for LoRaWAN from the literature. Therefore, we normalize the energy consumption of all processes during a LoRaWAN transmission and identify key metrics that influence the energy consumption. We use this as input for an energy model and analyze different channel access approaches in detail. Consequently, we provide a generic model that is able to describe the energy efficiency of a wide range of LoRaWAN deployments.

III. Energy Model Description

This section introduces a hardware independent energy model for LoRaWAN channel access approaches. First, the general energy consumption for a LoRa message transmission is modeled. Afterwards, different channel access specific characteristics are explained and modeled in detail. The energy

TABLE I
SUMMARY OF IMPORTANT MODEL NOTATIONS (RANDOM VARIABLES IN CAPITAL LETTERS AND CONSTANTS IN LOWERCASE LETTERS)

variable	explanation	variable	explanation	variable	explanation
E_i	energy consumption in process i	E_0	channel access independent energy between two transmission starts	E_i^*	energy in process i in time $t_{\min}$
T_i	time in process i	$t_{\min}$	minimal transmission time on air for normalization	T_i^*	normalized time in process i
p_i	required constant average power in process i	T_{toa}	time on air for transmission	c_i	constant scaling parameter
E_{total}	energy consumption of complete transmission cycle	E_{lora}	energy consumption of all transmission cycles in the complete LoRaWAN	E_{eff}	energy efficiency of the LoRaWAN
			channel access specifics		
N_j	req. number of receive windows if access approach j is used	$T_{i,j}$	time in process i if access approach j is used	$t_{i,j,o}$	constant time in process i with access approach j if receive window is opened
$T_{\text{drift,total}}$	clock drift since last re-synchronization	t_{slot}	slot length in seconds	T_{rcf}	duration between drift out of slot and actual successful re-synchronization
T_{drift}	time drift in seconds between two transmissions	$p_{\text{sync,sc,perfect}}$	probability to receive sync message in perfect time slotted	$p_{\text{sync,sc}}$	probability to receive sync in collision affected time slotted
$p_{\text{coll,sync}}$	collision probability of sync message	X	number of listening windows	$p_{\text{coll,lbt}}$	probability detecting an lbt channel as occupied when listened on the channel

models from literature lack in two main situations: 1) a more complex behavior during channel access, i.e., multiple listening on the channel operations, wait times, or optional states that do not occur for all transmissions are not properly covered and 2) a comparison of different channel access approaches without specific hardware specifications and thus, the energy requirement between them is not possible. To overcome these limitations, we express the transmission behavior for different channel access approaches in more detail in the following. This includes the behavior during message collision, listening on the channel, waiting for transmission, and time synchronization for different channel access approaches. According to literature [7], listen before talk and time scheduled are the most promising alternative approaches to current random channel access. For that reason, these approaches are studied in more detail in this work. Since slotted ALOHA is only suggested in very specific scenarios, in which the message time on air variance is small [7], [18], it is omitted in the following consideration.

A. LoRa Transmission Processes

Different channel access approaches can be compared by analyzing the sensor's behavior during *transmit*, *wait*, *receive*, and *process* [11]. The probability to start one of these procedures or processes, the duration of each process, and the energy consumption for a sensor in each process is of major interest. However, the energy requirement to process any received data is not directly related to the transmission behavior itself and is dependent on the used hardware. To cover this, a random variable E_0 is added to model every additional energy consumption with a constant power p_0 and time T_0, being the time between two data transmission starts of the sensor. It covers, among others, the sensor's general energy requirement for data measurement, processing of any data, and the energy consumption during sleep mode. In this work, no parameter study of the named procedures is done as the energy consumption of them is independent on the channel access approach. Please note that we omit the analysis of minor changes in the duration of the sleep mode as a result of different channel access approaches, as the energy consumption for idle or sleep modes is several orders of magnitude smaller compared to all active processes [10], [27], [28]. The general equation for the total energy consumption E_{total} based on the energy consumption for transmit (E_1), wait (E_2), and receive (E_3) and independent on the access approach is given by

$$E_{\text{total}} = E_1 + E_2 + E_3 + E_0. \tag{1}$$

B. General Notation Details

In the following, this very general equation is further analyzed. All important variables for the following energy model with an explanation are summarized in Table I. The convention with the indices is as follows. Every variable with a single index (e.g., E_i) describes the behavior in a specific process with $i = 1$ for transmit, $i = 2$ for wait, $i = 3$ for receive, and $i = 0$ for channel access independent procedures. The second index is for channel access specific variables (e.g., $E_{i,j}$) with $j =$ ra for random access, $j =$ lbt for listen before talk, and $j =$ sc for the time scheduled approach.

C. LoRaWAN Energy Model

The total energy consumption E_{total} described in (1) is still dependent on the used channel access approach and the hardware. Though, we can describe the required energy for a LoRa transmission by the power p_i over time T_i the end device is in the different processes i. Note, since we do not consider any power peaks, e.g., at the beginning of a transmission, we use a constant average power consumption p_i in process i here.

Then, we normalize the time with a standard time $t_{\min}$ during the well known transmission process. Therefore, we use the minimal time on air $t_{\text{toa,SF=7,sy=1}} = t_{\min}$, a sensor requires to transmit data in LoRaWAN for time normalization. This is the time on air to transmit one symbol *sy* with spreading factor SF 7 using random access without wait, receive, or any collisions. We obtain

$$E_{\text{total}} = \sum_{i=1}^{3} p_i \cdot T_i + E_0 = \sum_{i=1}^{3} p_i \cdot t_{\min} \cdot \frac{T_i}{t_{\min}} + E_0 \quad (2)$$

$$= \sum_{i=1}^{3} E_i^* \cdot T_i^* + E_0 \quad (3)$$

with E_i^* as required energy for the minimal duration $t_{\min}$ in process i and T_i^* as normalized time in process i. With this normalization with the minimal transmission time on air $t_{\min}$, we can directly express the difference between the investigated transmission and the best case with $T_1 = t_{\min}$ and no overhead from other processes.

Dependent on the underlying hardware, the required energy in the different processes can vary, while the normalized time T_i^* is usually dependent on the channel access approach and the amount of transmitted data. For example, if a transmission takes twice as long as the above-mentioned optimal one, e.g., due to double the number of transmitted symbols, we achieve $T_1^* = 2$ for the transmission process.

Next, we express differences in the energy consumption between transmit, wait, and receive for the same duration with constant scaling parameters c_2 for wait and c_3 for receive. These scaling parameters are the energy cost differences for wait and receive, respectively compared to transmit. They can reflect the influence of different sensors or the difference in the energy consumption between processes and need to be adjusted based on the underlying hardware. In this way, we can describe the energy consumption of all processes in relation to the consumption E_1^* for data transmission with

$$E_{\text{total}} = E_1^* \left(T_1^* + c_2 \cdot T_2^* + c_3 \cdot T_3^*\right) + E_0. \quad (4)$$

If a message transmission of one second requires ten times more energy than waiting for one second, c_2 is equal to 0.1.

Based on these findings, we can answer our first research question RQ1 as follows: It is possible to model the energy consumption of a LoRa transmission in general way for arbitrary hardware if the energy requirement is normalized by the minimal possible time on air. The energy requirement of processes like listening on the channel or additional wait times can be modeled with energy consumption ratios between these processes and the energy consumption during data transmission.

D. Process Duration Modeling

While the total energy consumption of a LoRa transmission cycle is described in (4), the different behavior during each process dependent on the channel access approach is not covered yet. Since the main influencing factor of the different processes on the overall energy consumption is its individual duration.

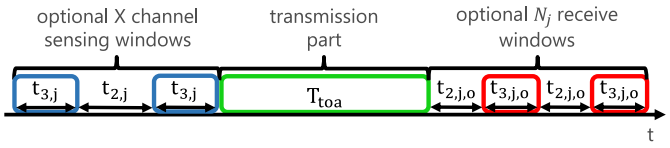

Fig. 1. Visualization of transmission cycle.

As introduced above, the transmit process is always required during a LoRa message transmission cycle. Its duration is the transmission time on air T_{toa} of the transmitted message and is independent on the channel access approach. Thus, we can describe the time on air for all approaches as

$$T_{\text{toa}} = T_{1,\text{ra}} = T_{1,\text{lbt}} = T_{1,\text{sc}} = T_1. \quad (5)$$

However, the total duration for wait and receive during a complete transmission cycle is dependent on the channel access approach. An example transmission cycle is visualized in Fig. 1. Ahead of the actual data transmission, shown in green, it is possible to open optional listening windows in blue, as it is done with listen before talk. After a transmission, optional receive windows are possible, shown in red. In the following, such listening and receive windows are investigated for each approach individually. In particular each duration is studied, relevant for the energy consumption calculation.

