L3SFA: Load Shifting Strategy for Spreading Factor Allocation in LoRaWAN Systems

Mohamed Hamnache
University of Toulouse
IRIT-ENSEEIHT
Toulouse, France
Email: mohamed.hamnache@irit.fr

Rahim Kacimi
University of Toulouse
IRIT-UT3
Toulouse, France
Email: rahim.kacimi@irit.fr

André-Luc Beylot
University of Toulouse
IRIT-ENSEEIHT
Toulouse, France
Email: andre-luc.beylot@irit.fr

2020 IEEE 45th Conference on Local Computer Networks (LCN) | 978-1-7281-7158-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/LCN48667.2020.9314777

***Abstract*—LoRaWAN Enabled networks are expected to have a dizzying growth. Thus, an efficient allocation of wireless resources so as to support a large number of nodes is a major concern. In this paper we propose an SF assignment approach paying attention on the traffic load both per Spreading Factor and over the channels. Indeed, our strategy consists in finding a better distribution of the nodes on the SF by orchestrating an effective load balancing. Moreover, the performance of our solution is evaluated under diverse network configurations taking into account the capture effect and the non-orthogonality of SFs. In addition, we validated some assumptions by full-scale experiments like for the 3GPP path loss model which is used for the first time in LoRa simulations. Our results suggest that Load Shifting leads to better performance in terms of DER (Date Extraction Rate) while guaranteeing good scalability on the network size and density.**

***Index Terms*—LoRa networks, resource allocation, Spreading Factor, capture-effect, Load Balancing algorithms, 3GPP urban path loss model, Experiments.**

I. Introduction

LPWA Networks are low-power, low-speed and low-range wireless networks. Combined to the Internet they are about to bring us another fascinating decade as they morph the new generation of Internet of Things. Indeed, it is estimated that more than 20% of Internet of Things connections are made via LPWANs, representing several billion connected objects. Therefore, in order to meet the requirements in terms of massive deployments, the channel utilization has been regulated with a duty cycle at 1% and the bit rates have been reduced to a minimum. For instance, a LoRa gateway is expected to receive traffic from several thousand of nodes and despite this low duty-cycle, transmissions issues such as collisions are still observed in particular because of the traffic load and the number of modules using the same spreading factor. LoRaWAN [3] has recently raised much attention in wireless communications community due to its high promises in terms of density, low-power, long-range capacities and flexibility. All these features make it suitable for a large set of internet of things applications such as: environmental monitoring, smart city, on-street lighting control and precision agriculture.

In this paper, we investigate a different approach for Spreading factor allocation under various load scenarios for LoRaWAN networks. We argue for the use of a load control and shifting among the different SFs. Therefore, we propose a novel strategy (L3SFA) which improves the global Data Extraction Rate (DER). The L3SFA strategy considers the global system load in terms of generated traffic or number of nodes to allocate the SFs to nodes. Its main purpose is to distribute load on different SF classes while insuring a good DER.

The validation scenarios of our strategy are twofold: in the first scenario, random generation of nodes has been considered. The originality of this evaluation besides the proposed SF allocation strategy is the use of the 3GPP path loss model to describe link behavior and for the best of our knowledge this is the first time where such a model is used for LoRaWAN networks. To overcome some simulation limits we further considered a second scenario. In fact, emulated nodes are retrieved from a LoRaWAN dataset gathered during real deployment to feed our simulations. Finally, to demonstrate the feasibility of our schemes, we designed and implemented an end-to-end flexible and extendable LoRaWAN environment for real experiments.

The remainder of the paper is structured as follows: Related work is introduced in section II. In section III, we describe the LoRaWAN features. We then detail in Section IV our system model and the proposed load shifting strategy for SF allocation. Section V presents an experimental platform For link behavior validation and a thorough performance analysis of the L3SFA scheme. Apart from simulations, an experimental deployment in order to provide valuable insights and to show the performance of a real LoRaWAN system is given in Section VI where we further discuss some key practical issues and reveal their potential weaknesses. Section VII concludes this paper and gives future research perspectives.

