Supporting Critical Downlink Traffic in LoRaWAN

Dimitrios Zorbas[a], Aruzhan Sabyrbek[a]

[a]*Nazarbayev University, School of Engineering & Digital Sciences, Astana, Kazakhstan*

Abstract

LoRaWAN, a low-power wide-area network (LPWAN) technology, has been successfully used in the Internet of Things (IoT) industry over the last decade. It is an easy-to-use, long-distance communication protocol combined with minimal power consumption. Supporting critical downlink traffic in LoRaWAN networks is crucial for ensuring the reliable and efficient delivery of essential data in certain actuating applications. However, challenges arise when prioritizing critical downlink traffic, including commands, alerts, and emergency notifications that demand immediate attention from actuating devices. This paper explores strategies to improve downlink traffic delivery in LoRaWAN networks, focusing on enhancing reliability, fairness, and energy efficiency through prioritization techniques and network parameter configurations in the EU868 spectrum. Theoretical as well as simulation results provide insights into the effectiveness of the available solutions for supporting critical downlink traffic in LoRaWAN networks.

Keywords: Internet of Things, LoRa, LoRaWAN, Actuators, Downlink, Capacity, Fairness, Simulations

1. Introduction

Supporting critical downlink traffic in LoRaWAN networks is imperative for ensuring the reliable and efficient delivery of essential data in various applications such as smart cities, industrial monitoring, and healthcare [1]. LoRaWAN is well-suited for transmitting small packets of data over long distances while consuming minimal power. However, challenges arise when prioritizing critical downlink traffic, which typically includes commands, alerts, and emergency notifications that require immediate attention from actuating devices [2, 3].

To enhance downlink traffic in LoRaWAN, network operators and developers employ several strategies. One approach involves the deployment of additional gateways, which consequently increase the downlink capacity [4], but also increase the expenditure and operating costs. Moreover, by assigning higher priority levels to critical downlink traffic, LoRaWAN gateways can enhance fairness and reduce latency [5]. Additionally, optimizing network parameters such as the spreading factors and the channel allocation can enhance the reliability and capacity of downlink transmissions, especially in congested or interference-prone environments [6, 7]. Apparently, employing advanced scheduling and synchronization algorithms can further improve the efficiency of downlink traffic delivery, mitigating the impact of network congestion and optimizing resource utilization [8, 9]. These solutions exhibit extra energy cost due to the synchronization overhead of LoRaWAN Class B, while may not be fully LoRaWAN-compliant. LoRaWAN Class A solutions still suffer from extensive collisions due to the Aloha-based medium access method [10, 11]. Apart from that, the lack of gateway time resources due to regional duty cycle restrictions in many regions can extensively delay actuating data transmissions [12].

Unlike the abovementioned works, in this paper, we study the problem of critical downlink traffic over LoRaWAN Class A and C mode networks without the use of any synchronization or scheduling method. We examine the ability of LoRaWAN networks to send downlink commands to actuators acting as Class C devices to perform a task under the presence of typical LoRaWAN Class A traffic. Actuators in our scenario are devices without energy constraints as they typically appear in the industry (e.g., streetlight switches, solar trackers, traffic lights, cameras etc.). To facilitate the co-existence of typical LoRaWAN nodes and actuators, different resource allocation schemes are presented according to the existing EU868 channel availability and channel access rules. Even if the EU868 spectrum is employed in this paper as the baseline, the proposed resource allocation schemes can be applied to other regions as well. The contributions of this paper are summarized as follows:

- Solutions to support actuating downlinks in LoRaWAN are proposed.
- The effectiveness of a newly introduced EU868 spectrum band for downlink in the context of actuation is assessed.
- A probability analysis for the proposed solutions for typical LoRaWAN scenarios is presented.

The rest of the paper is organized as follows. Section 2 presents some LoRa and LoRaWAN fundamentals to in-

Email addresses: dimitrios.zorbas@nu.edu.kz (Dimitrios Zorbas), aruzhan.sabyrbek@nu.edu.kz (Aruzhan Sabyrbek)

troduce non-expert readers to the concepts of this work. The examined scenario is also presented in this section. The proposed resource allocation schemes are explained in Section 4 pointing out their advantages and disadvantages. A probability analysis is also provided and results are drawn based on typical LoRaWAN traffic scenarios. Section 5 presents extensive simulation and comparison results. Finally, Section 6 draws conclusions and ideas for future work.

2. LoRa/LoRaWAN fundamentals & Examined Scenario

2.1. LoRa fundamentals

LoRa is a modulation technique that provides long-range communications combined with low power consumption, making it ideal for IoT devices. It uses a proprietary spread spectrum modulation to transmit data over a wide range of frequencies, allowing for robust communication in noisy environments. Spreading Factors (SF) in LoRa refer to the spreading of the signal over that wide frequency range, which increases the signal robustness and the communication range but reduces the data rate. SFs range typically from SF7 (fastest, lowest energy consumption, and shortest range) to SF12 (slowest, larger energy consumption, and longest range).

A LoRa packet consists of the preamble, an optional header, and the payload. The role of the preamble is to help synchronize, modulate, and demodulate transmissions. In general, it plays a vital role in ensuring that the receiver can reliably decode the incoming LoRa signal and recover the transmitted data. The header contains important information to help the receiver interpret the packet correctly. It contains information about the payload length, a checksum (CRC) that the receiver can use to verify the integrity of the packet, the Coding Rate (CR) of the packet for error correction, and whether the implicit or the explicit mode is used. In the implicit mode, the packet length and CRC are fixed and known in advance. In explicit mode, these values are included in the header.

LoRa packets will most likely collide with each other if they overlap in time, SF, and frequency [13]. There is also a high chance of collision even between packets with different SF due to the imperfect orthogonality [14]. The capture effect helps to reduce that probability either for co-SF or inter-SF transmissions, but usually the signal with higher received power will be decoded [15].

In terms of hardware, typical LoRa transceivers (i.e., SX127x, SX126x) are half-duplex and can receive or transmit data from one channel and one SF at a time. Sub-GHz LoRa gateway transceivers have a more advanced hardware and can listen to all SFs and up to 8 channels at the same time. Nonetheless, they are still half-duplex. All LoRa transceivers are also equipped with a Channel Activity Detection (CAD) mechanism, which is designed to detect the presence of a LoRa preamble for a specific SF in the channel. CAD can be employed as a Carrier Sense Multiple Access (CSMA) approach to alleviate collisions [16].

2.2. LoRaWAN fundamentals

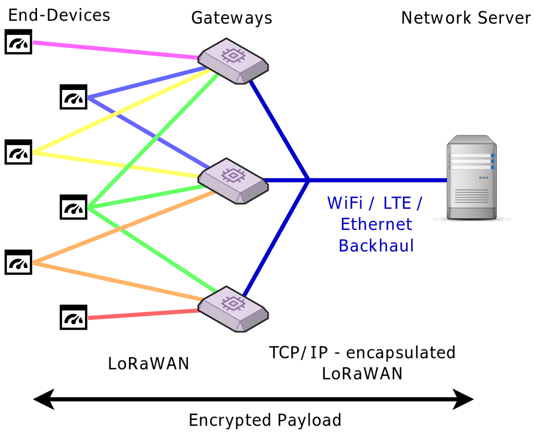


Figure 1: LoRaWAN architecture typically consisting of end-devices, gateways, and a backhaul network with a Network Server.

The LoRaWAN specification is a communication protocol and system architecture designed for LoRa-enabled LPWANs. It defines the communication protocol between IoT devices and the corresponding network infrastructure. The specification provides guidelines for how devices should communicate with gateways, how messages should be formatted, and how security should be implemented. As illustrated in Fig. 1, the main components of the LoRaWAN architecture are the end-devices (EDs), the gateways, and the Network Server (NS). The communication between the EDs and the gateways is LoRaWAN-based, while the communication between the gateways and the NS is typically done over a high data rate network (e.g., Wi-Fi, LTE, the Internet).

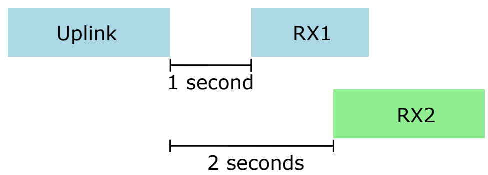


Figure 2: Default downlink scheme after an uplink transmission in LoRaWAN Class A.

