IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 24 September 2024, accepted 30 October 2024, date of publication 8 November 2024, date of current version 22 November 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3494038

RESEARCH ARTICLE

Make or Break? How LoRaWAN Duty Cycle Impacts Performance in Multihop Networks

JEFERSON RODRIGUES COTRIM AND CÍNTIA BORGES MARGI, (Senior Member, IEEE)
Escola Politécnica da Universidade de São Paulo, São Paulo 05508-010, Brazil

Corresponding author: Jeferson Rodrigues Cotrim (jcotrim@larc.usp.br)

This work was supported in part by the Coordenaçãao de Aperfeiçoamento de Pessoal de Nível Superior-Brasil (CAPES) under Grant 001, and in part by the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) Project under Grant 2022/07523-8. The work of Jeferson Rodrigues Cotrim was supported by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) Fellowship under Grant 151163/2022-7. The work of Cíntia Borges Margi was supported by CNPq Fellowship under Grant 311687/2022-9.

ABSTRACT LoRaWAN is a widely adopted protocol for Internet of Things applications with energy constraints and large coverage area demands. The protocol standardization defines a star-of-stars topology, where end-devices send their packets directly to a gateway that forwards the messages to the network server, where they are handled depending on the application. LoRaWAN is at the top of the Long Range (LoRa) physical layer based on the Chirp Spread Spectrum (CSS) modulation, where signals are spread along the frequency by the Spreading Factor (SF). The modulation technique gives the LoRaWAN long range coverage characteristics, that could reach 45 kilometers in theory. Furthermore, the protocol adopts the Industrial, Medical, and Scientific (ISM) frequency band plans, which impose spectrum occupation limitations, and the constraints are defined by a device Duty Cycle (DC). The DC imposes a restriction of 0.1% up to 10% in transmission time, limiting the total amount of messages an end-deceive could send. In order to increase even more the coverage area, many works proposed the adoption of multihop techniques in LoRaWAN networks. Proposed solutions are based on routing and relay approaches, both with benefits and limitations. However, the literature does not explore the DC constraints on the multihop LoRaWAN solutions since packets might be enqueued. The packet delay increases due to the time spent in the buffer, and the delivery ratio decreases because packets are dropped when the buffer is full. To understand such impact, we provide a model to generalize the DC behavior on multihop LoRaWAN networks, focused on the packet delivery ratio and delay metrics. We evaluate our model analytically and experimentally. First, we conducted a numerical comparison between the standard network with the multihop Long Range Wide Area Network (LoRaWAN) over duty cycle constraints. The comparison considers a rural area and the coverage area limited by the 802.11 ah propagation model. The results elucidated the tradeoff between packet size, number of packets, and delay. The two-hop solution could decrease the delay compared with direct transmission using high spreading factors. However, in dense networks, the intermediate device buffer increases the packet delivery delay. It could reduce the packet delivery ratio because the device must respect the duty cycle to forward packets. On the other hand, higher spreading factors decrease the payload size and need more packets to transmit the same message. Then, we compared analytical and experimental results in four relayed scenarios with different numbers of devices and different intermediate device buffer sizes. The results show the DC significantly increases the network delay and packet losses. The intermediate device also causes packet prioritization, making the network unfair. The multihop strategies are relevant for many applications, including coverage area extension, but the solutions must consider the DC constraints as a limitation to their adoption.

INDEX TERMS LoRaWAN, Internet of Things, multihop, duty cycle.

The associate editor coordinating the review of this manuscript and approving it for publication was Kaigui Bian.

I. INTRODUCTION

LoRaWAN is a prominent wireless communication protocol in the Internet of Things (IoT) context, designed for applications with power constraints and long-range connectivity demands [1], [2]. The protocol works in the sub-GHz ISM frequency band on a typical star-of-stars topology. The network consists of end-devices, gateways, and a network server. Since the protocol uses ISM frequencies, there are severe duty cycle limitations to access the air interface [3]. Every country or region decides the amount of time each end-device could access the spectrum. The LoRa Alliance provides a document standardizing regional parameters [4]. The CSS modulation allows the transmission range to reach up to, in theory, 45 km. However, many scenarios, such as urban areas, present poor performance due to signal fading. Several works propose multihop solutions over LoRaWAN to increase the coverage area or protocol reliability. These proposals are classified as routing and relay approaches [5], [6].

The Duty Cycle (DC) imposes a limitation on channel access for every device on the network. The values range from 0.1% up to 10%. In Europe, for example, it is common for devices to adopt 1% DC, which means, after a one-second transmission the device must stay quiet for 99 seconds. The DC impact on the LoRaWAN network has been studied in the literature. Adelantado et al. [1], analytically explore the LoRaWAN limits by providing an overview of throughput and network maximum number of devices, including DC restrictions. The work considers the LoRaWAN standardization for network definitions and regional limitations. Sørensen et al. [7] explore a model to understand the latency impact in Class A devices. The evaluation shows the DC increases the total network latency and decreases the total number of devices. Benkahla et al. [8] propose a mechanism of time-sharing, where devices with available time donate their time to devices with transmission requirements above the DC restrictions. Deng et al. [9] proposed an improvement for LoRaWAN based on an adaptive DC, where end-devices dynamically change their DC based on metrics, such as energy consumption. The results show a decrease in the network energy consumption and an increase in Packet Delivery Ratio (PDR) for a larger number of devices. The aforementioned works discuss the DC effects on a standard LoRaWAN network. The works explore the DC limitations without any concern about uplink or downlink messages. Most of the works focus their efforts on the uplink messages since it is the most common traffic in LoRaWAN networks.

The literature on multihop LoRaWAN includes routing and relaying proposals, system evaluations [10], and modeling [11]. Most of these proposals do not include the DC restrictions. The evaluation conducted in [10] and the system model in [11] include the DC restrictions, but without the complexity the subject requires. The lack of discussion and evaluation of the DC over a multihop LoRaWAN network leads us to evaluate its impact and performance.

In order to understand how the DC impacts the operation and performance of the multihop LoRaWAN network, we employ two different techniques: modeling and testbed evaluation. The model describes the system using queuing theory, where the main characteristics are the presence of a buffer and the number of transmission frequencies available on the intermediate devices.

We do not consider any specific approach (relaying or routing) for the multihop network in the proposed model. The devices between end-devices and the gateway are considered intermediate devices capable of receiving and forwarding packets. According to the related literature and LoRaWAN applications, most traffic originates from the end-device and goes through the gateway, commonly referred to as uplink messages. However, the protocol is bidirectional and the downlink messages, which go from the network server to the end-devices, have an important role in the network for device actuation and updates. For example, the Adaptive Data Rate (ADR) mechanism uses the downlink messages to set up new configurations on the end-devices. However, from the intermediate devices perspective, all messages are incoming messages, and without prior distinction set by network managers, all packets receive equal treatment.

The objective of our work is to provide a mechanism to generalize DC behavior on multihop LoRaWAN networks. Additionally, we intend to understand how the DC impacts the network over critical metrics such as delay and PDR. Thus, we are able to evaluate the feasibility of the multihop approaches.

The theoretical model describes the network behavior for PDR and delay metrics, providing insight into the feasibility of multihop solutions. We describe the general intermediate device DC behavior using a flowchart. Based on the proposed model, we design a set of practical experiments and software for analytical evaluation. The experiments and analytical results show the DC increases the packet delay and the packet loss. The delay reached values that cause packet aging, making the packet unnecessary for most applications. The results also indicate a packet prioritization in the intermediate device buffer for some scenarios. This behavior happens in scenarios with regular packet transmission and payload size. An end-device with higher data transmission could also monopolize all the intermediate device capabilities and prevent other end-devices from having their packets forwarded.

The LoRaWAN multihop networks are relevant for many applications to increase coverage area, decrease device energy consumption and delay, and increase packet delivery. However, the intermediate devices need to implement a mechanism to avoid unwanted traffic and packet prioritization. The adoption of multihop networks must include an evaluation of DC constraints due to packet aging effects and increased intermediate device energy consumption caused by packet concentration.. In dense networks, packet loss is another concern due to the intermediate device buffer restrictions.

The remainder of this paper is organized as follows: Section II presents characteristics of LoRaWAN and related work. Section III describes the proposed DC model. Section IV details experiments, presents the results, and evaluate their implications. Section V concludes with a summary of the findings and suggestions for future work.

