

# Cross-Layer Framework and Optimization for Efficient Use of the Energy Budget of IoT Nodes

Gilles Callebaut, Geoffrey Ottoy, Liesbet Van der Perre

*KU Leuven, ESAT-DRAMCO, Ghent Technology Campus*

Ghent, Belgium

gilles.callebaut@kuleuven.be, geoffrey.ottoy@kuleuven.be, liesbet.vanderperre@kuleuven.be

**Abstract**—Both physical and MAC-layer parameters impact the autonomy of IoT devices. We present an open-source cross-layer assessment framework for Low Power Wide Area networks (LPWANs) in this paper. It extends the state-of-the-art with energy models, downlink messages, and adaptive datarate features. Hence, hypotheses and transmission schemes can be tested and evaluated. As a representative case, the LoRaWAN protocol is assessed. The findings demonstrate that a cross-layer is imperative to effectively realize LPWANs in terms of energy efficiency and throughput. For instance, up to a factor of three reduction in energy consumption can be achieved by transmitting longer packet on quasi-static channels. Yet, under adverse dynamic conditions, an energy penalty will occur.

**Index Terms**—IoT, Energy Efficiency, LPWAN, cross-layer

## I. INTRODUCTION

There is a rapidly increasing demand to interconnect devices. Sensors and actuators cooperate, often through a cloud infrastructure, enabling new applications in smart homes, cities, and sustainable environments. Many of these Internet-of-Things (IoT) applications face strict energy constraints as they rely on battery powered devices. Typically, a wireless IoT sensor will wake up periodically to collect measurements, e.g., temperature or chemical substances. The data can be sent immediately to the cloud infrastructure or accumulated in the node to reduce communication overhead. The device is in a low-power *sleep state* for the rest of the time. Several dedicated communication systems have been developed for long range low power IoT connectivity, including LoRaWAN [1], SigFox and NB-IoT [2].

Characterizing the energy of communication is a daunting task. It depends on numerous interrelated effects and aspects. To start with, the application determines how many bytes are in a packet and at what rate they are being sent. The latter depends, among others, on the propagation conditions. Secondly, the number of devices in the network effects the energy consumption, i.e., increasing traffic eventually will cause more collisions. A third aspect is the position of the devices and the state of the communication channel, which determines the path loss and noise. Finally, the communication protocol and radio hardware determine how long the radio is active and at what power. All these effects influence each other. For example, a noisy channel with a high path loss, may cause more retransmissions, which in turn increases the chance for collisions.

Simulators have been established in order to assess and optimize the LoRaWAN protocol specifically [3]–[5]. Here, we present a cross-layer simulation framework to realistically analyze and optimize the energy consumption of IoT devices. To this end, in extension to [6], [7], both Adaptive Data Rate (ADR) and downlink messages have been included. We show that this has a significant impact on the number of collisions and the data extraction rate. Secondly, the payload size and packet rate can be changed to correspond with real-life applications, whereas this was fixed in prior work [8]. Hence, the framework allows application specific monitoring of the LoRaWAN Networks. Thirdly, the energy consumption corresponding to the IoT nodes are based on measurements from a power-optimized node [9]. Many nodes can be positioned relatively to the gateway and subjected to different signal loss and noise. The source code (Python) of the simulator is publicly available [10]. We welcome researchers to tailor the simulator to their own needs.

The next section will briefly summarize the features of LoRa and LoRaWAN with an emphasis on parameters affecting the power consumption. Section III highlights the key components and operation of the simulator. In Sect. IV we show the simulations and discuss the results of a case study. Conclusions and future work are presented in Sect. V.

## II. LPWAN ASSESSMENT - LORA AND LORAWAN

In our investigation, we consider the LoRa PHY and LoRaWAN MAC schemes as a first representative case. We introduce both layers with a focus on the governable parameters that affect energy consumption. Notably, a number of parameters can be adjusted in the application layer as well.

