Battery Life Optimization in LoRa Networks using Spreading Factor Reallocation

Ian H. de Andrade[a], Luís Henrique M. K. Costa[a,*], Rodrigo S. Couto[a]

[a] *Universidade Federal do Rio de Janeiro - PEE/COPPE/GTA*

Abstract

This paper proposes a dynamic strategy for the allocation of resources used by end devices in LoRa networks, which employ chirp spread spectrum modulation. The proposed battery life optimization (BLO) strategy splits end devices into different spreading factor (SF) groups. The basic idea is to reduce the collisions between end devices using the same SF. Moreover, BLO also considers the current battery level of each end device, and periodically reallocates the SF groups to optimize the battery consumption of all nodes and extend the network lifetime. The main innovation of BLO is to consider in addition to the RSSI the air time of different SFs as a weighting factor in SF allocation. We compare BLO to state-of-the-art (SoA) SF-allocation strategies, achieving 77% improvement in successful message delivery compared to LoRaWAN's ADR scheme. Furthermore, we obtain better energy efficiency with BLO. In a scenario with one gateway and 500 devices operating over 24 hours, the remaining energy with BLO is 10 and 3.6 times larger than with EXPLoRa-SF and EXPLoRa-AT SoA strategies, respectively.

Keywords: LPWAN; green networking; LoRaWAN; chirp spread spectrum; Internet of things; radio resource allocation.

*Corresponding author.

Email addresses: `andrade@gta.ufrj.br` (Ian H. de Andrade), `luish@gta.ufrj.br` (Luís Henrique M. K. Costa), `rodrigo@gta.ufrj.br` (Rodrigo S. Couto)

1. Introduction

The Internet of Things (IoT) is a well-known concept used to refer to the interconnection of different objects on the Internet, making them capable of transmitting and/or receiving data. These data can be of the most varied types, from critical information such as the temperature of a boiler in a chemical plant, to more mundane information such as the filling level recycle bins [1].

Since its beginning, some IoT characteristics have been defined as essential for its diffusion, such as the low energy consumption of devices, so that they can last for years without having to charge. Long transmission range is also important, to make large-scale deployment feasible in remote locations or broader areas such as cities and industries [2].

Over the last decade, different wireless networks classified as "Low Power Wide Area Networks" (LPWANs) emerged to fulfill the needs of IoT solutions [3]. LPWANs, besides the aforementioned aspects, are also characterized by low data rates, small packet sizes, a large number of connected devices, and simple network topologies. Among LPWANs, LoRa (Long Range) has attracted major attention [4] [5] due to its scalability [6], low cost radio modules and the usage of an unlicensed spectrum, allowing the implementation of private networks anywhere.

The LoRa technology was developed by Semtech, which is the patent holder for the employed modulation. The LoRa Alliance[1] on the other hand is a non-profit organization composed of more than 500 member companies which actively develop the LoRaWAN open standard. LoRa employs a Chirp Spread Spectrum (CSS) type modulation [7]. In CSS, each bit of data is represented by multiple information symbols according to the chosen Spreading Factor (SF). On the other hand, LoRaWAN is a standardized bidirectional messaging protocol of the Medium Access Layer (MAC) over LoRa modulation. LoRaWAN is responsible for managing topology, channels, data rates, and MAC commands.

[1] https://lora-alliance.org/.

The LoRaWAN network defines a star-of-stars topology in which gateways relay messages between end devices and a central network server. Gateways serve as the essential link between LoRaWAN end devices (sensors, actuators, etc.) and the network server (NS). They enhance coverage, ensure redundancy, handle multiple channels and data rates, and support network management. Gateways are connected to the NS using a backhaul (e.g., cellular, Ethernet, satellite, or Wi-Fi). The NS is responsible for decrypting the packet, removing duplicates if the packet was received by more than one gateway, and forwarding the data to the corresponding applications.

LoRaWAN operates in various frequency bands depending on the region (e.g., 868 MHz in Europe, 915 MHz in the US) [8]. Within these bands, LoRaWAN uses a set of predefined channels. In addition to random transmissions across these channels, end devices use a variable Data Rate (DR) setting. The DR is determined by the bandwidth (BW) and the Spreading Factor (SF) used. SF selection allows a dynamic trade-off between communication range and the transmission's Air Time (AT), which refers to the amount of time a radio signal is actively being transmitted from a device, that's is directly related to power consumption. By using a lower SF, the transmission will have a shorter range, but the Air Time is also reduced, consuming less battery and opening up more potential space for other nodes to transmit. Additionally, the SFs are orthogonal, so that signals modulated with different spreading factors and transmitted on the same frequency channel at the same time do not interfere with each other, increasing the network capacity.

To maximize the battery life of end devices, the LoRaWAN Network Server (NS) manages the DR setting and RF output power for each end device individually through the Adaptive Data Rate (ADR) scheme [9]. The ADR is the default mechanism for optimizing the SF selection, bandwidth, and power transmission individually for each end device, to optimize device power consumption while ensuring that messages are still received at the Gateways (GWs). To determine the optimal data rate, the network needs some measurements (i.e., a few uplink messages). These measurements contain the frame counter, signal-to-noise ratio

(SNR), and the number of GWs that received each uplink message. For each of these measurements, the ADR takes the SNR of the best gateway, and computes the optimal data rate. It is important to mention that this choice is made to achieve the lowest power consumption possible and it does not take into consideration the network's health, such as the Data Extraction Rate (DER) or SF congestion (i.e., many nodes using the same SF) and the network throughput. We define the DER as in [10], i.e., the ratio of received packets by the GWs to transmitted packets from all the network EDs, over a period of time.

Lately, some studies have started to evaluate the performance of LoRaWAN networks, targeting to improve aspects such as scalability and performance limits. The focus of some works is a more efficient SF allocation scheme for the end devices [11] [12] [13]. Again, the ADR mechanism does not consider the overall performance of the network for SF allocation. Therefore, taking advantage of SF's orthogonality, it is possible to develop a mechanism to reduce interference among groups of devices which are assigned to different SFs, and therefore improve network capacity.

This paper proposes a new mechanism to allocate spreading factors periodically, called Battery Life Optimization (BLO). BLO leverages the concept of splitting end devices into SF groups, to reduce the collisions in larger groups using the same SF, improving the network performance. The major difference in the BLO as compared to the literature, is that it take into account the current battery level of the end device and to schedule a periodic reallocation of SF groups to optimize, in a fair way, the battery life of all nodes in the network. To evaluate the performance of BLO we conduct experimental measurements of LoRa transmission power consumption, differently from most work which is based only on simulations or mathematical models [14]. This study further develops a method to solve the optimization problem of allocate an appropriate spreading factor to EDs as a mixed-integer linear program (MILP). BLO is effective especially at high traffic load, when there are hundreds or thousands of nodes transmitting in an area and in applications with high packet rates (e.g., 1 packet/min), when the increased energy demands from more frequent radio module usage can significantly reduce battery life, thus requiring energy management strategies. Simulations reveal that BLO outperforms other proposals, achieving a remarkable 77% increase in successful message delivery rates when compared to LoRaWAN's Adaptive Data Rate (ADR) scheme. Furthermore, we observe significant increase in network lifetime comparing BLO with SoA strategies, with improvements of 10 and 3.6 times over EXPLoRa-SF and EXPLoRa-AT, respectively, in a network with 500 devices served by one LoRa GW.

This paper is organized as follows. Section 2 presents the main works in the current literature. The LoRaWAN protocol and other useful information, such as the Air Time concept, are described in Section 3. Section 4 presents some experimental contributions on LoRaWAN's energy consumption, using a state-of-art power monitoring equipment, designed for measuring low-power devices. Section 5 presents the BLO proposal and the related optimization problem. Section 6 shows the performance analysis, using the NS-3 simulator. Section 7 concludes the paper and proposes future work directions.

2. Related Work

Over the past years, many studies have been done to evaluate the performance of LoRaWAN in different scenarios and to understand the scalability of this technology [15]. Among these works, some of them already propose new algorithms to better allocate in a suitable way the SFs to end devices in a LoRaWAN network. They generally focus on improving DER and the network throughput.

