Ad Hoc Networks 122 (2021) 102598

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Adaptive priority-aware LoRaWAN resource allocation for Internet of Things applications

Eduardo Lima [a,*], Jean Moraes [a], Helder Oliveira [a], Eduardo Cerqueira [a], Sherali Zeadally [b], Denis Rosário [a]

[a] Federal University of Pará, Belém, Brazil
[b] University of Kentucky, KY, USA

ARTICLE INFO

Keywords:
IoT
LoRaWAN
Priority-aware
And resource allocation

ABSTRACT

LoRaWAN is the most adopted communication technology for the Internet of Things (IoT) applications for enabling flexible, cost-effective long-range communication with low power consumption. The deployment of a wide range of IoT devices brings many benefits to many applications, but new challenges arise from the network point of view with this deployment. For instance, IoT applications have different requirements in terms of Quality of Service (QoS) and energy efficiency in dense IoT scenarios. In this context, it is crucial to design an efficient resource allocation mechanism that enables real-time adjustments of radio-related parameters to improve the system's scalability while providing QoS and reducing IoT devices' energy consumption depending on the application requirements. This article proposes an adaptive priority-aware resource allocation mechanism to improve LoRaWAN scalability and energy consumption in a dense IoT scenario, called APRA. The simulation results show the benefits of APRA in improving the energy consumption by 95% and increasing the battery discharge time of the end device up to 5 years while yielding high packet delivery and low delay to high priority applications.

1. Introduction

Internet of Things (IoT) is expanding its portfolio to include a wide range of IoT applications, mainly due to the advances in different areas, such as embedded systems, microelectronics, communication, and sensing [1,2]. As a result, the number of 5G-connected IoT devices might reach 4.1 billion by 2024 [3]. In this context, IoT has been receiving much attention from both academia and industry due to its potential for use in the most diverse areas of human activity, such as smart cities, healthcare, agriculture, environmental monitoring, logistics, home/building automation, smart grid, critical infrastructure monitoring, and several other application areas [4–6].

IoT applications require low energy consumption (to address 10-years battery discharge time), high coverage, and cost-effectiveness [7–9]. So the communication technology used to transmit the collected IoT data plays a vital role in the massive adoption and deployment of IoT applications. To satisfy IoT applications' requirements, the Low-Power Wide-Area Network (LPWAN) emerged as a promising communication technology for supporting a wide variety of IoT applications in rural and urban areas [10]. Over the last few years, LPWAN has been increasingly used on a large scale by the industry. Recent market data shows an increase of 109% per year of connected LPWAN devices and an annual investment of more than US$4.5 billion from 2018 to 2023 [11].

Industry and research communities support different LPWAN technologies, such as LoRaWAN, SigFox, and NB-IoT [12]. Instead of common cellular infrastructures such as 3G and 4G, LPWAN solutions implement a communication technology with lower operating costs, low bit rate, long-range, and low energy consumption [13]. According to Haxhibeqiri et al. [7], the number of publications on Long Range (LoRa) areas has grown tremendously in the past years. In this context, LoRa is a wireless communication technology under patent by Semtech. LoRa refers to the radio modulation and the devices that use this modulation type. LoRa enables the devices to transmit over distances up to hundreds of kilometers. On the other hand, LoRa Alliance [14] considers LoRa radio as the physical layer and defines the upper layers and network architecture of LoRaWAN, *i.e.*, open network protocol and ecosystem [15,16]. In this way, LoRaWAN offers a cost-effective way to enable a large-scale deployment of End Devices (ED) that require less-complex medium access control mechanisms at the expense of low throughput [17].

* Corresponding author.
E-mail addresses: eduardo.lima.silva@itec.ufpa.br (E. Lima), jean.anjos.moraes@itec.ufpa.br (J. Moraes), heldermay@ufpa.br (H. Oliveira), cerqueira@ufpa.br (E. Cerqueira), szeadally@uky.edu (S. Zeadally), denis@ufpa.br (D. Rosário).

https://doi.org/10.1016/j.adhoc.2021.102598
Received 15 December 2020; Received in revised form 24 June 2021; Accepted 28 June 2021
Available online 8 July 2021
1570-8705/

Future LoRaWAN-based scenarios, especially in urban areas, will be composed of several thousand IoT devices per square meter. However, the densification of LoRaWAN generates a severe problem when more connected devices coexist in the same area with limited radio resources [18]. This issue significantly impacts the number of packets lost due to collision and interference thereby affecting network scalability and efficiency [19]. For instance, LoRaWAN Gateway (GW) might receive messages from many EDs on the same channel, leading to interference and poor application performance. In this context, the LoRaWAN physical layer considers a set of radio parameters that can be adjusted on-the-fly to provide a trade-off among transmission range, bit rate, airtime, energy consumption, and interference [18,20]. Existing works have demonstrated that an efficient combination of these radio-related parameters configured by a resource allocation mechanism has a significant impact on the IoT applications, resulting in better coverage, data delivery, and robustness with lower energy consumption [21,22].

The LoRaWAN architecture allows the deployment of many IoT devices and applications to coexist in the same physical space. In this context, there is a high heterogeneity of IoT applications in terms of Quality of Service (QoS) requirements, such as delay and reliability [23]. This scenario leads to new challenges in the design an efficient resource allocation mechanisms, which must consider the application priority to configure radio-related parameters to efficiently satisfy the required QoS for each IoT application cost-effectively. Hence, it is essential to consider key application requirements individually rather than a one-size-fits-all solution. It is important to advance the state-of-the-art with a resource allocation mechanism that can dynamically adjust LoRaWAN radio-related parameters based on the application requirements and radio conditions. As a result, the system will be able to use the available channels efficiently while minimizing both the number of collisions and the energy consumption.

This article proposes an *Adaptive Priority-aware Resource Allocation* mechanism called APRA. It aims to improve LoRaWAN scalability, efficiency, and energy consumption by selecting radio-related parameters on-the-fly while performing a priority-aware resource control for each type of IoT application. Initially, EDs have distributed among the Spreading Factor (SF) channels according to the application priority, favoring the allocation of EDs in the lowest SFs to reduce collisions. Besides, the highest priority EDs are configured with a higher Bandwidth (BW), if possible. Finally, to reduce energy consumption, the algorithm performs an optimal Transmission Power (TP) selection. Simulation results obtained demonstrate APRA's efficiency in Packet Delivery Ratio (PDR), Packet Error Ratio (PER), Time on Air (ToA), and energy consumption compared to recently proposed state-of-the-art algorithms. Specifically, APRA optimizes up to 71% of the network energy consumption, improves the ED's battery discharge time by 3.5 years, delivers above 99% of packets, and reduces the high priority EDs ToA.

APRA aims at advancing the state-of-the-art in improving LoRaWAN scalability and energy consumption in dense IoT scenarios. In this context, we summarize the main contributions of this work as follows:

1. We propose an optimal SF allocation scheme by considering the requirements of IoT applications to deliver smaller ToA and reduce collisions.
2. We present a BW selection mechanism to enable a higher bit rate for high-priority EDs.
3. We introduce a TP allocation scheme to improve the energy efficiency of the IoT system.
4. We also perform simulation experiments to evaluate the impact and benefits of APRA. The results show that APRA can effectively mitigate the challenges related to specific radio-related parameters on-the-fly to improve LoRaWAN scalability, efficiency, and energy consumption.

We organize the rest of the article as follows. Section 2 outlines the state-of-the-art about LoRaWAN resource allocation for IoT applications. Section 3 describes the scenario overview and APRA algorithm. Section 4 discusses the simulation description, methodology, and results. Finally, Section 5 concludes the work and outlines future directions.

2. Related work

Existing resource allocation approaches focus on improving the LoRaWAN scalability and reliability by adjusting different radio parameters. However, the diversity of IoT applications in terms of QoS requirements has not received commensurate attention. This section describes the state-of-the-art research results on resource allocation algorithms, and we discuss their strengths and weaknesses.

The standard resource allocation algorithm, also known as Adaptive Data Rate (ADR), analyzes the maximum Signal-to-Noise Ratio (SNR) among the last 20 uplinks and configures the respective ED according to the GW with the highest SNR. Next, ADR computes the difference between SNR and the required SNR to demodulate the packet. ADR uses this margin to increase or decrease the Data Rate (DR) or TP. In this context, ADR shows excellent energy-saving and packet delivery on networks with distributed devices. However, it loses performance for networks with centralized devices. In this scenario, ADR focuses on all packets with similar SNRs in the same SF, which results in significant interference.

Khaled et al. [24] divided the IoT scenario into groups of EDs and proposed a mechanism to compute the optimal configuration of SF and TP radio parameters for all the EDs in each group. The objective is to address the unfair LoRaWAN characteristic when EDs are very close to the GW or use lower SFs. However, even demonstrating promising results in PDR, do not treat the IoT applications requirements variety can harm QoS for services that need a higher bit rate and reliability.

