



**HAL**  
open science

# Simulating LoRa-Based Direct-to-Satellite IoT Networks with FLoRaSat

Juan A. Fraire, Pablo Madoery, Ait Mehdi, Oana Iova, Fabrice Valois

► **To cite this version:**

Juan A. Fraire, Pablo Madoery, Ait Mehdi, Oana Iova, Fabrice Valois. Simulating LoRa-Based Direct-to-Satellite IoT Networks with FLoRaSat. WoWMoM 2022 - IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks, Jun 2022, Belfast, United Kingdom. hal-03698223v1

**HAL Id: hal-03698223**

**<https://hal.science/hal-03698223v1>**

Submitted on 17 Jun 2022 (v1), last revised 22 Jul 2022 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Simulating LoRa-Based Direct-to-Satellite IoT Networks with FLORASAT

Juan A. Fraire<sup>\*†</sup>, Pablo Madoery<sup>†</sup>, Mehdi Ait Mesbah<sup>\*</sup>, Oana Iova<sup>\*</sup>, Fabrice Valois<sup>\*</sup>

<sup>\*</sup>Univ Lyon, INSA Lyon, Inria, CITI, F-69621 Villeurbanne, France

<sup>†</sup>CONICET - Universidad Nacional de Córdoba, Argentina

**Abstract**—Direct-to-Satellite-IoT (DTS-IoT) is a promising approach for data transfer to/from IoT devices in remote areas where deploying terrestrial infrastructure is not appealing or feasible. In this context, Low-Earth Orbit (LEO) satellites can serve as passing-by IoT gateways to which devices can offload buffered data to. However, transmission distance and orbital dynamics, combined with highly constrained devices on the ground makes DTS-IoT a very challenging problem. In fact, existing IoT medium access control protocols, negotiations schemes, etc. need to be revised and/or extended to scale up to these challenging conditions. The intricate time-dynamic aspects of DTS-IoT networks require of adequate simulation environments to assess the expected performance of enabling technologies. To make up for the lack of such tools, we present a novel event-driven open-source end-to-end simulation tool coined FLORASAT. The simulator leverages Omnet++ and includes a benchmarking DTS-IoT scenario comprising 16 cross-linked LEO satellites and 1500 IoT nodes on the surface. Satellites and devices are connected using the standard LoRaWAN Low-Power Wide Area (LPWAN) protocol (Class A and B). FLORASAT allows the easy implementation and study of DTS-IoT radio access and core network protocols, and we take advantage of this flexibility to investigate expected network metrics and non-intuitive phenomena emerging from the resulting multi-gateway setup.

**Index Terms**—Satellite Communication, LEO satellite, LoRaWAN, LoRa, IoT, simulation, OMNeT++

## I. INTRODUCTION

The recent spur in space projects has revamped the interest in satellite communication. This is especially observed in the Internet of Things (IoT) community that constantly seeks to diversify the application scenarios [1], but also, from the telecommunication operators that want to expand the IoT network coverage to offer a promising omnipresent world-wide service [2], for which orbiting assets can be a game changer.

Low-Earth Orbit (LEO) satellites such as CubeSats are the embodiment of the new space movement (cheap deployment, quick procurement) and their inherent characteristics (orbital periods of 90 minutes at  $\sim 500$ km altitude) can play a prominent role in the new IoT era. Indeed, these cost-effective satellites can serve as passing-by gateways that would provide 6-12 min connectivity over a given geographical area, 2-4 times a day. IoT devices using low-power, long-range technologies such as NB-IoT and LORA, and with a clear line-of-sight to the sky can directly communicate with the gateway situated on-board of satellites. Due to the low data rate, link budgets can be closed even with low-cost antennas, and just firmware modifications to the device [3], [4]. This can be extremely useful to applications in less accessible regions

(*i.e.*, oceans, mountains, poles) that might not justify or even hinder the deployment of IoT gateways [5].

However, transmission distances, channel dynamics, and thousands of highly constrained devices on the ground under the coverage of multiple orbiting gateways make Direct-to-Satellite IoT (DTS-IoT) a very challenging problem [6]. Besides the challenge in the channel access negotiation domain, the core network of the DTS-IoT system extends into the space domain, and needs to robustly operate over low-bandwidth inter-satellite links (ISL), which also change in time, due to the orbital dynamics.

One of the main challenges with DTS-IoT is their evaluation, as real in-orbit deployments are extremely costly and rigid. Object of study includes existing terrestrial IoT protocols when applied to the space context unchanged, and specific adaptations/configurations of such technologies to efficiently inter-operate in space-terrestrial deployments. As a result, we need realistic simulators implementing both IoT and satellite dynamics as detailed as possible, which currently do not exist. Space and IoT software tools, in general, remain separated, and the few papers that looked into DTS-IoT either lack of realistic orbital mechanics, detailed link layer protocol modeling, or end-to-end networking analysis capabilities.

To this end, we propose FLORASAT, a novel event-driven open-source end-to-end simulation tool that leverages OMNeT++ and includes a benchmark DTS-IoT scenario comprising 16 cross-linked LEO satellites and 1500 IoT end-devices. We present the modular architecture that allows FLORASAT to be extended with new protocols/methods, and to become the de facto DTS-IoT simulation evaluation framework.

The remainder of this paper is structured as follows. Section II presents the state of the art of IoT, satellite technologies, and simulation tools. FLORASAT use cases, features, architecture, and metrics are presented in Section III. A realistic DTS-IoT scenario and new evaluation metrics are introduced in Section IV, and the results are analyzed in Section V. The final conclusions and future perspectives are summarized in Section VI.

