Future Generation Computer Systems 134 (2022) 287–302

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

A novel time-slotted LoRa MAC protocol for scalable IoT networks

Hanan Alahmadi [a,*], Fatma Bouabdallah [a], Ahmed Al-Dubai [b]

[a] Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
[b] School of Computing, Edinburgh Napier University, Edinburgh, United Kingdom

ARTICLE INFO

Article history:
Received 28 August 2021
Received in revised form 2 April 2022
Accepted 5 April 2022
Available online 12 April 2022

Keywords:
Internet of Things
LPWAN
LoRa
MAC protocols
Time-slotted algorithms
Energy efficiency

ABSTRACT

Long Range (LoRa) networks provide long range, cost-effective and energy-efficient communications by utilizing the free unlicensed ISM band, which makes them appealing for Internet of Things (IoT) applications. However, in high density networks, reliable performance might be hard to achieve due to the nodes' random-access method. Furthermore, the duty cycle restrictions that are imposed on nodes and gateways transmissions can limit the scalability of the network. More importantly, the duty cycle restrictions that are imposed on the downlink communication from the server to nodes can impose further challenges. Consequently, the server in high density networks might not be able to communicate with all network nodes due to its limited duty cycle. Besides, the server might not be able to send individual controlling packets from server to nodes. One way to mitigate such a limit is to allow nodes autonomously determine their transmission parameters without the need for any downlink transmission from the server. Thus, this paper presents the Sector-Based Time Slotted SBTS-LoRa MAC protocol that allows nodes to determine their transmission parameters autonomously based on their location to the gateway. SBTS-LoRa is targeting large scale networks. Simulation results show that our proposed protocol significantly enhances the scalability and outperforms its counterparts by maximizing throughput without compromising the energy efficiency. Specifically, the average throughput for dense networks was enhanced 14 times compared to the Adaptive Data Rate ADR-LoRaWAN.

1. Introduction

Internet of Things (IoT) networks play a crucial role in enabling smart scenarios. IoT can be defined as a connection of objects that have the ability to sense and collect data from the surrounding environment and deliver the collected data through reliable connections [1]. In fact, Low Power Wide Area (LPWA) technologies are considered the most suitable technology to effectively enable Internet of Things (IoT) paradigm. In fact, LPWA networks provide Long-range connections with low energy consumption and cost. These features are quite convenient for IoT applications as these applications generally cover large-scale areas and connect (things) that are mostly powered with batteries. Examples of such applications include smart cities, smart agriculture, and smart metering [2]. Many LPWA technologies have been invented, which can be divided into LPWA technologies that either operate on Cellular networks or Wireless network. LTE-M [3] and NB-IoT [4] are examples of LPWA technologies for cellular networks, while SigFox [5] and LoRa [6] are examples of LPWA technologies that work on the free unlicensed band [7].

Among LPWA technologies for the unlicensed band, LoRa provides the longest coverage range thanks to the Chirp Spread Spectrum (CSS) modeling technique used at the LoRa physical layer. Specifically, signals that are transmitted in LoRa medium are more robust against interference, which is common in the free ISM band, hence they can travel for longer distances. Furthermore, LoRa physical layer provides variety of transmission parameters that highly affect the overall performance of the network. These parameters include the support of multi-channel communications, different Spreading Factors (SFs) that act as virtual channels, different transmission power levels (Txs), different channel bandwidth (BWs), and different Coding Rates (CRs). The supported number of channels, the transmission power levels, and the used bandwidths highly depend on the deployed region of the Industrial, Scientific and Medical (ISM) band. However, the number of supported Spreading factors (SFs) and the Coding Rates (CRs) are fixed for all regions. In fact, spreading factors determine the number of data bits that are modulated in each signal. For example, there are 10 data bits modulated in a signal that is transmitted with SF10. In fact, increasing the spreading factor will result in increasing the transmission time or the Time on Air (ToA) of a packet. Specifically, the data rate of a transmission with smaller SF is greater than the data rate of a transmission with larger SF. Besides controlling the data transmission

* Corresponding author.
E-mail address: hjalahmadi1@kau.edu.sa (H. Alahmadi).

https://doi.org/10.1016/j.future.2022.04.003
0167-739X/

List of abbreviations

ADR	Adaptive Data Rate.
BRR	Bit Rat Ratio.
CF	Carrier Frequency.
CR	Coding Rate.
DCL	Duty Cycle Limit.
ISM	Industrial, Scientific and Medical.
LoRa	Long Range.
PER	Packet Error Rate.
SBTS	Sector-Based Time-Slotted.
SF	Spreading Factor.
TDMA	Time Division Multiple Access.
ToA	Time on Air.
TX	Transmission power.

rates, SFs control the sensitivity level of the receiver. Whenever a spreading factor of a signal increases, the receiver sensitivity of that signal is also increasing. As a result, the communication ranges are extended for signals with larger SFs. According to LoRa, there are six spreading factors ranges from SF7 to SF12. Signals with different SFs are considered orthogonal, which means that two or more packets transmitted on the same channel but with different spreading factors can be successfully decoded at the receiver. Consequently, we can think of spreading factors as if they are virtually partitioned each physical channel into six sub-channels with the same bandwidth. Obviously, this will expand the number of parallel successful transmissions.

In fact, exploiting the variety of transmission parameters by assigning different combinations of them to nodes will result in simultaneous collision-free transmissions, which will maximize the network throughput. Furthermore, by using diverse transmission parameters among the nodes, the scalability of the network is also maximized [6]. However, as explained previously, there are different data rates, and hence different ToAs, and different receiver sensitivity levels for each spreading factor, which will result in different transmission qualities. In other words, having just unique set of parameters for each neighbor node is not the optimal distribution that will result in a high network throughput. The optimal distribution of transmission parameters is the distribution that maximizes the throughput without compromises the energy consumption. In order to achieve that, there are three main challenges of LoRa networks that need to be overcome as described in the following sections.

1.1. Scalability issue

The main attractive feature that makes LoRa networks appropriate for IoT applications is the long-range coverage, where the gateway could receive packets from thousands of nodes that are far from it for up to 14 km. However, long-range coverage is associated with the capture effect issue. Capture effect happens when two or more nodes located at different distances from the gateway transmit simultaneously on the same frequency. The gateway in this case will decode only the transmissions of the closer nodes, ignoring hence the transmissions from farther nodes. This is because the link budget of the closer nodes is much larger than the link budget of the farther ones. Hence, the transmissions of the closer nodes are dominant compared to the transmissions of the farther ones. In fact, the capture effect results from the use of uncontrolled transmission power levels. Thus, adopting some transmission power control technique as well as isolating far nodes from closer ones by assigning them different frequencies will inevitably mitigate such effect.

1.2. Collisions due to ALOHA access method

According to LoRa physical layer, sensor nodes access the channel using ALOHA access method. In other words, once a node has a packet to transmit, it will wake-up and transmit it immediately without any carrier sensing or time regulations techniques. In fact, although there are multiple channels for transmissions in LoRa physical layer, nodes by default use the default channels only, which are three channels only in Europe band. Furthermore, as mentioned earlier, spreading factors are orthogonal, where transmissions with different SFs on the same channel might be successfully received at the gateway. However, inefficient distribution of SFs may result in having same SFs used by most of neighbor nodes. Consequently, with numerous connected nodes, collisions are not avoidable. Using some Time-Division Multiple Access (TDMA) techniques and having optimal distribution of SFs among the connected nodes are the key approaches in order to mitigate collisions.

1.3. The duty cycle restrictions

Since LoRa networks use the unlicensed ISM band, duty cycle regulation is imposed on the unlicensed ISM band, which is considered a key limiting parameter [8]. Duty cycle is the fraction of time during which a node is allowed to transmit to the medium. For example, in Europe region, the maximum duty cycle of uplink transmissions of nodes is 1%, which means that nodes are allowed to transmit for only 36 s per hour. In fact, this is highly affected by the used SF, as SFs control the Time on Air (ToA) of transmitted packets. For example, the ToA of a packet with 20 bytes payload that is transmitted in a channel with 125 kHz and a coding rate equals 4/5 is ranging from 56.6 ms to 1319 ms depending on the used SF [9]. In addition to that, duty cycle restrictions are also imposed on the downlink communication of the gateway. Accordingly, with large number of connected nodes, the gateway cannot acknowledge all nodes due to the duty cycle restrictions. Furthermore, in the previous section, we mentioned that replacing the ALOHA medium access method with TDMA method could reduce collisions and hence enhance the network throughput. However, the process of disseminating medium access schedules through downlink communications from the gateway may be disrupted because the gateway may reach its duty cycle limit before completing the schedule dissemination process.

To tackle these challenges, we propose Sector-Based Time Slotted SBTS-LoRa, a MAC protocol that autonomously distributes LoRa transmission parameters among the connected nodes. Furthermore, it regulates the access to the transmission medium by autonomously assigning timeslots to nodes without any downlink transmissions from the gateway. Based on nodes distance to the gateway, they can individually determine their transmission parameters and timeslots. We leverage some tools from the geometry of circles to determine the timeslots of nodes. SBTS-LoRa is targeting large-scale dense networks where the previously mentioned challenges are highly arising. The main objective of the proposed protocol is to maximize the network scalability by increasing the network throughput without compromising the energy efficiency.

The contributions of this paper are summarized as follows:

(1) Presentation of a new comprehensive probability collision model that takes into account all possible events that could result in packet collisions at the receiver. The model is general and does not suppose a specific statistical distribution for the data generation rate or other factors. Most importantly, the model considers all supported transmission parameters of LoRa physical layer (Section 3).

(2) Designing and deploying SBTS-LoRa protocol as a novel MAC protocol to overcome key LoRa challenges such as the capture effect, the limited scalability, and the duty cycle. SBTS-LoRa replaces the ALOHA random access method with a TDMA access method. It allows each node to autonomously determine its transmission parameters by knowing only the gateway location and its own location. To do that, SBTS-LoRa firstly divides the network field into a set of annulus cells. Each cell will have a unique frequency, a set of eligible SFs, and a specific transmission power level. Nodes that are located on a specific cell boundary will use the transmission parameters assigned for that cell. Moreover, SBTS-LoRa divides each cell into a set of sectors. The sector ID to which a node belongs is the timeslot ID for which a node is allowed to transmit. The main novelty of SBTS-LoRa is the use of a decentralized approach that accurately and efficiently determine node's transmission parameters without burden the network with extensive control packets from the network server. (Section 4)

(3) Extensive analysis and simulation experiments using OMNET++ simulator under FLoRa framework have been conducted to explore the performance of the SBTS-LoRa under different operating conditions. [10] (Section 5). In order to critically evaluate the proposed protocol, we implement EXPLoRa-AT [11,12] besides the Adaptive Data Rate (ADR) of LoRaWAN [6] protocol. We evaluate the performance of all protocols in large-scale extremely dense networks, where nodes are distributed in an area with a maximum distance of 14 km from the gateway and the number of connected nodes ranges from 1000–5000 nodes. To the best of our knowledge, no proposed protocol for LoRa networks is evaluated under these challenged conditions and considering all LoRa transmission parameters.

The rest of the paper is organized as follows. Section 2 provides background information about LoRa networks and surveys recent related works. Section 3 presents the probability of collision model for LoRaWAN protocol. Section 4 describes in details the proposed SBTS-LoRa including the sectorization mechanism to autonomously determine node's timeslots. Section 5 presents the results of the performance evaluation process of the proposed protocol. Finally, Section 6 concludes the paper and provides insights for future works.