EDs in LoRaWAN are divided into three classes. Class A devices are the most common ones and have the lowest power consumption. As depicted in Fig. 2, they have scheduled receive windows where they can receive downlink messages (Acknowledgments or commands) from the network after transmitting an uplink message. Uplink transmissions are performed over a random channel (from the list of available ones). After an uplink transmission, an ED opens a first receive window (RX1) – typically after 1 second – with the same uplink SF and channel. If it does not receive any data, it opens a second receive window

(RX2) – typically after 2 seconds from the uplink transmission – using a fixed SF and channel. The list of available uplink channels, the RX2 SF, and the RX2 channel are communicated to the ED during its registration to the network (Over The Air Activation – OTAA) or are hard-coded to the ED's memory (Activation By Personalization – ABP).

Acknowledgments (acks) in LoRaWAN may be sent in RX1 or RX2 once the ED has set the corresponding flag *on* in the LoRaWAN header of the uplink packet. The NS will examine the radio duty cycle availability of the gateways and will find the most appropriate one (if any) to send the ack in one of the two receive windows. If the ED has not set this flag on, it will still have to open the two receive windows because network commands may be sent by the NS through one of the gateways. If an ack is expected by an ED, but the ack is lost or not transmitted because no gateway was available, the ED will retransmit the same uplink until the maximum retransmission attempts are reached.

Class B devices have additional receive windows that are synchronized with the network's schedule, allowing for more frequent downlink communication. We do not deal with Class B devices in this paper. Class C devices have continuous receive windows, allowing for nearly constant communication but with higher power consumption. Actuators usually belong to this category.

2.3. Examined Scenario

The examined scenario consists of resource constrained Class A EDs whose energy consumption is important, as well as of actuators that are considered as Class C devices without specific energy constraints but with delay constraints. The actuators can receive downlink packets from the NS (through a gateway) to perform a task (e.g., switch on a pump). Actuators are always on devices so that they can receive critical commands and minimize the response time to these commands (minimize delay). EDs and actuators co-exist in the same LoRaWAN network, and thus, they share the same or similar resources.

In the examined scenario, actuating requests are generated at random times at the NS and downlink messages have to be sent to a specific actuator every time. The NS selects a gateway to serve each of those requests. Once a request is received by the actuator, an ack is sent back to the NS.

The main objective and research question of this paper is to propose and assess mechanisms that maximize the probability of delivery of downlink commands from the NS to the actuators, thus, increase the downlink packet reception ratio to actuators. To this context, a number of channel allocation strategies are presented and evaluated by means of computer simulations. Other objectives constitute the minimization of the response time once an actuating request is generated (delay), the delivery ratio of the actuators' acks, the packet delivery ratio of EDs, the energy consumption of EDs, and the actuating fairness.

3. Related Research

The effect of downlink traffic on the network performance has been investigated by several studies in the literature, as mentioned in the introduction. Even though the effect of critical traffic has been explored in some studies [17, 18], all of them consider only critical uplink traffic. The question of how urgent/critical downlink data can be sent to devices has remained unexplored. Nevertheless, this section goes through recent studies related to techniques and approaches to support critical data transmissions over LoRaWAN through resource allocation or scheduling methods. The advantages, disadvantages as well as research gaps are highlighted.

Many works deal with the problem of efficiently allocating resources to EDs by tuning their transmission parameters in order to decrease collisions and improve efficiency [19, 20]. Urgent traffic can be prioritized by adjusting certain settings such as the transmission power [19] or by allocating priority transmissions to low traffic channels [17]. Adapting the number of retransmissions in a congested network is also an option to alleviate additional traffic and improve efficiency for urgent transmissions [20].

A key-point to improve the efficiency of prioritized traffic is to increase the probability of getting this traffic delivered at the gateways. In one of the recent works, Carvalho et al. [21] propose an approach that uses three successive uplink transmissions to increase the probability of packet delivery. Each successive transmission is done over a higher SF to avoid collisions in highly congested SFs and eventual losses due to the path-loss. Even though valid, the approach exhibits some strong negatives. First, increasing the number of uplinks, the total load in the network gets higher which will quickly lead to the saturation of the available channels. Second, the performance of the approach is heavily depended on the distribution of the EDs; in deployments with high average SFs, only a few higher SFs are available for most of the EDs. Third, the default receive window delay intervals must be adjusted for most of the EDs to accommodate the successive transmissions. Fourth, more downlink traffic is generated to acknowledge replicated uplinks. Finally, the approach – in the best case scenario – doubles the energy consumption of the EDs.

Elbsir et al. [22] propose a solution based on the LoRaWAN Class B mode. The authors schedule downlinks to be received by synchronized receivers, such as actuators, in the first or second receive window. A similar approach is proposed by Todoli et al. [23]. The advantage of employing the Class B mode is that the receivers can be in sleep mode between successive beacons. The biggest disadvantage is that the synchronization beacons are performed over Class A channels, thus, the probability of collisions may be high even in low congestion networks. Moreover, the Class B mode of LoRaWAN is usually used for limited time periods because beacons reserve duty cycle resources that are critical for other downlink operations. As a consequence, a fewer downlink can be dedicated to actuating downlinks.

Regarding non-LoRaWAN solutions, scheduling of transmissions is a way of overcoming collisions caused by the Aloha-based MAC of LoRaWAN. To this end, several scheduling approaches have been proposed in the literature to support critical traffic. The works of [24] and [25] are two typical examples where the transmissions are scheduled in unique or shared slots. Even though these solutions alleviate or eliminate collisions, they are not LoRaWAN-compliant, or they inherit an extra synchronization overhead. Partially LoRaWAN-compliant approaches are proposed in [26] and in [27], however, the first one supports only uplink data while the second one supports only direct communication from an event-triggered sensor to an actuator in order to quickly disseminate an alert.

4. Actuating Downlink Solutions

This section presents a number of solutions to support actuating data over LoRaWAN. The solutions differ in how downlink channels can be utilized, and they are listed starting from the most naive one to the most sophisticated one. Naivety does not necessarily translate to a bad performance, but rather to a trade-off between performance compromises and implementation simplicity and adaptability to the existing LoRaWAN infrastructure.

All the presented schemes are framed around the EU868 sub-GHz spectrum, the corresponding regional regulations imposed in that spectrum, and mainly the TTN's[1] proposed frequency plans for that spectrum (see Table 1). However, the overall concept of prioritization described later in the text can be adapted to any other regional frequency plan as well.

Recent EU regulations on spectral usage have opened up four channels for LoRaWAN gateways, allowing for up to 500 mW ERP and a 10% radio duty cycle to be shared among all four channels [28]. This includes the addition of a dedicated band (i.e., Band 47b) with a 10% duty cycle for downlinks, which significantly enhances the downlink performance of the protocol. This means that the downlink time in RX2 can effectively double because the total duty cycle time is 20%. Moreover, since the band can be divided into four channels, redundancy increases when multiple gateways are present, as up to four gateways in the same area can simultaneously send downlink data without their transmissions colliding. Band 47b is used by network providers together with the existing Band 54 to enhance downlink capabilities and avoid interference with other neighboring networks.

4.1. Typical LoRaWAN

In typical LoRaWAN, we assume 8 uplink channels spanning 2 bands and one downlink channel for RX2, as it is depicted in Table 1. Actuating data can be sent over the shared RX2 channel because it is the one with the highest availability (i.e., 10% duty cycle). Acknowledgments from actuators to the NS can be performed over the 8 uplink channels.

Table 1: TTN uplink (U) and downlink (D) LoRaWAN bands and channels for EU868. (St. Fr. = Starting Frequency) [12]

Band	Usage	Downlink Window	St. Fr. (MHz)	BW (kHz)	Duty Cycle	SF
48	U/D	RX1	868.0	125	$\leq 1\%$	7-12
	U/D	RX1	868.2	125		7-12
	U/D	RX1	868.4	125		7-12
47	U/D	RX1	867.0	125	$\leq 1\%$	7-12
	U/D	RX1	867.2	125		7-12
	U/D	RX1	867.4	125		7-12
	U/D	RX1	867.6	125		7-12
	U/D	RX1	867.8	125		7-12
54	D	RX2	869.525	125	$\leq 10\%$	9

Due to the half-duplex nature of gateways and the presence of the shared channel in RX2, a gateway may be occupied transmitting an ack or receiving an uplink when an actuating request is generated. This means that there is a certain blocking probability which can be modeled as described in the following paragraphs. The process of modeling the problem assumes that two or more LoRa signals interfere if they overlap in SF, time, and frequency [13]. The capture effect is not taken into account, thus, the analysis represents the worst case scenario.

In the following analysis, we formulate the blocking probability by calculating the likelihood of one arbitrary actuating downlink packet finds all gateways busy, so it cannot be served. An other way to calculate this probability is to calculate the number of blocked packets among m transmitted downlink packets and average it, producing the complementary of the packet reception ratio (i.e., $1 - PRR$). However, the second option requires complex math analysis as the events of different downlink packets being blocked are not independent of each other. Nevertheless, the two approaches are in essence similar and both of them describe the "busy time" of the network, which is our objective.