II. LoRaWAN

LoRaWAN is a data link layer protocol developed by the LoRa Alliance designed at IoT applications with low energy consumption and large coverage area requirements [12]. The network topology is a star-of-stars composed of three main elements: the end-devices, the gateway, and the LoRaWAN Network Server. Figure 1 presents the LoRaWAN architecture with its components.

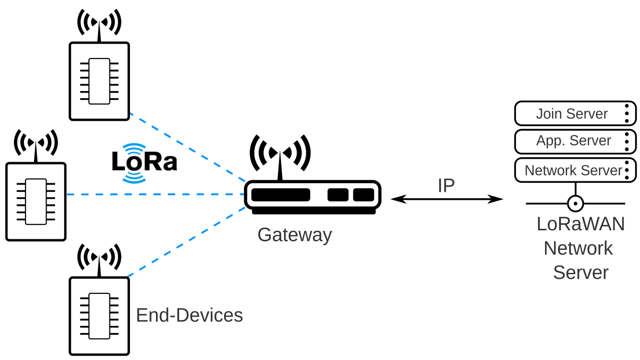


FIGURE 1. LoRaWAN architecture.

The end-devices are responsible for generating and transmitting new packets. The gateway receives a packet from the end-devices and forwards it using IP protocol to the Network Server. The LoRaWAN network server is composed of a network server, an application server, and a join server. The network server manages all tasks related to the LoRaWAN network. The application server connects the data to the specific applications. The join server handles end-device validation on the network.. LoRaWAN network implements three transmission modes called Classes. Class A must be implemented in all end-devices, and it determines that after a transmission the device must open two reception windows to receive possible downlink packets. If the end-device receives a packet in the first reception window, the second reception window is not opened. Class B introduces beacon times to increase the possibility of end-device receiving downlink messages other than in the two main reception windows. Class C keeps the radio on after every transmission. LoRaWAN is a bidirectional protocol, supporting uplink messages from end-devices to the network server and downlink messages from the network server to the end-devices. The downlink messages are often used for end-devices firmware updates (Firmware Over The Air (FOTA)) or network actuation. The uplink messages represent most of the network traffic due to data collection from end-devices. According to regional parameters documentation, the uplink and downlink traffic are set up in different channel frequencies, and more frequencies are available for uplink than downlink packets.

LoRaWAN operates in the ISM frequency band, which imposes severe limitations von channel occupation for devices. The Regional Parameters document defines the DC restrictions for each country or region [4]. In the literature, it is common to use European regulation as a reference for DC restrictions. The DC imposes a time limit to use the air interface, for example, 1% in Europe. It means a one-second transmission requires 99 seconds of silence. Other DC values defined in regional parameters range from 0.1% to 10%.

In a regular network, the DC limits the number of messages an end-device is allowed to send. One effect is that a device would not be able to send all necessary data for an application within a given time period. Additionally, DC impacts downlink messages, as the network server is unable to transmit packets to each end-device due to gateway transmission restrictions [1]. Moreover, any of these scenarios create a bottleneck in the network in terms of delay and throughput. However, these challenges could be addressed by changing the network parameters such as SF, bandwidth, and payload size.

The effects of DC on a multihop network are very different. In this context, we use the term intermediate device to refer to a device in the middle of the communication between the end-device and the gateway. The intermediate device receives packets from multiple end-devices and need to forward them. However, if the number of received packets exceeds the intermediate device transmission capacity, these packets would be dropped. An option is to buffer the incoming packets. However, the buffer size is limited by the hardware characteristics and the total packet delivery time could reach high values, potentially causing the application to discard packets due to aging. Figure 2 depicts the multihop topology.

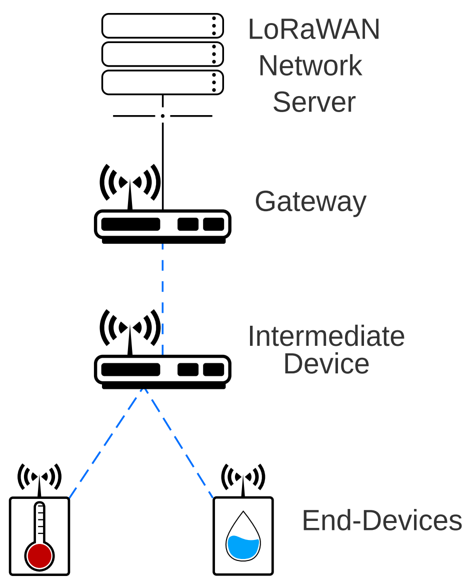


FIGURE 2. General multihop topology.

Despite the fact that LoRaWAN achieves long ranges, there are scenarios where the protocol presents a poor performance. In urban areas the presence of many obstacles decrease the transmission range due to fading and multi-path. In rural or wild areas, such as dense forests and mountain reliefs, the coverage area decreases due to shadow areas and obstacles. Therefore, several works in the literature introduce multihop solutions on the LoRaWAN network. The solutions are split into routing or relaying, all showing the benefits and constraints of multihop approach to LoRaWAN [13]. However, no work includes a detailed evaluation of the duty cycle effect on the multihop LoRaWAN networks. The modeling designed in [11] explores the multihop characteristics with a brief discussion about the DC. The evaluation conducted by Islam et al. [10] includes the DC effect over three routing algorithms. The results show a decreasing in PDR with the increase of DC percentage. However, the authors do not expand the discussion of the effects of this behavior on the network and devices. Also, they do not observe the causes besides the routing algorithm. Most of the solutions developed for multihop LoRaWAN networks use a routing technique that requires message exchange between devices. However, these works do not include the DC limitations or do not show if the DC is, or not, a problem to the proposed system. Fedullo et al. [14] address the DC constraints on regular LoRaWAN and propose a solution using a relay system and a hybrid medium. Thus, the article does not explore the solution and the DC for a multihop network. Shida et al. [15] introduces a synchronization mechanism for multihop LoRaWAN networks including DC constraints. Although, authors do not include results and discussion about the DC effects on the network performance. In 2022, the LoRa Alliance standardized relay devices, including a synchronization mechanism between end-devices and relay, as well as a wake-on-radio packet to wake up the relay for packet reception [16]. However, the standardization does not include any reference to duty cycle constraints, leading us to understand the relay is under the regular DC restrictions.

The proposed multihop solution generally focus on the uplink messages, as they represents the most common traffic in the network. A few solutions give an alternative to increase the evaluation metrics for downlink messages. Qin et al. [17] developed a new scheme to guarantee and improve the downlink packet delivery using a ZigBee alternative to connect the devices. Their approach involves using a separate channel for downlink packets, eliminating the need for end devices to open a reception window. The results show a better performance for networks with high demands for actuation, however, the solution includes another protocol in the stack and does not provide any discussion about the Zigbee performance in the scenarios. Tian et al. [18] developed a new protocol called LoRaHop that includes uplink and downlink messages in the solution. The proposed protocol requires a synchronization scheme, which could make the solution unfeasible due to the DC restriction, although the authors do not consider this limitation in their analyses. The evaluation does not include a specific view of the downlink messages performance. Authors in [19] adopt the downlink messages to determine if an end-device is inside of a gateway range. However, the multihop network is focused on the uplink packet performance. Evaluating other works in the field, including surveys [5], [6], we notice that the multihop proposals are focused on the uplink messages. Moreover, evaluations are conducted from the network perspective, and when conducted at the intermediate device level, messages are treated without distinction between uplink or downlink messages.

Our work explores the DC constraints in a multihop LoRaWAN network and aims to understand the protocol performance. To achieve this objective, we have designed an analytical model to describe system behavior and generalize the solution to both routing and relay networks. This approach allows the evaluation of the network without considering any specific solution. We are interested in understanding limitations in terms of delay and packet losses, as well as discussing other network aspects such as energy consumption, throughput, and network size.. The next section presents the proposed theoretical model, which describes the DC behavior in a multihop LoRaWAN network.

III. THE DUTY CYCLE MODEL

This section presents the proposed model designed to understand the multihop LoRaWAN network behavior under the Duty Cycle (DC) limitations. Since the goal is to evaluate the effects of the DC, we do not specify whether the intermediate device acts as a relay or a router.