2. Background and related work

2.1. Background

A LoRa system consists mainly of sensor nodes, gateways, and an application server. In typical LoRa networks, the network topology is a star topology, where sensor nodes connect with peer-to-peer communications to one or more gateways. Fig. 1 shows the general architecture of LoRa networks. Generally, sensor nodes sense and collect data from the environment, which will send them through LoRa networks to gateways. Gateways in turn, forward the received data packets through typical IP network to the server for further processing. Obviously, star topologies are simple compared to mesh network architecture, where the complexity in the former is reduced and hence nodes consume less energy compared to the latter. LoRa is the physical layer of LoRa technology that uses Chirp Spread Spectrum (CSS) modulation, which provides the long-range communication link. LoRaWAN, on the other hand, is the MAC protocol that is deployed on top of LoRa physical layer. Generally, since MAC protocols control the access to the shared medium as well as determine transmission parameters to nodes, they have crucial effect on network performance and scalability. Consequently, although LoRa physical layer provides efficient techniques to transmit packets for longer distances with minimum interference,

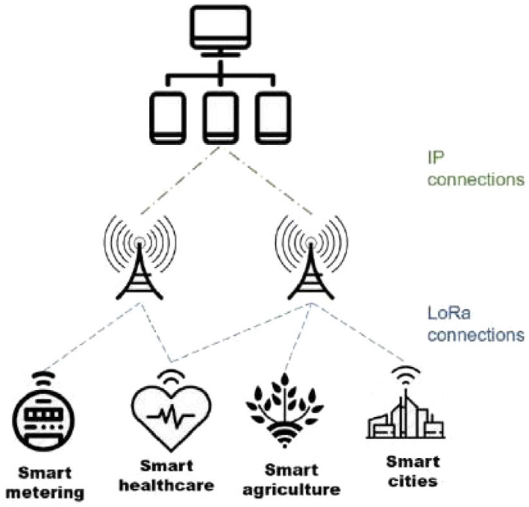

Fig. 1. LoRa network architecture.

without efficient MAC protocols that exploit all LoRa physical features the optimal performance of LoRa networks could be hard to achieve. According to LoRaWAN class A nodes, nodes access the medium in ALOHA manner. In other words, whenever a node has a packet, it will randomly select an SF, a Tx and a channel, then it will immediately send its packet. After that, a node will schedule two timeslots to receive any possible acknowledgment or MAC command from the server as follows: the first timeslot is at least 1 s latter after the end of packet transmission, while the second timeslot is exactly after 1 s from the first receiving timeslot. If a node receives a downlink transmission from the server in its first receiving window, it will not open the second receiving window. In other words, a node could receive a downlink transmission in one of its receiving windows. LoRaWAN has an algorithm called the Adaptive Data Rate to enhance node's transmission parameters as described in the following section.

2.1.1. Adaptive Data Rate (ADR) algorithm [6]

The Adaptive Data Rate (ADR) algorithm of LoRaWAN tends to enhance the network performance by adapting both the node's data rates, which is mainly affected by the node's spreading factor, and the transmission power levels. To do that, during the initialization phase, nodes select randomly a spreading factor and a transmission power in order to send their join requests and packets to the server. When a node sends multiple packets without receiving any acknowledgment packet from the server, the node supposes that their transmissions cannot be reached to the server. Accordingly, the node will gradually increase its spreading factor and/or their transmission power. Once the server receives packets from nodes, it will record the Received Signal Strength Indicator (RSSI) for each received packet. After the server receives a specific number of packets from a given node, it will estimate the most suitable SF and transmission power level, based on the recorded RSSIs for that node and transmit them in a downlink communication. In other words, if a node has a high link budget, the server will adapt its transmission parameter by reducing the data rate and/or the transmission power level.

Although ADR-LoRaWAN has efficient performance in small- to medium-scale networks, its efficiency dramatically decreases in large-scale dense networks, as demonstrated in Section 5. The main reason is the centralized approach of the ADR algorithm, where the server needs to update each node independently by a dedicated downlink communication and this update process could be obstructed due to duty cycle limitation. Consequently,

when nodes did not receive downlink communications from the server in their expected periods, it will suppose that their transmissions are not reachable. Hence, it will increase its spreading factor and/or transmission power. Eventually, most of end nodes will end up with the highest spreading factor with the longest time on air, which will further increase the probability of collisions and hence reduce the overall network throughput.

2.2. Related work

In order to mitigate collisions and hence enhances LoRa network performance and make them more scalable, research works can be divided into two main categories: ones that are related to improving the distribution of the available transmission parameters of LoRa physical layer among nodes and others that aim at changing the ALOHA medium access method to Time-Division Multiple Access (TDMA) access method. The following sections describe these two main categories.

2.2.1. Improving the distribution of transmission parameters

Regarding the first category, most of the proposed distribution algorithms focus only on distributing Spreading Factors (SFs) [11, 13,14] ignoring hence both the multi-channel support and the availability of multiple transmission power levels. By doing so, collisions related to the capture effect issue are considered unresolved as all nodes in the deployment area, regardless of their RSSI at the gateway, use the same transmission power and the same channel. As a result, the gateway will mainly decode the transmissions of the close nodes only. ExploRa-AT [11] is an example of such algorithms. EXPLoRa-AT uses LoRa default channel and a fixed transmission power (14 dBm). It distributes spreading factors among nodes by dividing nodes into six groups, which corresponds to the number of supported SFs in LoRa physical layer, such that all groups eventually will have similar Time-on-Air (ToA). Similar to ADR algorithm of LoRaWAN, EXPLoRa-AT runs at the server and distributes the recommended SF for each node in a centralized way. In other words, the server has to update individually each node with the recommended SF. Specifically, once the server receives a packet from a node, it will send the updated SF to that node in its receiving windows. Therefore, duty cycle limitation imposed on gateway's downlink communication could be the bottleneck of the system. According to EXPLoRa-AT algorithm, all nodes start with SF12 to make sure that their joining requests are not received below the sensitivity at the server. Then, after the joining requests transmission phase, the server will run the algorithm, determine the recommended SF for each node according to the join packet's RSSI, and then transmit that SF on nodes' upcoming receiving windows. Since all nodes are sending only on the default channel using SF12, the collisions between the joining requests may prohibit some nodes from accessing the gateway. In fact, according to our simulation experiment, the number of nodes that could reach the server is decreasing with the increase of node density.

In fact, with extremely large number of connected nodes, the server cannot receive packets from all nodes with the same SF and on the same channel due to collisions. According to that, only a small portion of nodes will succeed delivering their messages to the gateway, and hence their SFs will be appropriately adapted. Fig. 2 shows the number of nodes successfully initialized at the server, i.e. the number of nodes the server receives packets from and hence records their RSSIs in order to adapt their SFs compared to the actual number of nodes in the network for 12 days simulation time.

Furthermore, in order for the server to run EXPLoRa-AT algorithm, it has to wait for the network to stabilize where all nodes have sent their join requests so the server can have node's RSSIs

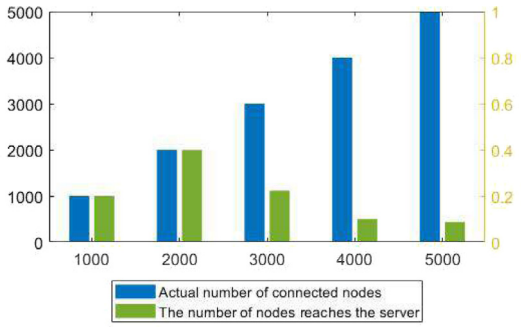

Fig. 2. Scalability limitation of EXPLoRa-AT algorithm.

in order to be able to estimate the appropriate SF according to the RSS of node's join request packets. This condition could be hard to satisfy especially with large number of connected nodes (2000 nodes or more), or with mobile nodes.

On the other hand, other researches such as algorithms that are proposed in [12,15] are taking into account the distribution of both the spreading factors and the different transmission power levels among the nodes. However, they only use LoRa default channels and hence they are not exploiting the multi-channel communication feature that could boost up the network throughput. Specifically, the algorithm in [12], we call it the Bit Rate Ratio (BRR) algorithm, exploits the fact that the transmissions with smaller SFs have higher bit rates, and hence lower Time on Air (ToA), than the transmissions with larger SFs. Therefore, the BRR algorithm distributes the spreading factors among nodes such that large portions of nodes use small SFs and only small portions of nodes use large SFs. By doing that, collisions resulting from long transmissions could be mitigated. To do that, the portion of nodes Ps_i that use a given SF_i is calculated according to the following equation [12]:

$$P_s = \frac{\frac{SF}{2^{SF}}}{\sum_{i=7}^{12} \frac{i}{2^i}} \tag{1}$$

Similar to EXPLoRa-AT [11]and ADR [6] algorithms, the Bit Rate Ratio [12] algorithm is implemented in a centralized approach, where it runs at the server that is responsible of distributing SFs using downlink communications.

2.2.2. Improving medium access methods

As for the second category, many researches have been proposed to mitigate collisions of LoRa networks by changing the ALOHA medium access mechanism. In fact, most of the work opting for TDMA based medium access adopt a centralized approach, where the network server is responsible for setting up and disseminating the schedule to all nodes on the network [16–20]. The adopted centralized approaches can be further divided into beaconic-based approach and synchronization-based approach. According to the beaconic-based approach, the gateway initiates the transmission by sending beacons that hold control information to nodes at the beginning of each frame. Nodes use these beacons to set up their transmission parameters and their timeslots. RS-LoRa [17], RT-LoRa [18], and DG-LoRa [21] protocols are examples of beaconic-based algorithms.

In DG-LoRa [21], the frame structure is divided into two main periods, the Up-link Transmission Period (UTP) and the Downlink Transmission Period (DTP) where the gateway sends a beacon message at the beginning of each frame. Nodes during the UTP

uses ALOHA access method while gateways use TDMA access method in the DTP. Hence, DG-LoRa does not comply with LoRaWAN MAC protocol, where after each Uplink transmission, there are two downlink transmission phases. Unlike DG-LoRa, in SBTS-LoRa, no beacons are needed from the gateway to control nodes. Furthermore, SBTS-LoRa uses TDMA in uplink transmissions without violating LoRaWAN standard. Hence, SBTS-LoRa could be easily adopted in LoRa networks with very minor modifications. In general, the initialization and synchronization process of LoRaWAN class A devices is always triggered by end nodes as the downlink transmission to these devices is always triggered by an uplink transmission from a node. Consequently, using beaconing-based approaches for synchronization may conflict with the behavior of LoRaWAN class A devices.

Regarding the synchronization-based approach, it has generally two main phases, the synchronization phase and the data transmission phase. During the synchronization phase, unlike beaconic-based protocols, nodes initiate the connection by sending synchronization requests including information that help the server to determine node's transmission parameters. The proposed protocols in [19,20] are examples of the synchronization-based approach.

In fact, the overhead of the initialization phase in synchronization-based algorithms limits the scalability of the network as each node should send/receive join-request/accept packets. Moreover, in saturated networks, nodes would not have slots and must wait for a long time either because the network server reach its duty cycle limit or there is no free timeslots that can be used to transmit the synchronization replies to nodes. This will result in a limited scalability of LoRa networks.