II. Related Work

Plenty of works have been carried out on the impact of using different combinations of LoRa transmission parameters on LoRaWAN networks. Some studies on LoRa densification focused on how to determine network capacity around one single gateway [6], [13], [7].
Other contributions propose schemes for spreading factor allocation to optimize a specific performance criterion. Multiple approaches are explored to bring solutions:

- Heuristics such as EXP-SF and EXP-AT in [10] to achieve a high overall data rate, EXPLoRa-TS in [11] to equalize the traffic load in terms of symbol times.
- Optimization problem formulation in [19], [21] where authors modeled LoRaWAN resource allocation as a maximization or minimizing problem solved using meta-heuristics such as Genetic Algorithm and learning methods such as Reinforcement Learning.
- Applied machine learning methods in [11], [19], [5]. Several data science techniques were considered. For instance, k-mean supervised learning has been applied in [11] and [5] to find suitable clusters for SF allocation. Other papers are based on reinforcement learning where a gateway plays the role of an agent to control the environment (nodes) by sending actions (configurations). Convolutional neural networks were also used to solve optimization problems in [19].

However, these papers either, give just an analysis of LoRaWAN performance such as [6] [13] without proposing solutions or schemes to improve performance. Indeed, most of these studies focus on densification, coverage, and energy consumption to determine the lifespan of batteries but they fail in investigating on fundamental LoRa issues. In fact, they:

- neglect in the mathematical model or simulation some LoRa physical phenomena which have a direct impact on LoRaWAN network performance such as capture effect (Intra-SF Interferences) and imperfect SF orthogonality (Inter-SF Interferences) [10], [11], [21].
- mainly use Log-Distance path loss model [19], [2], [20] with a path loss at a distance $d_0 = 40$ meters equals to $127.41dB$. This is a penalizing assumption and can provide only deployment around a gateway with short distances which contradicts the main purpose of LPWAN technologies.
- poorly consider the important impact factors such as the traffic generation rate and largest network sizes while studying overloaded LoRa systems [12].

For these reasons, we revisited the SF assignment problem in order to a propose a novel strategy especially for high loaded LoRaWAN systems taking into account high generated traffic rates and a large number of nodes. Moreover, we argue for the use of the 3GPP model to describe the link behavior in LoRaWAN systems. As far as our knowledge, this is the first attempt and we endorsed this choice by an experimental study.

III. LoRaWAN Features

In this section we will give a brief overview of the LoRa physical layer as well as the LoRaWAN MAC layer.

A. LoRa: Physical Layer

LoRa (Long Range) uses a proprietary Chirp Spread Spectrum (CSS) modulation developed by Semtech in 2004 and integrated as a physical layer to LoRaWAN technology. This modulation technique is behind the key features offered by this technology. Indeed, LoRa transceivers characteristics are: long-range and low-power transmissions, high robustness and Doppler resistance. The quality of a transmission using this kind of modulation depends on the best choice of a combination of set of transmission parameters presented in the next section.

TABLE I: LoRa Parameters under EU regulation

Parameters	Values
SF	7, 8, 9, 10, 11, 12
TP	2, 5, 8, 11, 14 [dBm]
BW	125 kHz, 250 kHz, 500 kHz
CR	4/5, 4/6, 4/7, 4/8
CF	886.1, 886.3, 886.5, ...

B. Transmission Parameters

A radio transmission in LoRaWAN can be characterized by a combination of a set of configuration parameters. We can mainly define five parameters [7] [16] which have a direct impact on energy consumption, transmission range and robustness. See table I

1) Spreading Factor (SF): LoRa is a Chirp spreading spectrum modulation. The Spreading Factor is the parameter which controls how these chirps are spread. A lower spreading factor decreases the Signal to Noise Ratio (SNR), and thus sensitivity and range, but also decreases the airtime of the packet. The spreading factor values are between 7 and 12 and the choice is made according to a trade-off between the energy consumption and the robustness.

