Design Considerations for Time-Slotted LoRa(WAN)

Dimitrios Zorbas
Tyndall National Institute, University College Cork
CONNECT Centre
dimzorbas@ieee.org

Abstract

One of the most common issues in wireless networks is the problem of increasing the network capacity by alleviating or eliminating collisions with the minimum possible cost. The combination of time division protocols together with efficient slot allocation mechanisms is an effective way to achieve scalability and provide high reliability. In this paper, I describe the parameters that must be taken into account when designing LoRa(WAN)-based time-slotted protocols. I show that the LoRaWAN case differs from any other synchronous wireless solution mainly due to the characteristics of the LoRa physical layer and the regional radio duty cycle restrictions. I also propose a frame structure which has been adopted during the implementation of a time-slotted LoRa approach.

1 Introduction

The vision of Industry 4.0 is to reduce operating costs by automating a high number of systems that involve sensors, robots, autonomous vehicles, and humans. This relies on the design of reliable and cost-effective Industrial Internet of Things (IIoT) networking solutions that can achieve low-latency, high packet delivery ratio, and low power consumption.

Current wireless IIoT protocols such as the WirelessHART [1] and the IEEE802.15.4e TSCH [16], suffer from limited mobility due to their short range radio technology. To extend coverage, those technologies rely on multi-hop deployments which, however, exhibit a higher deployment cost. Apart from that, these wireless standards operate in the 2.4GHz ISM spectrum, which is sometimes saturated by other wireless technologies creating significant levels of interference.

In contrast with the current IIoT physical layer solutions, a long range technology such as LoRa can tackle the problem of mobility as well as of the installation cost while exhibiting similar energy demands. However, the current LoRa-based standard, called LoRaWAN, is designed for battery longevity and deployment simplicity. As a consequence, the Aloha-based MAC layer cannot guarantee typical IIoT requirements such as a higher than 99% packet delivery ratio and a guaranteed low delay.

A time-slotted approach is a good candidate to meet the Industry 4.0 requirements. Using time-slotted communications, the collision rate can be reduced or eliminated, which enhances scalability and reliability. However, this does not come without pitfalls. For example, LoRa transmissions in sub-GHz bands are restricted by duty cycle regulations leading to a limited downlink availability, schedule dissemination issues, and thus poor reliability [12].

In this paper, I present a number of key parameters that can affect the design, the cost, and the performance of a time-slotted LoRa approach in the context of the IIoT. I also present implementation details related to the accuracy of slots and the synchronisation. The design parameters are valid for both native LoRa and over LoRaWAN time-slotted approaches.

2 Background

This section is an introduction to LoRa, LoRaWAN, and time-slotted communications. Its purpose is to familiarise the reader with some main concepts needed in the next sections.

2.1 LoRa and LoRaWAN

LoRa (**Lo**ng **Ra**nge) is a proprietary spread spectrum modulation technique developed by Semtech [14]. Its main characteristic is that it can trade data rate with sensitivity by selecting the amount of spread to use. To do so, it makes use of a radio parameter, called Spreading Factor (SF) which typically ranges from 7 to 12. The higher the SF, the higher the sensitivity for the same channel bandwidth (BW) and, thus, the longer the transmission range. Moreover, transmissions performed on different SFs are almost orthogonal to each other increasing the network capacity. However, the data rate decreases substantially as higher SFs and/or narrower BWs are selected. LoRa uses license-free sub-gigahertz radio frequency bands. Depending on the region the central frequency may be the 433MHz (Europe and Asia), the 868MHz (Europe) or the 915 MHz (Australia and North America).

