Enabling RPL multihop communications based on LoRa

Benjamin Sartori
ETRO, Vrije Universiteit Brussel
Brussels, Belgium
imec, Leuven, Belgium
Email: bsartoria@etrovub.be

Maite Bezunartea
ETRO, Vrije Universiteit Brussel
Brussels, Belgium
imec, Leuven, Belgium
Email: mbezunar@etrovub.be

Steffen Thielemans
ETRO, Vrije Universiteit Brussel
Brussels, Belgium
imec, Leuven, Belgium
Email: sthielem@etrovub.be

An Braeken
INDI, Vrije Universiteit Brussel
Brussels, Belgium
Email: an.braeken@vub.ac.be

Kris Steenhaut
ETRO, Vrije Universiteit Brussel
Brussels, Belgium
imec, Leuven, Belgium
Email: ksteenha@etrovub.be

***Abstract*—New Long-Range radio technologies have recently emerged in the IoT landscape. These technologies work in the Sub-GHz bands, allowing low-power communications over long distances. They are typically based on star-topology networks, where nodes send the data directly to a base station connected to the Internet. One of such technologies is LoRa. Enabled LoRa-based multihop communications would open up new possibilities. In this paper we present a solution allowing to have multihop LoRa communications. This is done by means of a newly designed MAC protocol (RLMAC), used to select the Spreading Factor for each available neighbor. The proposed Objective Function used by RPL will therefore be able to select the routing path that minimizes time on-air. By selecting the path with the lowest time on-air, the power consumption can be decreased, enlarging network lifetime. Preliminary validation tests are also presented in the paper.**

***Keywords*-LoRa; IPv6; RPL; Multihop communications; MAC.**

I. Introduction

The interest in and the market of the Internet of Things (IoT) and its related technologies is indeed steadily increasing. As it is well known, the IoT aims at connecting and automating every aspect of daily life. Some of the most important IoT applications are: predictive maintenance, resource control, natural disaster prevention or fast detection, medical monitoring, intelligent building and smart home, production chain control and monitoring, etc.

Within IoT, Low-power and Lossy networks (LLNs) are of great importance. These are networks of spatially distributed autonomous motes used to measure and monitor specific physical or chemical environmental magnitudes, and to send this information wirelessly to a control node to process or store it. They are meant to be directly connected to the Internet to guarantee interoperability, ease data exchange through the use of simple and well-known web technologies such as Web Services, offer easy management and configuration options and make each sensor individually reachable from outside its network via its own IP address.

In order to be able to achieve this, it is necessary to create and develop adequate and efficient communication protocols and radio technologies specifically designed for meeting the requirements of LLNs, as opposed to traditional IP-based networks. That is the reason why many standardization efforts have been made to fully integrate resource-constrained mesh networks in the Internet, using IPv6 as network protocol (6LoWPAN, the RPL routing protocol, CoAP, etc).

Recently, new long-range, low-bitrate radio technologies have emerged in the IoT landscape [1]. These new radio technologies work in the sub-1GHz unlicensed ISM bands, offering power-efficient communication over long distances. Examples of such new technologies are LoRa, Sigfox and Weightless. They are typically networks of nodes communicating the data to a base station in a single-hop manner. The so created star topology is convenient for the ease of deployment and has advantages from a business perspective. Being able to have multi-hop between nodes though, could mitigate some upcoming issues of bandwidth congestion [2].

Moreover, multi-hop may be the only alternative for instance in wildlife monitoring application covering very large areas with a few base stations or LLNs located in

978-1-5386-3839-2/17/$31.00

the mountain or other environment where path loss is high. More generally, multi-hop can increase the amount of data sent or reduce the time needed to send the same amount of data by using lower LoRa Spreading Factors (SFs).

In this paper we present an adaptation to the typical protocol stack for LLNs to enable RPL-based multihop communications on top of LoRa [3], calculating the optimal per-link Spreading Factor (SF). On the one hand, we propose a modification of one of the standardised RPL Objective Functions (OF0) in order to compute rank, using the selected LoRa SF as routing metric. On the other hand, we propose a MAC mechanism to select the optimal SF per link. The proposed implementation is realized in in the well-known open-source ContikiOS.

The paper is structured as follows. Section II introduces the technical LoRa radio specifications and its particularities. Section III presents the RPL routing protocol. Section IV reviews the context and related work. Section V introduces the different issues arising from having LoRa-based multihop communications. Section VI introduces the proposed MAC and RPL solution. A first validation test for the presented solution is included in section VII. Finally, section VIII concludes and gives suggestions for future work.