The authors in [11] propose two algorithms, EXPLoRa-SF and EXPLoRa-AT, to allocate SFs in LoRaWAN systems to improve DER and the network throughput. The main idea is to allocate EDs transmitting at different SFs in order to not overload some and thereby reduce the number of collisions. The EXPLoRa-SF algorithm, which selects Spreading Factors (SF) based on the total number of connected devices, equally distributes SFs to end nodes, in six equal groups, based on RSSI values and thresholds. The EXPLoRa-AT algorithm, which is a more advanced heuristic, aims to ensure a fair distribution of Air Time among the nodes in the network. EXPLoRa-AT employs the ordered water-filling approach to allocate SFs and equalize the Air Time of packets transmitted by end devices in each SF group. The ordered water-filling strategy seeks a balanced distribution of the channel load, that basically try to ensure that the sum of air times in each SF group are equal. The simulation results show EXPLoRa-AT outperforms EXPLoRa-SF in terms of reliability. Furthermore, both strategies only consider the RSSI of each ED to perform the SF allocation and the energy efficiency or energy fairness are not considered.

The paper [13] proposes a reinforcement learning (RL) method called LR-RL. This method reduces packet collision rate (PCR) based on the principle of SF-channel traffic equilibrium, which indicates that SFs with higher data rate must undertake higher packet loads. The performance of LR-RL is compared with other SF allocation methods such as LR-opt-pro and LR-greedy, both proposed in the same paper, showing that it has lower PCR compared to these. However, the paper does not provide any analysis of LR-RL with other existing SF allocation methods in terms of network throughput, latency, or energy efficiency.

The work in [16] presents a scheme to optimize the packet error rate for users far from the base station, thereby improving fairness in these networks. This is achieved by improving the channel, SF and power transmission selection for each node. The paper validates the proposed algorithm by implementing it in NS-3 and comparing it to a scheme similar to ADR, efficiently controlling the power and spreading factor for each node, while avoiding near-far problems by allocating distant users to different channels. The near-far problem mentioned consists that nodes far from the base station are more willing to collisions then near nodes, causing an unfair packet error rate in the network. With this method, simulations show that the packet error rate can be decreased up to 50% for edge nodes. However, choosing a specific transmission channel goes against the LoRaWAN standard, which requires EDs to swap channels in a pseudo-random fashion for every transmission.

The proposal of AdapLoRa [17] addresses the issue of unfair energy consumption in LoRa networks due to static SF allocation, in which devices using slower SF results in to faster battery depletion and reduced network lifetime. AdapLoRa proposes a dynamic network resource allocation system that periodically adapts resource allocation considering the current estimated network lifetime, link quality variations, and network interference. It also considers the energy consumed by end devices to receive the configuration commands, ensuring that adaptation overhead is taken into account.

The paper [12] proposes two other schemes, EXPLoRa-KM and EXPLoRa-TS, based on the ordered water-filling approach that is also used in the EXPLoRa-AT. The EXPLoRa-KM uses K-means clustering to reduce the load in critical regions with a significant number of collisions by computing suitable adjustments on the SF allocation in these areas. The EXPLoRa-TS performs an equalization of the traffic load among the SF channels, taking into account that each device can transmit a variable amount of data at a different sending rate. Simulation results show that both EXPLoRa-KM and EXPLoRa-TS enhance network performance and scalability in LoRaWAN networks for heterogeneous IoT scenarios, when different traffic loads are generated around a LoRaWAN gateway.

The paper [18] proposes a networking solution called EF-LoRa to achieve fair energy consumption among end devices in LoRa networks. It formulates the energy fairness problem as an optimization problem, that seeks to allocate different network resources, including frequency channels, spreading factors, and transmission power. The proposed EF-LoRa solution improves the energy fairness of legacy LoRa networks by 177.8% according to simulation results. However, it only consider the energy efficiency and the optimization of the LoRaWAN performance is not considered.

The development of accurate network simulators for LoRaWAN systems is of immense importance, given the challenges of testing and researching on real systems. The work in [19] is presented a comprehensive survey of the available tools for simulating LoRa networks in the NS-3 network simulator. The study has highlighted the implementation, features, and limitations of each tool, providing valuable insights into the world of LoRaWAN system simulations.

In [20], the authors investigated the battery life of LoRa nodes under two extreme parameter configurations, considering 15 min interval between transmissions and a battery capacity of 2 Ah. The first setup used SF7 with a transmission power of 2 dBm, while the second utilized SF12 with a power level of 20 dBm, representing the lowest and highest energy consumption settings, respectively. Although the average battery life of LoRa nodes is typically expected to be around ten years, the study found average lifetimes of 4.60 years for the first configuration and 1.37 years for the second. These results indicate that the resource allocation (e.g. SF and transmission power) and duty cycle play a crucial role in energy consumption and node longevity.

Beltramelli et al. [21] propose the S-LoRa protocol, a Slotted ALOHA medium access method that employs out-of-band synchronization using FM-radio data system (FM-RDS) to enhance scalability and energy efficiency in dense LoRaWAN networks. They utilize a clock time and date (CT) group to transmit synchronization information to EDs every minute, which must have FM-RDS receivers to capture these messages. For a node to initiate transmission, it needs to receive two successive synchronization frames and select a random time slot. The study revealed that the gain in energy efficiency of S-LoRa over LoRaWAN becomes more significant for short transmission intervals and large payload sizes. However, for applications requiring sporadic transmission of small payloads, S-LoRa can lead to a considerable reduction in energy efficiency, consuming nearly 40% more battery compared to the standard LoRaWAN protocol.

Table 1 shows the comparison of existing studies regarding some important aspects as the network performance analysis, energy efficiency, compatibility with the LoRaWAN standard and experimental contributions. Our main contribution is a periodic SF allocation scheme that considers the current battery of each device, improving the energy consumption fairness while also carrying about the network performance.

Table 1: Summary and comparison with previous work.

Ref.	Summary	Network performance	Energy efficiency	LoRaWAN compatible	Exp. Analysis
[11]	Improve DER by SF allocation	✓		✓	
[13]	Reduce packet collision rate (PCR)	✓		✓	
[16]	Optimize the packet error rate for users far from the base station	✓			
[17]	Dynamic network resource allocation to improve energy consumption fairness		✓	✓	✓
[12]	Enhance network performance and scalability	✓		✓	
[18]	Improve energy consumption fairness		✓	✓	
[21]	Enhance scalability and energy efficiency	✓	✓		✓
BLO	Battery Life Optimization through periodic SF reallocation	✓	✓	✓	✓

3. The LoRaWAN Protocol

LoRaWAN networks uses a star-of-stars topology in which gateways relay messages between end devices and a central Network Server (NS). Then, the NS routes the packets from each device of the network to the associated Application Server allowing this information to be displayed in any application. The communication between end devices and gateways is spread out on different frequency channels and Spreading Factors (ranging from 7 to 12). The data rate (DR) is determined by the bandwidth (BW) and SF used, with the total achievable DR varying based on the regional parameters. For example, in the AU915-928 band, that's used in Brazil, the DR ranges from DR0 to DR15, as shown in Table 2. The selection of the SF is a trade-off between communication range and message duration (the time on air, which impacts the energy consumption). Additionally, the bandwidth (BW) typically remains fixed, meaning the data rate (DR) is primarily influenced by the spreading factor (SF). The SF and DR are inversely related, meaning that increasing the SF decreases the DR and vice versa. In other terms, raising the SF lowers the data transmission rate in exchange for a higher communication range. Furthermore, the SFs are orthogonal to each other, meaning that even packets transmitted on the same channel but using different SFs do not interfere with each other. This characteristic contributes to the scalability of the network.

In order to achieve a frequency diversity to make the network more robust to interferences, End devices may transmit on any channel available at any time, using any available data rate, but have to swap channels in a pseudo-random fashion for every transmission [22].

LoRaWAN employs the Adaptive Data Rate (ADR) scheme to optimize both the battery life of end devices and the overall network capacity. This feature enables the network server to dynamically adjust the SF and transmit power of end devices based on their signal strength and quality. The ADR algorithm continuously monitors the communication between the end devices and the gateway, using the received signal strength indicator (RSSI) and signal-

Table 2: AU915-928 Data rate table [8].