The LoRa Alliance, a non-profit association consisting of Semtech as well as other companies and universities from across the world, have proposed an open standard, called LoRaWAN. LoRaWAN supports device registration, bi-directional communication, end-to-end security, synchronisation, and localisation services [10]. An end-device in LoRaWAN can belong to one of the following three classes. Class A devices are those whose transmissions are performed at sparse intervals utilising the minimum possible amount of energy. For this reason, the MAC layer of class A end-devices is Aloha-based. Every transmission can optionally be followed by two downlink receive windows. Class A devices constitute the majority of nodes in a LoRaWAN network. Class B devices have extra receive windows that are scheduled by the gateway using time-synchronised beacons. The purpose of this class of devices is to have the nodes available for reception at predictable times (e.g., for over-the-air firmware update purposes). Class B operation is still low-power since the nodes only wake-up at periodical pre-defined times. Finally, Class C devices are devices that continuously listen for incoming data when they are not transmitting. They are usually devices with unlimited power resources.

2.2 Time-slotted Communications in IIoT

Time-slotted access is a channel access method for shared medium networks. It is a time-division method that allows multiple users to share the same frequency channel. It achieves this by slicing the time in slots whose length usually depends on the payload size and specific radio parameters (e.g., transfer rate). The most common time-slotted technique is the Time Division Multiple Access (TDMA) which is widely used in cellular telecommunication systems.

In the context of IIoT and low-power networks, time-slotted access appears in several protocols. The Time Slotted Channel Hopping (TSCH) and the WirelessHART are the most common ones. Both protocols are designed to operate over the IEEE802.15.4 physical layer in the 2.4GHz ISM band. Due to the short range feature of the physical layer, TSCH can extend coverage by organising the nodes in multi-hop deployments. Thus, a routing mechanism, such as the RPL [17], is needed to calculate routes initiated by the gateways. An IPv6 stack has recently been proposed by IETF 6TiSCH group to achieve synchronisation and define network joining methods and security mechanisms [16].

IIoT protocols follow a different medium access approach compared to traditional TDMA protocols. In a typical TDMA protocol, a node requests access to the medium every time it desires to transmit data, and a number of slots is assigned to it depending on the available resources. However, in IIoT protocols, the interaction between the gateway and the nodes is limited in order to minimise energy consumption and delay. Hence, transmissions between a pair of nodes are performed on strict timeslots after following a schedule which defines the exact timings to turn on/off the radio. Every timeslot accommodates a uni-directional transmission as well as an acknowledgment. Successive transmissions are performed by repeating frames which actually repeats the computed schedule. A channel hopping technique is also applied to bounce successive transmissions over different channels and, thus, mitigate potential external interference on specific radio channels.

Table 1. Transmission power (TP) and duty cycle regulations per sub-band for the EU868 band.

Frequency		TP	Duty Cycle
K	863 – 865 MHz	25 mW ERP	$\leq$ 0.1% or LBT
L	865 – 868 MHz	25 mW ERP	$\leq$ 1% or LBT
M	868 – 868.6 MHz	25 mW ERP	$\leq$ 1% or LBT
N	868.7 – 869.2 MHz	25 mW ERP	$\leq$ 0.1% or LBT
O	869.4 – 869.65 MHz	500 mW ERP	$\leq$ 10% or LBT
P	869.7 – 870 MHz	5 mW ERP	No requirement
Q	869.7 – 870 MHz	25 mW ERP	$\leq$ 1% or LBT

Table 2. LoRaWAN EU uplink and downlink channels.

Uplink #	Downlink/ Uplink (D/U)	Downlink phase	Frequency (MHz)	BW (kHz)
1	D/U	RX1	868.1	125
2	D/U	RX1	868.3	125, 250
3	D/U	RX1	868.5	125
4	D/U	RX1	867.1	125
5	D/U	RX1	867.3	125
6	D/U	RX1	867.5	125
7	D/U	RX1	867.7	125
8	D/U	RX1	867.9	125
	D	RX2	869.525	125

3 Enabling LoRa(WAN) Synchronous Transmissions

In this section, I discuss a number of parameters and design characteristics that need to be taken into account when shifting from the Aloha-based LoRaWAN to a time-slotted LoRa(WAN) protocol. I also describe the reasons why current IIoT protocols cannot directly be used in a LoRa(WAN) environment.

