



# IoT scheduling for higher throughput and lower transmission power

Husam Rajab<sup>1</sup> · Tibor Cinkler<sup>1</sup> · Taoufik Bouguera<sup>2</sup>

Published online: 23 March 2020  
© The Author(s) 2020

## Abstract

The massive increase in the Internet of Things (IoT) has brought a diverse long-range, low-power, and low bit-rate wireless network technologies. The LoRa a low-power wide area network (LPWAN) gained popularity as radio technology for the realization of many IoT applications. LoRa is typically employed together with LoRaWAN MAC protocol and operates in the license-free ISM bands. LoRa networks have an issue with scalability when the number of end nodes connected to one network is larger than the shared number of channels, that causes a collision and packets loss through receiving a wide range of different message sizes from various application. In this paper, we describe an accurate and efficient way confirmed by simulation to calculate the probability of collision rate and packet loss in LPWANs under various circumstances. Moreover, based on the LoRaWAN specification, we consider a dense network deployment of IoT devices. In the event of collisions, our proposed algorithms is classified to two approaches. Firstly a time scheduling algorithm is proposed for LoRaWAN networks that consist of devices supporting LoRaWAN class C mode for synchronization in the beginning between the gateway and the end nodes. Afterwards these devices switch to class A to significantly decrease the collision and to enhance scalability by assigning a Guard Time to each end node. Gateway acknowledgment (ACK) messages to the end nodes are used through class C. Secondly, we also propose a distance spreading factor algorithm according to the distance of the end nodes from the gateway to reduce the probability of collision. Furthermore, many of these devices are battery powered, therefore low power consumption is required.

**Keywords** IoT · LPWAN · LoRa · LoRaWAN · Scheduling · Collision

## 1 Introduction

The “Internet of Things” or IoT comprise a group of devices connected to the Internet which are usually characterized by their low complexity, and their low power consumption, such as battery-powered sensors connected to the Internet [1]. These Internet of Things (IoT) devices

produce new requirements and challenges for wireless networks [2]. Recently, the current measured value of data has gained most attention in the industrial IoT [3]. These “things” are transmitting a small amount of data such as humidity, temperature, location, etc. The majority of those devices will use wireless technology to transmit data and connect to the network backbone, that requires a high scalability wireless network that can serve massive numbers of end nodes [4]. Wide area networks (WANs) play an essential role in connecting the primary facilitation through Low Power Wide Area Networks (LPWAN) is enabling power-efficient wireless communication over long distances. One of the LPWAN technologies that have reached great interest in recent studies is LoRaWAN. LoRa is one of the promising and significant in many Internet of Things application scenarios, such as smart grids, smart cities, geological monitoring [5]. Figure 1 illustrates the comparison between LPWAN technologies [5].

LoRa is a non-cellular modulation technology for LoRaWAN. LoRaWAN is the standard protocol for WAN

---

✉ Husam Rajab  
husamrajab@tmit.bme.hu

Tibor Cinkler  
cinkler@tmit.bme.hu

Taoufik Bouguera  
bouguerra.taoufik@gmail.com

<sup>1</sup> Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary

<sup>2</sup> University of Bretagne Loire, Polytech Nantes, IETR, 44300 Nantes, France

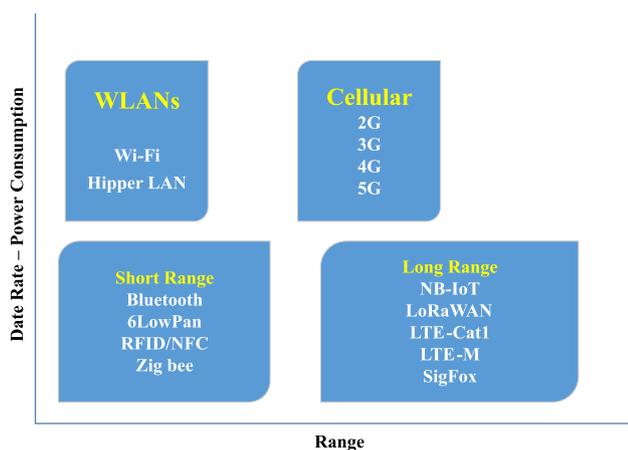


Fig. 1 Low power wide area technologies

communication. LoRa describes as good radio network for IoT solutions and has more reliable link budgets than other comparable radio technologies. LoRaWAN is an open standard protocol which defines the MAC layer for LPWAN technology designed by the LoRa Alliance. LoRaWAN uses an unlicensed spectrum, also it can be used by non-mobile operator customers to implement solutions. It should be noted that LoRaWAN networks interfere when more than one is operated in an area. LoRaWAN is essentially designed and optimized for up links where the end devices broadcast data to the gateway using periodic and aperiodic communication or due to a specified radio duty-cycle. However, this design does not support large data communication, such as video, photos. [6]. However, this technology has limitations and challenges that need to be clearly understood to avoid disappointment. The nodes have a high probability of losing packets due to collisions, mainly when high spreading factors are used that increase time on air [7]. In addition, the capture effect performance of LoRa transmission depends on transmission timings and power determined by empirical measurements. The capture effect occurs when two signals are present at the receiver, and the stronger signal represses the weaker signal [8]. The selection of communication technology and LoRa parameters are used for transmissions to assume the probability of packet collisions and its scalability in [9]. Furthermore, the range of values for the spreading factor of LoRa will affect the coverage area, and provide high density device deployments. Moreover, the probability of collisions is also influenced by traffic features, mainly the transmission periodicity, and payload size. The more increase in the payload size is required the more extended the time on-air and the channel occupancy will be.

In this paper, an accurate and efficient way is proposed to calculate the probability of collision probability and

packet loss in LoRaWAN under various circumstances. Moreover, based on the LoRaWAN specification, we propose scheduling algorithm to significantly decrease the collision probability and to enhance the network scalability. Firstly, the proposed time scheduling algorithm is coordinated by the gateway, based on the idea that the most significant part of the data. For example, in agriculture, the data is sent periodically. In this case, the gateway adds guard time to each packet through the synchronization process while using class C after a collision occurred. Afterwards, the end nodes switched to class A for data transmission between end nodes and the gateway. Secondly, the distance spreading factor scheduling algorithm is proposed. The gateway decides which is the appropriate SF value to set for end nodes according to the distance between the gateway and the end nodes location.

The remainder of the paper is organized as follows: Related work is presented in Sect. 2. In Sect. 3, we give an overview of LoRa and LoRaWAN and discuss the main LoRa parameters. Section 4 states the problem statement for LoRaWAN class A, followed by a detailed description of our proposed work and novel scheduling algorithms in Sect. 5. Section 6 presents the mathematical analysis. We present the simulation results for the proposed algorithms in Sect. 7. Finally, Sect. 8 presents the conclusion and future work.

## 2 Related work

In this section, we summarize state-of-the-art researches related to our proposed algorithms. Significant contributions have been done in our community investigating the collision, packet loss, and the scalability for long range networks. Bor et al. [10] present a simulator called LoRaSim used to analyze and estimate the scalability of LoRa networks. The authors examine the range of communication options (bandwidth, spreading factor, carrier frequency, and coding rate) for a transmitter. Furthermore, they investigate the collision avoidance scenarios and the possibility of implementing a carrier activity detection mechanism. The achieved results confirm that to keep the Data Extraction Rate (DER) above 0.9, only 120 users are supported per antenna using standard LoRa settings. DER defined as the ratio of successfully received data by LoRa Gateway to transmitted data by the LoRa nodes within an estimation window. Georgiou and Raza [8] consider a single gateway LoRa network and propose a stochastic geometry framework to model their performance. Their analysis considered a single bandwidth frequency for all the nodes. The estimation of the coverage probabilities and the outage is based on the signal to noise ratio and on the co-spreading sequence interference. Their obtained result