Let us consider a period of time t, where t is much longer than a duty cycle periods (or rounds)[2]. Let us also consider a large number of EDs, n, with their transmission attempts following a Poisson distribution characterized by an intensity λ_n per duty cycle round. This stochastic behavior mirrors the random nature of traffic in communication networks. Similarly, let us assume k actuators with a request arrival rate also modeled as a Poisson process with average rate λ_k. Moreover, let g be the number of gateways. Let us denote with t_{ul}, t_1, and t_2 the transmission

[1]The Things Network: the largest public LoRaWAN network in Europe (`https://www.thethingsnetwork.org`).

[2]We assume that the duty cycle resources are evenly divided in rounds; that is, if t_x is the transmission time, then the gateway must wait $w \cdot t_x = (\frac{1}{\delta} - 1)t_x$ amount of time to transmit again, where δ is the duty cycle (e.g., 1%). This is because $\frac{t_x}{t_x + w \cdot t_x} = \delta$.

length of uplink packets, acknowledgment packets in RX1, and acknowledgment packets in RX2, respectively. Moreover, t_{dl} represents the transmission length of actuating packets.

For simplicity, it is assumed that the percentage of acks transmitted in RX1 or RX2 is proportional to the capacity of the channels used in these windows. Thus, the percentage of acks transmitted in RX1 is $\frac{c_1}{c_1+c_2}$, while the percentage of acks redirected to RX2 is $\frac{c_2}{c_1+c_2}$, where c_1 and c_2 are the capacities in RX1 and RX2, respectively [12]. $c_1 = 2\lceil \frac{t \cdot \delta_1}{t_1} \rceil$ depends on the average uplink SF in the network (affecting t_1), the number of available bands (i.e. 2 in our case – see Table 1) as well as the duty cycle of these bands (δ_1). $c_2 = \lceil \frac{t \cdot \delta_2}{t_2} \rceil$ depends on the RX2 SF (affecting t_2) and the duty cycle of that channel (δ_2).

The probability of a actuating downlink packet being blocked, $P_{\mathcal{B}}$, represents the likelihood that an arbitrary downlink packet finds all available gateways busy and being postponed. This may happen when the gateways are busy receiving uplink packets, or transmitting acknowledgment packets, or transmitting other downlink packets, or their duty cycle resources have been diminished.

Since a downlink packet could be handled in either RX1 or RX2, $P_{\mathcal{B}}$ is calculated for both scenarios. Importantly, the blocking events in RX1 and RX2 are considered independent of each other. Therefore, the overall blocking probability is determined by combining the probabilities from both windows. For mathematical modelling purposes, the uplinks are divided in those that will be acked in RX1 and those that will be acked in RX2.

Given the independence of the blocking events in RX1 and RX2, $P_{\mathcal{B}}$ can be computed using the formula for the union of two independent events:

$$P_{\mathcal{B}} = P_{\mathcal{B}}^{\mathrm{RX2}} \vee P_{\mathcal{B}}^{\mathrm{RX1}} = P_{\mathcal{B}}^{\mathrm{RX2}} + P_{\mathcal{B}}^{\mathrm{RX1}} - P_{\mathcal{B}}^{\mathrm{RX2}} \cdot P_{\mathcal{B}}^{\mathrm{RX1}}. \quad (1)$$

For RX1, the blocking probability $P_{\mathcal{B}}^{\mathrm{RX1}}$ can be modeled as:

$$\begin{aligned} P_{\mathcal{B}}^{\mathrm{RX1}} &= P_{\mathcal{B}}^{up} \vee P_{\mathcal{B}}^{RX1ack} \\ &= P_{\mathcal{B}}^{up} + P_{\mathcal{B}}^{RX1ack} - P_{\mathcal{B}}^{RX1ack} \cdot P_{\mathcal{B}}^{up}, \end{aligned} \quad (2)$$

where $P_{\mathcal{B}}^{up}$ represents the probability that a gateway is busy processing uplink packets and $P_{\mathcal{B}}^{RX1ack}$ is the probability that a gateway is busy transmitting acks for uplink packets acknowledged in RX1. We assume that $P_{\mathcal{B}}^{up}$ and $P_{\mathcal{B}}^{RX1ack}$ are independent events, taking into consideration that a gateway can receive a downlink packet or another uplink after an uplink.

Similarly, for RX2, the blocking probability $P_{\mathcal{B}}^{\mathrm{RX2}}$ can be modeled as:

$$P_{\mathcal{B}}^{\mathrm{RX2}} = P_{\mathcal{B}}^{RX2ack} \vee P_{\mathcal{B}}^{act} = P_{\mathcal{B}}^{RX2ack} + P_{\mathcal{B}}^{act} - P_{\mathcal{B}}^{RX2ack} \cdot P_{\mathcal{B}}^{act}, \quad (3)$$

where $P_{\mathcal{B}}^{RX2ack}$ represents the probability that a gateway is busy transmitting acks for uplink packets acknowledged in RX2, and $P_{\mathcal{B}}^{act}$ represents the probability that a gateway is busy transmitting another actuating downlink packet. Both of these events are also independent.

According to Table 1, we have multiple bands and channels per band in RX1 and one channel in RX2. Therefore, to calculate the probabilities $P_{\mathcal{B}}^{up}$, $P_{\mathcal{B}}^{RX1ack}$, $P_{\mathcal{B}}^{RX2ack}$, and $P_{\mathcal{B}}^{act}$, we must first determine the probability of a downlink packet being non-blocked for a certain channel i, denoted as $\widehat{P}_i$. The following general formula can be used for all four probabilities:

$$P_{\Omega_s} = 1 - \bigwedge_{i=1}^{|\Phi|} \widehat{P}_i = 1 - \prod_{i=1}^{|\Phi|} \widehat{P}_i, \quad (4)$$

where Ω_s is a set of parameters describing scenario s (i.e., uplink, ack in RX1 or RX2, or actuating downlink) and contains l, Φ, ν, λ, g. l is the length of the packet transmission specific to the type of packets being considered (such as uplinks or acks). Φ is a set containing all individual duty cycles per available channel, assuming that the load is distributed evenly among the channels. It holds that $\sum_{i=1}^{|\Phi^1|} \Phi_i^1 = \delta_1$ and $\sum_{i=1}^{|\Phi^2|} \Phi_i^2 = \delta_2$, where Φ^1 and Φ^2 are the sets of available channels in RX1 and RX2, respectively. ν represents the total number of devices, which could be either actuating devices or EDs. The intensity of transmissions from these ν devices is represented by λ_ν, modeled using the Poisson distribution. $\widehat{P}_i$ expresses the probability of not finding channel i blocked, and it can be written as follows:

$$\widehat{P}_i = \sum_{j=1}^{\infty} \left(\frac{t - l \cdot \frac{100}{i}}{t} \right)^{\frac{j}{|\Phi| \cdot g}} \cdot \frac{(\nu \lambda_\nu)^j}{e^{\nu \lambda_\nu} j!}, \ i \in \Phi. \quad (5)$$

In Eq. (5), $\frac{j}{|\Phi| \cdot g}$ reflects the distribution of packets across the available channels and gateways. For acks, this term is further modified to account for the distribution of acks in RX1 and RX2, using $\frac{j \cdot \frac{c_1}{c_1+c_2}}{|\Phi| \cdot g}$ for RX1 and $\frac{j \cdot \frac{c_2}{c_1+c_2}}{|\Phi| \cdot g}$ for RX2.

Table 2: Blocking probability parameters per scenario.

