LoRaOpp: A Protocol for Opportunistic Networking and Computing in LoRa Networks

Nicolas Le Sommer*
*IRISA, Université Bretagne Sud
France
Email: Nicolas.Le-Sommer@univ-ubs.fr

Lionel Touseau†*
†CReC, Saint-Cyr Coëtquidan Military Academy
France
Email: Lionel.Touseau@st-cyr.terre-net.defense.gouv.fr

***Abstract*—LoRa is one of the major communication technologies for Low Power Wide Area Networks, and a convenient technology for the Internet of Things. LoRa is usually associated with LoRaWAN, which is a protocol relying on a star network topology, in which end-nodes are connected through one-hop wireless links to one or several gateways. Despites the long range characteristics of the LoRa physical layer, physical obstacles, interferences and mobility can sometimes prevent end-nodes to communicate with gateways through single-hop links.**

This paper presents a new opportunistic protocol for LoRa, called LoRaOpp, that supports multi-hop communications between pairs of nodes and between nodes and gateways, even without end-to-end paths between them. This paper also presents experimental results obtained for LoRaOpp in real conditions and in emulation.

***Index Terms*—IoT, Opportunistic Networking, LoRa**

I. Introduction

Over the last years, Low-Power Wide Area Networks (LPWANs) have received much attention from the research community and the industry for IoT applications dedicated to various domains, such as smart cities, smart agriculture and smart industry. Among existing communication technologies for LPWANs, LoRa is one of the most promising standards. It is robust to interferences, it utilizes unlicensed spectrum sub-GHz ISM bands, and it provides symmetric modulation for uplink and downlink, thus facilitating device-to-device communications and the construction of ad hoc networks. Some experiments showed that the LoRa communication range can be longer than 10 km in advantageous outdoor conditions (i.e., without physical obstacles between end-nodes and gateways), but also that it can be drastically reduced when devices are deployed in environments with natural or artificial obstacles (e.g. mountains, buildings, undergrounds). The simple star network topology considered with LoRaWAN, in which nodes and gateways must be in the radio range of each other to communicate, can be limiting in certain challenging environments. Several works have investigated these last years solutions relying on multi-hop transmissions [1] in order to extend the network coverage in such environments, without having to deploy additional gateway devices. Many of these works rely on assumptions that make them appropriate only for specific use cases. For instance, they often assume that the periods of activity and of communication of nodes are synchronized in order to create end-to-end paths between nodes and gateways. A few number of these works [2], [3], [4] implement opportunistic networking techniques to support the temporal, and sometimes the spatial, disruptions that can occur in the communication paths due to the mobility or to the sleep phases of the nodes acting as relays. But here again they rely on strong assumptions such as the fact that nodes must always be up to receive beacons from the gateways in order to send them their data [3].

In this paper, we propose a new multi-hop routing protocol for LoRa, called LoRaOpp. This protocol implements opportunistic networking techniques in order to allow devices to exchange data even if there is no end-to-end path between them at anytime. It combines several mechanisms to efficiently forward packets in the network while limiting the number of packets that are disseminated in this one. Some of these mechanisms have been studied in past research works dealing with disruption-tolerant and opportunistic networking [5] (e.g. source routing, "spray and focus" approach [6], limitation of the number of hops and of the lifetime of packets), but they have not been considered so far in works implementing opportunistic networking techniques in LoRa-based protocols [2], [3], [7], [4]. With this protocol we want to evaluate the communication performances that can be obtained, notably in terms of delivery time and ratio, on various network setups without requiring a synchronization of the activity and communication periods of nodes, and while complying with sub-Ghz regulation – 1 % duty cycle must be observed.

The rest of this paper is organized as follows. Section II presents research works investigating LoRa opportunistic communications. Section III details the main features of LoRaOpp. The experimental results we obtained for this protocol, both in real conditions and in emulation, are presented in section IV. Section V concludes this paper by summarizing our contribution and by mentioning the future improvements we would like to make for LoRaOpp.