## II. BACKGROUND

This section presents the main characteristics of satellite and IoT technologies on which we base our contribution, and details the existing simulation frameworks.

### A. LoRa and LoRaWAN

LoRa [7] is one of the most representative IoT technologies for exchanging low-volume data over long distances (up to 10-15 kilometers), while keeping very low energy consumption for the end devices. This is achieved by using a proprietary chirp spread spectrum modulation that offers a trade-off between sensitivity and data rate.

LoRaWAN [8] is the standard protocol that defines the network architecture and the communication protocols to be used on top of the LoRa modulation. End devices send data to all the gateways that are in their communication range, which is afterwards forwarded to a central entity called the network server, before being dispatched to a specific application server. The network server is a key component of the LoRaWAN architecture as it is responsible for managing the gateways and end-devices, establishing end-to-end security, deduplicating uplink data, and scheduling downlink traffic. For the medium access, LoRaWAN uses an Aloha-based protocol, for which it defines three modes of operation: (i) Class A, where end-devices spend their time in an idle state, waking up only when they have new data to transmit and consecutively wait for a possible downlink communication, (ii) Class B that offers regularly-scheduled, fixed-time opportunities for downlink communication in so-called ping slots, and (iii) Class C, where end-devices are always listening for downlink messages, unless they are transmitting an uplink packet.

There are several custom-build simulators for LoRaWAN networks [9]. One of the first and most used LoRaWAN simulators that implemented an accurate physical layer was LoRaSim<sup>1</sup>, a discrete-event simulator based on SimPy [10]. With LoRaSim, one can simulate collisions in LoRa networks with multiple gateways, however several key elements are missing, such as the downlink communication and the implementation of a network server. The main computer network simulation environments NS-3 and OMNeT++ were recently enriched with modules to support LoRaWAN networks. The NS-3 module<sup>2</sup> developed by To *et al.* is a simplistic model of a LoRaWAN network that supports only one gateway [11], limiting its use. On the other hand, the OMNeT++ addition, FLoRa (Framework for LoRa)<sup>3</sup>, allows the creation of LoRaWAN networks with modules for LoRa nodes (Class A only), gateways and network server [12]. The main advantage of this simulator is its accurate model of the LoRa physical layer and its support of full end-to-end communications.

### B. Satellites in the IoT context

Satellites in Low Earth Orbit (LEO) are deployed at low altitudes (160 – 1,000 km) and have orbital periods in the order of 90 minutes. This implies that the duration of a typical satellite pass over a given geographical area on Earth will be in the order of 10 to 3 minutes (for a perfectly zenithal pass and a pass over the horizon, respectively). During this period, the

channel conditions will vary drastically from more than 2,000 km to the actual satellite altitude (at the zenith position as seen from the device perspective). The coverage region of a single LEO satellite will thus move at the same speed over the surface (approx. 7 km/s), implying that the set of served devices will rapidly change in time. In order to ensure continuous services, LEO satellites have to be deployed in constellations (as one satellite hides in the horizon, another is rising to serve a given device on the surface). In these cases, Inter-satellite links (ISL) allow LEO fleets to coordinate and relay user data with the ground station connected to the core network.

Satellite orbits are *propagated* by models to determine the position of the spacecraft at any time, based on a given initial condition (*i.e.*, orbital parameters). Analytical propagators leverage easy-to-compute mathematical expressions based on simple assumptions (*i.e.*, two-body problem with perfectly spherical Earth), while numerical propagators deliver more accurate trajectory estimations considering other factors (*e.g.*, non-spherical Earth, gravitational force from the Moon and the Sun, atmospheric drag, and solar pressure, among others). Multiple orbital analysis tools include orbit propagators. Systems Tool Kit (STK) [13] is the most popular commercial software, and includes several extension modules provided by AGI<sup>4</sup>. The General Mission Analysis Tool (GMAT), developed in C++ by NASA, is widely adopted by the open-source community [14]. Orekit is a free Java-based alternative [15].

The integration of propagators with channel and link communication models is achieved in plenty of tools, but only a few works looked into the specifics of satellite IoT. Afhamisis *et al.* [16] use custom-made simulators based on Matlab that lack accurate modeling of the network and medium access layers. The satellite IoT simulator from Kodheli *et al.* [17] is also limited to radio link aspect evaluations. An end-to-end NS-3 satellite IoT simulator for NB-IoT is presented in [18], but the tool lacks realistic orbital trajectories. Another work from Jit *et al.* [19] presents a simulator based on orbital data exported from STK, but only considers protocol-agnostic traffic demands aspects.

To wrap-up, simulators for IoT – and more specifically, LoRa/LoRaWAN – are widely available, but separated from the extensive set of free and commercial tools that properly account for the satellite dynamics. Satellite simulators account very well for orbital parameters but lack the specifics of IoT networks. There is a clear gap in the satellite IoT domain, particularly in DTS-IoT, regarding end-to-end protocol simulators with realistic orbital mechanics, inter-satellite link modeling, and proper abstractions of radio access and core network aspects of state-of-the-art IoT technologies.

## III. FLORASAT

In order to realistically represent the environment in which future DTS-IoT systems will operate, we need (i) an accelerated simulation approach so that we can quickly simulate

<sup>1</sup> Available at <https://www.lancaster.ac.uk/scc/sites/lora/lorasim.html>.