II. Long Range (LoRa) Communications

LoRa stands for Long Range. LoRa provides long-range, low-power and low-bitrate communications. LoRa is one of the most popular Low Power Wide Area Network (LPWAN) technologies today and was developed by Semtech.

LoRa is the combination of two distinct technologies: the LoRa physical layer and the LoRaWAN MAC protocol. The former specifies the physical modulation, whereas the latter specifies a secure ALOHA-based MAC protocol. The LoRaWAN specification can be particularly interesting for many IoT applications with low bandwidth requirements, allowing straightforward and cheap one-hop communication. In this study, we will focus on the LoRa physical layer, since the LoRaWAN MAC is not used in the proposed solution.

The LoRa RF physical layer uses a kind of spread spectrum modulation, called Chirp Spread Spectrum (CSS), which is a proprietary modulation technique. This CSS modulation offers a good resilience to interference and is Doppler-resistant [4] [5]. Depending on the geographical region, it operates in different ISM bands. This is summarized in Table I.

Table I
LoRa Frequency bands

	ISM Band
Europe	868 MHz
North America	915 MHz
Asia	433 MHz

When working with radio technologies transmitting in ISM bands, some regulatory limitations should be met. In particular, for the 868 MHz band, the Duty Cycle (maximum transmitter activity time, expressed as a percentage) is limited to a maximum of 1% of the time.

The LoRa PHY layer is especially interesting since several parameters can be configured in order to optimize wireless data transmission. Table II shows the most important reconfigurable parameters, being bandwidth, transmission power and spreading factor.

Table II
LoRa PHY Configurable Parameters

	Values
Spreading Factor	6 to 12
Bandwidth	125kHz, 250kHz and 500kHz
Transmission Power	-1dBm to 14dBm

One of the most interesting features of LoRa is that it offers the possibility of trading throughput for link budget. This can be done by adapting the Spreading Factor (SF), with values ranging from 6 to 12, where each spreading factor is orthogonal. The combination of different frequencies with different SFs defines what we call ”sub-channels”. The SF controls how many chips will be used to represent a symbol. A symbol encodes SF bits in the LoRa modulation. The higher the spreading factor, the better the sensitivity and the lower the throughput. An example of different sensitivities and corresponding throughput that can be achieved for the SX1272 radiochip, in function of the bandwidth and the spreading factor, is shown in Table III.

Table III
Example of LoRa performances with a 4/5 coding rate obtained from [6]

Bandwidth (kHz)	**Spreading Factor**	**Sensitivity** (dBm)	**Throughput** (bps)
125	6	-122	9375
125	12	-137	293
250	6	-119	18750
250	12	-134	586
500	6	-116	37500
500	12	-131	1172

The drawback of achieving a better sensitivity is that the symbol period increases with the spreading factor. Obviously, increasing the symbol period implies that the time on air and the power consumption will also increase. The symbol period (Ts) can be calculated as follows:

$$T_s = 2^{SF}/BW \tag{1}$$

where SF and BW are the selected spreading factor and bandwidth, respectively. LoRa data rates range from 0.3 kbps to 50 kbps, depending on the selected communication settings. The Data Rate (DR) can be calculated as follows:

$$DR = SF * (BW/2^{SF}) * CR \tag{2}$$

where SF, BW and CR are the Spreading Factor, Bandwidth and Coding Rate, respectively.

III. The RPL Routing Protocol

The IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) is defined in RFC 6550 [3]. RPL is the protocol in charge of finding multi-hop routes to reach every destination within a LLN. Figure 1 shows the typical protocol stack for Low-Power and Lossy Networks (LLNs) using RPL and 6LoWPAN. It depicts the typical use of RPL on top of IEEE 802.15.4 MAC and PHY layers.

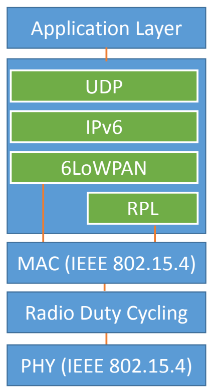

Figure 1. Typical protocol stack for a multihop LLN.