II. Related Work

Works [8], [9], [10], [11], [12], [13] have investigated and proposed solutions to support multi-hop communications in LoRa networks, mainly with the aim of creating and maintaining end-to-end paths between the nodes and the gateways. These solutions rely on strong assumptions, such as the synchronization of the activity and communication periods

978-1-6654-6975-3/22/$31.00 ©2022 IEEE

of nodes and on relay devices that are always up and powered. To overcome these constraints and to support the volatility and the mobility of nodes and gateways, which induce disruptions in communication paths, a small number of works have considered opportunistic communication techniques [7], [3], [4] to forward packets in intermittently-connected networks.

In [7], Almedia et al. proposed a platform for data gathering in smart cities using both mobile and static devices equipped with Wi-Fi and LoRa interfaces. Three types of devices are considered in this work: 1) the Data Collecting Units (DCU), which are static or mobile devices equipped with sensors, 2) the sink stations, which are fixed devices that are connected to a server accessible from the Internet, 3) and mobile nodes that serve as relays to opportunistically deliver data from the DCU to the sink stations. This platform does not rely on LoRaWAN, but on an alternative LoRa MAC protocol designed to exploit the multiple sink stations that are present in the network. In [3], Florita et al. also investigated data gathering in smart cities with an opportunistic communication system. This system is composed of two types of devices: fixed sensor nodes and mobile gateways, which are used to collect data from sensors. Gateways have two types of wireless communication interfaces: a LoRa interface to communicate with the sensor nodes, and a Wi-Fi interface to upload collected data to a server connected to the Internet. Unlike the system proposed in [7], which uses either Wi-Fi or LoRa communications to exchange data between DCU and mobile devices, this one only supports LoRa communications between sensor nodes and mobile gateways for energy saving purposes. These gateways are embedded in vehicles moving around the city, allowing a node to send its data when the gateway is in its communication range. This communication system includes a new MAC layer protocol for the communications between a sensor node and a gateway, enabling a pull-based transfer of data. In [7] and [3], opportunistic communications are only considered between nodes and mobile devices, while they could also be applied between fixed nodes as proposed in [4]. Opportunistic networking and network coding techniques are combined in [4] in order to avoid a bottleneck that could be created by individual emissions of packets. In order to reduce unnecessary transmission between nodes and relays, an on demand feedback mechanism is implemented in a LoRaWAN forwarder. Relays only forward packets that are missing in feedback messages. Disruption-tolerant and opportunistic networking techniques studied in past research works [5] have not been considered so far in previous works, while they could help addressing packets forwarding issues in intermittently-connected LoRa networks. Moreover, these works do not investigate the impact of the variation of the number of nodes and gateways on the communication performances, knowing that duty-cycling constraints must be observed for LoRa on sub-GHz ISM bands.

III. The LoRaOpp protocol

LoRaOpp is an opportunistic communication protocol designed to ensure data transmission in intermittently-connected LoRa networks, such as the network illustrated in Figure 1. LoRaOpp supports multi-hop data exchange between the nodes themselves and between the nodes and the gateways. LoRaOpp combines several methods and techniques in order to limit the number of packets that are forwarded in the network while providing efficient delivery times and ratios, such as source routing, transmission feedback techniques and the "spray and focus" method [6]. It also implements several strategies to select the data transmission paths. The remainder of this section outlines how the protocol LoRaOpp works and presents its main features.

A. Devices discovery

Three types of devices are currently considered in LoRaOpp, namely end-nodes, peer-nodes, and gateways. Both nodes and gateways are involved in the packet forwarding process. Transmission strategies of data packets and of presence advertisement packets depend on the type of the device. In use cases envisioned for LoRaOpp, end-nodes can measure and process physical quantities before transferring them to a server through gateways. They can access gateways directly or via intermediate nodes. Conversely, remote servers can send data to end-nodes through gateways and intermediate nodes. Like end-nodes, peer-nodes can transfer the physical quantities they have measured and processed to a server through gateways, can receive packets from this server, and on top of that, they can communicate with each other. They allow to build more complex network topologies than the star network topology considered by LoRaWAN. If the next intermediate device towards the destination is not reachable at the forwarding time, gateways and nodes temporarily store the packets that must be forwarded, and wait for a next contact opportunity to forward them.

