

Received December 8, 2021, accepted January 7, 2022, date of publication January 25, 2022, date of current version February 1, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3146021

SALSA: A Scheduling Algorithm for LoRa to LEO Satellites

MOHAMMAD AFHAMISIS^{1,2}, (Member, IEEE), AND
MARIA RITA PALATTELLA¹, (Member, IEEE)

¹Luxembourg Institute of Science and Technology, 4362 Esch-sur-Alzette, Luxembourg

²Faculty of Science, Technology and Medicine, University of Luxembourg, 4365 Esch-sur-Alzette, Luxembourg

Corresponding author: Mohammad Afhamisis (mohammad.afhamisis@list.lu)

This work was supported by the Design of LoRaWAN protocol optimisation over SATellite Connection for precision agriculture applications (LORSAT) Project, through the National Research Fund Luxembourg (FNR), under Grant CORE/C19/IS/13705191.

ABSTRACT Satellites are going to revolutionize the Internet of Things (IoT) connectivity enabling the ubiquitous coverage targeted by 5G and upcoming 6G networks. While the potentials of satellite IoT are unquestionable, many challenges still remain unsolved. In the present work, we design a Long Range Wide Area Network (LoRaWAN) with mobile gateways (GWs) installed on Low Earth Orbit (LEO) satellites. We address synchronization issues due to intermittent link availability between the End Devices (EDs) and the GW, and we propose a Scheduling Algorithm for LoRa to LEO Satellites (SALSA). SALSA ensures reliable communication, avoiding packets drops and packet collisions, by using a Time Division Multiple Access (TDMA) approach, rather than classic ALOHA-based LoRa. The uplink transmissions of the EDs are scheduled considering the satellite availability period, the satellite footprint, and the specific visit time for each ED. Two different policies are proposed: a First Come First Served (FCFS), and a FAIR policy. Simulation results, obtained in real conditions, with real satellites visibility, demonstrate the better performance achievable with SALSA (regardless of the policy) compared to standard LoRa. The FAIR policy outperforms the FCFS policy by giving all the EDs an equal chance to transmit, even to those that are visited last by the satellite. The performance of SALSA can be further improved considering larger constellations of LEO satellites.

INDEX TERMS Satellite communication, Internet of Things, performance evaluation, optimization methods, scheduling algorithm, LoRaWAN, synchronization, fairness.

I. INTRODUCTION

Satellites are going to revolutionize Internet of Things (IoT) connectivity enabling the ubiquitous coverage targeted by 5G and Beyond 5G networks. Satellite for IoT is becoming more and more affordable, available, and accessible, thanks to the launch of several low-cost miniaturized Low Earth Orbit (LEO) satellites (CubeSats) [1]. Those LEO satellites are the most appealing ones for IoT applications, due to the shorter delay that they introduce ($\simeq 40$ ms), compared to Geostationary Earth Orbit (GEO) satellites ($\simeq 500$ ms). However, their intrinsic orbital properties imply limited visibility time. This issue can be overcome by using large constellations of LEO satellites, able to provide almost continuous coverage; and it will be further solved with relay networks from LEO to GEO satellites. While LEO satellite constellations can

reduce latency to very low levels, compared to GEO satellites, it will still not be suitable to meet the needs of 5G Ultra Reliable Low Latency Communications (URLLC), such as Augmented Reality/Virtual Reality (AR/VR) applications, and Tactile Internet. However, the delay is more than adequate for many other services that do not require extremely low latency, like precision agriculture, maritime surveillance, environmental monitoring, etc.

Moreover, being closer to the Earth, LEO satellites can support direct communication with IoT devices on the ground. Over the last years, several satellite operators and IoT companies have recognized the potential innovation and consequent business opportunities in enabling direct IoT to satellite. Among different IoT technologies, the Long-Range Wide Area Network (LoRaWAN) technology [2] has played, and is still having a significant role in the real deployment of IoT. Companies like Lacuna Space [3] and Wyld [4] have already demonstrated how global IoT connectivity can be

The associate editor coordinating the review of this manuscript and approving it for publication was Rute C. Sofia ^{1b}.

achieved by integrating a LoRaWAN gateway on a LEO satellite.

Satellite LoRaWAN has drawn the attention also of academia, and researchers. Several works have identified the open research challenges and proposed a preliminary network architecture [5]–[8]. In a recent survey [5], the authors have discussed the pros and cons of satellite access for IoT, current limitations, and ongoing initiatives for making it possible. Very few works have provided technical solutions, as enhancement of LoRa PHY modulation [9], and management of Doppler effect [10].

While the use of a satellite gateway increases the coverage range of the LoRaWAN network, it also introduces several challenges due to its limited visibility time. The probability of collision due to the random ALOHA-like access scheme - already high in classical terrestrial LoRaWAN networks [11], [12] - can only get worst since many End Devices (EDs) will compete for the same resource and attempt an uplink transmission during the limited availability of the gateway. To address such a problem and ensure reliable communication, scheduling techniques are needed. Several ones have been proposed in literature for terrestrial LoRaWAN networks [13]–[20]. Zorbas *et al.* tackled the issue of a LoRaWAN mobile gateway, installed on a drone [21], [22]. To deal with the limited availability of the flying gateway, they have proposed to transmit a bulk of data and reduce the ToA for the grouped transmissions. A couple of recent works considered direct LoRa to satellite, but they did not enhance network performance with scheduling techniques. Ullah *et al.* [23] modeled the PHY channel between the EDs and the satellite gateway and assumed EDs transmit according to the ALOHA-like LoRa standard. Also, Tondo *et al.* [24] proposed a system model for direct IoT to satellite, with a constellation of LEO satellites. They evaluated throughput and packet loss rate as a function of the erasure probabilities at each satellite. As in our work, they demonstrated the network performance could improve with larger constellations. Besides that, they assumed the EDs transmit using slotted ALOHA.