Alenezi et al. [25] proposed a priority scheduling method using the K-Means clustering algorithm to reduce collision and achieve a trade-off between delay and throughput. The main idea is to create wait periods, where the lower priority EDs have to stop transmission until EDs with higher priority end their packet transmissions. To achieve this goal, the system fixes radio-related parameters and classifies the priority of groups of EDs according to the environmental measurements. However, this approach does not consider the optimal LoRaWAN parameters configuring when fixed, harming the network scalability and energy consumption.

Cuomo et al. [26] introduced resource allocation mechanisms called EXPLoRa-SF and EXPLoRa-AT, aiming to optimize LoRaWAN performance by configuring the SF values for the EDs based on the signal strength. EXPLoRa-SF attempts to equally distribute EDs in the GW's radio range to each SF, only restricted by their Received Signal Strength Indicator (RSSI) values and relevant thresholds. On the other hand, EXPLoRa-AT propagates impartial allotment of the ToA between the end nodes in the network, prioritizing the lower SFs to reduce the time collision probability. In this way, a higher signal strength leads to lower SF for a given ED, where it considers a limited number of EDs in each SF based on ToA. Nevertheless, the authors do not consider fundamental issues for the network performance: the EDs are not prioritized to obtain more reliable communication for more urgent IoT applications in terms of low delay. Additionally, the energy cost is not considered in the network configuration.

Dawaliby et al. [23] proposed two interrelated approaches. First, the authors aim to improve network flexibility and control through a network slicing strategy. This procedure classifies the IoT applications into three priority categories based on their QoS and reliability requirements. Additionally, the authors proposed an optimization algorithm that sets the best slicing method with SF and TP combination for each ED, which results in QoS maximization while reducing energy consumption and reliability cost. But, this solution, while it does not consider the BW parameter to offer a higher bit rate to most essential applications, also proves very operation costly because it does not present an on-the-fly functionality to classify the network slices and optimal allocate the LoRaWAN physical parameters.

El-Asser et al. [27] improved the LoRaWAN packet delivery success and throughput using two mechanisms, namely SensitivitySF and AssignmentSF. This approach considers the area of a signal in a specific SF that can be successfully demodulated by a receiver as an SF ratio and assigns the best SF service radius to each ED. SensitivitySF allocates the SF parameter and maximizes the receiver's sensitivity with a Coding Rate (CR) parameter selection. The AssignmentSF mechanism performs the same steps as well as a load balancing within each SF channel. Still, both of the algorithms do not consider essentials radio-related parameters configuration such as BW and TP. In this sense, the lack of a TP optimal configuration expresses the algorithms as costly mechanisms in terms of ED's energy consumption. Also, this work does not consider the network heterogeneity of IoT application requirements in terms of QoS.

Zorbas et al. [28] proposed a resource allocation mechanism to improve the LoRaWAN capacity by applying multiple communication parameters. They model the average success probability per set as a density function, analyzing intra-SF and inter-SF collisions. Based on such a model, each ED has different communication settings in terms of BW and SF. In this context, this work tries to assign the majority of EDs on a given BW and SF to increase the packet delivery probability for each BW and the distance between ED and a given GW. However, allocating BW and SF without considering the application requirements' priority-aware functionality affects the network's reliability. Besides, with a fixed TP configuration, the mechanism has an associated cost issue with the device's power consumption.

Moraes et al. [29] proposed a resource allocation mechanism called CORRECT to dynamically adjust CF and SF LoRaWAN parameters in order to minimize the channel utilization to reduce packets collisions. This work introduced a mechanism to find the optimum CF and SF settings based on the signal strength and the distance between the device and the GW. This work assigns the SF value by guaranteeing that the selected SF value provides packet reception at the GW with enough power, and also defines the ratio of devices to be assigned for each SF value based on the ToA, which gives priority to more IoT devices in the lowest SF values. Simulation results show that the proposed mechanism yields results close to the optimum model.

Babaki et al. [30] proposed a solution to improve the default LoRaWAN ADR algorithm by dynamically allocating the physical parameters, SF and TP, using the Ordering Weight Average Operator (OWA). This work aims to improve the network noise resilience and PDR in dense IoT scenarios by considering the OWA decision-making characteristic and the Packet Loss ratio (PLR) metric. However, this algorithm does not consider the BW configuration and the network application's requirements.

Table 1 summarizes the main characteristics of the analyzed resource allocation mechanisms based on the following characteristics: optimization goal, energy-efficiency, requirement-based differentiation decisions, and LoRaWAN radio parameters considered, meaning their strengths and weaknesses as a supported and not supported feature. Such characteristics can significantly improve the system performance in terms of reliability and energy. Based on our state-of-the-art analysis, we conclude that only a few works [23,24,26,30] provide energy-efficiency through TP adjustment or SF allocation. Also, existing works [24,26–30] do not deliver application requirements for resource allocation. Finally, none of the existing work efficiently combines SF, TP, and BW's adjustment to improve the scalability, reliability, and energy-efficiency. To the best of our knowledge, only APRA considers every critical feature previously mentioned not provided by existing resource allocation mechanisms.

3. Adaptive priority-aware resource allocation mechanism

This section proposes the APRA mechanism to configure LoRa physical parameters with LoRaWAN protocol. In addition, we present the impacts of each LoRaWAN parameter in LoRa transmissions. The mechanism aims to improve the LoRaWAN scalability and efficiency while considering the IoT application's network requirements for reducing energy consumption.

3.1. Network and system model

We consider a scenario composed of n EDs $ed \in ED = \{1, 2, \dots, n\}$ deployed in the monitored area and m LoRaWAN GW $gw \in GW = \{1, 2, \dots, m\}$ deployed based on a positioning algorithm [31,32]. Each ED has an identification $i \in [1, n]$, and a tuple $ed_i = (x_i, y_i, z_i)$ represents its geographical coordinates. Moreover, the tuple $gw_j = (x_j, y_j, z_j)$ represents the geographical coordinates of a given GW with an identification $j \in [1, m]$. LoRaWAN considers a star topology and a single-hop communication between EDs ed_i and GWs gw_j over several channels. A given ED ed_i broadcasts the message for neighboring GWs gw_j that forward the received message to the network server through an IP network. LoRaWAN communication is bi-directional, although the uplink communication from EDs to the network server is strongly favored as expected in many IoT applications. As Fig. 1 shows, the network server implements the resource allocation mechanism, such as APRA, to return the configuration of radio-related parameters, *i.e.*SF, BW, and TP, to EDs through the downlink communication.

We consider a scenario with diverse IoT applications with different QoS requirements that coexist in the same physical space. In this context, the network server could identify IoT applications by analyzing the message size and the flow volume along with statistical properties (*e.g.*, the maximum, average, and minimum of volume flow) as described by Vergutz et al. [33]. Based on this classification and the application requirements, the network server assigns a priority level $P = \{p_{l,i} | (l \in \mathbb{N}) \wedge (1 \leqslant l \leqslant o)\}$ for each ed_i, where o denotes the priority level. We consider three levels of priority ($o = 3$), *i.e.*, 1, 2, and 3, to represent high, medium, and low application priority, respectively [23]. Based on these priority levels, we can provide adaptability for the proposed mechanism. For instance, high priority EDs, such as industrial automation processes applications, require low latency and message transmission interval and low tolerance to data loss. In contrast, a medium priority group, such as smart home automation applications, requires high reliability and low latency but lower than the high priority. Finally, the low priority EDs, such as smart meters applications, have a high delay tolerance.

LoRaWAN enables the on-the-fly adjustment of different radio-related parameters, increasing or decreasing the channel utilization. Specifically, SF is one of the vital radio parameters configured to scale the network better. This parameter enables multiple EDs to simultaneously communicate on the same channel without interfering with each other due to an orthogonal design [27]. However, in a dense IoT environment, the resource allocation mechanism needs to choose different SFs and center frequencies to avoid collisions [34]. A given LoRaWAN GW gw_j cannot correctly decode messages transmitted simultaneously using the same SF and the same center frequency. As expected in future IoT scenarios, there is a higher probability of the ED transmitting on the same SF and CF in dense LoRaWAN. $SF = \{sf_k | (k \in \mathbb{N}) \wedge (7 \leqslant k \leqslant 12)\}$ defines how many chirps are sent per second by the data carrier. Higher SF values increase the receiver sensitivity and radio range at the cost of increasing the ToA and energy consumption to transmit a packet. As a result, SF has a significant impact on coverage, energy consumption, collisions, and ToA. Low SFs provide higher bit rate at the cost of reduced ranges, energy consumption, and ToA, while high SFs are the opposite of low SFs [35]. For instance, the ToA increases from 659 ms to 1318 ms for a packet with a payload of 20

Table 1
Summary of resource allocation mechanisms analyzed for a scenario with heterogeneous IoT applications.