Data Rate	Configuration (SF / BW)	Indicative physical bit rate [bit/s]
DR0	SF12 / 125 kHz	250
DR1	SF11 / 125 kHz	440
DR2	SF10 / 125 kHz	980
DR3	SF9 / 125 kHz	1760
DR4	SF8 / 125 kHz	3125
DR5	SF7 / 125 kHz	5470
DR6	SF8 / 500 kHz	12500
DR7	RFU	-
DR8	SF12 / 500 kHz	980
DR9	SF11 / 500 kHz	1760
DR10	SF10 / 500 kHz	3900
DR11	SF9 / 500 kHz	7000
DR12	SF8 / 500 kHz	12500
DR13	SF7 / 500 kHz	21900
DR14/DR15	RFU	-

to-noise ratio (SNR) to determine the optimal data rate and transmit power for each device. Unlike analyzing the overall distribution per SF, the algorithm focuses on individual SF allocation to minimize power consumption for each ED.

Furthermore, LoRaWAN allows the configuration of end devices into three types of classes, all of them enabling bidirectional communications. Although end devices can send uplink messages at will, the class determines when the end device can receive downlink messages (i.e., packets coming from network to the ED). The class also affects the energy efficiency of a device and consequently its battery life [22]. Next, we detail each LoRaWAN class.

- **Class A:** Communication is always initiated by the device and is completely asynchronous, meaning that the device can transmit a packet whenever it wants (ALOHA Protocol). After each transmission, the device opens two reception windows (RX1 and RX2) to receive an ACK or a downlink message, as shown in Figure 1. Therefore, the device has the higher energy efficiency since it does not need to be listening to the medium access all the time [23]. Class A is the most widely used among the three classes and must be supported by all devices.

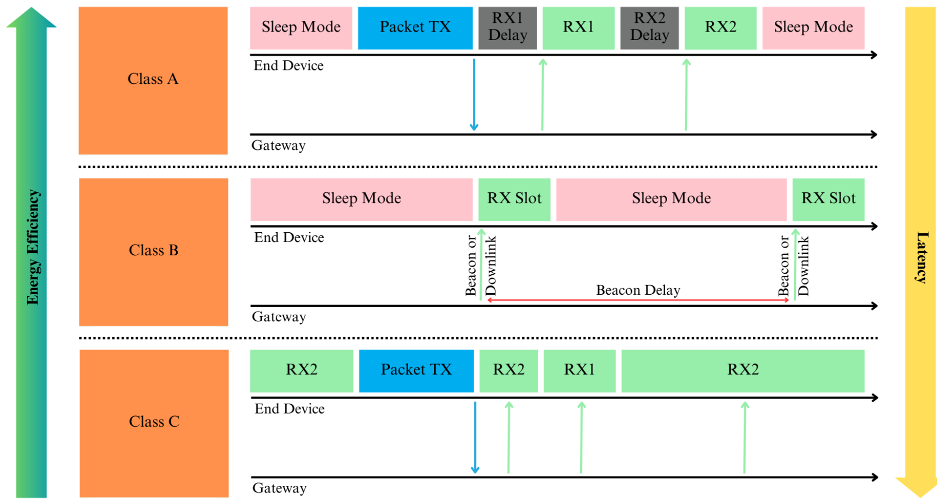


Figure 1: Temporal diagram of LoRaWAN Classes.

- **Class B:** Besides the identical functioning of class A (opening two reception windows after a transmission), it has scheduled receive windows that allow more frequent downlink communication from the network to the end devices, in comparison to Class A [23].

- **Class C:** C-Class devices keep the reception window always open when not transmitting (Half-Duplex), meaning it is possible to receive a message at any time. This mode has the highest energy consumption (≈50 mW while listening), but with the lowest latency. It is ideal for devices that have a permanent external power supply. It is possible to transition between classes A and C, and one of the use cases could be remote firmware updates (Over-The-Air Update).

3.1. MAC Commands

MAC (Medium Access Control) commands in LoRaWAN are used for various purposes such as configuring the end devices, choosing communication parameters, and requesting specific actions. These commands are essential for managing the communication parameters of the end devices and ensuring efficient and reliable operation within the LoRaWAN network. Typically, these commands are sent by the network server to the end devices but some of them are replies from the EDs to the NS.

For example, in the ADR scheme a MAC command can be used to adjust the data rate of an end device and change its transmit power. Additionally, the replies can bring some useful information like the battery percentage of each device. This information can be obtained with the NS issuing a `DevStatusReq` command to the end device which responds with a `DevStatusAns` command which contains the battery level [22].

3.2. Theoretical Calculation of Air Time

The transmission air time is critical for analyzing the energy consumption of battery-powered ED, because it represents the duration for which the radio is actively transmitting. Since transmission is one of the most power-hungry activities, accurately calculating air time allows for more efficient use of battery power by minimizing unnecessary transmission durations and idle times.

A raw LoRaWAN packet consists of a preamble containing control bits standardized by the protocol and a data information payload. Thus, the total Air Time of a LoRa packet transmission (T_{packet}) is the sum of the preamble's ($T_{preamble}$) and the payload's ($T_{payload}$) transmission times:

$$T_{packet} = T_{preamble} + T_{payload}. \tag{1}$$

To calculate each of these parcels, we first obtain the transmission parameter ratio that is given by the symbol rate (R_{sym}):

$$R_{sym} = \frac{BW}{2^{SF}}, \tag{2}$$

where BW is the selected bandwidth (125, 250 or 500 kHz) and SF is the Spreading Factor (that ranges from 7 to 12).

Using R_{sym}, the transmission time per symbol T_{sym} is calculated as:

$$T_{sym} = \frac{1}{R_{sym}}. \tag{3}$$

Thus, the parcel corresponding to the transmission time of the preamble is given by:

$$T_{preamble} = (n_{preamble} + 4.25) \times T_{sym}, \tag{4}$$

where $n_{preamble}$ is the programmed length of the preamble, which is by default equal to 8.

The number of symbols in the payload parcel (n_p) is given by:

$$n_p = 8 + \max\left(\left\lceil \frac{8 \times PL - 4 \times SF + 28 + 16 \times CRC - 20 \times IH}{4 \times (SF - 2 \times DE)} \right\rceil (CR + 4), 0\right), \tag{5}$$

where:

- PL is the number of bytes in the payload (from 1 to 255);
- SF is the Spreading Factor (7 to 12);
- $IH = 0$ when the header is enabled (default) or $IH = 1$ when the header is disabled;
- $DE = 1$ when `LowDataRateOptimize`$= 1$ (only used for $SF \geq 11$);
- CR is the coding rate (1 corresponds to 4/5 (default), 4 to 4/8).

Finally, the portion corresponding to the payload transmission time is given by:

$$T_{payload} = n_p \times T_{sym}. \tag{6}$$

Section 4 presents the experimental power consumption measures of LoRaWAN's ED, using a state- of-art power monitoring equipment. The experimental values obtained will be compared with the theoretical computation of the air time.

4. Power Consumption Experimental Evaluation

The experimental contribution of the power consumption of IoT devices in the literature are very shallow, mainly because they are very low-power and due to the difficulty in finding measurement tools that have sufficient precision for this. To measure the consumption of LoRa transmissions, a *Power Profiler Kit II* device from Nordic Semiconductor was used, which is a tool for measuring the average and dynamic power consumption in embedded low-power solutions, with a resolution that can reach up to 100 nA. The time resolution is also high enough to detect energy spikes, as seen in LoRa Transmissions. This is achieved by having a 100 ksps sampling rate of the current consumed.

In addition, to perform the LoRa transmissions, a LilyGO T-Motion development board was used, which contains a S76G System in Package (SiP) from AcSiP. The S76G integrates an STM32L073Z microcontroller, a Semtech's SX1276 equipped with the LoRa proprietary transceiver modem, and a Sony CXD5603GF GNSS with ultra-low power consumption. The SX1276 can deliver up to +20 dBm RF output, consuming from 22 mA to 125 mA during the transmission, and a high sensitivity down to -137 dBm. This SiP is ideal for tracking applications, but as we just want to evaluate the LoRaWAN transmissions consumption, the GNSS block was turned off during the experiments. Figure 2 shows the experimental setup used to perform the power consumption evaluation.