1) *LoRa PHY scheme*: LoRa, short for Long Range, is a proprietary modulation technique developed by Cycleo; later acquired by Semtech. The modulation technique is based on Chirp Spread Spectrum (CSS), which is similar to Direct-Sequence Spread Spectrum (DSSS). LoRa encodes information by means of chirps, in contrast to modulation with pseudorandom binary sequences in DSSS. A chirp is a sinusoidal signal whose frequency monotonically increases (*upchirp*) or decreases (*downchirp*). The symbol duration is based on the spreading factor ( $SF$ ) and the bandwidth ( $BW$ ). Each LoRa symbol is composed of  $2^{SF}$  chirps each covering the entire

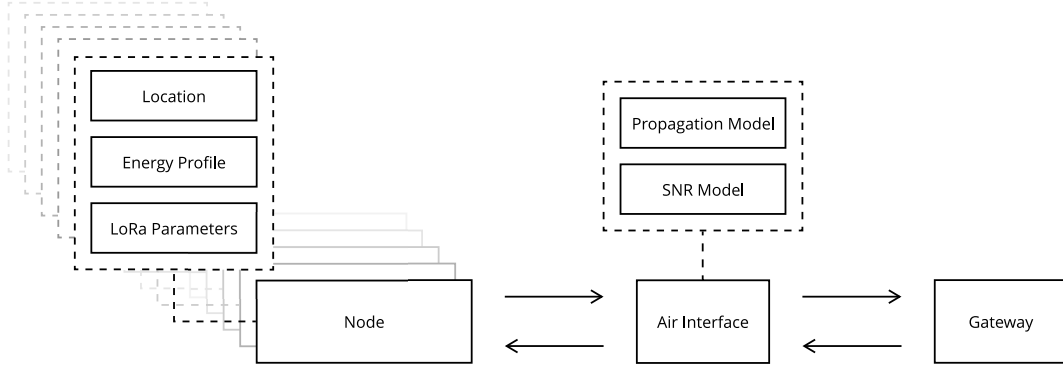


Fig. 1: LPWAN Simulation Framework based on a modular design for cross-layer assessment and optimization.

bandwidth. The symbol duration of a LoRa symbol is defined as:

$$T_{sym} = \frac{2^{SF}}{BW} \quad (1)$$

A LoRa message consists of a preamble and data. The preamble contains only upchirps while the data part comprises upchirps with discontinuities. The position of the discontinuities—in frequency—is what encodes the transmitted information. To ensure that multiple packets can be demodulated concurrently, LoRa packets can be encoded with different orthogonal spreading factors. This yields a robust and long-range communication link for IoT devices.

2) *LoRaWAN MAC scheme*: On top of the physical LoRa layer, the LoRaWAN defines the multiple access control (MAC) layer and the network architecture. Opposed to the proprietary modulation technique LoRa, LoRaWAN is an open standard specified by the LoRa Alliance. LoRaWAN defines three device classes each targeting different use cases. In general, LoRa devices initiate communication by means of transmitting a message to the gateway. By means of *confirmed messages*, the nodes can request acknowledgments to ensure that the packets are successfully received by the gateway. After an uplink message, the node opens two slots to receive downlink traffic from the gateway. This communication scheme is optimized for low power because of its uplink-centric design. LoRaWAN mandates that each LoRa device implements this scheme. The compliant devices are called class A devices. Class B and C devices extend the communication capabilities of class A devices by defining additional receive slots. Class B devices have periodic receive slots while class C devices continuously listen for incoming messages. These additional downlink receive slots reduce the downlink latency yet yield a higher power consumption.

3) *Governable parameters*: LoRaWAN facilitates controlling the airtime (Eq.1), data rate and energy consumption of LoRa nodes in order to optimize the overall energy consumption of the network. This is done by adapting the data rate and

transmission power to the propagation characteristics of the LoRa link. Increasing the spreading factor results in a higher airtime, which allows the receiver to better demodulate the message. Despite the better range, a node will consume more power when transmitting with a higher spreading factor. In addition to modifying the spreading factor, the transmission power can be altered to further increase the range or decrease the energy consumption.