shows that the probability range decreases exponentially as the number of end-devices increases because of interfering signals using the same spreading factor. Abeelee et al. [11] measure and analyze the scalability performance of LoRaWAN through a LoRa ns-3 module. Their work explains the harmful effect of the downstream traffic on the delivery ratio of the upstream traffic. The authors showed that the increase of the gateway density could improve these effects, as stringent duty cycle requirements for gateways continue to limit downstream opportunities. Furthermore, with the obtained simulation results, the authors show that LoRaWAN can send six times more traffic compared to pure Aloha in a single-cell. By using 125-kHz channel bandwidth in the LoRaWAN network with the same number of end devices to avoid collision between the transmitted packets. In [12], the authors show a study for the uplink traffic in LoRaWAN where the packet delivery ratio decreases exponentially with increasing the number of end nodes in the network. Mikhaylov et al. [8] present an estimation of the throughput of LoRa technology, considering the time on air of the packet transmission. Accordingly, it is possible to determine the maximum number of end nodes that can communicate with the Gateway. Furthermore, the main technical difference is well explained in [9]. The authors evaluated the main difference between LoRa and Sigfox by simulation. Voigt et al. [13] show that the impact of two methods to reduce the inter-network interference and enhance the reception rate. They consider the usage of directional antennas and the usage of multiple base stations. Through simulations, the authors determine that the multiple base stations and having end nodes with Omni-directional antennas. The data extraction rate is 56% corresponded to only 32% when directional antennas are used. Simulator tool extended to LoRaSimtool to support for LoRaWAN MAC protocol in [14], which utilize bi-directional communication. Furthermore, the authors present how downlink traffic considerably affects the uplink throughput. Optimizing the power and avoiding collisions allocating problems for using different channels, also presenting the effect of the spread-spectrum sensitivity is shown in [15]. In [16], the authors present low-power distributed queuing (LPDQ), with highly competent and low power MAC protocol. LPDQ can self-schedule data transmissions; furthermore, the authors propose a design for implementing the LPDQ by using low-power commercial radio transceivers. However, there is extensive research to extend the LPDQ to multi-hop networks, which can enable a further range of wireless networks and improve the scalability and maintain their performance. LoRaWAN channel access limits presented in [17], the authors are focusing on the evaluation of the channel access performance as the most critical point for massive machine type communication. So far, a few works have

been proposed to enhance the scalability issues, particularly by improving the spreading factor assignment to nodes [18, 19]. In [18], the authors present two novel approaches for allocating wireless resources, through SF assignments, in LoRaWAN systems. The idea of the authors is to consolidate the SF orthogonality and radio range to enhance the number of end nodes transmitting in the network. The future work for this research can be applied to the analysis of multi-gateway scenarios. In [19], the authors considered the Adaptive Data Rate (ADR) to evaluate the LoRa networks. Also, the authors develop an end to end LoRa simulator to study the energy efficiency and the reliability for LoRa networks. Ultimately, the authors showed that the results of ADR are useful when the variance of the channel is zero or very low. To the best of our knowledge, there are no relevant works that contribute information on this determination, nor studies the impact to select the best LoRaWAN throughput.

In this paper, unlike other works, we thoroughly study LoRaWAN performance by presenting the probability of collision and packet loss through both times and spreading factor parameter. We are improving the network scalability and increase the LoRaWAN networks throughput, which is the crucial contribution of this work. Scheduling algorithms are proposed to maintain a stable network structure while decreasing the collision probability and reducing the amount of data retransmitted by considering a synchronization ACK message by employing Class-C and adding the guard time to each packet. The main contribution of this paper is follow as:

1. Mathematical close form expression of the probability of collision and packet loss.
2. *Time scheduling algorithm* is proposed. Through employing devices supporting LoRaWAN class C mode for synchronization. Afterward, these devices switch to class A to decrease the collision significantly and to enhance scalability by assigning a Guard Time to each end node.
3. *Distance spreading factor algorithm* is proposed. According to the distance of the end nodes from the gateway, and the time on-air values, we assign the best SF value for each packet. That leads to a decreased probability of collision and higher throughput.

### 3 LoRa and LoRaWAN overview

In this section, we overview the main LoRa and LoRaWAN properties and define the basic parameters required for the implementation of our LoRaWAN channel simulations.

### 3.1 Overview

Many radio technologies co-exist to deploy LPWA networks. Onward with the Ultra-Narrowband (UNB) protocol, LoRa and its MAC layer implementation—LoRaWAN—is one of the LPWA network solutions. Recently they are gaining the most considerable traction to sustain IoT applications and services. This modulation operates in the ISM bands (1.Industrial, 2.Scientific, 3.Medical) through the CSS (Chirp Spread Spectrum) feature, also supports bidirectional communication through spreading the narrowband signal over a larger frequency band. There is a significant differentiation between LoRa (Long Range) and LoRaWAN (Long Range Wide Area Network), which are usually both referred to as LoRa, LoRa Architecture shown in Fig. 2.

More precisely, LoRa and LoRaWAN refer to the physical and the MAC layer of the LoRa protocol stack, respectively. LoRaWAN (Long Range Wide Area Network) is the MAC layer protocol of the protocol stack. LoRaWAN network architecture determines a star-of-stars topology of end-devices, gateways, and of central network server in the back-end. The LoRa Alliance defines the protocol. It is based on LoRa modulation, that exceeds the traditional modulation schemes, such as Frequency-Shift Keying (FSK). The LoRa performance is well suited to low-power and long-range transmissions discussed in [20]. Gateways are connecting end-nodes, and the network server, While end-nodes communicate with the gateways over LoRa or FSK. The backhaul connectivity between gateways and the network server is based on Internet Protocol (IP). It allows much higher capacity, e.g., via Ethernet or 3G [20]. End-devices are typically sensors or actuators with restricted data rate specifications, short messages, and

low traffic intensity. LoRaWAN enables three different classes (Class A, B, and C) with different schemes of scheduling for downlink communication shown in Fig. 3.

The LoRaWAN specification define three different operating classes to address various applications:

- **Class A** (Bi-directional end devices): End devices of class A allow for bi-directional communications. Each end device wakes up and transmits its messages to the gateway when the message is ready based on its preference, similar to the ALOHA based protocols. Each up-link transmission trigger two short down-link receive windows, After receive the ACK messages from the gateway the nodes go back to sleep mode. Class A is compatible with every LoRaWAN end device, and it provides the lowest power consumption.
- **Class B** (Bi-directional end devices with scheduled receive slots): End devices of class B allow for periodic receive slots. Class B devices open extra receive windows at scheduled times; the gateway transmits beacon frames transmitted at regular intervals. This allows the server to know when the end-devices are listening.
- **Class C** (Bi-directional end devices with maximal receive slots): End devices continuously keep open receive windows closed only while transmitting. Class C end devices consume the most power but have the lowest latency for the server to end device communications.

### 3.2 LoRa parameters

In this section, we introduce the operational parameters that are critical for the deployment of our LoRaWAN