3.1 Radio Duty Cycle and Transmit Power Restrictions

LoRa currently operates in the sub-GHz ISM band, thus, its transmissions are subject to strict duty cycle and transmission power regulations. In Europe LoRa devices use the 868MHz ISM band which is divided in several sub-bands [6]. Each of these sub-bands has its own rules on duty cycle and maximum transmit power as it is summarised in Table 1. Furthermore, Table 2 lists the LoRaWAN uplink and downlink channels along with the list of SFs and channel bandwidth for each channel. Comparing the two tables, we can see that all uplink and downlink LoRaWAN channels have a 1% duty cycle, except of the RX2 downlink channel which has a 10% duty cycle.

The duty cycle restriction is the root of many issues in a LoRaWAN system such as limited gateway availability, transmission delays, and long registration times. I will describe those issues in the next subsections.

3.2 Unequal Slot Length

Unlike other radio technologies, LoRa transmission time (for the same payload) increases as we switch to higher SFs. This means that the slot length cannot be equal for all the transmissions unless all the nodes use the same SF. Dividing the time in equal slots of a maximum length (e.g., based on SF12) is not an efficient solution since this will lead to excessive transmission delays and small capacity.

A solution to this problem would be to have multiple frames, each of them dedicated to one of the available SFs. Since LoRa SFs are almost orthogonal, the frames can run in

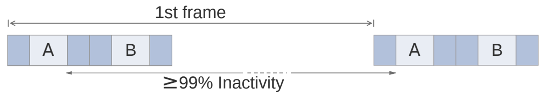

Figure 1. A frame accommodating two slots.

parallel. The solution can exhibit some collisions due to the inter-SF interference, which can be eliminated by either adjusting the transmission power of the nodes [4, 3], or reserving a different channel for each frame (SF) [18]. However, both solutions may require adjustments on the nodes' side.

3.3 Acknowledgments

Unlike conventional time-division protocols, in LoRaWAN, the gateway cannot always acknowledge all the transmissions due to the extremely low radio duty cycle. Even with a 10% duty cycle channel, the number of supported (i.e., acknowledged) slots would still be very limited. This causes many reliability issues as well as a waste of energy in case of re-transmissions because the nodes are not always aware of the delivery status of their previously transmitted packets. Moreover, the problem is enhanced by the fact that LoRa links are half-duplex, so a gateway cannot transmit and receive at the same time.

In a synchronous environment this problem can be tackled by grouping the acknowledgements in a single packet and transmitting it in the end of the frame [3, 7]. Since the acknowledgment is a short piece of information, hundreds of nodes may fit into a single packet. An inevitable drawback of this solution is that all nodes have to wake-up at that timeslot to receive the acknowledgments, thus, precise synchronisation and some additional energy cost are required.

3.4 Frame Size

As I explained in Section 2.2, in a time-slotted environment, the time is divided in frames and each frame accommodates a number of slots. Assuming that a slot can be reserved by only one node, one could say that the number of slots per frame has to be equal to the number of nodes. However, due to the duty cycle rules, this is not always possible. Assuming a 1% duty cycle, a node is allowed to transmit again after 99 times the transmission time of the last transmission. As a result, the empty space between two successive transmissions must be filled up with either other transmissions (slots) or just empty slots [18], as it is depicted in Fig.1. This practically means that a minimum frame length must be used when the number of transmissions is not adequate to fill up the empty space. Assuming a single downlink slot per frame and two guard times per slot, the frame size can be computed by the following equation:

$$frame\_size = \begin{cases} 100T, & \text{if } n \leq \left\lceil \dfrac{100T}{T+2g} \right\rceil, \\ n(T+2g)+DL, & \text{if } n > \left\lceil \dfrac{100T}{T+2g} \right\rceil, \end{cases} \quad (1)$$

where T is the data transmission time, g the guard time, DL is the downlink slot length, and n the number of nodes in the frame.