Resource allocation mechanisms	Year	Optimization goal	Energy	Application requirements	LoRa parameters		
					SF	TP	BW
ADR	2016	Maximize the transmission range and energy-saving	✓		✓	✓	✓
Khaled et al. [24]	2018	Address the unfair LoRaWAN characteristic	✓		✓	✓	
Alenezi et al. [25]	2019	Reduce collisions		✓			
Cuomo et al. [26]	2017	Reduce collisions	✓		✓		
Dawaliby et al. [23]	2019	Maximize QoS	✓	✓	✓	✓	
El-Asser et al. [27]	2018	PDR and throughput			✓		
Zorbas et al. [28]	2018	Improve LoRaWAN capacity and reduce collisions			✓		✓
Moraes et al. [29]	2020	Maximize the channel utilization and reduce collisions	✓		✓		
Babaki et al. [30]	2020	Improve the noise resilience and PDR	✓		✓	✓	
APRA (proposed approach)	**2020**	**Improve scalability and energy optimization**	✓	✓	✓	✓	✓

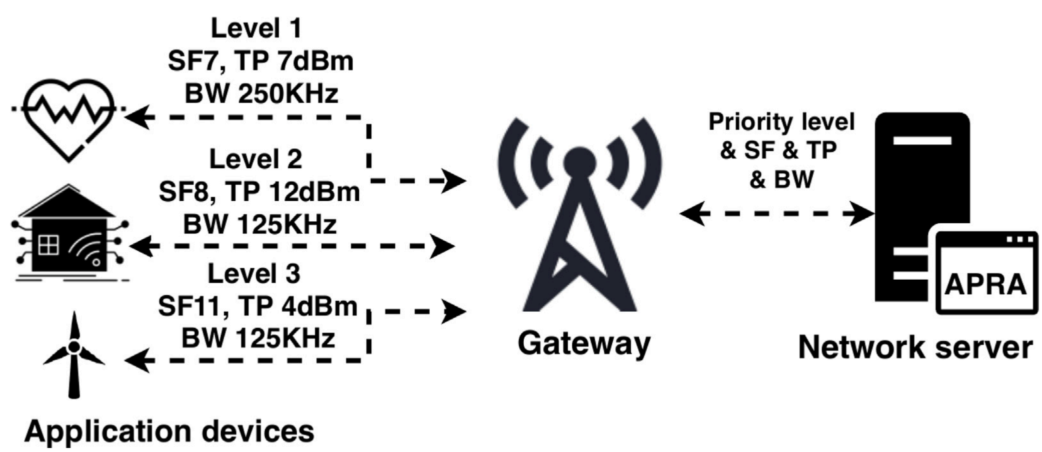


Fig. 1. LoRaWAN scenario overview.

bytes transmitted with sf_{11} instead of sf_{11}. Furthermore, sf_{11} consumes ten times more energy for transmission than sf_7 [36].

A given ED could use a specific BW (*i.e.*, $bw_w \in BW = \{125, 250, 500\text{ kHz}\}$) to transmit each message, which indicates the width of the frequencies in the transmission band [34]. A higher BW value allows higher bit rate, a lower ToA, and less coverage. Lower BW values behave inversely. Besides, CF is the center frequency of the chirps, with steps of 61 Hz. This parameter can be from 137 MHz to 1020 MHz, but the region parameters regulation limits it. This paper considers the European frequency plan EU863-870 [37], which defines the center frequency as 863 MHz to 870 MHz. Finally, TP (*i.e.*, $tp_p \in TP = \{tp_p | (p \in \mathbb{N}) \wedge (0 \leqslant p \leqslant maxTP)\}$) is the transmission power in dBm, where $maxTP$ is the maximum transmission power. Values close to $maxTP$ increase the transmission range at the cost of increasing the energy consumption. According to the SX1276 datasheet [38], TP can be configured from 0 dBm to 20 dBm in 1 dBm steps.

The indicative bit rate (b_{rate}) directly depends on the SF, BW, and CR values, *i.e.*, increasing the SF decreases the bit rate. Based on this, we compute the b_{rate} based on Eq. (1).

$$b_{rate} = SF \times \frac{BW}{2^{SF}} \times CR \tag{1}$$

As Eq. (2) [39] shows, we consider the Okumura–Hata propagation model to compute the path loss L_{pl} for the transmission between a given ED ed_i and GW gw_j. The model is computed for each ED ed_i and uses the center frequency (f) in MHz, the GW gw_j and the ED ed_i height in meters (z_i and z_j), and the Euclidean distance between them in kilometers ($dist_{gw,ed}$). Also, we consider the correction factor denoted by $a(ed)$ to simulate an urban environment.

$$\begin{aligned} L_{pl} &= 69.55 + 26.16 \cdot \log_{10}(f) - 13.82 \cdot \log_{10}(z_j) \\ &-a(ed) + (44.9 - 6.55 \log_{10}(z_j)) \cdot \log_{10}(dist_{gw,ed}) \end{aligned} \tag{2}$$

$$a(ED) \begin{cases} 8.29 \cdot (\log_{10}(1.54 \cdot z_i))^2 - 1.1 & , if \quad 150 \leq f \leq 200 \\ 3.2 \cdot (\log_{10}(11.75 \cdot z_i))^2 - 4.97 & , if \quad 200 < f < 1500 \end{cases} \tag{3}$$

As Eq. (4) shows, the Received signal strength indication ($RSSI_{j,i}$) is the signal strength of a packet received by a given GW gw_j from a ED ed_i, which considers the TP (tp_p), the antenna gain ($G_{antenna}$), and losses during the transmission by the environment (L_{pl}).

$$RSSI = tp_p + G_{antenna} - L_{pl} \tag{4}$$

The EU863-870 Regional Parameters [37] use $DR = \{dr_r | (r \in \mathbb{N}) \wedge (0 \leqslant r \leqslant 6)\}$ to represent SF and BW configurations. The packets' RSSI must be higher than the receiver's sensibility to demodulate the packets correctly. The receiver sensitivity depends on SF and BW configuration. Therefore, we can express the receiver sensitivity as $Sensitivity_{dr_r}$ (Table 2).

3.2. APRA operations

APRA takes into account the $RSSI_{j,i}$ and the ED's priority $p_{l,i}$ to determine the configuration of radio parameters, *i.e.*, SF, BW, and TP.

Table 2
Receiver sensitivity for SX1272 module in each DR and LoRa configuration.

DR	LoRa configuration	Sensitivity
dr_0	sf_{12} and bw_{125}	−137
dr_1	sf_{11} and bw_{125}	−134
dr_2	sf_{10} and bw_{125}	−132
dr_3	sf_9 and bw_{125}	−129
dr_4	sf_8 and bw_{125}	−126
dr_5	sf_7 and bw_{125}	−123
dr_6	sf_7 and bw_{250}	−120

Initially, APRA considers all EDs configured with the highest SF sf_k available because a high SF increases the transmission range and the probability to reach a GW gw_j. As Eq. (5) shows, a matrix R represent the $RSSI_{j,i}$ value perceived by a given GW gw_j from each ed_i, where each row in R represents a gw_j, each column represents an ed_i, and each value represents $RSSI_{j,i}$ measurement.

$$R = \begin{pmatrix} RSSI_{1,1} & RSSI_{1,2} & \cdots & RSSI_{1,n} \\ RSSI_{2,1} & RSSI_{2,2} & \cdots & RSSI_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ RSSI_{m,1} & RSSI_{m,2} & \cdots & RSSI_{m,n} \end{pmatrix}. \tag{5}$$

Next, APRA creates a priority matrix $Prior$ based on the values from the R, where each $Prior$ column is the result of multiplying each $R_{l,c}$ column by the respective EDs' priority value $p_{l,i}$, as Eq. (6) shows. Hence, the high value in $Prior$ represents the highest priority which will use the highest DR dr_r configuration.

$$Prior = \begin{array}{c} \\ gw_1 \\ gw_2 \\ \vdots \\ gw_m \end{array} \begin{array}{c} \begin{array}{cccc} ed_1 & ed_2 & \cdots & ed_n \end{array} \\ \begin{pmatrix} R_{1,1} \times p_{l,1} & R_{1,2} \times p_{l,2} & \cdots & R_{1,n} \times p_{l,n} \\ R_{2,1} \times p_{l,1} & R_{2,2} \times p_{l,2} & \cdots & R_{2,n} \times p_{l,n} \\ \vdots & \vdots & \ddots & \vdots \\ R_{m,1} \times p_{l,1} & R_{m,2} \times p_{l,2} & \cdots & R_{m,n} \times p_{l,n} \end{pmatrix} \end{array}. \tag{6}$$

APRA uses an array $\lim_{sf_k}$ to limit the number of EDs in each sf_k. The $\lim_{sf_k}$ array depends on W_1, W_2, n, and a constant WF. W_1 aims to provide EDs uniformly distributed in SFs sf_k (*i.e.*, 16.6% EDs in each SF), as Eq. (7) shows. W_2 aims to provide EDs distributed in SFs based on the b_{rate} inverse to reduce packet collision in lower SFs sf_k, as Eq. (8) shows. The constant WF adjusts the distribution and APRA adaptive characteristics, where values close to 1 result in a uniform EDs distribution on SFs, and $WF = 0$ results in an unfair distribution (more EDs in high SF).

$$W_1 = \frac{1}{\sum_{sf_k=7}^{12} 1}, \quad \forall sf_k \in SF \tag{7}$$

$$W_2 = \frac{b_{rate}^{-1}}{\sum_{sf_k=7}^{12} (b_{rate})^{-1}}, \quad \forall sf_k \in SF \tag{8}$$

$$\lim_{sf_k} = n \times [W_1 \times WF + W_2 \times (1 - WF)], \quad \forall sf_k \in SF \tag{9}$$

APRA sets the ed_i with the highest $Prior$ value to the highest dr_r, by considering the $RSSI_{j,i}$ perceived by a given GW gw_j from such ed_i, receiver sensitivity $Sensitivity_{dr_r}$ for each dr_r, and also limits the number of EDs in each sf_k (*i.e.*, $\lim_{sf_k}$). Specifically, as soon as the SF limit $\lim_{sf_k}$ for a given sf_k value has been reached, the limit or the $RSSI_{j,i}$ of a given ed_i is lower than or equal to the receiver's sensitivity in the calculated DR, APRA decreases the dr_r until ed_i can communicate with gw_j without exceeding the maximum number of EDs in each sf_k. Therefore, if any high priority ED has connection problems, it will be set to a low DR (high SF).