The DC acts differently for regular and multihop LoRaWAN networks. The standard network is a star-of-stars topology, with the end-devices sending their packets to the gateway. In this case, the DC decreases the total time each end-device access the air interface based on the relation between payload size and SF. The multihop network introduces at least one more hop between end-device and the gateway. This intermediate device also has to comply with a DC, which introduces additional delay to the overall packet delivery time and tends to decrease the network throughput. On the standard LoRaWAN, the packet delay is the transmission time between end-device and gateway, and it is equal to the Time on Air (ToA). The DC introduces a delay D between packets. On the other hand, in a multihop network, the delay between packets caused by the DC is added to the total delay time. Furthermore, the intermediate device buffer size introduces other constraints in the system because, in dense networks, the buffer could not be enough to receive all messages, increasing the network packet loss. Figure 3 depicts the DC behavior in regular and multihop LoRaWAN networks for uplink packets, focused on the delay. The figure does not include the downlink traffic to simplify the system, although the downlink messages are important to LoRaWAN networks and must be included in the solution.

First, we need to establish some general definitions of the intermediate device characteristics. Incoming packets may arrive in different channel frequencies, although we adopted a perspective where all packets are enqueued in buffers. Considering this approach all incoming packets could be regarded as a single flow of messages. The number

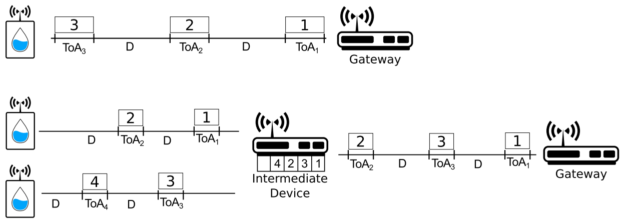


FIGURE 3. Duty Cycle behavior in a standard and multihop network, for uplink traffic.

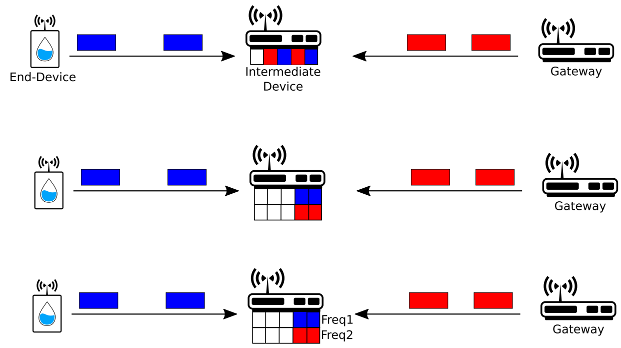


FIGURE 4. Uplink and Downlink packets behavior on intermediate device buffer.

of radios increases the total number of incoming packets, although, from the intermediate device perspective, all the data must be allocated in a buffer to be later delivered. The packet transmission needs to follow the regional channel definitions. Devices perform a frequency hopping across available frequencies in each channel. For example, in the United States, the end-device performs the frequency hopping over eight frequencies available for each channel. The DC is related to the device, then the transmission frequencies need to share the available transmission time. For instance, if the device operates under the US parameters, it has eight transmission frequencies and a 1% DC, each frequency would effectively have a DC of 0.125%. Unlike the intermediate device does not implement any of the regular Classes available in the LoRaWAN standardization.

The proposed model does distinguish between uplink and downlink messages. All intermediate device received packets are incoming messages and are handled according to the network-established rules. Also, the network manager could implement separate buffers for uplink and downlink messages according to the application needs. The implementation of different buffers opens the possibility of packet prioritization in the intermediate device. If the hardware adopted for the intermediate device implementation is robust with multiple radios like the gateways, it is possible to define specific frequencies for uplink and downlink messages. The idea is to follow a frequency plan like to the ones made in the regional parameters for reception and transmission. Figure 4 depicts three possible situations of downlink and uplink messages in the intermediate device. First, the intermediate device shares the buffer between uplink and downlink messages without distinction when forwarding them. Second, the intermediate device has separate buffers for uplink and downlink messages, allowing packet prioritization between uplink or downlink packets. The third option is a different buffer for uplink and downlink messages and dedicate frequencies for them.

Figure 5 describes the proposed DC system model. This flowchart describes the behavior of end-device, gateways, or intermediate devices. The packet arrival for the end-device is the new data collection that needs to be transmitted. In the gateway, the packet arrival is the packets from the network server to be sent to the end-devices. The packet arrival in the intermediate devices refers to the incoming packets that could be uplink or downlink messages from end-devices and gateway, respectively. All devices must implement a buffer due to the DC transmission constraints. If the network server, for example, needs to transmit packets for many end-devices, the gateway will enqueue these packets to comply with the DC. The model is based on two events: the arrival of a new packet and the end of the DC. If the DC period finishes, the intermediate device checks the buffer. If there is at least one packet in the queue, the intermediate device transmits the packet and initiates a new DC cycle. We do not specify the queue discipline because we understand it is an application decision and does not impact the system model. If the buffer is empty, the intermediate device waits for another event. When a packet arrives on the intermediate device, it triggers one of two events. First, if the end-device is waiting for the end of DC, the packet will be queued if the buffer has available slots. If the buffer is full the intermediate device drops the incoming packet. Second, if the intermediate device is ready to transmit, it will check the queue to transmit the old packets and put the new incoming packet in the queue. If the buffer is empty, the arriving packet is transmitted.

The Duty Cycle (DC) general behavior is described by the queuing theory with a single arrival queue, a variable buffer size, and n servers to attend. The buffer size is dependent on the resources available on the intermediate device hardware. A constrained device has a small buffer or even no buffer. In contrast, an intermediate device built with hardware like a Raspberry Pi, for instance, can offer a more substantial buffer. In our model, each server is a transmission frequency. For a better definition, we followed the common channels specified in the regional parameters document, which defines the end-device having between and eight transmission frequencies. With one server, we observe the basic system behavior, and with 8 frequencies, we approximate the model to real scenarios.

Based on the definitions, we classify the intermediate device behavior under the Kendall notation for queuing theory. The proposed model could be generalized as an M/M/c/K queue, where the first two "M's" refer to an exponential distribution for the arrival and server rate. The "c" defines the number of servers, and the "K" is related to the buffer size. We consider all incoming packets to be managed by a single server, with all packets allocated to the same queue. The buffer size depends on the intermediate

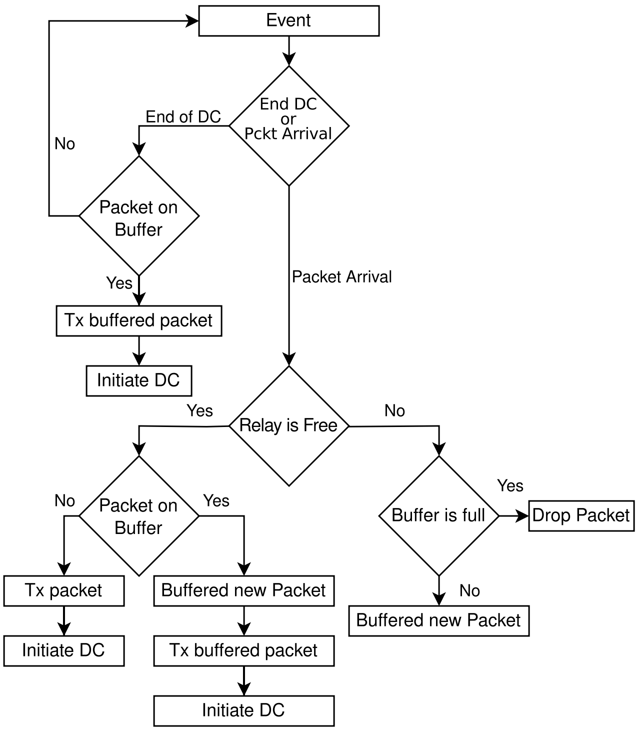


FIGURE 5. **Duty cycle flowchart.**

device hardware, and then the queue model is defined as M/M/1/K. The buffer size could be large enough, and the model simplified to a M/M/1 queue.

