Packet Delivery Ratio Guarantees for Differentiated LoRaWAN Services

Alessandro Aimi*†, Fabrice Guillemin*, Stéphane Rovedakis† and Stefano Secci†
*Orange Labs, France. {alessandro.aimi, fabrice.guillemin}@orange.com
†Cedric, Cnam, France. {alessandro.aimi, stephane.rovedakis, stefano.secci}@cnam.fr

***Abstract*—Motivated by the rapid deployment of applications based on connected objects, we propose in this paper an approach to differentiating Packet Delivery Ratios (PDRs) in LoRa Wide Area Networks (LoRaWAN). This type of network is simple to deploy and operate at the expense of loose commitments in terms of quality. To overcome this shortcoming, we propose an access control method for isolating clusters of devices and meeting differentiated PDR targets. Results show that our method outperforms known one in terms of PDR via improved parameter allocation and achieves high level of intra-cluster fairness, at the expense however of decreasing the maximum cell range.**

***Index Terms*—LoRaWAN, Access Control, Delivery Ratio.**

I. Introduction

Connected objects are today rapidly proliferating for diverse applications (e.g., smart cities, eHealth, agriculture, etc.). Among the many existing Internet of Things (IoT) environments, Low Power Wide Area Networks (LPWANs) targets long range sensing and monitoring applications. LPWAN technologies such as Long Range (LoRa) and NB-IoT gained a significant market share worldwide. In particular, the Long Range Wide Area Network (LoRaWAN) technology is becoming very popular as it is relatively cheap and easy to deploy and to operate. In LoRaWAN, devices generate a light load of sparse or periodical traffic, which rarely exceeds 50 bytes per payload, with a periodicity lower than 10' [1].

The inevitable densification of LoRaWAN deployments, boosted by the proliferation of connected objects, has drawn the attention of researchers toward technologies to guarantee reliability, quality and scalability levels. One of the design choices in LoRaWAN is the uncoordinated access to the radio medium [2]. This is aimed at minimizing the power consumption of intermittently active devices by simplifying the communication protocol stack. As a consequence, collisions are frequent in dense deployments and result in a best effort operation and an inherently inefficient utilization of resources [3]. A key performance metric for an application using LoRaWAN is the Packet Delivery Ratio (PDR) [4].

To mitigate this problem, the LoRaWAN protocol offers primitives to dynamically manage transmission parameters. Many techniques have been proposed to exploit these primitives and achieve a better radio resource utilization [5], [6]. In particular, a growing body of literature has shown interest in the integration of the network slicing paradigm in LoRaWAN [7]–[15], with a focus on the optimization of differentiated radio traffic (reliability, throughput, latency, and energy consumption). These approaches assign cluster of devices to independent interference domains, i.e. disjoint sets of frequency channels, to enforce traffic isolation. Moreover, they use LoRaWAN primitives to optimize transmission parameters according to the different requirements of clusters.

Despite optimizations, quality does degrade when a high number of devices transmit in the vicinity of a radio gateway [16], [17]. Resource allocation algorithms currently proposed do not consider limiting device access and/or traffic to prevent performance degradation. Such actions may be in contradiction with the openness and best effort principles of LoRaWAN, but seem to be inevitable in light of the expected densification of connected objects in quality demanding applications.

In this paper, we propose a radio resource management scheme to integrate quality differentiation in LoRaWAN in dense scenarios. For this purpose, we propose an approach to exploit the isolation provided by the available radio frequencies to group devices into clusters with different PDR targets, and to optimize behavior configuration in order to reach the desired PDR objectives. With respect to existing propositions, we introduce two new contributions: (1) a technique to include PDR requirements in channel assignment, and (2) an access control policy to accordingly limit traffic. Moreover, we evaluate two variants of the approach at different levels of isolation between clusters. Simulation results of our proposal show that PDR-differentiation is fairly achieved as a trade-off between device density and maximum cell range.

The paper is organized as follows. Section II recalls key elements of LoRaWAN and positions our proposition with the state of the art. Section III presents our PDR differentiation framework. Section IV describes the numerical evaluation. Concluding remarks are presented in Section V.