Fig. 2 LoRa architecture

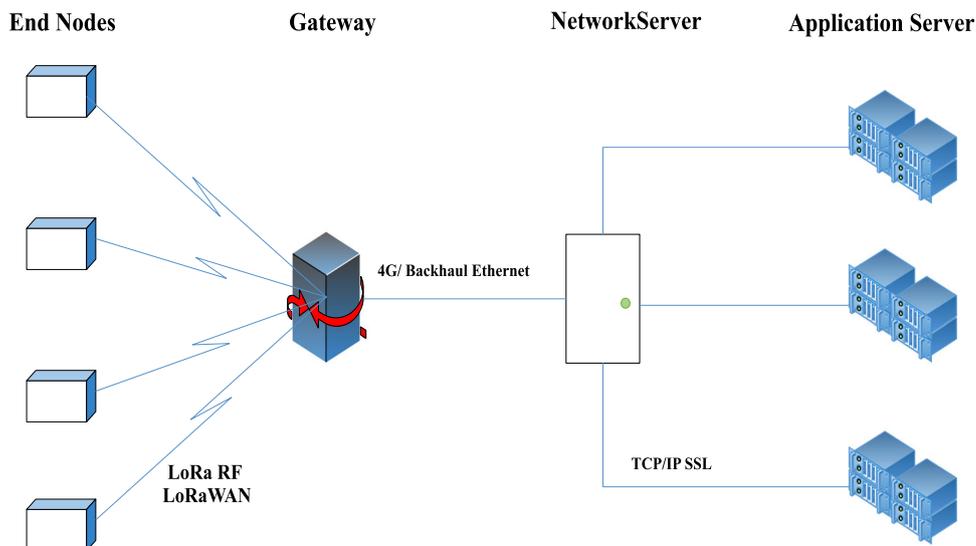
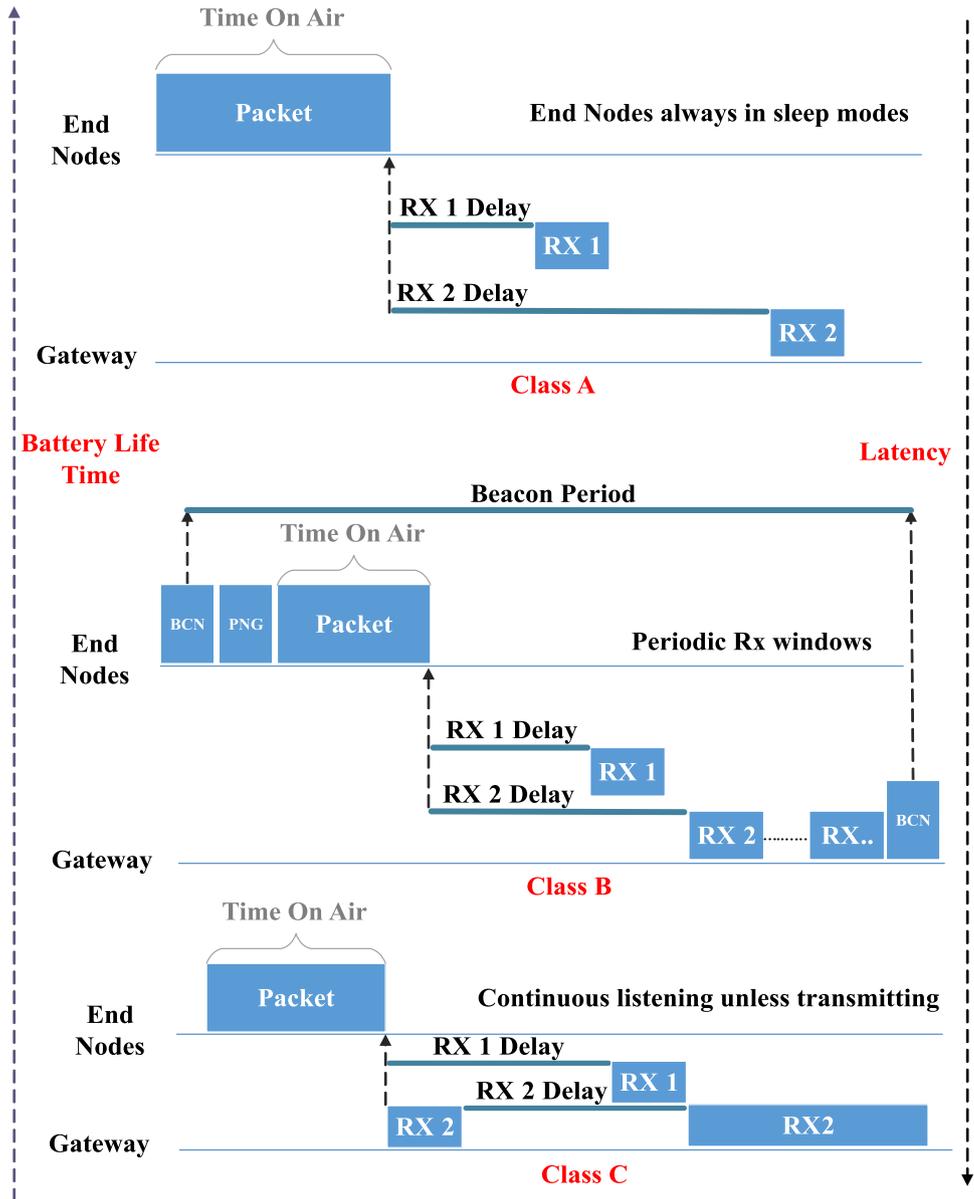


Fig. 3 LoRaWAN classes



channel simulator. Adjustment of these parameters permits tuning the tradeoffs between communication range and throughput. The throughput for radio transmission depends on the signal bandwidth (BW). The ISM stands for Industrial, Scientific, and Medical. These regulations allow the transceiver operating in an unlicensed band. Moreover, the end nodes do not require to obtain a government permit to use the spectrum. The standard width of the LoRaWAN channels in Europe is 125, 250, and 500 kHz. The LoRa modulation divides the data stream to one chip per second per Herz. Thus, the BW of the data transmission must be traded versus the frequency reduced the receiver sensitivity by introduced additional noise. On the contrary, decreasing the bandwidth results in higher receiver sensitivity,

although it increases the Time on Air investigated in [21, 22]. LoRa is a Chirp Spreading Spectrum (CSS) technology, and it is **Spreading Factor** (SF) defines the number of chirps per symbol. SF can be configured from 7 to 12. A high value of spreading factor drives to increase the signal to noise ratio (SNR) and hence to enhanced receiver sensitivity to an increased range of the signal. The SF values in LoRa is considered to be orthogonal. Consequently, simultaneous transmissions with various values of SF do not collide with each other and can be successfully decoded. Considering LoRa modulation uses  $2^{SF}$  chips per symbol, the chipping used by LoRa relates bit rate ( $R_b$ ) and chip rate ( $R_c$ ) by [23]:

$$R_c = 2^{SF} R_b \tag{1}$$

Accordingly, the pair of operating frequency and spreading factor composes a logical channel [20]. The symbol time ( $T_{sym}$ ) of LoRa is given by the formula [23]:

$$T_{sym} = \frac{2^{SF}}{BW} s \tag{2}$$

Thus, a higher value of spreading factors rise the symbol duration that leads to an increase of the time on-air; however, also that increases the receiver sensitivity. Therefore, larger SFs can increase the transmission range. Finally, the **Code Rate** (CR) determines the redundancy bits used by Lora’s Forward Error Correction (FEC) rate used by the LoRa modem to protect against interference. Possible values for the CR are in [1–4] corresponding to the coding rates of  $[\frac{4}{5}, \frac{4}{6}, \frac{4}{7}, \frac{4}{8}]$  resulting in the formula [24].

$$\text{Rate Code} = \frac{4}{4 + CR} \tag{3}$$

Reducing the Rate Code, respectively, increasing the CR results in lower Packet Error Rates (PER), but also increases the Time on Air. The higher the **CR**, the more robust the transmission. If there are too many interferences on the channel, then it is recommended increasing the value of CR.

### 4 Problem statements

Certain conditions cause collision such as arrival time overlap, spreading factor overlap, Coding Rate collision, etc. several conditions define whether the receiver can decode one or two simultaneous packets or nothing at all. The probability of packet collision also relies on the selection of LoRa parameters used for transmitting a packet. In this paper, we focus on two practical alternative methods to deal with interference. In the first case, packets are considered lost when the end nodes are transmitting the packets at the same time. This will be referred to as “Time Collision”. In the second case, packets are corrupted when the end nodes are using the *same spreading factor* for sending the packets at the same time. This will be referred to as “Spreading Factor Collision”. The collision probability model is analyzed in the following section. Both approaches aim at improving the collision probability, decrease the packet loss, and reach higher throughput. In Fig. 4 the algorithm show how to check cause of the collision probability at the end nodes. In this paper, we discuss the time collision and the spreading factor collision probability and propose two algorithms (M1 and M2) to decrease the rate of packet loss as explained in the following sections.

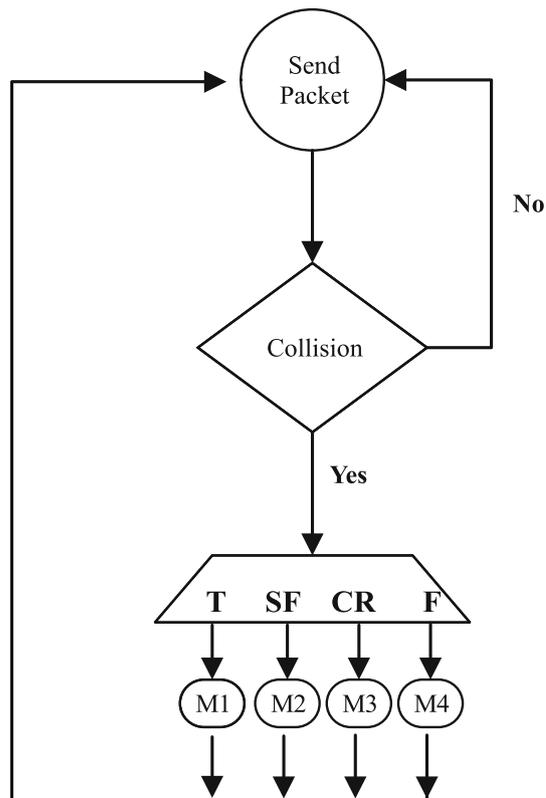


Fig. 4 Algorithm for checking packet collision

### 5 Proposed work

We assume that LoRa network consists of one gateway and many end nodes operating in the frequency of the main channels and two downlink channels.