Gateways announce their presence to the nodes by broadcasting dedicated packets at several hops in the network. Before exchanging data with their direct neighbors, end-nodes advertise their presence at one hop. End-nodes will be discovered by gateways after sending them a packet. Gateways can deduce from these packets the distance between them and the nodes expressed in number of hops and time. As far as peer-nodes are concerned, they broadcast their advertisements at several hops in order to be discovered by the nodes with which they are expected to exchange data and to form a distributed IoT system. Developers of such IoT systems can specify the maximum number of hops that must be considered for the discovery of the nodes composing their systems and for transmission of data packets.

B. Forwarding algorithm

The general principle of the LoRaOpp's forwarding algorithm executed by the nodes is presented in Algorithm 1. It combines several transmission strategies depending on the type of the devices and on the type of packets that are emitted or received by the node. It is executed when the node starts for the first time, and whenever it wakes up.

A LoRa node will only receive the packets transmitted by the neighbor node having the strongest transmission signal when several of its neighbor nodes emit their packets simultaneously. Consequently, packets sent by some nodes can be lost if they are not retransmitted. To avoid these unsuccessful transmissions, and knowing that LoRa node transceivers operate on half-duplex communications, LoRaOpp implements both a collision detection mechanism based on a channel listening, and a retransmission process based on a transmission feedback (i.e. on an acknowledgment listing the packets that have been received). Emissions of packets are constrained by the duty cycling that must be observed on ISM frequency bands lower than 1 GHz.

LoRaOpp makes it possible to specify both the wake-up periods and the running times of the nodes. These parameters are taken into account in the transmission process, and have therefore an impact on the energy consumption of the nodes and on the packet transmission rate. They allow to mimic the behavior of class A, B or C nodes defined in the LoRaWan protocol.

Algorithm 1 Algorithm applied on node startup or wake-up.

```
rt: the node running time
pkt: the packet that has been recevied
next_snd_time: the next time the node can send a packet considering
the duty cycling

next_sleep_time←current_time+rt
repeat
    rcv_delay← max(compute_backoff(), next_snd_time - current_time)
    rcv_delay← min(rcv_delay, next_sleep_time - current_time)
    pkt←rcv_pkt(rcv_delay)
    if pkt = NULL then
        // send advertisement
        if next_advertisement_time < current_time then
            send_advertisement()
            next_advertisement_time←current_time+device_advertisement_period
        if packets_to_acknowledge then
            send_ack()
        // notification of the application that data packets will be forwarded
        application.on_forward()
        // Forwarding of data packets
        snd_start_transfer_pkt()
        for each pkt ∈ pkt_cache do
            snd_pkt(pkt)
        snd_end_transfer_pkt()
        // wait for the reception of an acknowledge packet
        pkt←rcv_pkt(min(next_sleep-current_time, MAX_ACK_RCV_DELAY))
        if pkt ≠ NULL then
            process(pkt)
    else
        process(pkt)
until next_sleep_time ≥ current_time()
```

C. LoRaOpp packets processing

Six types of packets are currently considered in LoRaOpp, which can respectively be used by nodes and gateways to advertise their presence in the network (`hello` packets) or their (temporary) leaving from this one (`bye` packets), to send data (`data` packets), to acknowledge the reception of data packets (`ack` packets), and to notify their direct neighbor when they start and stop data transmissions (`start_transfer` and `end_transfer` packets). Packets are implemented as payload of the physical layer.