LoRaWAN devices need to comply with the regulations imposed in the industrial, scientific and medical (ISM) radio bands in which they operate. These regulations include a limitation in the duty cycle of transmissions and excited transmit power. Concretely, LoRaWAN enforces a per band duty-cycle limitation. After transmitting a message, the node needs to wait  $T_{off}$  seconds before transmitting again in that band as per Eq. 2. Considering the case<sup>1</sup> of sending a message with a payload size of 51 bytes and a spreading factor of 12 and respecting a duty cycle limit of 1%, the time off is 4 minutes.

$$T_{off} = \frac{T_{air}}{T_{dc}} - T_{air} \quad [s] \quad (2)$$

### III. CROSS-LAYER ASSESSMENT FRAMEWORK

The presented cross-layer simulator (Fig. 1) provides a generic framework to evaluate and co-optimize PHY, MAC and network parameters under realistic conditions. To accomplish a use-case agnostic modular platform, the simulator is structured on the basis of individual components. In the framework, each class A *Node* sends LoRa packets to the *Air Interface* where collision, propagation, and SNR models operate on the messages in progress. Finally, the *Gateway* receives and processes the packets. In the case of confirmed messages, the gateway will transmit a message in the downlink to the corresponding node to acknowledge the received uplink message.

<sup>1</sup>This is a worst-case scenario, where the airtime of one packet is maximized.

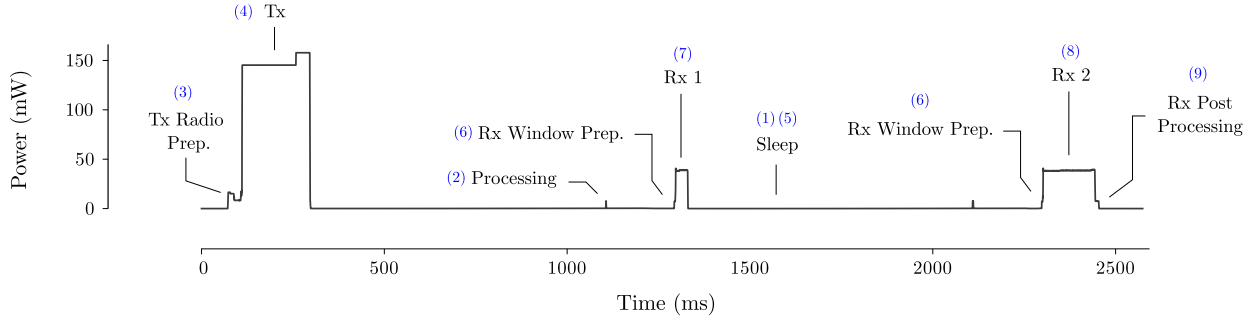


Fig. 2: Measured power states of a LoRa node [9]. These measurements are also summarized in Table I. This profile clearly shows the energy impact of transmitting a message. In this case a confirmed message was sent with SF9 and a payload of 32 bytes.

#### A. Nodes

Each node is characterized by an energy profile, a set of LoRa parameters and a location. The default energy profile used in the simulator is based on the energy consumption of [9]. We opted for this profile due to its power efficiency. Still, the simulator is not constrained to one energy profile. Different distinct profiles can be allocated to nodes, mimicking various nodes. The different energy states of the simulated node are summarized in Table I and illustrated in Figure 2. A similar experiment has been conducted in [11]. Notably, there are significant differences between the energy profile as measured on our LoRaWAN-enabled node and those reported on in [11]. For instance, the node –evaluated in [11]– consumes twice the power of the node in our experiments [9] in transmit mode.

As can be observed from Figure 2 and Table I, other power states are taken into account besides transmit, receive and sleep. First, simple processing (*state 2*) is simulated. Secondly, the states prior to transmitting and receiving (*state 6*) are related to waking-up and setting up the radio. Finally, after receiving a downlink message, the downlink message is processed and the MAC-related functionality is executed (*state 9*).