For the consumption analysis of LoRa transmissions, we use the maximum payload length for each spreading factor, as shown in Table 3. We also did measurements with the 51-byte payload (the maximum for SFs 10 to 12) for all SFs, to have a comparison with the same packet size. For each configuration we perform 10 packet transmissions and measuring the instantaneous current (in mA), and the time that the transceiver stays on during each transmission (in ms). This is the time during which the LoRa transceiver is in the transmission mode. After these measures we can compare this experimental data with theoretical air time values detailed in Section 3.2, and the theoretical consumption

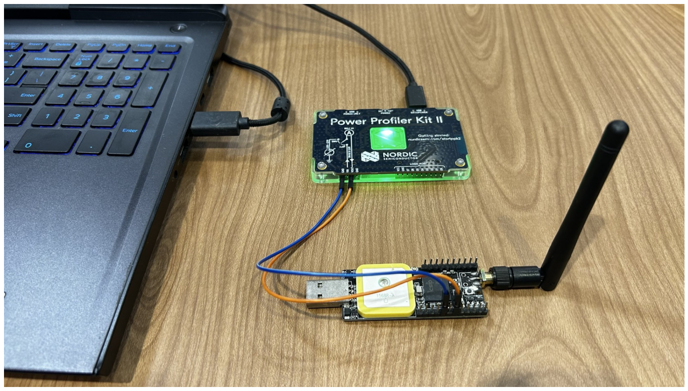


Figure 2: Experimental setup.

Table 3: Bit rate and maximum payload considering the AU915-928 and BW of 125 kHz.

DR	SF	Bit rate	Maximum Payload
0	12	250 bit/s	51 bytes
1	11	440 bit/s	51 bytes
2	10	980 bit/s	51 bytes
3	9	1760 bit/s	115 bytes
4	8	3125 bit/s	242 bytes
5	7	5470 bit/s	242 bytes

in the transmission mode using the 20 dBm RF output.

Figure 3 shows a screenshot of the nRF Connect software, the graphical tool of *Power Profiler Kit II*. We used the LilyGO T-Motion development board to perform the experiment. We setup the device to send 10 51-byte messages with 10 seconds between each other, and repeat the procedure for each SF. As this is a commercial board, it is not possible to measure the radio transmission consumption directly, but we can infer this consumption by analyzing the operation consumption cycle. In idle mode the device drains 30 mA on average, and when the radio is actively transmitting it reaches 130 mA on average, that means the amount spent on transmission mode is around 100 mA. The selected area of the plot shows the peak corresponding to the transmission current consumption, i.e. to send a 51-byte payload in SF10 the radio stayed on for 698.4 ms.

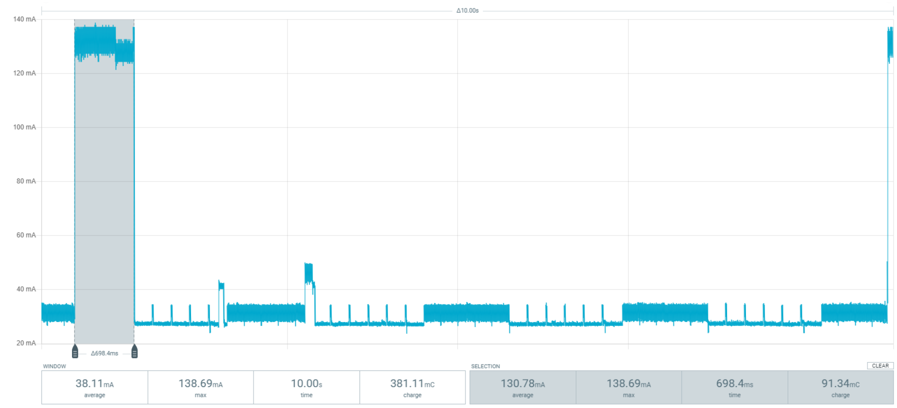


Figure 3: Consumption profile of a 51 Bytes transmission using SF10 (produced with the nRF Connect Power Profiler software.).

Table 4 show the experimental results using a 51-byte payload and all possible SFs. The time that the device stays in transmission mode is labeled as Air Time (AT) - Experimental. Comparing these values with the Theoretical AT formula presented in Section 3.2, we obtain an error rate varying from $13,3\%$ to $18,7\%$ which is considered reasonable, given an experimental setting. Yet, this difference is important once the energy spent on packet transmissions is responsible for the higher part of the operation consumption cycle. Additionally, it shows the power consumption per packet transmission and per byte for this setup. I.e one transmission using SF12 costs 77.60 µAh, almost 24 times higher than with SF7 (3.28 µAh), confirming that the SF selection is extremely important for the battery life. Figure 4 illustrates the energy cost per byte, graphically demonstrating the difference in power consumption across various spreading SFs.

Table 5 shows the experimental results using the maximum payloads for each SF, as presented in Table 3. The transmissions using higher payloads show lower error between the theoretical and experimental AT. Also, comparing Tables 4 and 5, the 51-byte transmissions consume 0.0643 µAh/byte while the 242-byte ones consume 0.0459 µAh/byte, this cost per byte reduction is due to the preamble overhead, that has more impact as less payload (useful

Table 4: Air time and power consumption for different SFs and 51-byte payload.

	Air time			**Power consumption**	
	Theoretical	Experimental	Error	(µAh/packet)	(µAh/byte)
SF7	102.65 ms	118.00 ms	14.95 %	3.28	0.0643
SF8	184.83 ms	215.56 ms	16.62 %	5.99	0.1174
SF9	328.70 ms	390.17 ms	18.70 %	10.84	0.2125
SF10	616.45 ms	698.42 ms	13.30 %	19.40	0.3804
SF11	1314.82 ms	1560.72 ms	18.70 %	43.35	0.8501
SF12	2465.79 ms	2793.75 ms	13.30 %	77.60	1.5216

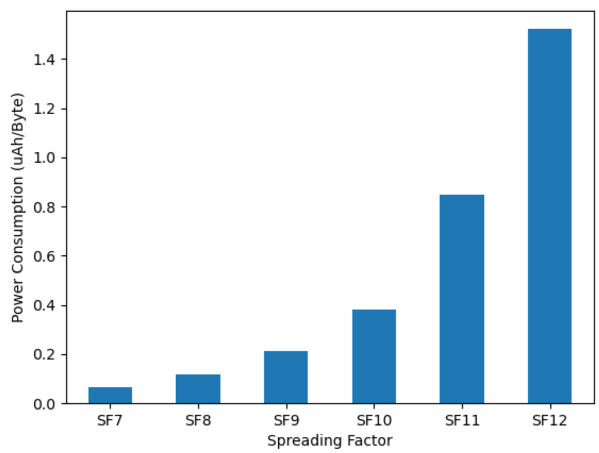


Figure 4: Power consumption as a function of the SF for 51-byte payload.

information) is transmitted, as shown in Section 3.2.

5. Battery Life Optimization

The basic idea of the Battery Life Optimization (BLO) algorithm is to periodically allocate SFs to end devices to improve the fairness in battery consumption. The fairness is achieved as we use the current battery percentage under consideration to perform the SF allocation, so that devices with less remaining energy uses the lowest possible SF, which minimizing energy consumption. We assume that all LoRa end devices are battery-powered. Moreover, they have the

Table 5: Air time and power consumption considering the maximum payload of each SF.

		Air time			Power consumption	
	Payload	Theoretical	Experimental	Error	(µAh/packet)	(µAh/byte)
SF7	242 B	379.13 ms	399.64 ms	5.41 %	11.10	0.0459
SF8	242 B	666.11 ms	707.13 ms	6.16 %	19.64	0.0812
SF9	115 B	615.42 ms	676.91 ms	9.99 %	18.80	0.1635
SF10	51 B	616.45 ms	698.42 ms	13.30 %	19.40	0.3804
SF11	51 B	1314.82 ms	1560.72 ms	18.70 %	43.45	0.8501
SF12	51 B	2465.79 ms	2793.75 ms	13.30 %	77.60	1.5216

same battery capacity. End devices may be mobile or mounted at a fixed location. The battery level of each device can be obtained with the `DevStatusReq` command presented in Section 3.1.

The selection of a specific SF value depends on the value of the current battery percentage and the Received Signal Strength Indicator (RSSI) with which the messages of each device arrive at the gateway, once the RSSI value limits the choice of SFs the device may use.

The RSSI value is affected by several parameters, including the device's power transmission, the distance between the device and the gateway, obstacles and environmental factors. An end device, according to its RSSI, has a minimum Spreading Factor that it can use to maximize the probability that the transmitted packet will be correctly demodulated at the gateway. This minimum possible SF is used as input to our optimization problem to guarantee that the ED uses this SF or a higher one.