A. ARRIVAL AND SERVICE RATE RELATION

The arrival (λ) and service (μ) rates are defined as the ratio between the number of packets to the total time (T_{Total}). Both metrics are used to evaluate the system performance and design a network, as they provide insights of the device reception and transmission capability.

$$\lambda, \mu = \frac{\sum_{i=0}^{n} Packets(i)}{T_{Total}} \tag{1}$$

The LoRaWAN packet size is strictly related to the transmission time, defined as ToA and standardized [20]. The ToA is dependent on the adopted transmission SF and payload size. Then, we define λ and μ as

$$\lambda = \frac{\left[\sum_{i=0}^{n} \left(ToA_{in}(i) + ToA_{in}(i) \times (100 - DC)\right)\right] \times N}{T_{Total}} \tag{2}$$

where ToA_{in} is the ToA of the incoming packets, DC is the Duty Cycle (DC), and N is the number of end-devices and,

$$\mu = \frac{\left[\sum_{i=0}^{n} \left(ToA_{out}(i) + ToA_{out}(i) \times (100 - DC)\right)\right] \times f}{T_{Total}} \tag{3}$$

where ToA_{out} is the ToA of forwarded packets and f is the number of available transmission frequencies.

The relation between λ and μ defined the system utilization ρ as

$$\rho = \frac{\lambda}{\mu} = \frac{\left[\sum_{i=0}^{n} \left(ToA_{in}(i) \times (101 - DC)\right)\right] \times N}{\left[\sum_{j=0}^{m} \left(ToA_{out}(j) \times (101 - DC)\right)\right] \times f} \tag{4}$$

Considering a scenario where all packets have the same size and the transmission and reception use the same SF, Equation 4 is reduced to

$$\rho = \frac{N}{f} \tag{5}$$

that shows the number of end-devices N must be equal or less to the number of available frequencies to the system works without packet loss. We also understand N as the sum of devices sending several packets that do not overcome the transmission capability, i.e. the total incoming packet request a lower ToA than the available in the intermediate device.

Equation 4 enables the system block probability estimation when the intermediate device has a buffer size of K slots.

$$P_{Block} = \frac{(1 - \rho) \times (\rho^K)}{1 - \rho^{K+1}} \tag{6}$$

The P_{Block} and ρ allow us to estimate the packet loss in the intermediate device for a multihop LoRaWAN network giving the possibility for a manager to design the network according to these restrictions. We estimate the individual PDR for end-devices (uplink) or gateway (downlink) is equal to the probability of an incoming packet access the intermediate device, then

$$P_{free} = PDR_{device} = 1 - P_{Block} \tag{7}$$

where P_{free} is the probability of an incoming packet being received in the intermediate device. The proposed model presents the effects of the bottleneck caused by the DC of an intermediate device in the PDR. We ignore other packet losses in the system, for example, the ones caused by the ALOHA protocol or gateway congestion. In a scenario with one intermediate device with a single transmission frequency and a single buffer slot ($K = 1$), with four end-devices connected, the ρ ratio is equal to 4, and the P_{Block} is equal to 0.8. We estimate the PDR_{device} is limited to 20% due to the blocking probability being equal to 0.8. In this scenario, with a single intermediate device, the network PDR is equal to the individual PDR, since we consider all the incoming packets are delivered to the gateway for uplink messages, or end-devices for downlink messages.

B. DELAY

The delay is one the most critical metrics in a multihop network, and even more in a LoRaWAN network that naturally introduces large latency due to the CSS modulation. We previously defined [11] the packet delay (D) for a multihop LoRaWAN network, and here we present a modified version in Equation 8. The uplink or downlink packet delay for a multihop network is the sum of the transmission times and the time spent in the buffer of the intermediate devices. The transmission time is determined by the ToA, which is related to the payload size and SF. According to the CSS definitions, a higher SF produces a higher transmission time [20]. The time spent in the intermediate device buffer varies according to the buffer size and the amount of messages the device receives. The time spent in the buffer is also dependent on the payload size because intermediate device transmission time is affected by the DC, which is determined by the packet size. The same intermediate device, with the same buffer size, produces different delays for a network with a small or large average packet size. In dense networks, a large buffer size introduces a higher delay because the number of incoming messages overcomes the transmission capabilities of the intermediate device.

$$D = \sum_{i=0}^{n_{Hops}} ToA(i) + \sum_{j=0}^{n_{Hops-1}} \Big(T_B(j)\Big) + \Delta_T \tag{8}$$

where Δ_T is an interval between 0 and ToA, and it is the time related to the transmission occurring in the intermediate device while the new incoming packet arrives. The T_B is the time a packet spent in the buffer, defined as

$$T_B = \sum_{i=1}^{B_p} \big(ToA_{prev}(101 - DC)\big) \tag{9}$$

where B_p is the packet position in the buffer. Notice we do not consider in our definition the time spent processing due to the fact that the periods on the buffer tend to be higher than any processing time.

LoRaWAN transmits confirmed and unconfirmed messages. These type of packets affects the delay by increasing, or decreasing, the number of packets on the buffer. Confirmed messages require an acknowledgment (ACK) from the Network Server, that sends it as a downlink message. The ACK is treated by the intermediate device as an incoming message and entered in the buffer queue. The presence of ACK messages in the buffer increases the total delay time. The unconfirmed messages do not require the ACK, no increasing the number of messages in the buffer. The intermediate device changes the expected delay on the network and the multihop solutions must consider the new delay and how it could affect the reception windows from end-devices. A regular end-device opens one or two reception windows after transmitting an uplink message to receive any kind of downlink packet including an ACK. If the downlink packet stays in the intermediate device buffer, the delay could be higher than the reception window open time, and the message, when it comes to the end-device, could not be received. In the case of an ACK, the packet loss generates unnecessary retransmissions.

IV. EXPERIMENTS AND RESULTS

In this section, we explore application scenarios for multihop networks, evaluating the protocol performance under the DC constraints, applying the proposed model, and comparing it with the regular LoRaWAN. First, we numerically analyze the behavior of a multihop LoRaWAN network under DC constraints and compare it with that of a standard network. Next, we compare the experimental and analytical results acroos four scenarios with different numbers of end-devices and data traffic.

A. STANDARD AND MULTIHOP COMPARISON

Rural areas are commonly associated with LoRaWAN applications due to their characteristics, including vast areas and limited energy availability. However, rural scenarios could present shadow areas or obstacles that decrease the coverage area, making the multihop solution a real option. The propagation models for LoRaWAN are defined for specific scenarios, for example, in [21] authors explored the path losses in Oulu City, Finland. Cotrim and Kleinschmidt [11] explored the IEEE 802.11ah propagation model [22] for LoRaWAN and provided the maximum coverage area for each SF, with values from 3.43 km for SF7 up to 7.62 Km for SF12. We adopted the 802.11ah propagation model in our analysis because it is a general model for LPWANs working on 915 MHz frequency band.

For comparison purposes, we considered a circular area with a 7 km radius allowing direct communication using SF12 in the border of the area and the adoption of an intermediate device to allow two-hop communication with devices using SF7. Figure 6 depicts the proposed scenario, first with direct communication using SF7 in the inner circle and SF12 in the outside circle. The second scenario describes the adoption of relays to connect the distant end-devices through two hops using SF7. In our analysis, we evaluate the behavior of one relay and the set of end-devices connected to it.

We considered each sensor transmits 1 packet every 30 minutes because rural scenarios do not require constant updates. We simplified the scenario to consider only uplink messages.