Figure 2 illustrates the frame size for different node populations and SFs, assuming a data packet of 50 Bytes, a guard time of 15ms and a DL size of $\lceil n/8 \rceil$ bytes. We can see that approximately 90 nodes (slots) can be accommodated before reaching the duty cycle limit and then start expanding the frame. Moreover, due to the unequal slot length, a SF increase of 1 unit leads to a frame of an almost double size.

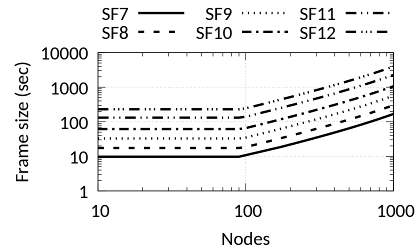

Figure 2. Frame size for different node populations and Spreading Factors.

3.5 Application Duty Cycle

A time-slotted system must be designed in a way that satisfies the application data generation periodicity. For example, if a particular application requires a packet generation every 30 seconds, the maximum frame length cannot exceed that value. This causes a network capacity problem since it limits the number of slots that need to be reserved per frame.

As we saw in the previous subsection, the frame size increases rapidly as we switch to higher SFs and by adding a high number of slots. This means that as the frame size increases, the application duty cycle that can be satisfied increases as well. Thus, certain applications may not be supported by all the SFs or may be supported by frames with a low number of nodes.

3.6 Large Number of Registrations

A critical issue of LoRaWAN in an industrial (time-slotted) environment – which is also related to the duty cycle restrictions – is the registration of a large number of devices. Imagine the case where many devices are powered-up (almost) simultaneously and all those devices send join requests to the gateway. Since the nodes' transmissions are not yet allocated to slots, this process is Aloha-based, thus, a high number of collisions may happen. Apart from that, if a node does not get a response due to collisions or due to duty cycle restrictions of the gateway, it has to wait a long time until it starts transmitting another batch of requests[1]. As a consequence, the registration process through the traditional over the air activation mechanism of LoRaWAN (OTAA) may take significant amount of time and can cause a waste of energy and time. A simple solution to the problem would be to add several gateways in order to increase responsiveness, increasing however the cost of the deployment.

3.7 Scheduling

One of the fundamental issues of time-division protocols is the scheduling of transmissions. The scheduling problem is translated to a problem of computing the most convenient/fair arrangement of the transmissions in slots so that no collisions (or the minimum possible number of collisions) occur and the maximum possible number of nodes is accommodated within the frame structure.

[1] a total time of 36 seconds is allowed per hour with 1% duty cycle.

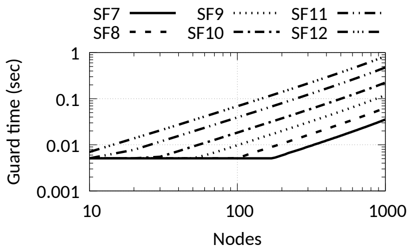

Figure 3. Theoretical guard time for different node populations and Spreading Factors.

However, the most critical issue of scheduling in LoRaWAN, is the dissemination of the schedule. Due to the limited downlink availability of the gateway (due to the duty cycle restrictions), transmitting the schedule to the nodes may be a time-consuming task. For example, imagine the case where a number of nodes has already reserved some slots and new joining nodes need to be added into the frame. The gateway must find an efficient way to expand the frame and fit the additional nodes in it and, at the same time, to let the rest of the nodes know about the frame expansion. Fitting all this information into one or a few only downlink slots is not an easy task when hundreds of nodes exist in the network.