In this paper for the first time, we address reliability and scalability issues in LoRa to LEO satellites communication. We propose SALSA, a scheduling algorithm that cancels packet drops and packet collisions by design, and ensures fair access to all the nodes in the network. We have designed the algorithm considering the worst-case scenario (i.e., a single satellite available), while it is still applicable and extendable to LEO constellations. Two different access policies are proposed: a First Come First Served (FCFS), and a FAIR policy. Simulation results, obtained in real conditions, with real satellites visibility, demonstrate the better performance achievable with SALSA (regardless of the policy) compared to standard LoRa. The FAIR policy outperforms the FCFS policy by giving all the EDs an equal chance to transmit, even to those that are visited last by the satellite. The performance of SALSA can be further improved considering larger constellations of LEO satellites.

The remainder of this paper is structured as follows. Section II introduces the system architecture and the background information about the LoRa PHY and LoRaWAN MAC protocols. In Section III, we describe the proposed SALSA algorithm in all the details, including the system assumptions and the scheduling policies. Section IV first presents the simulated environment and then, the performance achievable with SALSA, considering real conditions. Through simulation results, we show that SALSA cancels by design collisions due to random ALOHA MAC, and packet drops due to intermittent link availability. When using the FAIR policy and having a constellation of satellites, the end devices get an equal chance of transmitting several uplink packets. Finally, Sec. V draws the conclusion and future research direction.

II. SYSTEM ARCHITECTURE

We consider a satellite LoRaWAN network, whose architecture is depicted in Fig. 1. Several end-devices (EDs) deployed on the ground can communicate through wireless LoRa links with a LoRaWAN gateway (GW) installed on a LEO satellite. Only the EDs within the satellite footprint can reach the GW.

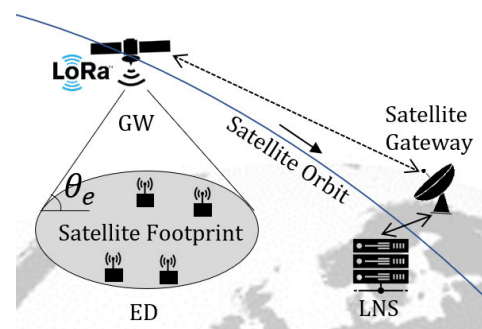


FIGURE 1. LoRaSAT Network Architecture.

The footprint represents the area of the Earth covered by microwave radiation from a satellite dish (transponder). Thus, it corresponds to the coverage area. The size of the footprint depends on several factors: the location of the satellite along its orbit, the shape, and size of the beam produced by its transponder, and the distance from the earth. In particular, the footprint is impacted by the elevation angle θ_e , which is the angle between the beam pointing direction, directly towards the satellite, and the local horizontal plane. As shown in Fig.2, the satellite footprint decreases with increasing θ_e . In the example, the size of the footprint A is smaller than the size of footprint B, being $\theta_{eA} > \theta_{eB}$. The optimum value of θ_e depends on the satellite orbit, the budget link, and the communication link design. To properly schedule uplink transmission from the EDs to the GW, it is of foremost importance to know the movement of the satellite and its relative position to the IoT devices. Only with such information, it is possible ensuring precise synchronization with a fixed θ_e .

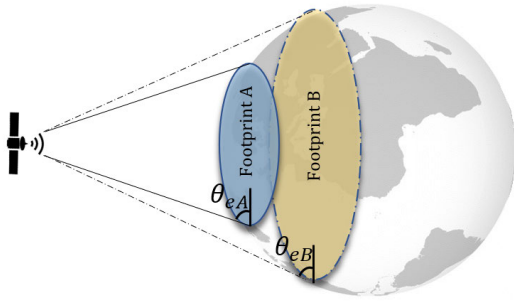


FIGURE 2. Relation between the elevation angle and the satellite footprint. The footprint A is smaller than the footprint B in size being the elevation angle $\theta_{eA} > \theta_{eB}$.

While we describe the architecture with a single satellite and thus a single GW, the system can be extendible to multiple gateways installed on a constellation of LEO satellites.

Multiple GWs are connected to a central LoRaWAN Network Server (LNS) on the ground through a reliable satellite backhauling network (via a ground satellite gateway). The GW acts as a forwarding device. It is responsible only for re-dispatching raw data packets from EDs toward the LNS, encapsulating them in IP packets. The LNS manages the GWs and the entire end-to-end system; it is responsible for the LoRaWAN protocol configuration and execution in the network; it collects and decodes the received packets. In the present work, we focus on the first segment of the end-to-end system, i.e., the intermittent link between the EDs and the LEO satellite(s). We assume there is a ground satellite gateway in the visibility of the satellite, and it ensures the reception of the packet at the LNS on the ground.