Obviously, all previous researches use centralized approaches to schedule and distribute the schedule among the nodes. However, gateways should respect the duty cycle of the ISM band, which prevents the downlink transmissions from gateway to nodes most of the time, especially with dense networks. One possible solution is to allow nodes autonomously determine their transmission parameters and most importantly their timeslots independently without extensive downlink transmissions from the gateway. To the best of our knowledge, there are few algorithms where nodes autonomously determine their transmission parameters [22,23]. Both algorithms in [22,23] determine the timeslots of nodes by applying the modulo operator on some features of nodes.

In [22], nodes determine their timeslots by applying a modulo operation on their MAC addresses, which does not guarantee the unicity of slots used by each node. In addition, the proposed solution is not completely autonomous as nodes still need to receive the frame size from the gateway to calculate their slot IDs. Furthermore, using MAC-based solution results in large number of empty slots in a given frame and this number is exponentially increasing with the increase of the number of nodes, which makes this solution not suitable for dense network scenarios.

Alternatively, instead of using node's MAC addresses, TS-LoRa [23] protocol utilizes the DeVAddr attribute of the Join-Accept packet, which is created and transmitted by the network server during the joining phase.TS-LoRa has six frames, one frame for each SF. All frames have a maximum length, which is broadcasted by the network server. According to TS-LoRa, when the server receives a join request packet from a node, it will generate a device address 'DeVAddr' for the node such that the generated 'DeVAddr' modulo the frame length results in a unique slot ID for that node on its frame. To guarantee such unicity, the modulo operation is done recursively by the server until it founds the appropriate 'DeVAddr'. Once the server founds the appropriate 'DeVAddr' that results in a unique slot ID, it will replay with a join reply including that 'DeVAddr'. On the node side, it will perform the modulo operation only once to retrieve its slot ID. Nevertheless, the SFs assignment was not discussed in the paper. Furthermore, they assume a maximum frame length (S), which may limit the scalability of the network. Note that in [22,23], the node configuration is not fully autonomous as nodes still need some information from the gateway or the network server to set their transmission parameters especially their slot numbers. Table 1 summarizes the main features and limitations of the related work researches.

On the other hand, in this paper, the proposed algorithm provides a comprehensive solution that takes advantage of all variant of transmission parameters that are provided by the underlining physical layer. Furthermore, based on node's location to the gateway, each node independently determines its time slot for transmission. By doing so, the protocol avoids the overhead associated with TDMA solutions that are related mainly to synchronization. To the best of our knowledge, no research provides such comprehensive and autonomous solution to mitigate collisions and hence improving the scalability of LoRa networks.

3. Collision probability model of LoRaWAN

In this section, we focus on deriving the probability of unsuccessful transmission attempt, $\beta(U, G)$, from a node U to the gateway G due to either channel error or collision in legacy LoRaWAN protocol. The main objective of this analysis is to mathematically assess the performance of LoRaWAN in order to define its weaknesses to help in proposing the right solution that aims at enhancing LoRaWAN performance. To the best of our knowledge, there is no model that compute the probability of collisions using discrete events. Furthermore, the proposed model is more accurate as it considers all the transmission parameters of LoRa without simplifying the model with certain assumptions like assuming a specific distribution of nodes. Moreover, it considers all possible events that result in collision. To the best of our knowledge, no events have been neglected in order to simplify the model. The proposed model can be a starting point for future research work aiming at mathematically assessing the performance of LoRaWAN in order to optimally distribute the LoRa parameters in any random network topology.

According to LoRaWAN, nodes access the medium using ALOHA mechanism, where nodes randomly choose their spreading factors and their communication channel in order to proceed sending their messages at anytime without performing channel listening while respecting the duty cycle constraint. We calculate $\beta(U, G)$ by mimicking the real case scenario to the most possible extent. To do so, we determine the different events that arbitrate the LoRaWAN channel access. Accordingly, a transmission from node U is unsuccessfully received by the gateway G if one or more of the following events occur:

- A: a packet error occurs during the transmission on the wireless link (U, G)
- B: one or more sensor nodes within the transmission ranges of G transmit at the same time as node U in the same channel using the same spreading code.
- C: node U transmits while G is busy with a transmission from a neighbor node. For example, node U uses a given spreading factor SF_U. There is a node N within the transmission range of G. Node U will access during node N transmission using the same spreading factor as node N and the same channel as node N.
- D: the gateway G receives transmissions from another node while node U transmission is still in progress. Typically, if during the packet transfer from U to G in a given channel and using a given spreading factor, node N transmits to G using the same channel and the same spreading factor as U.

Table 1
Comparison of related work researches.

Ref.	Objective	Approach	Main idea	Channel access	Does considering capture effect?	Is it scalable?
[11]	Enhance throughput	Centralized	Divide nodes into six groups with equal ToA	Random	✗	✗
[12]	Fairness packet error rate	Centralized	Assign SFs to nodes based on a proposed probability distribution model	Random	✓	✗
[13]	Maximize packet success probability	Centralized	Divide nodes into six groups using stochastic geometry	Random	✗	✗
[14]	Maximize the network capacity	Centralized	Estimate the percentage of nodes that use a given SF to meet a given average success probability	Random	✗	✗
[15]	Fairness distribution of SFs	Centralized	Assign SF and TX to nodes based on a proposed probability distribution model	Random	✓	✗
[16]	Enhance the throughput	Centralized	Using S-ALOHA instead of P-ALOHA	Time-slotted	✗	✗
[17]	Improve scalability and reliability	Centralized	Provides nodes with coarse-grained scheduling to facilitate their selections	Hybrid	✓	✓
[18]	Support real time traffic	Centralized	Support both random and time-slotted traffics	Hybrid	✗	✗
[19]	Improve Packet Delivery Ratio	Centralized	Synchronize nodes with on-demand compressed schedule packets	Time-slotted	✗	✗
[20]	Support bulk data collection	Centralized	Using fine-grained schedule to speed up the data collection time while minimize the energy consumption	Time-slotted	✗	✓
[21]	Improve the reliability of the network	Centralized	Separating the uplink and downlink transmission periods and grouping the ACK of nodes	Hybrid	✗	✓
[22]	Enhance collision probability	Decentralized	Nodes determine their timeslots by applying a modulo operation on their MAC addresses	Time-slotted	✗	✗
[23]	Enhance the packet Delivery Ratio	Decentralized	Nodes determine their timeslots by applying a modulo operation on their DeVAddr attribute	Time-slotted	✗	✗
SBTS-LoRa	Maximize the throughput	Decentralized	Nodes determine their transmission parameters by leveraging tools from circle geometry	Time-slotted	✓	✓

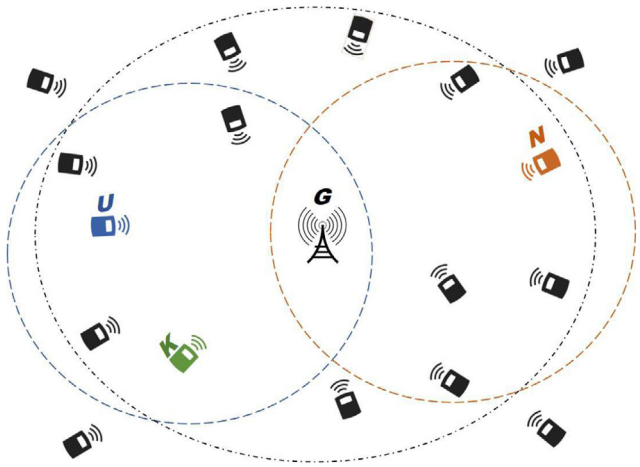

Fig. 3. Probability collision model of LoRaWAN.

Fig. 3 shows a demonstrative example of the events.

Assuming the independence of the aforementioned events, the probability of successful transmission from node U to the gateway G can be expressed as follows

$$1-\beta(U,G) = (1-Pr\{A\})(1-Pr\{B\})(1-Pr\{C\})(1-Pr\{D\}) \quad (2)$$

where the $Pr\{A\} = l(U,G)$ is the packet error rate on link (U,G). In what follows, we provide a detailed description on how to calculate the probabilities of events B, C, and D. To do so, let us discretize the time into so small slots of time.

3.1. Calculating Pr{B}

Theorem 1. *For every node U, the probability Pr{B} that one or more nodes in H(G)\{U} transmit in the same slot as node U, in the same channel and using the same spreading factor is given by*

$$Pr\{B\} = 1 - \prod_{k \in H(G)\setminus\{U\}} \left(1 - \frac{1}{N_{ch}} \times \frac{1}{N_{sf}} \times \frac{N_s}{N_{st}} \times \rho_K\right) \quad (3)$$

To describe $Pr\{B\}$, we have defined the following events:

$$E = \{\text{no nodes in } H(G) \text{ accesses in a given slot using a given channel and a given spreading factor}\} \quad (4)$$

$$F = \{\text{node } U \text{ accesses the same channel using the same SF in that slot}\}$$

Then we have,

$$Pr\{B\} = 1 - Pr\{E|F\} \tag{5}$$

Let us consider sensor node $K \in H(G) \setminus \{U\}$. According to that, we can define the following events:

$$\begin{aligned} X &= \{\text{node } K \text{ accesses a given channel using a given SF in a given slot}\} \\ Y &= \{\text{node } U \text{ accesses that channel using that SF in that slot}\} \\ Z &= \{K \text{ has a packet to transmit}\} \\ Q &= \{\text{node } K \text{ is in active state}\} \end{aligned} \tag{6}$$

Then we have,

$$\begin{aligned} Pr\{X \mid Y\} &= Pr\{X,\ Z,\ Q \mid Y\} \\ &= Pr\{X|Q\} \times Pr\{Q|Z,Y\} \times Pr\{Z\} \\ &= Pr\{X|Q\} \times Pr\{Q|Z\} \times Pr\{Z\} \end{aligned} \tag{7}$$

$$Pr\{Z\} = min(1,\ \lambda_K \times E[T_K]) = \rho_K \tag{8}$$

$E[T_K]$ can be defined as the time on air (ToA) of a packet from node K to G. In fact, computing the average value of $E[T_K]$ necessitates the estimation of the SFs distribution in the network according to LoRaWAN protocol. This distribution depends both on the network topology (nodes' positions) and mostly the network density. That being said, in dense network, regardless the nodes' positions, most of LoRaWAN nodes will end up choosing SF_{12} as they will not receive the acknowledgment from the gateway due to duty cycle limitations [24]. For this reason, and since the time on air depends on the used SF, the authors opt for choosing SF_{12} to calculate the $E[T_K]$ as follows [25] :

$$E[T_K] = ToA_{12} = \frac{2^{12}}{BW} \times \frac{PL}{12} \tag{9}$$

Moreover, recall that the total time on air for every sensor is limited by the duty cycle restrictions. Thus:

$$\lambda_K \times E[T_K] \leq \frac{N_s}{N_s t} < 1 \tag{10}$$

$$Pr\{Q|Z\} = \psi = \frac{N_s}{N_{st}} \tag{11}$$

$$Pr\{X|Q\} = \frac{1}{N_{ch}} \times \frac{1}{N_{sf}} \tag{12}$$

It is worth pointing out that X, Y, Z and Q are granular events that have been defined in order to meticulously calculate $Pr\{B\}$. Regarding X and Z, Z denotes the activity ratio of a node which is limited by the packet generation rate λ_K. Moreover, X can happen only if Z and Q occur. In other words, a node K can only access the channel using a given SF only if it has a packet to be transmitted and is allowed to access according to the duty cycle. In other words, if a node K has a packet to be transmitted but its duty cycle has been consumed; then, node K cannot access the channel. Consequently, in order for node K to access the channel, two conditions have to be satisfied, node K has a packet in its queue and node K is active (its duty cycle has not been consumed yet).