II. Related Work and Contributions

LoRaWAN uses spread-spectrum modulation on the frequency channels in the unlicensed industrial, scientific, and medical (ISM) bands. Devices are assigned to a common pool of frequency channels and they must randomly select one for each new transmission. This behaviour is referred to as *Frequency Hopping* and is introduced to mitigate the impact of external interference [2]. Uplink transmissions are received by LoRaWAN gateways in range and forwarded to a network server. In turn, the server sends downlink configuration frames through the gateway measuring the best reception.

Six Spreading Factor (SF) configurations that range from SF7 to SF12 are used, as specified in Table I. The SF is a modulation parameter that controls the spreading of symbols at modulation time. With a higher SFs, transmission range is increased at the expense of longer transmission time on the air interface. Moreover, transmissions that are on the same SF and on the same frequency channel collide, while different SFs are almost orthogonal [18]. In addition, Transmission Power (TP) can be configured to improve capacity as in cellular networks.

TABLE I
LoRa transmission data rates [19], [20].

Data-Rate	Configuration	Transmission duration of a packet with a 51B payload [ms]
0	SF12 / 125 kHz	2793.5 (0.3 kb/s)
1	SF11 / 125 kHz	1560.6 (0.5 kb/s)
2	SF10 / 125 kHz	698.4 (1.0 kb/s)
3	SF9 / 125 kHz	390.1 (1.8 kb/s)
4	SF8 / 125 kHz	215.6 (3.1 kb/s)
5	SF7 / 125 kHz	118.0 (5.5 kb/s)

A 1% duty cycle limitation is introduced to maintain fairness in the ISM band [21] and can be additionally lowered by using protocol primitives. If the duration of packet transmission is τ time units, then the transmission starting times of two consecutive packets must be separated by $\frac{\tau}{\delta}$ time units, where δ is the duty cycle [2].

To enable collision detection, the LoRaWAN standard provides the option of using 'confirmed' (acknowledged) traffic. Precisely, a downlink acknowledgment is sent back to the devices after every uplink transmission, hence possibly triggering re-transmissions. However, as this can halve the network capacity, it is considered viable only in small networks [22].

In dense deployments, the increase of network traffic leads to a high number of collisions. This can be mitigated by distributing traffic over multiple SFs and regulating TP. Many propositions exist to tackle this problem, improving network scalability, throughput or energy consumption [6]. Existing implementations minimize the SF of each device according to the measured Signal to Noise Ratio (SNR) of transmissions [5]. TP is lowered only if a device is already using SF7.

A. *Quality Differentiation and Slicing*

In [7]–[15] the authors propose to introduce network slicing in LoRaWAN. They provide a high level view of the end-to-end architecture and put more attention on the radio parameters assignment problem. Specifically, the assignment of independent interference domains allow them to achieve differentiated traffic quality. These works follow a common methodology consisting of two steps as follows.

a) Channel Allocation: In each gateway, a portion of the available frequencies is reserved for each cluster. In [7] this is done proportionally to the average device throughput of each cluster. In [8], [11] the methodology is refined by considering, for each gateway, only nodes in range instead of the total number of devices for the average device throughput calculation. In [13] this method is compared with an approach based on mini-batch Gradient Descent to obtain the portion of frequencies for each cluster. In [9] Bankruptcy Games and Matching Theory are used for frequency allocation.

b) SF and TP Allocation: An assignment of SFs and TP to devices is achieved for the different clusters, according to their needs. In [9], [11] this task is done with a Multi-Criteria Decision Analysis approach proposed in [10]. In [7], [8] a further step is specified to find for each device the best path to a gateway. In [12], this problem is addressed via a Transfer learning-based Multi-agent Deep Deterministic Policy Gradient algorithm, and in [14] via Deep Federated Reinforced Learning, and in [15] via Deep Reinforced Learning.

For cluster requirements in terms of quality, authors adopt those originally defined for Machine Type Communications (MTC) in 5G networks, categorized in terms of ultra-high, high, and low requirements. Latency and reliability bounds are used to guide the optimization objectives of each cluster. These latency (hundreds of ms or less) and quality (0.999 or more PDR) objectives may be difficult to meet in LoRaWAN, since this technology has very specific bandwidth, transmission speeds and medium access control. Such objectives are actually not met in the numerical and experimental results reported in [3], [4], [18], [23], [24].