The behavior of the node is designed as specified by Semtech [12]–[14] and their LoRaWAN node implementation.<sup>2</sup> In spite of the openness of the LoRaWAN MAC protocol, not all LoRa-specific documents are publicly accessible. In addition, the network operator can, to some extent, freely define the network’s behavior. In our assessment, this functionality is based on the open-source implementation of The Things Network.<sup>3</sup>

In order to optimize the energy budget of LoRa class A nodes, a downlink message can only be received after transmitting an uplink message. Hence, the LoRa nodes only need to listen to incoming messages at specific times. As previously mentioned, Class A LoRaWAN devices [1] utilize two receive windows. The data rate and center frequency of the downlink messages depends on the used receive window,

TABLE I: Energy profile [9] used in the case.

State No.	State Description	Power (mW)	Duration (ms)
1	Sleep	5.7e-3	-
2	Processing	15	5
3	Tx prep.	12.5	40
4	Tx	Tab. II	Eq. 1
5a	Wait Rx 1	5.7e-3	1000
5b	Wait Rx 2	5.7e-3	1000 - len(state 7)
6	Rx prep.	8.25	3.4
7	Rx1	36.96	airtime(DR=DR_tx)
8	Rx2	34.65	airtime(DR=3)
9	Rx post proc.	8.3	10.7

the data rate and center frequency of the uplink message. By default, the downlink message scheduled for the first receive window (RX 1) uses the same frequency and data rate as the uplink message. In the second receive window (RX 2), a fixed predefined frequency and data rate are being used. In the remainder of this paper, the receive windows will be denoted as *RX 1* and *RX 2*. In the case a downlink message was received in RX 1, the node will not use the second receive slot.

In RX 2, Semtech defines a spreading factor of 12 while in our assessment we prefer the SF9 as proposed by The Things Network. As a lower spreading factor is favored because the base station can transmitted with higher power. The lower spreading factor results in a faster reception, which in turn yields a lower energy consumption at the node. A channel frequency of 868.525 MHz was selected for RX 2; conform to The Things Network.

The channel frequency of the uplink packets are selected on basis of the channel availability. The end-device chooses a channel with the lowest  $T_{\text{off}}$  (Eq. 2). The device respects the duty cycle regulations and waits to transmit a message if the required  $T_{\text{off}}$  is not satisfied. A default transmission rate ( $\lambda$ ) of 0.02 bits per second is chosen which is equivalent to transmitting a 9 byte message every hour.

<sup>2</sup><http://stackforce.github.io/LoRaMac-doc/>

<sup>3</sup><https://github.com/thethingsnetwork>

TABLE II: Measured transmit power [9] for the defined finite transmit power states.

Transmit Power (dBm)	2	5	8	11	14
Power (mW)	91.8	95.9	101.6	120.8	146.5

### B. Air Interface

The air interface includes three main components. First, the propagation channel introduces a path loss. Secondly, a simple SNR model is provided to translate the Received Signal Strength (RSS) to an SNR value. Finally, a collision model determines the collided packets, which occurs particularly in the uplink in a typical LPWAN case.

1) *Propagation Model*: Currently, the framework features two channel models. First, a log-distance channel model with shadowing is provided, where the path loss is characterized by:

$$PL(d) = PL(d_0) + 10 \cdot n \log \frac{d}{d_0} + X_\sigma \text{ [dB]} \quad (3)$$

By default, the following parameters [15] are used:

$$\begin{aligned} d_0 &= 1000 \text{ m} \\ PL(d_0) &= 128.95 \text{ dB} \\ X_\sigma &= 7.8 \text{ dB} \\ n &= 2.32 \end{aligned} \quad (4)$$

An additional path loss can be included in the log-distance model to simulate indoor positioned nodes and gateways to accommodate for the additional path loss [16] due to the penetration of a building. Secondly, a COST 231 model [17] implementation can be used to model specific scenarios.

2) *SNR Model*: The current version of the simulator takes into account the noise floor, as described in [12]. In future extensions more complex models can be included and interference could be added.