After select the dr_r configuration (*i.e.*, SF and BW), APRA performs the TP adjustment. In this way, APRA must define a minimal $RSSI_{j,i}$ value, called the $RSSI_{threshold}$, where a high $RSSI_{threshold}$ value means a higher chance of losing packets, while a low value results in energy savings. Based on the dr_r value, it is possible to compute the difference between the $RSSI_{j,i}$ and the receiver $Sensitivity_{dr_r}$. As a result, as soon as this difference is higher than $RSSI_{threshold}$, APRA decreases tp_p by a determined value until the difference between the $RSSI_{j,i}$ and the receiver $Sensitivity_{dr_r}$ is less than or equal to $RSSI_{threshold}$. In this way, the APRA reduces the energy consumption by decreasing the TP without affecting the transmitted packets' integrity.

Algorithm 1 shows APRA's operations. APRA needs the set of EDs, sensitivity in each DR dr_r (Table 2), and $\lim_{sf_k}$ (Eq. (9)) as inputs. For each iteration, APRA chooses the ED ed_i with the highest value in $Prior$ (Line 2) and computes its higher dr_r configuration. At that moment, we disregard $\lim_{sf_k}$ and we consider only the receiver's sensitivity $Sensitivity_{dr_r}$ and packet's RSSI $RSSI_{j,i}$. APRA starts with dr_6 (sf_7 at bw_{250}) (Lines 3), and decreases dr_r if the EDs' RSSI $RSSI_{j,i}$ is not enough to decode the packet (Lines 4 and 5). To use the $\lim_{sf_k}$, we translate the dr_r to an sf_k and bw_w values (Line 6), where bw_{250} is only available in sf_7, according to the EU863-870 Regional Parameters. In the second While loop (Line 7), if the $\lim_{sf_k}$ value is less than 1 in that sf_k, the next sf_{k+1} is checked (Line 8). With sf_k and bw_w values computed, we can translate them to a dr_r value (Line 9) which will be used in TP adjustment to compare RSSI $RSSI_{j,i}$ to receiver sensitivity $Sensitivity_{dr_r}$ values in each dr_r. After setting the ed_i to the chosen bw_w and sf_k (Line 10), we compute this ED in $\lim_{sf_k}$ (Line 11). At Line 12, we start the TP calculation by setting the $RSSI_{threshold}$ and TP_{step} parameters. APRA calculates the exceeded RSSI (Line 13), and while this value is higher than $RSSI_{threshold}$, the tp_p is decreased by TP_{step} (Lines 14–16). To calculate the next EDs' sf_k and bw_w set, APRA updates the EDs' column in $Prior$ to a low value (Line 17).

We analyze the complexity of APRA as follows. In the worst case, APRA's complexity is $O(n * (|DR| + |SF| + RSSI_{Threshold}))$. Given the cardinality of the set DR ($|DR| = 7$) and the set SF ($|SF| = 6$) for EU868 regional parameters. We use $RSSI_{Threshold} = 6$ in our simulations as the APRA's default value. The number of EDs (n) can be considered a constant value. Hence, the APRA algorithm's complexity is then O(n), where it depends only on the number of EDs (*i.e.*, n).

Algorithm 1: APRA

Input: EDs, $Sensitivity_{dr_r}$ and lim_{sf_k}(Eq. (9))
Output: All EDs sf_k, BW and TP parameters configured
1 **for** EDs **do**
2 Choose the ED ed_i with max value in $Prior$;
3 Starts at maximum DR value(dr_6);
4 **while** $RSSI_{j,i} < Sensitivity_{dr_r}$ ς $dr_r > 0$ **do**
5 dr_{r--};
6 Update sf_k and bw_w from dr_r;
7 **while** $lim_{sf_k} < 1$ ς $sf_k < 12$ **do**
8 sf_{k++};
9 Update dr_r value from sf_k and bw_w for the ed_i;
10 Update lim_{sf_k};
11 Set $RSSI_{Threshold}$ and TP_{step} values;
12 **while** $RSSI_{j,i} - Sensitivity_{dr_r} > RSSI_{Threshold}$ **do**
13 $tp_{(p-TP_{step})}$;
14 Update EDs' $Prior$ column with a lowest value;

4. Evaluation

This section presents the simulation environment, describes the parameters used in our LoRaWAN scenario, as well as the performance metrics used to evaluate the proposed resource allocation mechanism in a scenario with diverse IoT applications coexisting in the same physical space.

Table 3
Simulation parameters used.

Parameter	Value
Center frequency	EU863-870 MHz
Antenna gain	0 dBm
BW	125 kHz
CR	4/5
Packet length	20 bytes
Packet Transmission Interval	5 mins
Simulation time	60 mins
Number of GW	1
Number of EDs	[1000–5000]
Field size area	4 × 4 km
Transmission power	20 dBm

4.1. Scenario description and methodology

We use the LoRaWAN simulation tool, LoRaSim [22], to implement the resource allocation mechanisms. LoRaSim allows a radio-related parameter configuration and simulates uplinks as LoRaWAN traffic with packet collision and interference. We deployed one GW centralized in a two-dimensional scenario with 4 × 4 km, and a different number of EDs around the GW, in a range of 1000 to 5000 EDs in steps of 1000, as expected in massive IoT deployments.

We consider a scenario with a distinct set of IoT applications coexisting in the same physical space with different QoS requirements. The packets airtime and energy consumption for high and medium priority EDs should be minimal. Thus, these EDs should be set at the lowest SF, while low priority EDs should be placed only after all medium and high priority EDs are already set. For simulations, the EDs' priority level are randomly chosen, where EDs have the same probability for each priority level. To generate realistic data traffic, we configure EDs to transmit data packets with 20 bytes and sending period fixed at 5 min. Table 3 summarizes the main simulation parameters.

We conducted simulations with six different resource allocation mechanisms: Min_Airtime, ADR, CORRECT, EXPLoRa-SF, EXPLoRa-AT, and APRA. Specifically, Min_Airtime assigns sf_7 to all EDs with a BW value of 125 kHz to capture a maximum bit rate with minimum airtime [40]. It implements the ADR mechanism provided by *The Things Network*, which adjusts SF and the TP based on the SNR of the packets [41]. EXPLoRa-SF uniformly distributes EDs in each SF to reduce packet collision [26]. The EXPLoRa-AT computes a fairness ToA distribution to normalize the ToA for all SF configurations [26]. Consequently, EXPLoRa-AT prioritizes the lower SFs to reduce the total network ToA. CORRECT considers a ToA array to normalize the ToA of each ED by considering the CF configuration to reduce packet collision [29]. Finally, APRA optimizes the SF and BW parameters in a combined approach according to the EDs' priorities to improve the network reliability while optimizing the EDs' energy consumption through an efficient TP reduction, as described in Section 3.2.

We performed a greedy search to find the ideal WF value. In this context, we conclude that $WF = 0.9$ (distribution based on W_1) provides better results. Based on this WF, the portion of EDs in each SF is 15.20%, 15.4%, 15.7%, 16.2%, 17.7%, 19.8%, for sf_7, sf_8, sf_9, sf_{10}, sf_{11} and sf_{11}, respectively. Based on these limits and $Prior$, APRA sets high priority EDs in lower SF and low priority EDs in higher SFs. With these settings APRA maximizes the bit rate and minimizes ToA for high priority EDs.

4.2. Performance metrics

We used the following six performance metrics to evaluate the performance of the resource allocation mechanisms analyzed for LoRaWAN. These metrics include: SF allocation, PDR, PER, ToA, energy consumption, and battery discharge time.

The ***SF allocation*** shows the EDs' Cartesian position and the SF snapshot for the simulated scenario with EDs $n = 1000$ and one seed (*i.e.*, $seed = 0$). Each small dot represents an ED at the $[x, y]$ position, and the color of the dot represents the SF. The dot is brown if it cannot communicate with the GW. The big red point represents the GW at the $[0, 0]$ position. With this metric, we can validate the implementation of the state-of-the-art algorithms and analyze the behavior of APRA when considering the EDs' priorities. This metric helps to understand the PDR, PER, ToA, and energy results based on the SF distribution because the SF is one of the leading radio parameters that affects the range, ToA, collision, and energy consumption.