B. *Contributions*

We propose a PDR differentiation method via frequency allocation that differs from previous works as follows:

- The share of radio resources, i.e., the number of channels, assigned to each cluster is scaled according to the PDR requirement. In [9], [13], this share is only based on total cluster throughput.
- We limit the number of devices using each SF in a cluster according to the number of channels assigned. This is important to allow more precise control over the satisfaction of requirements, and to address network congestion if the population of devices around a gateway increases. For example, this could happen in a Smart City scenario due to bicycle tracking. To the best of our knowledge, in LoRaWAN literature this has not been considered before.

Moreover, in this paper we define classes of services in terms of PDR. Latency is a secondary metric because LoRaWAN use cases are not time-critical [23], and throughput is closely related to PDR due to duty-cycle and packet size constraints [19]. Energy considerations are out of the scope of this paper and will be addressed in further studies. We still adopt an ultra-high, high, and low categorization as in 5G MTCs for the Industrial IoT [13] but change the requirements to 0.97 PDR, 0.90 PDR, and 0.70 PDR based on the experimental results in [24]. We consider cluster membership as predetermined, based on the cluster PDR-requirements.

III. LoRaWAN Packet Delivery Differentiation

Since our proposal aims at differentiating the PDR of devices according to their cluster membership, we first present an estimation of the PDR followed by the algorithmic framework to assign radio resources and transmission parameters.

A. Offered Traffic Based on PDR Estimation

We rely on the capacity model in [17]. The average number of concurrent transmissions on a channel is assessed via the offered traffic defined as $\nu = \tau\lambda$, where τ is the average transmission duration, λ is the arrival rate of frames. From the perspective of a frame arriving at a gateway, the probability of meeting k additional overlapping transmissions is

$$P(X = k) = \frac{(2\nu)^k e^{-2\nu}}{k!}, \tag{1}$$

as there should be no other transmission events for a time 2τ to avoid overlap. The PDR is then derived as the sum of the probability of no collisions ($k = 0$), and the probability of having a non destructive collision ($k = 1$) as

$$PDR = e^{-2\nu} + \frac{2\nu}{\gamma+1} e^{-2\nu} \stackrel{\text{def}}{=} h(\nu), \tag{2}$$

where $\gamma = 6$dB is the difference in received power necessary to capture one of two overlapping transmissions on a SF [17], and $(\gamma+1)^{-1}$ is the probability that an exponential random variable is γ times greater than another one.

From Eq. (2), we get the estimate ν of total offered traffic on a frequency and SF to respect the desired PDR. As the result is the same for each SF, the maximum traffic supported by a gateway can eventually be estimated as 6ν multiplied by number of frequencies used. By inverting Eq. (2) we obtain

$$h^{-1}(PDR) = -\frac{1}{2} \cdot \mathcal{W}_{-1}\left(-\frac{\xi}{e^{\xi}} \cdot PDR\right) - \frac{\xi}{2} = \nu, \tag{3}$$

where $\mathcal{W}_{-1}$ is the lower branch of the Lambert W function and $\xi = \gamma + 1$.

The duty cycle sets an upper bound for the offered traffic ν_d of a single device. In real LoRaWAN scenarios, devices usually transmit much less than 1% of time [1]. Thus, supposing to know a priori the maximum throughput t_d needed by a device d, we get a better offered traffic estimate with

$$\nu_d = \frac{p}{dr_{SF}}\lambda = \frac{t_d}{dr_{SF}}, \tag{4}$$

where p is the packet length and dr_{SF} is the data-rate of the SF to be used by the device. We use this estimate for the resource allocation of the proposed technique.

B. Parameter Allocation Technique

Our approach is based on four algorithmic steps to ensure that PDR targets are closely met for clusters by means of device configuration. We design our procedure to be re-executed whenever the network detects a significant change in device population around a gateway.

We assume that devices transmit periodically, with complete heterogeneity in terms of inter-transmission period and packet size. As a prerequisite, each device is assigned to a cluster and has to provide the network operator with its maximum planned throughput (bit/s). We further assume that each device is positioned in the radio range of a gateway and that we already cold started devices collect SNR data.

Let D be the set of devices; $d \in D$ has declared throughput t_d and belongs to cluster c_d. Clusters are defined in terms of a minimum PDR. The set of clusters is denoted by C, cluster c has required PDR noted PDR_c; G is the set of gateways and F the set of frequencies. See Table II for a definition of the various parameters.

TABLE II
LIST OF PARAMETERS.