Random Access: Random access has the least overhead if data is transmitted without any possibility to receive messages and the sensor immediately returns to sleep mode after a transmission. Besides this simple procedure, a transmission with random access can contain an optional data reception possibility, where one or more receive windows are opened. This adds one or more wait and reception process to the actual data transmission, since according to LoRaWAN standardization, ahead of each reception window, the sensor waits for a predefined duration [30]. This is symbolized by the optional receive windows after the transmission part in Fig. 1.

The wait time $T_{2,\text{ra}}$ during a transmission cycle can then be calculated with

$$T_{2,\text{ra}} = N_{\text{ra}} \cdot t_{2,\text{ra},o} \quad (6)$$

with $N_{\text{ra}} \in \mathbb{N}_0$ as random variable for the number of receive windows and $t_{2,\text{ra},o}$ as predefined constant single wait time. Considering all transmissions in the network where random access is used, the expected total wait time $E[T_{2,\text{ra}}]$ is

$$E\left[T_{2,\text{ra}}\right] = E[N_{\text{ra}}] \cdot t_{2,\text{ra},o}. \quad (7)$$

The random variable to describe the reception time $T_{3,\text{ra}}$ during a transmission process with random access can be defined by

$$T_{3,\text{ra}} = N_{\text{ra}} \cdot t_{3,\text{ra},o} \quad (8)$$

with $t_{3,\text{ra},o}$ as duration for the opened receive window. Again, considering all transmissions in the network, the expected reception time $E[T_{3,\text{ra}}]$ during a transmission procedure for random access can be defined by

$$E\left[T_{3,\text{ra}}\right] = E[N_{\text{ra}}] \cdot t_{3,\text{ra},o}. \quad (9)$$

Time Scheduled: Next, the wait and reception time is analyzed if the time scheduled (sc) approach is used. The wait time $T_{2,\mathrm{sc}}$ and the reception time $T_{3,\mathrm{sc}}$ calculation if a reception window is opened is similar to random access. However, the constant wait duration $t_{3,\mathrm{sc},o}$, and the number of receive windows N_{sc} can be different.

Data that need to be received if the time scheduled approach is used can include, similar to random access, message acknowledgments and, different to random access, resynchronization messages. While the decision, whether receive windows are opened to receive acknowledgments is dependent on message acknowledgment settings, similar to random access, it is different for the probability $p_{\mathrm{sync,sc}}$ to resynchronize the sensor's clock. This resynchronization probability is dependent on the clock drift of the end device, the slot length, and the collision probability for the resynchronization message. In the following, we assume no additional required receive windows for message acknowledgments, to not require additional resources at the end device. In a working time scheduled approach, no messages collide due to distinct message slots [7]. In any other case, for example if messages collide with random cross-traffic, the sender is either not informed about the data loss or the required energy for additional receive windows needs to be added to the energy consumption calculation afterwards. Considering the resynchronization messages only, the probability $p_{\mathrm{sync,sc,perfect}}$ to open a receive window and receive a resynchronization message in perfect conditions without any collision is

$$p_{\mathrm{sync,sc,perfect}} = P\big(T_{\mathrm{toa}} + T_{\mathrm{drift,total}} > t_{\mathrm{slot}}\big) \tag{10}$$

with t_{slot} as total slot length and $T_{\mathrm{drift,total}}$ as total time drift of the sensor's clock since last resynchronization. However, real LoRaWANs suffer from message collisions or losses. For that reason, we extend the equation to

$$p_{\mathrm{sync,sc}} = P\big(T_{\mathrm{toa}} + T_{\mathrm{drift,total}} > t_{\mathrm{slot}} + T_{\mathrm{rcv}}\big) \tag{11}$$

with T_{rcv} as total duration between the message transmission exceeded the slot because of the clock drift and the actual resynchronization message arrived successfully. Considering the expected value of the random variable T_{drift} for the clock drift, a fixed slot length t_{slot}, and a collision probability of the resynchronization messages $p_{\mathrm{coll,sync}}$, we can achieve a single equation for the probability $p_{\mathrm{sync,sc}}$ to receive a resynchronization message in the time scheduled approach after a message is sent in a collision affected channel with

$$p_{\mathrm{sync,sc}} = \frac{E[T_{\mathrm{drift}}]}{(t_{\mathrm{slot}} - E[T_{\mathrm{toa}}]) + \frac{E[T_{\mathrm{drift}}]}{1-p_{\mathrm{coll,sync}}} - E[T_{\mathrm{drift}}]}. \tag{12}$$

Note that we assume a loss of all colliding messages. Since we assume transmissions of individual sensors in fixed slots and thus, after constant time intervals similar to [7], we can explain the calculation with Fig. 2. The figure shows seven example slots over time for a message with an expected length $E[T_{\mathrm{toa}}]$, symbolized by the green boxes. Note that for simplicity reasons, we did not visualize any guard times in this example. In a real setup, they are required between slots to avoid collisions with messages in the following slots. However, adding guard times does not influence the calculation of $p_{\mathrm{sync,sc}}$. From slot to slot, the expected time drift of the messages is $E[T_{\mathrm{drift}}]$.The message has been resynchronized to start its transmission at the beginning of its first slot. That is, why the sensor has $t_{\mathrm{slot}} - E[T_{\mathrm{toa}}]$ 'space' to drift. Dividing this by the expected drift time $E[T_{\mathrm{drift}}]$, the synchronization probability is achieved in a perfect network without any collision. But since we also consider message collisions in our LoRaWAN, this synchronization probability must be adjusted, as explained in the following.

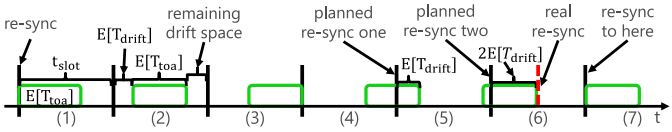

Fig. 2. Example visualization of slots and clock drift.

A sensor is resynchronized when its message transmission drifts out of the assigned slot, as shown between slot four and five. However, the resynchronization message is assumed to be lost in this example. In contrast to other operations, where a lost message directly influences the device behavior and increases or reduces the energy consumption, in LoRaWAN both the sensor and the gateway do not know the loss until the next message is sent from sensor to gateway. Then, the next message exceeds its slot limits again, as shown between slot five and six, and the gateway sends a new resynchronization. Until this point, no additional energy is consumed by the sensor and the probability to receive a resynchronization message $p_{\mathrm{sync,sc}}$ is not adjusted. This is only required, when the sensor actually receives the resynchronization message.

Since a message loss is equal to a less frequent reception of resynchronization messages at the end device, the calculation of the resynchronization probability $p_{\mathrm{sync,sc}}$ can be adjusted by extending the slot length by one expected time drift $E[T_{\mathrm{drift}}]$ for each unsuccessfully transmitted resynchronization. Thus, we add $[(E[T_{\mathrm{drift}}])/(1 - p_{\mathrm{coll,sync}})]$ for each required resynchronization message to the denominator and subtract $E[T_{\mathrm{drift}}]$ for the successfully received resynchronization. Contrary to common sense, many colliding messages reduce the sensor's energy consumption since fewer messages are received and the resynchronization rate is reduced. However, a too high-collision rate of resynchronization messages may lead to sensors drifting out of their slots and colliding with messages transmitted from other sensors in adjacent slots. This can lead to a complete break in the time scheduled channel access approach as described in [7]. Thus, it is essential to set up the slot length in a suitable way, reducing both energy consumption and avoiding collisions.

Finally, we achieve a wait time $T_{2,\mathrm{sc}}$ during a transmission cycle if receive windows are only opened when a resynchronization message arrives correctly of

$$T_{2,\mathrm{sc}} = p_{\mathrm{sync,sc}} \cdot N_{\mathrm{sc}} \cdot t_{2,\mathrm{sc},o} \tag{13}$$

and an expected total wait time $E[T_{2,\mathrm{sc}}]$ of

$$E\big[T_{2,\mathrm{sc}}\big] = p_{\mathrm{sync,sc}} \cdot E[N_{\mathrm{sc}}] \cdot t_{2,\mathrm{sc},o}. \tag{14}$$

The random variable $T_{3,\text{sc}}$, for the reception time when the time scheduled approach is used is

$$T_{3,\text{sc}} = p_{\text{sync,sc}} \cdot N_{\text{sc}} \cdot t_{3,\text{sc},o} \tag{15}$$

and the expected reception time $E[T_{3,\text{sc}}]$ considering all transmissions in the network is

$$E\big[T_{3,\text{sc}}\big] = p_{\text{sync,sc}} \cdot E[N_{\text{sc}}] \cdot t_{3,\text{sc},o}. \tag{16}$$

Note that in reality it is very unlikely that the sensor opens receive windows only when a resynchronization message arrives correctly. Thus, this consideration is a best case investigation and the actual probability to open receive windows is most likely higher.

Listen Before Talk: For listen before talk, the time to transmit data and the handling of receive windows is similar to random access. However, the listening on the channel process, shown ahead of the transmission in Fig. 1 must be taken into consideration to avoid collisions.