Firstly, we consider a Class-C for the synchronization between the gateway and the end nodes to specify the Guard Time as it is possible the same procedure used for FOTA (Firmware Upgrade Over The Air). Then switch to Class-A if the collision happened. That will be consider when the end nodes transmit the packets at the same time. The end nodes generate random packets according to a Poisson distribution process with total intensity  $\lambda$  (that can be defined) and determine the mean time difference between two packets in milliseconds. The number of end devices operating in the channel can be configured in the simulation. The simulator generates packets for each end node, corresponding to the number of end nodes. The end nodes first send the data to the gateway as periodic sending through class-A, and then open up two reception windows (RX1 and RX2), providing the opportunity to the gateway for sending downlink acknowledgment message. In this transmission scheme, the gateway has the control on the end nodes through the concurrence using Class C, and this leads to increased latency of the downlink communications since it is required to wait for the next scheduled uplink

transmission. This impacts on several applications where both low-latency and low-power consumption are playing essential issues such as structural health or seismic activity monitoring. [23, 24]. If no collision happens for up-link, the end nodes are acknowledged by the gateway. After each up-link transmission, the end nodes wait for an ACK from a gateway.

In order to save energy through decreasing the collision probability and packet loss. Secondly, by proposing distance spreading factor algorithm. When the end nodes send packets using the same spreading factor will cause packet loss and high collision probability. For this purpose, our algorithm compares the distance values for each collide packets of end nodes from the gateway, to give each end nodes the appropriate SF value according to the distance between the end nodes and the gateway.

## 6 Mathematical analysis

In this section, the mathematical models for collision probability and Gateway scheduling algorithms are presented. To overcome the collision problem, we developed a mathematical model of the re-transmission process according to the Gateway scheduling algorithm. The added time to each end node after receiving the acknowledgment message from the gateway is called Guard Time (GT) to decrease the collision probability. The time between the start of the guard time and the moment when the packet is sent is set by the gateway using Class-C between the specified parameters to decrease the random collisions in the radio environment.

### A. Preliminaries

For downlink frames, the end nodes received the acknowledgment from the central server; the PHY frame structure is similar: all the fields are identical, except the Cyclic redundancy check (CRC) field, which is not present. This work is done to have the messages as short as possible, with minimum impact on any duty-cycle restrictions of the ISM bands used in [25].

It is possible to calculate the packet time-on-air using the formula given in [24]. A packet consists of a Preamble, followed by a header encoded with a code rate of  $\frac{4}{8}$  and protected by a CRC. The code rate encodes the Payload, and it is CRC. The preamble duration  $T_{preamble}$  depends on the number of preamble  $n_{preamble}$  and symbols duration  $T_{sym}$  configured and can be measured by the following formula:

$$T_{preamble} = (n_{preamble} + 4.25) T_{sym} \quad (4)$$

The number of symbols of the packet payload  $N_{sym_{payload}}$  is calculated by [22]:

$$N_{sym_{payload}} = 8 + \max\left(\text{ceil}\left(\frac{[8PL - 4SF + 28 + 16CRC - 20H]}{4(SF - 2DE)}\right), 0\right)(CR + 4), 0 \quad (5)$$

where:

- *PL*: Number of Payload bytes.
- *SF*: Spreading Factor 7 – 12.
- *H*: Header: 0 = enabled, 1 = no header.
- *DE*: Low Data Rate Optimization: 1 = enabled, 0 = disabled.
- *CR*: Coding rate.
- *CRC*: Cyclic Redundancy Check.

To indicate fixed-length packets for the header (H) can be set to 1. In this illustration, the header can be eliminated to decrease the length of the payload. To obtain more robust communication to frequency variations. The low data rate optimization can further help [24]. This selection is not explicitly documented in the manuals and guides provided by Semtech. Still, the formula for the payload symbol calculation exposes that decreases the number of transmitted bits per symbol decrease by a factor of 2. Furthermore, to calculate the total duration of the packet by summing up the preamble and the payload duration:

$$T_{packet} = T_{preamble} + T_{payload} \quad (6)$$

$$T_{payload} = T_{preamble} + N_{sym_{payload}} \quad (7)$$

All formulas of this section have been adopted from [25], for example, how to calculate the Time on Air using the LoRa calculator provided by Semtech. Moreover, the authors in [20] explain the physical parameters of the LoRa modulation and derive a modified formula to compute the payload symbol number.

Furthermore, the number of end nodes operating in the channel can be configured for the simulation. The simulator generates packets for each end node. A random packet length is uniformly distributed, and it is randomly assigned to each packet. The arrival time of a packet is Poisson distributed but has to respect the duty cycle. The algorithm for meeting the regulatory duty cycle constraint was implemented as specified by the LoRaWAN Specification 1.0. The formula given in the specification determines the backoff time ( $T_{off}$ ):

$$T_{off} = \frac{TimeOnAir}{DutyCycle} - TimeOnAir \quad (8)$$

After sending a packet, end nodes wait at least for  $T_{off}$ , before they send the next packet. If the arrival time of the next packet generated by the Poisson process has a silent period lower than  $T_{off}$ , the start time of the next packet is set to previous arrival time +  $T_{off}$ . The latest LoRaWAN specification gives the implementation of the algorithm

guaranteeing the compliance to the duty cycle restriction open to the manufacturer. The second step is to find the colliding packets and finds the collision probability — all packets are generated by the end nodes randomly. Collisions are assumed by determining if the duration between a packet arrival and end overlaps within the period of another packet arrival. Also, collision happens when there is overlapping in Spreading Factor when two or more packets used the same SF value since the SF is pseudo-orthogonal in LoRa. Even when the same channel and the same bandwidth are used, the receiver may decode if it receives different SFs.

To express the base-band signal transmitted by the end nodes is given as follows:

$$t_{r_n}(t) = \sum_{k \in S_i} e^{j2\pi f_c \cdot k(t - kT_s) + j\phi} \tag{9}$$

where:

- $f_c, k$ : the transmitted chirp at time  $kT_s$ .
- $S_i$ : the transmitted symbols inside the packet  $p$ .
- $k$ : the  $T_s$  size.
- $\phi$ : the initial phase.

The sleep period for the LoRa node will be at least equal to  $\frac{KT_s}{dc} - KT_s$ , where  $dc$  is the duty cycle. According to the complex envelop determination of (9), we consider a system that is composed of one LoRa gateway and N LoRa end nodes. The end nodes  $n^{th}$  transmit packets  $P_{s_i}$ , and each packet sent by the end nodes with time on-air  $T_a$ . We note  $a_n$  the channel coefficient associated with the end nodes  $n^{th}$ .

Considering the received signal at the LoRa gateway  $R(t)$  is:

$$R(t) = \sum_{n=1}^N a_n \sum_{p=1}^{P_n} t_{r_n}(t - T_b^n) \tag{10}$$

Let  $T_b^n$  be the start time of the LoRa packet  $p$ . According to this, we define:

$$T_b^n = (p - 1)(T_a + T_{aff}) + \sum_{s=0}^p T_s \tag{11}$$

$$T_b^n = (p - 1) \frac{T_a}{dc} + \sum_{s=0}^p T_s \tag{12}$$

where  $T_s$  is a stationary random process with uniformly distributed entries, also it is used to model the various end node constraints in terms of communications with the gateway. We assumed that  $T_s$  is uniformly distributed in the interval  $[T_{min}, T_{max}]$ . This supposition is explained due to the massive number of IoT applications.