Symbol	Description
ν	Offered traffic
τ	Frame transmission duration
λ	Frame arrival rate
$h^{-1}(PDR)$	Capacity of a SF on a frequency at desired PDR
d, D	Device, set of devices
c, C	Cluster, set of clusters
g, G	Gateway, set of gateways
F	Set of frequencies
$SNR_{g,d}$	SNR measured from device d at gateway g
c_d	Cluster of device d
t_d	Throughput needed by device d
PDR_c	PDR requirement of cluster c
$D_{g,c}$	Subset of devices of cluster c assigned to gateway g
$w_{g,c}$	Resource demand of cluster c at gateway g
$w'_{g,c}$	Ideal bandwidth share of cluster c at gateway g
$\overline{w}_{g,c}$	Number of frequencies assigned to cluster c at gateway g
$F_{g,c}$	Subset of frequencies assigned to cluster c at gateway g
dr_{SF}	Data-rate of SF

The four main steps of our approach are detailed hereafter.

1) Step 1: Device Grouping: We subdivide devices according to the gateway g measuring the highest SNR. Then, each device is assigned to a set $D_{g,c}$ according to the cluster c_d they belong to: it so minimizes the transmission parameters of devices to reduce interference at farther gateways.

2) Step 2: Computation of Spectrum Shares: For each (g, c) gateway-cluster pair, we compute a weight $w_{g,c}$ and we use it to determine the share of the total number of frequencies to be allocated to cluster c at gateway g. The weight $w_{g,c}$ is proportional to the total resource demands of devices in each cluster. Then, if the number of devices accessing the network exceeds the capacity, we exclude a uniform percentage of them between the clusters.

Devices at the highest levels of quality need more resources to maintain a low number of collisions. Hence, we determine the transmission bit-rate needed to achieve the declared throughput t_d while respecting the offered traffic constraint obtained with the capacity model of Equation (3) at the desired PDR_c. Then, we sum them up so that

$$w_{g,c} = \sum_{d \in D_{g,c}} \frac{t_d}{h^{-1}(PDR_c)} \quad \forall g \in G, \forall c \in C. \tag{5}$$

Finally, the shares $w_{g,c}$ are proportionally normalized to sum up to $|F|$, the total number of frequencies,

$$w'_{g,c} = |F| \cdot \frac{w_{g,c}}{\sum_{c' \in C} w_{g,c'}} \quad \forall g \in G, \forall c \in C. \tag{6}$$

3) Step 3: Frequency Allocation: Because the quantity $w'_{g,c}$ is real and possibly less than one, this value cannot be directly used as the number of frequencies to allocate to cluster c at gateway g, which has to be an integer larger than or equal to 1. To overcome this problem, we propose the two following methods of fixing the set $F_{g,c}$ of frequencies such that $|F_{g,c}| = \overline{w}_{g,c}$ and $\bigcup_{c \in C} F_{g,c} = F$.

a) Hard Isolation: We do not mix devices of different clusters in a same frequency. The number of frequencies is obtained by minimizing the sum of the gaps between the continuous frequency share values $w^{'}_{g,c}$ computed in the previous step, and the final discrete numbers of frequencies $\overline{w}_{g,c}$ to be assigned to each cluster in a gateway:

$$\min \sum_{c \in C} |w^{'}_{g,c} - \overline{w}_{g,c}| \quad \forall g \in G \tag{7}$$

Exact solutions are obtained in $O(|G| \cdot |C| \log(|C|))$ complexity with a trivial greedy algorithm that (i) reserves $\lfloor w^{'}_{g,c} \rfloor$ to each cluster, and (ii) assigns the remaining frequencies one at a time to the clusters ordered by highest decimal part of $w^{'}_{g,c}$.

b) Soft Isolation: In this policy we exploit the fact that we can upgrade devices to higher PDR clusters without breaking the requirements. In each gateway, starting from the highest PDR cluster c we allocate $\overline{w}_{g,c} = \lceil w^{'}_{g,c} \rceil$, and we randomly choose devices to be upgraded (without changing their cluster membership) from the next lower cluster, so to fill the $\overline{w}_{g,c} - w^{'}_{g,c}$ difference. In line with Equation (5), each new device d occupies an amount equal to $t_d/h^{-1}(PDR_c)$. This procedure is repeated for all following clusters in terms of PDR until we have frequencies to allocate.