The maximum service rate of each relay is estimated according to the packet size of received messages that determines the ToA and the respective DC. Admitting a packet size of 200 Bytes, the ToA is equal to 0.3382 s and the DC is 33.81 s, a total of 33.81 seconds to forward the message and await the end of DC. Then, the service rate of relay per hour is

$$\mu = \frac{3600}{33.81} = 106.5 \; packets/hour$$

The service rate μ and the arrival rate λ must be equal to avoid blocks on the relay, then the maximum number of end-devices per relay is

$$106.5 = 2 \times N$$
$$N = 53$$

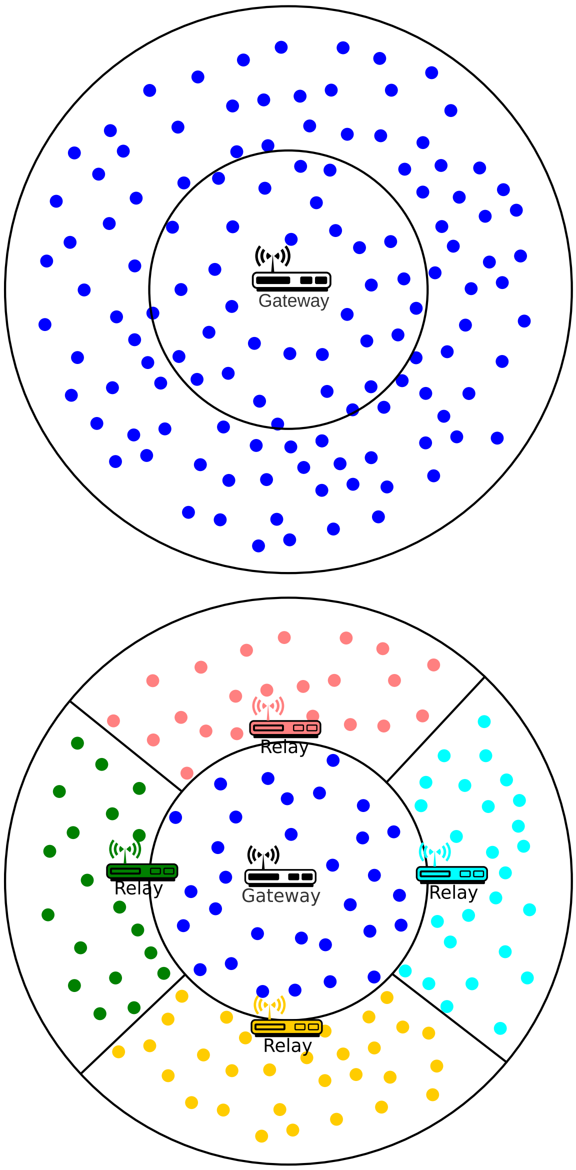


FIGURE 6. Standard and relay scenarios for performance comparison.

The packet delay could achieve values from the sum of ToAs of every hop to achieve the gateway up to the ToAs plus the time spent on the intermediate device buffer. In our example, there is only one relay between the end-device and the gateway, the minimum delay is

$$Delay_{packet} = 2 \times ToA$$
$$Delay_{packet} = 2 \times 0.3382 = 0.6764\ s$$

The maximum packet delay depends on the relay buffer size and the position the packet enters in the buffer. Admitting the relay has 20 buffer slots and the packet enters the last position, the maximum packet delay is

$$Delay_{packet} = 2 \times ToA + 20 \times (99 \times ToA)$$
$$Delay_{packet} = 2 \times 0.3382 + 20 \times 33.48$$
$$Delay_{packet} = 670.27\ s$$

approximately 11 minutes to delivery a packet.

In our example, the arrival and service rates are equal to avoid blocks in the intermediate device. However, in real deployments, packets could arrive at the same time or during the DC period, and if the buffer is already full the new incoming packets are dropped. Furthermore, the end-devices dispute the air interface to send their packets and LoRaWAN adopts ALOHA as the MAC layer that is well-known to have a low performance under dense networks.

The standard LoRaWAN delay is associated only with the ToA. In the example, end-devices using SF7 present a delay of 0.3382 s. When end-devices adopt SF12 the regional parameters limits the packet size to 51 Bytes. Considering this limitation, to transmit the same 200B the end-devices using SF12 must split the packet into 4 small packets of 50 Bytes. The ToA of each packet is 2.8 s and the DC is equal to 277.2 s. Then, to transmit the original message of 200 Bytes the delay time is equal to 1120 s (almost 19 minutes).

B. ANALYTICAL AND EXPERIMENTAL COMPARISON

This section presents the experiments and results of DC evaluation for different scenarios. Since the main idea is to evaluate the behavior of the DC, we implement the intermediate device as a relay. The testbed equipment includes a Dragino LoRaWAN shield and an Arduino Uno for the end-device, while the relay and the gateway are based on a RisingHF LoRaWAN hat and a Raspberry Pi 3B+. The Network Server is the chirpstack project [23]. In all experiments, the end-devices and the relay are configured with 1% DC. We implemented the software for the analytical evaluation in Python, following the algorithm presented in Figure 5. The topology of evaluated scenarios is based on the multihop Figure 6, with one Network server, one gateway, and one relay. The number of end-devices varies from 4 up to 16.

The decision to use robust hardware for relay implementation instead of a simple device is because the Raspberry Pi provides a larger buffer size, more processing capabilities, and no energy constraints. This hardware allows us to focus only on the DC constraints. The intermediate device energy consumption, in dense networks, depletes the battery faster than the network expected lifespan [11]. This behavior occurs because the increase in traffic prevents the intermediates to sleep, and the forward transmission is the most costly activity performed by the devices. Furthermore, using the gateway hardware to implement the intermediate devices allows us to use more frequencies to transmit or receive packets, increasing the network capabilities. In real applications it is common to adopt a constrained device, such as an Arduino, to implement the devices, because it implements a sleep mechanism to turn off the device when it is not performing any activity to save energy. However, the intermediate device must be awake to receive any incoming packet demanding the device stay on all the time or implement a synchronism technique. When the radio is kept turned on, the battery depletes faster. The synchronism mechanism requires the implementation of it on every device on the network, increasing message exchange and processing time, causing more energy consumption and delay increasing. The adoption of a robust device to implement the intermediate device demands the installation of solar panels, for example, for energy supply increasing the solution cost. However, it allows the adoption of LoRaWAN radio with more frequencies, increasing the reception and transmission capabilities. Both solutions with constrained or unconstrained devices must be considered for real application scenarios and the decision must consider the above considerations.

We designed four experimental scenarios, summarized in Table 1. There are no downlink packets in the scenario since all received messages in the intermediate device are considered incoming packets, as explained in Section III. Scenarios 1 and 2 have 4 end-devices sending packets continuously, only complying with the DC, with 242 Bytes payload size. The difference between both scenarios is the buffer size, with 20 slots in scenario 1 and 2000 slots in scenario 2. Scenario 3 has four end-devices, like the previous ones, but the transmission occurs in a time interval of 120s and the payload size of each packet varies from 1 up to 242 Bytes. Every time an end-device sends its packet, the time slot restarts after the end-device awaits the DC. The intermediate device has 20 buffer slots in scenario 3. In scenario 4, we increase the number of end-devices to sixteen, and the time interval between transmissions to 600 seconds. The payload size for each packet varies from 1 up to 242 Bytes and the intermediate device has 20 slots in the buffer. The reasoning behind the choice of scenarios is to emulate the constraints of the different types of hardware available to design the intermediate device. In all scenarios the devices are configured with SF7 and 14 dBm transmission power. The experiments were conducted in the laboratory to minimize interference.

TABLE 1. Experiments summary.

Scenario	EDs	Tx	Payload (B)	Buffer
1	4	Continuous	242	20 Slots
2	4	Continuous	242	2000 Slots
3	4	120s Time slot	1 - 242	20 Slots
4	16	600s Time slot	1 - 242	20 Slots

Figure 7 shows the delay results for scenario 1, refereed in Table 1, with a small buffer size relay. Observing the delay, we notice it increases quickly and achieves a maximum value, and then the latency of all packets is almost constant. This behavior occurs because of the limited buffer size: if a packet enters the buffer, the time spent on it could be estimated and tends to be constant. This scenario presents a packet prioritization behavior on the intermediate device. The behavior is more evident in the analytical result (depicted in Figure 7a) because the operations intervals are always the same, and we do not include the processing time. The real testbed suffers from the Raspberry processing time, which minimizes the packet prioritization by increasing the time a packet spends within the device (shown in Fig. 7b). The packet prioritization could also occur due to different transmission data rates. An end-device with a higher data rate will consume all the intermediate device buffer resources, preventing other end-devices from accessing the intermediate device.

Apart from the packet prioritization, the total packet loss is almost the same for the analytical and the experimental results. We notice in Figure 8, the end-devices have a low PDR caused by the DC restriction. The average network PDR is 24.75% for the analytical model and 26.25% for the practical experiment. These results are in accordance with the theoretical modeling since the end-device has the same DC restriction applied to the end-devices. The relay receives constant messages from the end-devices, but the available transmission time could support only 1/4 of the received messages.