3) *Collision Model*: The collision model considers the center frequency, spreading factor, timing and power to determine whether packets collide. The model is based on the findings reported in [6]. Due to the orthogonality of the specified spreading factors, two messages encoded with different spreading factors can be demodulated concurrently without colliding.

### C. Gateway

The gateway model is mainly based on the popular RF solution iC880A [18]. This LoRa concentrator is able to receive up to eight packets simultaneously sent with different spreading factors on different channels. This restriction is not considered in the assessment in this paper. A message can be received by the gateway if it has not collided and the signal strength is higher than the sensitivity of the gateway [18], [19]. After demodulating the received message, the network executes Adaptive Data Rate (ADR) –if enabled– following a mechanism inspired by the implementation of The Things Network.<sup>4</sup> According to the ADR specification, the network is

capable of increasing the data rate and changing the transmit power of the node, while the nodes can only decrease their data rate. This can result in a low power transmit trap where nodes are no longer capable of communicating with the gateway [4].

Depending on the MAC LoRaWAN parameters of the uplink message, the gateway responds with a downlink message. We currently assume that every scheduled downlink message will be received by the end-device considering gateways have a higher permitted transmit power. The gateway will first try to schedule a message in the receive slot which requires less energy. For instance, if a message with SF12 was sent, the gateway will try to schedule a downlink message on the second receive slot with SF9 opposed to the first receive slot with SF12, in order to save significant air time, and hence, energy. We measured an energy gain of four when utilizing this approach compared to using the first receive slot. This is one of the cross-layer energy optimizations already implemented in present networks.

## IV. ENERGY ASSESSMENT AND OPTIMIZATION: RESULTS AND DISCUSSION

The impact of LPWAN parameters on energy consumption and performance was assessed by performing experiments with the framework. Consequently, we analyzed the effect of the packet length on different performance parameters. Particularly, the impact of the new capabilities in our framework were assessed, most prominently the options to perform ADR and use confirmed messages. To adequately evaluate the network, each experiment has been repeated 1000 times by means of **Monte-Carlo simulations**. The default parameters – for the conducted experiments – are displayed in Table III. The performance of the network and individual nodes have been evaluated based on the **data extraction rate (DER)**, the **energy per payload byte** and the **channel variance**.

The data extraction rate defines the average ratio of the number of uniquely received packets on the base station to the uniquely transmitted packets per node. It indicates how reliable the intended payload bytes are received by the gateway. It differs from the packet delivery success ratio because it does not include re-transmissions in the calculation of the unique transmitted packets. Hence, the DER can be improved by utilizing re-transmissions in order to accommodate for packet loss.

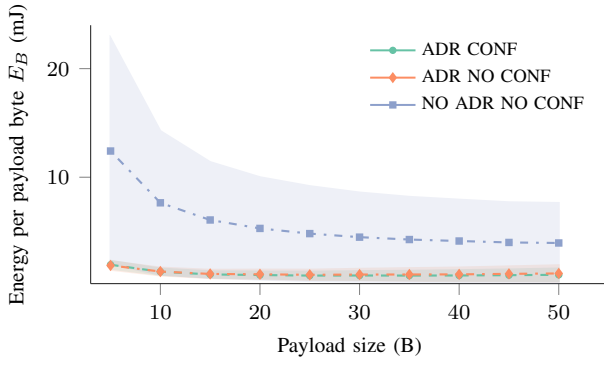
$$DER = \frac{\text{Number of uniquely received bytes}}{\text{Number of uniquely transmitted bytes}} \quad (5)$$

We prefer this parameter over the packet delivery success ratio because the main objective is that the intended payload is received by the gateway. The number of re-transmissions necessary to achieve this goal is of secondary importance.

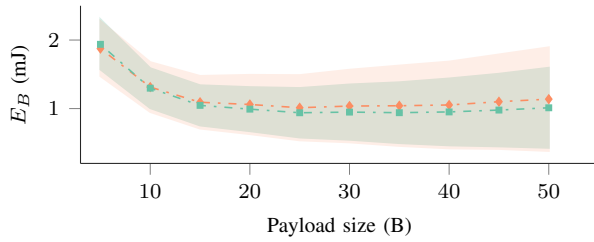
In the experiments, the energy per payload byte indicates the impact of the set of defined parameters on energy efficiency.