Scenario (s)	Ω_s in Eq. (4)
up	$\Omega_{up} = \{t_u, \{100\}^*, n, \lambda_n, g\}$
$RX1ack$	$\Omega_{RX1ack} = \{t_1, \{3 \times \frac{1}{3}, 5 \times \frac{1}{5}\}, n, \lambda_n, g\}$
$RX2ack$	$\Omega_{RX2ack} = \{t_2, \{10\}, n, \lambda_n, g\}$
act	$\Omega_{act} = \{t_{dl}, \{10\}, k, \lambda_k, g\}$

*This means that there is no waiting time due to duty cycle after a gateway receives an uplink

Using Eq. (4, 5) as well as Table 2, we derive the following blocking probabilities for typical LoRaWAN:

$$P_{\mathcal{B}}^{up} = 1 - \sum_{j=1}^{\infty} \left(\frac{t - t_u}{t} \right)^{\frac{j}{g}} \cdot \frac{(n \lambda_n)^j}{e^{n \lambda_n} j!}, \quad (6)$$

$$P_{\mathcal{B}}^{RX1ack} = 1 - \left(\sum_{j=1}^{\infty} \left(\frac{t - 300t_1}{t} \right)^{\frac{j \frac{c1}{c1+c2}}{8g}} \cdot \frac{(n\lambda_n)^j}{e^{n\lambda_n} j!} \right)^3 \cdot \left(\sum_{j=1}^{\infty} \left(\frac{t - 500t_1}{t} \right)^{\frac{j \frac{c1}{c1+c2}}{8g}} \cdot \frac{(n\lambda_n)^j}{e^{n\lambda_n} j!} \right)^5 , \tag{7}$$

$$P_{\mathcal{B}}^{RX2ack} = 1 - \sum_{j=1}^{\infty} \left(\frac{t - 10t_2}{t} \right)^{\frac{j \frac{c2}{c1+c2}}{g}} \cdot \frac{(n\lambda_n)^j}{e^{n\lambda_n} j!}, \tag{8}$$

$$P_{\mathcal{B}}^{act} = 1 - \sum_{j=1}^{\infty} \left(\frac{t - 10t_{dl}}{t} \right)^{\frac{j}{g}} \cdot \frac{(k\lambda_k)^j}{e^{k\lambda_k} j!}. \tag{9}$$

Moreover, other downlink transmissions from neighboring gateways may be happening during a downlink transmission on the same channel. The collision probability $P_{\mathcal{C}}$ captures this risk, including interference with acks in the RX2 channel, which is the only shared channel between actuating downlink traffic and ED acknowledgements. It is calculated by considering the likelihood that an actuating downlink packet collides with another packet (either an ack or actuating) when both are transmitted in the same channel. The formula for $P_{\mathcal{C}}$ is as follows:

$$\begin{aligned} P_{\mathcal{C}} &= P_{\mathcal{C}}^{act} \vee P_{\mathcal{C}}^{RX2ack} \\ &= P_{\mathcal{C}}^{act} + P_{\mathcal{C}}^{RX2ack} - P_{\mathcal{C}}^{RX2ack} \cdot P_{\mathcal{C}}^{act}, \end{aligned} \tag{10}$$

where $P_{\mathcal{C}}^{RX2ack}$ accounts for the scenario where uplinks are acknowledged in RX2. It is calculated with the assumption that $\frac{c_2}{c_1+c_2}$ portion of uplinks are acknowledged in RX2. In this case, $\Omega_{RX2ack} = \{t_2, \{10\}, k, \lambda_k, g\}$. Furthermore, $P_{\mathcal{C}}^{act}$ represents the probability of an actuating downlink being transmitted at the same time, with $\Omega_{act} = \{t_{dl}, \{10\}, k, \lambda_k, g\}$.

$$P_{\mathcal{C}}^{RX2ack} = 1 - \sum_{j=1}^{\infty} \left(\frac{t - 10t_2}{t} \right)^{\frac{j \frac{c2}{c1+c2}}{g}} \cdot \frac{(n\lambda_n)^j}{e^{n\lambda_n} j!}, \tag{11}$$

$$P_{\mathcal{C}}^{act} = 1 - \sum_{j=1}^{\infty} \left(\frac{t - 10t_{dl}}{t} \right)^{\frac{j}{g}} \cdot \frac{(k\lambda_k)^j}{e^{k\lambda_k} j!}. \tag{12}$$

Advantages: The scheme is very easy to be implemented and operate over any existing LoRaWAN infrastructure.
Disadvantages: The scheme uses a shared downlink channel for actuating data with typical LoRaWAN downlink data (acks and commands). Thus, the performance dramatically degrades with high number of uplinks and actuators.

4.2. Typical LoRaWAN with Controlled Downlinks

To eliminate the risk of collisions between actuating data and other downlink transmissions, the downlink activity can be controlled by the NS. This means that every time some new downlink data has to be transmitted by a gateway to an actuator, all other pending or future downlink transmissions in the neighborhood that may overlap with the actuation downlink are canceled or postponed, including other actuating data transmissions. Since actuating data is sent over the RX2 channel, RX1 traffic remains untouched.

Considering the NS's control over downlink activities the probability of having a collision with other downlink packets is zero (i.e., $P_{\mathcal{C}} = 0$), reflecting the effective management of downlink transmissions by the NS to avoid any overlap, thereby completely negating the chance of collision between actuating data and other downlink activities. In practice, the behavior may be slightly different due to delays in delivering commands from the NS to the gateways, especially when this is done over the Internet. However, these special cases are not taken into account in this paper.
Advantages: The controlled downlink scheme eliminates collisions because only one transmission is performed at a time.
Disadvantages: Some actuating downlink transmissions may be postponed, leading to delays. Moreover, acknowledgment transmission may not be delivered, resulting in retransmissions and higher energy consumption for some EDs. The scheme also requires extra programming effort at the NS to control downlink transmissions and perform mutual exclusion.

4.3. LoRaWAct: actuating over additional downlink channels

LoRaWAct makes use of additional downlink channels in the EU868 spectrum dedicated only to actuating data, separating ack transmissions to EDs from actuating data. More specifically, 4 additional channels (Band 47b) with 10% total duty cycle time can be shared to send data to actuators. The number of actuators assigned to each of those 4 channels can be decided according to their actuating traffic (i.e., how frequently the NS sends data to each of them). Acks from actuators to the NS are performed over the typical shared uplink channels.

With the introduction of these dedicated actuating channels, $P_{\mathcal{B}}$ remains unchanged from its original formulation in Eq. (1). However, we introduce an adjustment to $P_{\mathcal{B}}^{RX2ack}$ and $P_{\mathcal{B}}^{act}$ to reflect the distribution of actuating traffic across the four new channels. It assumed that the available actuators are equally divided into the 4 channels of Band 47b, thus $\Omega_{RX2ack} = \{t_2, \{\frac{10}{4}, \frac{10}{4}, \frac{10}{4}, \frac{10}{4}\}, n, \lambda_n, g\}$.

The blocking probabilities can be written as follows:

$$P_{\mathcal{B}}^{RX2ack} = 1 - \left(\sum_{j=1}^{\infty} \left(\frac{t - 40t_2}{t} \right)^{\frac{j \frac{c2}{c1+c2}}{4g}} \cdot \frac{(n\lambda_n)^j}{e^{n\lambda_n} j!} \right)^4 , \tag{13}$$

$$P_{\mathcal{B}}^{act} = 1 - \left(\sum_{j=1}^{\infty} \left(\frac{t - 40t_{dl}}{t} \right)^{\frac{j}{4g}} \cdot \frac{(k\lambda_k)^j}{e^{k\lambda_k} j!} \right)^4 . \tag{14}$$

With the elimination of $P_{\mathcal{B}}^{RX2ack}$ due to the new downlink channel structure, $P_{\mathcal{C}}$ is now equal to $P_{\mathcal{C}}^{act}$.

$$P_{\mathcal{C}} = P_{\mathcal{C}}^{act} = \left(1 - \sum_{j=1}^{\infty} \left(\frac{t - 40t_{dl}}{t} \right)^{\frac{j}{4g}} \cdot \frac{(k\lambda_k)^j}{e^{k\lambda_k} j!} \right)^4 . \tag{15}$$

Advantages: The scheme separates the actuating traffic to other downlink traffic, providing additional capacity dedicated to actuators. As shown in the analysis below, the probability of collisions is drastically reduced. The scheme also gives more room to typical LoRaWAN data transmissions.
Disadvantages: The frequency plan needs to be modified and include additional channels for the actuators. Moreover, those additional channels may not be available in all regions. Acks from actuators still use the shared uplink channels, even though CSMA techniques can be used to alleviate collisions [16, 29].

4.4. LoRaWAct with Controlled Downlinks

Similar to the second scheme, LoRaWAct can be combined with the controlled downlink scheme. In this case, the manipulation of the downlink transmission time does not affect the rest of the network but only other actuating data transmissions.

In this configuration, the calculations for $P_{\mathcal{B}}$ and $P_{\mathcal{C}}$ follow the structure outlined in Section 4.3 and 4.2, respectively.
Advantages: The scheme inherits all the advantages of LoRaWAct enhanced with additional protection from collisions due to the presence of parallel actuation downlinks.
Disadvantages: The disadvantages of LoRaWAct are disadvantages of this scheme as well. An additional drawback is that some actuating transmissions may be postponed.

4.5. Other schemes & Non-working schemes

Combinations of the already proposed schemes could be applied to support critical downlink traffic. For example, someone could use uplink channels as downlink for actuating purposes, even though this would theoretically lead to a worse performance. CSMA approaches as the one recently suggested by the LoRa Alliance [16] could also be applied on top of the proposed ones, especially for actuators' acks.