5.1. Problem Statement

The main objective of BLO is to minimize the overall network energy consumption, in a fair way, considering the current battery of each device. The problem can be formulated as a Mixed Integer Linear Problem (MILP). The notations used in the formulation are shown in Table 6.

The first step is to verify what is the minimum SF that each device can use in order to reach the gateway, given its current RSSI. With this information, we

Table 6: Notations used in the MILP formulation of BLO.

Variable	**Meaning**
$\mathcal{N}$	Set of nodes
$\mathcal{S}$	Set of SFs
M_i	Total packets sent in each interaction
E_j	Energy cost per packet, for each SFj
P_i	Remaining Battery of node i (mAh)
β_j	Number of nodes in each SFj
w_j	Air Time weight for SFj
AT_j	Air Time for SFj (ms)
C_{ij}	Cost of node i using SFj
x_{ij}	Entry of output vector containing SFj chosen for node i

calculate the energy cost of each node i to transmit using each viable SF as:

$$C_{ij} = M_i * E_j, \tag{7}$$

where M_i is the number of packets that node i will transmit before another execution round of the algorithm and E_j is the cost per packet sent using SFj, which was obtained experimentally (Section 4). We define the execution round as the period during which the ED sends all the M_i packets before the gateway runs the BLO algorithm again. To ensure that BLO does not compromise communication reliability, we define the cost C_{ij} for selecting SFs below the minimum SF required for reliable communication, based on the RSSI/SNR thresholds, as infinity. Furthermore, if the node is so far from the gateway that it is out of range of the gateway regardless of the SF used, the cost C_{ij} for all SFs is also defined as infinity in the formulation.

Therefore, the MILP problem is formulated as follows:

$$minimize \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{S}} \frac{C_{ij} x_{ij}}{\mathcal{P}_i} \tag{8}$$

$$\sum_{j \in \mathcal{S}} x_{ij} = 1, \forall i \in \mathcal{N} \tag{9}$$

$$\sum_{i \in \mathcal{N}} x_{ij} \leq \beta_j, \forall j \in \mathcal{S} \tag{10}$$

$$C_{ij}x_{ij} \leq P_i, \forall i \in \mathcal{N}, \forall j \in \mathcal{S} \tag{11}$$

$$x_{ij} \in \{0,1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{S} \tag{12}$$

The objective function is defined in Equation 8. We define $\mathcal{P}_i$ as the amount of energy still available at node i, in mAh. The idea is to weight the cost $\mathcal{C}_{ij}$ by the amount of energy available so that nodes with more energy have lower cost. Therefore, the algorithm prioritizes smaller SFs, with lower energy cost for devices with less battery. Equation 9 forces that only one SF is allocated for each device. Equation 10 guarantees that the number of devices in each SF do not exceed the optimal distribution calculated (represented by β_j), as will be discussed next. Equation 11 ensures that the allocation does not exhaust the battery of the device until the end of the $\mathcal{M}_i$ transmissions. Finally, Equation 12 specifies the output variable x_{ij} which indicates whether node i is configured with SFj.

To define the overall allocation, BLO uses the strategy to equalize the total Air Time (AT) of packets transmitted by EDs in each SF group, in order to reduce the collisions in larger groups using the same SF, especially the lower data-rate ones which have higher AT. First, we calculate the the air time weight for each SF based on the relative proportion to the shortest air-time SF, SF7:

$$w_j = \frac{AT(SFj)}{AT(SF7)}, \qquad j = \{7, 8, 9, 10, 11, 12\}. \tag{13}$$

In order to exemplify the logic, consider a 51-byte payload. According to Table 4, $AT(SF7) = 118\,\text{ms}$ and $AT(SF8) = 215.56\,\text{ms}$. Then, we have:

$$w_8 = \frac{AT(SF8)}{AT(SF7)} = \frac{215.56}{118} \approx 1.83. \tag{14}$$

Then, we can calculate the congestion channel index vector (V_j), as proposed in [11]. The main idea is to fill each V_j with EDs using the weight vector calculated, so that all EDs are allocated to one SF and all have approximate values in the congestion channel index vector, considering the 51-byte payload. To guarantee that $V_7 \approx V_8$, as $w_7 = 1$ and $w_8 = 1.83$, as shown in Equation 14, we allocate approximately 1.83 times more devices in SF7 as compared to SF8.

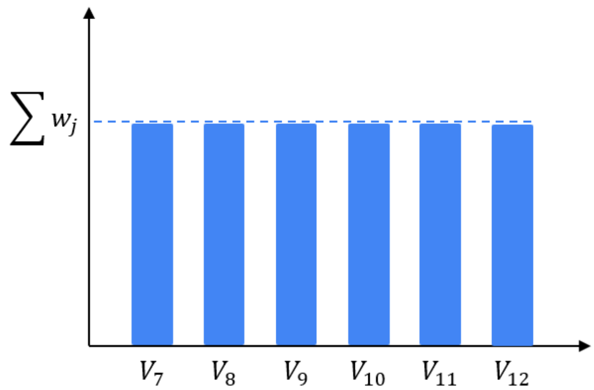


Figure 5: Water filling strategy to equalize the AT in each SF.

After this procedure, we can compute the overall allocation for each SF (β_j):

$$V_j = \beta_j * w_j \rightarrow \beta_j = \frac{V_j}{w_j}, \quad \forall j \in \{7, 8, 9, 10, 11, 12\}. \tag{15}$$

This proposed distribution of EDs across each SF aims to achieve similar channel congestion levels among the SFs groups. By doing so, we can reduce the high incidence of collisions typically expected in higher SFs, which have longer AT, leading to increased channel occupancy for each transmission. Therefore, in order to equalize the sum of AT in each SF group, the number of EDs allocated for each SF j (β_j) will be:

$$\beta_{SF7} > \beta_{SF8} > (...) > \beta_{SF11} > \beta_{SF12}. \tag{16}$$

5.2. Integrating BLO in LoRaWAN

The integration of the proposed BLO scheme in a LoRaWAN network is simple and does not require modifications to the protocol or any information which is not already available at the NS (Network Server). Furthermore, the BLO remains unaffected and fully compliant with regional regulatory requirements, and respects all current LoRaWAN specifications. The suggested integration considers the BLO Application running on the LoRaWAN NS, as shown in Figure 6. BLO uses the Signal-to-Noise Ratio and RSSI information, both measured by the LoRaWAN GW, the packet loss ratio computed by the NS, and the battery information that, as presented before, is obtained from the `DevStatusAns` reply message which is periodically sent by the device to the NS (after receiving the `DevStatusReq` message).

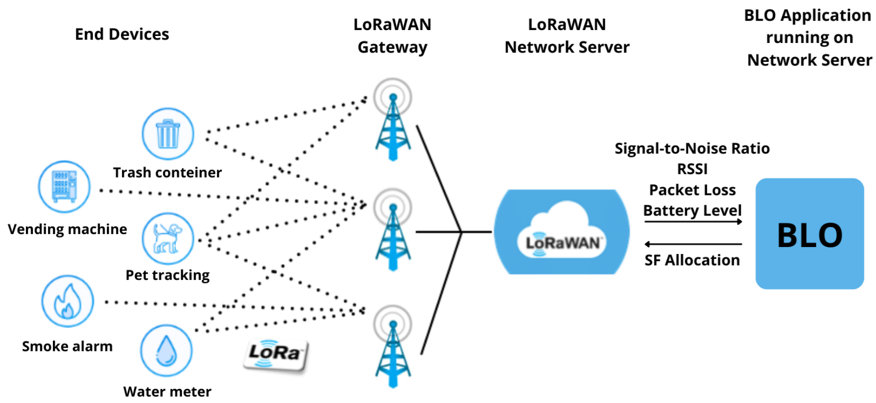


Figure 6: BLO integration with LoRaWAN, running on the Network Server.

To enhance the reliability of uplink communication, we first collect the maximum Signal-to-Noise Ratio (SNR) and Received Signal Strength Indicator (RSSI) from recent transmissions, which provides a benchmark for optimal performance. Next, we establish the minimum required SNR and RSSI necessary for demodulating uplink signals based on the current communication parameters, also considering the historical packet loss information. In cases when there are fewer available uplink measurements than needed, it is crucial to incorporate a safety margin to select the lowest acceptable spreading factor (SF), ensuring that sufficient margin remains for robust transmission. Once the lowest spreading factor is determined, alongside considering the battery status of each connected device, we execute the BLO (Battery Life Optimization) algorithm to carry out a new SF allocation, thereby enhancing overall system efficiency and device longevity.