A. PHY LAYER

At the Physical (PHY) layer, the communication between EDs and GWs follows the Long Range (LoRa) technology [25] based on a spread spectrum modulation. The LoRa modulation is available with several Spreading Factors (SFs) to trade-off between the resulting bit rate, the transmission range, and the energy required to transmit. According to the standard, the SF can assume integer values between 7 and 12. A lower value of SF allows faster modulation, shorter Time-on-Air (ToA), and lower energy consumption. On the other side, a signal with lower SF is more affected by noise, attenuation, and interference. Therefore, higher values of SF are more suitable for long-range communication. Recently, a new PHY layer, Long Range Frequency Hopping Spread Spectrum (LR-FHSS) [9] was proposed to address extremely long-range and large-scale communication scenarios such as direct IoT satellite. At the core of LR-FHSS, there is a fast frequency hopping technique that allows higher network capacity while offering the same radio link budget as LoRa. The LR-FHSS mechanism is one of the few concrete attempts to optimize the LoRa PHY protocol for direct communication with LEO satellites. Currently, the LR-FHSS is still under standardization process

by Semtech and not available in Commercial off-the-shelf (COTS) devices. Therefore, in our work, aiming to provide a concrete and feasible solution, we assumed EDs communicate over LoRa with $SF = 12$. Network scalability of LoRa to LEO satellites could be further improved in future work, combining SALSA with LR-FHSS. In fact, by leveraging on random frequency hopping and dividing the channel into several sub-channels, LR-FHSS allows using several data rates at the same time (two in the USA FCC and four in the EU ETSI) [9]. Many more transmissions could be scheduled in parallel during the limited satellite visibility, on orthogonal channels, without having collisions.

B. MAC LAYER

At the Medium Access Control (MAC) layer, the network adopts the LoRaWAN protocol, based on pure ALOHA random access [26]. The standard [2] defines three main channel access strategies. Class A (Aloha) includes the basic set of features needed by any ED to join a LoRaWAN network. Class A EDs use ALOHA protocol for an uplink packet transmission, followed by two short downlink receive windows, RX_1 and RX_2 . Thus, the downlink communication is triggered by the ED. Each downlink frame needs to wait for the uplink communication. The LoRaWAN specification provides two enhancements to Class A: Class B (Beacon) and Class C (Continuous) to increase the downlink possibilities. These downlink protocol enhancements are paid with additional energy consumption. Class B EDs open additional receive windows at scheduled times. The GW transmits beacons to keep the synchronization with EDs and opens new downlink slots. Thus, the server can directly deliver frames to correspondent EDs without waiting for them to start an uplink communication. Class C EDs keep their radio modules always on in reception mode, and thus they are available all the time for listening to downlink traffic unless they are transmitting. In our work, we assumed EDs operate in Class A and generate uplink traffic, being this the operative mode supported by all the COTS devices. In the future, we aim to design scheduling algorithms for bidirectional communication, and thus, we will also consider Class B and Class C devices, better supporting downlink traffic.

III. SALSA SCHEDULING ALGORITHM

In the present work, we propose *SALSA*, a *Scheduling Algorithm for LoRa to LEO Satellites*. In this Section, we describe the algorithm in all its details, from the system assumptions, to the different access policies.

A. SYSTEM ASSUMPTIONS

In the design of the algorithm, we assume that the LNS is aware of the location of the EDs and the availability time of the gateway (i.e., the visibility period of the satellite). The EDs may have been pre-registered in the network, using Over The Air (OTA) registration while the gateway was still on the ground. All the EDs operate in Class A. Moreover, we assume that the LNS sends in unicast (during the RX windows) the

scheduling table to each ED in the network. To consider the deviation of the current orbit of the satellite from its initial orbit, periodic scheduling updates should be sent directly to each device using unicast messages. Note that the exchange of the configuration parameters and scheduling tables is out of the scope of the present work, the reason why we made such assumptions.

The LNS is in charge of scheduling the uplink transmissions for each ED and assigning priorities. To guarantee long-range transmission from the EDs to the satellite gateway, we assume that all the EDs transmit using $SF = 12$. Using a single value of the SF implies that only a transmission per time can be scheduled, without incurring a collision. According to the standard, a single uplink transmission keeps the channel busy for a Time on Air (ToA) period. We propose to add two Guard Time, T_g , one before and one after the ToA for taking into account possible synchronization issues between the ED and the mobile gateway (see Fig. 3). Thus, whenever the LNS schedules a transmission for an ED, it reserves the channel for that ED for a period of time equal to $T_r = ToA + 2 \times T_g$.

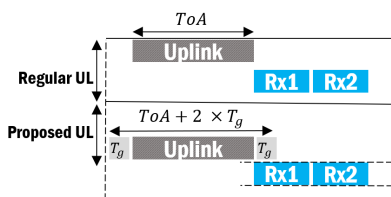


FIGURE 3. Time reserved T_r for each ED for an uplink transmission. $2 \times T_g$ have been introduced to mitigate uncertainties and inaccuracies of the synchronisation between the ED and the satellite gateway. The receive windows RX_1 and RX_2 are opened after the end of the uplink transmission, according to the standard.

B. FCFS POLICY

The LNS builds the scheduling tables for each of the EDs in the network based on their *discovering time* of the satellite. The latter depends on the satellite footprint, the satellite movement along its orbit, and the value of the elevation angle. As shown in Fig.4, while the satellite moves along its orbit, the satellite footprint also moves in time: therefore, different EDs will be progressively in the satellite visibility, one after the other, in different periods. To efficiently use the network resources and maximize the number of EDs that can communicate with the mobile gateway, the LNS gives priority to the EDs that are visited first by the satellite footprint. In the example in Fig.4, the ED_i will transmit before the ED_j being $r_i < r_j$, with r_i relative distance of ED_i from the satellite footprint.