To calculate the probability of collision  $PC_{n_1(p)}^{n_2(p)}$  between  $n_1(p)$  and  $n_2(p)$  at the LoRaWAN gateway, given that

$n_1(p)$  represent the node of interest. If  $c_{n_1(p)}^{n_2(p)}$  indicates the collision, based on (12), the collision case is defined as follows:

$$C_{n_1(p)}^{n_2(p)} = T_{n_2}(p) \in \Omega_C = [T_{n_1}(p) - T_{a_2}, T_{n_1}(p) + T_{a_1}] \tag{13}$$

The collision will happen between  $n_1(p)$  and  $n_2(p)$  if  $T_{n_2}(p) \in \Omega_C$ . Then we represent the probability of collision as follows:

$$P(C_{n_1(p)}^{n_2(p)}) = Pr(T_{n_2}(p) \in \Omega_C) \tag{14}$$

where the random variable  $T_{n_2}(p)$  is the summation of multiple uniform random variables see (12).

All packets are sorted by arrival time to find the collided packets. Packet collisions are assumed by deciding if the duration between a packet arrival and the packet end overlaps with the duration of any part of any packets. The following pseudo-code shows the condition for a packet collision:

$$\begin{aligned} & (Other\_End \geq Other\_Arrival \leq Arrival) \\ & (End \geq Other\_Arrival \leq Other\_End) \end{aligned}$$

To decrease the collision probability we define Guard Time  $T_{Guard}$ . The gateway (downlink) Ack messages to the end nodes are used to assign the value of the Guard Time utilizing Class-C. According to the down-link message through the gateway, each end nodes switch and re-transmit the collided packets with a specified delay called Guard Time in the interval  $[0, T_{max}]$  using Class-A.

Therefore, the channel consists not only of the new packets but also of the previously collided packets, and that increases the mean traffic generated. Also, Guard Time calculation depends on the maximal synchronization period in the network and the end nodes accuracy. In our consideration, we did not assume the processing time that is essential for each packet to be processed by the end nodes before transmitting.

After adding the  $T_{Guard}$ , we modify equation (12) to have the new start time packet  $T_b^{new}$  as follows:

$$T_b^{new} = (p - 1) \frac{T_a}{dc} + \sum_{s=0}^p T_s + T_{Guard} \tag{15}$$

Moreover, to calculate the collision probability according to the new value of the start time packet after adding the Guard Time, we have:

$$C_{n_1(p)}^{n_2(p)} = T_{n_2}(p) \in \Omega_C = [T_b^{new} - T_{a_2}, T_b^{new} + T_{a_1}] \tag{16}$$

So

$$\Omega = [T_b^{new} - T_{a2} > T_b^{new} + T_{a1}] \tag{17}$$

With an increasing number of nodes per gateway, the number of simultaneous LoRa transmissions will increase as well, increasing the probability of collisions as a result of this affecting throughput.

### 7 Simulation results

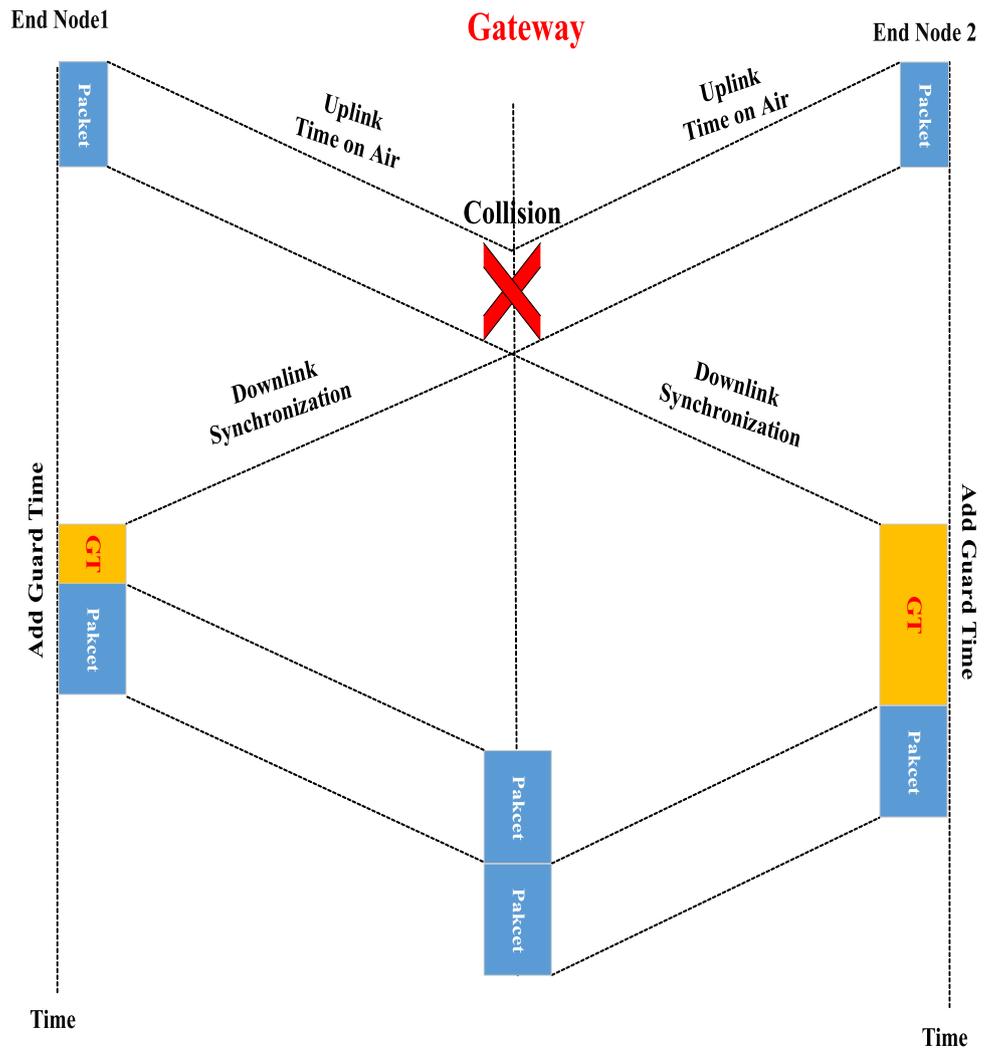
In this section, to validate our approach, we consider the system composed of one gateway that can establish connections with N end nodes using the same spreading factor. Our implementation for the proposed scheduling algorithms, the transmission through the gateway downlink message for end nodes is presented in two approach. The LoRa parameters used:  $fc= 868\text{MHz}$ ,  $BW= 125\text{kHz}$ , and  $CR= 1$ .

Firstly, the time scheduling algorithm is proposed. The time is divided into time slots that can accommodate the

communication in the network. The gateway is responsible for adding the guard time to each packet together with the acknowledgment message via class-C synchronicity with the end nodes. Then switch to Class-A if there is no collision happened and for periodicity. The implementation architecture for this scenario is shown in Fig. 5. Based on the time slot in the ACK message, the gateway determines the guard time after the packet collision happened from different end nodes. From that moment, after adding the guard time to each packet, the end nodes start increasing the time slot for each packet before transmitting arranged by the gateway.

Secondly, the distance spreading factor algorithm is proposed. Each end nodes randomly selects SF value from the allowed spreading factor from 7 to 12. A collision occurs when two end nodes send packets using the same spreading factor at the same time. Our proposed algorithm for nodes that are located close to the gateway assigns to them the lowest SF value. By comparing the distances for end nodes, how far there location from the gateway, we

Fig. 5 Time scheduling algorithm



assign the appropriate SF value for each end node. The determination of the distance according to the value of the received time on air. The details of the proposed algorithm for spreading factor scheduling collision is given in Algorithm 1, Fig. 6.