4) Step 4: Assignment of Frequencies and SF to Devices: We assign all frequencies in $F_{g,c}$ to each device in cluster c of gateway g. We maintain frequency hopping to mitigate the impact of external interference, and TP minimization in SF7 as in existing approaches [5].

We progressively assign a SF to each device by allocating higher SFs to devices $D_{g,c}$ ordered by descending SNR. Based on the model of Equation (3), each SF in a frequency of cluster c has a maximum offered traffic equal to $\nu = h^{-1}(PDR_c)$. So, starting from the lowest SF and the device $d \in D_{g,c}$ with the highest SNR, we allocate the SF to d and subtract from ν a quantity equal to the offered traffic share $\nu_d = t_d/dr_{SF}$ of d. If the capacity of the last SF is depleted we need to avoid more devices around a gateway (e.g., by limiting the duty cycle to an extremely small value using MAC primitives).

A summary of the overall approach is given in Algorithm 1. If we consider a uniform density of nodes/km^2, we obtain an estimate of the gateway range based on the last device served. Hence, if path loss data of an area is available, the technique can be used beforehand by the operator to dimension the number of cells in the network based on maximum device density and offered levels of PDR.

Algorithm 1 LoRaWAN Clusters Configuration

Inputs: D, C, G, F, $SNR_{g,d}$, c_d, t_d, PDR_c, `hard` (flag)
Use $SNR_{g,d}$ and c_d to insert d in $D_{g,c}$ $\quad \forall d \in D$
for all $g \in G$ **do**
  Compute cluster shares $w^{'}_{g,c}$ $\quad \forall c \in C$
  if `hard` **then**
    Hard Isolation to get the number of frequencies $\overline{w}_{g,c}$
  else
    Soft Isolation to get the number of frequencies $\overline{w}_{g,c}$
  end if
  Reserve $F_{g,c}$ frequencies based on $\overline{w}_{g,c}$ $\quad \forall c \in C$
  for all $c \in C$ **do**
    Order $D_{g,c}$ by descending $SNR_{g,d}$
    Assign all $F_{g,c}$ frequencies to each $d \in D_{g,c}$
    Assign an SF to each $d \in D_{g,c}$
  end for
end for

IV. Numerical Results

To evaluate our proposed scheme, we have developed a lightweight simulator for the LoRa uplink traffic physical layer in the Ns-3 simulation environment. Interference and path loss computations follow the state of the art model from [16].

A. Simulation Setup

Seven gateways are placed using hexagonal tiling as in Figure 1, where circle radius is 7.5km. Devices are uniformly placed in range of gateways, and they transmit with a periodical traffic pattern, interfering with other devices both in the same and other cells. To simulate heterogeneity, the inter-transmission time and payload of each device are extracted from a truncated Gaussian random variable with mean 600s and variance 300s, and with mean 31B (13B for headers) and variance 10B, respectively [1]. The transmission parameters of devices are listed in Table III.

TABLE III
End devices transmission parameters.

Parameter	Value(s)
Antenna ERP power (dBm)	14, 12, 10, 8, 6, 4, 2, 0
Frequency (MHz)	868.1, 868.3, 868.5, 867.1, 867.3, 867.5, 867.7, 867.9
Spreading Factor	7-12
Bandwidth (kHz)	125
Coding rate	4/5
Preamble length	8
Explicit header	Disabled
CRC	Enabled
Low data rate optimization	Enabled (SF11/12)

Packet transmission time is computed following the SX1272/3/6/7/8 LoRa Modem Design Guide [20]. Levels of gateway to SF Sensitivity are in Table IV. In interference calculations, we adopt the empirical Signal-to-Interference thresholds matrix presented in [25]. We simulate the network running for 10 h and we replicate simulations 30 times to be able to draw figures with 95% confidence intervals.