Furthermore, when an ED experiences degraded RF conditions that fall outside its initially assigned SF range, due to factors such as mobility or obstacles, and fails to receive acknowledgments (ACKs) from the gateway, it will adjust the SF to a higher one to increase the communication range or rise the transmission power following its own strategy, until it re-establishes the connectivity with the network, according to the LoRaWAN specification [22]. At the end of each round, BLO assesses the current network conditions, verifying that the network conditions (RSSI and SNR) have changed since previous round and will update the minimum SF possible, ensuring that each ED operates at an appropriate SF.

Figure 7 illustrates the run-time performance of the BLO as the number of EDs increases. The MILP optimization was executed on an Ubuntu 20.04.3 LTS Virtual Machine (VM) with 1 vCPU and 6 GB of RAM, running on a PC equipped with an Intel(R) Core (TM) i7-8750H CPU @ 2.20 GHz. The results show that for small-scale networks (10 to 100 EDs), the run-time remains below 0.09 seconds. Considering a 1000 EDs scenario, the run-time reaches approximately 0.68 seconds. When scaling to 10,000 devices, the run-time is around 7 seconds. Although an MILP may be hard to solve, the results of our specific case show that BLO can be considered computationally light-weight even with thousands EDs.

Moreover, the optimization process for allocating SFs only occurs periodically, specifically after each round. A round can be defined by a total number of transmissions or a configurable amount of time, e.g. one hour, allowing for efficient processing without placing excessive demands on the NW's computing power. For example, considering a round of one hour with a packet rate of 1 packet/min, each ED will transmit a total of 60 packets. The round duration is user-defined and will only affect how often the BLO scheme will run to optimize the SF allocation in the NS.

Once a round is finished and the NS has all the information needed, it runs the BLO optimization. After the optimized allocation is computed, in the next reception window of each device, the NS sends the `LinkADR` message to the ED to setup the optimized SF value defined by the BLO. The next reception window is determined by the packet rate, thus giving sufficient time for the BLO

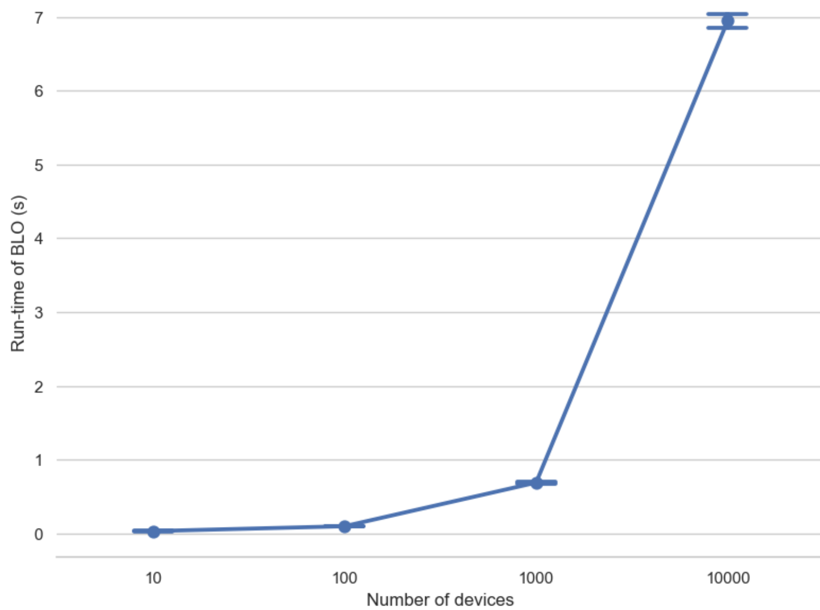


Figure 7: BLO run-time as a function of the number of EDs.

execution without computational overhead. The NS only sends the `LinkADR` message to the ED if the new optimized SF differs from the current one. This approach helps to minimize downlink traffic after each round. Furthermore, the reconfiguration of SF allocation within the network does not introduce any extra energy overhead on the EDs, because in the standard LoRaWAN operation all classes open reception windows after every transmission, as illustrated in Figure 1, and the NS uses these windows to send the `LinkADR` message.

6. Performance Evaluation

We use the NS-3 simulator [24] to evaluate the performance of BLO. The baseline benchmark is ADR, the allocation scheme included in LoRaWAN [22]. We also simulate other two solutions of the state-of-the-art, EXPLoRa-SF and EXPLoRa-AT (described in [11]). We evaluate different scenarios where N end devices are randomly distributed in a bi-dimensional space around a single

Table 7: Configuration parameters used in the simulations.

Parameter	Value
Carrier Frequency	915 MHz
Bandwidth	125 kHz
Code Rate (CR)	4/5
Duty cycle	[0.027 – 10] %
Message size	[20 – 51] bytes
Message period	[10 – 3600] seconds
Number of gateways	1
Number of nodes (N)	500 – 1000
Path loss parameters	$d_0 = 40\,\text{m}, \gamma = 2.08, \sigma^2 = 0, L_{pl}(d_0) = 127.41$ dB

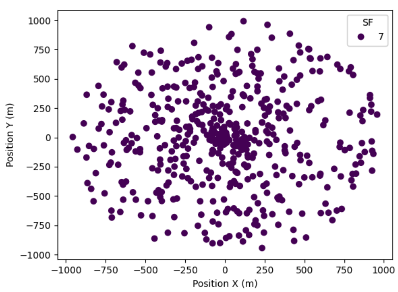
gateway (GW). The default payload size is 51 bytes, the highest possible value for SFs 10, 11 and 12, as shown in Table 3. Simulation results are presented with 99% confidence intervals. End devices use the communication transmission parameters listed in Table 7.

To compare BLO with other approaches, we use the DER metric as defined in [11], i.e., the ratio of messages that are correctly received by the GW to the total number of messages transmitted by all network EDs, within a time frame. The DER assesses the overall behavior and scalability of the network, instead of the individual node performance. While the NS-3 simulation environment provides a framework for simulating the network behavior of LoRaWAN devices, the battery level computation is done separately from NS-3. After each round of simulation, we calculate and register the remaining battery levels for each device based on prior battery data and the energy consumption observed in the current round. The energy consumption is determined using the experimental power consumption measurements for each SF, as detailed in Section 4, ensuring that we have a realistic representation of the battery status.

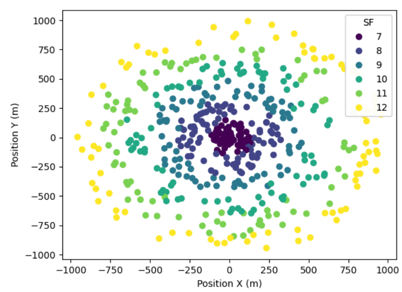
An example of the spatial distribution of SFs allocated by the different algorithms is shown in Figure 8. In this scenario 500 devices are distributed over a circle of 1000 m around the LoRa GW. All devices are within range of the GW independently of the SF chosen. ADR chooses the same SF for all devices, therefore producing the highest collision probability. EXPLoRa-SF distributes the devices into 6 equal groups and allocates the SF according to the RSSI of each device, meaning that the devices close to the GW use SF7, whereas farther devices are allocated higher SFs. EXPLoRa-AT allocates the maximum number of devices per SF using the channel congestion strategy, also considering the RSSI of each device. Therefore, the geographical distribution is similar to the EXPLoRa-SF pattern, which means devices close to the GW will be assigned lower SFs. Finally, BLO uses the overall distribution (number of devices allocated in each SF), similar to EXPLoRa-SF, but considering the current battery level in addition to the RSSI of each device. The BLO strategy produces an SF distribution which is not uniform according to the device's distance to the GW, as seen in Figure 8(d).

Figure 9 shows the minimum battery level (in %) over all 500 devices, after each round for each SF allocation method. Again, the 500 devices are distributed in a circle of 1000 m radius around the GW. All devices are within range and transmit one message per minute. Each simulation round lasts one hour, meaning that after 60 transmissions the SF allocation methods run again. The total duration of all rounds is 24 hours, allowing to infer the expected network lifetime, defined as the time at which the first node of the network runs out of energy. The updated information (mainly battery level and RSSI) is used as input by the algorithms after each round. Thus, the behavior of of each method over time can be assessed.