Method *process(pkt)* used in Algorithm 1 behaves as described below. It's algorithm is not detailed due to the lack of space. The packets that are received by the nodes are processed according to their type. `Hello` carry information required by devices to keep their routing table up to date. Entries of routing tables are composed of the ID, the type of the target device, the path to reach this device, the number of hops to the target device, the expected transmission delay to send data to this one, and the last update time of the entry. Packets that can be forwarded in the network (i.e. Packets whose number of hops is greater than 1 and whose lifetime has not expired) are put into the cache of the devices in order to be forwarded by these ones when they will be able to.

When nodes gracefully leave the network, temporarily or definitively, they broadcast `bye` packets. `Bye` packets are processed similarly to the `hello` packets, except that they are used to remove information about the leaving devices from the routing tables. When a device is about to forward the packets stored in its cache, it sends a packet of type `start_transfer` to advertise its direct neighbors of the amount of data it will send, thus allowing them to define adaptive listening windows to receive all packets, or only a part of them if they must be put in sleep mode before receiving all packets. If all the data have not been received, the emission of the `ack` packet is postponed. When a device receives an `end_transfer` packet, it sends back an `ack` packet to acknowledge the `data` packets it received, thus allowing both the initial emitter and the intermediate forwarders to remove these packets from their cache. Each time a device receives a `data` packet, it checks if this packet is still valid and has not already been received and relayed. If not, it checks if it is the final recipient of this packet. If so, it forwards the packet to the application for processing purposes. Otherwise, it decreases the number of hops before storing the packet into its local cache, and marks this one as to be forwarded and acknowledged for the next emission phase. When end-nodes want to send `data` packets to a remote application server, they look up their routing table to find the best route to reach a gateway, and send the packet to this one specifying the route to follow. Peer-nodes proceed in a similar way when they want to send `data` packets to a remote application server or to another peer-node.

IV. Experiments and Results

An outdoor experiment and several experiments in emulation were conducted in order to evaluate the performances of LoRaOpp and its scalability. The performance criteria we have sought to evaluate during these experiments are the delivery ratios and the delivery times of data packets. The delivery ratio is the number of packets received by their recipient(s) out of the number of packets that have been sent. The delivery time is the time between the moment a packet is received by its destination and the moment when the packet is sent by the source. The LoRaOpp's parameters and the number of nodes and of gateways used in these experiments are listed in Table I. In all these experiments only one copy of each packet is initially sent by the source node, and no additional copies

Parameters	Value
LoRa bandwidth	125 kHz
LoRa transmission power	2 dBm
LoRa coding rate	4/5
LoRa frequency band	868 MHz
LoRa spreading factor	7
Number of nodes	6, 25, 50, 100
Number of gateways	1, 2, 5, 10
Number of initial copies	1
Number of extra new copies	0
Number of hops	10
Path selection strategy	hop-based
Node wake up period	[5,15] minutes
Node activity time	[1,4] minutes

TABLE I
EXPERIMENT PARAMETERS.

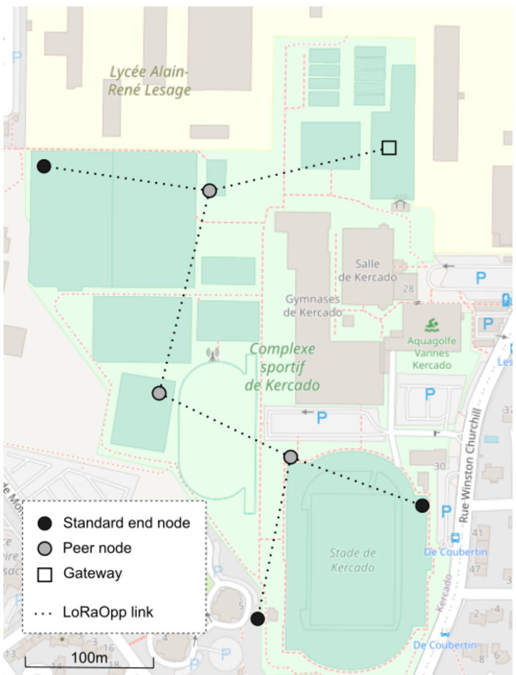


Fig. 1. Network topology adopted for outdoor experiments.

are created in the network by the intermediate nodes. The selection of the paths that must be followed by data packets is based on the number of hops between the source and the destination. The paths with less hops are selected. A 1 % duty cycle is observed to comply with sub-Ghz regulation (for all packet transmissions, including hello and bye packets).