RPL is designed to meet the requirements of LLNs. LLNs feature nodes with limited memory and processing capabilities, usually battery-powered and connected by lossy links with low data rates. RPL creates a Destination Oriented Directed Acyclic Graph (DODAG). In a DODAG, nodes are organized in a hierarchical way, forming a configuration that converges at the root node. RPL provides routing paths in both directions: from the nodes to the root (upwards) and from the root to the nodes (downwards). To create the DODAG, RPL uses Internet Control Message Protocol version 6 (ICMPv6) control messages. The three basic types of RPL control messages are:

- DODAG Information Object (DIO): It contains the information that allows a node to discover a RPL Instance, learns its configuration parameters, selects a DODAG parent set, builds and maintains upward routes.
- DODAG Information Solicitation (DIS): This type of message may be used to solicit a DIO from a RPL node.
- Destination Advertisement Object (DAO): These messages are necessary in order to build and maintain downward routes.

Depending on its functionality, a node can play different roles in a RPL network:

- Root: node that triggers the construction of the DODAG, by sending the first DIO message announcing its availability as a parent. All upward routes in the DODAG are oriented towards the root node.
- Router: a node that is downward-routable, because it sends DAO messages to announce its routes. It is a potential parent for other nodes since it also sends DIO messages to announce its routing availability.
- Leaf: a node that cannot be chosen as a parent by other nodes. Leaf nodes do not transmit DIOs to announce themselves as a potential parent. They may, however, issue DAO and DIS messages. Leaf nodes always have infinite rank (0xFFFF).

A. RPL routing metrics

The RPL routing protocol is basically a generic Distance Vector protocol, that leaves the process of route selection to an external mechanism called Objective Function (OF). An Objective Function defines how nodes select parents and how they translate one or more metrics and constraints into a value called Rank. The Rank indicates how close to the root a node is: the higher the rank, the further away the node is from the DODAG root. Each DODAG must use a common OF.

Only two OFs have been standardized in two RFCs. RFC 6552 [7] defines the Objective Function Zero (OF0), which minimizes hop count to sink. RFC 6719 [8] defines the Minimum Rank with Hysteresis Objective Function (MRHOF). MRHOF is used to minimize a certain metric (e.g. latency or Expected Number of Retransmissions), while using hysteresis to prevent route changes caused by small metric changes. According to [8], each node obtains its path cost by summing up the selected link cost to the path cost advertised by the parent.

In most studies, MRHOF with the Expected Number of Retransmissions (ETX) as a routing metric is used. ETX is the average number of retransmissions of a packet at the CSMA level. In every node, the ETX value is computed for each neighbor as a moving average. By minimizing ETX, power consumption and packet end-to-end latency are implicitly minimized by reducing the number of retransmissions necessary to transmit a data packet successfully.

IV. Context

To the best of our knowledge, and since it is a relatively novel technology, there have been few research publications on the multihop LoRa topic. The studies published so far mainly focus on range studies and analyse the limits of LoRaWAN. For instance, the authors in [9] [10] evaluate the range of LoRa radio communications, in different scenarios. Furthermore, the authors in [9] have presented a channel attenuation model. A study regarding the limits of LoRaWAN is described in [2]. The authors in [4] provide a comprehensive and detailed analysis of the LoRa modulation and present a testbed built to study the performance of LoRaWAN communications. They also raise the question of the suitability of some state of the art routing protocols (such as RPL) for LoRa.

Some LoRa-based deployments have already been rolled out. The authors in [11] present a bicycle tracking system based on LoRaWAN communications. The authors in [12] also propose a LoRa-based Sailing Monitoring System, proving that it meets the basic application requirements.

Related to multihop LoRa-based solutions, only one protocol has been proposed in the literature. This protocol for LoRa transceivers is called LoRaBlink. It supports multihop bidirectional communications, and is described in [13]. This seems an interesting first step towards extending LoRa communications with multihop mechanisms. However, the proposed solution works with a fixed Spreading Factor.

V. Multihop LoRa

Multi-hop communications offer a higher degree of redundancy and provide means to improve reliability in lossy networks. By using multi-hop communications, the geographical density of gateways can be decreased and alternative routes can be provided in case of gateway outage.

However, the RPL routing protocol has no provisions for selecting PHY layer parameters (such as LoRa's SF). Therefore, a new MAC mechanism is proposed in this paper in order to find the optimal SF per link. The selected SF is then used in the OF used by RPL to compute link cost, to be able to choose the preferred parent.

This presents some additional challenges. In LoRaWAN, messages are always exchanged between a LoRa concentrator acting as gateway and a mote. The concentrator can demodulate packets from different sub-channels (defined by frequency, SF and BW) simultaneously, making it possible for default LoRaWAN devices to send at any time, on whatever sub-channel. In our multi-hop network, however, the nodes can only listen to one sub-channel at a time. Therefore, the nodes need to agree on time slots dedicated to certain sub-channels with their neighbours in order to successfully transmit the data.