PDR $\in [0, 1]$ is an effective way to analyze the network deployment as a whole, *i.e.*, a PDR close to 1 means the optimal network deployment (minimum number of collisions and losses) [22]. As Eq. (10) shows, we compute PDR from total number of received packets (N_{rec}) and the total number of packets sent (N_{sent}).

$$PDR = \frac{N_{rec}}{N_{sent}} \tag{10}$$

PER $\in [0, 1]$ serves to analyze network packet losses and collisions. In optimal network deployment (*i.e.*, minimum number of collisions and losses), PER is close to 0. As Eq. (11) shows, we compute PER from the total number of packets that suffer collisions ($N_{collision}$), total number of packets lost (N_{lost}), and total number of packets sent (N_{sent}).

$$PER = \frac{N_{collision} + N_{lost}}{N_{sent}} \tag{11}$$

The ***ToA*** means the packet's demodulation time that depends on radio parameters. For instance, the ToA increases from $659_m s$ to $1318_m s$ for a packet with payload of 20 bytes transmitted with SF 12 instead of 11, respectively [36]. To compute the ToA value in milliseconds (ms), we need to calculate the preamble duration (D_{preamb}) and the payload duration ($D_{payload}$). The D_{preamb} (Eq. (12)) depends on preamble symbols (n_{preamb}) and the symbol duration D_{symb} (Eq. (13)). The $D_{payload}$ is given by Eq. (14), where nPS is the number of payload symbols. The nPS is given by Eq. (15). It depends on the number of PayLoad bytes (PL) and the SF and CR transmission parameters.

Furthermore, nPS depends on the boolean variables H and DE. The control variable H represents the use of an explicit header. The variable DE represents the low DR optimization. It is enabled ($DE = 1$) for DR_0 and DR_1, and disabled ($DE = 0$). Lastly, Eq. (16) gives the packet ToA.

$$D_{preamb} = (n_{preamb} + 4.25) \times D_{symb} \tag{12}$$

$$D_{symb} = \frac{2^{SF}}{BW} \tag{13}$$

$$D_{payload} = nPS \times D_{symb} \tag{14}$$

$$nPS = 8 + max\left(Ceil\left(\frac{(8PL - 4SF + 28 + 16 - 20H) \times (CR + 4)}{4(SF - 2DE)}\right), 0\right) \tag{15}$$

$$ToA = D_{preamb} + D_{payload} \tag{16}$$

We define the ***energy consumption*** as the total energy consumed in all successfully demodulated packets, which is computed based on the packets' airtimes [22,29]. This is because LoRaSim cannot calculate the energy consumption for sleep, receiving, and others device modes. In this way, the battery discharge time and energy consumption metrics consider only the transmission mode consumption. By analyzing the SX1276 LoRa Chip datasheet and based on the TP, we can infer the drain current i for each transmitted packet. In addition, we consider a 3.6 volts power supply and the packet's airtime ToA in ms. Finally, we convert the ToA in seconds (s) and sum all EDs' consumption to obtain the energy consumption (in joule - J), as Eq. (17) shows.

$$energy = \frac{\sum^{EDs} \sum^{pckts} V \times i \times ToA}{1000} \tag{17}$$

The ***battery discharge time (BDT)*** metric computes the average time (in years) that an ED discharges its battery. Therefore, the number of EDs will not affect the result because it is an average. We consider the average packet airtime and the ED's average drain current that depends on TP to compute the consumption on transmission mode (similar to energy consumption metric). To compute the ED's average energy consumption, we start with the basic consumption equation $C = i \times \Delta t$, where C is the consumption in A/h, i is the drain current of the load in A, and Δt is the time that load is draining. In Eq. (18), we compute the ED's average consumption in A/h, where i is the ED's drain current in A, $\Delta t = ToA$ is the packets' airtimes in ms, and n is the number of EDs. This metric considers only all successfully demodulated packets.

$$EDsAvgCons = \frac{\sum^{n} i \sum^{pckts} ToA}{n \times 1000 \times 60 \times 60} \tag{18}$$

In Eq. (19), we compute the BDT in hours from the battery capacity (2.6 Ah) and $EDsAvgCons$. Finally, we convert it in years.

$$BDT = \frac{2.6}{EDsAvgCons} \times \frac{1}{24 \times 30 \times 12} \tag{19}$$

4.3. Results

The SF is one of the leading radio parameters that affects the range, ToA, collision, and energy consumption. In this context, Fig. 2 shows the SF allocation for each resource allocation mechanism as soon as 1000 EDs are deployed around the GW. This behavior is quite similar to other number of EDs deployed around a LoRaWAN GW, *i.e.*, 2000, 3000, 4000, and 5000. By analyzing the results of Fig. 2, we observe that Min_Airtime allocates all EDs to the lowest SF (sf_7) regardless of the application's priority. This behavior results in a minimum transmission range and connection problems, *e.g.*, 11.8% of EDs are out of the range of the GW, which results in packet loss. The ADR prevents connection problems when allocating the SF according to the SNR of the last 20 EDs' packets. ADR sets EDs close to the gateway to the lowest SFs and EDs far away to the gateway to the highest SF.

EXPLoRa-SF, EXPLoRa-AT, and CORRECT mechanisms set the SF value for each ED regardless of the application's priority. Specifically, EXPLoRa-SF and EXPLoRa-AT mechanisms consider the RSSI to assign the SF value, where EDs with high RSSI values (*i.e.*, close to the GW) are allocated lower SF values (*i.e.*, sf_7/sf_8). Specifically, EXPLoRa-SF distributes EDs equally in the available SFs' values (*i.e.*, each SF has around 20% of EDs). On the other hand, EXPLoRa-AT and CORRECT set more EDs to the lowest SFs because these mechanisms define the ratio of EDs in each SF value based on the ToA. Specifically, the portion of EDs is 47.4%, 25.8%, 14.2%, 7.1%, 3.5%, and 0.2% to sf_7, sf_8, sf_9, sf_{10}, sf_{11}, and sf_{12}, respectively. CORRECT configures SF settings based on the signal strength and the distance between the ED and the GW.

Finally, APRA can adapt the LoRaWAN radio-related parameters on-the-fly by considering the priorities $p_{l,i}$ and RSSI instead of considering only the RSSI value as ADR, EXPLoRa, and CORRECT solutions propose. This behavior causes a scattering of SF allocations without affecting the packet delivery probability. For instance, applications with higher QoS requirements may take advantage of a higher packet transmission rate, resulting in higher reliability and lower energy consumption. Hence, APRA could adapt the radio parameters based on the application priority. This adaptive behavior prioritizes setting high-priority EDs to lower SF values (*i.e.*, $sf_7 - sf_8$) to guarantee lower ToA, packet collision, and energy consumption. In addition, APRA sets low priority EDs to higher SF values (*i.e.*, $sf_{11} - sf_{12}$) because such applications could experience a higher ToA. APRA assigns the radio parameters to reach the required RSSI to the GW correctly demodulate packets of any priority level.

Fig. 3 shows the PDR provided by Min_Airtime, ADR, EXPLoRa-SF, EXPLoRa-AT, CORRECT, and APRA mechanisms for a scenario with a different number of EDs and three application priority levels. Min_Airtime has the lowest PDR due to settings all EDs in the sf_7 which increase the likelihood of packet collisions, and also suffers from a low transmission range resulting in packet errors. ADR loses performance in networks with many EDs nearby because it sets the SFs according to the SNR. As a result, PDR decreases for networks with many EDs in a single sf_k because the likelihood of packet collisions increases. By uniformly distributing EDs in each sf_k, EXPLoRa-SF reduces the collision probability which increases the PDR results. By limiting EDs in each SF according to ToA, EXPLoRa-AT and CORRECT reduce packets collisions by balancing the ToA of packets in each SF and decrease packet losses when using lower SF ties. Both works assign the SF value by guaranteeing that the selected sf_k value provides packet reception at the GW with enough RSSI without considering the application priority. APRA increases the PDR by up to 5.68%, 3.65%, 1.26%, and 1.22% compared to Min_Airtime, ADR, EXPLoRa-AT, and CORRECT respectively, for high-priority EDs and dense LoRaWAN (EDs n = 5000). Furthermore, APRA increases the PDR by up to 5.97%, 3.89%, 1.54%, and 1.17% compared to Min_Airtime, ADR, EXPLoRa-AT, and CORRECT respectively, for medium-priority EDs and dense LoRaWAN (EDs n = 5000). Finally, APRA provides a lower packet loss ratio at all priority levels, resulting in a better PDR performance compared with other mechanisms. Furthermore, the WF value adjusts the APRA behavior in terms of packet collision. For instance, $WF = 1$ decreases high priority packet collisions and low priority packet losses resulting in low priority PDR increases.