A. Outdoor experiments

The outdoor experiment was conducted on a large sports complex in the city of Vannes, France. The network was composed of 7 Heltech ESP32 devices equipped with a SX1276 LoRa node chip (868 Mhz), and was deployed following the topology shown in Figure 1. Three of the seven devices act as end-nodes, three as peer nodes, and one device as a gateway. The wake up and running periods of nodes are defined respectively between 5 and 15 minutes, and between 1 and 4 minutes. The experiment lasted for 2 hours. At each wake up periods, the nodes generate a data packet that is intended for a gateway, and whose payload is a string of 32 bytes including a JSON object that carries a temperature measurement.

Measure	Median	Average
Data reception delay (s)	28.49	33.42
Data reception ratio (%)	100	100
Number of hops	2	2.5

TABLE II
RESULTS OF THE EXPERIMENT CONDUCTED IN REAL CONDITIONS.

The averages and the medians of the delivery delays and ratios of data packets that were measured during the experiment, are presented in Table II. All the data packets sent by the nodes were received by the gateway. They were delivered with an average delay and a median delay of 334 and 285 milliseconds respectively. The median and the average values of the hops that were necessary to deliver these packets are 2 and 2.5. Thus, the average of transmission delay per hop is around 133 milliseconds, knowing that the optimal transmission delay for a data packet with a payload of 32 bytes with a spreading factor of 7, a coding rate of 4/5, a signal bandwidth of 125 kHz and a frequency band of 868 MHz is 95 milliseconds. The best value obtained in this experiment for the one-hop transmission delay is consistent with this theoretical value. The best values obtained for several hops are also consistent with the theoretical values related to the implementation of LoRaOpp, and which follow the formula $t^h_{data} = h * (t_{ack} + t_{data})$, where h is the number of hops between the source and the destination, and t_{ack} and t_{data} are respectively the transmission delays for the acknowledgment and the data packets at one hop. In these favorable transmissions, the period of emission and reception of devices are in phase, allowing to build an end-to-end path between the source and the destination.

B. Performance evaluation in emulation conditions

In order to evaluate LoRaOpp in larger networks, we have conducted experiments in emulation. The code running on emulated nodes is the same as the code running on actual nodes. The behavior of the LoRa physical layer is reproduced as accurately as possible using a software stack. For comparison purposes, we first evaluated LoRaOpp in emulation with a network topology comparable to that considered in the outdoor experiment, and on 4 additional randomized network topologies also composed of 6 nodes and 1 gateway. Each setup was run 10 times, with randomly varying node runtime periods and wake up intervals. The results obtained in emulation for a network topology similar to the one considered in the outdoor experiment, are very close to those obtained in this experiment. The results obtained for the other randomized network topologies, gave us slightly better results. The delivery ratios also reached 100%. The average of the number of hops needed to deliver data packets dropped from 2.5 to 2.2 in emulation, and the average and the median of the delivery

Configuration (nodes/gateways)	Traffic ratio (ER/LoRaOpp)
6/1	2.02
25/2	6.31
50/5	20.04
100/10	72.84

TABLE III
TRAFFIC RATIOS BETWEEN THE EPIDEMIC ROUTING AND THE LORAOPP PROTOCOLS.

times dropped respectively from 33.42 and 28.49 seconds in actual conditions to 21.20 and 14.06 seconds in emulation. These better results can be explained by the fact that in some topologies nodes are closer to the gateway, what causes packets to be forwarded faster.