<sup>2</sup> Available at <https://github.com/drakkar-lig/lora-ns3-module>

<sup>3</sup> Available at <https://flora.aalto.fi/>.

<sup>4</sup> Home page: <https://www.agi.com/products/stk>

LEO satellite orbits spanning several days/weeks, (ii) a realistic implementation of the physical and medium access layers, accompanied by a detailed modeling of end-to-end network layer, and (iii) an environment that allows running multiple seeds and parametric studies. To this end, we propose FLORASAT (*Framework for LoRa-based Satellite IoT*)<sup>5</sup>, an open-source, event-driven simulator that mimics satellite and LORA/LORAWAN-based IoT systems, and their interactions.

FLORASAT leverages (i) the well-known OMNET++ framework<sup>6</sup> with its INET extension for the Internet stack, and (ii) the open-source LORA-based simulator FLORA with its basic implementation of the LORAWAN [12] protocol. We adapted, extended, and ported the physical layer (LORA) and the link layer (LORAWAN) from FLORA to the DTS-IoT configuration, to which we added a set of new groundbreaking features related to the satellite IoT context. We also implemented a benchmark scenario in which end devices communicate directly with LORAWAN gateways situated on LEO satellites that are organized in a constellation. Gateways send beacons at regular time intervals to announce their presence using Class B mode of LORAWAN, and end devices buffer data until a satellite is in sight. Satellites also communicate among each other using inter-satellite links. We present next the architecture of FLORASAT and its core features.

### A. Features

a) *Orbital mechanics*: The orbital dynamics elements are in charge of propagating satellites along realistic trajectories. We created an orbital processing pipeline in which the scenarios are designed using Contact Plan Designer [20], an STK plugin that was extended to export the position of each satellite in  $X, Y, Z$  coordinates in the Earth-centered, Earth-fixed frame (ECEF), as well as in Latitude, Longitude and Altitude (LLA) on a second-by-second granularity. Due to the ECEF coordinate reference system, IoT devices are seen as fixed points (satellites are the only element with a time-evolving position in ECEF), so they do not need to be created in STK. During simulation time, these files are consulted by the satellite mobility module, the inter-satellite distance computation module, and by the visual module.

b) *Inter-satellite routing*: Since satellite IoT services need to be deployed in networked constellations to cover large geographical areas, we include an inter-satellite link (ISL) model with a basic LEO-to-LEO routing solution. The scenario described above features a grid-shaped LEO constellation, in which a single satellite implements a contact with a ground station (e.g., satellite [15] is the *sink node* of the fleet, as shown in Figure 1). For this case, we implement a simple *right-up* and *down-left* routing approach for the uplink and downlink traffic flows respectively. More sophisticated and general routing approaches (e.g., Dijkstra or topology-aware Contact-Graph Routing [21]) can be easily added in the future thanks to our modular design.

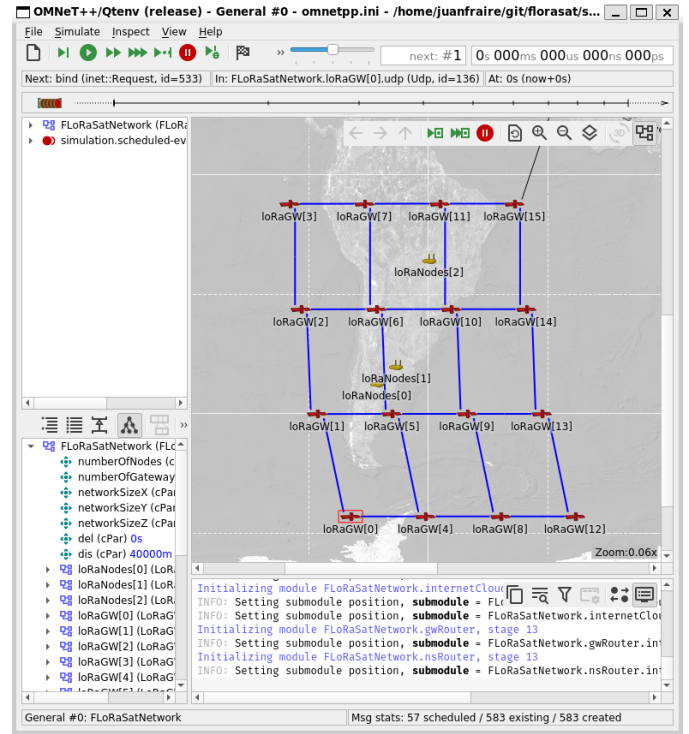


Fig. 1. FLORASAT user interface. Node [15] represents a satellite that links with the ground station.

c) *Beacon-based radio*: As discussed by Vogelgesang *et al.* [22], beacons can play an important role in announcing satellite presence and synchronizing end devices on the ground. In LORAWAN, this can be achieved leveraging the Class B mode of communication where the gateway periodically sends a beacon to synchronize the clock of end-devices and the reception windows for downlink slots. We implemented Class B in FLORASAT, while maintaining the support to the legacy Class A as initially present in FLORA.

d) *Application model*: In the DTS-IoT scenario, end devices cannot always assume an IoT gateway will be present in the sky. Therefore, we extended the application model to also include a delay-tolerant communication, where data is buffered in the end-device until a satellite is present.

e) *Channel model*: Similarly to [23], we employ a simple channel model that accounts for attenuation, interference, and the capture effect at the receiver, considering that those are the aspects that have the most impact on the performance of the uplink and downlink transmissions.