The channel is characterized by its variance  $\sigma$  (in dB) according to Eq. 3. This variance with respect to the average path loss is a consequence of varying propagation characteristics in both space and time. In the following experiments, the defaults of Eq. 4 are used; if not stated otherwise.

<sup>4</sup><https://www.thethingsnetwork.org/docs/lorawan/adr.html>



(a) Energy consumption per transmitted payload byte for different configurations. Disabling adaptive data rate may result in up to an order of magnitude higher average energy consumption and considerably increase its spread.



(b) Zoom in showing the further energy reduction by disabling confirmed messages, especially reducing the spread for larger packet sizes.

Fig. 3: Energy per transmitted payload byte as a function of payload size. The average is depicted by the markers while the deviation is illustrated by the shaded area. The experiment simulated 30 operation days.

#### A. Validation of the Simulation Model

The cross-layer simulation framework has been evaluated and validated by checking the results with findings from related work. As advised by Semtech and The Things Network, the **ADR rate should only be enabled when a node has a fixed location**. This is confirmed by our simulations. If the channel is dynamic, the effect of ADR is nullified and even reduces the data extraction rate (Fig. 4), i.e. fewer packets are successfully received by the gateway. Furthermore, the impact of the duty

cycle limit on the downlink capabilities of the gateway has been assessed as well. The experiments confirm the findings reported in [7]. If only the default channels (Table III) are utilized, the **gateway is incapable of acknowledging all confirmed messages**. Consequently, the number of retransmitted packages increases, which in its turn **yields a lower DER**, as also observed in [7]. Hence, the scalability of the network is mainly constrained by (I) employing confirmed messages and (II) the duty cycle limitation.

#### B. Results - Cross-Layer Approach to the Rescue

We have assessed the impact of **package length as a first cross-layer optimization opportunity**. As expected, the average energy consumption per payload byte decreases when sending larger packets (Fig. 3). To save energy, **non-time-critical data can be accumulated**, because by increasing the payload size

- 1) the overhead related to header information decreases,
- 2) the overhead of starting and initializing a transmission lowers,
- 3) the number of retransmissions in a stable propagation environment reduces,
- 4) the number of downlink receive windows is also lower.

**Substantial energy savings, up to an order of magnitude, can be achieved by enabling ADR** as indicated in Fig. 3a. This obviously demonstrates the importance of including ADR in the assessment and optimization of transmission parameters in LPWANs to ensure long battery lifetime of IoT nodes.

Despite the aforementioned beneficial effects of increasing the payload size, sending more bytes per packet increases the total number of bytes which are sent sub-optimal. Only after receiving 20 uplink messages, the network will respond with the adequate ADR parameters to accommodate for non-optimal propagation matched LoRa parameters. **For higher payload sizes this implies that more bytes have been sent before the LoRa parameters are adjusted to the channel**. In addition, ADR changes the parameters in steps yielding an even slower adaption to the propagation environment for larger payload sizes. This effect is clearly notable when observing the energy consumption over a short time period or when nodes have a slow data transmission rate. The phenomenon **results in a higher energy spread** as depicted in Figure 3b. In quasi-static situations the impact will become negligible on the longer term. In dynamic situations, however, the trade-off on packet length may yield a different result.

**To faster adapt to the channel, LoRa devices could first sent 20 smaller packets**. This will result in reduced airtime and energy for packets which are sent with non-optimal parameters. The further in-depth investigation of packet length versus dynamics in the channel can be performed conveniently in the presented framework.

#### V. CONCLUSIONS AND FUTURE WORK

A modular cross-layer framework for LPWANs has been presented. It allows assessing energy and reliability. Furthermore, diverse scenarios can be analyzed based on detailed

TABLE III: Parameters for the experiments reported on.