Channel Listening Procedure: The listening on the channel process determines whether a channel is free for a device to send its data. Besides avoiding potential collisions, it adds additional receive and wait times and increases the energy consumption of the device. Listening on a channel is equal to a data reception process and occurs before each transmission. In addition, the wait process occurs if the channel is detected as occupied during receive. Wait and receive before each transmission can be described by the geometric distribution, while wait is not necessarily always needed, in particular not, if the first receive process detects no other message. Let X be a random variable describing the number of listen windows until encountering an empty one. In this case, X follows the geometric distribution that describes the number of trials k until first success ($k = 1, 2, \ldots$) with $p_{\text{coll,lbt}}$ being the mean probability to detect a channel as occupied when listened on a channel. We can hence compute the mean number of trials before the channel is free as the first moment of the geometric distribution with

$$E[X] = \frac{1}{1 - p_{\text{coll,lbt}}}. \tag{17}$$

Since the wait process is not required when the channel is free, the random variable $T_{2,\text{lbt}}$ to describe the wait time in a complete transmission cycle can be achieved with

$$T_{2,\text{lbt}} = E\big[t_{2,\text{lbt}}\big] \cdot (E[X] - 1) + N_{\text{lbt}} \cdot t_{2,\text{lbt},o}. \tag{18}$$

The predefined expected wait time $E[t_{2,\text{lbt}}]$ when another message occupies the channel is multiplied with $E[X] - 1$ to achieve the expected wait time ahead of the actual transmission. A single wait time $t_{2,\text{lbt}}$ can be defined as constant value or as random variable. For example, a general simulation study in [7] shows good results with a random value between 0.4 and 1.75 s. Furthermore, similar to random access, the expected wait time after a transmission is added, achieved by $N_{\text{lbt}} \cdot t_{2,\text{lbt},o}$. Again, this leads to an expected total wait time $E[T_{2,\text{lbt}}]$ considering all listen before talk transmissions in the LoRaWAN of

$$E\big[T_{2,\text{lbt}}\big] = E\big[t_{2,\text{lbt}}\big] \cdot (E[X] - 1) \; + E[N_{\text{lbt}}] \cdot t_{2,\text{lbt},o}. \tag{19}$$

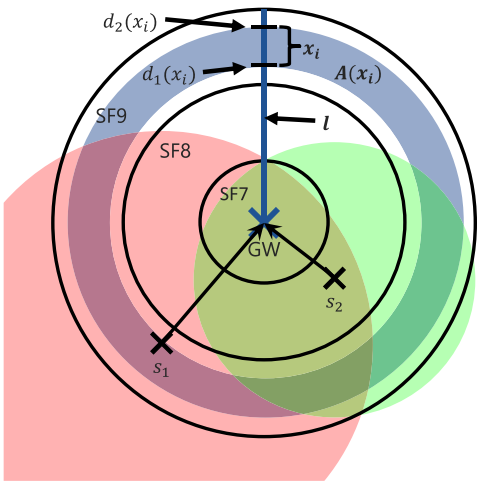

Fig. 3. Schematic representation of the proposed collision avoidance model for listen before talk.

The receive time $T_{3,\text{lbt}}$ is achieved by

$$T_{3,\text{lbt}} = t_{3,\text{lbt}} \cdot E[X] + N_{\text{lbt}} \cdot t_{3,\text{lbt},o} \tag{20}$$

where the channel listening time $t_{3,\text{lbt}}$ to avoid collisions is set to a predefined constant value and is multiplied with the expected value $E[X]$ for the number of channel listening operations. Again, with $N_{\text{lbt}} \cdot t_{3,\text{lbt},o}$, the receive time after the actual message transmission is added similar to random access. Thus

$$E\big[T_{3,\text{lbt}}\big] = t_{3,\text{lbt}} \cdot (E[X]) + E[N_{\text{lbt}}] \cdot t_{3,\text{lbt},o} \tag{21}$$

is the expected total receive time considering all listen before talk transmissions in a LoRaWAN. While all wait and receive times can be defined in the network setup phase, the probability $p_{\text{coll,lbt}}$ to detect another message at the channel when a sensor listens to it during listen before talk is dependent on the load in the network and the network setup. However, to fully describe the behavior of listen before talk, an in-depth understanding of this probability is required.

Collision Avoidance Probability: In a well planned LoRaWAN, each sensor transmits to its closest gateway, interferes with a minimal number of other sensors for a minimal duration to avoid collisions [15], [31], but also avoids the hidden node problem. The hidden node problem in LoRaWAN can occur, if obstacles are in the line of sight transmission of sensors, but also if sensors are placed in opposite directions from a gateway with possible transmission distances to the gateway but not to the other sensor. Since the location of obstacles is highly dependent on, among others, the network setup, urbanization, and topology, it is omitted here and the focus is on sensor location only. The influence of the sensor location on the collision probability by the hidden node problem is visualized in Fig. 3. The figure displays a center gateway (GW) and two sensors s_1 and s_2. Both sensors transmit to the central gateway GW indicated by the black arrows. Dependent on the distance to the gateway, a sensor receives a spreading factor. Larger spreading factors allow transmissions across longer distances at the cost of longer transmission airtime [31], [32]. Thus, unnecessarily

large spreading factors should be avoided to minimize the time on air of the transmissions [15]. In the example figure, areas where sensors are able to transmit to the gateway are marked with different spreading factors. These areas are placed circularly around the gateway starting with spreading factor 7 in the middle and using spreading factor 9 in the outside circle. Higher spreading factors are not added for better visibility.

In this example, sensor s_1 uses spreading factor 9 for its transmissions in the spreading factor 9 area to make sure its data can be transmitted to the gateway. In contrast, sensor s_2 uses spreading factor 8. The usage of larger spreading factors and thus, longer transmission distances lead to larger interference areas for both sensors. Sensor s_1 interferes all sensors in the red shaded area in this example. In this area, the gateway is included showing that a transmission to the gateway is possible. In addition, sensor s_2 is also in this area and if sensor s_1 transmits data, sensor s_2 can also hear the transmission. For that reason, it is not influenced by the hidden node problem when sensor s_1 is transmitting. In such a case, sensor s_2 could avoid a collision by delaying its own transmission if listen before talk is used.

However, the situation is different if sensor s_2 starts a transmission. Its green interference area does not include sensor s_1 and if sensor s_2 transmits data while sensor s_1 plans to start a transmission, it can not hear sensor s_2's transmission. Thus, sensor s_1 would also transmit and the messages collide.

To model this behavior for all sensors s_i in a network with a set of sensors S, we the probability that a sensor s_i can hear any other sensor s_j with $s_i, s_j \in S$. Then, we can determine the probability for a wait procedure when listen before talk is used. Therefore, we map the area of a complete LoRaWAN to a single line l. In Fig. 3, this line l connects the gateway with the outer circle of spreading factor 9, while for the calculation it also includes the other spreading factors up to spreading factor 12 in more outside regions. In the next step, the line is discretized in n sections, highlighted by the example section x_i for section i. It is delimited by the distance $d_1(x_i)$ and $d_2(x_i)$ to the gateway GW. By calculating the area $A(x_i)$ with $i \in 1, 2, \ldots, n$ of the complete circle, shown in blue, with a minimal distance to the gateway of $d_1(x_i)$ and a maximal distance to the gateway of $d_2(x_i)$, the circular representation of the network can be mapped to a one line representation with $l(i) = A(x_i)$. For the area calculation, the following holds true:

$$\sum_{i=0}^{n} A(x_i) = \sum_{i=7}^{12} A(\mathrm{SF}_i) = A(\mathrm{GW}) \tag{22}$$

with $A(\mathrm{SF}_i)$ as area where sensors transmit with spreading factor i and $A(\mathrm{GW})$ as complete area the gateway GW "covers." After this step, each area $A(x_i)$, or $l(i)$ contains the following information: area size, minimal, maximal, and mean distance to the gateway, number of sensors in the area it represents, and the spreading factor all these sensors transmit with to be able to reach the gateway. In this work, we assume an uniform spacial distribution of sensors in the complete network to achieve the average hidden node probability per sensor transmitting with spreading factor i. However, the calculation works similarly for each area $A(x_i)$ if an uniform spacial distribution of sensors in $A(x_i)$ but differences between $A(x_i)$ and $A(x_j)$ are visible. Otherwise, the calculation is done for each sensor individually.

In a last step, the interference percentage is calculated for each $l(i)$ with each other $l(j)$ with $j \in 1, 2, \ldots, n$. Thus, as a result, we receive for each area $l(i)$ the number of sensors located in the area and the number of sensors that can be heard in area $l(i)$ from any other area $l(j)$ in the network. If we map this to sensors and spreading factors, we can, for example, calculate the number of sensors with their respective spreading factors that are heard by a sensor s transmitting with its spreading factor sf located in $l(i)$ or in general in a specific spreading factor area. If this is done for all sensors in the network, or averaged for all spreading factors in the network, we can determine the hidden node probability for all sensors in the network.