### B. Architecture

We integrated all these new features into the FLORASAT simulator, using a modular architecture as depicted in Figure 3. Each of the illustrated elements corresponds to an OMNET++ compound or `cSimpleModule` with the corresponding C++ class specification. `LoRaNic` and `LoRaGWNic` implement the LORA and LORAWAN layers in the device and in the gateway situated inside the satellite. The `packetForwarder` module takes care of multiplexing and relaying LORAWAN

<sup>5</sup>Available at: <https://gitlab.inria.fr/jfraire/florasat>.

<sup>6</sup>Home page: <https://omnetpp.org/>

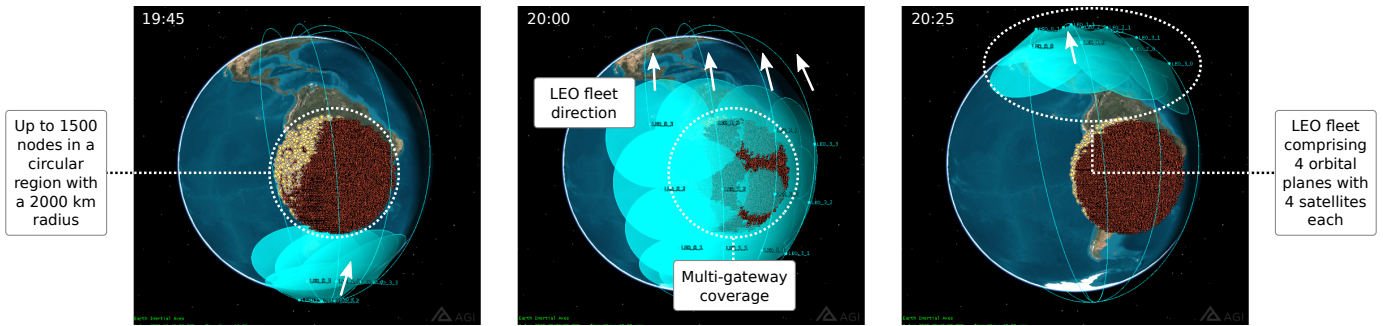


Fig. 2. FLORASAT benchmark scenario.

frames to and from the LORA radio and the ground station link (available in satellite [15]), while the ISLpacketForwarder relay encapsulated frames via ISLs links (up to 4 depending on the satellite position on the grid). When forwarded to the ground station, the LORAWAN frame is encapsulated in an UDP datagram, which is delivered to LORAWAN network server through the Internet. Other modules like mobility, interfaces, and routingTable elements leveraged from INET are illustrated in Figure 3. Finally, a visualizer module takes care of the world-map graphics (see Figure 1) and a LoRaMedium component affects LORA frames with the channel model impairments.

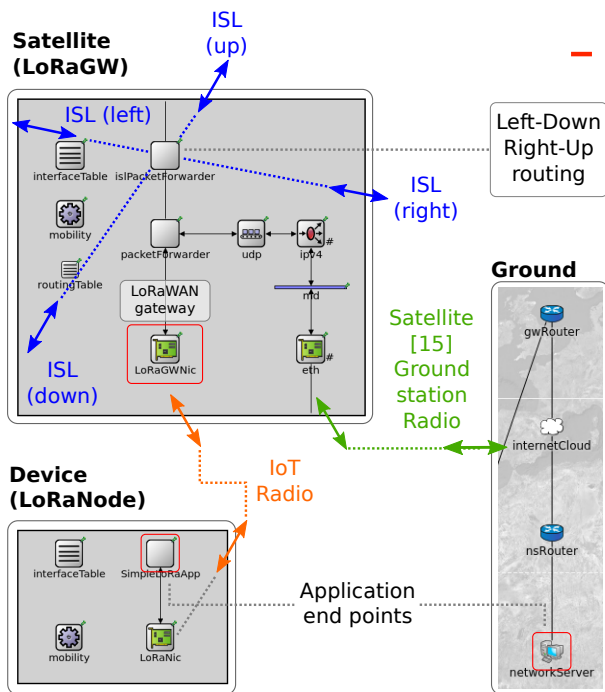


Fig. 3. FLORASAT architecture: terrestrial device, satellite, and ground segments.

#### IV. SCENARIO

In order to validate our simulation framework, we propose a benchmark scenario that includes a constellation of 16 LEO satellites (600 km altitude,  $98^\circ$  inclination, and

ascending nodes in  $310^\circ$ ,  $330^\circ$ ,  $350^\circ$ , and  $370^\circ$  for a total of 4 orbital planes with 4 equally separated satellites each) and 1500 devices distributed uniformly in a circular region with a 2000 km radius (in South America)<sup>7</sup>. The resulting scenario, illustrated in Figure 2, is an appealing benchmark as it includes zenithal and over-the-horizon passes, and multi-gateway coverage during the  $\sim 30$  minutes fly-by of the 16-satellite fleet (half-an-orbit).

#### A. Metrics

FLORASAT disposes of a plethora of evaluation metrics, such as transmission/reception frame count, collision tracking, and end-to-end latency. However, none of these metrics takes into account the challenging conditions in a DTS-IoT environment.