Parameter	Default Value
Channel Variance $\sigma$	7.8 dB
Number of Nodes	100
Data Transmission rate $\lambda$	0.02 bps
Initial Transmit Power	14 dBm
Channels	868.1, 868.3 and 868.5 MHz
RX2 Channel	868.525 MHz
RX2 Data Rate	DR3 (SF 9)
Cell Radius	1000 m



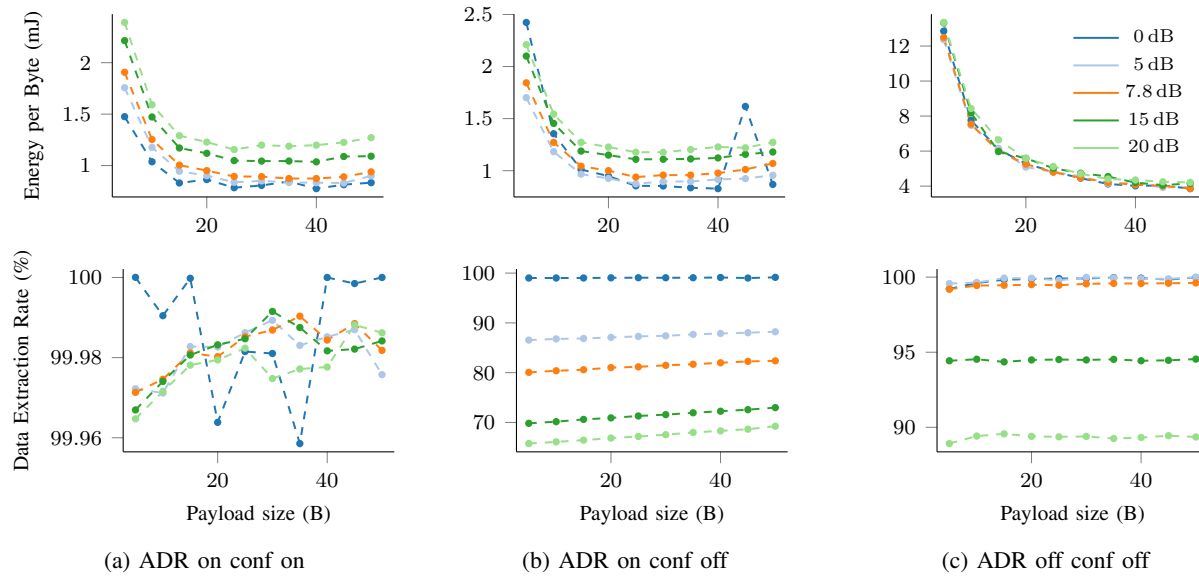


Fig. 4: Impact of the channel variance and payload size on the energy and data extraction rate.

energy profiles of IoT nodes. The framework has been extended with downlink messages, Adaptive Data Rate and fine-grained monitoring of various parameters (e.g., energy, collided packets). It is of particular interest to study the impact of scaling up to large numbers of nodes in a network in both quasi-static and dynamic scenarios. Our results show that many parameters impact the energy on the link, and they do influence each other. This has been illustrated for example for the packet length and ADR parameters. **The importance of a cross-layer approach is evidenced by a first specific assessment on packet length and payload size.** The proposed cross-layer approaches will be validated by conducting real experiments. We see many interesting cross-layer opportunities to further improve energy efficiency and reliability for massive Machine-Type Communication (mMTC). These include, among others, optimizing packet length taking into account channel dynamics.