3.2. Calculating Pr{C}

The probability $Pr\{C\}$ that the gateway G is already busy with a transmission on a given channel with a given spreading factor when it receives a transmission from U on the same channel using the same spreading factor can be expressed as follows

$$\begin{aligned} Pr\{C\} &= Pr\{G \text{ is already busy on a given channel with a given SF} \mid \text{node } U \text{ accesses that channel to transmit to } G \text{ using that SF}\} \\ &= Pr\{G \text{ is busy on a given channel only due to nodes other than } U \text{ using a given SF}\} \\ &= Pr\{G \text{ is busy on a given channel } Ch_U \text{ due to nodes other than } U \text{ that use a given } SF_U \mid G \text{ is busy}\} \times Pr\{G \text{ is busy}\} \end{aligned} \tag{13}$$

The first element can be written as follows

$$Pr\{G \text{ is busy on a given channel } Ch_U \text{ due to nodes other than } U \text{ that use a given } SF_U \mid G \text{ is busy}\} = \frac{\left[1 - \prod_{K \in H(G)\setminus\{U\}} (1 - \omega_K)\right]}{1 - \prod_{K \in H^-(G)} (1 - \omega_K)} \tag{14}$$

$$\begin{aligned} \omega_K &= Pr\{X|Z\} \times Pr\{Z\} \\ &= Pr\{X,\ Q|Z\} \times Pr\{Z\} \\ &= Pr\{X|Q,\ Z\} \times Pr\{Q \mid Z\} \times Pr\{Z\} \\ &= \frac{1}{N_{ch}} \times \frac{1}{N_{sf}} \times \psi \times \rho_K \end{aligned} \tag{15}$$

The second element can be expressed as follows

$$\begin{aligned} Pr\{G \text{ is busy}\} &= 1 - Pr\{G \text{ is not busy}\} \\ &= 1 - \prod_{K \in H(G)} \delta(G,\ K) \end{aligned} \tag{16}$$

$$\begin{aligned} \delta(G,\ K) &= Pr\{G \text{ is not occupied by a transmission from } K\} \\ &= 1 - Pr\{G \text{ is occupied by a transmission from } K\} \\ &= Pr\{G \text{ is occupied by a transmission from } K \mid K \text{ is active}\} \times Pr\{K \text{ is active}\} \\ &= \psi \times \left(\lambda_K \times \overline{N}_C(K, G) + 1\right) \times ToA_{12} \\ &\quad + \lambda_K \times T_{ACK} \\ &= \psi \times \lambda_K \left[\left(\overline{N}_C(K, G) + 1\right) \times ToA_{12} + T_{ACK}\right] \end{aligned} \tag{17}$$

$$\delta(G,\ K) = 1 - \left[\psi \times \lambda_K \left[\left(\overline{N}_C(K, G) + 1\right) \times ToA_{12} + T_{ACK}\right]\right]$$

$$\overline{N}_C(K, G) = \frac{\beta(K, G)}{1 - \beta(K, G)} \tag{18}$$

$\overline{N}_C(K, G)$ can be derived as follows: $N_C(K, G)$ be a random variable representing the number of unsuccessful transmissions experienced by a given data message from K to G. If $\beta(K, G)$ is the probability that a transmission a tempt from K to G is failed. Then, $N_C(K, G)$ is a geometric random variable and thus we have

$$\overline{N}_C(K, G) = \frac{\beta(K, G)}{1 - \beta(K, G)}$$

where $\beta(K, G)$ is the probability of collision on link (K, G). Finally, substituting Eqs. (14) and (16) into Eq. (13), we get expression of $Pr\{C\}$.

3.3. Calculating Pr{D}

We firstly define the vulnerability period T_v as follows:

$$T_v = \frac{ToA_{12} + D}{Slot} \tag{19}$$

Table 2
Notations and descriptions.

Notations	Description
$\beta(U, G)$	The probability of unsuccessful transmission from a node U to the gateway G.
$H(G)$	All the nodes within the transmission range of the gateway. G
N_{ch}	Number of available channels.
N_{sf}	Number of spreading factors.
N_s	Number of active slots per unit of time .
N_{st}	Total number of slots per unit of time.
ρ_K	The utilization of node K.
$E[T_K]$	The average transmission time from node K to the gateway G.
λ_K	The traffic rate from node K to the gateway G.
ToA_{SF}	The Time on Air (ToA) of a packet using the spreading factor SF.
BW	The channel bandwidth.
PL	The payload length.
ψ	Node duty cycle.
ω_K	The probability that node K transmits on the medium in a given slot in a given channel using a given SF.
$\delta\ (G,\ K)$	The probability that G is not occupied by a transmission from K.
$\overline{N}_C\ (K, G)$	The average number of unsuccessful transmissions from K to G before being successfully received.
D	The maximum propagation time.
T_v	The vulnerability period.

T_v is the vulnerability period in terms of slots during which the transmission of a node within the range of G prevents the success of the in-progress transmission from U to G. Hence, the probability $Pr\{D\}$ that the gateway G receives transmission from other node than U while the transmission of node U is still in progress can be derived as follows:

$$\begin{aligned} Pr\{D\} &= 1 - Pr\{\text{no node in } H(G) \setminus \{U\} \text{transmits} \\ &\quad \text{during } T_v \text{ slots}\} \\ &= 1 - \prod_{K \in H(G)\setminus\{U\}} (1 - \omega_K)^{T_v} \end{aligned} \tag{20}$$

Finally, substituting Eq. (3), Eq. (13) and (20) into Eq. (2), we obtain the probability of collision $\beta(U, G)$. Table 2 represents the notations and description used in this section.

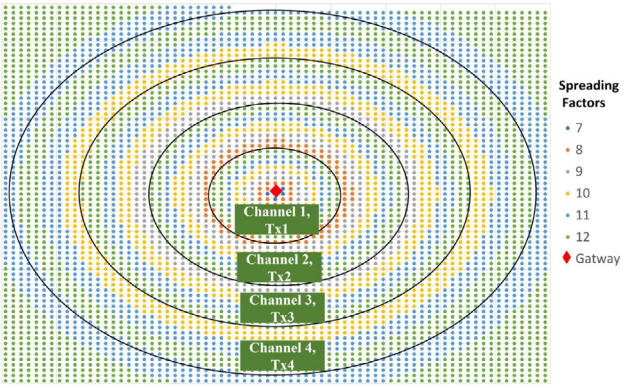

Fig. 4. Annulus-based partitioning of LoRa network.

4. Sector-based time-slotted LoRa protocol

4.1. Overview

As mentioned before, LoRa physical layer provides variant transmission parameters that if they are distributed and used in an efficient way, they could overcome most of LoRa network challenges. Hence, in this paper, the researchers are taking advantage of all transmission parameters to provide comprehensive protocol that eliminate collisions and hence increase the scalability of the network to the most possible extent. Accordingly, our research work can be divided into two main stages: (i) distributing all transmission parameters that affect the scalability of the network, (ii) adding a TDMA layer on top of the proposed parameters distribution strategy, to allow collision-free transmissions for nodes with similar transmission parameters. These two stages are described in the following sections.

4.2. Annulus-based distribution of transmission parameters

According to [24] our protocol supposes a star network topology where the gateway is located at the center and nodes are distributed randomly around the gateway with a maximum distance R. SBTS-LoRa protocol starts by dividing the area around the gateway into six annulus cells. Each cell is assigned a unique channel and a transmission power level. By doing so, collisions emanating due to the capture effect are avoided as all nodes on a given channel use the same transmission power and they are spatially close to each other. After that, we define a set of eligible SFs for each cell. The set of eligible SFs of a given cell is the set of SFs that can be used by the nodes on that cell such that their packets transmissions will not be discarded by the gateway because the strength of the received signal was below the sensitivity of the gateway. According to LoRa standard, each spreading factor has a given sensitivity level at the gateway. Specifically, when we increase the SF, the sensitivity of the gateway is increasing too, allowing successful reception for far nodes. Hence, the set of eligible SFs for a given cell is decreasing while the cell is getting farther from the gateway. Indeed, if nodes in the first closest cell to the gateway are allowed to use any SF (SF7 to SF12), nodes in the farthest cell are only allowed to use SF12 as it is the only SF that guarantees successful reception at the gateway. Once the set of eligible SFs is defined for every cell, we proceed distributing the SFs between the nodes of that cell. To do so, we further divide each cell into a number of sub-cells that equals the number of eligible SFs for that cell. In fact, if the number of eligible SFs on a given cell is 4, it will be partitioned into 4 sub-cells. Fig. 4 demonstrates the SF distribution according to our proposed algorithm. By doing so, collisions between nodes on the same cell but on different sub-cells are avoided as they use different spreading factors. However, collisions between nodes that are on the same sub-cell may still arise, as they use the same SF on the same frequency CF.

4.3. Sector-based time-slotted LoRa (SBTS-LoRa) protocol description

In order to avoid collisions between nodes on the same sub-cell that are sharing the same channel, the same transmission power and the same spreading factor, we regulate their transmissions by assigning a unique timeslot for each node. To do so,

we propose SBTS-LoRa protocol that is described in details in the following sections.

4.3.1. Overview

Many researches have emphasized on the importance of changing the ALOHA access method of LoRa networks and using instead Time-Division Multiple access (TDMA) [8,26]. For instance, authors in [16] found that the maximum LoRaWAN throughput with slotted Aloha S-ALOHA is the double of LoRaWAN throughput with pure-Aloha P-ALOHA. These findings confirm the efficiency of adopting TDMA approaches on top of LoRa physical layer to mitigate collisions and hence increase its scalability. According to that, there are some researches that have proposed TDMA medium access algorithms for LoRa networks [17–21]. However, all these algorithms use centralized approaches to schedule transmissions between nodes. In other words, both the newly-joined and the already-joined nodes need to receive periodic transmission schedules from the gateway using extensive downlink transmissions. As a result, the scalability will be compromised because of the duty cycle restrictions of the downlink transmissions that are imposed by the ISM band. However, in our proposed algorithm, the schedules are determined by the nodes autonomously without any need of downlink transmissions from the gateway. Specifically, based on node's location to the gateway, each node independently determines its slot number and hence its transmission time. Since LoRa networks have a star network topology with a gateway centered in the middle and nodes are distributed around the gateway, we get inspired from the geometry of circles and sectors of circles to specify independently the transmission schedule of each node. The following sections describe in details the initialization and data transmission phases of the proposed SBTS-LoRa protocol.

4.3.2. The initialization phase

In order to avoid collisions between nodes that are on the same sub-cell, which they use the same SF on the same channel, we further divide cells into sectors to which we assign numbers, as shown in Fig. 5. The number assigned to each sector is simply the slot number for every node belonging to that sector. By doing so, the same slot number can be reused by nodes that are in different subcells without any collision since they are using different orthogonal SFs. As a result, the frame size can be reduced such that only a limited number of timeslots can safely serve large number of nodes. As for nodes that are in the same sub-cell, they will get different slot numbers as they reside in different sectors.