In order to enable RPL-based multihop communications, the ContikiOS was chosen. Contiki is a lightweight, open-source OS designed to operate on IoT devices. It is particularly interesting because it includes implementations for the most important and well-known communication protocols for LLNs. Specifically, it provides IPv6-based conectivity by means of its 6LoWPAN (IPv6 over Low-Power and Lossy Networks) implementation, along with an implementation of the RPL routing protocol and its standardised OFs. Additionally, Contiki also provides support for well-known (lightweight) application layer protocols (CoAP, MQTT). The implementation of all these communication protocols is developed in a platform-independent way. This approach makes it possible to run the implemented protocols in any platform supported by the ContikiOS in a straightforward manner. Therefore, the necessary support for including the LoRa PHY layer was the first step to enable standardised multihop communications based on LoRa. This was done for the LoRaMote platform, a demo platform featuring Semtech's SX1272 radio transceiver [14].

The proposed Contiki-based protocol stack is depicted in Figure 2. We make use of the μIP stack implemented in Contiki, including UDP, IPv6, 6LoWPAN and the RPL routing protocol. The standard MAC layer typically used (see Figure1) is replaced by the proposed MAC protocol, called the RPL+LoRA MAC protocol (RLMAC). Finally, the LoRa PHY layer is used to transmit the data over long distances wirelessly.

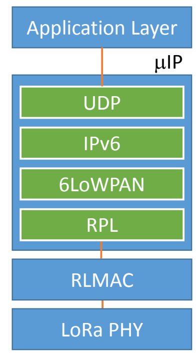

Figure 2. Proposed Contiki-based protocol stack for LoRa multihop.

The proposed RLMAC layer and the adaptations made in RPL will be presented in detail in the following sections.

VI. Proposed Solutions

As explained earlier, a multi-hop LoRa network could have a full base-station acting as RPL node but this paper focuses on the more complex scenario in which all the nodes (root, routers and leaves) have a radio that can listen to one sub-channel at the time (e.g. Semtech's SX1272/6). The first challenge of multi-hop is the discovery of the neighbours. In order to simplify the problem, we consider that all devices use the same frequency and bandwidth. The proposed mechanism aims at choosing the optimal SF.

A. Neighbour discovery and SF selection

In order to select the optimal SF per link, the process is split in two distinct phases: neighbor discovery and SF selection. Three ideas were considered for both phases at MAC layer:

- All devices initially use SF12. This is the most basic solution, however it presents several disadvantages: the messages will be sent with the slowest data rate, increasing the power consumption and the risk of collision due to the long time-on-air and the augmented transmission range. The devices will need additional control messages to agree on the optimal SF for the links they are involved in.
- Synchronized timeslots. The nodes have access to a global clock, for instance, from a GPS module included in the motes. This would allow for all of them to try

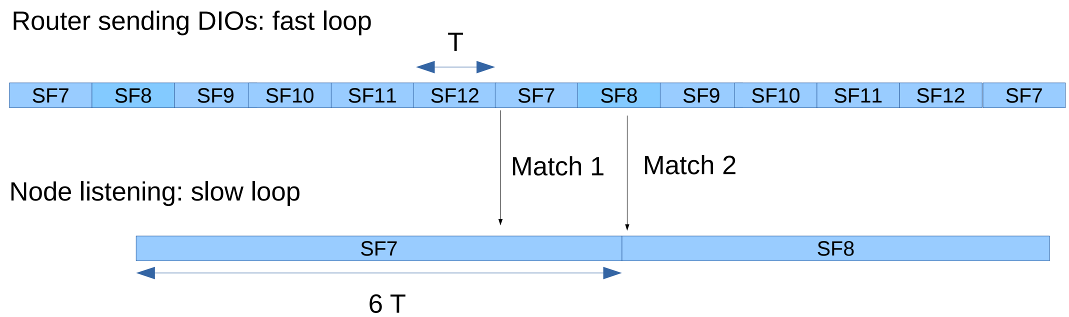

Figure 3. Timeline of the SF loop method for a router node and a node waiting to join

the different SFs in a more time-efficient way, but with a high the risk of interference.