Listen Before Talk Back-Off Model: The wait probability for listen before talk is equal to the probability a sensor s_i detects another transmission from any other sensor s_j when it wants to start its own transmission or during a predefined sensing interval before the transmission attempt. To determine the wait probability and duration, the probability another message occupies the channel and is detected must be calculated. This wait duration, a so called back-off, is usually a delay for the next transmission attempt for a specific duration. Thus, besides the arrival rate of new transmission attempts, also these retrials influence the general load in the network and must be taken into consideration.

Assuming exponentially distributed interarrival times, we can model our network as a simple $M/GI/1$ queuing model if the number of transmissions, and thus, arrivals is large enough and sensor transmission start times are random [33], [34]. The transmission channel is a single processing unit with general independent processing time distribution equal to the time on air. Messages arriving when the channel is occupied are immediately rejected (perfect listen before talk without retrial queue). If we extend the system to listen before talk with retrials after a channel was occupied, an additional finite-source retrial queue can be modeled, presented for example in [35] and surveyed in [36]. It extends the system to a 2-D queuing model with the number of messages in the back-off phase in the second dimension. A numerical solution for the system for an exponential processing and back-off time is provided by [37]. However, we show in [7] that an exponential back-off time is not reasonable in LoRaWAN. Especially very short and long back-offs lead to unnecessary more back-offs or require long waiting times. Both would increase the energy consumption unnecessarily. Thus, the final listen before talk model for LoRa can be described as $GI/GI/1$ model with a finite-source retrial queue. A similar model is evaluated by Wüchner et al. [38]. However, the authors rely on the MOSEL-2 evaluation tool [39] to obtain numerical results. For that reason, we also use a simulation in this work to validate our modeling approach, discuss influencing factors for variances, and obtain a performance evaluation for the presented channel access approaches in LoRaWAN.

E. Energy Consumption and Energy Efficiency

Last, the average energy consumption E_{lora} of all transmission cycles in the LoRaWAN can be calculated with

$$E_{\text{lora}} = \frac{\sum_{i=1}^{|S|} \left(E_{\text{total},s_i}\right)}{|S|} \tag{23}$$

for all sensors $s \in S$ with S as the set of all sensors, dependent on the used channel access approach. However, this does not consider potentially lost messages due to message collisions. As we outline above, the collision and back-off probability for listen before talk can be obtained via simulation. Furthermore, perfect time scheduled can avoid collisions completely if slots are chosen large enough, sensors are resynchronized correctly and timely, and the amount of cross-traffic is low [7]. If the currently state-of-the-art channel access approach random access is used, the arrival process of new messages can be estimated as Markov arrival process according to Metzger et al. [33], if the number of devices in the network is large. We assume a sufficiently large number of messages since for a few sensors, this analysis is pointless due to the low number of collisions. Thus, the expected collision probability $p_{\text{coll},s}$ for a sensor s can be calculated in the random channel access case for a given time frame T with

$$p_{\text{coll},s} = 1 - \left(\prod_{a \in S} \left(1 - P_{s,a}\right)\right) \tag{24}$$

with

$$P_{s,a} = \frac{T_{\text{toa},s} + T_{\text{toa},a}}{T} \tag{25}$$

with $P_{s,a}$ as probability for a collision of a message from sensor s with transmission time on air $T_{\text{toa},s}$, and sensor a with transmission time on air $T_{\text{toa},a}$ in the time interval T. By the product of the complementary probabilities $(1 - P_{s,a})$ of the collision probabilities between s and all $a \in S \backslash s$, we achieve the probability for no collision of the message transmitted from s in the time interval T. As a consequence, $p_{\text{coll},s}$ is the collision probability of a message transmitted by s in the time interval T. Thus, the collision probability is dependent on the load in the network and the sensor's sending rate. Depending on the collision probability and additional overhead from wait and receive, the energy efficiency E_{eff} can then be determined for the complete LoRaWAN. It is achieved by normalizing the expected required energy to transmit data $E[E_1]$ with the average energy consumption E_{lora} for a complete transmission cycle. Furthermore, the collision probability p_{coll} needs to be considered. This leads to

$$E_{\text{eff}} = \frac{E[E_1] \cdot (1 - p_{\text{coll}})}{E_{\text{lora}}} = \frac{E^*}{E_{\text{lora}}}. \tag{26}$$

We can extend the energy efficiency consideration by adjusting the required energy to transmit data. Instead of using $E[E_1]$ as expected required energy to transmit a message, E^* can be defined as minimal possible required energy to successfully transmit a message. With this adjustment, additional parameters like message overhead as a result of the coding rate or the message header can be included. Furthermore, the influence of the spreading factor, unnecessarily large payload, and message collision can be considered. Thus, with the adjustment, further studies on the Quality of Information in a LoRaWAN can be conducted by comparing the minimal required payload, spreading factor, and thus message transmission time on air with the minimal possible energy consumption. However, such studies directly influence the collision or loss probability, e.g., if the used spreading factor is adjusted. For that reason we focus on energy efficiency studies without additional message parameter or spreading factor adjustments to compare the channel access approaches only.

To this end, we can answer our second research question RQ2. The key metric to quantify the energy consumption to transmit data is the time on air of LoRa messages. To achieve results for energy efficiency, in addition, the collision probability is of major interest. For a further insight in the behavior of all channel access approaches, the probability to open, and the duration of receive windows is of interest. Finally, to determine channel access specific results, the device clock drifts influence the required number of resynchronization messages that must be received when the time scheduled approach is used and the number and duration of back-offs for listen before talk influence the general energy consumption.

IV. Simulation Study and Scenarios

To validate the presented model and outline influencing parameters leading to variances in the results or deviations from the model, a LoRaWAN channel access simulation is implemented. First, the simulation approach is introduced and afterwards, channel access specific characteristics, simulation parameter, and evaluation scenarios are defined and discussed.

A. General Simulation Approach

The general simulation approach contains four steps, introduced in the following.

Step 1—Establish Sensors: First, all sensors are established with their locations. Therefore, a center gateway at locations $x = y = 0$ is established, and all sensors are evenly placed around it up to the maximal transmission distance with spreading factor 12. To calculate the distance a sensor can transmit with a specific spreading factor, the Hata path loss model is used [40]. The model is commonly used for LoRa or other mobile communication types while the usage of other models is described in literature [41] and can be applied interchangeably. For the model, we apply a gateway height and a sensor height of 3 m to achieve bi-directional communication. Furthermore, a tolerable Received Signal Strength Indicator of −131, −134, −137, −140, −141, and −144 dBm for spreading factors 7 to 12, respectively, and an urban area is used. Then, the possible transmission distance for spreading factors 7 to 12 is 715, 843, 995, 1174, 1240, and 1463 m. Thus, the maximal transmission distance with spreading factor 12 is 1463 m based on the model. Note that neither the used path loss model nor the parameter settings influence the model or the statement of this work. Other models and parameters can be used interchangeably. The same holds true if sensors are distributed differently. Afterwards, a

spreading factor is assigned to each sensor dependent on its distance to the center gateway and the Hata model distances.

Step 2—Access Specific Setup: In this step, all channel access specific settings are processed. Since the behavior and the energy consumption of the time slotted approach can be calculated based on the message length, the slot length, and the clock drifts, no simulation study is done. To compare our results with a simulation study, we refer to [7]. For listen before talk, it is required to know which sensors are in direct transmission range and thus, which message transmissions can be detected when listened on the channel. These sensors are determined for all listen before talk sensors by the possible transmission distance and the location of the sensors considering the hidden node problem as visualized in Fig. 3.

Step 3—Message Transmission: When everything is established, a random payload in the range of 1 B – 51 B is assigned to each sensor to emulate different transmission behaviors. This range is chosen since it is the maximal range a sensor can transmit with spreading factor 12. All other LoRa message parameters are kept as defined in the background Section II. Then, the time on air for all transmitted messages is calculated. Please note that the time on air calculation and all required parameters are independent on the channel access approach for better comparability. For each transmission, a transmission start time is calculated as a random time between 0 and one transmission time frame. Each sensor transmits once in this time frame. By adding the calculated time on air to the transmission start time, the transmission end time is achieved and thus, the time each sensor blocks the frequency band. If the transmission end time of a transmission exceed the time frame, it is considered again in the following one.

Step 4—Collision Simulation: After all transmissions are set up, each sensor transmits with its predefined approach according a predefined scenario. If messages overlap in time, they are considered as message collisions. Furthermore, the number and duration of back-offs is monitored during the listen before talk approach. After the simulation, different performance metrics are extracted based on the evaluation scenario to calculate, among others, energy consumption and energy efficiency of the approaches.

B. Scenario Description

To evaluate the presented model, compare the channel access approaches, and conduct comprehensive results regarding energy consumption and efficiency of different LoRaWAN channel access approaches, different scenarios are defined with specific goals, introduced in the following.