3.8 Clock Drift & Synchronisation

As with all traditional time-division protocols, guard times need to be added between successive slots to tolerate small de-synchronisations due to the clock drift of the nodes. In LoRaWAN, the guard times may need to be longer – compared to other IIoT protocols – due to the longer times, and thus, the longer timeslots and frames. Apart from that, the system designers must take into account the unequal slot length per SF, the number of nodes per frame, and the synchronisation periodicity. Since the timeslots of higher SFs are longer, the guard times must be adjusted accordingly for those transmissions. Moreover, the guard times may be multiple times higher if the synchronisation periodicity happens every several frames and not at every frame.

Figure 3 depicts the guard time for different node populations and SFs assuming that the synchronisation occurs at every frame. The results are based on experimental values using the TS-LoRa implementation [2] with a payload of 50 Bytes and assuming a linear clock drift increase over time. We can observe that in case of hundreds of nodes, the guard time can reach 1 second. As it was described in Section 3.4, this is due to the increased frame size as the number of nodes increases which causes a guard time adjustment as well.

3.9 Propagation Time

Since LoRa is a long range technology the signal travelling time may affect the timeslot length. For example, a node located a few meters away from the gateway will be able to send a packet with no delays. However, assuming that signals travel at the speed of light, a node located 5km away will do it with a delay of approximately 17μs, leading to a potential violation of the timeslot bounds.

There are two solutions to this problem. The first one is for each node to calculate its distance to the gateway and adapt its wake-up time accordingly as devices usually do in 2G networks. This solution exhibits extra overhead because it requires extra transmissions as well as timestamps to be included in the packets. An easier solution is to incorporate a maximum propagation delay (based on a maximum deployment distance) into the guard time. This will slightly increase delay for all the nodes but will exhibit zero overhead and much less programming complexity.

3.10 Battery Lifetime

Battery lifetime is an important issue of all time-slotted technologies due to the extra cost of synchronisation. This is because a node must periodically turn its radio on to receive the synchronisation packet. It is obvious that the shorter this packet, the lower the energy consumption. Thus, it is important to include in it as little amount of information as possible so that the minimum number of bytes is transmitted. For example, in LoRaWAN Class B, the end devices may use the beacon periodicity to calibrate their clock rather than sending timestamps as other protocols do [16].

Moreover, in the case of acknowledgments, if multiple acknowledgments are grouped in a single packet as previously mentioned, the data must be encoded efficiently permitting a low decoding computation cost together with short payloads.

3.11 Multiple Gateways

It would be interesting to see how multi-gateway deployments could be achieved in a time-slotted environment in order to extend coverage. In LoRaWAN, a transmission may be received by multiple gateways. In this case, a global synchronisation mechanism among gateways may be required so that those transmissions do not interfere with nodes registered in a different gateway. For example, overlapping gateways may share some common slots.

Moreover, nodes may be moving between different gateways. This will require the development of roaming mechanisms that will allow a smooth transition between different coverage areas. In such a situation, the frame size must dynamically be adjusted to adapt to topology changes.

3.12 Security

In LoRaWAN, authentication, integrity, and encryption are achieved by using a couple of security keys that are either pre-installed on the nodes (ABP) or generated during the OTAA registration mechanism. The second method is more secure because the keys are negotiated with every activation and are valid only for the specific session. However, since a session typically expires after some hours (it depends on the configuration), the devices need to re-register in order to renew the keys. Thus, an efficient mechanism is needed here to ensure that the registration is done without interrupting the data collection process and without violating the duty cycle rules.

Furthermore, time-slotted transmissions are more vulnerable to selective jamming attacks. For example, an attacker can synchronise with the network and jam the downlink synchronisation slot causing a network desynchronisation. This problem can be tackled by applying a channel hopping approach similar to TSCH [15]. Other important issues are the encryption of the downlink data for one-to-many communications and the key distribution mechanism to multiple end-devices. For instance, LoRaWAN Class B specifications

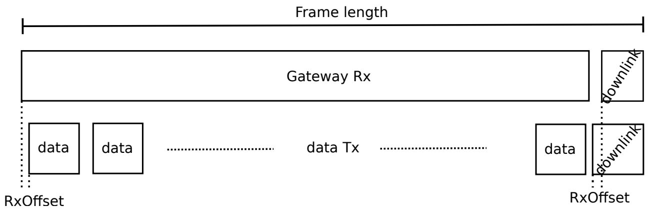

Figure 4. Overview of the frame bounds for the gateway (upper) and the nodes (lower).