We assume that a node n knowing its geographical coordinates $(X_n,\ Y_n)$ and the ones of the gateway $(X_G,\ Y_G)$ will automatically and autonomously know to which sector it belongs and hence its slot number for transmission. To do that, each node located on $cell_i$ calculates an angle α_i according to which $cell_i$ will be partitioned into sectors of angle size α_i with the gateway as sector origin. Fig. 5 shows an example of sectors division of $cell_3$. As shown in Fig. 5, $cell_3$ has 4 sub-cells and 8 different sectors each with an angle α_3. The sectors will be assigned a number according to their relative position from the gateway as it will be detailed later. The intersection between a sub-cell and a sector is called *Annulus-Sector*. Given that, each sub-cell is assigned a different SF, all nodes located on a given sector use that sector number as their slot number. In other words, the same slot number can be used by nodes on different sub-cells without any collision since they use different SFs. Similarly, in order to avoid collisions between nodes on the same sub-cell, we aim at guaranteeing the existence of a unique node per *Annulus-Sector*. In other words, to guarantee that only one node exists in every *Annulus-Sector*, the following equation should be satisfied:

$$\left[\frac{\alpha_i}{2} \times \left(ir^2 - \left(ir - \frac{r}{6-(i-1)}\right)^2\right)\right] \times d = 1 \tag{21}$$

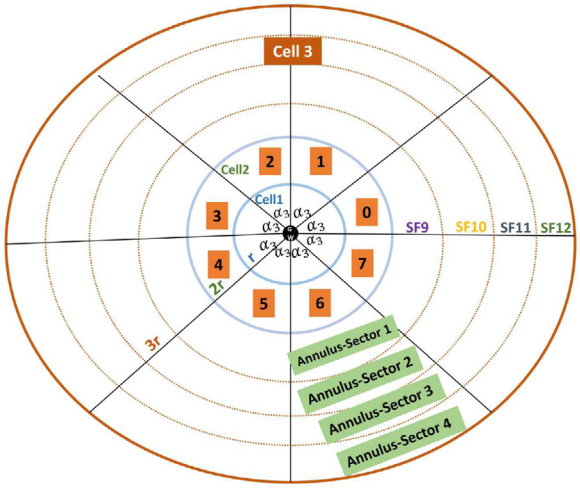

Fig. 5. Dividing cells into sectors.

where i is the cell identifier that ranges from $i = 1$(the closet cell) to $i = 6$ (the farthest cell), d is the node density, α_i is the sector angle of $cell_i$, and r is the annulus width, which is the maximum distance from the gateway, R divided by 6.

The first term on the left hand-side of Eq. (21) simply denotes the area of the *Annulus-Sector* of the last sub-cell of $cell_i$, which is multiplied by the node density d to estimate the number of sensor nodes in the last *Annulus-Sector*. According to Eq. (21), the number of nodes in the last *Annulus-Sector* must be equal to 1 in order to have just a unique node uses a given slot number and a given SF. Any other node on the same cell has either a different SF or a different slot number. Indeed, if we guarantee that the largest *Annulus-Sector* in a given sector contains only one node, we can deduce that for all previous *Annulus-Sector*, there is at maximum one node since previous *Annulus-Sector* have smaller areas than the last one. By doing that, we can achieve collision-free transmissions without the need for extensive down-link transmissions. To demonstrate that, let us consider the example that is shown in Fig. 5. The last and largest *Annulus-Sector* here is *Annulus-Sector 4*, if we guarantee that there is only one node in that sector, we can guarantee different slot numbers for nodes that use same SF on the same channel. Furthermore, by guaranteeing uniqueness timeslots for nodes located on the largest *Annulus-Sector*, we can deduce that nodes that are located on smaller *Annulus-Sector*, which are in our example the *Annulus-Sectors* ranging from 1 to 3, have unique slot numbers.

From the previous equation, we can derive the angle of $cell_i$ as follows:

$$\alpha_i = \frac{2}{d \times \left(ir^2 - \left(ir - \frac{r}{6-(i-1)}\right)^2\right)} \tag{22}$$

In fact, the size of a sector angle α is affected by two main factors: (i) the node density of the network, d and (ii) the radius of $cell_i$ that equals ir. Furthermore, the size of α has a direct impact on the frame size. The frame size in SBTS-LoRa is the total number of available time slots per $cell_i$. According to SBTS-LoRa, each cell has its own frame size that depends on the radius of that cell. In other words, closer cells have frame size smaller than the farther cells. This means that farther cells with less number of eligible SFs have more time slots. This will have the advantage of mitigating collisions on farther cells due to the reduced number of eligible SFs on these cells. For example, since nodes on $cell_6$ are so far from the gateway, they can use only SF12. Hence, if there are two

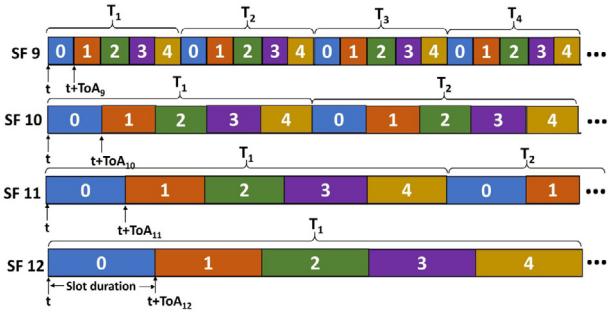

Fig. 6. SBTS-LoRa frame structure.

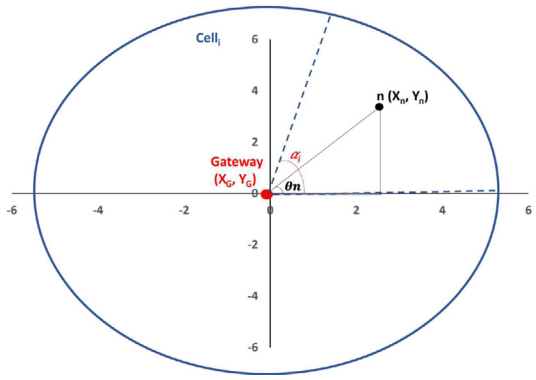

Fig. 7. Polar coordinates of node n.

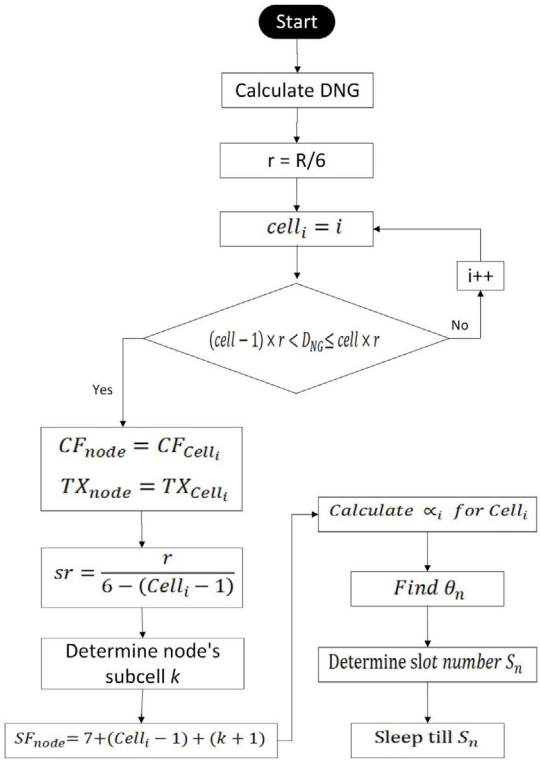

Fig. 8. Flowchart diagram of SBTS-LoRa protocol.

or more nodes on $cell_6$ transmitting simultaneously, there will be collisions between them. By having large frame size for far cells, we allow for more collision-free communication for those nodes.

After a node calculates the angle α_i of its $cell_i$, it can easily calculate the frame size m_i of its cell by the following formula:

$$m_i = \frac{2\pi}{\alpha_i} \tag{23}$$

Note that, according to SBTS-LoRa protocol, there is a fixed frame size for each cell. However, the slot duration is varying depending on the given sub-cell that reflects the used SF. This is because the frame size depends on the angle of α_i , where the α_i depends on the cell radius. The cell radius is doubled when cells get farther from the gateway. For example, as shown in Fig. 5, there are 8 timeslots for $cell_3$ regardless of the used SF. However, the slot duration for nodes located on the first sub-cell is different from the slot duration of nodes located on the last sub-cell since they use different SFs in their transmissions. In fact, the slot duration here acts as a third dimension besides the coordinates of the nodes. Fig. 6 shows a demonstrative example of the frame structure of $cell_3$. As shown in the figure, the frame consists of 5 timeslots regardless of the used SF. However, the duration of the slots for each frame is varying depends on the used SF.

After that, a node n calculates its polar angle θ_n based on the following formula:

$$\theta_n = tan^{-1}\left(\frac{Y_n - Y_G}{X_n - X_G}\right) \tag{24}$$

where (X_n, Y_n) are the coordinates of node n and (X_G, Y_G) are the coordinates of the gateway. Fig. 7 shows the polar coordinates of node n. Finally, each node n finds its slot number s_n by applying the following equation:

$$S_n = \frac{\theta_n}{\alpha_i} \tag{25}$$

Fig. 8 shows a Flowchart diagram of the proposed SBTS-LoRa protocol. Furthermore, Algorithm 1 illustrates the main steps of the proposed protocol. Once the slot number is determined, every node sleeps until its timeslot begins to start its data transmission.

4.3.3. Data transmission phase

During the data transmission phase, each node wakes up at the beginning of its timeslot to transmit its packet. According to SBTS-LoRa protocol, each node transmits one packet per frame. As mentioned earlier, there is a given frame size for each cell, which depends on the angle size of the sectors of a given cell. To make sure that nodes are transmitting once per frame without violating their duty cycle, the following condition must be satisfied:

$$t \times m_i \times ToA_{SF} \geq \psi_n \tag{26}$$

where t refers to an integer that denotes the time frame ID, ToA_{SF} is the Time on Air of a packet that uses a given SF, m_i is the number of timeslots per $cell_i$, and ψ_n is the duty cycle of node n, which is the time a node must wait before it can transmit again.

In other words, once a node succeeds its transmission of a packet, it will schedule the time for the next packet such that it will not violate its duty cycle ψ_n. To do that, a node will calculate the left-hand side of Eq. (26) by multiplying the frame ID t (start from 1) by the frame length m_i and ToA_{SF} of the selected SF. If the resulting waiting time for the node was less than the duty cycle ψ_n, which means that the node cannot sent a packet in the next upcoming frame, it will postpone the transmission to the after the next frame by increasing the frame ID by t 1. This process is repeated until the node finds the frame ID t for the next packet transmission that comply with its duty cycle ψ_n. By doing that, nodes respect their duty cycles as well as their schedule of transmissions.