In the specific case of LORAWAN, the DTS-IoT system has to operate on tight latency constraints between the server and multiple gateways. When downlink messages have to be sent to the end device (e.g., ACKs), the network server has to timely schedule the downlink data so that it properly fits in the device's reception windows, as LORAWAN gateways cannot buffer any data. This scheduling has to be very precise, since end devices open only two reception windows precisely one and two seconds after the uplink attempt is concluded, and they last for only one second (note that in Class B, an end device can also receive network-initiated traffic data during a ping slot, which is also timely fixed). The resulting effects of this setup demand the consideration of two specific metrics: *DPAD* and *wasted decoding effort*.

a) *DPAD*: Accomplishing such a precise timing is very challenging, as the network server needs to estimate the latency the downlink packet will suffer during its transmission via the Internet to the ground station, plus the propagation delay to the satellite, plus the multi-hop ISL path ( $n$  times propagation plus processing time), plus the final downlink propagation delay (see Figure 4). This behavior is already embedded in FLORASAT's `networkServer`. However, the traffic load in the satellite fleet is unknown to the server, and congestion scenarios – a likely issue due to the low-data rate ISLs – might provoke a late arrival of the ACKs, and their consequent loss. To keep track of this phenomena, we define

<sup>7</sup>See scenario animation in this video: <https://youtu.be/Xa5EITAMqQ>



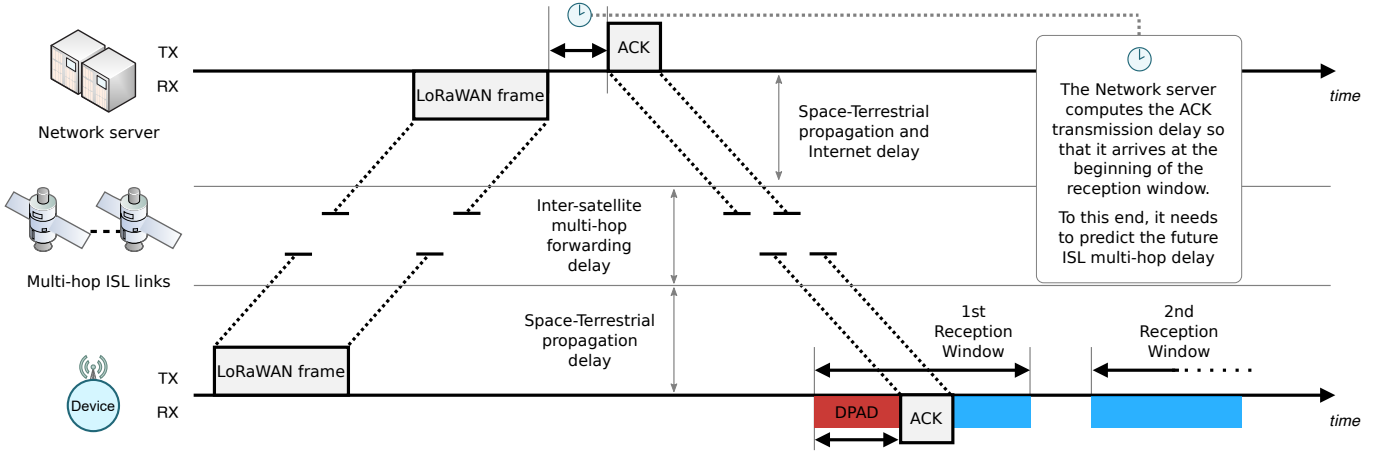


Fig. 4. DPAD metric.

the *Downlink Packet Arrival Delay Since Beginning of the Rx Window* (DPAD) metric, which measures the delay from the start of the reception window to the arrival of the ACK.

b) *Wasted Decoding Effort*: Another effect observed in the multi-gateway DTs-IoT scenario under the most demanding traffic load, is that multiple nodes can have the reception window opened simultaneously. This occurs when simultaneous uplink packets were properly received by two different satellites in distant locations (thus, no collision). In this case, receiving devices might incorrectly hook to the downlink message preamble destined to another node. As a result, a wasted reception effort is observed and measured.

## V. RESULTS

In this section we present the results of the benchmark scenario for varying device count. We set ISL data rate to 100 Kbps and the uplink traffic model assumes devices generates one 20 Bytes LORA frame every 40-120 seconds with an uniform distribution ( $\sim 1\%$  duty cycle). For the link budget, we consider LoRa standard power and antenna values, with a 12 dBi gain antenna at the satellite. On the downlink, the network server generates a 8 Bytes ACK sent to the sender of each received frames. Unless indicated otherwise, the chirp-based LORA modulation is configured with a spreading factor (SF) of 12 for both uplink and downlink, and a single LORA channel is assumed (thus no cross-channel interference due to Doppler shifts: the impact of channel effects are a future research topic). The simulated scenario spans 1 hour, enough for the satellite fleet to pass-by the whole circular region with up to 1500 end devices. Each simulation takes, on average, less than 2 wall-clock minutes.

a) *Collisions*: The plot in Figure 5 (a) presents the drop rate (due to collisions) in the uplink (*i.e.*, user frames) and the downlink (*i.e.*, ACKs) separately, and the resulting transmission failure (when the uplink or the downlink collides). Results show that 80% drop rate can be expected for 200 nodes under a highly demanding bidirectional traffic load.