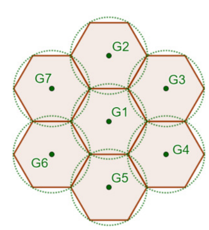
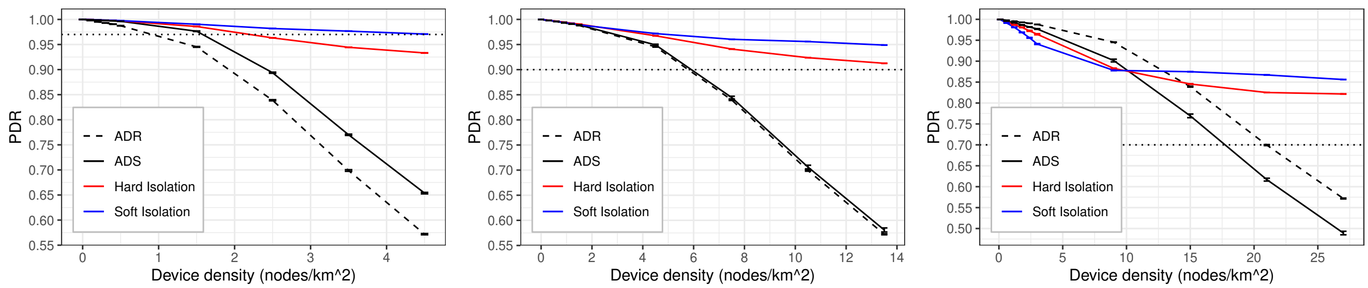


Fig. 1. Placement of gateways and devices in simulations.

TABLE IV
Sensitivity levels (dBm) required for correct packet reception on the different SFs [26]

SF7	SF8	SF9
-126.5	-129.0	-131.5
SF10	SF11	SF12
-134.0	-136.5	-139.5



(a) 0.97 PDR cluster. (b) 0.90 PDR cluster. (c) 0.70 PDR cluster.

Fig. 2. PDR performance of devices in different clusters. The device densities for each plot is related to the number of devices of the associated cluster.

B. Results Analysis

In the experiments we evaluate the Hard and Soft Isolation policies for frequency assignment proposed in Section III against increasingly high network densities, up to 45 devices/km^2 based on the scalability study in [18] (<0.65 PDR after 600 nodes per channel). We use three clusters with the PDR requirements (0.97, 0.90 and 0.70) motivated in Section II. Devices assignment to clusters (10%, 30% and 60%, respectively) follows the percentages proposed in [8].

For comparison, we implement the technique for channel allocation proposed in [8], called Adaptive Dynamic Slicing (ADS). In terms of PDR optimization, ADS has proven to be comparable to the other propositions [9], [13]. As a baseline, we also implement the classical Adaptive Data Rate (ADR) configuration currently used in LoRaWANs [5].

1) PDR Compliance of Clusters: We measure the PDR of traffic for each cluster over the density of devices they serve. The density range of each cluster reflects the percentage of input devices assigned to them.

Measurements for the 0.97 PDR cluster are displayed in Figure 2(a). Soft Isolation shows better performance and it is able to comply with the PDR requirement up to the maximum density. Hard isolation goes under 0.97 PDR after 2 nodes/km^2. ADS is able to outperform ADR but falls under 0.97 PDR after 1.5 nodes/km^2.

Measurements for the 0.90 PDR cluster are displayed in Figure 2(b). Both Hard and Soft Isolation remain compliant with the PDR objective, Soft Isolation shows a better performance. ADS and ADR show comparable degradation and they both go under 0.9 PDR after 6 nodes/km^2.

Measurements for the 0.70 PDR cluster are displayed in Figure 2(c). Hard and Soft Isolation remain over 0.82 PDR. From this we understand that our technique over-assigns resources at lower PDR levels. ADR and ADS fall under the 0.7 threshold after 21 nodes/km^2 and 17.5 nodes/km^2.

From the results we conclude that Soft Isolation is the technique obtaining best performances, by being able to maintain differentiated PDR levels in scenarios with the same network density of current state-of-the-art scalability estimates. Comparison with state-of-the-art propositions for traffic quality differentiation show the gain in terms of requirements satisfaction that can be achieved by introducing access control.

2) Network Throughput and Gateway Range: We measure the throughput of the network and the range of the gateways against the density of devices served by the network. The range is measured as the maximum distance from the closest gateway among served devices. In our technique the range is reduced by access control. We compare to ADR and ADS, where the range limit is a consequence of path loss. Results are displayed in Figure 3 and in Figure 4 for throughput and range respectively.

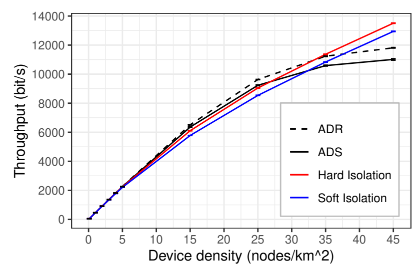


Fig. 3. Throughput of the network under different loads.