S1—Model Validation and Variation Determination: In the first scenario S1, the presented energy model and the listen before talk modeling approach is validated by the simulation. Key influencing metrics on the results are discussed and variances between model and simulation are quantified. These key metrics are the spreading factor, the payload, the collision probability, the sensors in range that can be heard to avoid collisions in listen before talk, and their respective spreading factor. Variations of these metrics have a direct influence on the resulting energy consumption and efficiency.

S2—Channel Access Approach Comparison: The focus of scenario S2 is on the different channel access approaches. For each approach, variable parameters are studied to obtain the most energy efficient parameter setting. For random access, the number of sensors, the probability and duration before a receive window is opened, and the actual receive window length can be varied. When listen before talk is used, in addition, different sensing and back-off duration is investigated. Lastly, the main focus for time scheduled is on the probability to receive resynchronizations. This can be influenced by the slot length and the average clock drift of sensors.

S3—Energy Value Adjustment and Approach Comparison: In the last and third scenario S3, the goal is to determine the best channel access approach for different LoRaWAN deployments and discuss benefits and drawbacks of the approaches. Therefore, to consolidate our model with realistic energy values for transmission, message reception, and wait operations, we consulted the work of other authors [27], [28]. From these, we extracted a range from 18 to 28 mA for the power consumption during transmission and 10.5 mA for receive. During waiting or idle periods the transceiver needs 2.033 mA. Based on this, the power consumption to receive data is varied between 30% and 60% of transmit, and wait requires between 7% and 12% of transmit. With these values as input, we compare the different channel access approaches, random access, listen before talk, and time scheduled.

General Simulation and Evaluation Settings: For each simulation in each scenario, we simulate between 50 and 800 sensors in steps of 50 to achieve results with small and larger load and different expected collision probability in the network. To obtain statistically significant results, 200 simulation runs are conducted for each individual parameter setting. The duration of a single simulation run is 3600 s and each sensor transmits once in this time frame. This is valid for a sufficiently large number of devices according to Metzger et al. [33]. The locations for all sensors and the payload is reassigned 20 times to investigate the influence of different payloads and spreading factors.

V. Evaluation

This section evaluates the defined scenarios and answers the remaining research question. Therefore, the energy model is validated by a simulation and key influencing metrics for a LoRa transmission and the energy consumption are presented. Afterwards, an energy efficiency study for different channel access approaches is presented, followed by a numerical evaluation with real energy values from literature.

A. Scenario S1—Model Validation

To validate the model and determine the possible variance in the transmission behavior influencing the presented energy model, first, key transmission metrics are discussed. If the general structure of a LoRa message is not adjusted and the focus is only on the transmitted data and the distance to the gateway, the message payload and the used spreading

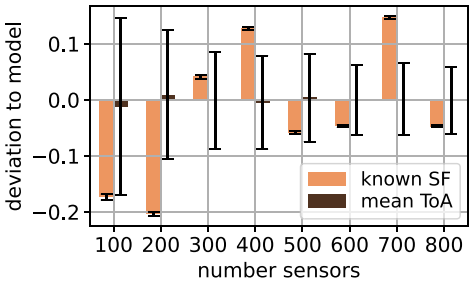

Fig. 4. Comparison random access: Impact of payload and spreading factor.

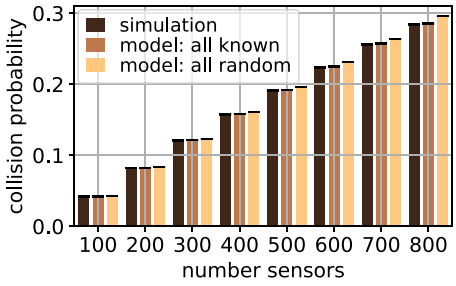

Fig. 5. Comparison random access: Collision probability.

factor for the transmission are of key interest. Both influence the transmission time on air and thus, the energy consumption during a LoRa transmission.

Spreading Factor and Payload: For that reason, the expected energy consumption E_1 for the actual LoRa transmission is calculated first. Besides a random payload assignment between 1 B and 51 B, the spreading factor the sensors transmit with is achieved for each sensor by placing them in a circular way around the gateway, as described in the first step of the simulation approach in Section IV-A. This leads to 23.872% of all sensors transmitting with spreading factor 7, 9.374% with spreading factor 8, 12.951% with spreading factor 9, 18.101% with spreading factor 10, 7.520% with spreading factor 11, and 28.181% with spreading factor 12. By assigning a random payload to all sensors, an average time on air of 0.789 s is achieved. This is the average time in the transmission procedure described by the random variable T_1 in (2). The minimal possible time on air t_{min} is 0.029 s and thus, T_1^* in (2) is 27.268 by a normalization of the average time on air with the minimal possible time on air. All results are rounded to three digits after the comma to simplify reading.

This average theoretical value is compared to the result from the simulation study for random access in Fig. 4 for different numbers of sensors in steps of 100 for better readability. The *y*-axis shows the deviation between the model and the simulation, the *x*-axis shows the number of sensors. The errorbars are the 90% confidence interval. A positive deviation indicates higher value by the simulation compared to the model, and vice versa. For the *known SF* result, presented by the orange bar, the spreading factor is achieved from the sensor and only the payload is unknown. Thus, the time on air is calculated for the model with a random payload and the known spreading factor. For the *mean ToA* result, presented by the brown bar, both the payload and the spreading factor are not known in the model and the simulation result is compared to the average theoretical value of $T_1^* = 27.268$.

First, the results show a small difference between model and simulation. The maximum deviation is about $-0.2 \cdot t_{\text{min}}$, and thus, only $0.2 \cdot 0.029$ s. Second, we see that knowing the spreading factor reduces the variance a lot but increases the average deviation between simulation and the model. If an average time on air is used, the model matches with the simulation result very accurately on average and the confidence intervals always overlap with zero deviation. With a variance of less than $0.2 \cdot t_{\text{min}}$, the result is also acceptable. Consequently, the result shows that knowing the spreading factor is a good indicator for the model in the first glance. This knowledge can narrow down the possible time on air. However, if a more accurate prediction is required, it is essential to know the exact time on air determined by the spreading factor and the payload. We can conclude that the model can handle the large variance in the time on air for LoRaWAN and we can describe the transmission process accurately.

Collision Probability: Next, the collision probability is of key interest, as it determines the percentage of messages arriving correctly at the receiver side. This value also influences the total energy efficiency of the channel access approaches. Therefore, Fig. 5 presents a comparison between simulation and modeling results to answer the question, whether the model can deal with the large variance of the time on air of LoRa transmissions in the average case if the collision probability is the target metric. The *y*-axis of the figure represents the collision probability and the *x*-axis, again, the number of sensors in steps of 100. The black bar presents the simulation results, the brown bar the modeling result, where the spreading factor and the payload is known. Both are compared to the yellow line, where again the average time on air of 0.789 s is applied. This is similar to no prior knowledge about the transmission time on air of the sensors. The 90% confidence intervals are very small and show thus, only small variance in the results. First, as expected, the collision probability is increasing for both the model and the simulation with an increasing number of sensors. If we compare 100 and 200 sensors, the increase in collision probability nearly doubles while the increase flattens down slowly with more sensors. This behavior is already visible with increasing load in LoRaWAN in [13]. Furthermore, we see that the simulation matches with the model if the time on air is known with less than 0.2% deviation in the worst case. The confidence intervals for the simulation and the model, where all information is known overlap for all number of sensors, except 800. Thus, we see no statistically significant difference. If the time on air is not known, highlighted by the yellow bar, the model matches with the simulation or overestimates it slightly. Differences are between less than 0.1% and 0.9%. This minor difference is a result of the large variance in the time on air for all transmissions. Thus, we can conclude that the model can deal with the large variance of the time on air as a result of different spreading factors also when we model the collision probability. This is relevant for all channel access approaches but with this insight, the model is validated for random access.

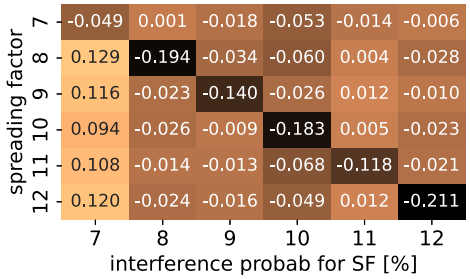

Fig. 6. Comparison listen before talk: Hidden node study.

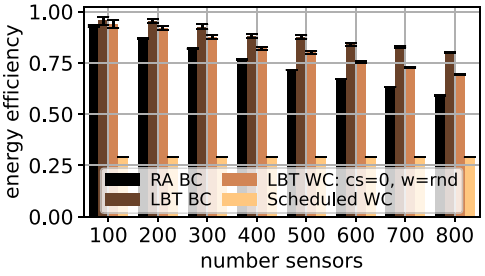

Fig. 7. S2—theoretical comparison: Energy efficiency of different approaches.