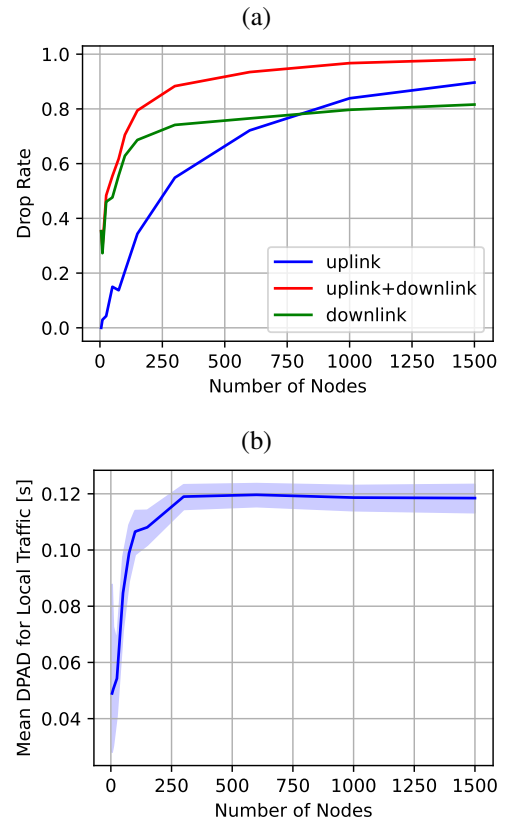


Fig. 5. Drop Rate due to collisions (a) and mean DPAD for frames destined to the local node (b) (shadowed area indicates 95% CI)

b) *DPAD vs Node Number*: DPAD for decoded downlink ACKs destined to the local node (*i.e.*, decoded packets destined to other nodes are not counted) are presented in Figure 5 (b). Results show that 100 kbps in the ISL (and the custom *right-up-down-left* routing) are enough to accommodate a multi-hop delivery of the downlink message without exceeding a DPAD of 0.12 seconds (an acceptable result given that the ACK still fits in the 1-second reception window).

c) *DPAD vs ISL data rate*: Figure 6 (a) presents a possible parametric study obtained from FLORASAT. The plot illustrates the variation of DPAD for different data rates for the ISLs connecting the 16-satellite fleet. It confirms that DPAD can approximate 0.5 seconds for data rates below 10 kbps – not a surprisingly low data rate for ISLs in resource-constrained CubeSats on link ranges of thousands of kilometers.

d) *DPAD vs SF*: The plot in Figure 6 (b) illustrates the impact on DPAD if other LORA spreading factors are used in the downlink, instead of the chosen SF=12. Node count is set to 150. As expected, lower SFs (more aggressive modulation) render shorter frames, and thus reduced propagation and processing latencies along the multi-hop LEO path.

e) *Wasted Decoding Effort*: Figure 6 (c) presents the effective decoding effort, namely, the percentage of decoded packets addressed to the local node (*i.e.*, 1 - wasted decoding effort). The plot confirms that for more than 150 nodes, there is a 50% probability that packets in the reception are not destined to the local node (wasted effort). The effect seems to stabilize around 70% of wasted effort for larger node counts. This is an interesting effect of LORAWAN multi-gateway configurations we plan to research as future work.

f) *ISL Utilization*: Finally, FLORASAT is able to compute the inter-satellite networking load, as illustrated in Figure 7. The figure highlights the high congestion in the link between nodes [14] and [15], due to the simple *right-up* routing scheme. Notably, for more than 200 nodes, the increased collisions on the radio access side reduces the ISL utilization of the satellite fleet, as less traffic need to be coursed to and from the network server. This output also favors an in-depth analysis of appropriate and generic routing solutions for DTS-IoT networks.

## VI. CONCLUSIONS

The main contribution of this paper is FLORASAT, a realistic simulation tool for DTS-IoT systems. We also showed that even in its preliminary form, FLORASAT, allows us to study a set of interesting metrics to characterize the performance of LORAWAN DTS-IoT networks. This is achieved thanks to unique orbital dynamics feature, empowered by realistic inter-satellite linking models, and accurate channel, link and end-to-end network layer abstractions. In particular, this promising satellite IoT simulator already enabled non-intuitive observations such as *i)* the effect of DPAD and *ii)* the issue of detecting preambles from frames destined to other nodes in a multi-gateway case. We identified several perspectives:

a) *LORA/LORAWAN*: Implementing new LORAWAN features such as Class-C mode would allow us to make a detailed Class-A, Class-B and Class-C comparison as well as LORA Doppler effects implications on the DTS-IoT architecture, and the limited demodulation paths on the orbiting gateway.

b) *LR-FHSS*: As discussed in recent related work [24], the new LoRa modulation, the LR-FHSS (long-range frequency hopping spread spectrum) [25] could increase the DTS-IoT scalability beyond thousand end-devices. FLORASAT will