Fig. 4 shows the PER for different resource allocation mechanisms with a different number of EDs and application priorities (low, medium, and high). In contrast to PDR values, low PER values mean better PDR performance. PER results confirm the APRA's benefits in packet delivery for all priority levels by selecting the SF and BW according to the receiver sensitivity and the packets' RSSI of the devices. For instance, APRA has better performance than EXPLoRa-SF in high and low priorities EDs because it also makes BW allocation instead of only SF allocation. On the other hand, the Min_Airtime has the highest PER and, consequently, the worst result in all ED priorities. ADR mechanisms achieve high PER results for all EDs' priorities because this mechanism does not use all available SFs. The EXPLoRa-AT and CORRECT mechanisms obtain similar results for all priorities by setting more EDs in the lowest SFs (sf_7-sf_8). Hence, PER and PDR results confirm that APRA obtains better performance and it is the best to adapt to different application priorities. For high-priority EDs, APRA decreases PER by up to 80.8%, 73.4%, 49.5% and 48.7% compared to Min_Airtime, ADR, EXPLoRa-AT, and CORRECT, respectively (EDs n = 5000). Furthermore, APRA decreases the PER by up to 82.5%, 75.8%, 56.0%, and 49.2% compared to Min_Airtime, ADR, EXPLoRa-AT, and CORRECT, respectively, for medium-priority EDs and dense LoRaWAN (EDs n = 5000).

Fig. 5 shows the average ToA of packets received for different resource allocation mechanisms and different application priorities (*i.e.*, low, medium, and high). We conclude that Min_Airtime, ADR, EXPLoRa-SF, EXPLoRa-AT, and CORRECT have the same behavior for all priority levels in terms of ToA because such mechanisms do not consider the application priority for the SF allocation. Min_Airtime has the lowest ToA regardless of the application priority because all EDs are set to the lowest SF (*i.e.*, sf_7). On the other hand, EXPLoRa-SF has a high ToA value compared to EXPLoRa-AT, CORRECT, and Min_Airtime because it equally distributes the EDs along the available SF, leading to having many EDs with higher SFs. Both EXPLoRa-AT and CORRECT mechanisms assign the EDs' ratio for each SF value based on ToA, giving priority to more EDs in low SF values, leading to a low ToA compared to EXPLoRa-SF. Finally, the APRA mechanism assigns the highest DR (*i.e.*, lower SF and high BW) for high priority EDs, leading to lower ToA values for this application priority. For instance, high values of BW give a higher bit rate, leading to a lower ToA. We can also conclude that ToA increases when the application priority decreases for the APRA mechanism because it assigns a higher SF and low BW values to EDs transmitting data of low priority applications. Specifically, for high-priority EDs, APRA reduces ToA by up to 83.5%, 52.3%, and 53.2% compared to EXPLoRa-SF, EXPLoRa-AT, and CORRECT, respectively

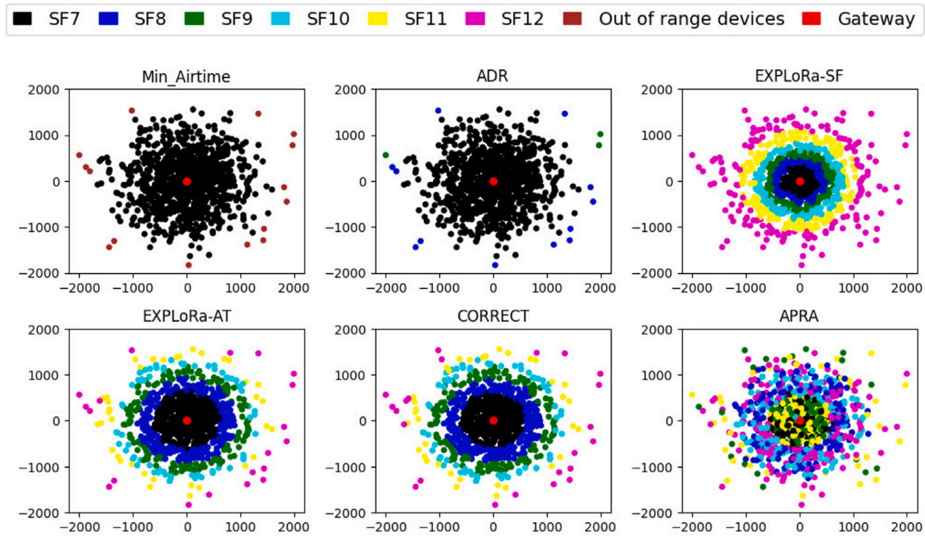


Fig. 2. SF allocation for each resource allocation mechanism for a scenario with 1000 EDs and a centralized GW.

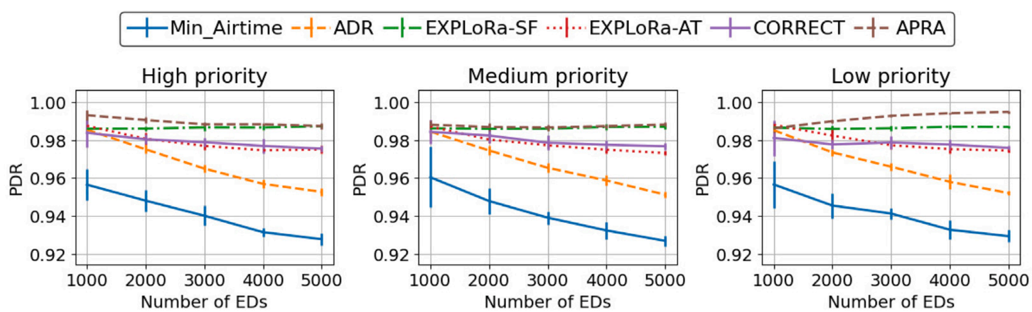


Fig. 3. PDR according to each priority and number of EDs for each resource allocation mechanism.

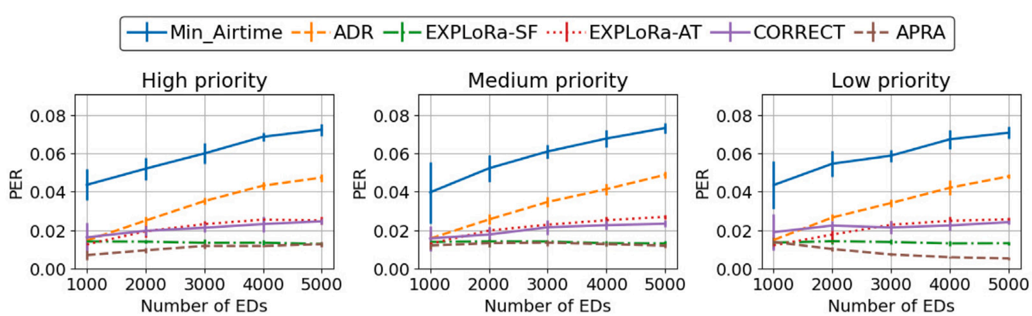


Fig. 4. PER according to each priority and number of EDs for each resource allocation mechanism.

(EDs n = 5000). Furthermore, APRA decreases the ToA by up to 22% compared to EXPLoRa-SF for medium-priority EDs and dense LoRaWAN (EDs n = 5000). For high and low priority levels, APRA increases ToA to increases PDR.

Fig. 6 shows the energy consumption of each resource allocation mechanism and each application priority level. Min_Airtime, ADR, and APRA have the lowest energy consumption for high and medium priority EDs. For MinAirtime, this occurs because sf_7 has low ToA. For instance, sf_{11} consumes ten times more energy than sf_7. ADR makes the TP allocation to save energy on EDs with high SNR. APRA has both advantages. Decreasing TP for high RSSI EDs and using BW allocation, APRA reduces the ToA of high priority EDs (sf_7-sf_8). Even for high priority EDs, EXPLoRa-SF, EXPLoRa-AT, and CORRECT had the highest energy consumption because these mechanisms do not have TP and BW allocation. APRA has the second highest energy consumption for low priority EDs because the lowest ToA resources have been used in high priority EDs. For high priority EDs, APRA reduces energy consumption by up to 57.2%, 19.1%, 95.0%, 85.6% and 85.9% compared to Min_Airtime, ADR, EXPLoRa-SF, EXPLoRa-AT and CORRECT, respectively (EDs n = 5000). APRA reduces the energy consumption for medium priority EDs by up to 84.2%, 53.7%, and 55.5% compared to EXPLoRa-SF, EXPLoRa-AT, and CORRECT, respectively (EDs n = 5000). Finally, Min_Airtime, ADR, EXPLoRa-SF, EXPLoRa-AT, and CORRECT mechanisms shows a similar behavior in terms of energy consumption for all application priorities because they do not consider application priority when allocating the EDs' radio parameters.