Hidden Node Problem: To model listen before talk, the validation must be extended to cover the hidden node problem. Therefore, the question is answered, whether the model is accurate if the collision avoidance probability is determined. The result is presented in Fig. 6. The figure shows the difference in percent between the model and the simulation for the probability of a sensor transmitting with a specific spreading factor *SFx* (*y*-axis) that it is heard by another sensor transmitting with *SFy* (*x*-axis). This is a measure for collision avoidance potential, as all sensors that hear each other can avoid collisions using listen before talk. For example, the 0.053% value in the top column of Fig. 6 is the probability difference between the simulation and the model that a sensor transmitting with spreading factor 7 can be heard by any other sensor transmitting with spreading factor 10. The smallest difference of 0.001% is visible when a sensor is transmitting with spreading factor 7 that can be heard by other sensors transmitting with spreading factor 8 shown in the top column. Furthermore, we see for more then 10% of the presented combinations a difference of less than 0.01% and for more than 70% a deviation of less than 0.1%. The largest difference is detected, when a sensor is transmitting with spreading factor 12 and whether it is heard by another sensor in the spreading factor 12 area. However, with about 0.2% deviation, the model is validated as working correctly with only minor mismatch for each single spreading factor. Furthermore, the average probability that a random sensor hears any other sensor is calculated as 35.13% by the model. Using the simulation, the results are between 34.76% and 35.37% for 50 to 800 sensors in steps of 50, and as a result, very small for all numbers of sensors. For that reason, if a sensor does not reattempt its transmission, both the collision probability and the probability for an unsuccessful transmission attempt can be modeled.

B. Scenario S2—Channel Access Approach Comparison

Besides model validation, we investigate different channel access approaches by means of energy efficiency. Therefore, the channel access approaches random access, listen before talk, and time scheduled are compared in the following with different parameter settings. Since the sensor location and thus, spreading factor assignment and payload determination is the same for each approach, only the performance of the channel access influences the energy efficiency.

Energy Efficiency Comparison: For that reason, different parameter settings for the three channel access approaches, random access, listen before talk, and time scheduled are compared initially, summarized in Fig. 7. The *y*-axis represents the energy efficiency defined in (26), with an energy efficiency $E_{\text{eff}} = 1$ defined as optimal case and a collision free data transmission without any wait, back-off, listen, or reception time. The errorbars indicate the 90% confidence intervals again. The *x*-axis plots the number of sensors and the different colors represent different parameter settings for the channel access approaches. The black bar shows the best case random access (RA BC) scenario from an energy efficiency point of view. There, no receive window is opened and thus, only data transmission is considered. The deviation from the most energy efficient transmission is only a result of message collisions, increasing with more sensors because of a load increase with more messages in the network. Similarly, the best case listen before talk (LBT BC) depicts listen before talk with listening on the channel, a random back-off between 0.4 and 1.75 s, as suggested in [7] if the channel is occupied, and message transmission if no other message is detected. In this best case scenario, it is assumed that both listening on the channel and any back-off consumes no energy. Thus, the difference between the base case random access and the best case listen before talk presents the maximal possible energy efficiency improvement potential if listen before talk is used.

Since in reality, additional listening on the channel and back-off consumes energy, this is included in the next listen before talk scenario, presented by the orange bar (LBT WC; $cs = 0$, $w = \text{rnd}$). The cs parameter describes the channel listening duration, and the w parameter the wait duration for the back-off if another message has been detected. Thus, if $cs = 0$, it is only checked whether the channel is free at the point in time, a sensor attempts to transmit data. In this example, we assume no energy consumption for this procedure. However, if another message is detected, a back-off is started and energy is consumed. Here, we assume the same energy consumption for the same duration of wait and transmit as worst case assumption.

The last scenario discussed in Fig. 7 is the worst case time scheduled (Scheduled WC). An analysis of the best case time scheduled approach is pointless, since its energy efficiency is equal to 1. In that case, no collisions occur and no resynchronization is required because of no device clock drifts. In the theoretical worst case, each sensor is resynchronized after each transmission which leads to an open receive window wait time and a data reception after each transmission. In this worst case assumption, we further assume that the energy required

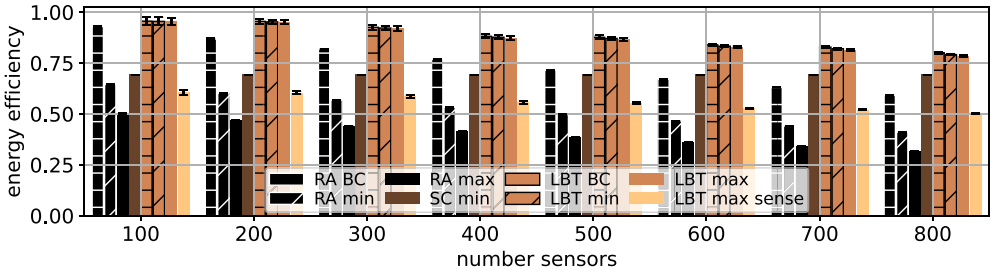

Fig. 8. S3—realistic energy values from literature: Comparison of energy efficiency of different approaches.

for wait and receive is equal to transmit. The wait time is set to 1 s before a reception window is opened, stated as default value to open the first receive window in literature [12]. The reception duration is set to 0.926 s as maximal duration to transmit data with spreading factor 12. Thus, we guarantee that a sensor can receive data, independently on the used spreading factor. The results show a drop in the energy efficiency of time scheduled down to 29% in the worst case. However, there is different improvement potential by limiting the reception or wait duration, consuming less energy during wait or receive, or trigger the resynchronization process less frequently by less clock drift of the sensors or larger slots.

In general, we can draw several conclusions from these studies. First, when the load in the network is low, random access performs similar to listen before talk from an energy efficiency point of view. In this case, only few collisions occur anyway. Second, time scheduled performs, although avoiding all collisions, much worse than listen before talk and random access in the worst case for all investigated loads. However, there is improvement potential to perform a lot better than the worst case setup. Third, if the load is increased, listen before talk improves in general against random access if the energy requirement to listen on the channel is not taken into consideration. Consequently, we already see statistically significant different energy efficiency results comparing listen before talk and random access having 200 sensors in the network. Thus, listen before talk is a viable alternative to random access with large load in the network if we analyze LoRaWAN from an energy efficiency point of view. Lastly, the performance of listen before talk is highly dependent on the energy consumption ratio for the back-off and for data transmission. If additional consumption during back-off is small, the resulting energy efficiency converges toward the brown bar. In contrast, if it is high, it converges toward the orange bar and an energy consumption maximum during back-off similar to the value during transmit can be assumed. Please note, to determine the final energy efficiency of listen before talk we also need to include the consumption during listening on the channel discussed in the following.

Listen Before Talk Energy Efficiency Study: To study the energy efficiency of listen before talk, the impact of different listening and back-off duration on the collision probability and the total back-off duration is investigated. Longer channel listening time and additional back-offs is only meaningful if in return, the collision probability or the total back-off duration is reduced. Our simulation results show that a longer back-off does not impact the collision probability since it only changes the next transmission attempt time and does not modify the load in the network. In contrast, longer channel listening duration even increases the average collision probability. This behavior is expected since longer channel listening requires a free channel for a longer time to start another transmission. Thus, short free channel slots are not used and total load increases. Taking the total back-off duration and the total number of back-offs into consideration, a good tradeoff is already achieved with a random back-off between 0.4 and 1.75 s as described in [7]. A longer listening time detects more messages but again, also avoids short free channel slots. Thus, the total number of back-offs and the total back-off duration is increased. For that reason, a minimal duration to listen on the channel followed by a back-off between 0.4 and 1.75 s is suggested for listen before talk. To this end, the general energy efficiency for listen before talk is delimited by the energy efficiency of the best case listen before talk (LBT BC) and the worst case listen before talk with up to no channel listening time and the random back-off (LBT WC; $cs = 0$, $w = \text{rnd}$), as plotted in Fig. 7. Note, in addition to the energy efficiency for LBT WC; $cs = 0$, $w = \text{rnd}$, the required additional energy to listen on the channel must be added. This impairs the overall worst case energy efficiency for an additional percentage, dependent on the energy requirement for channel listening and is considered in the following.