Algorithm 1 SBTS-LoRa MAC Protocol

Input: node coordinates (X_n,Y_n), Gateway coordinates (X_G, Y_G), node density d, and p
Output: *cell*, CF_{node}, TX_{node},SF_{node}, and slotID S_n
$SFs \leftarrow [7, 8, 9, 10, 11, 12]$
$CFs \leftarrow [cf_1, cf_2, .., cf_6]$
$TXs \leftarrow [Tx_1, Tx_2, .., Tx_6]$
$D_{N_G} \leftarrow calculateEuclideanDistance(X_n, Y_n, X_G, Y_G)$
r is the annulus width, R is the field radius
$r \leftarrow R/6$
for $i \leftarrow 1$ to 6 **do**
  if $(D_{N_G} > (i-1)\cdot r)$ & $(D_{N_G} \leq (i \cdot r))$ **then**
    $cell \leftarrow i$
    $CF_{node} \leftarrow CFs[i-1]$
    $TX_{node} \leftarrow TXs[i-1]$
    $ns \leftarrow 6-(cell-1)$
    $sr \leftarrow r/ns$
    for $k \leftarrow 1$ to ns **do**
      $min \leftarrow ((k-1)\cdot sr)+((cell-1)\cdot r)$
      $max \leftarrow ((k)\cdot sr)+((cell-1)\cdot r)$
      if $(D_{N_G} > min)$ & $(D_{N_G} \leq max)$ **then**
        $subCell \leftarrow k$
        $SF_{node} \leftarrow 7+(cell-1)+(k-1)$
      end if
    end for
  end if
end for
$temp \leftarrow cell \cdot r-(r/(6-(cell-1)))$
$\alpha_{cell_i} \leftarrow 2\cdot p/d\cdot((cell\cdot r)^2-temp^2)$ ▷ Eq.(22)
$m_{cell_i} \leftarrow 2\cdot\pi/\alpha_{cell_i}$
$\theta_n \leftarrow \arctan(Y_n-Y_G/X_n-X_G)$
$S_n \leftarrow \theta_n/\alpha_{cell_i}$

4.4. Finding the optimal frame size

According to SBTS-LoRa protocol, the frame size of cells that are closer to the gateway is less than the frame size of cells that are farther from the gateway. This is mainly because the angle of closer cells is larger than the angle of farther cells. This has an advantage of having larger frame sizes for nodes on farther cells where there is a limited number of SFs that could be used. However, extremely large frame size is undesirable as it will increase the waiting time for a node to transmit again especially for farther nodes, where large SFs with high ToA are used and thus the network throughput maybe badly affected. Hence, a trade off between the frame size m_i and the number of nodes per *Annulus-Sector* of a given cell, p, must be performed.

As mentioned in the previous section, the frame size m_i (Eq. (23)) is mainly affected by α_i which in turn depends on the number of nodes per *Annulus-Sector* of a cell, called hereafter as p. Accordingly, Eq. (22) can be generalized as follows:

$$\alpha_i = \frac{2\times p}{d\times\left(ir^2-\left(ir-\frac{r}{6-(i-1)}\right)^2\right)} \tag{27}$$

Note that in Eq. (22), p is set to be equal 1. Indeed, setting p equals 1 means that we impose the presence of a unique sensor in the last *Annulus-Sector* of a given cell. Imposing a unique sensor in the last *Annulus-Sector* will result in at most one sensor in the other *Annulus-Sectors* of the same cell as the last *Annulus-Sector* has the largest area. Obviously, if we increase the p value, more nodes will be sharing the same slot which may result in more collisions but at the same time increasing p will increase the α_i of a given $cell_i$ and hence the frame size will be reduced and thus the end-to-end delay will be reduced. Clearly, an optimal p value will maximize the throughput since there is a tradeoff between the collisions and the end-to-end delay.

Hence, we have run SBTS-LoRa with different node densities d and p values to find an optimal p that maximize the network throughput. Fig. 9 illustrates the result of the experimental analysis, where p equals 0.5, 1, or 3 and with a number of nodes ranging from 1000 to 5000 nodes. We evaluate the impact of p in terms of the probability of collision (Fig. 9.a), the end-to-end delay (Fig. 9.b), the throughput (Fig. 9.c), and the total energy consumed per a successful reception of a bit (Fig. 9.d). As shown in Fig. 9.c, the maximum network throughput was achieved when $p = 1$. Furthermore, although the probability of collision and the end-to-end delay is not the best with $p = 1$, this is not compromising the energy consumption as it is also optimized with $p = 1$. Hence, the optimal p for our simulation experiments is 1. Thus, in Section 5, we evaluate SBTS-LoRa protocol with $p = 1$.

5. Performance evaluation

This section presents a simulation-based assessment that is performed on SBTS-LoRa protocol using OMNET++ [27] simulator under FLoRa framework [10]. FloRa is an open source framework that implements LoRa network elements such as LoRa nodes, gateways, and network servers. Specifically, it implements LoRa physical layer and LoRaWAN MAC layer of LoRa nodes. It also includes a module to depict energy consumption of LoRa nodes. FLoRa framework provides flexible configurations for different physical transmission parameters such as SFs, transmission powers, coding rates, and bandwidths. However, it only supports the default sub-band of LoRaWAN protocol. Hence, we modify the framework to include all the supportive sub-bands in Europe region. Specifically, our proposed protocol is mainly developed at the node's application layer of FLoRa framework. No required modifications were needed at the network server entity of FLoRa framework, as SBTS-LoRa protocol is completely distributed. We suppose that simultaneous transmissions with different SFs are considered orthogonal, which is the same assumption implemented in FLoRa framework and used by different studies [10–12,28]. As mentioned earlier, SBTS-LoRa is intended for large scale environments, where the number of connected nodes is extremely large. According to that, the number of connected nodes in the performance evaluation ranges from 1000 to 5000 nodes that are randomly distributed within a radius of 14 km from the gateway. Regarding the packet size, each node generates a packet of 20 bytes length with an inter-arrival time between packets following the exponential distribution with 1000 s mean. Similar to [10], we used the European regional parameters for the LoRa physical layer with 1% duty cycle for both the LoRa nodes and the gateway. Table 3 Summarizes the simulation parameters of the simulations. We compare the performance of our SBTS-LoRa protocol to EXPLoRa-AT [11], BitRateRatio (BRR) [12], and the Adaptive Data Rate (ADR) algorithm of LoRaWAN [6]. Although BRR in [12] is implemented using a centralized approach, we implement the algorithm in a distributed manner, so our proposed protocol can be evaluated with respect to both centralized and distributed algorithms. According to our implementation of BRR [12] algorithm, each node firstly determines the set of eligible SFs that can select from according to its path loss to the gateway. To do so, we use the cell-based distribution of SFs similar to [24] and the SBTS-LoRa algorithms. Then, each node autonomously calculates the Ps_i (Eq. (1)) for each SF. After that, each node will generate a vector of aggregates Ps_i, called APs_i,for each SF. Fig. 10 shows an example of aggregated ratios where

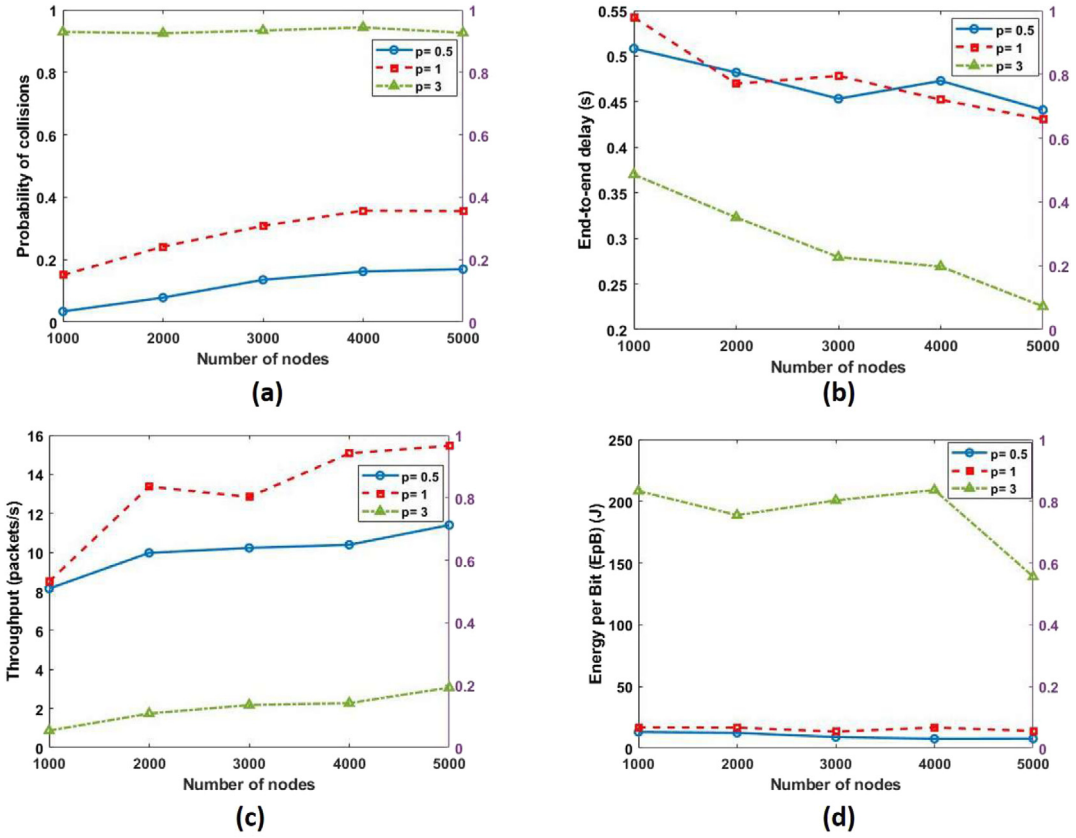

Fig. 9. Finding optimal p value.

Fig. 10. Example of the implemented Bit Rate Ratio Algorithm.

the set of eligible SFs accommodate all SFs (SF7–SF12). After that, each node will generate a random number x between (0,1] and check at which SF interval it exists. Then, the node will select the corresponding SF for that period. For example, as shown in Fig. 10, when the generated random number x = 0.8, the node will select SF9 for its transmissions. As illustrated in Fig. 10, smaller SFs have larger intervals of random numbers than larger SFs. Note that, in order to determine the set of eligible SFs for each node, we have been inspired in [24] by dividing the area around the gateway into cells such that each cell has different set of eligible SFs that depends on the distance to the gateway. However, note that BRR [12] uses only the default channels unlike [24] and SBTS-LoRa algorithms.

In this section, we demonstrate the performance evaluation of the proposed protocol with respect to the end-to-end delay, network throughput, probability of collisions, Packet Error Rate (PER), and energy consumption metrics.

5.1. The probability of collision

Fig. 11 shows the probability of collision as a function of the number of nodes. First, the probability of collision of all protocols is increasing with the increase of the number of nodes as the traffic rate is getting higher. However, SBTS-LoRa is increasing slowly compared to other protocols. Most importantly, SBTS-LoRa protocol is achieving the lowest probability of collision compared to other protocols thanks to the efficient distribution of transmission parameters and timeslots among the nodes that aim at mitigating collisions to the most possible extent. Our implementation of the BRR algorithm has the second lowest probability of collisions since it is distributed, where each node autonomously determines its transmission parameters.

On the other hand, ADR-LoRaWAN and EXPLoRa-AT have the highest probability of collisions because they are centralized protocols, where the selection and the distribution of the transmission parameters are performed at the server. Furthermore, LoRaWAN and EXPLoRa-AT use only the default channels without exploiting the multi-channel feature of LoRa physical layer and thus all the nodes tend to communicate on the same channel which will further increase the number of collisions. Specifically, since these protocols are centralized protocols, they have both uplink and downlink communications that are performed only on

Table 3
Simulation parameters.