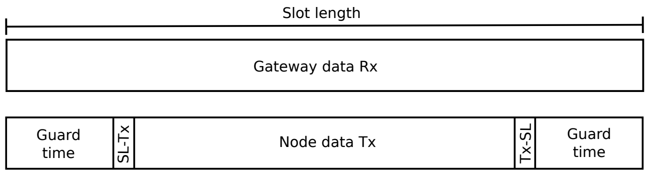

Figure 5. Uplink slot structure.

mention that downlink multicast transmissions must be associated with the corresponding encryption keys but do not specify means to remotely setup such a multicast group or securely distribute the required multicast key material [10].

4 Proposed Frame Structure

The purpose of this section is to give some implementation insights as they were captured through the development of the TS-LoRa platform [2]. The platform is programmed in Micropython programming language and it is tested on Pycom Lopy4 nodes[2]. It allows collision-free transmissions over different SFs, synchronisation, and acknowledgments for all the transmissions. Its frame structure consists of several uplink slots and one downlink slot in the end of the frame. The frames are repeated over time. The downlink slot is used for both synchronisation and acknowledgments. Due to the limited size of this paper, the discussion is limited to the slot/frame time accuracy and the time synchronisation.

4.1 Slot and Frame Time Accuracy

In a time-slotted environment, receivers and transmitters must be perfectly synchronised to successfully carry out the data transfer. We also need to make sure that the receiver radio is switched to the receive mode (i.e., Rx) when the other side starts transmitting. To avoid a partial packet reception, nodes' transmissions can start with a delay of $RxOffset$ as it is depicted in Figure 4. $RxOffset$ mainly depends on the selected guard time but other parameters can affect its length such as (a) the time needed to switch the radio from the receive to transmit mode (i.e., Tx) and (b) the processing time the gateway needs to encode the downlink packet or the nodes to decode it.

[2] https://pycom.io

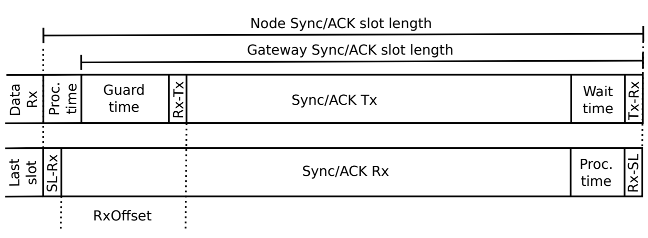

Figure 6. Downlink slot bounds and structure for the gateway (upper) and the nodes (lower).

The uplink slot structure is depicted in Figure 5. The slot consists of two guard times added in the two edges of the slot and the data transmission. The rationale of using two guard times is to avoid transmission overlaps in the case two successive slot transmissions drift to different directions; one positively and one negatively. Some extra short time is spent to switch radio mode.

Figure 6 illustrates the timing template for the downlink slot. We can see that a total delay defined by the gateway processing time and the guard time exists between the two devices. This delay must be appropriately adjusted by taking into account the maximum processing time of the gateway and the maximum drift of the clock for the given frame size. After the downlink data transmission, the nodes spend some time for processing. The processing time may be much longer than the gateway's one earlier because of the lower processing capabilities of the nodes. During this processing time a waiting time (not longer than the nodes' processing time) is added on the gateway side so that the latter does not start a new frame too early. The implementation must also take into account, the time needed to switch from the receive to transmit radio mode ($Rx - Tx$) (or the reverse ($Tx - Rx$)) or from the sleep mode to receive mode ($SL - Rx$) (or the reverse ($Rx - SL$)).