Moreover, someone could dedicate one or two channels from Band 47b to uplink actuator transmissions and the rest of the channels to actuating downlink data. Nevertheless, as LoRaWAN specifications dictate, three basic channels have to be used for uplinks by the EDs, thus, the use of shared resources between actuators and EDs is unavoidable. Moreover, the use of channels from Band 47b for uplinks automatically implies the use of fewer channels from other bands because the maximum number of uplink channels cannot be more than 8.

4.6. Coexistence with other networks

When multiple operators coexist, there is a chance of external interference. Since the downlinks channels are most likely the same, this interference can be considered as additional internal traffic for a single network. In this case, the equations are still valid as only λ_n and λ_k change. The effect of external interference that is not captured by the equations is the one on the controlled-downlink schemes. In that case, we assume that there is no interference in the neighborhood, because we allow only one downlink transmission at a time. If multiple networks coexist, the collision probability is not zero, but it depends on the presence of other actuators/EDs belonging to other networks in the area. In that case, the equations presented for the non-controlled-downlink version can be used for the controlled-downlink schemes, assuming that the average packet arrival rates of actuating data and ED data are known.

4.7. Theoretical findings

This subsection presents theoretical findings for the four proposed approaches based on the equations presented in the previous paragraphs. We first present findings for the blocking and collision probabilities using typical values for the corresponding network parameters, and then we model the theoretical maximum delay that can be caused due to the blocking probability per scheme. The four schemes are referred to as "LoRaWAN", "LoRaWAN Cntr-DL", "LoRaWAct", and "LoRaWAct Cntr-DL", respectively.

4.7.1. Blocking & collision probabilities

The results for the blocking and collision probabilities are illustrated in Figure 3. The upper part of the figure presents the blocking probabilities when different network parameters vary. It can be observed that the probability of a new actuating request finds the gateways busy is high even with low number of EDs, actuators, and low packet arrival rate. This finding indicates the increased probability of delaying an actuating downlink, especially under the presence of a single shared downlink channel, and it is further analyzed in Section 4.7.2.

Moreover, as the lower part of the figure reveals, the collision probability in typical LoRaWAN is high even with

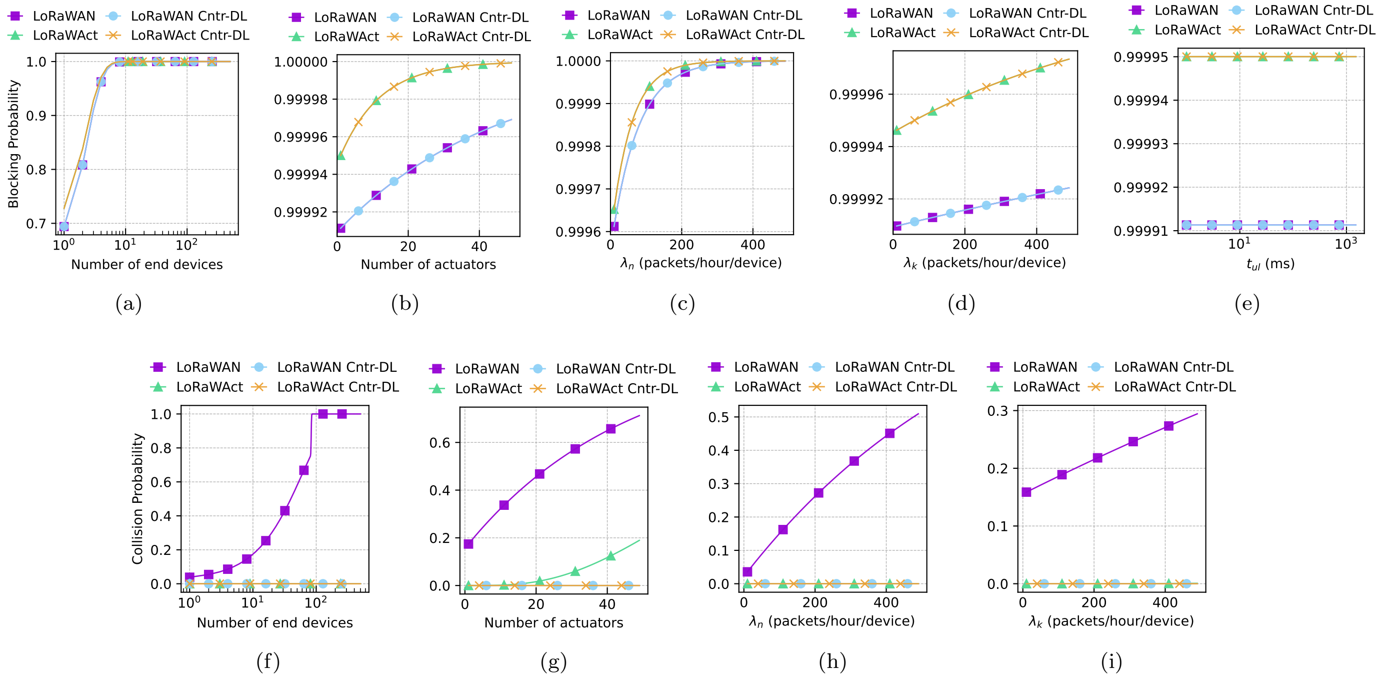
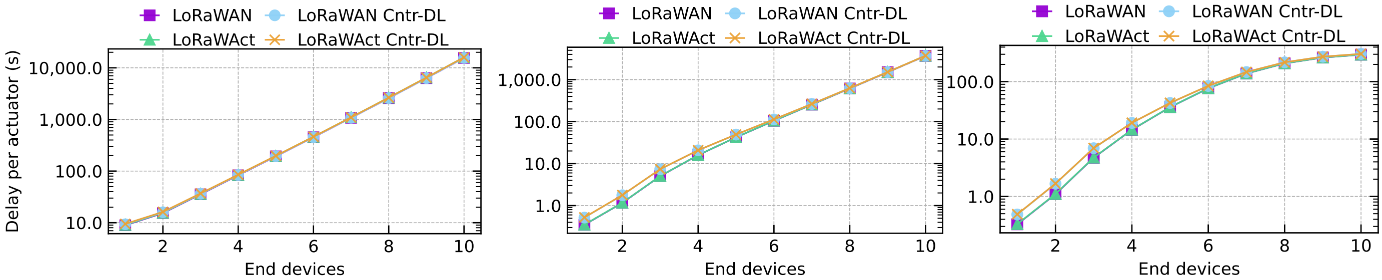


Figure 3: Blocking (a-e) and collision (f-i) probabilities for different parameters for the four approaches. (Typical traffic values are considered: t = 3600s, c_1 = 436 (avg RX1 SF=9) [12], c_2 = 272 (RX2 SF=12) [12], n = 10 (unless it varies), k = 1 (unless it varies), g = 1, λ_n = 12 (unless it varies), λ_k = 6 (unless it varies), t_1 = 0.165s (avg SF=9), t_2 = 1.319s (SF12 ack), t_{ul} = 0.226s (avg SF=9 with 16 bytes payload + LoRaWAN overhead), t_{dl} = 1.319s (SF12)).



(a) 2 gateways (b) 4 gateways (c) 8 gateways

Figure 4: Theoretical delay per actuator for different numbers of gateways. The parameters are the same as in Fig. 3.

low number of EDs or actuators. As it was expected, the packet arrival rate as well as the number of EDs (i.e., n) also play an important role in the network performance. We can also observe that most of these parameters have the same effect on the blocking or collision probabilities. This happens because the performance actually depends on the load in the network, where the load is a function of n, k, λ_n, and λ_k.

4.7.2. Actuating delay

As we showed in Fig. 3, the blocking probability is high for all schemes. The controlled downlink scheme decreases slightly the blocking probability, but as parallel transmissions on the same channel are not allowed, the delay may be substantially high as subsequent actuating requests may be queued at the NS. Requests that are placed in the back of the queue will need much time to be served because of the long transmission time of actuating data (i.e., approximately 1.32s excluding the LoRaWAN overhead) and the waiting time between successive transmissions due to the duty cycle requirement, as it explained below.

The delay experienced by an actuator due to a gateway being blocked can be modeled by considering the probability that a gateway remains blocked over multiple duty cycle rounds. Specifically, a gateway can be blocked in r consecutive duty cycle rounds if, during each round, it receives at least one actuating request (denoted as A). As a result, the actuator incurs a total delay of $10 \cdot r \cdot t_{\rm dl}$, assuming a typical RX2 duty cycle of 10%. In the case of multiple gateways (denoted by g), the total delay is adjusted to $(g \cdot r + \max(10 - g, 0) \cdot r + m) \cdot t_{\mathrm{dl}}$, where m represents the number of gateways that remain blocked in the $(r+1)^{\mathrm{th}}$ round. As, the actuator must wait an additional $m \cdot t_{dl}$ time.