C. Scenario S3—Realistic Energy Value Study

In the last evaluation scenario, the energy efficiency for the different channel access approaches is compared, using realistic energy values from literature [27], [28]. The result is plotted in Fig. 8 for different parameter settings. The best case random access (RA BC), with no receive windows, and the best case listen before talk (LBT BC), with no required energy for the listening and the back-off process are taken from the previous evaluation as reference values. Both bars are plotted with horizontal line markers for better visibility. The results are compared to minimal (min, diagonal line marker) and maximal (max, no line marker) consumption results for random access and listen before talk. In the minimal (min) case, the power consumption to receive data is 30% of transmit as defined as minimal value in the scenario definition (Section IV-B). For wait, 7% is used. In the maximal (max) case, 60% and 12% is used for receive and wait, respectively. In addition, the

time scheduled approach with minimal energy consumption for wait and receive is plotted by the brown bar. Furthermore, the *LBT max sense* scenario, presented by the yellow bar in the figure, also considers the energy consumption of a one second listening interval. An additional energy consumption of 0.45% of transmit is added to listen on the channel, as average receive energy consumption. The results show that, again, listen before talk performs similar in little load situations and better than random access for high load, if no energy is required to listen on the channel. In the channel listening case (LBT max sense), listen before talk performs worse than best case random access (RA BC) but similar or better than random access with receive windows (RA min and RA max). Moreover, the decline of the energy efficiency using random access or listen before talk stems from two components. First, the message collisions and second, the overhead from other operations, including the opening of a receive window, when we need to listen on the channel, or when the sensor has to wait. The number of collisions increase with an exponential relationship to the load in the network [37], and thus, to the number of sensors. In contrast, the overhead from other operations increases linearly. Consequently, the decline using random access, and in particular using the best case random access (RA BC) where the energy efficiency decline only stems from collisions, is faster than when we use listen before talk. If we further compare random access and listen before talk, more message collide with smaller load using random access. Consequently, already a minor load increase in the network has a larger influence on the collision rate, and thus the energy efficiency of random access. We can see this behavior when we compare the different black bars for random access with a different number of sensors in the network, in particular comparing 100 sensors and 200 sensors, where the largest difference is achieved. Furthermore, the open receive window and receive procedure is similar for random access and the time scheduled approach. Though, in contrast to random access, no collisions occur when time scheduled is used. This leads to a better energy efficiency than random access with receive windows if the load is small and even improvement against best case random access for a large load and many sensors. If data reception is only assumed when a message is successfully transmitted to the gateway without any collision, the energy efficiency for random access is improved by up to 3% for 800 sensors (RA min scenario). For RA max, it is up to 4% improvement. In the time scheduled case, the energy efficiency can be improved by 10% if every second transmission is resynchronized only. A further improvement by 7% and close to 10% is achieved if only 20% and 10% of all message must be resynchronized. With 10% message resynchronization rate, we achieve an energy efficiency of 89%. Finally, we can answer our last research question RQ3 as follows. The most energy efficient channel access approach is random access if the load in the network is small, no receive windows are required, and no data collide. If message collision is taken into consideration, listen before talk improves energy efficiency against random access if the energy required during channel listening can be minimized. Nevertheless, the best channel access approach is time scheduled, as it can avoid all collisions and works in a very energy efficient way if the resynchronization rate as a result of sensor's clock drifts is small.

D. Example Calculation for Practical Usability

A network provider wants to deploy sensors for a LoRaWAN. Before the deployment starts, the question arises which channel access approach has to be chosen in terms of energy efficiency. Our evaluations with realistic values from literature show in Fig. 8 that the usage of different channel access mechanisms has a significant influence on the energy efficiency and thus, on potential deployment decisions. In the following, we perform an example calculation to demonstrate this impact in a real LoRaWAN of an example provider. According to energy consumption measurements from literature [27], the energy demand to transmit LoRa messages is up to 39.43 mA. The average time on air to transmit LoRa messages with spreading factor seven and a random payload between 1 B and 51 B is 89.81 ms. Consequently, this is the shortest average duration a transceiver needs to be powered to transmit messages. Note, we use the range of 1 B to 51 B payload since this payload can be transmitted with all available spreading factors. Multiplying the energy demand with the transmission duration, we require 3.54 mA on average with perfect energy efficiency for one transmission using 39.43 mA. In addition, 2.268 mA is required to wake up the transceiver [27]. If we assume the usage of a cheap and tiny battery with 500 mAh where we can use 85% of the total capacity as suggested by Semtech [42] and 25% of the remaining capacity (106.25 mAh) for transmission only to have additional battery for, among others, sensor operation, sleep mode, and data processing, we are able to transmit about 108 000 messages with one battery if we consider only message transmission or about 66 000 messages if we also consider transceiver wake up. This is equal to a battery life time of 12.3 years if we do not consider the transceiver wake up and 7.5 years if we consider transceiver wake up, respectively. Consequently, the battery can run 7.5 years with perfect energy efficiency (energy efficiency equal to 1.00 in Fig. 8) and a message transmission once an hour if we consider transceiver wake up. In contrast, applying the best case usage of random access (RA BC), we achieve a battery life time of 7.18 years for 100 sensors and 5.29 years for 800 sensors with the values from Fig. 8 if we would like to transmit the same number of messages without errors. If we consider 800 sensors and BC max, we can only achieve a battery life time of 4.00 years. Consequently, the selection of an energy efficient channel access approach has a massive impact on the battery life of sensors, and in return the service quality of a LoRaWAN, the frequency a provider has to exchange batteries or the complete sensors, and thus, the resource demand for new sensors in such a networks.

VI. Conclusion

The recent opening of Amazon Sidewalk to the public using LoRaWAN for Long Range communication will further foster the adoption of LoRa. However, to work reliably and in an energy efficient way, alternatives to the error-prone

random channel access are currently researched. Listen before talk and a time scheduled channel access have been proven as suitable alternatives but an in-depth and general energy study was missing. We close this gap and propose a general and hardware-independent energy consumption and energy efficiency model for LoRaWAN describing data transmission, wait times, and data reception. Furthermore, we model random access, listen before talk, and time scheduled and compare the approaches in this work. Our results show that the time on air and the collision probability, as the most important metrics to determine the energy efficiency, can be modeled very accurately. The hidden node model for listen before talk achieves only a deviation of 0.2% to a simulation in the worst case. Our general energy efficiency study suggests to use random access only, if no additional receive windows are opened and network load is small. Listen before talk can improve energy efficiency against random access by up to 20% in high-load situations if minimal energy is required to listen on the channel. Thus, we suggest to focus on energy consumption reduction to listen on the channel in the listen before talk case. Lastly, the time scheduled approach is the best choice, if the clock drift of sensors is small. Nevertheless, this is the most complex approach and further studies are required on the additional energy demand at the sensor to resynchronize the clock. In addition, not all sensors are capable of clock resynchronization or some clock drifts can be too large for a practical usage.

REFERENCES

[1] "Amazon sidewalk paves the way for more connected communities." Accessed: Apr. 6, 2023. [Online]. Available: https://developer.amazon.com/en-US/blogs/alexa/device-makers/2020/09/amazon-sidewalk-paves-the-way-for-more-connected-communities

[2] *Enabling Amazon Sidewalk with LoRa*, Semtech, Camarillo, CA, USA, Accessed: Apr. 5, 2023. [Online]. Available: https://info.semtech.com/sidewalk

[3] "Coverage sidewalk Amazon." Accessed: Apr. 5, 2023. [Online]. Available: https://coverage.sidewalk.amazon/

[4] *LoRaWAN Market Report*, Global Market Insights Inc., Selbyville Delaware, USA, Accessed: Apr. 5, 2023. [Online]. Available: https://www.gminsights.com/industry-analysis/lorawan-market

[5] "What are LoRa and LoRaWAN." Accessed: Apr. 5, 2023. [Online]. Available: https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/

[6] L. Beltramelli, A. Mahmood, P. Österberg, and M. Gidlund, "LoRa beyond ALOHA: An investigation of alternative random access protocols," *IEEE Trans. Ind. Inform.*, vol. 17, no. 5, pp. 3544–3554, Feb. 2020.

[7] F. Loh, N. Mehling, and T. Hoßfeld, "Towards LoRaWAN without data loss: Studying the performance of different channel access approaches," *Sensors*, vol. 22, no. 2, p. 691, 2022.

[8] J. Ortín, M. Cesana, and A. Redondi, "Augmenting LoRaWAN performance with listen before talk," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3113–3128, Jun. 2019.

[9] L. Casals, B. Mir, R. Vidal, and C. Gomez, "Modeling the energy performance of LoRaWAN," *Sensors*, vol. 17, no. 10, p. 2364, 2017.

[10] T. Bouguera, J.-F. Diouris, J.-J. Chaillout, R. Jaouadi, and G. Andrieux, "Energy consumption model for sensor nodes based on LoRa and LoRaWAN," *Sensors*, vol. 18, no. 7, p. 2104, 2018.

[11] F. Loh, S. Raffeck, S. Geißler, and T. Hoßfeld, "Generic model to quantify energy consumption for different LoRaWAN channel access methods," in *Proc. Int. Conf. Wireless Mobile Comput., Netw. Commun.*, 2022, pp. 290–295.

[12] D. Bankov, E. Khorov, and A. Lyakhov, "On the limits of LoRaWAN channel access," in *Proc. Int. Conf. Eng. Telecommun.*, 2016, pp. 10–14.

[13] F. Loh, S. Raffeck, F. Metzger, and T. Hoßfeld, "Improving LoRaWAN's successful information transmission rate with redundancy," in *Proc. Int. Conf. Wireless Mobile Comput., Netw. Commun.*, 2021, pp. 313–318.