Fig. 7 shows the average BDT for each resource allocation mechanism. We note that APRA has the best battery duration due to an optimal TP allocation while also achieving an intelligent SF and BW allocation, which results in minimum packet airtime. ADR has good

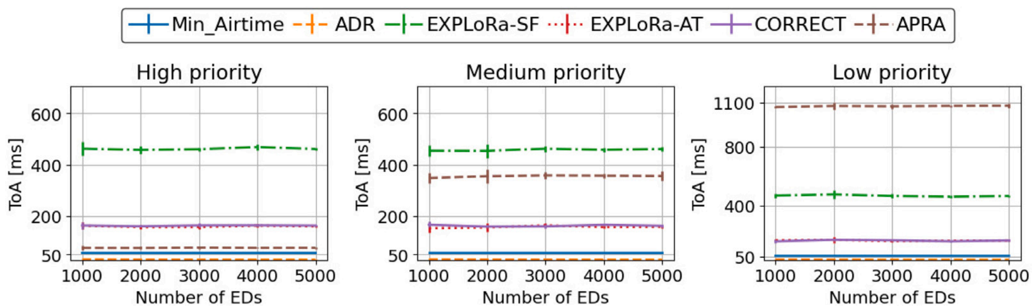


Fig. 5. ToA according to each priority and number of EDs for each resource allocation mechanism.

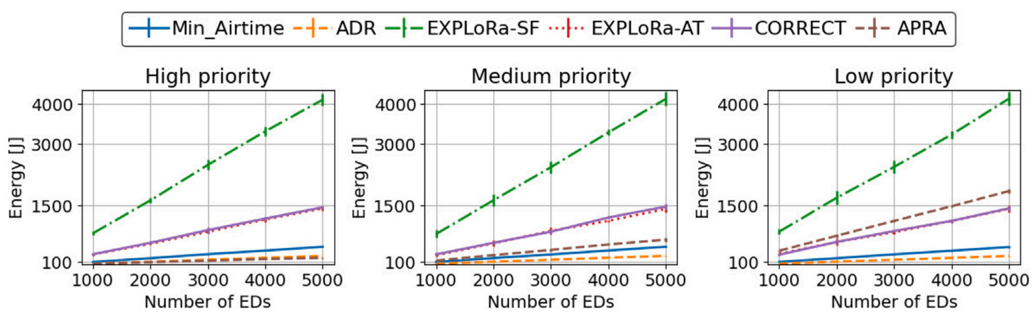


Fig. 6. Energy consumption according to each priority and number of EDs for each resource allocation mechanism.

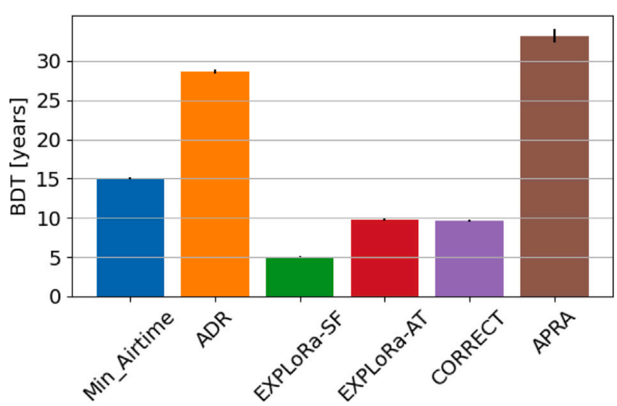


Fig. 7. BDT according to number of EDs for each resource allocation mechanism.

performance because the TP is reduced according to the packet SNR. Min_Airtime has a good BDT result because it sets all EDs to sf_7, which results in a lower ToA at the expense of high packet losses. EXPLoRa-AT and CORRECT mechanisms have similar results because both set EDs in SF based on the ToA. As expected, EXPLoRa-SF configures many EDs in higher SFs, *i.e.*, sf_{11} and sf_{12}, and incurs the highest energy consumption. In summary, APRA increases the duration time by up to 18.5, 4.4, 28.8, 24.1, and 24.1 years than Min_Airtime, ADR, EXPLoRa-SF, EXPLoRa-AT, and CORRECT, respectively.

From our performance evaluation analysis, we found that APRA demonstrated a better performance in delivering high PDR values in all priority groups which resulted in lower PER values, while showing lower ToA for high priority EDs due to its smart allocation of SF and BW. Besides, APRA is also energy-efficient for high priority EDs and yields reasonable battery duration due to the useful TP allocation in each ED. As expected, the performance of APRA reduces when the application priority decreases because it considers the application priority in order to allocate radio-related parameters to efficiently meet the required QoS for each IoT application cost-effectively. Min_Airtime and ADR have lower ToA and energy consumption at the cost of a high packet loss ratio regardless of the application priority, which is explained by the fact that more EDs are set to the lower SFs (sf_7/sf_8). EXPLoRa-SF reduces the packet collision probability at the cost of increasing the ToA and the energy consumption by uniformly distributing the number of EDs in each SF because there are many EDs in higher SFs. Both EXPLoRa-AT and CORRECT mechanisms use all SFs, but has more EDs in low SF values which lead to low ToA compared to EXPLoRa-SF. However, Min_Airtime, ADR, EXPLoRa-SF, EXPLoRa-AT, and CORRECT mechanisms do not consider application priority to allocate the radio parameters thereby yielding similar performance results for all priority levels.

5. Conclusion

The massive use of IoT in smart spaces is transforming everything around the world and it is paving the way for the creation of smart cities, industries services 4.0. It is mandatory to improve the use of LoRaWAN with a better channel utilization scheme. This article presented APRA, which is an efficient priority-aware method for distributing configurable ratio parameters (such as SF, TP, and BW) for LoRaWAN. Due to the optimal TP configuration, APRA significantly reduces the total amount of network energy consumption compared to state-of-the-art algorithms and it also significantly increases the battery discharge time of the EDs. Moreover, the proposed resource allocation mechanism exhibited a robust gain in terms of ToA for high and medium priorities EDs, and better results in PDR for all priority groups compared to the other mechanisms. Hence, APRA shows promise as a reliable solution for meeting the QoS requirements of applications while significantly improving LoRaWAN's energy consumption. We plan to make APRA more dynamic in the future. APRA will be able to change the radio parameters of the EDs in case there is any change in the network. With this, the EDs will have more dynamism and packet collisions will decrease.

Declaration of competing interest

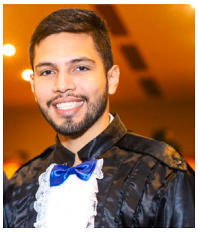
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We thank the anonymous reviewers for their valuable comments which helped us improve the quality, content, and presentation of this paper. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