The time a packet waits in the buffer is at least 99 times the ToA, considering a 1% DC. Thus, the time spent in the buffer determines the packet delay. The buffer size also imposes a maximum delay to a packet. Figures 7a and 7b show a delay of 800 seconds when the networks reach a stationary regime. Considering the proposed delay Equation 8 and the value of ToA for a packet with 242 Bytes in SF7 (0.39 seconds), we find the maximum delay for a packet to be delivered on a two hops network, where all packets have the same size and the buffer has 20 slots.

$$D = 2 \times 0.39 + 20 \times (0.39(101 - 1) + \Delta_T$$
$$\begin{cases} D = 780.78\ s, & \text{if } \Delta_T = 0 \\ D = 781.17\ s, & \text{if } \Delta_T = ToA \end{cases}$$

Figure 9 presents the delay results for scenario 2, with a relay with a large buffer size. For this experiment, we consider the number of packets sent and not a period for analysis because we are interested in showing the delay behavior. We notice the total delay increases linearly for all packets, achieving values of 20000 seconds ($\approx$ 6 *hours*). The huge delay makes the multihop solution unfeasible due to an aging effect on the information. Then, from an application point of view, the long delay turns a packet useless. The PDR results showed in Figure 10 reaches values closer to 1, which means there are no losses in the network. The analytical PDR results do not achieve 100% because the remaining packets in the buffer are disregarded in the calculation. This evaluation allows us to understand the trade-off between PDR and delay caused by the buffer size. If we decrease the buffer size, we also decrease the delay, although we increase the packet loss (PDR decreases).

In scenario 3, we assume the end-devices are sending one packet in a fixed time slot of 120 seconds. Each end-device chooses a time to send its packet during the time slot. After the end-device sends a packet, the time slot is reset. Furthermore, each packet has a variable payload size from 1 to 242 Bytes. The reasoning behind this assumption is to include more randomness in the experiments, which makes relation to

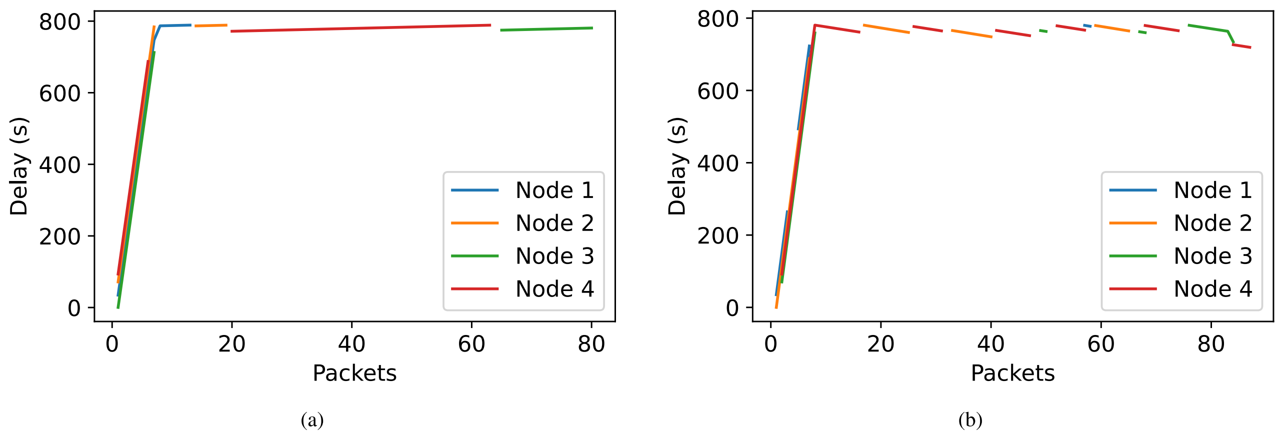


FIGURE 7. **Results for scenario 1 with 4 end-devices and relay buffer with 20 slots. (a) Analytical Delay. (b) Experimental Delay.**

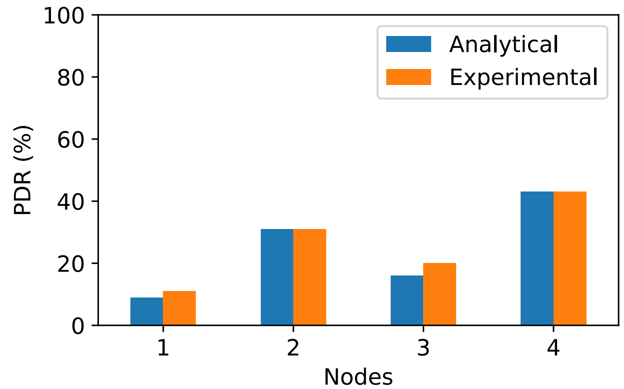


FIGURE 8. **PDR results for scenario 1.**

the real applications. Figure 11 presents the average packet delay for each end-device and the PDR for analytic and experimental scenarios.

The delay presents a difference between analytic and experimental results. The analytical results show an average delay of 400 seconds, while experimental results show an average delay of 330 seconds. This difference occurs because the average packet size differs from both analytical and experimental cases in 13 Bytes. The analytical scenario has a larger average packet size of 137 Bytes, while in the experimental case, the average packet size is 124 Bytes. These packet sizes lead to a ToA of 246 ms and 225 ms, producing a DC of 24.6s and 22.5s, respectively. Evaluating the system behavior, we noticed the whole buffer is used. Considering the average packet size in both situations and the buffer size of 20 slots, we calculated that the theoretical time on the buffer is 490 seconds for the analytical scenario and 450 seconds for the experimental case, a difference of 40 seconds in total delay time. Observing the results in Figure 11a, we noticed almost the same difference in the delay for each end-device. The PDR complements the delay results because if the packets stay more time in the buffer, the probability of new incoming packets being dropped increases. However, the difference between results is around 5%, showing the increase in delay is more significant than PDR.

In scenario 4, we expand the experiment scenario 3 to 16 end-devices and a time slot of 600 seconds for packet transmission. Figure 12 shows the experimental and analytical results. The general behavior is similar to the results obtained with four end-devices, with a PDR performance increase. This scenario and results are closer to a real application due to its characteristics, such as the number of devices, variable packet size, and random transmission time. The scenario evaluation evidences the constraints of multihop network over Duty Cycle (DC) limitations. The delay reaches high values, making some applications unfeasible. The buffer size ensures low packet loss but increases the delay.

In the experimental scenario, the relay is a simplistic device without any mechanism to avoid unwanted traffic. During the data analysis, we noticed the presence of some end-devices that did not belong to our network. These devices take place in the relay buffer, and their packets are also forwarded to our gateway. This unwanted traffic increases the network packet loss and accelerates the buffer fullness.

The difference in the PDR for the results in Figures 11 and 7 occurs because the number of devices increases the dispute to access the relay buffer. In a network with fewer end-devices it is easier for one device to occupy more slots in the buffer, causing device prioritization, as already shown in Figure 7. Furthermore, the decrease in delay indicates the average packet size is lower than the values in scenario 3. A small packet size decreases the time spent on the buffer, and the delay decreases as a consequence. The small packet also stays less time in the buffer, and the probability of a new incoming packet finding a free space in the queue is higher, decreasing the packet loss.

The experiments focus on uplink traffic, although LoRaWAN is a bidirectional protocol with Network Server sending packets to the end-devices. The intermediate device

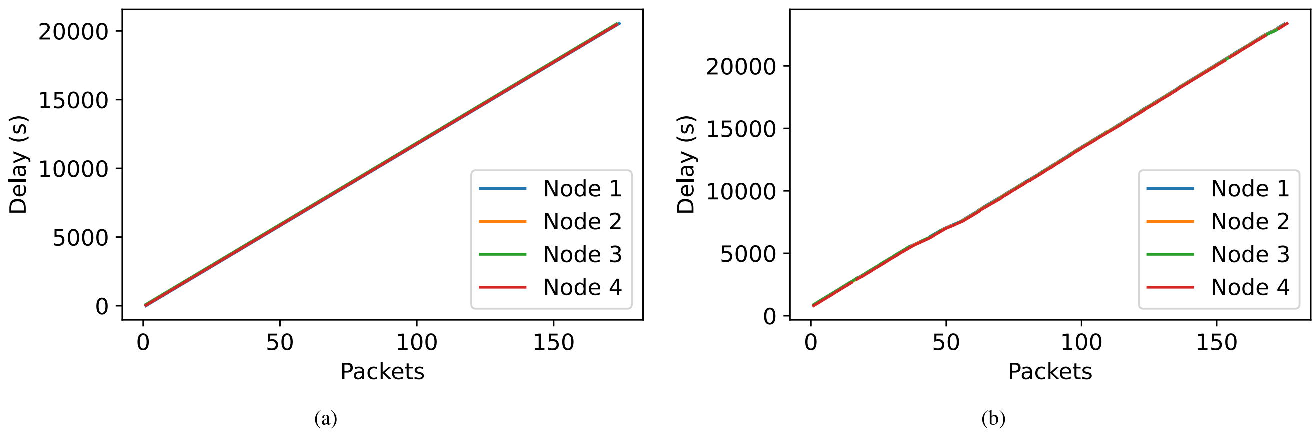


FIGURE 9. Delay results for scenario 2 with 4 end-devices and a relay with 2000 slots buffer size. (a) Analytical. (b) Experimental.