Many nodes may be concentrated in the same area (i.e., dense area), one very close to another, and thus, while an ED is still transmitting, another one could enter the satellite footprint and attempt a transmission as well. To avoid such a scenario generating collisions, we introduce the concept of *slide*. We divide the satellite footprint into several slides. They

can have variable time duration: in the simpler case, a slide is equal to the time needed for completing a single uplink transmission T_r . It is the case when there is no overlap between concurrent transmissions from several EDs because the EDs are one far away from another. On the contrary, when there is an overlap (i.e., a dense area with many EDs concentrated together), and an ED cannot finish a transmission before another ED enters the satellite coverage, the LNS allocates a larger slide. The slide’s duration is set to $N_s \times T_r$, where N_s is the number of neighbor nodes within the identified dense area. The LNS defines the starting time T_{S_i} of the ED_i transmission. If the Start of the Satellite Visibility for that ED_i , T_{SSV_i} overlaps with an ongoing transmission from another ED_j , the ED_i has to wait till T_{E_j} (ending time of ED_j transmission). Either wise, it can start transmitting as soon as it sees the satellite (see Algorithm 1). In the example in Figure 4, the ED_4 will not transmit as soon as it enters the satellite footprint, but after the ED_3 finishes its transmission. By doing so, SALSA with its TDMA approach and First Come First Served (FCFS) policy results to be a collision-free scheduling algorithm. It intrinsically avoids collision, not allowing conflicting parallel overlapping transmissions.

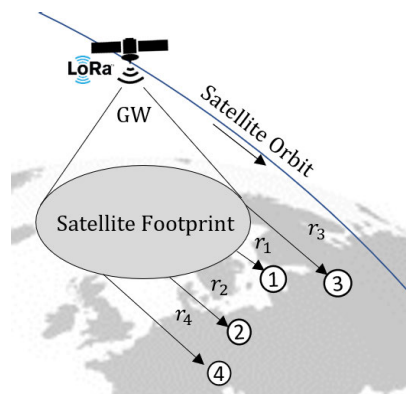


FIGURE 4. Scheduling of EDs transmissions according to the satellite movement along its orbit, and the visit time of the EDs. EDs that are visited first have first chance to transmit.

Algorithm 1 SALSA With FCFS Policy

```

for  $slide_k$  do
  for  $ED_i$  do
    if  $T_{SSV_i} < T_{E_j}$  then
       $T_{S_i} \leftarrow T_{E_j}$  ▷ TX after last ED
    else
       $T_{S_i} \leftarrow T_{SSV_i}$  ▷ TX when satellite is available
    end if
  end for
end for
    
```

C. FAIR POLICY

The LEO satellite is available for a limited time, equal on average to 100s. The visibility time could be shorter than the

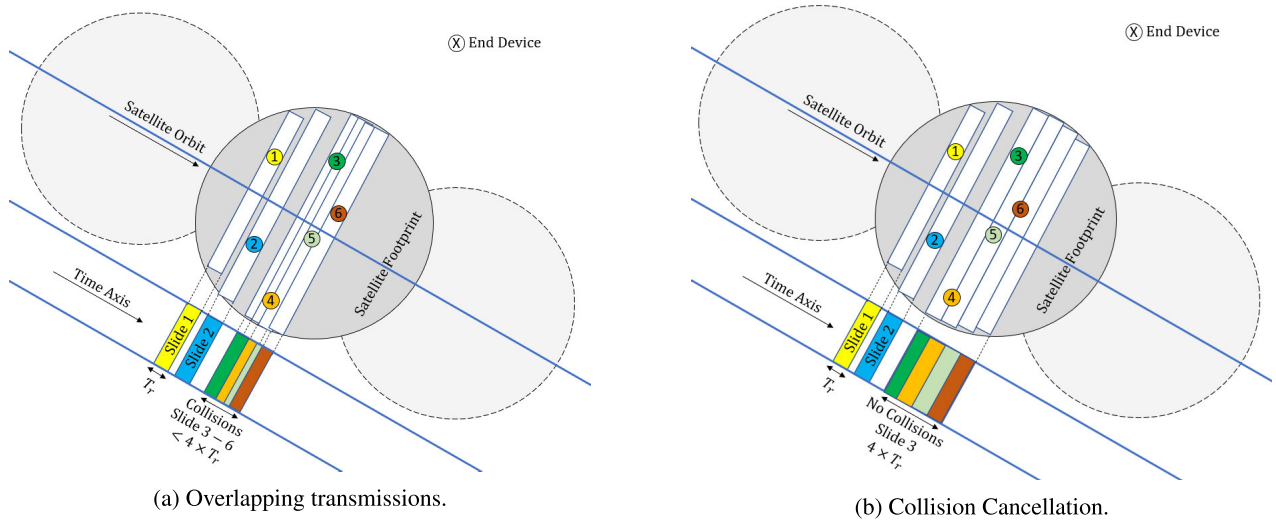


FIGURE 5. Satellite footprint and slides. Cancellation of collision with TDMA approach and FCFS policy.

total time needed for scheduling at least one uplink transmission from all the EDs within every single slide. In a dense area with hundred of EDs, several EDs that are visited last by the satellite may never get a chance to transmit. They will miss the satellite each time it is their turn to transmit, according to the FCFS policy. We propose the FAIR policy to overcome such shortcomings and ensure that all the EDs have an equal chance to transmit. The logic behind it is described by Algorithm 2.