End nodes are geographically dispersed in the network. Simulations are done by Matlab using different number of end nodes (5,10,20,50,100) for each scenario. In the first scenario, we simulated with a slightly increased number of end nodes. Figure 7 shows the evaluation of the Time Scheduling algorithm. Thus, the assumed number of collided packets is 2 when we applied it for five end nodes. After using the time scheduling algorithm, the simulation results show improvements in terms of decreasing the probability of collision by 18%. To be able to evaluate the proposed solution under various conditions, Fig. 8 presents the collision parameters when the number of end nodes increases from 5 to 10. The results showed that increasing

**Algorithm 1** D-SF-Scheduling collision

**Input** D: Distance , SF: Spreading Factor , T: Time on Air

Begin : Generate random packets from different nodes

**For** each  $kk \in K$  and  $ff \in F$  **Do**

Start from the first packet from the first node

Search D with value of SF

Compare D

**Description :**

**IF**  $100m \leq D \leq 5000m$

**Then**  $SF = 7$  or  $8$

**Else If**  $5000m \leq D \leq 10000m$

**Then**  $SF = 9$  or  $10$

**Else**  $SF = 11$  or  $12$

**END**

Fig. 6 Distance spreading factor scheduling algorithm

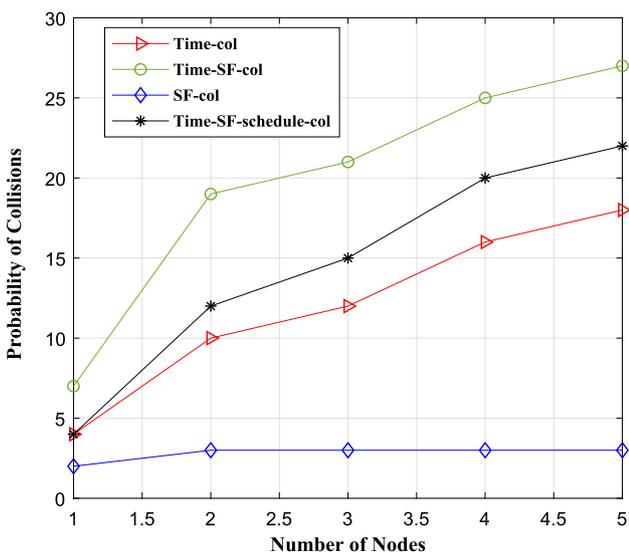


Fig. 7 Probability of at least one collision for 5 EN

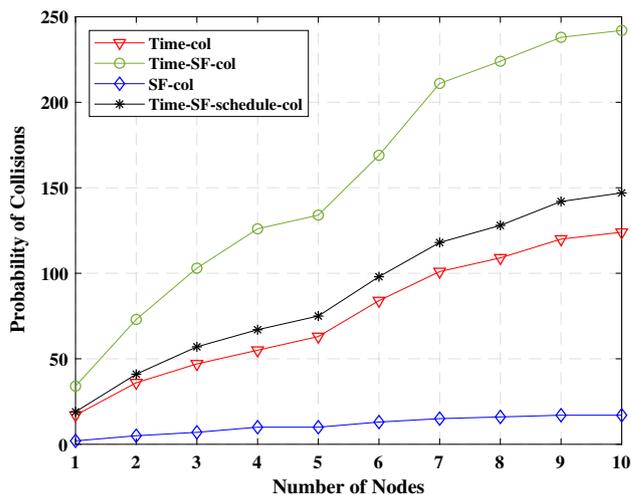


Fig. 8 Probability of at least one collision for 10 EN

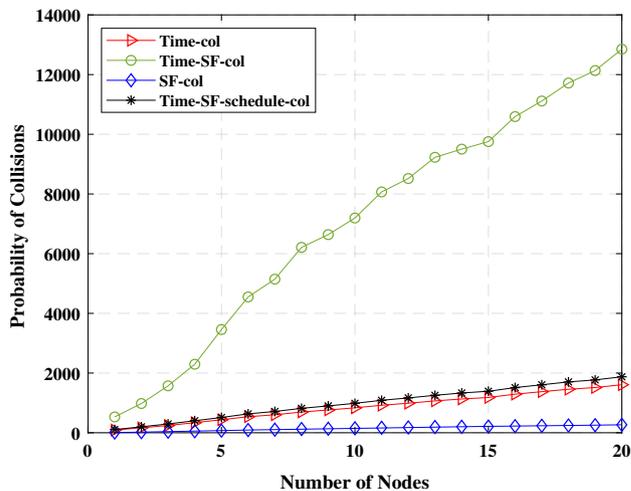


Fig. 9 Probability of at least one collision for 20 EN

the collision probability due to the rise of the number of end nodes. From our obtained results, we can realize that after applying the time scheduling algorithm, the improvement of decreasing the probability of collision is 39.85%.

Moreover, to be able to evaluate our proposed algorithm, the results in Figs. 9 and 10 confirmed that the consumed energy increased with the increase of the number of colliding packets and the value of SF. By assuming that the number of end nodes increases from 5 and 10 to 20, and to 50, respectively. This leads to the efficient use of the proposed time scheduling algorithm. This minimizes the consumed energy, as well as the collision probability about 85.42 %, and 98.95%, respectively for 20 and 50 nodes.

This claim can support the results in Table 1. The table shows the various time and number of end nodes and

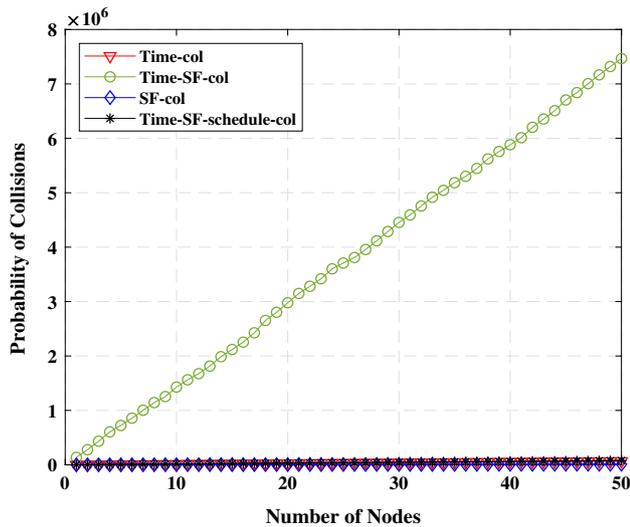


Fig. 10 Probability of at least one collision for 50 EN

used in the simulation. Also, it shows that the improvement of minimizing the collision probability through increasing the number of end nodes in the network.

In the second scenario, we are proposed a solution to overcome the spreading factor collision when two or more end nodes using the same spreading value to transmit packets. By comparing the distance value how far it is located from the gateway for each end node to the time on air values. However, the experiments were performed considering the impact on the throughput by applying the distance spreading factor algorithm. The proposed algorithm is assumed to be periodic for each end node with a uniform distribution of transmission by end nodes in the first data period. End nodes are geographically distributed in the network. Figures 11 and 12 present the spreading factor collision, time spreading factor collision, and our proposed algorithm for two cases, namely for 5 and 10 end nodes. Simulation results show the improvement on the collision probability of the network after applying our proposed algorithm by 100%, 86.8% , respectively for 5 and 10 end nodes.

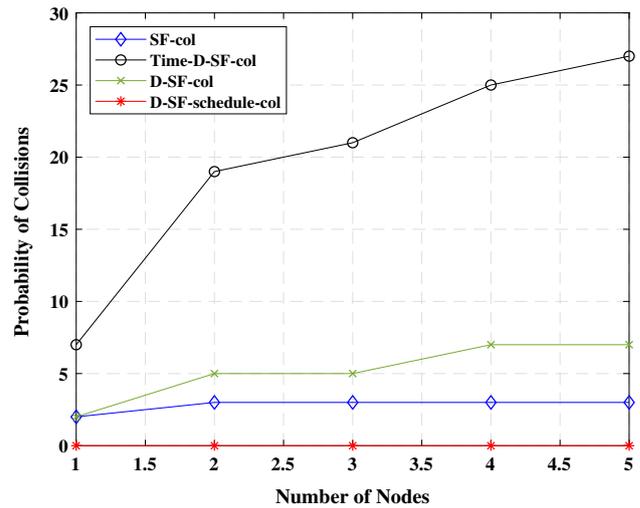


Fig. 11 D-SF scheduling algorithm for 5 end nodes

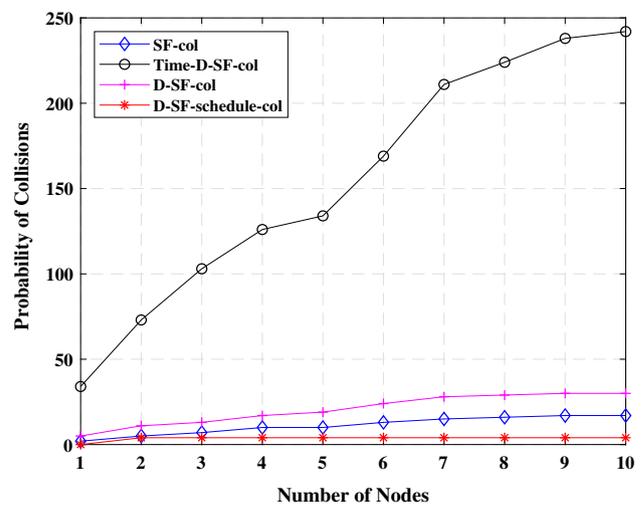


Fig. 12 D-SF scheduling algorithm for 10 end nodes

Furthermore, Figs. 13 and 14 shows obtained results after applying the distance spreading factor algorithm. Simulation runs using 20, 50 end nodes, respectively. It is known that the number of collisions increases with the number of end nodes raise. This increase leads to the rise in