Three additional configurations, with specifically different node/gateway distributions, namely 25 for 2, 50 for 5, and 100 nodes for 10 gateways, have been considered in order to assess the efficiency of the LoRaOpp protocol in the gateway and path selection process. These configurations were each run on 5 randomized network topologies. Each setup was run 10 times, with randomly varying node runtime periods and wake up intervals. The results of these 150 additional runs were evaluated on the same criteria as the actual experiment. As works presented in [7], [3], [4] are not publicly available, and thus cannot be tested in real and emulation conditions, we executed, for comparison purposes, on the same setups the Epidemic Routing (ER) protocol [14], which is an opportunistic protocol known to provide the best results in terms of delivery times and ratios, but as being a bad one in terms of network load as packets are forwarded by, and replicated on, all the nodes of the network.

The results are presented in Figure 2. They show that the performances provided by the LoRaOpp and the ER protocols are slightly impacted by the increase of the number of nodes in the network, and by the resulting larger number of packets that are exchanged. Few variations in the packet delivery times and rates can indeed be observed.

On one hand, the shorter delivery times for an increasing number of nodes can be explained by the fact that, for very similar contact durations, there is more contact opportunities as each node has more neighbors (respectively an average of 1, 2, 4 and 9 neighbors for the nodes/gateways configurations 6/1, 25/2, 50/5 and 100/10). Therefore it is easier to find a route to reach the destination. The increase in the number of gateways and therefore of possible destinations also contributes to this performance improvement.

On the other hand, we can notice that the packet delivery rate decreases inversely with the number of nodes. For example for LoRaOpp, 100% of packets are delivered for network topologies composed of 6 nodes and 1 gateway, while the average delivery ratio drops to 93% for 100 nodes and 10 gateways. The increase in the number of possible destinations, through the increase in the number of gateways, however slightly improves the delivery rate, as observed when comparing the results for 100 nodes and for 50 nodes. Less hops are indeed required to deliver a packet for 100 nodes and 10 gateways than for 50 nodes and 5 gateways.

To confirm this hypothesis, we conducted other experiments to study how LoRaOpp behaves when the number of destinations varies while the number of nodes remains the same. In these experiments we considered 1, 2, 5 and 10 gateways as destinations in a network composed of 100 nodes. The results are presented in Figure 3. They show that the LoRaOpp protocol can benefit from the presence of several gateways in the network to improve packet delivery. They also show that the more gateways there are, the less nodes are solicited to forward packets. As a consequence, more packets can overall be delivered because the retransmission load is more evenly distributed across the network, and therefore the duty cycle limit is reached less quickly on each node.

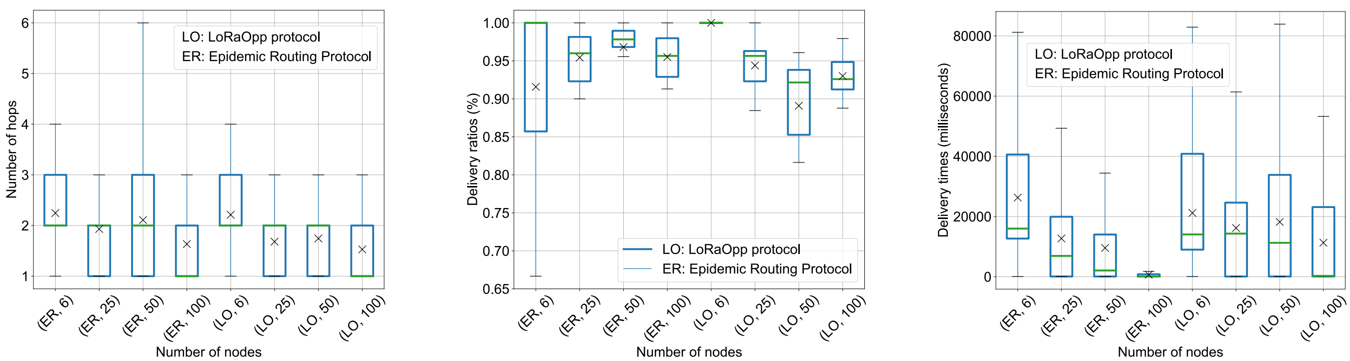
In addition, Figure 2 highlights that even though ER shows better performances than LoRaOpp, it however causes a significantly higher network traffic as presented in Table III. ER generates 2.02, 6.31, 20.04 and 72.84 times more packets than LoRaOpp does in the considered configurations. The better performances achieved with ER can be explained by the fact that ER allows all nodes to forward packets in the network, while LoRaOpp only uses the nodes considered to be the best relays. ER thus makes it possible to use nodes that are less relevant for routing, but which are able to forward packets because they have not reached their transmission time limit set by the duty cycle.