The average delay $\overline{D}$ can then be expressed as the sum over an infinite series, as follows:

$$\overline{D} = \sum_{r=1}^{\infty} \sum_{m=0}^{g-1} P(A) \cdot (g \cdot r + \max(10-g,0) \cdot r + m) \cdot t_{dl} \quad (16)$$

where $P(A)$ represents the probability that the gateway is blocked for r rounds. For the controlled versions of LoRaWAN and LoRaWAct approaches, $P(A)$ is modeled as $P^{g \cdot r} \cdot \binom{g}{m} \cdot P^m \cdot (1-P)^{g-m}$, reflecting the scenario where all g gateways are blocked across r consecutive rounds, causing a processing delay, and with exactly m gateways being blocked in the $(r+1)^{\mathrm{th}}$ round (because only one downlink is allowed at a time). In contrast, for the LoRaWAN and LoRaWAct approaches, $P(A)$ is given by $P^{g \cdot r} \cdot (1 - P^g)$, representing the likelihood that all g gateways are blocked across r consecutive rounds, with at least one gateway becoming available in the $(r+1)^{\mathrm{th}}$ round. Additionally, because parallel transmissions are allowed, an actuator will never need to wait for other transmissions to be completed, making the consideration of m unnecessary.

As we observe from Figure 4, the delay increases exponentially for up to 9 EDs, while for more EDs, it becomes infinite because the blocking probability is 1. This happens because the equations take into account the worse case scenario which implies that the service rate (number of downlinks transmitted per hour) is lower than request arrival rate (i.e., λ_k). We can also observe that the controlled-downlink schemes exhibit muliple times higher delay than the non-controlled ones, which is more visible in figure (c) due to the lower range of the Y axis.

5. Simulations & Discussion of the Results

In this section, the proposed schemes are evaluated by means of simulations. For this purpose, we utilized the LoRaWAN-SIM simulator (v2024.1.14-EU868)[3], a tool that has been recently employed in several research papers. The simulator implements a terrain generator with random (uniform or not) ED and gateway placement, a path-loss model with shadowing and Rayleigh effects, intra- and inter-SF collisions, capture effect, multiple uplink and downlink channels, two receive windows for acknowledgments and network commands, the ADR mechanism, LoRaWAN header overhead, and radio duty cycle per band. Generation of actuating data and support for Band 47b were implemented in the newer version for the needs of this research.

Table 3: Simulation Parameters

Parameter	Value
Simulation time	100,000s (1+ day)
End-Devices (EDs)	100 – 500
Actuators	$\frac{\#\text{ of EDs}}{10}$
Gateways	2 / 4 / 8
Terrain side	2500m
EDs/GWs position	Random
Spreading Factors	7 – 12
Channel bandwidth	125 KHz
Preamble symbols	8
Coding Rate	4/5
SFs for RX 1/2	SF7–12 / SF12
Uplink/Downlink channels	8 / 8+1 (TTN EU868)
ED Payload size	Randomly selected [SF7-8:222, SF9:115, SF10-12:51] Bytes [30] (overall average ≈ 40 Bytes)
Actuator payload size	1 Byte (rounded to 16 Bytes)
Path loss model	$\overline{L_{pl}}(d_0) = 110$dBm, $d_0 = 40$m, $\gamma = 2.08$, $\sigma_{dBm} = 3.57$
Receiver sensitivities	Typical Semtech SX1276
Tx power	2, 7, 14 dBm (ADR adjustable)
Max current consumption (Tx, Rx, Idle, Sleep)	75, 45, 30, 0 mA
Voltage	3.3 V (average)
ED uplink rate	1 pkt every 5 min (average)
Actuating requests rate	1 request every 10 min (average)
ED retransmissions	1

5.1. Simulation setup

The four actuating approaches were implemented as they are described in Section 4. The Packet Reception Ratio (PRR) of the actuators, the Packet Delivery Ratio (PDR) of the actuators, the PDR of the EDs, the energy consumption of the EDs, the total actuating delay, and the actuating fairness are reported. PRR is defined as the ratio between the total received packets by the NS (or actuators) and the total uniquely transmitted packets by EDs (or the NS). PDR is defined as the ratio between the total acknowledged packets by the NS (or actuators) and the total uniquely transmitted packets by EDs (or the NS). The energy consumption is measured as a sum of transmitting, receiving, and idle energy expenditure based on experimental measurements of SX1276 transceivers [26]. The actuating delay is measured as the accumulated delay of transmitting an actuation downlink due to delays caused by the radio duty cycle constraints and the controlled-downlink mechanism (when applicable). The actuating fairness is measured as the standard deviation (stddev) of the individual PRRs among actuators. High stddev values indicate poor fairness, while values closer to zero indicate higher degrees of fairness.

In all simulations, we assess the approaches with an increasing ED density and number of gateways. We consider a ratio between actuators and EDs of 1:10. Configurations with 1 to 8 gateways were tested, but since the results of the approaches with 1 gateway are very similar due to the similar blocking probability, they are omitted in the figures. All scenarios are run for 50 times and the average

[3] https://github.com/deltazita/LoRaWAN-SIM

results are presented along with the 95% confidence intervals. Table 3 summarizes the simulation parameters and the corresponding values.

5.2. Results

Figure 5 illustrates the performance of the approaches in terms of actuators' PRR with an increasing number of actuators (as well as EDs) and different number of gateways. A high PRR indicates successful downlink delivery to actuators, which is the main objective of the approaches. The results reveal (a) the superiority of LoRaWAct and more especially of the controlled-downlink version; (b) the positive effect of additional gateways in the same approach; (c) the weakness of the conventional LoRaWAN to perform as well as LoRaWAct due to the increased collision probability, even though the controlled-downlink version exhibits an acceptable performance; and (d) comparing to the theoretical result (see Figure 3f-g), the latter is confirmed, because the controlled-dowlink schemes are not affected (or very slightly affected) by the increase of EDs and actuators, while LoRaWAN seems to be affected the most. LoRaWAct exhibits similar behavior with the theoretical trend as the collision probability increases with higher load.

Moreover, another insteresting observation can be made. LoRaWAN-based schemes are affected by the number of actuators, and more specifically, PRR may be lower with higher number of gateways. This happens because, with only two gateways deployed, the chance of having a collision (between actuating data and acks) is low, because the number of devices that can reach both gateways is also low and not many parallel transmissions can occur. As the number of gateways increases, the number of parallel transmissions for acks and actuating requests increase as well. The increase of parallel transmissions leads to a higher chance of collisions. Collisions lead to retransmissions and lower PRR. However, when the number of gateways gets very high, the distance between the gateways and the EDs becomes shorter, and thus, lower SFs are used on avearage. As lower SFs imply shorter transmission times, the number of collisions, and thus of retransmissions is low which leads to a higher PRR. Non-LoRaWAN schemes do not have this issue because they use Band 47b for actuating requests, while acks are sent over Bands 47 and 48.

Apparently, the positive effect of the controlled-downlink mechanism in PRR is transformed to a disadvantage when observing the actuating delay per actuator in Figure 6. Both controlled-downlink approaches exhibit multiple times higher delay compared to the approaches without this mechanism because only one downlink is allowed at a time in the neighborhood (for the same channel). However, the delay rapidly decreases with additional gateways. This behavior confirms the theoretical results presented in Figure 4, even though the theoretical findings correspond to the worst case scenario.

The LoRaWAct approaches perform worse in terms of actuators' PDR, as observed from the results of Figure 7. This happens because in LoRaWAct the traffic in the RX2 ED channel is higher as more uplinks are acknowledged due to the increased capacity. As a consequence, the probability of downlink collisions is higher. Since actuators do not have a critical energy consumption requirement, this problem can be solved my allowing the actuators to transmit more than one successive acknowledgment at a time. For example, this solution is employed with big success in [21] for uplink transmissions.