## REFERENCES

- [1] N. SORNIN and A. YEGIN, *LoRaWAN™ Specification*, LoRa Alliance Technical Committee Std., 2017, v1.1.
- [2] Y. P. E. Wang, X. Lin, A. Adhikary, A. Grovlen, Y. Sui, Y. Blankenship, J. Bergman, and H. S. Razaghi, “A Primer on 3GPP Narrowband Internet of Things,” *Comm. Mag.*, vol. 55, no. 3, pp. 117–123, Mar. 2017. [Online]. Available: <https://doi.org/10.1109/MCOM.2017.1600510CM>
- [3] M. Bor and U. Roedig, “LoRa transmission parameter selection,” in *Proceedings of the 13th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Ottawa, ON, Canada, 2017, pp. 5–7.
- [4] M. Slabicki, G. Premsankar, and M. Di Francesco, “Adaptive Configuration of LoRa Networks for Dense IoT Deployments,” 2018.
- [5] B. Reynders, Q. Wang, P. Tuset-Peiro, X. Vilajosana, and S. Pollin, “Improving Reliability and Scalability of LoRaWANs Through Lightweight Scheduling,” *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2018.
- [6] 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*. ACM, 2016, pp. 59–67.
- [7] A.-I. Pop, U. Raza, P. Kulkarni, and M. Sooriyabandara, “Does bidirectional traffic do more harm than good in LoRaWAN based LPWA networks?” *arXiv preprint arXiv:1704.04174*, 2017.
- [8] J. Haxhibeqiri, F. Van den Abeele, I. Moerman, and J. Hoebeke, “LoRa scalability: A simulation model based on interference measurements,” *Sensors*, vol. 17, no. 6, p. 1193, 2017.
- [9] G. Ottoy, G. Leenders, and G. Callebaut, “LoRaWAN EFM32,” doi: 10.5281/zenodo.1209414. [Online]. Available: [https://github.com/DRAMCO/LoRaWAN\\_EFM32](https://github.com/DRAMCO/LoRaWAN_EFM32)
- [10] G. Callebaut, “LoRaWAN Network Simulator,” doi: 10.5281/zenodo.1217124. [Online]. Available: <https://github.com/GillesC/LoRaEnergySim/tree/v0.1.0>
- [11] L. Casals, B. Mir, R. Vidal, and C. Gomez, “Modeling the Energy Performance of LoRaWAN,” *Sensors*, vol. 17, no. 10, p. 2364, 2017.
- [12] *LoRa™ Modulation Basics*, Semtech Std., 2015, aN1200.22. [Online]. Available: <https://www.semtech.com/uploads/documents/aN1200.22.pdf>
- [13] *SX1272/3/6/7/8: LoRa Designer's Guide*, Semtech Std., 2013, aN1200.13. [Online]. Available: [https://www.semtech.com/uploads/documents/LoraDesignGuide\\_STD.pdf](https://www.semtech.com/uploads/documents/LoraDesignGuide_STD.pdf)
- [14] *SX1272/3/6/7/8: LoRa Modem Low Energy Consumption Design*, Semtech Std., 2013, aN1200.17. [Online]. Available: [https://www.semtech.com/uploads/documents/LoraLowEnergyDesign\\_STD.pdf](https://www.semtech.com/uploads/documents/LoraLowEnergyDesign_STD.pdf)
- [15] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pet-tissalo, “On the coverage of LPWANs: range evaluation and channel attenuation model for LoRa technology,” in *ITS Telecommunications (ITST), 2015 14th International Conference on*. IEEE, 2015, pp. 55–59.
- [16] *Compilation of measurement data relating to building entry loss*, International Telecommunication Union Std., 2015, report ITU-R P.2346-0. [Online]. Available: [https://www.itu.int/dms\\_pub/itu-r/opb/rep/R-REP-P.2346-2015-PDF-E.pdf](https://www.itu.int/dms_pub/itu-r/opb/rep/R-REP-P.2346-2015-PDF-E.pdf)
- [17] E. Damosso, *Digital mobile radio towards future generation systems: COST action 231*. European Commission, 1999.
- [18] *WiMOD iC880A datasheet*, IMST Std., 2015. [Online]. Available: [https://wireless-solutions.de/downloads/Radio-Modules/iC880A/iC880A\\_Datasheet\\_V0\\_50.pdf](https://wireless-solutions.de/downloads/Radio-Modules/iC880A/iC880A_Datasheet_V0_50.pdf)
- [19] *SX1301 Datasheet*, Semtech, 2017, v2.3.