References

[1] Kais Mekki, Eddy Bajic, Frederic Chaxel, Fernand Meyer, A comparative study of LPWAN technologies for large-scale IoT deployment, ICT Express 5 (1) (2019) 1–7.
[2] Wafa'a Kassab, Khalid A. Darabkh, A–Z survey of Internet of Things: Architectures, protocols, applications, recent advances, future directions and recommendations, J. Netw. Comput. Appl. 163 (2020) 102663.
[3] Patrik Cerwall, P Jonsson, R Möller, S Bävertoft, S Carson, I Godor, Ericsson mobility report, in: On the Pulse of the Networked Society. Hg. V. Ericsson, 2015.
[4] Gianni Pasolini, Chiara Buratti, Luca Feltrin, Flavio Zabini, Cristina De Castro, Roberto Verdone, Oreste Andrisano, Smart city pilot projects using LoRa and IEEE802. 15.4 technologies, Sensors 18 (4) (2018) 1118.
[5] Badis Hammi, Rida Khatoun, Sherali Zeadally, Achraf Fayad, Lyes Khoukhi, Internet of things (IoT) technologies for smart cities, IET Netw. 7 (2017).
[6] Erwin Adi, Adnan Anwar, Zubair Baig, Sherali Zeadally, Machine learning and data analytics for the IoT, Neural Comput. Appl. 32 (2020) 16205–16233.
[7] Jetmir Haxhibeqiri, Eli De Poorter, Ingrid Moerman, Jeroen Hoebeke, A survey of lorawan for iot: From technology to application, Sensors 18 (11) (2018) 3995.
[8] Sherali Zeadally, Faisal Karim Shaikh, Anum Talpur, Quan Z Sheng, Design architectures for energy harvesting in the Internet of Things, Renew. Sustain. Energy Rev. 128 (2020) 109901.
[9] Renato Mota, André Riker, Denis Rosário, Adjusting group communication in dense internet of things networks with heterogeneous energy sources, in: 11th Brazilian Symposium on Ubiquitous and Pervasive Computing, SBC, 2019.
[10] Fei Gu, Jianwei Niu, Landu Jiang, Xue Liu, Mohammed Atiquzzaman, Survey of the low power wide area network technologies, J. Netw. Comput. Appl. 149 (2020) 102459.
[11] IoT Analytics, LPWAN Market report 2018-2023, 2017, https://iot-analytics.com/product/lpwan-market-report-2018-2023/, (Online; accessed 28-May-2020).
[12] Felipe S Dantas Silva, Emidio P Neto, Helder Oliveira, Denis Rosário, Eduardo Cerqueira, Cristiano Both, Sherali Zeadally, Augusto V Neto, A survey on long-range wide-area network technology optimizations, IEEE Access (2021).
[13] Sarath Chandu Gaddam, Mritunjay Kumar Rai, A comparative study on various LPWAN and cellular communication technologies for IoT based smart applications, in: 2018 International Conference on Emerging Trends and Innovations in Engineering and Technological Research, ICETIETR, IEEE, 2018, pp. 1–8.
[14] LoRa Alliance, White Paper: A Technical Overview of Lora and LoRaWAN, The LoRa Alliance, San Ramon, CA, USA, 2015, pp. 7–11.
[15] Luca Sciullo, Angelo Trotta, Marco Di Felice, Design and performance evaluation of a LoRa-based mobile emergency management system (LOCATE), Ad Hoc Netw. 96 (2020) 101993.
[16] Andres Vazquez-Rodas, Fabian Astudillo-Salinas, Christian Sanchez, Braulio Arpi, Luis I Minchala, Experimental evaluation of RSSI-based positioning system with low-cost LoRa devices, Ad Hoc Netw. (2020) 102168.
[17] Jothi Prasanna Shanmuga Sundaram, Wan Du, Zhiwei Zhao, A survey on LoRa networking: Research problems, current solutions and open issues, IEEE Commun. Surv. Tutor. (2019).
[18] Nagib Matni, Jean Moraes, Lucas Pacheco, Denis Rosário, Helder Oliveira, Eduardo Cerqueira, Augusto Neto, Experimenting long range wide area network in an e-health environment: Discussion and future directions, in: Proceedings of the 16th International Wireless Communications Mobile Computing Conference, IWCMC 2020, Limassol, Cyprus, 2020.
[19] Orestis Georgiou, Usman Raza, Low power wide area network analysis: Can LoRa scale? IEEE Wirel. Commun. Lett. 6 (2) (2017) 162–165.
[20] Rachel Kufakunesu, Gerhard P. Hancke, Adnan M. Abu-Mahfouz, A survey on adaptive data rate optimization in LoRaWAN: Recent solutions and major challenges, Sensors 20 (18) (2020) 5044.
[21] Ramon Sanchez-Iborra, Jesus Sanchez-Gomez, Juan Ballesta-Viñas, Maria-Dolores Cano, Antonio F Skarmeta, Performance evaluation of LoRa considering scenario conditions, Sensors 18 (3) (2018) 772.
[22] Martin C. Bor, Utz Roedig, Thiemo Voigt, Juan 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, 2016, pp. 59–67.
[23] Samir Dawaliby, Abbas Bradai, Yannis Pousset, Adaptive dynamic network slicing in LoRa networks, Future Gener. Comput. Syst. 98 (2019) 697–707.
[24] Khaled Q. Abdelfadeel, Victor Cionca, Dirk Pesch, Fair adaptive data rate allocation and power control in lorawan, in: 2018 IEEE 19th International Symposium on" a World of Wireless, Mobile and Multimedia Networks", WoWMoM, IEEE, 2018, pp. 14–15.
[25] Mohammed Alenezi, Kok Keong Chai, Shihab Jimaa, Yue Chen, Use of unsupervised learning clustering algorithm to reduce collisions and delay within LoRa system for dense applications, in: 2019 International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob, IEEE, 2019, pp. 1–5.
[26] Francesca Cuomo, Manuel Campo, Alberto Caponi, Giuseppe Bianchi, Giampaolo Rossini, Patrizio Pisani, EXPLoRa: Extending the performance of LoRa by suitable spreading factor allocations, in: 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob, IEEE, 2017, pp. 1–8.
[27] Minar El-Aasser, Tallal Elshabrawy, Mohamed Ashour, Joint spreading factor and coding rate assignment in LoRaWAN networks, in: Global Conference on Internet of Things, GCIoT, IEEE, 2018, pp. 1–7.
[28] D. Zorbas, G. Z. Papadopoulos, P. Maille, N. Montavont, C. Douligeris, Improving LoRa network capacity using multiple spreading factor configurations, in: Proceedings of the 25th International Conference on Telecommunications, ICT 2018, 2018, pp. 516–520.
[29] Jean Moraes, Nagib Matni, Andre Riker, Helder Oliveira, Eduardo Cerqueira, Cristiano Both, Denis Rosário, An efficient heuristic LoRaWAN adaptive resource allocation for IoT applications, in: 25th IEEE Symposium on Computers and Communications, ISCC, IEEE, 2020, pp. 1–6.
[30] J. Babaki, M. Rasti, R. Aslani, Dynamic spreading factor and power allocation of LoRa networks for dense IoT deployments, in: 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, 2020, pp. 1–6.
[31] Nagib Matni, Jean Moraes, Denis Rosário, Eduardo Cerqueira, Augusto Neto, Optimal gateway placement based on fuzzy C-means for low power wide area networks, in: Proceedings of the IEEE Latin-American Conference on Communications, LATINCOM, IEEE, 2019, pp. 1–6.
[32] Nagib Matni, Jean Moraes, Helder Oliveira, Denis Rosário, Eduardo Cerqueira, LoRaWAN gateway placement model for dynamic internet of things scenarios, Sensors 20 (15) (2020) 4336.
[33] Andressa Vergutz, Iago Medeiros, Denis Rosario, Eduardo Cerqueira, Aldri Santos, Michele Nogueira, A method for identifying ehealth applications using side-channel information, in: Proceedings of the IEEE Global Communications Conference, GLOBECOM, IEEE, 2019, pp. 1–6.
[34] Qahhar Muhammad Qadir, Tarik A Rashid, Nawzad K Al-Salihi, Birzo Ismael, Alexander A Kist, Zhongwei Zhang, Low power wide area networks: A survey of enabling technologies, applications and interoperability needs, IEEE Access (2018).
[35] Mehmet Ali Ertürk, Muhammed Ali Aydın, Muhammet Talha Büyükakkaşlar, Hayrettin Evirgen, A survey on LoRaWAN architecture, protocol and technologies, Future Internet 11 (10) (2019) 216.
[36] Andrzej Duda, Martin Heusse, Spatial issues in modeling LoRaWAN capacity, in: ACM MSWiM, 2019, pp. 191–198.
[37] LoRa Alliance, LoRaWAN 1.1 regional parameters, in: Technical Specification, 2017.
[38] LoRa SX1276, 77/78/79 datasheet, 2019, Online: https://www.Semtech.Com/Images/Datasheet/Sx1276.Pdf.
[39] Eugen Harinda, et al., Comparative performance analysis of empirical propagation models for LoRaWAN 868MHz in an urban scenario, in: 5th World Forum on Internet of Things, WF-IoT, 2019, pp. 154–159.
[40] Eduardo Sallum, Nuno Pereira, Mário Alves, Max Santos, Improving quality-of-service in LoRa low-power wide-area networks through optimized radio resource management, J. Sensor Actuator Netw. 9 (1) (2020) 10.
[41] Adaptive data rate, in: The Things Network, 2019, https://www.thethingsnetwork.org/docs/lorawan/adaptive-data-rate.html.



Eduardo Lima graduated in Computer Engineering at the Federal University of Para, Brazil. Currently, he is doing a Master degree in Computer Science at the Federal University of Para, Brazil. His current research interests include the following: Smart Cities, Internet of Things, Low-Power Wide-Area Network, and Sensor Networks.

Jean Moraes Currently, he is doing Computer Engineering at the Federal University of Para, Brazil. His current research interests include the following: Smart Cities, Internet of Things, Low-Power Wide-Area Network, and Sensor Networks.

Helder Oliveira is an adjunct professor at the Institute of Exact and Natural Sciences (ICEN) at UFPA. He holds a Ph.D. in Computer Science from the Institute of Computing at the State University of Campinas (2018). Helder also did a post-doctoral internship in Computer Science at the Institute of Computing at the State University of Campinas (2019). His main topics of interest include the internet of things, computer networks, protection, fault tolerance, and data analysis.

Eduardo Cerqueira received his Ph.D. in Informatics Engineering from the University of Coimbra, Portugal (2008). He is an associate professor at the Faculty of Computer Engineering of the Federal University of Para (UFPA) in Brazil, as well as invited researcher at the Network Research Lab at UCLA-USA and Centre for Informatics and Systems of the University of Coimbra-Portugal. His publications include 5 edited books, 5 book chapters, 4 patents and over than 180 papers in national/international refereed journals/conferences. He has been serving as a Guest Editor for 6 special issues of various peer-reviewed scholarly journals. His research involves Multimedia, Future Internet, Quality of Experience, Mobility and Ubiquitous Computing.

Sherali Zeadally earned his bachelor's degree in computer science from the University of Cambridge, England. He also received a doctoral degree in computer science from the University of Buckingham, England. He is currently an Associate Professor in the College of Communication and Information, University of Kentucky. His research interests include Cybersecurity, privacy, Internet of Things, computer networks, and energy-efficient networking. He is a Fellow of the British Computer Society and the Institution of Engineering Technology, England.

Denis Rosário received his Ph.D. degree in Electrical Engineering at the Federal University of Para, Brazil with joint supervision undertaken by the Institute of Computer Science and Applied Mathematics of University of Bern, Switzerland in 2014. Currently, he is a Professor at Federal University of Para. His current research interests include the following topics: Multimedia, Wireless Networks, FANET, VANET, Mobility, Quality of Experience, and Software Defined Network.