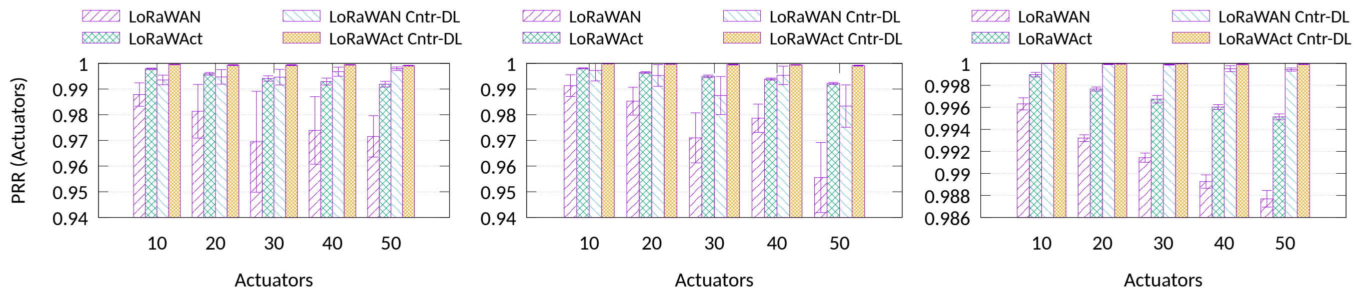
In terms of actuation fairness, as shown in Figure 8, the controlled downlink approaches exhibit the best result. LoRaWAct follows, while typical LoRaWAN exhibits the worst performance, even with 8 gateways. The result is reasonable because the downlink channel is free for all actuators when the controlled downlink scheme is applied. The opposite holds for the other approaches. To be fair, this reasonable result is also driven by the fact that the same number of downlinks is triggered for all actuators.

Despite the decreased performance of the actuators' PDR in LoRaWAct versions, the equivalent metric for the EDs is improved. Figure 9 depicts the corresponding results. It can be observed that the overall PDR is much higher in most of the scenarios because the downlink capacity in RX2 is all dedicated to EDs rather than being shared among EDs and actuators. More room for ED transmissions translates to fewer collisions and improved performance.

Finally, in terms of energy consumption of EDs, there is no considerable difference between the approaches, as shown in Figure 10. However, the LoRaWAct approaches exhibit a slightly better result due to the higher downlink capacity, which implies a decreased probability of retransmissions. LoRaWAN-Cntr-DL exhibits the worst performance because many RX2 acknowledgments are eliminated in favor of actuating downlinks. This action causes retransmissions and thus increased energy consumption.

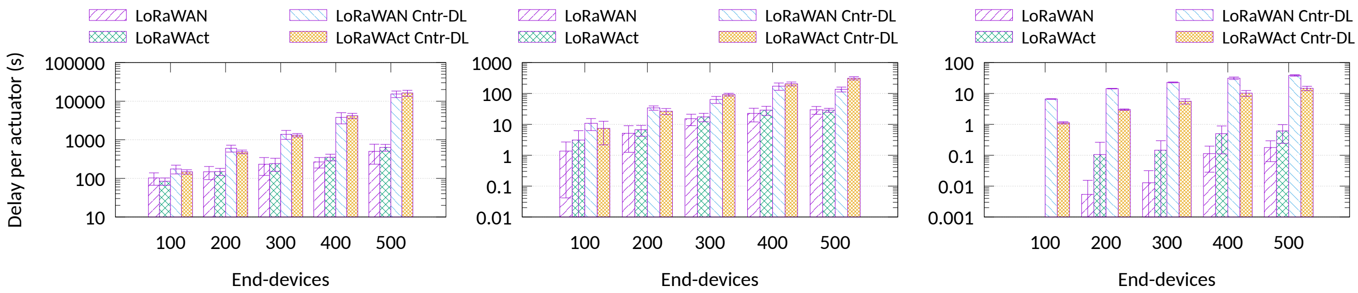
6. Conclusions & Future Work

In conclusion, the study on supporting critical downlink traffic in LoRaWAN networks sheds light on the importance of several factors that affect performance. By proposing and evaluating mechanisms to maximize the probability of delivering downlink commands to actuators, this research contributes to enhancing the reliability and responsiveness of critical data transmissions. Through the analysis of different resource allocation schemes and simulation results, valuable insights have been gained into improving downlink traffic performance, actuator acknowledgment ratios, and overall network efficiency. The findings highlight the significance of prioritizing critical downlink traffic to meet the demands of real-time downlink applications. Since this action comes with additional delay due to the high blocking probability and the controlled downlink mechanism, additional gateways are recommended



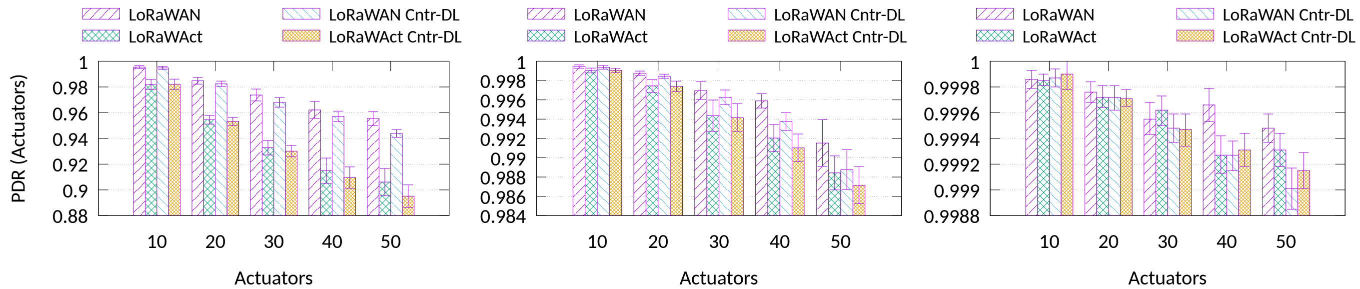
(a) 2 gateways (b) 4 gateways (c) 8 gateways

Figure 5: Packet Reception Ratio of actuators for different gateway populations.



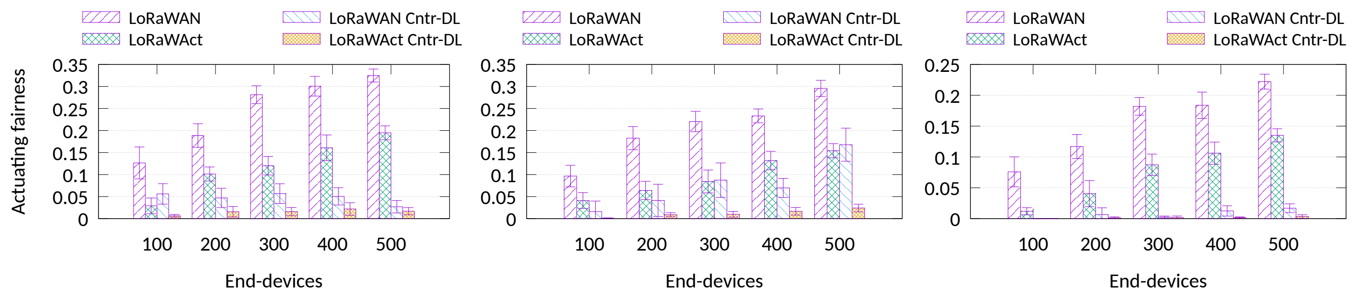
(a) 2 gateways (b) 4 gateways (c) 8 gateways

Figure 6: Average delay per actuator for different gateway populations.



(a) 2 gateways (b) 4 gateways (c) 8 gateways

Figure 7: Packet Delivery Ratio of actuators for different gateway populations.

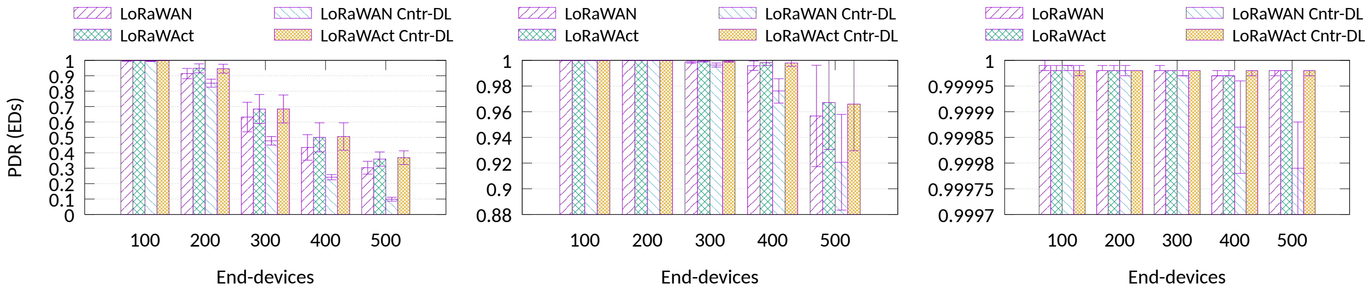


(a) 2 gateways (b) 4 gateways (c) 8 gateways

Figure 8: Fairness of actuators for different gateway populations (the lower, the better).