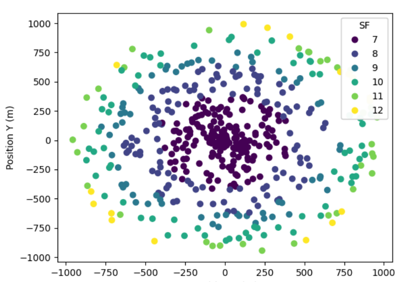
Figure 9 shows that ADR presents the least steep curve, maintaining a steady battery level superior of 90% after all rounds, providing the smallest energy consumption. This is expected since as the devices are near the gateway, ADR allocates all of them to SF7 during all the rounds. This produces the highest collision rate and lowest DER. BLO demonstrates better energy efficiency, with the minimum remaining battery gradually decreasing to approximately 85% by the end of 23 rounds. This is explained because BLO divides the EDs into different SF groups and, after each round, the devices with the lowest battery will be assigned to the lowest SF possible, which will spare less battery than



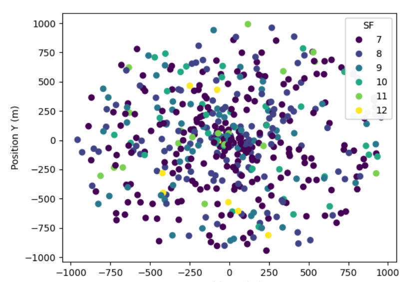
(a) ADR.



(b) EXPLoRa-SF.



(c) EXPLoRa-AT.



(d) BLO.

Figure 8: Allocation of SF for 500 EDs using different the four different algorithms.

with ADR constant SF. Nonetheless, EXPLoRa-SF and EXPLoRa-AT show more significant energy consumption. They also divide the devices into SF groups, but both ignore remaining battery information and allocate devices to SFs based only on RSSI data, which mainly depends on the distance to the GW. EXPLoRa-SF experiences the steepest decline, with the minimum battery level dropping below 40%, indicating the highest energy demand among all methods. Similarly, EXPLoRA-AT depletes the battery faster than ADR and BLO, maintaining a minimum battery level slightly above 50% after 24 hours. The results highlight that while ADR is the most battery-efficient algorithm, it may not optimize other network performance metrics. The energy remaining after 24 hours operation is 10 and 3.6 times larger with BLO than with EXPLoRa-SF and EXPLoRa-AT, respectively.

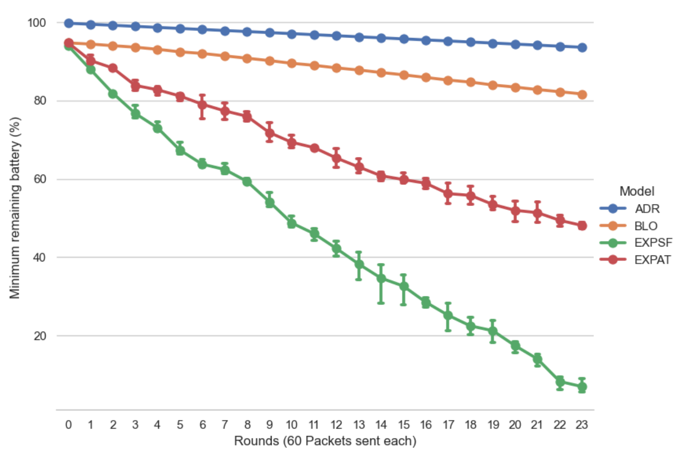


Figure 9: Lowest remaining energy as a function of rounds, with 500 EDs transmitting 51-byte packets each 60 s.

Figure 10 shows the behavior of the four algorithms with increasing packet rates, from 1 packet per hour to 6 packets per minute, again using 51-byte payload. ADR shows the worst DER in all cases. As the same SF7 is assigned to all EDs, the number of collisions is much larger than with the other three approaches. BLO and EXPLoRa-AT both allocate resources fairly considering the equalization of time of transmissions for each SF group. As a consequence, they show the best DER values. EXPLoRa-SF performs slightly better for the highest packet rate. This shows that the strategy to allocate devices into different SF groups (i.e. six equal groups) presents the advantage of decreasing the collision rate, even if using a simpler strategy which ignores energy consumption.

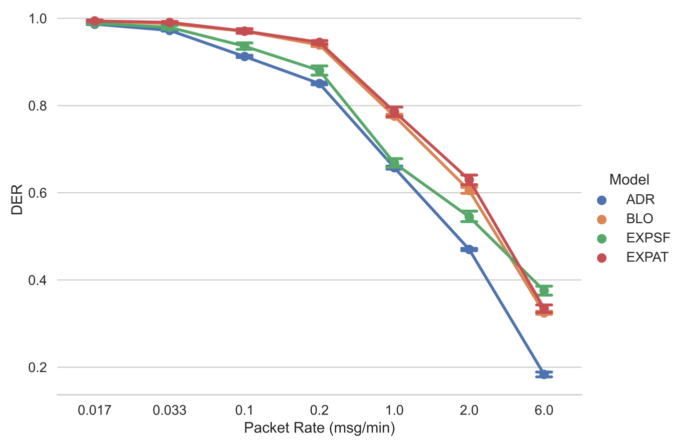


Figure 10: DER as a function of packet rate, with 500 EDs and 51-byte packets.

Figure 11 shows the performance of all approaches when the size of the area over which EDs are spread varies. BLO and EXPLoRa-AT allocation methods present the highest DER for all sizes, outperforming EXPLoRa-SF and ADR. When the distance increases, we see a decrease on the allocated SFs. This is due to more devices entering a zone where higher SFs are needed in order to improve the receiver sensitivity. The side effect is that more collisions will occur in these higher SFs. Again, we note that ADR produces the lowest DER for close ranges, because it allocates all devices to the same SF7.

Finally, we investigate the tradeoff between energy spent and information successfully transmitted, to assess the algorithms efficiency. Figure 12 shows the energy consumed per successfully received byte, varying the size of the area where EDs are located. Each ED sends one message per minute. Note that regardless of the distance to the GW, all the curves are reasonably stable. The ADR scheme has the lowest energy cost, starting below 5 mWh per byte and increasing only slightly at the maximum distance of 3000 m. This increase

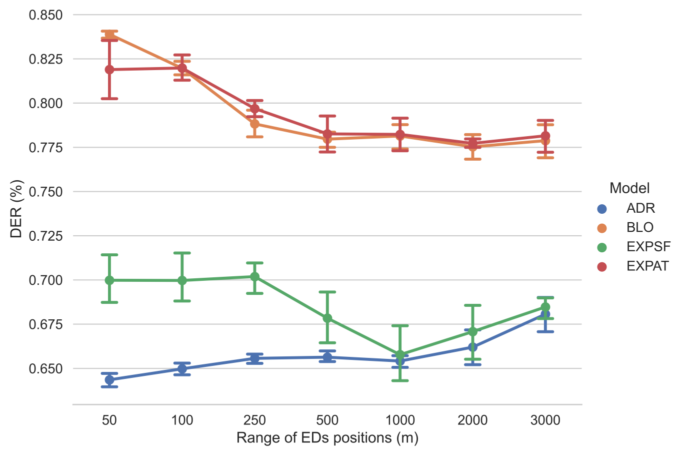


Figure 11: DER as a function of the maximum range where EDs are located, with 500 EDs and 51-byte packets.

is expected because as the ED moves away from the GW it is necessary to increase its SF, consuming more energy. We note that BLO and EXPLoRa-AT consume moderate energy, with BLO maintaining a stable consumption just below 8 mWh per byte, while EXPLoRa-AT is around 11 mWh per byte. In contrast, EXPLoRa-SF is the least energy-efficient, with an energy consumption above 25 mWh per byte for all distances. Comparing with the SoA approaches, BLO has approximately one third of the cost of the EXPLoRa-SF and 25% less than EXPLoRa-AT. The results suggest that ADR is the most energy-efficient algorithm. Although allocating all devices to the same SF causes a high collision rate, the cost of lower SFs transmissions are multiple times reduced comparing with higher SFs. Therefore, this energy-efficiency presented by ADR constitutes a tradeoff with other metrics, such as the DER in high-traffic network scenarios.