Parameter	Value	Comments
SF	7 to 12	Spreading Factors.
TP	(2, 5, 8, 11, 14) dBm	Transmission power.
CR	4/8	Coding Rate.
CF	{868.1, 868.3, 868.5, 867.1, 867.3, 867.5}	Carrier Frequencies (MHz).
BW	125 kHz	Bandwidth.
PL	20 Bytes	Payload length.
R	14 km	Field radius.
N	1000–5000	Number of nodes.
Simulation time	10	Days.

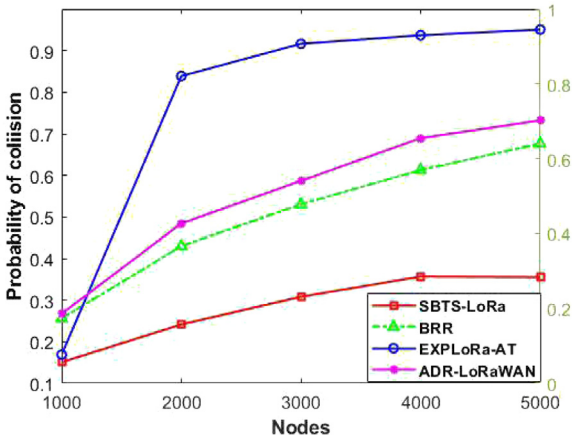

Fig. 11. Probability of collision.

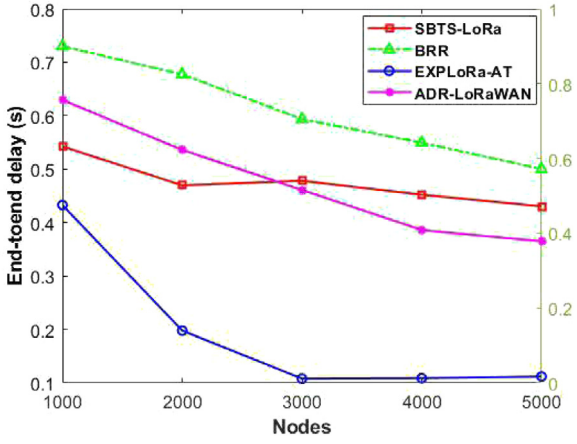

Fig. 12. The end-to-end delay.

the default channels, which will further increase the probability of collisions. EXPLoRa-AT has the highest probability of collisions, especially with extremely high number of nodes (more than 2000 nodes) because all nodes, according to EXPLoRa-AT algorithm, start their transmissions with SF12, to insure the successful reception of packets regardless of node's distance to the gateway. However, with an increasing number of nodes that send packets with the same highest SF12, the probability of collision is getting worse.

5.2. The end-to-end delay

Fig. 12 shows the end-to-end delay as a function of the number of nodes. The end-to-end delay is greatly affected by the used SF. Indeed, the usage of smaller SFs results in lower end-to-end delay as the packets that are transmitted by small SFs will have less Time on Air (ToA). According to SBTS-LoRa, SFs are assigned based on node's locations. This explains the minor decrease of SBTS-LoRa curve at N = 2000 nodes compared to others. Furthermore, the delay is only considered for the successfully received packets by the gateway. This explains why the end-to-end delay of BRR, EXPLoRa-AT, and ADR-LoRaWAN are generally decreasing with the increased traffic, as collisions for these protocols are increasing with the increased traffic (Fig. 11), and hence the number of successfully received packets, for which the end-to-end delay is counted for decreases. Moreover, EXPLoRa-AT has the lowest end-to-end delay compared to other protocols, especially when N = 2000 or more, as the number of nodes that successfully reaches the server is getting too small as shown in Fig. 2. With high number of connected nodes, only the traffic of a small portion of nodes reaches the server, which explains the small end-to-end delays achieved by EXPLoRa-AT.

The BRR algorithm, on the other hand has the highest end-to-end delay because the majority of nodes (about 75%) have large SFs (SF10–SF12). Obviously, this distribution is not compliant with the main objective of BBR, where the majority of nodes is supposed to use small SFs. In our experiment, nodes are distributed in large-scale environment and the area around the gateway is divided into annulus-cells such that the cells that are located farther from the gateway have higher area and hence accommodate higher number of nodes. Consequently, nodes located farther from the gateway are eligible to select only large SFs, which will result in higher delays.

5.3. The throughput

The throughput is mainly affected by the end-to-end delay and the probability of collisions. Fig. 13 demonstrates the network throughput as a function of the number of nodes. First, it is worth pointing out that SBTS-LoRa is by far achieving the highest throughput compared to other protocols with almost 15 packets/second that can be successfully received by the gateway when N = 5000 nodes. Indeed, SBTS-LoRa is achieving in average 1362% larger throughput than the second best protocol BRR. Indeed, BRR achieves only in average 1.18 packets/second that could be successfully received by the gateway on the same network density. In other words, the network throughput with SBTS-LoRa protocol is at least 8 times greater than the achieved throughput with other protocols. In fact, this emphasizes how changing the access method from ALOHA to TDMA is vital in terms of network throughput. In fact, having a timeslot for each node for transmission instead of the random access to the medium will inevitably decrease collisions, as demonstrated in Fig. 11. Specifically, with large number of connected nodes that are scattered in a wide-coverage area, which is the case of LPWA networks, nodes will not just encounter destructive collisions, where all collided packets are destroyed, but also encounter the capture effect, where strong signals are dominant on weak ones. This in fact limits the scalability of the network, which is considered as a key challenge in LoRa networks as mentioned in Section 1.1. Furthermore, TDMA methods are best suited for high density networks, where timeslots are efficiently used by nodes, and hence the end-to-end delay is minimized. Consequently, using TDMA method will increase the network throughput and hence, the network scalability will be enhanced.

According to Fig. 13, the throughput of SBTS-LoRa protocol is generally increasing with the increased number of nodes. However, at N = 3000 nodes, the throughput is slightly decreased, which reflects the increase of the end-to-end delay at the same node density as shown in Fig. 12. As known, while the end-to-end delay increases, the overall throughput of the network will decrease. As explained previously in Section 5.2, the end-to-end delay is directly affected by SF distributions, which depends on node's locations.

Indeed, despite the fact that EXPLoRa-AT and LoRaWAN are achieving lower end-to-end delays than our protocol (Fig. 12), they end up with lower throughput as the impact of collision is much important (Fig. 11). Specifically, the throughput of EXPLoRa-AT is decreasing with the increase of the network density due to the increase of the probability of collisions (Fig. 11). On the other hand, the throughput of LoRaWAN and BRR is almost stabilized even with the increase of the network density due to the stabilization of collisions as the network is saturated (see

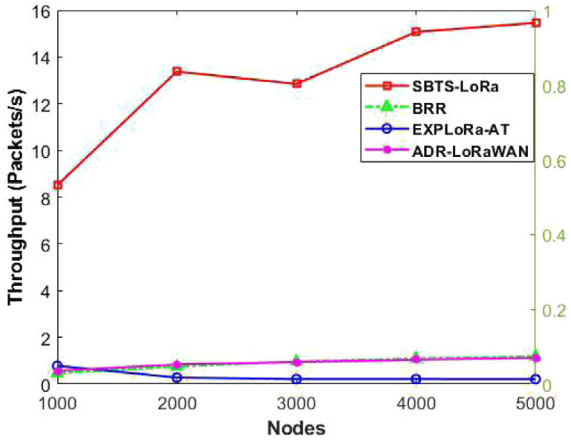

Fig. 13. The network throughput.

Fig. 11). However, it is worth pointing out that the throughput of SBTS-LoRa protocol depicts a clear increase with the increase of the number of nodes. Thus, our protocol is much more scalable compared to other protocols.

5.4. The packet error rate (PER)

The PER is the ratio of the total number of packets that are received under the gateway sensitivity. Since EXPLoRa-AT and LoRaWAN use centralized approaches in how they assign SFs to nodes, they have the lowest PER compared to the distributed ones. In fact, the server in centralized network has better knowledge about the network. Consequently, the distribution of SFs and other transmission parameters is more accurate. This is true for ADR-LoRaWAN algorithm as shown in Fig. 14 . However, unlike the ADR-LoRaWAN, the PER of EXPLoRa-AT is almost zero as all nodes start their traffic using SF12 and then change their SFs if they received downlink communication from the server that contains the new selected SF. Note that the server in this algorithm only receives traffic from a very small portion of nodes (Fig. 2), that are close to the gateway. As a result, only those nodes use smaller SFs than SF12 while all the remaining ones are using SF12. Hence, if no collision happens, the delivery of EXPLORA-AT is guaranteed as nodes mostly use high SFs with the maximum transmission power. On the other hand, BRR is achieving the highest PER as nodes randomly choose the transmission power level and the SFs selection is based on a bounded random variable x, as explained previously. The PER of SBTS-LoRa increases with extreme high number of nodes (more than 4000 nodes), as more nodes could inappropriately select SFs and/or transmission power level. Although the PER of our protocol increases with the increase of the number of connected nodes, this is not affecting the overall network throughput (Fig. 13).

5.5. The energy consumption (energy per bit)

Fig. 15 shows the total energy consumed by nodes to successfully deliver one bit as a function of the number of nodes. Obviously EXPLoRa-AT has the highest energy consumption since it has the highest collisions (Fig. 11). On the other hand, the energy consumption of both ADR-LoRaWAN and BRR algorithms is increasing with the increase of the network density, which is mainly due to their high probability of collisions (Fig. 11), the high packet error rate (Fig. 14), and the high end delays (Fig. 12). However, SBTS-LoRa protocol achieves the lowest energy consumption since it has the lowest probability of collisions.

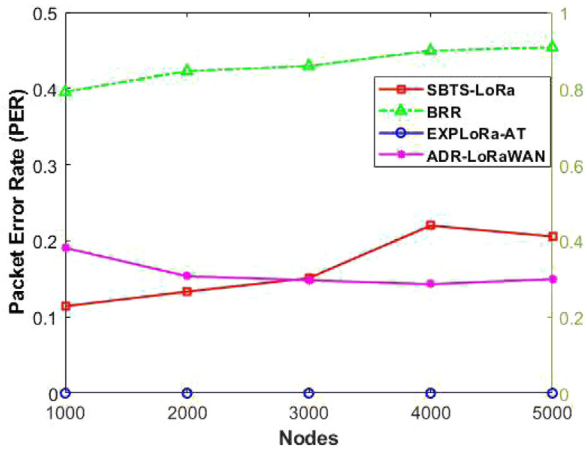

Fig. 14. The packet error rate (PER).

Table 4
A summary of performance metrics .

	Collisions	End-to end delay (s)	Throughput (packets/s)	PER	Energy per bit (J)
SBTS-LoRa	28%	0.47	13.05	16%	15.49
BRR	50%	0.61	0.89	43%	61.30
EXPLoRa-AT	76%	0.19	0.34	0%	257.08
ADR-LoRaWAN	55%	0.48	0.90	16%	54.85

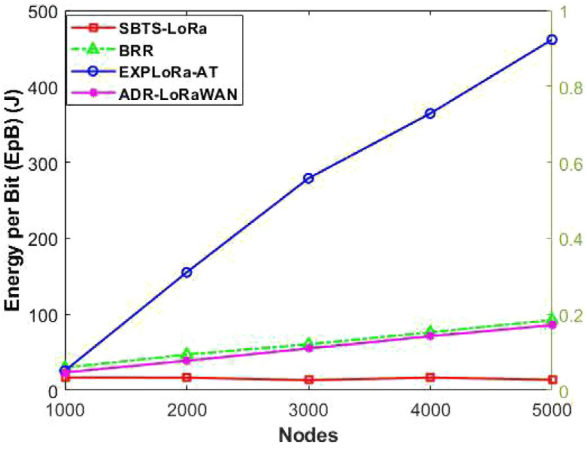

Fig. 15. The energy consumption per bit (EpB).