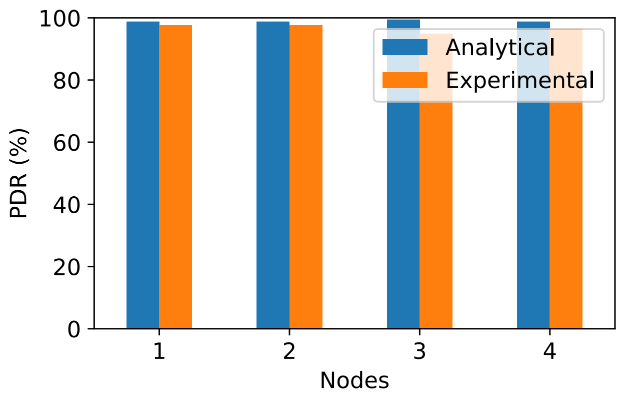


FIGURE 10. PDR results for scenario 2.

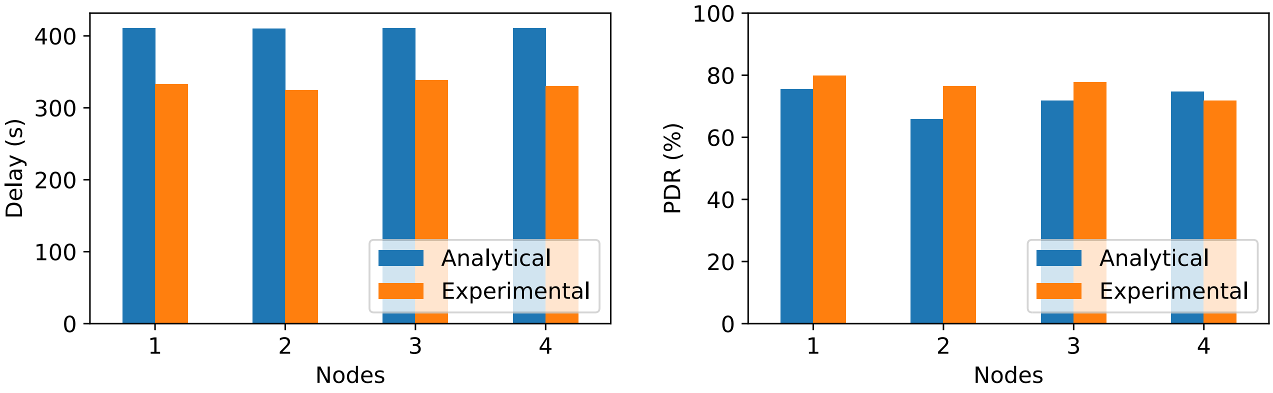
does not differ between uplink or downlink messages, considering both as an incoming packet. However, the packet losses for downlink messages could be more relevant than uplink messages because they carry sensitive information. For example, downlink messages carry data for actuation or device firmware updates. The ADR mechanism also requires the downlink messages to set up new configurations on the end-devices, to change transmission power, and SF to guarantee the minimum requirements for a transmission to occur. To minimize the DC effects on the uplink messages the intermediate device should implement separate buffers for uplink and downlink, or a packet prioritization based on flags in the packet header.

C. STANDARD VERSUS LORAWAN MULTIHOP SOLUTIONS

The literature overview provides a vision of the trade-offs between standard and multihop solutions for the LoRaWAN networks. The multihop approach reduces the overall energy spent compared to the regular network because it is possible to use lower power and lower SF to reach the same coverage area. The throughput could achieve better values in multihop networks due to the possibility of reducing the SF, allowing packets with bigger payload sizes [11]. On the other hand, multihop solutions could require extensive message exchange in the case of routing solutions. The intermediate devices must implement mechanisms to avoid unwanted traffic, even for relaying systems, avoiding unnecessary processing from packets outside the network.

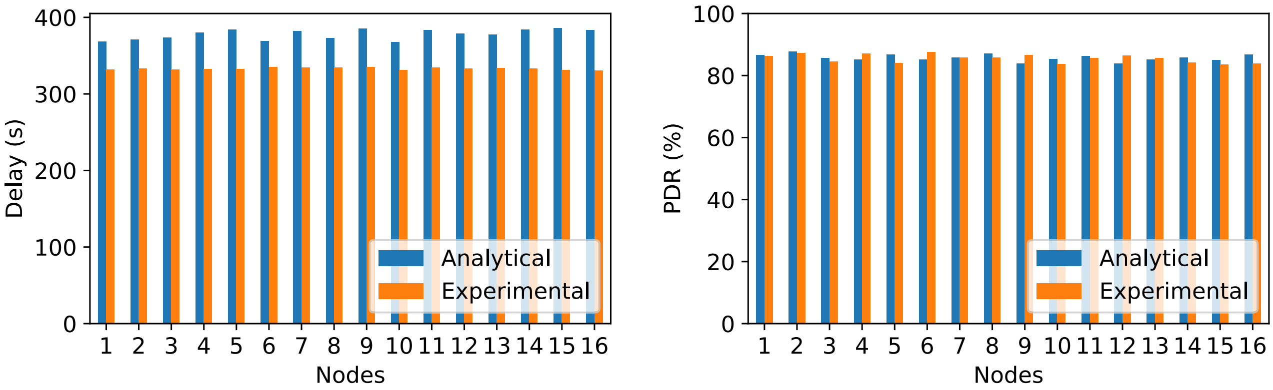
The DC restrictions introduce a challenge to the multihop solutions and change the trade-off perceptions for dense networks. First, as explained in Section IV, the intermediate device represents an intense bottleneck to the network performance, reducing the throughput and increasing the packet delay. The delay causes an effect on the packets, making them obsolete for some applications. Furthermore, we also observe packet prioritization on networks with regular data transmission, increasing the performance of a group of end-devices to the detriment of another devices. In comparison, in the standard network the end-devices access to the air interface is only limited by the DC and the ALOHA protocol. The regular network does not require any distinction between uplink and downlink traffic. The mechanism to receive downlink packets is described by the Classes definitions. For multihop networks, the uplink and downlink traffic could have a different treatment in the intermediate device and also a new reception mechanism. The LoRa Alliance already defined new reception windows for relay devices and changes in the two Class A reception windows when devices are connected to a relay [16]. In most cases, the downlink data requires privileges since it carries update messages to end-devices. These update messages are used for the ADR mechanism or firmware changes, for example.

Furthermore, the intermediate devices must implement techniques to avoid unwanted traffic and packet loops, which have high costs for constrained devices. The intermediate devices consume more energy than the regular end-devices due to the packet concentration and retransmissions. It could cause battery depletion and network isolation. The alternative is to adopt robust intermediate devices with fixed power supply, which is unavailable for all applications and solutions.



(a) (b)

FIGURE 11. **Results for scenario 3 with 4 end-devices, relay with 20 buffer slots, and variable packet size. (a) Delay. (b) PDR.**



(a) (b)

FIGURE 12. **Results for scenario 4 with 16 end-devices, relay with 20 buffer slots, and variable packet size. (a) Delay. (b) PDR.**

V. CONCLUSION

The Duty Cycle (DC) restriction imposes severe limitations for LoRaWAN networks and must be part of the solution design. In the standard LoRaWAN network, the DC decreases the total amount of packet transmitted by each device, affecting the throughput and increasing the need for devices to improve application requirements. On the other hand, multihop networks are more susceptible to DC effects.