7. Conclusion

This paper has presented a dynamic strategy for optimizing the SF allocation in LoRa networks, named BLO for Battery Life Optimization. The key idea is to periodically group EDs into SF groups based on their current battery levels, minimizing collisions within these groups, improving the data extraction

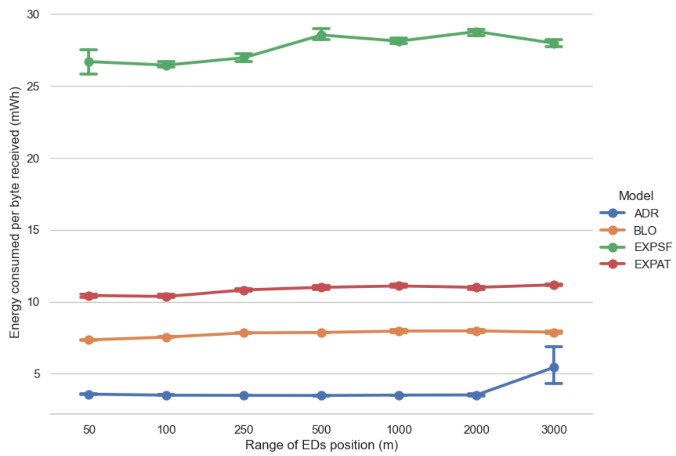


Figure 12: Energy cost per byte successfully received as a function of the maximum ED distance from the GW, with 500 EDs and 51-byte packets.

rate and fairness in energy consumption among the network. The periodic reallocation ensures a more balanced distribution of battery consumption across the EDs. The BLO strategy is compared with state-of-the-art SF allocation methods, outperforming LoRaWAN's ADR in term of DER, especially under high traffic conditions. Additionally, BLO outperforms both EXPLoRa-SF and EXPLoRa-AT in term of energy consumption, contributing to a extended network lifetime. In the future work, an analysis of the impact of transmission power control over the network lifetime and experiments with testbeds to evaluate the real-world performance and robustness of the BLO framework should be performed.

Acknowledgments

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. It was also supported by CNPq Grants SEI:01300.010794/2023-1, 408255/2023-4 and 309304/2021-0, FAPERJ Grants E-26/204.562/2024 and E-26/204.122/2024, and FAPESP Grants 23/00673-7 and 23/00811-0.

References

[1] P. Asghari, A. M. Rahmani, H. H. S. Javadi, Internet of things applications: A systematic review, Computer Networks 148 (2019) 241–261. doi:10.1016/J.COMNET.2018.12.008.

[2] U. Raza, P. Kulkarni, M. Sooriyabandara, Low power wide area networks: An overview, IEEE Communications Surveys & Tutorials 19 (2) (2017) 855–873. doi:10.1109/COMST.2017.2652320.

[3] C. Goursaud, J.-M. Gorce, Dedicated networks for IoT: PHY/MAC state of the art and challenges, EAI Endorsed Transactions on Internet of Things 1 (2015) 150597. doi:10.4108/eai.26-10-2015.150597.

[4] M. Bor, J. Vidler, U. Roedig, LoRa for the internet of things, in: International Conference on Embedded Wireless Systems and Networks (EWSN), Junction Publishing, USA, 2016, p. 361–366.

[5] M. Jouhari, N. Saeed, M.-S. Alouini, E. M. Amhoud, A survey on scalable LoRaWAN for massive IoT: Recent advances, potentials, and challenges, IEEE Communications Surveys & Tutorials 25 (3) (2023) 1841–1876. doi:10.1109/COMST.2023.3274934.

[6] O. Georgiou, U. Raza, Low power wide area network analysis: Can LoRa scale?, IEEE Wireless Communications Letters 6 (2) (2017) 162–165. doi:10.1109/LWC.2016.2647247.

[7] L. Vangelista, Frequency shift chirp modulation: The LoRa modulation, IEEE Signal Processing Letters 24 (12) (2017) 1818–1821. doi:10.1109/LSP.2017.2762960.

[8] LoRa Alliance, LoRaWAN regional parameters rp002-1.0.4 (2022). URL https://resources.lora-alliance.org/technical-specifications/rp002-1-0-4-regional-parameters

[9] Semtech Corporation, Understanding the LoRa® adaptive data rate (2019).
URL https://lora-developers.semtech.com/uploads/documents/files/Understanding_LoRa_Adaptive_Data_Rate_Downloadable.pdf

[10] M. C. Bor, U. Roedig, T. Voigt, J. M. Alonso, Do LoRa low-power wide-area networks scale?, in: 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), 2016, pp. 59–67. doi:10.1145/2988287.2989163.

[11] F. Cuomo, M. Campo, A. Caponi, G. Bianchi, G. Rossini, P. Pisani, EXPLoRa: Extending the performance of LoRa by suitable spreading factor allocations, in: 13th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2017, pp. 1–8. doi:10.1109/WiMOB.2017.8115779.

[12] F. Cuomo, J. C. C. Gámez, A. Maurizio, L. Scipione, M. Campo, A. Caponi, G. Bianchi, G. Rossini, P. Pisani, Towards traffic-oriented spreading factor allocations in LoRaWAN systems, in: 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), 2018, pp. 1–8. doi:10.23919/MedHocNet.2018.8407091.

[13] S. Hong, F. Yao, F. Zhang, Y. Ding, S.-H. Yang, Reinforcement learning approach for SF allocation in LoRa network, IEEE Internet of Things Journal (2023) 1–1doi:10.1109/JIOT.2023.3279429.

[14] L. Casals, B. Mir, R. Vidal, C. Gomez, Modeling the energy performance of LoRaWAN, Sensors 17 (10). doi:10.3390/s17102364.
URL https://www.mdpi.com/1424-8220/17/10/2364

[15] J. P. Shanmuga Sundaram, W. Du, Z. Zhao, A survey on LoRa networking: Research problems, current solutions, and open issues, IEEE Communications Surveys & Tutorials 22 (1) (2020) 371–388. doi:10.1109/COMST.2019.2949598.

[16] B. Reynders, W. Meert, S. Pollin, Power and spreading factor control in low power wide area networks, in: IEEE International Conference on Communications (ICC), 2017, pp. 1–6. doi:10.1109/ICC.2017.7996380.

[17] W. Gao, Z. Zhao, G. Min, AdapLoRa: Resource adaptation for maximizing network lifetime in lora networks, in: 28th IEEE International Conference on Network Protocols (ICNP), 2020, pp. 1–11. doi:10.1109/ICNP49622.2020.9259398.

[18] W. Gao, W. Du, Z. Zhao, G. Min, M. Singhal, Towards energy-fairness in LoRa networks, in: IEEE 39th International Conference on Distributed Computing Systems (ICDCS), 2019, pp. 788–798. doi:10.1109/ICDCS.2019.00083.

[19] J. C. da Silva, D. de L. Flor, V. A. de Sousa Junior, N. S. Bezerra, A. A. M. de Medeiros, A survey of LoRaWAN simulation tools in ns-3, Journal of Communication and Information Systems 36 (1) (2021) 17–30. doi:10.14209/jcis.2021.2.
URL https://jcis.sbrt.org.br/jcis/article/view/741

[20] J. C. Liando, A. Gamage, A. W. Tengourtius, M. Li, Known and unknown facts of lora: Experiences from a large-scale measurement study, ACM Trans. Sen. Netw. 15 (2). doi:10.1145/3293534.
URL https://doi.org/10.1145/3293534

[21] L. Beltramelli, A. Mahmood, P. Österberg, M. Gidlund, P. Ferrari, E. Sisinni, Energy efficiency of slotted LoRaWAN communication with out-of-band synchronization, IEEE Transactions on Instrumentation and Measurement 70 (2021) 1–1. doi:10.1109/TIM.2021.3051238.

[22] LoRa Alliance, LoRaWAN specification v1.1 (2017).
URL https://resources.lora-alliance.org/technical-specifications/lorawan-specification-v1-1

[23] L. Alliance, What is LoRaWAN?, Access: Mar 16th, 2025. https://lora-alliance.org/about-lorawan. (2020).

[24] NS-3, Detailed documentation of NS-3 (2024).
URL `https://www.nsnam.org/documentation/`