Table 1 Time scheduling algorithm

No. of end nodes	Ratio of time collision when T = 0.1 s	Ratio of time scheduling algorithm	Improvement (%)	Ratio of time collision when T = 10 s	Ratio of time scheduling algorithm	Improvement (%)
5	0.20	0.16	18.1	0.09	0.10	15.3
10	0.45	0.27	39.8	0.22	0.17	22.7
20	7.15	0.95	86.6	0.98	0.23	84.7
50	592.2	6.16	98.9	603.6	6.24	98.9
100	1.8e+03	24.6	99.8	750	6.59	99.1

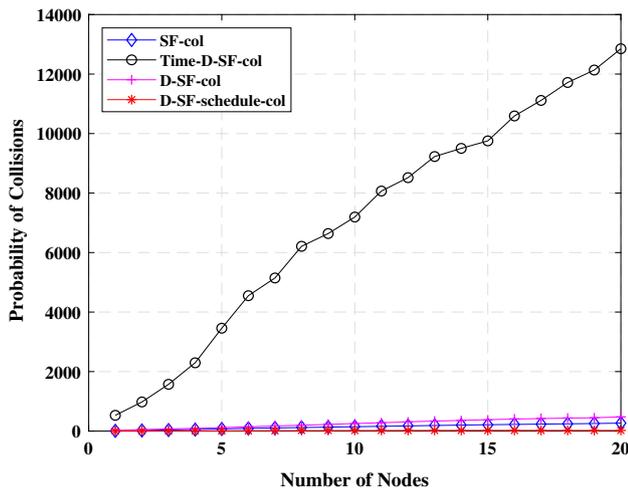


Fig. 13 D-SF scheduling algorithm for 20 end nodes

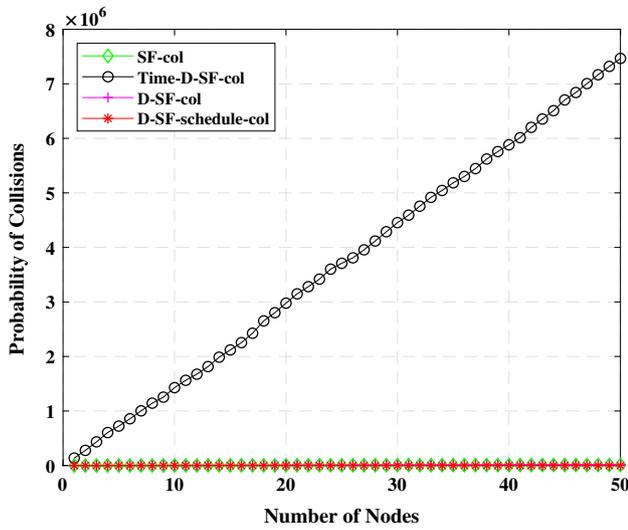


Fig. 14 D-SF scheduling algorithm for 50 end nodes

the number of frames re-transmit in LoRaWAN. By applying the proposed algorithm, the probability of spreading factor collision improved to 95.86% and 96.67%, respectively.

Thus, to further assess the performance improvement of our solution, we compare various number of end nodes. Table 2 describes a comparison that gives more insights into the improvements that the proposed distance spreading factor algorithm brings. It can be realized in case of increasing the number of end nodes. Also, changing the value of the guard time has no significant effect on the obtained result.

### 8 Conclusion

Collision is one of the main factors that negatively impact the LoRaWAN throughput. In this paper, we present an efficient way to express the probability of collision rate and packet loss mathematically. Furthermore, this work proposes a time scheduling algorithm for LoRaWAN Class C used for synchronization and Class A used for periodicity. Simulations are carried out using matlab. We model a network with a single gateway and n end-nodes. The obtained results show the significant decrease of the collision probability. Moreover, we studied the number of packets collided when the end nodes are using the same spreading factor for transmitting the data. The proposed algorithm relies on the distance value for each end node through how far it is located from the gateway, and it is time on air values. The proposed methods work particularly well in case of large loads and in case of a high number of End Nodes. The gateway ACK messages to the end nodes are used in the proposed algorithm to decrease the collision probability and improve the scalability. The guard time introduced by the propagation distance, and the periodicity

Table 2 Distance spreading factor scheduling algorithm

No. of end nodes	Ratio of SF collision when T = 0.1 s	Ratio of distance SF algorithm	Improvement (%)	Ratio of SF collision when T=10 s	Ratio of distance SF algorithm	Improvement (%)
5	0.05	0.00	100	0.01	0.00	100
10	0.05	0.007	86.8	0.04	0.001	95.8
20	0.20	0.008	95.8	0.23	0.009	95.9
50	1.37	0.04	96.6	1.35	0.04	96.4
100	5.50	0.14	97.4	1.52	0.035	97.6

of each packet is recorded. In our future work, we aim to propose an optimal algorithm using machine learning based on our scheduling algorithm to find the optimal gateway placement and optimize power consumption.