In consideration with the previous observations, it seems interesting to take into account the ability of nodes to retransmit data when defining forwarding paths (i.e., when selecting the nodes that will be used as relays). Paths should therefore be also selected according to their transmission rate, and not solely on the basis of the number of hops and/or the forwarding delays recorded between the source and the destination.

V. CONCLUSION AND FUTURE WORK

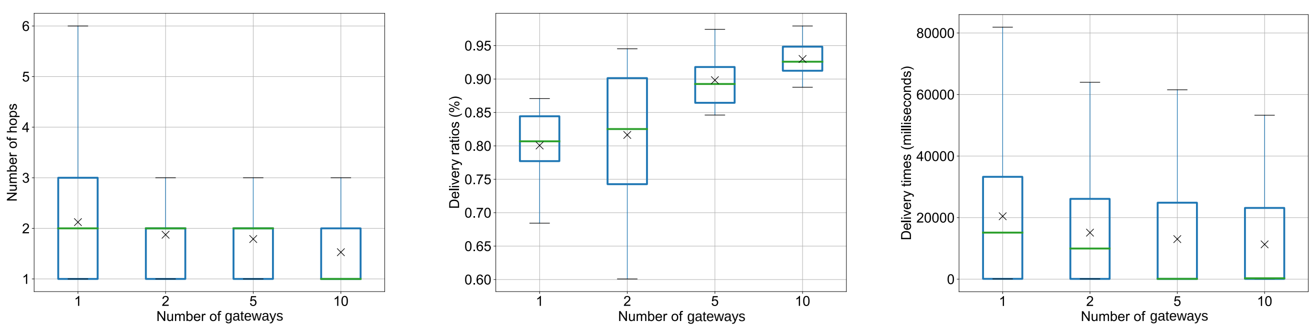
In this paper we proposed an opportunistic multi-hop routing protocol for LoRa-based intermittently-connected networks, called LoRaOpp. This protocol is versatile enough to be used in the most diverse situations. It does not require a specific network topology. Different types of nodes and gateways can form the network. It supports multi-hop data exchange between the nodes themselves and between the nodes and the gateways, and it implements several techniques and methods to limit the number of packets that are forwarded in the network. The experiments conducted both in real conditions and in emulation have shown that the LoRaOpp protocol performs well in different setups, and that it allows to forward packets at several hops, even if the period of activity of devices are not synchronized, and therefore even if there is no end-to-end paths between devices at a given time.

In future work, we would like to improve the performances of LoRaOpp and to introduce LoRa nodes running on the 2.4 GHz band. LoRaOpp multi-hop routing would be even more relevant with such nodes, since the communication range is shorter than with sub-GHz frequencies, while an higher



(a) Number of hops for packet delivery. (b) Packet delivery ratios. (c) packet delivery times.

Fig. 2. Performance results of the LoRaOpp protocol *vs* the Epidemic Routing protocol for 6 nodes/1 gateway, 25 nodes/2 gateways, 50 nodes/5 gateways, 100 nodes/10 gateways.