[14] "LoRaWAN frequency plans." Accessed: Jan. 25, 2023. [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/frequency-plans/

[15] F. Loh, N. Mehling, S. Geißler, and T. Hoßfeld, "Efficient graph-based gateway placement for large-scale LoRaWAN deployments," *Comput. Commun.*, vol. 204, pp. 11–23, Apr. 2023.

[16] M. A. A. Khan, H. Ma, S. M. Aamir, and Y. Jin, "Optimizing the performance of pure ALOHA for LoRa-based ESL," *Sensors*, vol. 21, no. 15, p. 5060, 2021.

[17] J. Haxhibeqiri, F. Van den Abeele, I. Moerman, and J. Hoebeke, "LoRa scalability: A simulation model based on interference measurements," *Sensors*, vol. 17, no. 6, p. 1193, 2017.

[18] F. Loh, N. Mehling, S. Geißler, and T. Hoßfeld, "Simulative performance study of slotted aloha for LoRaWAN channel access," in *Proc. Netw. Oper. Manage. Symp.*, 2022, pp. 1–9.

[19] X. Xia, Y. Zheng, and T. Gu, "FTrack: Parallel decoding for LoRa transmissions," in *Proc. Conf. Embed. Netw. Sens. Syst.*, 2019, pp. 192–204.

[20] F. Loh, C. Baur, S. Geißler, H. ElBakoury, and T. Hoßfeld, "Collision and energy efficiency assessment of LoRaWANs with cluster-based gateway placement," in *Proc. Workshop Green Sustain. Netw.*, 2023, pp. 391–396.

[21] D. Bankov, E. Khorov, and A. Lyakhov, "Mathematical model of LoRaWAN channel access with capture effect," in *Proc. Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun.*, 2017, pp. 1–5.

[22] T. Polonelli, D. Brunelli, A. Marzocchi, and L. Benini, "Slotted ALOHA on LoRaWAN-design, analysis, and deployment," *Sensors*, vol. 19, no. 4, p. 838, 2019.

[23] A. Xanthopoulos, A. Valkanis, G. Beletsioti, G. I. Papadimitriou, and P. Nicopolitidis, "On the use of backoff algorithms in slotted ALOHA LoRaWAN networks," in *Proc. Int. Conf. Comput., Inf. Telecommun. Syst.*, 2020, pp. 1–4.

[24] I. Cheikh, E. Sabir, R. Aouami, M. Sadik, and S. Roy, "Throughput-delay tradeoffs for slotted-ALOHA-based LoRaWAN networks," in *Proc. Int. Wireless Commun. Mobile Comput.*, 2021, pp. 2020–2025.

[25] H. H. R. Sherazi, L. A. Grieco, M. A. Imran, and G. Boggia, "Energy-efficient LoRaWAN for industry 4.0 applications," *Trans. Ind. Inform.*, vol. 17, no. 2, pp. 891–902, Feb. 2021.

[26] T. Hoßfeld, S. Raffeck, F. Loh, and S. Geißler, "Analytical model for the energy efficiency in low power IoT deployments," in *Proc. Int. Conf. Netw. Softwarizat.*, 2022, pp. 19–24.

[27] S. Maudet, G. Andrieux, R. Chevillon, and J.-F. Diouris, "Refined node energy consumption modeling in a LoRaWAN network," *Sensors*, vol. 21, no. 19, p. 6398, 2021.

[28] R. K. Singh, P. P. Puluckul, R. Berkvens, and M. Weyn, "Energy consumption analysis of LPWAN technologies and lifetime estimation for IoT application," *Sensors*, vol. 20, no. 17, p. 4794, 2020.

[29] J. Finnegan, S. Brown, and R. Farrell, "Modeling the energy consumption of LoRaWAN in ns-3 based on real world measurements," in *Proc. Global Inf. Infrastruct. Netw. Symp.*, 2018, pp. 1–4.

[30] "Device classes." Accessed: Apr. 4, 2023. [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/classes/

[31] F. Loh, N. Mehling, S. Geißler, and T. Hoßfeld, "Graph-based gateway placement for better performance in LoRaWAN deployments," in *Proc. Mediterr. Commun. Comput. Netw. Conf.*, 2022, pp. 190–199.

[32] F. Loh, D. Bau, J. Zink, A. Wolff, and T. Hoßfeld, "Robust gateway placement for scalable LoRaWAN," in *Proc. IFIP Wireless Mobile Netw. Conf.*, 2021, pp. 71–78.

[33] F. Metzger, T. Hoßfeld, A. Bauer, S. Kounev, and P. E. Heegaard, "Modeling of aggregated IoT traffic and its application to an IoT cloud," *Proc. IEEE*, vol. 107, no. 4, pp. 679–694, Apr. 2019.

[34] C. Palm, *Intensitatsschwankungen Im Fernsprechverker*. Stockholm, Sweden: Ericsson, 1943.

[35] T. Phung-Duc, "Retrial queueing models: A survey on theory and applications," 2019, *arXiv:1906.09560*.

[36] A. Nazarov, J. Sztrik, and A. Kvach, "A survey of recent results in finite-source retrial queues with collisions," in *Proc. Int. Conf. Inf. Technol. Math. Model.*, 2018, pp. 1–15.

[37] P. Tran-Gia and T. Hoßfeld, *Performance Modeling and Analysis of Communication Networks, A Lecture Note*, Würzburg, Germany: Würzburg Univ. Press, 2021.

[38] P. Wüchner, J. Sztrik, and H. de Meer, "Finite-source retrial queues with applications," in *Proc. Int. Conf. Appl. Inform.*, 2010, pp. 275–285.

[39] P. Wüchner, H. de Meer, J. Barner, and G. Bolch, "A brief introduction to MOSEL-2," in *Proc. GI/ITG Conf. Meas., Modelling Eval. Comput. Commun. Syst.*, 2006, pp. 1–4.

[40] M. Hata, "Empirical formula for propagation loss in land mobile radio services," *Trans. Veh. Technol.*, vol. 29, no. 3, pp. 317–325, Aug. 1980.

[41] F. Loh, N. Mehling, F. Metzger, T. Hoßfeld, and D. Hock, "LoRaPlan: A software to evaluate gateway placement in LoRaWAN," in *Proc. Int. Conf. Netw. Service Manage.*, 2021, pp. 385–387.
[42] "Example: Select a battery." Accessed: Jan. 17, 2024. [Online]. Available: https://lora-developers.semtech.com/documentation/tech-papers-and-guides/the-book/example-select-a-battery/

Frank Loh (Member, IEEE) received the Ph.D. degree from the University of Würzburg, Würzburg, Germany, in 2024, for his thesis "Monitoring the Quality of Streaming and Internet of Things Applications."

He is currently with the Chair of Communication Networks, where he leads the "Green Communication Networks" Research Group. His current research includes performance analysis and investigation, Quality of Service and Quality of Experience assessment, and energy and sustainability quantification for current and future communication networks. There, he studies a greener way of communication without impacting the user perceived application quality.

Simon Raffeck received the master's degree from the University of Würzburg, Würzburg, Germany, in 2022, where he is currently pursuing the Ph.D. degree.

He is currently with the Chair of Communication Networks, University of Würzburg, where he is a part of the Cloud Applications and Networks Research Group. His current research is focused on the analytical and simulative performance evaluation and optimization of next-generation communication technologies, including 5G and beyond, Internet of Things, and TSN, as well as energy efficiency of wireless communication and systems.

Stefan Geißler received the Ph.D. degree for his thesis "Performance Evaluation of Next-Generation Data Plane Architectures and their Components" from the University of Würzburg, Würzburg, Germany, in 2022.

He is a Research Associate with the Chair of Communication Networks, University of Würzburg, where he leads the Cloud Applications and Networks Research Group. His current research is focused on the analytical and simulative performance evaluation and optimization of next-generation communication technologies, including TSN, 5G, and Internet of Things.

Tobias Hoßfeld (Senior Member, IEEE) received the Ph.D. degree from the University of Würzburg, Würzburg, Germany, in 2009.

He finished his professorial thesis (habilitation) "Modeling and Analysis of Internet Applications and Services" with the University of Würzburg in 2013, where he was also heading the "Future Internet Applications & Overlays" Research Group. He has been a Professor with the Chair of Communication Networks, University of Würzburg since 2018. From 2014 to 2018, he was the Head of the Chair "Modeling of Adaptive Systems" with the University of Duisburg-Essen, Duisburg, Germany. He has published more than 100 research papers in major conferences and journals.

Prof. Hoßfeld received five best conference paper awards, three awards for his Ph.D. thesis, and the Fred W. Ellersick Prize 2013 (IEEE Communications Society) for one of his articles on QoE. He is a member of the advisory board of the ITC conference and the editorial board of IEEE COMMUNICATIONS SURVEYS & TUTORIALS.