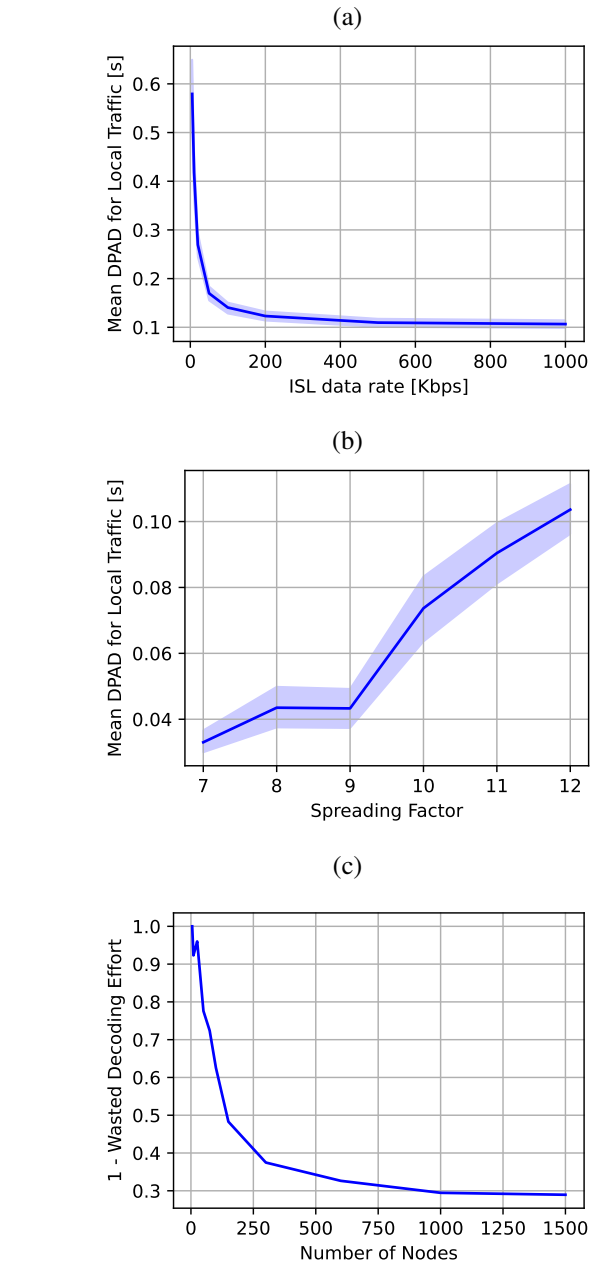


Fig. 6. Mean DPAD for local traffic for varying ISL data rate (a) and for varying SF (b). The Effective Decoding effort ratio (1 - Wasted Decoding Effort) is shown in (c) (shaded area indicates 95% CI)

enable a realistic comparison with legacy LORA and allow the evaluation of LR-FHSS implications in the end-to-end setup.

c) *NB-IoT*: The state-of-the-art of DTS-IoT presents a yet-to-be-solved competition between LORAWAN and NB-IoT [1], [6]. We plan to implement the right NB-IoT protocol stack abstractions in future FLORASAT.

d) *Scenarios*: Another appealing research axis involves the DTS-IoT satellite constellation design [26]. Since the current version of FLORASAT already allows to leverage STK [20], the study of different topologies (*e.g.*, including geostationary –GEO– relay satellites) is straightforward. This

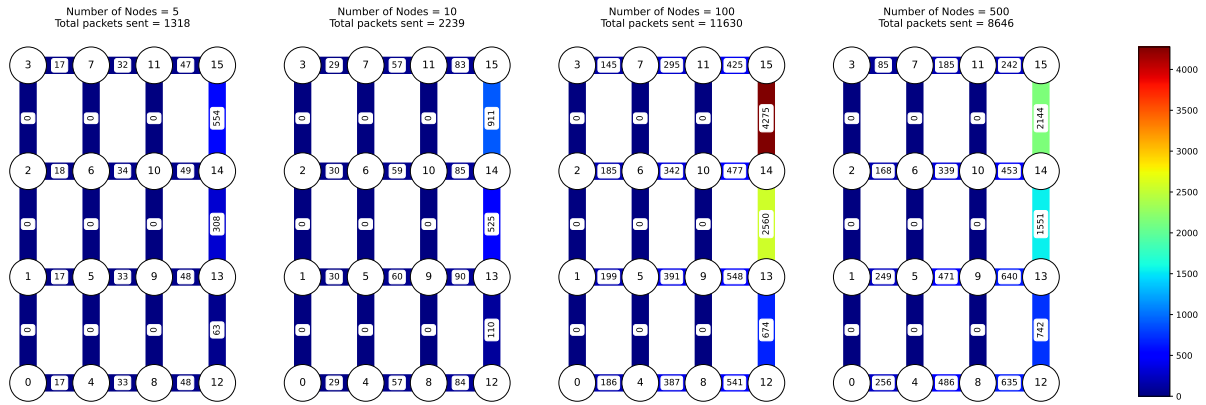


Fig. 7. ISL utilization.

task also include the analysis of sparse/connected constellations, the study of different routing schemes, and the evaluation of more specific satellite-to-ground station link protocols.

e) *Network functions*: Finally, the DTS-IoT topic and the FLORASAT tool unlocks unprecedented applied research opportunities to adapt and extend LoRa/LORAWAN and NB-IoT to the satellite context. These include novel scheduling approaches to profit from orbital predictability, as well as learning techniques that profit from the repetitive revisit characteristics of satellite IoT.

#### ACKNOWLEDGEMENT

This research has received support from the Project STI-CAMSUD 21-STIC-12, and the European Union's Horizon 2020 R&D programme under the Marie Skłodowska-Curie grant agreement No 101008233 (MISSION project).