Algorithm 2 SALSA With Fair Policy

```

for slidek do
  for EDi do
    if ntxi ≤ ntxj then           ▷ Fairness check
      if TSSVi < TEj then
        TSi ← TEj                 ▷ TX after last ED
      else
        TSi ← TSSVi           ▷ TX when satellite is available
      end if
    else
      Do not TX                       ▷ Give the chance to the next ED
    end if
  end for
end for
    
```

Let consider N EDs within a given slide of the satellite footprint. The LNS first checks if the ED _{i} had already several transmission opportunities, n_{tx_i} , higher than the other end devices ED _{j} , within the same slide. In this case, with $n_{tx_i} > n_{tx_j}$, and $j \neq i$, the ED _{i} will not be granted an uplink slot. On the contrary, if it can transmit because it transmitted fewer times than its neighbours, the LNS has to define the starting time T_{S_i} of its transmission, according to the FCFS policy (see Algorithm 1).

D. OPTIMAL SCHEDULE

After scheduling the EDs’ transmissions according to the FAIR policy, there might be still several slots available, depending on the satellite visibility and the density of the EDs in the coverage area. Due to (i) the short availability time of the satellite and (ii) the LoRa duty cycle limitations imposed by the LoRaWAN regional parameters, it is very unlucky that a given ED have a second chance of transmitting within a single satellite visit. For instance, the maximum ToA for {51 Bytes, EU868, SF12, 125KHz} [27] is equal to 2793.5ms while the duty cycle is 1% in the EU region. So the ED must wait for 279.35s before having a new uplink opportunity, while the availability time for a LEO satellite is shorter than 120s at 500Km height above 30° elevation.

While running the SALSA algorithm the LNS also implements a check on the duty cycle limitations and further optimizes the use of the limited network resources. And whenever it identifies a gap (empty slot) in the schedule, it allocates it to another ED _{i} , at the conditions that: (1) The new expected reserved time, T_{r_i} , does not create any conflict in the scheduling table with the already reserved slots for other EDs. (2) The ED _{i} has visibility with the satellite in that expected time.

IV. PERFORMANCE EVALUATION

In this section we first describe the simulated environment used for reproducing the system behaviour, and for implementing the SALSA algorithm. Then, we evaluate the performance achievable with SALSA, and compare them with the benchmark solution (ALOHA).

A. SIMULATED ENVIRONMENT

We have implemented the algorithm in a Matlab environment to evaluate the performance achievable with SALSA, FCFS, and FAIR policies. We consider network size variable

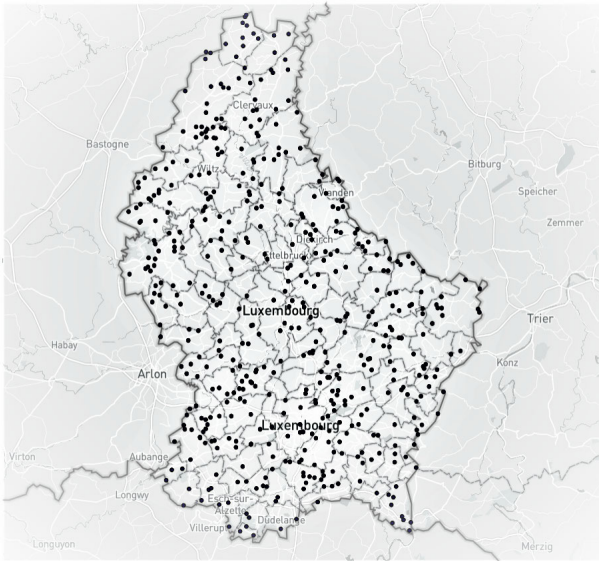


FIGURE 6. Location of EDs over the country of Luxembourg, following a random uniform distribution. Visualization done with [28].

in the range [50, 500]. The EDs are located according to a random and uniform distribution in a target area. In this work, we assume the target area coincides with the Luxembourg country (Fig.6). We derived the exact location of the EDs using random uniform distribution in the GEOPY [29] library of Python programming language. Besides the considered scenario, the algorithm is not limited to this selected geographical area and applies to any other one.

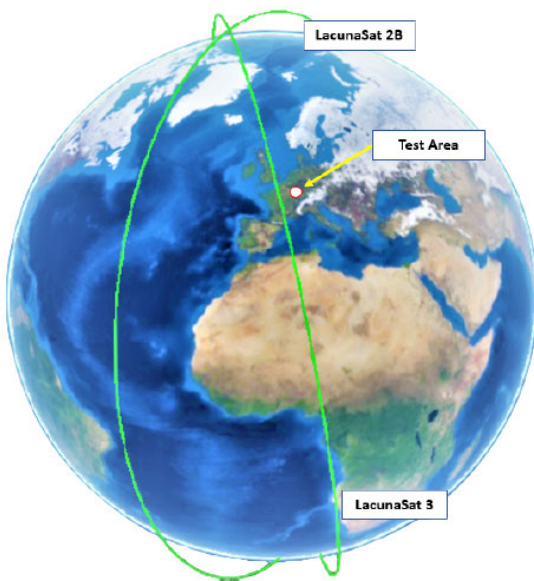


FIGURE 7. Orbits of LacunaSat 3 and LacunaSat 2B satellites compared to the target area on the Earth, Luxembourg.