Furthermore, the energy consumption level of nodes is stable even with the increase of the network density thanks to the efficient use of LoRa transmission parameters and the judicious allocation of timeslots that result in lower collisions. To sum up, Table 4 shows the average of collisions, end delay, throughput, PER, and energy per bit for all evaluated protocols.

6. Conclusion and future work

6.1. Conclusion

This paper develops the SBTS-LoRa, a novel MAC protocol that is targeting large-scale networks. SBTS-LoRa improves the

scalability of the network by mitigating collisions resulting from nodes ALOHA-access method and avoiding downlink communication to control the transmission parameters distribution among nodes. SBTS-LoRa firstly divides the network into six annulus virtual cells such that all nodes on the same cell use the same channel and transmission power level. This will mitigate common issues such as the capture effect. Then, each cell is further divided into a number of annulus sub-cells that match the number of eligible SFs for a given cell. On top of that, each cell is further divided into sectors where the sector ID represents the timeslot ID for which a node transmits. Accordingly, nodes autonomously determine their transmission parameters and timeslot by only recognizing gateway and its coordinates. Simulation results show tremendous enhancements in collision rate, network throughput, and energy consumption. Specifically, for large-scale dense networks, the average throughput of SBTS-LoRa protocol is about 13 times better than the ADR-LoRawan.

6.2. Future work

Going forward, we will further enhance the network performance by considering a TDMA dynamic frame size instead of the static one. In fact, assuming static features for IoT networks may not be realistic for some scenarios. By adopting different node densities for cells with different areas, we can have dynamic frame length for each cell. In this case, we avoid having very long frame length for cells with less node density. This will further enhance the end-to-end delay and hence the network throughput is improved.

Moreover, further investigation needs to be accomplished in order to mathematically derive the optimal distribution of SFs among nodes for any random network topology. Indeed, analytical derivation of the optimal distribution of LoRa features will highly increase the network performance. Recall that, the main features of LoRa are: spreading factors, transmission powers, communication channels and coding rate. Deriving mathematically the optimal distribution of these parameters in any random network topology will be a huge scientific step that will clearly open new research directions in LoRaWAN. Finally, implementing our protocol in a real experimental testbed will give further insights into the protocol contributions and limitations.

CRediT authorship contribution statement

Hanan Alahmadi: Conceptualization, Methodology, Software, Validation, Investigation, Writing – original draft, Visualization. **Fatma Bouabdallah:** Conceptualization, Methodology, Writing. **Ahmed Al-Dubai:** Conceptualization, Methodology, Writing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors extend their appreciation to the Deputyship for Research and Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number (IFPRC-065-612-2020) and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

References

[1] Alireza Ghasempour, Internet of things in smart grid: Architecture, applications, services, key technologies, and challenges, Inventions 4 (1) (2019) 22.

[2] Usman Raza, Parag Kulkarni, Mahesh Sooriyabandara, Low Power Wide Area networks: An overview, IEEE Commun. Surv. Tutor. 19 (2) (2017) 855–873, http://dx.doi.org/10.1109/comst.2017.2652320.

[3] J. Lee, LTE-Advanced in 3GPP rel -13/14: an evolution toward 5G, IEEE Commun. Mag. 54 (3) (2016) 36–42.

[4] Hai Wang, Abraham O. Fapojuwo, A survey of enabling technologies of low power and long range machine-to-machine communications, IEEE Commun. Surv. Tutor. 19 (4) (2017) 2621–2639, http://dx.doi.org/10.1109/comst.2017.2721379.

[5] Augustine Ikpehai, Bamidele Adebisi, Khaled M. Rabie, Kelvin Anoh, Ruth E. Ande, Mohammad Hammoudeh, Haris Gacanin, Uche M. Mbanaso, Low-Power Wide Area network technologies for internet-of-things: A comparative review, IEEE Internet Things J. 6 (2) (2019) 2225–2240, http://dx.doi.org/10.1109/JIOT.2018.2883728.

[6] LoRa Alliance, What is lorawan, 2020, URL https://lora-alliance.org/resource-hub/what-lorawanr.

[7] Jothi Prasanna Shanmuga Sundaram, Wan Du, Zhiwei Zhao, A survey on lora networking: Research problems, current solutions, and open issues, IEEE Commun. Surv. Tutor. 22 (1) (2019) 371–388.

[8] Ferran Adelantado, Xavier Vilajosana, Pere Tuset-Peiro, Borja Martinez, Joan Melia-Segui, Thomas Watteyne, Understanding the limits of lorawan, IEEE Commun. Mag. 55 (9) (2017) 34–40, http://dx.doi.org/10.1109/mcom.2017.1600613.

[9] LoRaTools, Air time calculator LoRaTools, 2021, URL https://loratools.nl/#/airtime.

[10] M. Slabicki, G. Premsankar, M. Di Francesco, Adaptive configuration of lora networks for dense IoT deployments, in: NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium, 2018, pp. 1–9.

[11] F Cuomo, M Campo, A Caponi, G Bianchi, G Rossini, P Pisani, . Explora, Extending the performance of LoRa by suitable spreading factor allocations, in: 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2017, pp. 1–8.

[12] B. Reynders, W Meert, S. Pollin, Power and spreading factor control in low power wide area networks, in: 2017 IEEE International Conference on Communications (ICC), 2017, pp. 1–6.

[13] J. T Lim, Y. Han, Spreading factor allocation for massive connectivity in lora systems, IEEE Commun. Lett. 22 (4) (2018) 800–803.

[14] D Zorbas, G Z Papadopoulos, P Maille, N Montavont, C Douligeris, Improving LoRa network capacity using multiple spreading factor configurations, in: 2018 25th International Conference on Telecommunications (ICT), 2018, pp. 516–520.

[15] K.Q. Abdelfadeel, V. Cionca, D. Pesch, Fair adaptive data rate allocation and power control in lorawan, in: 2018 IEEE 19th International Symposium on "a World of Wireless, Mobile and Multimedia Networks" (WoWMoM), 2018, pp. 14–15.

[16] Tommaso Polonelli, Davide Brunelli, Achille Marzocchi, Luca Benini, Slotted aloha on lorawan-design, analysis, and deployment, Sensors 19 (4) (2019) 838.

[17] Brecht Reynders, Qing Wang, Pere Tuset-Peiro, Xavier Vilajosana, Sofie Pollin, Improving reliability and scalability of LoRaWANs through lightweight scheduling, IEEE Internet Things J. 5 (3) (2018) 1830–1842, http://dx.doi.org/10.1109/jiot.2018.2815150.

[18] Luca Leonardi, Filippo Battaglia, Lucia Lo Bello, RT-Lora: A medium access strategy to support real-time flows over lora-based networks for industrial IoT applications, IEEE Internet Things J. 6 (6) (2019) 10812–10823, http://dx.doi.org/10.1109/jiot.2019.2942776.

[19] Jetmir Haxhibeqiri, Ingrid Moerman, Jeroen Hoebeke, Low overhead scheduling of LoRa transmissions for improved scalability, IEEE Internet of Things Journal 6 (2) (2019) 3097–3109, URL.

[20] Khaled Q. Abdelfadeel, Dimitrios Zorbas, Victor Cionca, Dirk Pesch, $Free$ —fine-grained scheduling for reliable and energy-efficient data collection in lorawan, IEEE Internet of Things Journal 7 (1) (2020) 669–683, URL.

[21] Junhee Lee, Young Seog Yoon, Hyun Woo Oh, Kwang Roh Park, Dg-lora: Deterministic group acknowledgment transmissions in lora networks for industrial IoT applications, Sensors 21 (4) (2021) 1444.

[22] D. Zorbas, B. O'flynn, Autonomous collision-free scheduling for lora-based industrial internet of things, in: 2019 IEEE 20th International Symposium on "a World of Wireless, Mobile and Multimedia Networks" (WoWMoM), 2019, pp. 1–5.

[23] Dimitrios Zorbas, Khaled Abdelfadeel, Panayiotis Kotzanikolaou, Dirk Pesch, TS-Lora: Time-slotted LoRaWAN for the industrial internet of things, Comput. Commun. 153 (2020) 1–10, http://dx.doi.org/10.1016/j.comcom.2020.01.056.
[24] Hanan Alahmadi, Fatma Bouabdallah, Ahmed Al-Dubai, A new annulus-based distribution algorithm for scalable IoT-driven lora networks, in: ICC 2021 - IEEE International Conference on Communications, 2021, pp. 1–6, http://dx.doi.org/10.1109/ICC42927.2021.9500407.
[25] Semtech Corporation, Sx1272/3/6/7/8: LoRa modem. Designer's guide, 2013, URL shorturl.at/hwzHY.
[26] D. Zorbas, Design considerations for time-slotted LoRa(WAN), in: Proceedings of the 2020 International Conference on Embedded Wireless Systems and Networks on Proceedings of the 2020 International Conference on Embedded Wireless Systems and Networks, 2020, pp. 271–276.
[27] OMNeT++ Discrete event simulator, 2018, URL https://omnetpp.org/.
[28] Martin C. Bor, Utz Roedig, Thiemo Voigt, Juan M. Alonso, Do LoRa low-power wide-area networks scale?, in: Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, 2016, pp. 59–67.

Hanan Alahmadi She is currently a Ph.D. student at Edinburgh Napier University, School of computing. She is working as a lecturer at Faculty of Computing and Information Technology, King Abdulaziz University. Her research interests are in the areas of communications, Internet of things networks, Wireless sensor networks, designing MAC and routing protocols taking into account the energy efficiency of sensor nodes.

Fatma Bouabdallah received a B.S. degree in computer engineering from Ecole Nationale des Sciences de l'Informatique (ENSI), Tunis, Tunisia, in 2005 and the M.S. degree in network and computer science from the University of Paris VI, France, in 2006. She attended the doctoral program at the Institut National de Recherche en Informatique et en Automatique, Rennes, France from October 2006 to November 2008. In November 2008, she received her Ph.D. degree in computer Science. In 2010, she was with the University of Waterloo, Canada, as a Postdoctoral Fellow. She is currently an associate Professor in King AbdulAziz University, Jeddah, Saudi Arabia. Her research interests include, IoT design, wireless sensor networks, under-water acoustic sensor networks, crosslayer design, modeling and performance evaluation of computer networks.

Ahmed Y Al-Dubai received his Ph.D. in Computing from the University of Glasgow in 2004. In 2004, he joined the University of West London and then in 2005 he joined Edinburgh Napier University where he became in 2016 a Professor and the Program Leader of the Postgraduate Research Degrees of the School of Computing. He is the Head of the IoT and Smart Systems Research Group. He has been working on the area of group communication algorithms, smart spaces and high-performance wireless and mobile networks. He published widely in world leading journals and in prestigious international conferences. He has been the recipient of several academic awards and recognitions and a member of several editorial boards of scholarly journals. He served as Guest Editor for over 30 special issues in scholarly journals and Chaired and Co-chaired over 30 International conferences/workshops. Prof. Al-Dubai is a Fellow of the British Higher Academy and a Senior Member of the IEEE.