- SF loops. The routers are configured to listen to each SF (7 to 12) during a time-slot of T seconds, the nodes that need to join the DODAG do the same with slots n_{SF} (the number of different SFs, 6 in this case) times longer, as shown in Figure 3. After $n_{SF}^2 \cdot T$, we ensure that two neighbours have met once for each SF. This method makes the discovery longer but avoids collisions. This method allows to select the optimal SF as soon as a new neighbor is discovered, provided that the fast loop of the RPL router began before the start of the slow loop of the other device.

The first method scales poorly due to collisions and requires extra control traffic. The second solution is not always suitable, since it requires a GPS signal, which may not always be available (for instance, for indoor or deep indoor applications). Therefore, the selected method is the SF loops. Even though it is slow, this mechanism allows to build a network directly with the lowest achievable SF (so highest bitrate) on each link, optimising time-on-air and saving power. It also reduces the probability of collisions with compared to the other two methods.

B. RLMAC

We propose the RLMAC protocol to allow multi-hop over LoRa devices.

Figure 4 shows the state of the devices during discovery and after having joined the DODAG. Their state depends on whether the node is a RPL root, a router or a leaf node.

RLMAC handles LoRa neighbours and delays the transmission of packets to the right SF slot of a parent or child. It includes MAC commands to manage the links. After the SF loop discovery, router nodes switch to fast loop and leaf nodes only listen to their preferred parent's SF. This change of state (from slow to fast loop or no loop) only occurs after the end of the slow loop. Doing so, the node will discover as many parents as possible by listening to all available SFs. RLMAC does not schedule messages at a precise moment, but within the right time slot (order of magnitude of tens of seconds). A Radio Duty Cycling protocol suited for LoRa could run below RLMAC (RDCs are out of the scope of this article).

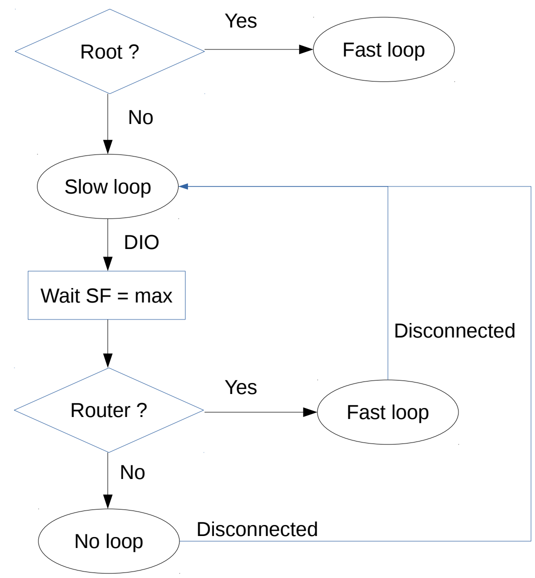

Figure 4. SF loop device state

1) Timing issues:

A device using RLMAC needs to know the SF fast loop interval (T seconds) and the list of used SFs, as well as the bandwidth, default frequency and Forward Error Correction (FEC) coding rate used. It also needs a timer (the SF timer), that starts at the same moment as the SF loops, and resetting after $n_{SF} \cdot T$ seconds. RLMAC relies on that timer and a g̀ood enough' approximation of the time offsets (see Equation 3) with respect to its neighbors to schedule transmissions.

At the reception of a valid DIO, a node adds its LoRa parent as well as its RPL parent. The parent can send the current value of its SF timer ($t_{DIO,A}$) in the MAC header of the message containing the DIO (we call that 'explicit mode') and the child can then compute the true timing offset between them. Otherwise, in the so called implicit mode, the child assumes that the DIO was sent at the beginning of the slot. It makes an approximation of the parent's SF timer

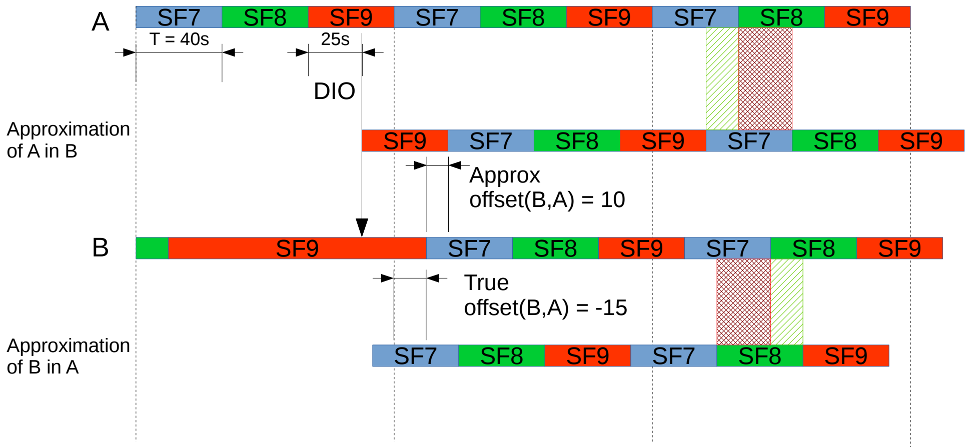

Figure 5. Time slots of two neighbor router nodes and approximations

value following the formula:

$$t_{DIO,A} \simeq (SF_{link} - SF_{min}) \cdot T + ToA(SF_{link}, L) \quad (3)$$

Where SF_{link} is the Spreading Factor used to send the DIO, SF_{min} is the lowest SF and ToA is the time on air for a DIO message of length L. Compared to the magnitude of these intervals, which is in the order of seconds, the time to process the message can be neglected.

The estimation error on $t_{DIO,A}$ in implicit mode can be as large as T, which has a very negative impact on the ability to schedule messages correctly. Figure 5 shows that a child receiving a DIO at 25 seconds into a 40 seconds time slot will use the previous parent's SF for 62.5% of the time. A node could consider only a portion of the parent's timeline as reliable. Another way of avoiding this negative effect is to forbid routers to send implicit DIOs after a certain time in a time slot.

A parent never computes its offset in respect to a child by itself. It will wait for a MAC command containing that offset, which is the opposite of the child's offset if the child is a router, and "infinite offset" (0x7FFF) if it is a leaf node. A neighbor listed with a valid SF_{link} and an infinite offset can be reached with that SF at any time from the MAC layer's perspective. For instance because it is a leaf node and only listens to the SF of the link with its preferred parent.

Figure 6 summarize the execution of the send_packet function in RLMAC. Broadcast and unicast packets to unknown addresses are sent right away, with the current SF. Unicast messages to the LoRa neighbors are sent only if that neighbor is currently listening to the right SF, otherwise the packet is queued and a callback timer will trigger its transmission after a certain delay. This introduces a MAC queue that could fill up and force the device to drop packets.

Since all the routers stay in the fast loop state, a node can always decide to send a packet at higher SF than SF_{link} to avoid waiting several seconds for getting access to the right time slot. This should not be done systematically as too many messages would be sent with the highest SF in a dense network.

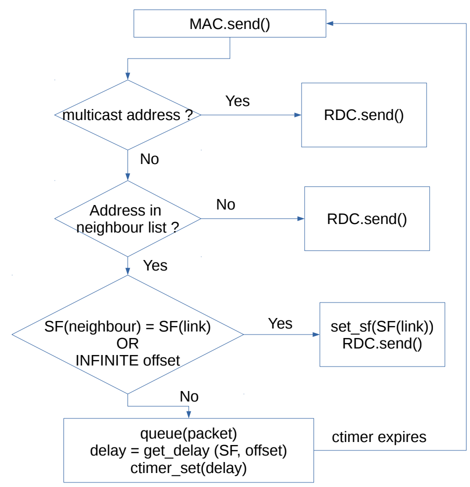

Figure 6. Execution of the RLMAC send_packet function

2) MAC commands:

The MAC commands are used to enable the timing of RLMAC and manage the physical parameters to be used by nodes, much like in LoRaWAN MAC. The header should consist of the receiver and sender link addresses, a single octet Command ID (CID) and 0 to 2 bytes of parameters, depending on the Command.

The first two commands have already been discussed: advertisement of the sender's SF timer and of the link offset. Each value is sent as a 16-bit integer (time in seconds). In addition to those, three pairs of request/ACK commands are needed to:

- Change SF_{link}.
- Change the transmission power used.
- Change the channel used by a leaf node.

A different channel can only be used to send messages to a leaf, as routers need to keep the connection to multiple neighbors, on the default frequency.

A last feature that could be included in the MAC layer is the Open Specific Window (OSW), that also needs commands for requests and ACKs. An OSW request asks to listen for different SF, channel and bandwidth than the ones defined by the loop, after a given delay. That command allows for a whole new mode of operation: the receiver sets its radio to the parameters asked by the sender, although the default mode of operation is the sender waiting for the right time in the loop.