4.2 Time Synchronisation

Time synchronisation is achieved by allowing all nodes to have their radio on during the downlink slot. Because the nodes' clock may have drifted since the last synchronisation, the guard time must be appropriately selected so that none of the nodes wakes-up outside the $RxOffset$ limits. Once the nodes receive the sync packet, their clock is already calibrated in relation with the gateway's clock since the sync packet transmission is broadcasted and, thus, the packet reception is a process that all nodes do at the same time (assuming that the signal propagation time is negligible). However, there is an issue that appears when heterogeneous nodes exist in the network. Since the downlink slot may be used for other purposes as well (e.g., transmissions of acknowledgements), the nodes require some time to process data. Having nodes with different computational capabilities will lead to desynchronisations. Thus, some nodes need to wait for a short time interval after the reception of the sync packet to ensure that all of them will proceed to the new frame simultaneously. I must mention here, that this process assumes that the nodes use the gateway's clock as a reference to calibrate their clock. In case of multiple gateways, each of those gateways must have a similar background process to correct its clock according to a more global clock (e.g., application server clock).

5 Related Work

In this section, I briefly survey some recent studies aiming to improve LoRaWAN capacity and alleviate or eliminate collisions, through the adoption of a time-slotted environment.

Scheduling solutions in such an environment have been considered to improve scalability [9, 3] or to minimise the data collection time [18, 3]. This is done by organising transmissions in time space so that the minimum possible number of collisions occur. Lee *et al.* [9] schedule the transmissions of class B end-devices. However, the overhead of schedule dissemination is neglected. The problem of scheduling transmissions in order to achieve bulk data collection when a gateway is not available at all times is studied in [3] and [18]. On-the-fly as well as centralised scheduling algorithms are proposed. In the first case, the nodes are assigned with a SF and a slot as soon as they registered in the gateway, while in the second case a more global system information is required. The results show that the approaches can extend a typical LoRaWAN network performance by many times, however the on-line registration or the dissemination of the schedule may be two very time-consuming tasks.

Ebi *et al.* [5] study an approach where repeaters are used to synchronise a group of underground nodes. The repeaters are connected to a typical LoRaWAN network. The synchronisation is based on packet flooding and its overhead as well as the duty cycle bounds are not discussed in the paper. Autonomous and collision-free slot assignment with the minimum possible overhead is a step forward since this solution does not require dissemination of the schedule [19]. The approach is promising, however, it leaves many empty slots within the frame structure increasing delay.

Time-synchronised approaches that reduce but not eliminate collisions have also been proposed in the literature [13, 8, 11]. For example, a probabilistic data structure to share slots within the schedule is presented by Haxhibeqiri *et al.*, while a slotted-Aloha version over LoRaWAN is proposed in [11].

LoRa(WAN) time-slotted communications have been only superficially studied in the literature. None of the previously mentioned works presents a complete time-slotted solution including a registration mechanism, the computation of the schedule and its dissemination (if any), acknowledgments, as well as performance guarantees in terms of delay and packet delivery ratio. Moreover, the majority of the works have not been tested experimentally but are based on theoretical hypotheses and computer simulations. A radical and complete solution to allow long range time-slotted communications is still not available.

6 Conclusions

LoRaWAN is a long distance IoT protocol designed for deployment simplicity and energy efficiency. However, due to its Aloha MAC, it is not suitable for applications that require very high reliability and guaranteed delay such as the Industrial IoT applications. Switching to a time-slotted MAC is not an easy task mainly due to the characteristics of the LoRa physical layer and the radio duty cycle regulations. In this paper, all those parameters and characteristics that need to be taken into account when designing such a LoRa(WAN)-based time-slotted approach are described. Moreover, a time-slotted frame structure is proposed based on the experience acquired by developing an experimental time-slotted LoRa platform.

Acknowledgements

This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) and is co-funded under the European Regional Development Fund under Grant Number 13/RC/2077 and the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 713567.