Throughput results show that access control is able to balance out packet loss caused by higher interference in ADR and ADS, especially at high density (>40 nodes/km^2). By comparing ADS and our technique to ADR, we see that traffic differentiation generally comes at the cost of slightly lower throughput values. Also, higher precision in PDR control as in Soft Isolation results in lower throughput with respect to Hard Isolation, which is instead comparable to ADS. Figure 4 highlights the trade-off of the proposed technique: the capacity-based policy reduces the range in which devices can be served when the density increases. Overall Soft Isolation is more expensive in terms of range, losing almost 1300m against ADR and ADS. As a consequence, in large and dense deployments PDR differentiation comes at the expense of an higher number of gateways than best effort operation.

3) Intra-cluster Fairness: Let us now evaluate how well the PDR of clusters is distributed between their devices; in fact, even if the average PDR in a cluster is over the required level, the PDR distribution over the devices in the same cluster may still be unbalanced. We produce the Jain's Fairness Index for the devices PDR in each cluster, a common metric to evaluate whether participants are receiving a fair share of system resources. Results are displayed in Figure 5.

We can see that the level of fairness is related to the number

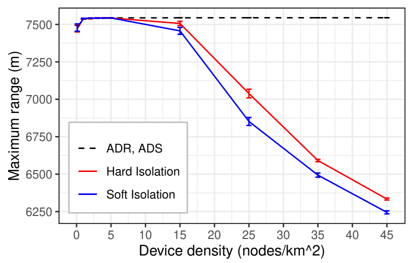
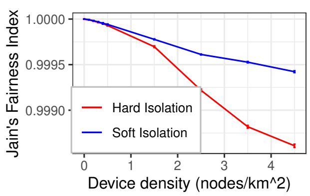
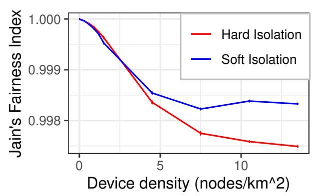


Fig. 4. Maximum gateway range under different loads.

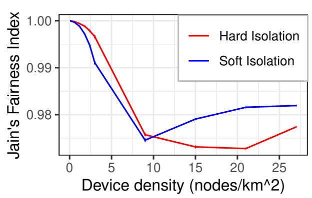
of served devices by a cluster. This results in high PDR clusters achieving near perfect fairness and the best effort cluster being less precise. Soft Isolation is more fair and the overall fairness does not fall under 0.97 which means our technique is highly fair among single devices in a cluster.



(a) 0.97 PDR cluster



(b) 0.90 PDR cluster



(c) 0.70 PDR cluster

Fig. 5. Inter-device fairness of the PDR in each cluster computed with the Jain's Fairness Index (JFI). Maximum fairness is 1 and minimum is $1/n$, with n being the number of participants.

V. Conclusion and Future Work

We have proposed an approach to differentiate the PDRs of LoRaWAN device clusters under dense settings by creating independent interference domains and limiting the number of transmitting devices with a form of access control. Our results show that the proposed approach is able of achieving PDR differentiation in high density scenarios identified by state-of-the-art scalability studies. The technique performs better than common heuristics for parameter allocation, at the expense however of decreasing the maximum cell range. Finally, the approach presents high fairness levels in requirements satisfaction between devices. Future work may consider online execution, raising the limit to the maximum number of clusters, energy related QoS requirements and automatic estimation of the channel capacity.

Acknowledgment

This work was funded by the French ANR INTELLIGENTSIA project (grant nb: ANR-20-CE25-0011).