We considered the Lacuna satellites, in particular *Lacuna-Sat 3* and *Lacuna-Sat 2B* which have the approximate height of 500Km to 600Km from the earth. In the performance

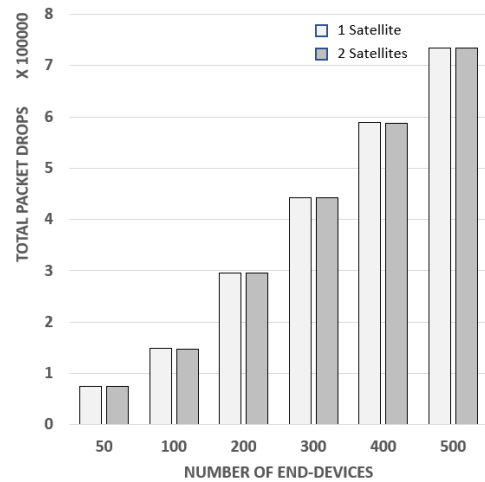


FIGURE 8. Packet drops during one month when the EDs transmit with ALOHA-based LoRa, generating periodic traffic, every 30 minutes.

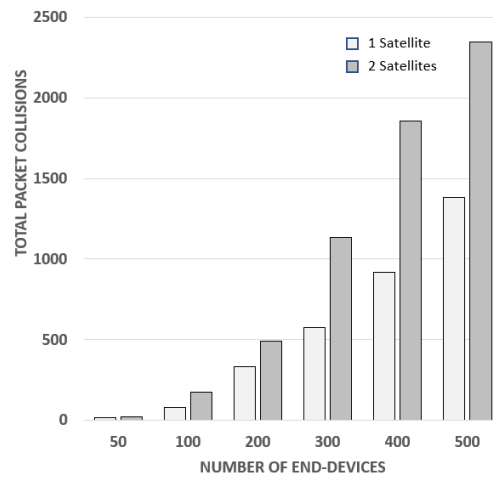
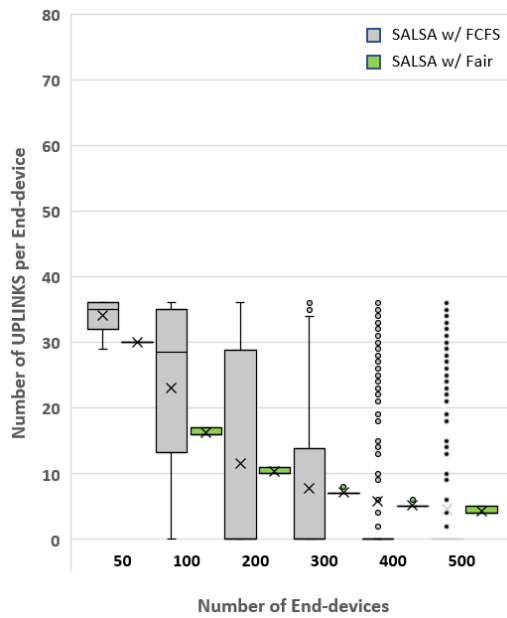


FIGURE 9. Packet collisions during the satellite visibility in one month with 1 and 2 satellites.

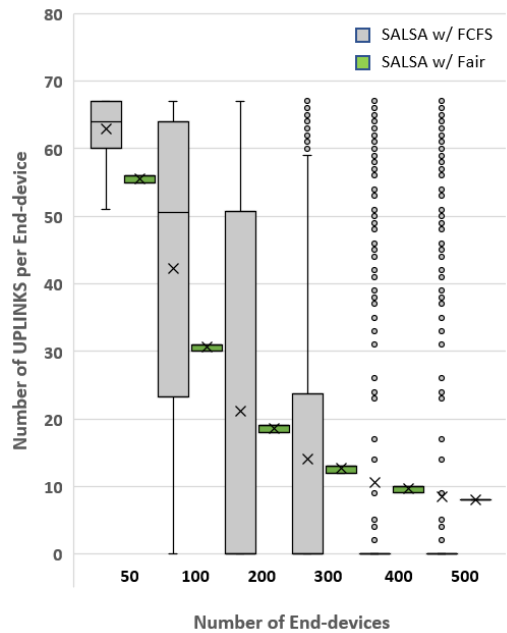
analysis, we adopted their real visibility over Luxembourg during October 2021. Moreover, we assumed an elevation angle $\Theta_e = 30^\circ$ for both satellites.

We used SKYFIELD library [30] with the CelesTrak [31] online Two Line Element (TLE) sources for generating the real visibility time tables of the satellites per each ED. A TLE provides the satellite location along its orbit for a given pointing time (the time epoch), by encoding the orbital elements. The specific location of the EDs (uniformly distributed over the Country of Luxembourg), the satellites TLE, and the fixed elevation angle ($\Theta_e = 30^\circ$) were given in input to the SKYFIELD. Finally, the obtained human-readable time outputs were translated into epoch timestamps and used in the Matlab algorithm, implementing SALSA. The code is public available at [32].

We assume the Guard Time T_g is equal to 10ms; and the ToA is the maximum allowed ToA in the EU868 region, with $SF = 12$ and the maximum payload of 51 bytes. It follows,



(a) 1 Satellite



(b) 2 Satellites

FIGURE 10. Average Number of Uplink Transmission for each ED, with FCFS and FAIR policy (a) with one satellite, and (b) with two satellites.

the time reserved for each ED, for a single transmission is $T_r = 2 \times T_g + ToA = 2813.5 \text{ ms}$.

B. ACHIEVABLE PERFORMANCE

We have compared the behaviour of the satellite LoRaWAN network in three different scenarios:

- **ALOHA:** the EDs transmit following the LoRa standard, with random ALOHA access. They transmit periodically, every 30 minutes, without having any

knowledge about the satellite visibility. Note that we introduced a random uniformly distributed offset in the start time of the first uplink per each ED.