**Acknowledgements** Open access funding provided by Budapest University of Technology and Economics (BME). “This work was supported by the Ericsson - BME 5G joint research and cooperation project, partly funded by the National Research, Development and Innovation Office, Hungary with Project No. 2018-1.3.1-VKE-2018-00005.” Also, this research is supported by Tempus Public Foundation, Stipendium Hungaricum Scholarship Programme and High Speed Networks Lab, Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer Network*, 54, 2787–2805. <https://doi.org/10.1016/j.comnet.2010.05.010>.
- Maria, P., Mischa, D., Alfredo, G., Gianluca, R., & Thomas, E. (2016). Internet of things in the 5G era: Enablers, architecture, and business models. *IEEE Journal on Selected Areas in Communications*, 34(3), 510–527. <https://doi.org/10.1109/JSAC.2016.2525418>.
- Xiong, X., Zheng, K., Xu, R., Xiang, W., & Chatzimisios, P. (2015). Low power wide area machine-to-machine networks: Key techniques and prototype. *IEEE Communications Magazine*, 53, 64–71.
- Bardyn, J., Melly, T., Seller, O., & Sornin, N. (2016). IoT: The era of LPWAN is starting now. In *ESSCIRC conference 2016: 42nd European solid-state circuits conference, Lausanne* (pp. 25–30). <https://doi.org/10.1109/ESSCIRC.2016.7598235>.
- Filho, H. G. S., Filho, J. P., & Moreli, V. L. (2016). The adequacy of LoRaWAN on smart grids: A comparison with RF mesh technology. In *2016 IEEE international smart cities conference (ISC2), Trento* (pp. 1–6). <https://doi.org/10.1109/ISC2.2016.7580783>.
- Ducrot, N., Ray, D., Saadani, A., Hersent, O., Pop, G., & Remond, G. (2016). LoRa Device Developer Guide. Orange, connected objects and partnership. technical document. <https://developer.orange.com/wp-content/uploads/LoRa-Device-Developer-Guide-Orange.pdf>. Retrieved 23 Jan 2020
- Georgiou, O., & Raza, U. (2017). Low power wide area network analysis: Can LoRa scale? *IEEE Wireless Communications Letters*, 6(2), 162–165. <https://doi.org/10.1109/LWC.2016.2647247>.
- Mikhaylov, K., Petäjälä, J., & Janhunen, J. (2017). On LoRaWAN scalability: Empirical evaluation of susceptibility to inter-network interference. In *2017 European conference on networks and communications (EuCNC), Oulu* (pp. 1–6). <https://doi.org/10.1109/EuCNC.2017.7980757>.
- Reynders, B., Meert, W., & Pollin, S. (2016). Range and coexistence analysis of long range unlicensed communication. In *Proceedings of the IEEE ICT*. <https://doi.org/10.1109/ICT.2016.7500415>.
- Bor, M. C., Roedig, U., Voigt, T., & Alonso, J. M. (2016). 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, ser. MSWiM '16*. New York, NY, USA: ACM (pp. 59–67). <https://doi.org/10.1145/2988287.2989163>.
- Abeele, F. V. d., Haxhibeqiri, J., Moerman, I., & Hoebeke, J. (2017). Scalability analysis of large-scale lorawan networks in ns-3. arXiv preprint [arXiv:1705.05899](https://arxiv.org/abs/1705.05899). <https://doi.org/10.1109/JIOT.2017.2768498>.
- Haxhibeqiri, J., Van den Abeele, F., Moerman, I., & Hoebeke, J. (2017). LoRa scalability: A simulation model based on interference measurements. *Sensors*, <https://doi.org/10.3390/s17061193>.
- Voigt, T., Bor, M., Roedig, U., Alonso, J. (2016). Mitigating inter-network interference in LoRa networks. In *Proceedings of the 2017 international conference on embedded wireless systems and networks (EWSN '17)* (pp. 323–328). USA: Junction Publishing.
- Pop, A., Raza, U., Kulkarni, P., & Sooriyabandara, M. (2017). Does bidirectional traffic do more harm than good in LoRaWAN based LPWA networks? In *GLOBECOM 2017–2017 IEEE global communications conference, Singapore* (pp. 1–6). <https://doi.org/10.1109/GLOCOM.2017.8254509>.
- Reynders, B., Meert, W., & Pollin, S. (2017). Power and spreading factor control in low power wide area networks. In *Proceedings of the IEEE ICC*. <https://doi.org/10.1109/ICC.2017.7996380>.
- Peiro, P., Gallego, F., Zarate, J., Alonso, L., & Vilajosana, X. (2015). LPDQ: A self-scheduled TDMA MAC protocol for one-hop dynamic low-power wireless networks. *Pervasive and Mobile Computing*, <https://doi.org/10.1016/j.pmcj.2014.09.004>.
- Bankov, D., Khorov, E., Lyakhov, A. (2016). On the limits of LoRaWAN channel access. In *Proceedings of the 2016 international conference on engineering and telecommunication (EnT), Moscow, Russia* (pp. 10–14. 20; 29–30). <https://doi.org/10.1109/EnT.2016.011>.
- Cuomo, F., Campo, M., Caponi, A., Bianchi, G., Rossini, G., & Pisani, P. (2017). Explora: Extending the performance of lora by suitable spreading factor allocations. In *Wireless and mobile computing, networking and communications (WiMob)* (pp. 1–8), IEEE. <https://doi.org/10.1109/WiMOB.2017.8115779>.
- Slabicki, M., Premsankar, G., & Di Francesco, M. (2018). Adaptive configuration of lora networks for dense iot deployments. In *NOMS 2018-IEEE/IFIP network operations and management symposium*. <https://doi.org/10.1109/NOMS.2018.8406255>.
- Augustin, A., Yi, J., Clausen, T., Townsley, W. M. (2016). A study of LoRa: Long range and low power networks for the internet of things. In *Special issue enabling the move from wireless sensor networks to internet of things and cyber-physical systems 16, 1466, Sensors*. <https://doi.org/10.3390/s16091466>.
- Semtech Corporation. Lora modulation basics. <https://www.semtech.com/uploads/design-support/SG-SEMTECH-WSP.pdf,2015>. Retrieved 25 Jan 2020.
- Semtech Corporation. SX1272/3/6/7/8: LoRa modem. Designer’s guide. AN1200.13; Semtech Corporation: Camarillo, CA, USA, 2013. [https://www.semtech.com/uploads/documents/LoraDesignGuide\\_STD.pdf](https://www.semtech.com/uploads/documents/LoraDesignGuide_STD.pdf). Retrieved 10 Feb 2020.
- Mansourkiaie, F., Ismail, L. S., Elfouly, T. M., & Ahmed, M. H. (2017). Maximizing lifetime in wireless sensor network for

structural health monitoring with and without energy harvesting. In *Special section on energy harvesting and scavenging: Technologies, algorithms, and communication protocols*; *IEEE Access*, 5 (pp. 2383–2395). <https://doi.org/10.1109/ACCESS.2017.2669020>.

24. Liu, G., Tan, R., Zhou, R., Xing, G., Song, W. Z., Lees, J. M. (2013). Volcanic earthquake timing using wireless sensor networks. In *Proceedings of the 12th international conference on information processing in sensor networks*, New York, NY, USA (pp. 91–102. 8–11). <https://doi.org/10.1145/2461381.2461396>.
25. LoRa Alliance. Lorawan specification v1.1. [https://loro-alliance.org/sites/default/files/2018-04/lorawantm\\_specification\\_v1.1.pdf](https://loro-alliance.org/sites/default/files/2018-04/lorawantm_specification_v1.1.pdf), 2017. Retrieved 27 Jan 2020.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Husam Rajab** received a B.Sc. degree in Computer and Communication Engineering at Al-Azhar University, Gaza, Palestine, in 2013. He received an M.Sc. degree in Electrical and Communication Engineering at Beijing University of Posts and Telecommunication in 2017. He is currently pursuing his Ph.D. studies at the Department of Telecommunications and Media Informatics at Budapest University of Technology and Economics, Hungary. His main

research interests are the Internet of Things, Radio Resource Management and QoS control for next-generation mobile and wireless networks, Machine Learning, cognitive radio, spectrum sensing.



**Tibor Cinkler** has received MSc (1994) and Ph.D. (1999) degrees from the Budapest University of Technology and Economics (BME), Hungary, where he is currently a full professor at the Department of Telecommunications and Media Informatics (TMIT). He habilitated in 2013 and received the DSc degree from the Hungarian Academy of Sciences the same year. He has been visiting professor at the Department of Computer Communications of the Gdansk

University of Technology, Poland in the academic year of 2018/2019.

His research interests focus on the optimization of routing, traffic engineering, design, configuration, dimensioning and resilience of IP, Ethernet, MPLS, OTN, 4G/5G and particularly of heterogeneous GMPLS-controlled WDM-based multi-layer networks. He is author of over 300 refereed scientific publications including four patents, with over 2100 citations. He has been involved in numerous related European and Hungarian scientific projects and many industrial co-operations including ACTS METON and DEMON; COST 266, 291, 293, 15127; IP NOBEL I and II and MUSE; NoE e-Photon/ONe, e-Photon/ONe+, and BONE; CELTIC PROMISE and TIGER2; NKFP, GVOP, ETIK, as well as Ericsson, Magyar Telekom, Vodafone industrial projects. He has been a member of ONDM, DRCN, BroadNets, AccessNets, IEEE ICC and Globecom, EUNICE, CHINACOM, Networks, WynSys, ICTON, RNDM, HPSR, etc. Scientific and Programme Committees. He has been guest editor of a Feature Topic of the IEEE ComMag and reviewer for many journals. He has organized DRCN 2001 and 2013, ONDM 2003 and 2017, as well as Networks 2008, HPSR 2015, RNDM 2011, ICUMT 2011 conferences in Budapest. He teaches various courses on networking and optimization at the university, as well as for companies and also gives tutorials at conferences and summer and winter schools. He received numerous awards including Dimitris Chorafas Prize for Engineering, ICC best paper award, numerous HTE awards (HTE is the Hungarian IEEE sister society), including Tivadar Puskas, Virag-Pollak 3 times, and the 60-year anniversary medal Bolyai Medal, etc. He has been an ERC panel member. He chairs the IFIP WG 6.10 on Optical Networking.



**Taoufik Bouguera** has received the B.Sc. degree in telecommunication engineering in 2015 and the M.Sc. degree in telecommunication systems from the National Engineering School of Tunis (ENIT), Tunisia. He received a Ph.D. from the University of Nantes in 2019, and his thesis project in the realization of an energy-autonomous communication sensor for IoT applications. He is currently an IoT project manager in Astek Group Paris. His

research interests include the new low power wide area technologies and energy optimization and mobile networks.