References

[1] N. Blenn and F. Kuipers, "LoRaWAN in the wild: Measurements from the things network," *arXiv preprint arXiv:1706.03086*, 2017.
[2] "LoRaWAN L2 1.0.4 specification (TS001-1.0.4)," LoRa Alliance, 2020.
[3] O. Georgiou and U. Raza, "Low power wide area network analysis: Can LoRa scale?" *IEEE Wireless Communications Letters*, vol. 6, no. 2, pp. 162–165, 2017.
[4] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "Do LoRa low-power wide-area networks scale?" in *ACM MSWiM*, 2016.
[5] M. Slabicki, G. Premsankar, and M. Di Francesco, "Adaptive configuration of LoRa networks for dense IoT deployments," in *IEEE NOMS*, 2018.
[6] R. Kufakunesu, G. P. Hancke, and A. M. Abu-Mahfouz, "A survey on adaptive data rate optimization in LoRaWAN: Recent solutions and major challenges," *Sensors*, vol. 20, no. 18, p. 5044, 2020.
[7] S. Dawaliby, A. Bradai, Y. Pousset, and R. Riggio, "Dynamic network slicing for LoRaWAN," in *IEEE CNSM*, 2018.
[8] S. Dawaliby, A. Bradai, and Y. Pousset, "Adaptive dynamic network slicing in LoRa networks," *Future Generation Computer Systems*, vol. 98, pp. 697–707, 2019.
[9] S. Dawaliby, A. Bradai, and Y. Pousset , "Distributed network slicing in large scale IoT based on coalitional multi-game theory," *IEEE Transactions on Network and Service Management*, vol. 16, no. 4, pp. 1567–1580, 2019.
[10] S. Dawaliby, A. Bradai, and Y. Pousset, "Network slicing optimization in large scale LoRa wide area networks," in *IEEE NetSoft*, 2019.
[11] S. Dawaliby, A. Bradai, and Y. Pousset , "Joint slice-based spreading factor and transmission power optimization in LoRa smart city networks," *Internet of Things*, vol. 14, p. 100121, 2021.
[12] T. Mai, H. Yao, N. Zhang, W. He, D. Guo, and M. Guizani, "Transfer reinforcement learning aided distributed network slicing resource optimization in industrial IoT," *IEEE Transactions on Industrial Informatics*, 2021.
[13] S. Messaoud, S. Dawaliby, A. Bradai, and M. Atri, "In-depth performance evaluation of network slicing strategies in large scale industry 4.0," in *IEEE SSD*, 2021.
[14] S. Messaoud, S. Bouaafia, A. Bradai, M. A. Hajjaji, A. Mtibaa, and M. Atri, "Network slicing for industrial IoT and industrial wireless sensor network: Deep federated learning approach and its implementation challenges," in *Emerging Trends in Wireless Sensor Networks*. IntechOpen, 2022.
[15] A. Tellache, A. Mekrache, A. Bradai, R. Boussaha, and Y. Pousset, "Deep reinforcement learning based resource allocation in dense sliced LoRaWAN networks," in *IEEE ICCE*, 2022.
[16] D. Magrin, M. Centenaro, and L. Vangelista, "Performance evaluation of LoRa networks in a smart city scenario," in *IEEE ICC*, 2017.
[17] M. Heusse, T. Attia, C. Caillouet, F. Rousseau, and A. Duda, "Capacity of a LoRaWAN cell," in *ACM MSWiM*, 2020.
[18] A. Mahmood, E. Sisinni, L. Guntupalli, R. Rondón, S. A. Hassan, and M. Gidlund, "Scalability analysis of a LoRa network under imperfect orthogonality," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1425–1436, 2018.
[19] "RP002-1.0.3 LoRaWAN Regional Parameters," LoRa Alliance, 2021.
[20] "SX1272/3/6/7/8 design guide, AN1200.13, V1," Semtech, 2013.
[21] "ETSI EN 300 220-2 V3.2.1," ETSI, 2018.
[22] J. M. Marais, A. M. Abu-Mahfouz, and G. P. Hancke, "A survey on the viability of confirmed traffic in a LoRaWAN," *IEEE Access*, vol. 8, pp. 9296–9311, 2020.
[23] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of LoRaWAN," *IEEE Communications magazine*, vol. 55, no. 9, pp. 34–40, 2017.
[24] J. Petäjäjärvi, K. Mikhaylov, M. Pettissalo, J. Janhunen, and J. Iinatti, "Performance of a low-power wide-area network based on LoRa technology: Doppler robustness, scalability, and coverage," *International Journal of Distributed Sensor Networks*, vol. 13, no. 3, p. 1550147717699412, 2017.
[25] D. Croce, M. Gucciardo, S. Mangione, G. Santaromita, and I. Tinnirello, "Impact of LoRa imperfect orthogonality: Analysis of link-level performance," *IEEE Communications Letters*, vol. 22, no. 4, pp. 796–799, 2018.
[26] "SX1301 Datasheet V2.4," Semtech, 2017.