- **SALSA w/ FCFS:** the EDs that are visited first by the satellites transmit first, but only when the channel is not busy (no collision due to overlapping transmissions).
- **SALSA w/ FAIR:** all the EDs get an equal chance to transmit thanks to the FAIR policy, in addition to FCFS.

When (i) the EDs are not aware of the visibility time of the satellite(s), and (ii) no scheduling algorithm is implemented, the network faces a high number of *packet drops* and *packet collisions*. Packet drops are due to the lack of availability of the satellite gateway, while packet collisions are due to concurrent random transmissions. As shown in Fig. 8, the adoption of a constellation of two satellites does not improve the network performance in the absence of a scheduling technique. As depicted in Fig. 9, the performance in term of the number of collisions get even worst with two satellites: it is almost double with a large network size.

Contrary to random ALOHA-like access, when adopting SALSA the LNS schedules the transmissions while aware of the location of the EDs and the satellite availability time for each of them.

We compared the performance achievable with FCFS and FAIR policy in the worst-case scenario, with a single LEO satellite, and with a constellation of two satellites (Fig. 10). FCFS policy schedules the transmissions according only to location-based priority. It follows that some devices (those visited last by the satellite) never get a chance of an uplink opportunity. Moreover, each device may get a different number of reserved slots (as confirmed by the large whiskers and the many outliers). This behaviour is more pronounced with two satellites. On the contrary, the FAIR policy guarantees all devices get the same chance, even in large networks, besides the size of the constellation. With two satellites, the network performance improves, and the average number of uplink transmissions per ED is almost duplicated compared to the scenario with a single LEO satellite. The obtained results confirm the reliability and scalability of the SALSA algorithm with the FAIR policy.

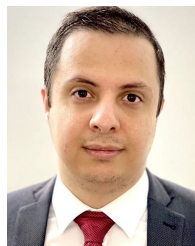
V. CONCLUSION

In the present work, we tackle the issue of a LoRaWAN mobile gateway installed on the LEO satellite. For the first time, we propose a TDMA scheduling technique, SALSA. It optimises the usage of the satellite bandwidth during limited satellite visibility. The proposed SALSA algorithm first avoids by design any collision, thanks to the TDMA approach; second, it avoids packet drop by considering the satellite availability; third, it ensures fair access for all the end devices in the network. With simulation results, we demonstrate in a realistic scenario that the SALSA algorithm can improve the network performance. The opportunities of uplink transmission per each ED decreases with increasing network size (more EDs are contending for the same channel). LEO constellations help overcome this issue.

The transmission opportunities are almost duplicated when considering two satellites, and they could be further improved with larger constellations. In our future work, we will extend the SALSA algorithm to support both uplink and downlink traffic, with Class A and B devices.