7 References

[1] WirelessHART Specification 75: TDMA Data-Link Layer. *HART Communication Foundation Std., Rev. 1.1,*, HCF SPEC-75, 2008.

[2] TS-LoRa. https://github.com/deltazita/ts-lora, 2019.

[3] K. Q. Abdelfadeel, D. Zorbas, V. Cionca, and D. Pesch. FREE–Fine-grained Scheduling for Reliable and Energy Efficient Data Collection in LoRaWAN. *IEEE Internet of Things Journal*, pages 1–1, 2019.

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

[5] C. Ebi, F. Schaltegger, A. Rüst, and F. Blumensaat. Synchronous LoRa Mesh Network to Monitor Processes in Underground Infrastructure. *IEEE Access*, 7:57663–57677, Sep 2019.

[6] ETSI EN 300 220-2. Short Range Devices (SRD) operating in the frequency range 25 MHz to 1 000 MHz; Part 2: Harmonised Standard for access to radio spectrum for non specific radio equipment. www.etsi.org/deliver/etsi_en/300200_300299/30022002/03.02.01_30 /en_30022002v030201v.pdf, 2018. Online; accessed 18-Oct-2019.

[7] Y. Hasegawa and K. Suzuki. A multi-user ack-aggregation method for large-scale reliable lorawan service. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, 2019.

[8] J. Haxhibeqiri, I. Moerman, and J. Hoebeke. Low overhead scheduling of LoRa transmissions for improved scalability. *IEEE Internet of Things Journal*, 6(2):3097–3109, 2018.

[9] J. Lee, W. Jeong, and B. Choi. A Scheduling Algorithm for Improving Scalability of LoRaWAN. In *International Conference on Information and Communication Technology Convergence (ICTC)*, pages 1383–1388. IEEE, Oct 2018.

[10] LoRa Alliance Technical Committee. LoRaWAN^{TM} 1.0.3 Specification. lora-alliance.org/sites/default/files/2018-07/lorawan1.0.3.pdf, 2018. Online; accessed 17-Oct-2019.

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

[12] A.-I. Pop, U. Raza, P. Kulkarni, and M. Sooriyabandara. Does bidirectional traffic do more harm than good in LoRaWAN based LPWA networks? In *GLOBECOM 2017-2017 IEEE Global Communications Conference*, pages 1–6. IEEE, 2017.

[13] B. Reynders, Q. Wang, P. Tuset-Peiro, X. Vilajosana, and S. Pollin. Improving Reliability and Scalability of LoRaWANs Through Lightweight Scheduling. *IEEE Internet of Things Journal*, 5(3):1830–1842, June 2018.

[14] Semtech Corporation. AN1200.22, LoRa^{TM} Modulation Basics. www.semtech.com/uploads/documents/an1200.22.pdf, 2015. Online; accessed 10-Oct-2019.

[15] R. K. Singh, R. Berkvens, and M. Weyn. Time synchronization with channel hopping scheme for lora networks. In L. Barolli, P. Hellinckx, and J. Natwichai, editors, *Advances on P2P, Parallel, Grid, Cloud and Internet Computing*, pages 786–797. Springer, 2020.

[16] T. Watteyne, M. Palattella, and L. Grieco. Using IEEE 802.15.4e Time-Slotted Channel Hopping (TSCH) in the Internet of Things (IoT): Problem Statement. RFC 7554, 2015.

[17] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and A. R. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550, 2012.

[18] D. Zorbas, K. Q. Abdelfadeel, V. Cionca, D. Pesch, and B. O'Flynn. Offline Scheduling Algorithms for Time-Slotted LoRa-based Bulk Data Transmission. In *IEEE 5th World Forum on Internet of Things (WFIoT)*, pages 1–6. IEEE, 2019.

[19] D. Zorbas and B. O'Flynn. Autonomous Collision-Free Scheduling for LoRa-based Industrial Internet of Things. In *20th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–5. IEEE, June 2019.