With the current state of RLMAC, the acceptance of an OSW means losing a part of a time slot, as the radio of the router is reconfigured for different PHY parameters, and possibly missing some packets. That issue could be solved by having one or several gaps in the fast loop that are available for OSW packets, for instance a gap of 5 second at the end of each slot.

C. Objective Function for LoRa

A new Objective Function was needed to compute the nodes' ranks in our multi-hop LoRa network. RFC6551 gives a list of link and node metrics - as well as constraints - that can be used by an OF. Using the LoRa modulation, the data rate is be twenty times faster with the lowest SF than with the highest. Therefor, the link cost is made proportional to the duration (time on air) of one bit:

$$linkCost \propto 2^{SF_{link}-SF_{min}} \tag{4}$$

A node selects its preferred parent to minimize the path cost, i.e. $rank(parent) + linkCost$, then it computes its own rank as $rank(parent) + rankIncrease$. The rankIncrease has to be in the $[minRankIncrease, 9 \cdot minRankIncrease]$ interval, otherwise the parent is not considered valid. The implementation uses integer multiples of $minRankIncrease$ for the ranks computation. Table IV shows the values used in this paper. The ranks will always advertise a lower value than the real path cost.

Table IV
RPL RANK INCREASE AND PATH COST FOR ALL SF

SF	rank increase	link cost
7	1	1.22
8	2	2.13
9	3	3.79
10	6	6.82
11	9	12.41
12	9	22.75

If SF_{link} is modified later on through the exchange of the adequate MAC commands, the rank can be changed accordingly, as long as it respects the two rules set by RPL. These rules dictate that the rank of a node may not become less than any parent's rank, and it may not become more than its previous value plus a constant, except to become INFINITE_RANK. Note that INFINITE_RANK is the rank of all leaf nodes.

This method aims at minimizing the total time on air for a packet traveling from a node to the RPL root. Since all nodes are limited to 1% duty cycle per channel, more data can be sent in the network if the fast links (links with lower SFs) are chosen, and bottlenecks appear less often. A router close to the root may end up with too many children. Therefore, a second metric, denoted ǹode availability', could be introduced to limit the risk of every route going through the same node. That is especially true for a router that needs to send a lot of data, thus will not have a lot of opportunities to forward the children's packets.

VII. Validation

In order to validate the correct functioning of the proposed solution, a small network consisting of 4 LoRaMotes was set up in an office building on campus. One of the LoRaMotes was configured as RPL root, and the other motes were configured as RPL routers. Each node was located in a different floor of the building: the root node was located on the 4th floor, and the other nodes were located on the 2nd, 1st and ground floor respectively.

The parameters used were: SFs from 7 to 10, fast loop time slots (T) are 15 seconds long and a maximum MAC queue length of 5. The nodes were configured to send 82-bytes UDP frames containing RPL-related information (e.g. rank and preferred parent) every 30 seconds.

The network topology obtained is shown in Figure 7, with the SF of the links as selected by the discovery.

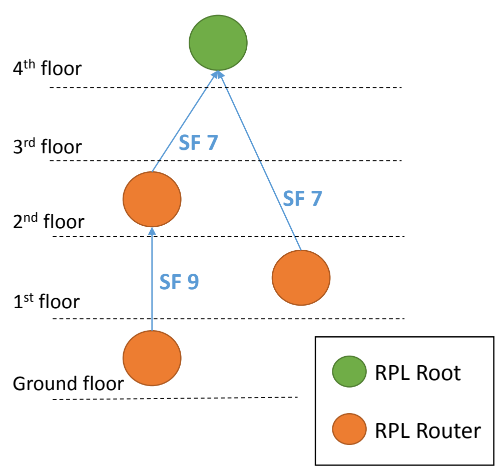

Figure 7. RPL network topology

Some unexpected behavior was found while performing this first test. As shown in Figure 7, two nodes selected SF7 to communicate with the root node. Since both nodes made this choice after having received the same broadcasted DIO message, they were always sending queued messages at the exact same time, which led to frequent collisions. This issue could be solved by adding an additional random delay for the queued packets at the MAC layer.

That feature was added using a feature of the LoRa radio itself: generating random values based on a 1-millisecond RSSI measurement. After that modification, the sink could receive packets from multiple neighbors using the same SF. The MAC layer was also modified to send packet only 70% of the time slots (the 70% at the beginning of the slot when sending to parents and those at the end of the slot when sending to children) to limit the effect of bad offset estimation described in figure 5.