REFERENCES

- [1] I. F. Akyildiz and A. Kak, "The internet of space things/cubesats," *IEEE Netw.*, vol. 33, no. 5, pp. 212–218, Sep. 2019.
- [2] LoRa Alliance. *LoRaWAN Specifications*. Accessed: Sep. 1, 2021. [Online]. Available: <https://lora-alliance.org/>
- [3] *Lacuna Space Website*. Accessed: Mar. 22, 2021. [Online]. Available: <https://lacuna.space/>
- [4] Wyld Networks. *Creating the Networks That Matter*. Accessed: Mar. 16, 2021. [Online]. Available: <https://wyldnetworks.com/wyld-announces-global-lora-satellite-deal/>
- [5] M. Centenaro, C. E. Costa, F. Granelli, C. Sacchi, and L. Vangelista, "A survey on technologies, standards and open challenges in satellite IoT," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1693–1720, 3rd Quart., 2021.
- [6] M. R. Palattella and N. Accettura, "Enabling internet of everything everywhere: LPWAN with satellite backhaul," in *Proc. Global Inf. Infrastruct. Netw. Symp.*, Oct. 2018, pp. 1–5.
- [7] L. Chaari, M. Fourati, and J. Rezgu, "Heterogeneous LoRaWAN LEO satellites networks concepts, architectures and future directions," in *Proc. Global Inf. Infr. Net. Symp. (GIIS)*, 2019, pp. 1–6.
- [8] Z. Qu, G. Zhang, H. Cao, and J. Xie, "LEO satellite constellation for Internet of Things," *IEEE Access*, vol. 5, pp. 18391–18401, 2017.
- [9] G. Boquet, P. Tuset-Peiro, F. Adelantado, D. Watteyne, and X. Vilajosana, "LR-FHSS: Overview and performance analysis," *IEEE Commun. Mag.*, vol. 59, no. 3, pp. 30–36, Mar. 2021.
- [10] A. Doroshkin, A. Zadorozhny, O. Kus, V. Prokopyev, and Y. Prokopyev, "Experimental study of LoRa modulation immunity to Doppler effect in CubeSat radio communications," *IEEE Access*, vol. 7, pp. 75721–75731, 2019.
- [11] G. Ferre, "Collision and packet loss analysis in a LoRaWAN network," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2017, pp. 2589–2590.
- [12] P. Gkotsiopoulos, D. Zorbas, and C. Douligeris, "Performance determinants in LoRa networks: A literature review," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1721–1758, 3rd Quart., 2021.
- [13] N. Chinchilla-Romero, J. Navarro-Ortiz, P. Mu noz, and P. Ameigeiras, "Collision avoidance resource allocation for LoRaWAN," *Sensors*, vol. 21, no. 4, p. 1218, Feb. 2021.
- [14] M. Alenezi, K. Chai, A. Alam, Y. Chen, and S. Jimaa, "Unsupervised learning clustering and dynamic transmission scheduling for efficient dense LoRaWAN networks," *IEEE Access*, vol. 8, pp. 191495–191509, 2020.
- [15] G. Yapar, T. Tugcu, and O. Ermis, "Time-slotted ALOHA-based LoRaWAN scheduling with aggregated acknowledgement approach," in *Proc. 25th Conf. Open Innov. Assoc.*, Nov. 2019, pp. 383–390.
- [16] G. Lee and J. Youn, "Group-based transmission scheduling scheme for building LoRa-based massive IoT," in *Proc. Int. Conf. Artif. Intell. Inf. Commun.*, Feb. 2020, pp. 583–586.
- [17] J. Pullmann and D. Macko, "A new planning-based collision-prevention mechanism in long-range IoT networks," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9439–9446, Dec. 2019.
- [18] C. E. Fehri, M. Kassab, N. Baccour, S. Abdellatif, P. Berthou, and I. Kammoun, "An uplink synchronization scheme for LoRaWAN class B," in *Proc. Int. Conf. Wireless Mobile Comput., Netw. Commun.*, Oct. 2019, pp. 47–52.
- [19] L. Chasserat, N. Accettura, and P. Berthou, "Short: Achieving energy efficiency in dense LoRaWANs through TDMA," in *Proc. 21st Int. Symp. World Wireless, Mobile Multimedia Netw.*, Aug. 2020, pp. 26–29.
- [20] 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 Things J.*, vol. 7, no. 1, pp. 669–683, Jan. 2020.
- [21] D. Zorbas, "A testbed for time-slotted LoRa communications," in *Proc. 21st Int. Symp. World Wireless, Mobile Multimedia Netw.*, Aug. 2020, pp. 182–184.
- [22] D. Zorbas, C. Caillouet, K. A. Hassan, and D. Pesch, "Optimal data collection time in LoRa networks—A time-slotted approach," *Sensors*, vol. 21, no. 4, p. 1193, Feb. 2021.
- [23] M. Asad Ullah, K. Mikhaylov, and H. Alves, "Enabling mMTC in remote areas: LoRaWAN and LEO satellite integration for offshore wind farms monitoring," *IEEE Trans. Ind. Informat.*, early access, Sep. 14, 2021, doi: 10.1109/TII.2021.3112386.
- [24] F. Tondo, S. Montejó-Sánchez, M. Pellenz, S. Céspedes, and R. Souza, "Direct-to-satellite IoT slotted aloha systems with multiple satellites and unequal erasure probabilities," *Sensors*, vol. 21, no. 21, p. 7099, 2021.
- [25] SEMTECH. (Feb. 2020). *LoRa and LoRaWAN: A Technical Overview*. [Online]. Available: <https://www.semtech.com/lora>
- [26] T. Polonelli, D. Brunelli, A. Marzocchi, and L. Benini, "Slotted ALOHA on LoRaWAN—design, analysis, and deployment," *Sensors*, vol. 19, no. 4, p. 838, Feb. 2019.
- [27] The Things Network. *ToA Calculator*. [Online]. Available: <https://www.thethingsnetwork.org/airtime-calculator>
- [28] *MapBox Studio*. Accessed: Nov. 31, 2021. [Online]. Available: <https://www.mapbox.com/>
- [29] *GEOPY, a Python Client for Several Popular Geocoding Web Services*. Accessed: Nov. 31, 2021. [Online]. Available: <https://github.com/geopy/geopy>
- [30] *Skyfield Repository*. Accessed: Nov. 1, 2021. [Online]. Available: <https://github.com/skyfielders/python-skyfield>
- [31] *CelesTrak*. Accessed: Nov. 1, 2021. [Online]. Available: <https://celestrak.com/>
- [32] *SALSA Algorithm*. Accessed: Jan. 22, 2022. [Online]. Available: <https://github.com/list-luxembourg/salsa>



MOHAMMAD AFHAMISIS (Member, IEEE) received the bachelor's degree in telecommunication engineering from Semnan University, Iran, and the Master of Science degree in satellite technology engineering from the Iran University of Science and Technology (IUST). He is currently pursuing the Ph.D. degree with the ERIN Department, Luxembourg Institute of Science and Technology (LIST), working on optimizing LoRaWAN protocol for integration with satellite for smart agriculture applications. His thesis was on optimizing the control of GEO satellites with large antennas and deployable solar cells. Also, he has six years of experience as a Senior Radio Communications Specialist and managing different projects on radio communications and the IoT monitoring topics in the Iranian Railways Headquarter Office.



MARIA RITA PALATTELLA (Member, IEEE) received the Ph.D. degree, in 2011. She is currently a Senior Research and Technology Associate at the LIST. She is leading the work on designing innovative network architectures and cross-layer protocols optimization for the Internet of Things applications for environmental monitoring, precision agriculture, and disaster management. She is investigating means of integrating satellite and terrestrial networks for future high-reliable end-to-end communication systems. She is a member of IEEE ComSoc WISE, and IETF 6TiSCH WG, and the Secretary of the IEEE 5G Mobile Wireless Internet Technical Subcommittee. She sits on the Editorial Board of the *Transactions on Emerging Telecommunications Technologies* and the *EAI Transactions on IoT*.