In this work, we design a model for DC analysis, which provides a tool for evaluation and further analysis of multihop LoRaWAN solutions. The results show the DC increases the packet delay for unfeasible values and also decreases the PDR. The total delay is also related to the buffer size available in the intermediate device. The greater the buffer is, the larger the packet delay. On the other hand, a small buffer increases the packet loss. Another effect noticed is the packet prioritization in the intermediate device that occurs more significantly on networks with regular traffic patterns. After the buffer becomes full, the incoming packet will be dropped until the next buffer slot is available. When the traffic is regular, the time a slot is available will always coincide with an incoming packet of the same end-device.

The intermediate device development should include the application needs given the direct relation between the device hardware capabilities, mainly the buffer size, and the evaluated metrics. A device with a large buffer size introduces delay in the packet delivery, although it decreases the network packet losses. Intermediate devices could have separated buffers introducing prioritization for different incoming packets, for example, uplink and downlink packets. This could enhance reliability for applications with high PDR or ensure the delivery of downlink configuration packets. However, different buffers increase data processing and require an approach to determine data buffer allocation. Furthermore, an intermediate device with a single transmission frequency limits the number of connected devices because it decreases the packet forwarding. The direct effect is the need for more intermediate devices to connect remote areas. On the other hand, implementations with robust hardware require a fixed energy power supply. For some IoT applications, the available devices are limited to battery-operated ones.

We notice the intermediate device requires mechanisms to avoid unwanted traffic for unauthorized end-devices because these devices increase the delay and forces packet drop when using the buffer. The intermediate device must implement communication with the network server to receive information about the allowed end-devices on the network. In general, downlink messages have higher priority than uplink messages since they carry relevant information to network actuation or device updates. Therefore intermediate devices could implement separate buffers for both traffic, or the Network Server should tag the downlink packet to introduce priority in the queues.

We expected this work to be used as a reference for new solution development in LoRaWAN multihop networks, including relay and routing solutions. Future work includes the design and implementation of mechanisms to avoid unwanted traffic and routing mechanisms evaluation over DC restrictions.

REFERENCES

[1] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of LoRaWAN," *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 34–40, Sep. 2017.

[2] M. Jouhari, N. Saeed, M.-S. Alouini, and E. M. Amhoud, "A survey on scalable LoRaWAN for massive IoT: Recent advances, potentials, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 3, pp. 1841–1876, 2nd Quart., 2023.

[3] J. Haxhibeqiri, E. De Poorter, I. Moerman, and J. Hoebeke, "A survey of LoRaWAN for IoT: From technology to application," *Sensors*, vol. 18, no. 11, p. 3995, Nov. 2018.

[4] (2020). *RP002-1.0.1 LoRaWAN Regional Parameters*. [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/rp_2-1.0.1.pdf

[5] A. Osorio, M. Calle, J. D. Soto, and J. E. Candelo-Becerra, "Routing in LoRaWAN: Overview and challenges," *IEEE Commun. Mag.*, vol. 58, no. 6, pp. 72–76, Jun. 2020.

[6] J. R. Cotrim and J. H. Kleinschmidt, "LoRaWAN mesh networks: A review and classification of multihop communication," *Sensors*, vol. 20, no. 15, p. 4273, Jul. 2020.

[7] R. B. Sørensen, D. M. Kim, J. J. Nielsen, and P. Popovski, "Analysis of latency and MAC-layer performance for class a LoRaWAN," *IEEE Wireless Commun. Lett.*, vol. 6, no. 5, pp. 566–569, Oct. 2017.

[8] N. Benkahla, H. Tounsi, Y.-Q. Song, and M. Frikha, "Enhanced dynamic duty cycle in LoRaWAN network," in *Ad-hoc, Mobile, and Wireless Networks*. Cham, Switzerland: Springer, 2018, pp. 147–162.

[9] T. Deng, J. Zhu, and Z. Nie, "An improved LoRaWAN protocol based on adaptive duty cycle," in *Proc. IEEE 3rd Inf. Technol. Mechatronics Eng. Conf. (ITOEC)*, Oct. 2017, pp. 1122–1125.

[10] Md. Rakibul Islam, Md. Bokhtiar-Al-Zami, B. Paul, R. Palit, J.-C. Grégoire, and S. Islam, "Performance evaluation of multi-hop LoRaWAN," *IEEE Access*, vol. 11, pp. 50929–50945, 2023.

[11] J. R. Cotrim and J. H. Kleinschmidt, "An analytical model for multihop LoRaWAN networks," *Internet Things*, vol. 22, Jul. 2023, Art. no. 100807.

[12] (2018). *LoRaWAN 1.0.3 Specification*. [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/lorawan1.0.3.pdf

[13] R. P. Centelles, F. Freitag, R. Meseguer, and L. Navarro, "Beyond the star of stars: An introduction to multihop and mesh for LoRa and LoRaWAN," *IEEE Pervasive Comput.*, vol. 20, no. 2, pp. 63–72, Apr. 2021.

[14] T. Fedullo, A. Mahmood, F. Tramarin, A. Morato, M. Gidlund, and L. Rovati, "Exploiting hybrid medium access control and relaying strategies to overcome duty-cycle limitations in LoRa-based sensor networks," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (I2MTC)*, May 2023, pp. 1–6.

[15] H. Shida, A. Kaburaki, and K. Adachi, "Energy-efficient multi-hop LoRaWAN with transmission and synchronization control," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Feb. 2024, pp. 1104–1108.

[16] (2020). *LoRaWAN Relay Specification*. [Online]. Available: https://resources.lora-alliance.org/technical-specifications/ts011-1-0-0-relay

[17] H. Qin, W. Chen, N. Li, T. Wang, Y. Deng, G. Yang, and Y. Peng, "CDR: Bandwidth-efficient cross-interface downlink relay scheme for low-power multihop wireless networks," *IEEE Internet Things J.*, vol. 11, no. 9, pp. 16212–16226, May 2024.

[18] P. Tian, C. A. Boano, X. Ma, and J. Wei, "LoRaHop: Multihop support for LoRaWAN uplink and downlink messaging," *IEEE Internet Things J.*, vol. 10, no. 17, pp. 15376–15392, Sep. 2023.

[19] M. A. Mojamed. (2023). *LTM-LoRawan: A Multi-Hop Communication System for LoRawan*. [Online]. Available: https://www.mdpi.com/2079-9292/12/20/4225

[20] Semtech. (2015). *LoRa-Modulation Basics Semtech*.

[21] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo, "On the coverage of LPWANs: Range evaluation and channel attenuation model for LoRa technology," in *Proc. 14th Int. Conf. ITS Telecommun. (ITST)*, Dec. 2015, pp. 55–59.

[22] R. Porat, S. Yong, and K. Doppler. (2015). *IEEE P802.11 Wireless LANs TGah Channel Model Proposed Text*. Accessed: Jun. 25, 2021. [Online]. Available: https://www.ieee802.org/11/Reports/tgah_update.htm

[23] (2023). *ChirpStack Open-Source LoRaWAN Network Server*. Accessed: Jul. 20, 2023. [Online]. Available: https://www.chirpstack.io/

JEFERSON RODRIGUES COTRIM received the B.S. degree in electrical engineering from Faculdade de Engenharia Industrial (FEI), Brazil, in 2008, and the master's and Ph.D. degrees in information engineering from the Federal University of ABC, Brazil, in 2017 and 2021, respectively. He is currently a Postdoctoral Researcher with the Computer and Digital Systems Engineering Department, Escola Politécnica da Universidade de São Paulo (EPUSP). His research interests include the IoT, routing protocols, and LPWANs.

CÍNTIA BORGES MARGI (Senior Member, IEEE) received the Electrical Engineering and M.S. degrees from Escola Politécnica da Universidade de São Paulo (EPUSP), São Paulo, Brazil, in 1997 and 2000, respectively, the Ph.D. degree in computer engineering from the University of California at Santa Cruz, Santa Cruz, CA, USA, in 2006, and the Habilitation degree (Livre Docência) in computer networks from Universidade de São Paulo, São Paulo, in 2015.

She joined the Department as an Assistant Professor, in 2010, and then was an Associate Professor, from 2015 to July 2023. Before that, from 2007 to 2010, she was an Assistant Professor with Escola de Artes, Ciências e Humanidades da Universidade de São Paulo (EACH-USP). She is currently a Full Professor with the Computer and Digital Systems Engineering Department, EPUSP. She has published more than 100 papers and coordinated research projects funded by The São Paulo Research Foundation (FAPESP), São Paulo. Her research interests include wireless sensor networks and the Internet of Things, security, and software-defined networking.