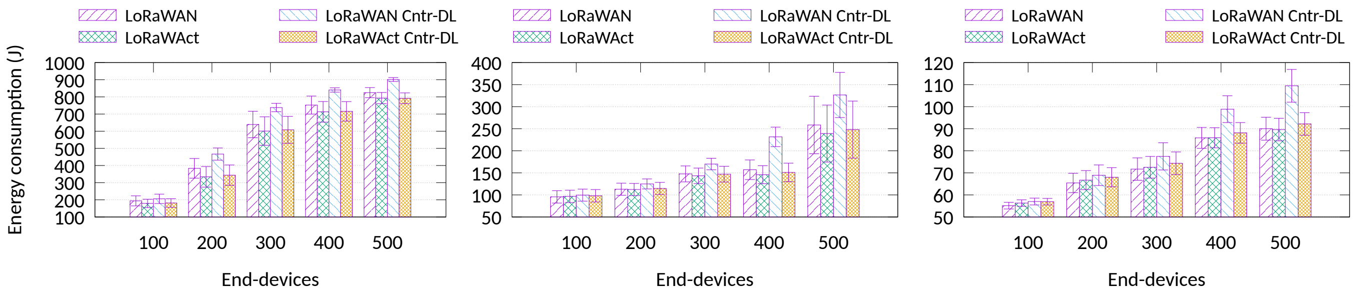
to be deployed. Nevertheless, the results show very high improvement in terms of delivery of the actuating commands, actuating fairness, and sometimes energy consumption. The use of Band 47b, whenever it is available, can significantly enhance performance by increasing the downlink capacity and separating ED to actuating downlink traffic.

Future work may focus on further refining these strate-



(a) 2 gateways (b) 4 gateways (c) 8 gateways

Figure 9: Packet Delivery Ratio of the EDs for different gateway populations.



(a) 2 gateways (b) 4 gateways (c) 8 gateways

Figure 10: Energy consumption of EDs for different gateway populations.

gies for other regional spectra, assess more configuration parameters such as the downlink SF, and explore solutions under the presence of a massive number of EDs.

Acknowledgement

This publication has emanated from research conducted with the financial support of Nazarbayev University grant No. 11022021FD2916 for the project "DELITMENT: DEterministic Long-range IoT MEsh NeTworks".

References

[1] I. Tomić, L. Bhatia, M. J. Breza, J. A. McCann, The Limits of LoRaWAN in Event-Triggered Wireless Networked Control Systems, in: UKACC 12th International Conference on Control (CONTROL), 2018, pp. 101–106.

[2] B. K. Al-Shammari, N. Al-Aboody, H. S. Al-Raweshidy, IoT traffic management and integration in the QoS supported network, IEEE Internet of Things Journal 5 (1) (2017) 352–370.

[3] A. Rayes, S. Salam, The things in IoT: Sensors and Actuators, in: Internet of Things From Hype to Reality: The Road to Digitization, Springer, 2022, pp. 63–82.

[4] A. Pötsch, F. Haslhofer, Practical limitations for deployment of LoRa gateways, in: 2017 IEEE International Workshop on Measurement and Networking (M&N), IEEE, 2017, pp. 1–6.

[5] S. Abboud, N. El Rachkidy, A. Guitton, H. Safa, Gateway selection for downlink communication in lorawan, in: 2019 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2019, pp. 1–6.

[6] A. Farhad, D.-H. Kim, J.-Y. Pyun, R-ARM: retransmission-assisted resource management in LoRaWAN for the Internet of Things, IEEE Internet of Things Journal 9 (10) (2021) 7347–7361.

[7] M. Capuzzo, D. Magrin, A. Zanella, Confirmed traffic in LoRaWAN: Pitfalls and countermeasures, in: 2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), IEEE, 2018, pp. 1–7.

[8] H. E. Elbsir, M. Kassab, S. Bhiri, M. H. Bedoui, Evaluation of LoRaWAN Class B efficiency for downlink traffic, in: 2020 16th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), IEEE, 2020, pp. 105–110.

[9] D. Ron, C.-J. Lee, K. Lee, H.-H. Choi, J.-R. Lee, Performance analysis and optimization of downlink transmission in LoRaWAN class B mode, IEEE Internet of Things Journal 7 (8) (2020) 7836–7847.

[10] J. M. Marais, A. M. Abu-Mahfouz, G. P. Hancke, A Survey on the Viability of Confirmed Traffic in a LoRaWAN, IEEE Access 8 (2020) 9296–9311.

[11] K. Mikhaylov, J. Petäjäjärvi, A. Pouttu, Effect of downlink traffic on performance of LoRaWAN LPWA networks: Empirical study, in: 2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), IEEE, 2018, pp. 1–6.

[12] D. Zorbas, Improving LoRaWAN downlink performance in the EU868 spectrum, Computer Communications 195 (2022) 303–314.

[13] M. C. Bor, U. Roedig, T. Voigt, J. M. Alonso, Do LoRa Low-Power Wide-Area Networks Scale?, in: Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '16, ACM, 2016, pp. 59–67.

[14] D. Croce, M. Gucciardo, S. Mangione, G. Santaromita, I. Tinnirello, Impact of LoRa Imperfect Orthogonality: Analysis of Link-Level Performance, IEEE Communications Letters 22 (4) (2018) 796–799.

[15] G. Zhu, C.-H. Liao, M. Suzuki, Y. Narusue, H. Morikawa, Evaluation of LoRa receiver performance under co-technology interference, in: 2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC), IEEE, 2018, pp. 1–7.

[16] Semtech Corporation, LoRaWAN CSMA Technical Recommendation TR013-1.0.0, https://resources.lora-alliance.org/technical-recommendations/tr013-1-0-0-csma, online; accessed 29-March-2024 (2023).

[17] A. Tsakmakis, A. Valkanis, G. Beletsioti, K. Kantelis, P. Nicopolitidis, G. Papadimitriou, Efficient Frequency Planning for LoRaWAN networks with urgent traffic, in: 2021 International Conference on Computer, Information and Telecommunication Systems (CITS), IEEE, 2021, pp. 1–5.

[18] A. Tsakmakis, A. Valkanis, G. A. Beletsioti, P. Nicopolitidis, G. Papadimitriou, On the Effect of Traffic Burstiness in LoRaWAN Networks' Performance, in: 2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), 2020, pp. 1–4.

[19] E. Lima, J. Moraes, H. Oliveira, E. Cerqueira, S. Zeadally, D. Rosário, Adaptive priority-aware LoRaWAN resource allocation for Internet of Things applications, Ad Hoc Networks 122 (2021) 102598.

[20] Z. Ali, K. N. Qureshi, A. S. Al-Shamayleh, A. Akhunzada, A. Raza, M. F. U. Butt, Delay optimization in LoRaWAN by employing adaptive scheduling algorithm with unsupervised learning, IEEE Access 11 (2023) 2545–2556.

[21] D. F. Carvalho, P. Ferrari, E. Sisinni, A. Flammini, Improving Redundancy in LoRaWAN for Mixed-Criticality Scenarios, IEEE Systems Journal 15 (3) (2021) 3682–3691.

[22] H. E. Elbsir, M. Kassab, S. Bhiri, M. H. Bedoui, Evaluation of LoRaWAN class B performances and its optimization for better support of actuators, Computer Communications 198 (2023) 128–139.

[23] D. Todoli-Ferrandis, J. Silvestre-Blanes, V. Sempere-Payá, Robust Downlink Mechanism for Industrial Internet of Things Using LoRaWAN Networks, Electronics 10 (17) (2021).

[24] G. Hassan, H. S. Hassanein, MoT: A Deterministic Latency MAC Protocol for Mission-Critical IoT Applications, in: 14th International Wireless Communications & Mobile Computing Conference (IWCMC), 2018, pp. 588–593.

[25] T. Polonelli, D. Brunelli, A. Marzocchi, L. Benini, Slotted aloha on lorawan-design, analysis, and deployment, Sensors 19 (4) (2019) 838.

[26] D. Zorbas, K. Abdelfadeel, P. Kotzanikolaou, D. Pesch, TS-LoRa: Time-slotted LoRaWAN for the Industrial Internet of Things, Computer Communications 153 (2020) 1 – 10.

[27] D. Zorbas, ActLoRa: Supporting Actuators in LoRaWAN, in: IEEE International Conference on Communications (ICC), 2022, pp. 4426–4431.

[28] Semtech Corporation, How to Qualify a LoRaWAN Device in Europe, https://lora-developers.semtech.com/?ACT=72&fid=30&aid=48_0znCpZpvImL3agza59hG&board_id=1, online; accessed 10-May-2024 (2018).

[29] N. Kouvelas, V. S. Rao, R. V. Prasad, G. Tawde, K. Langendoen, p-CARMA: Politely Scaling LoRaWAN, in: EWSN, Vol. 20, 2020, pp. 25–36.

[30] LoRa Alliance, RP002-1.0.3 LoRaWAN® Regional Parameters, https://lora-alliance.org/resource_hub/rp2-1-0-3-lorawan-regional-parameters, Online; accessed 11-May-2024 (2021).