(a) Number of hops for packet delivery. (b) Packet delivery ratios. (c) Packet delivery times.

Fig. 3. Performance results of LoRaOpp for 100 nodes and 1, 2, 5 and 10 gateways.

throughput and no duty cycle constraints would make data exchange between nodes more efficient.

REFERENCES

[1] J. R. Cotrim and J. H. Kleinschmidt, "LoRaWAN Mesh Networks: A Review and Classification of Multihop Communication," *MDPI Sensors*, vol. 20, no. 15, pp. 1–21, Jul. 2020.

[2] M. Bor, J. Vidler, and U. Roedig, "LoRa for the Internet of Things," in *the 2016 International Conference on Embedded Wireless Systems and Networks (EWSN'16)*, Graz, Austria, Feb. 2016, pp. 361–366.

[3] N. J. B. Florita, A. N. M. Senatin, A. M. A. Zabala, and W. M. Tan, "Opportunistic lora-based gateways for delay-tolerant sensor data collection in urban settings," *Computer Communications*, 2020.

[4] D. Tanjung, S. Byeon, D. H. Kim, and J. Kim, "OODC: An Opportunistic and On-Demand Forwarding Mechanism for LPWA Networks," in *the 34th International Conference on Information Networking (ICOIN)*, Barcelona, Spain, Jan. 2020, pp. 301–306.

[5] V. F. S. Mota, F. D. Cunha, D. F. Macedo, J. M. S. Nogueira, and A. A. F. Loureiro, "Protocols, Mobility Models and Tools in Opportunistic Networks: A Survey," *Computer Communications*, vol. 48, pp. 5–19, 2014.

[6] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility," in *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops '07. Fifth Annual IEEE International Conference on*, Mar. 2007, pp. 79–85.

[7] R. Almeida, R. Oliveira, D. Sousa, M. Luis, C. Senna, and S. Sargento, "A Multi-Technology Opportunistic Platform for Environmental Data Gathering on Smart Cities," in *2017 IEEE Globecom Workshops*. Singapore: IEEE, Dec. 2017, pp. 1–7.

[8] J. Dias and A. Grilo, "Lorawan multi-hop uplink extension," in *the 8th International Conference on Ambient Systems, Networks and Technologies (ANT 2018)*, ser. Procedia Computer Science, vol. 130. Science Direct, Elsevier, 2018, pp. 424–431.

[9] R. El Chall, S. Lahoud, and M. El Helou, "LoRaWAN Network: Radio Propagation Models and Performance Evaluation in Various Environments in Lebanon," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2366–2378, Mar. 2019.

[10] W. Zhou, Z. Tong, Z. Y. Dong, and Y. Wang, "LoRa-Hybrid: A LoRaWAN Based Multihop Solution for Regional Microgrid," in *the 4th IEEE International Conference on Computer and Communication Systems (ICCCS 2019)*, Singapore, Feb. 2019, pp. 650–654.

[11] M. O. Farooq, "Introducing Scalability in LoRa-Based Networks through Multi-Hop Communication Setups," in *IEEE Global Communications Conference (GLOBECOM 2019)*, Hawaii, USA, Dec. 2019, pp. 1–6.

[12] C.-H. Liao, G. Zhu, D. Kuwabara, M. Suzuki, and H. Morikawa, "Multi-Hop LoRa Networks Enabled by Concurrent Transmission," *IEEE Access*, vol. 5, pp. 21 430–21 446, 2017.

[13] A. Abrardo and A. Pozzebon, "A Multi-Hop LoRa Linear Sensor Network for the Monitoring of Underground Environments: The Case of the Medieval Aqueducts in Siena, Italy," *Sensors*, vol. 19, no. 2, 2019.

[14] A. Vahdat and D. Becker, "Epidemic Routing for Partially Connected Ad Hoc Networks," Duke University, Durham, USA, Tech. Rep. CS-200006, Apr. 2000.