In a new test including only 2 motes, the PDR was measured on a link using SF8 with the updated software: 43 packets were correctly received out of 50, yielding a 86% PDR. Part of the packet loss is due to the sender that was scheduling transmissions too close to each other. This can be fixed by checking the state of the radio before trying to send.

The performance of the protocol will get worse when more SFs and longer slots are used or messages are sent more often, as all packets are dropped once the MAC queue is full.

VIII. Conclusions

Enabling multi-hop with RPL on LoRa devices is a challenging issue. To deal with this, a new MAC protocol called RLMAC has been proposed. It features an asynchronous discovery mechanism of neighbor nodes and offers the possibility for a node with a small radio to act as a router, thanks to time slots dedicated to different SFs. This allows to select the optimal SF per link, optimizing the time on air as well as the power consumption. The protocol includes MAC commands (some directly inspired by LoRaWAN MAC) to manage the LoRa neighbors. By default, a node sending a packet does the effort of waiting for its neighbor to be in the right time slot. The proposed MAC commands, allow the sender to open a receive window with specific SF. This feature can be useful in applications in which messages need to be received with the lowest possible delay.

A new Objective Function has been proposed for RPL, in order to compute the path cost, based on the selected SF for the given link.

A small RPL network was deployed on campus, in order to validate the correct functioning of the proposed solution. Some unexpected behavior was found, and a solution adding an additional random timer is proposed.

Before considering the use of RLMAC in real life, a simulation tool would be needed to test its performance for a larger number of nodes and evaluate the power consumption of routers and leaf nodes.

Acknowledgment

This work is partially supported by the TETRA grant of the Flanders Innovation and Entrepreneurship agency (Vlaio), under the project HBC.2016.0101 - Horizontal-IoT.

References

[1] X. Xiong, K. Zheng, R. Xu, W. Xiang, and P. Chatzimisios. Low power wide area machine-to-machine networks: key techniques and prototype. *IEEE Communications Magazine*, 53(9):64–71, sep 2015.

[2] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, and J. Melia. Understanding the limits of LoRaWAN, CoRR, vol. abs/1607.08011, 2016. [Online]. Available: http://arxiv.org/abs/1607.08011.

[3] N. Sornin, M. Luis, T. Eirich, T. Kramp, and O.Hersent. LoRaWAN Specification, 2015.

[4] A. Augustin, J. Yi, T. Clausen, and W. Townsley. A Study of LoRa: Long Range & Low Power Networks for the Internet of Things. *Sensors*, 16(9):1466, sep 2016.

[5] T. Wendt, F. Volk, and E. Mackensen. A benchmark survey of long range (LoRaTM) spread-spectrum-communication at 2.45 GHz for safety applications. In *2015 IEEE 16th Annu. Wirel. Microw. Technol. Conf.*, pages 1–4. IEEE, apr 2015.

[6] Semtech. Datasheet: SX1272/73 - 860 MHz to 1020 MHz Low Power Long Range Transceiver, 2015.

[7] P. Thubert. RFC 6552 - Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL), 2012.

[8] O. Gnawali and P. Levis. RFC 6719 - The Minimum Rank with Hysteresis Objective Function, 2012.

[9] M. Aref and A. Sikora. Free space range measurements with Semtech LoRa technology. In *2014 2nd Int. Symp. Wirel. Syst. within Conf. Intell. Data Acquis. Adv. Comput. Syst.*, pages 19–23. IEEE, sep 2014.

[10] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo. On the coverage of LPWANs: range evaluation and channel attenuation model for LoRa technology. In *2015 14th Int. Conf. ITS Telecommun.*, pages 55–59. IEEE, dec 2015.

[11] D.H. Kim, J.B. Park, J. H. Shin, and J.D. Kim. Design and implementation of object tracking system based on LoRa. In *2017 International Conference on Information Networking (ICOIN)*. IEEE, jan 2017.

[12] L. Li, J. Ren, and Q. Zhu. On the application of LoRa LPWAN technology in Sailing Monitoring System. In *2017 13th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*. IEEE, feb 2017.

[13] K. Romer, K. G. Langendoen, and T. Voigt. LoRa for the Internet of Things. In *Proc. 2016 Int. Conf. Embed. Wirel. Syst. Networks*, pages 361–366, Graz, Austria, 2016. Junction Publishing.

[14] S. Thielemans, M. Bezunartea, and K. Steenhaut. Establishing transparent IPv6 communication on LoRa based Low Power Wide Area Networks (LPWANS). In *16th Annual Wireless Telecommunications Symposium*, Chicago, EEUU, apr 2017. IEEE.