#### REFERENCES

- [1] M. Centenaro, C. E. Costa, F. Granelli, C. Sacchi, and L. Vangelista, "A survey on technologies, standards and open challenges in satellite IoT," *IEEE Communications Surveys & Tutorials*, 2021.
- [2] M. R. Palattella and N. Accettura, "Enabling internet of everything everywhere: LPWAN with satellite backhaul," in *2018 Global Information Infrastructure and Networking Symposium (GIIS)*, 2018, pp. 1–5.
- [3] L. Ouvre, D. Lachartre, C. Bernier, F. Lepin, F. Dehmas, and V. Deslandes, "An ultra-low-power 4.7 ma-rx 22.4 ma-tx transceiver circuit in 65-nm CMOS for M2M satellite communications," *IEEE Transactions on Circuits and Systems II*, vol. 65, no. 5, pp. 592–596, 2018.
- [4] L. H. Trinh, N. V. Truong, F. Ferrero *et al.*, "Low cost circularly polarized antenna for IoT space applications," *Electronics*, vol. 9, no. 10, p. 1564, 2020.
- [5] M. De Sanctis, E. Cianca, G. Araniti, I. Bisio, and R. Prasad, "Satellite communications supporting internet of remote things," *IEEE Internet of Things Journal*, vol. 3, no. 1, pp. 113–123, 2015.
- [6] J. A. Fraire, S. Céspedes, and N. Accettura, "Direct-to-satellite IoT-A survey of the state of the art and future research perspectives," in *Int. Conf. on Ad-Hoc Networks and Wireless*. Springer, 2019, pp. 241–258.
- [7] O. Seller, "Wireless communication method," 2017, uS patent 9647718-B2, Semtech Corporation.
- [8] LoRa Alliance Technical Committee, "LoRaWAN 1.1 specification," LoRa Alliance, Tech. Rep., 2017.
- [9] C. Bouras, A. Gkamas, S. A. Katsampiris Salgado, and V. Kokkinos, "Comparison of LoRa simulation environments," in *Advances on Broad-Band Wireless Computing, Communication and Applications*, L. Barolli, P. Hellinckx, and T. Enokido, Eds. Springer International Publishing, 2020, pp. 374–385.

- [10] M. C. Bor, U. Roedig, T. Voigt, and J. 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*. New York, NY, USA: Association for Computing Machinery, 2016, p. 59–67. [Online]. Available: <https://doi.org/10.1145/2988287.2989163>
- [11] T.-H. To and A. Duda, "Simulation of LoRa in NS-3: Improving LoRa performance with CSMA," in *2018 IEEE International Conference on Communications (ICC)*, 2018, pp. 1–7.
- [12] M. Slabicki, G. Preamsankar, and M. Di Francesco, "Adaptive configuration of LoRa networks for dense IoT deployments," in *IEEE/IFIP NOMS*, 2018, pp. 1–9.
- [13] "AGI Systems Tool Kit (STK)," <http://www.agi.com/STK>.
- [14] S. P. Hughes, "General mission analysis tool (gmat)," <https://opensource.gsfc.nasa.gov/projects/GMAT>, Tech. Rep., 2016.
- [15] L. Maisonobe, V. Pommier, and P. Parraud, "Orekit: An open source library for operational flight dynamics applications," in *4th International Conference on Astrodynamics Tools and Techniques*, 2010, pp. 3–6.
- [16] M. Afhamisis and M. R. Palattella, "Salsa: A scheduling algorithm for LoRa to LEO satellites," *IEEE Access*, vol. 10, pp. 11 608–11 615, 2022.
- [17] O. Kodheli, N. Maturo, S. Andrenacci, S. Chatzinotas, and F. Zimmer, "Link budget analysis for satellite-based narrowband IoT systems," in *Int. Conf. on Ad-Hoc Networks*. Springer, 2019, pp. 259–271.
- [18] Z. Wang, G. Cui, P. Li, W. Wang, and Y. Zhang, "Design and implementation of NS3-based simulation system of LEO satellite constellation for IoTs," in *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, 2018, pp. 806–810.
- [19] C. Jin, X. He, and X. Ding, "Traffic analysis of LEO satellite internet of things," in *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*. IEEE, 2019, pp. 67–71.
- [20] J. A. Fraire, "Introducing contact plan designer: A planning tool for DTN-based space-terrestrial networks," in *2017 6th International Conference on Space Mission Challenges for Information Technology (SMC-IT)*. IEEE, 2017, pp. 124–127.
- [21] J. A. Fraire, O. De Jonckere, and S. C. Burleigh, "Routing in the space internet: a contact graph routing tutorial," *Journal of Network and Computer Applications*, vol. 174, p. 102884, 2021.
- [22] K. Vogelgesang, J. A. Fraire, and H. Hermanns, "Uplink transmission probability functions for LoRa-based direct-to-satellite IoT: A case study," in *IEEE GLOBECOM*.
- [23] J. Janhunen, J. Ketonen, A. Hukkoniemi, J. Ylitalo, A. Roivainen, and M. Juntti, "Satellite Uplink Transmission with Terrestrial Network Interference," in *IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–6.
- [24] M. A. Ullah, K. Mikhaylov, and H. Alves, "Analysis and simulation of LoRaWAN LR-FHSS for direct-to-satellite scenario," *IEEE Wireless Communications Letters*, 2021.
- [25] G. Boquet, P. Tuset-Peiró, F. Adelantado, T. Watteyne, and X. Vilajosana, "LR-FHSS: Overview and performance analysis," *IEEE Communications Magazine*, vol. 59, no. 3, pp. 30–36, 2021.
- [26] J. A. Fraire, S. Henn, F. Dovis, R. Garello, and G. Taricco, "Sparse satellite constellation design for LoRa-based direct-to-satellite Internet of things," in *IEEE GLOBECOM*. IEEE, 2020, pp. 1–6.