2) Transmission Power (TP): Using high TP improves the probability of receiving a message. However, it increases the energy consumption of the transmitter and the interference level.

3) Bandwidth (BW): it represents the width of a continuous band of frequencies. Higher bandwidth results in a shorter airtime but a lower sensitivity. LoRa is configurable with three possible values of the bandwidth [125 kHz, 250 kHz, 500 kHz] for both Uplink and Downlink.

4) Coding Rate (CR): CR is the forward error correction code added to a packet before transmission to offer protection against bursts of interference. LoRa uses four coding rates (4/5, 4/6, 4/7, 4/8). Undoubtedly, using higher CR provides more immunity but increases the time-on-air and consequently the energy consumption.

5) Carrier Frequency (CF): it is the center frequency and LoRa uses a multi-channel mechanism for uplink and downlink transmissions.

C. LoRaWAN MAC Layer

LoRaWAN Medium Access Control [14] is an open source protocol standardized by the LoRa Alliance that runs on top of LoRa physical layer and provides services to enable communication between devices and gateways.

A LoRaWAN network is stand on a star of stars topology (See Fig. 1), with no direct communications between the nodes. Indeed, nodes can only send data to one or more gateway. The gateways are connected to a centralized Network Server (NS). In such networks, the gateways mainly play the

role of Packet Forwarders. Thus, they receive frames from the end nodes and forward them to the NS. The NS is responsible on managing nodes and gateways, sending down-link packets and MAC commands towards the nodes, if needed. Finally, the communication ends on the right application profile.

LoRaWAN specify three different node classes {A, B, C} [8]. Class A and Class B are in most cases battery-powered, whereas Class C is mains-powered. By default, a LoRa node belongs to the Class A in order to meet the power saving requirements and can be configurable to work as Class B or Class C. The three classes differ in terms of the reception mode coming after the transmission one. Indeed, Class A nodes only have two very short receive windows after they have transmitted a frame. Class B nodes open extra receive windows within scheduled intervals (Beacon) and Class C nodes listen continuously for down-link incoming frames.

LoRaWAN protocol stack provides a set of methods and mechanisms to ensure secure and reliable communications. It proposes an Adaptive Data Rate (ADR) mechanism to remotely and dynamically manage the nodes in order to increase the packet delivery ratio. In addition, a device activation procedure is used to associate the nodes to the network. Indeed, each node must be authenticated (activated) to join a LoRaWAN network. Nodes can be activated over the air (OTAA) or by personalization (ABP).

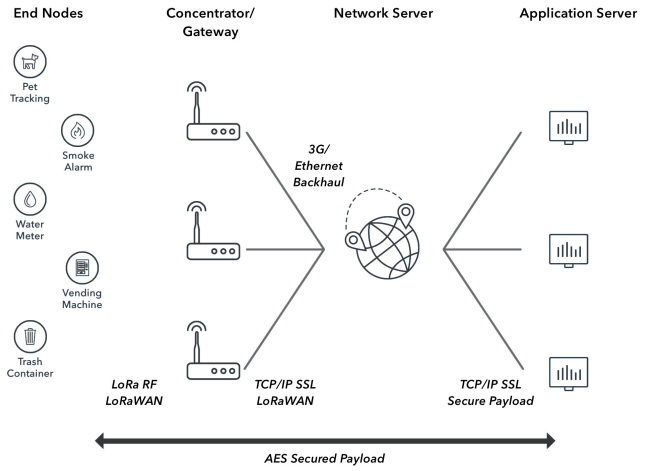


Fig. 1: A generic LoRaWAN architecture [17]

IV. System Model

We consider a LoRaWAN network composed of N nodes with one single gateway. Each node is initialized with $SF = i$ following the RSSI and the SNR thresholds given in Table III to send frames to the gateway. To determine the allowed SFs (between SF7 and SF12) for a given node, we define an accurate load balancing scheme.

A. SF shifting

Initially, nodes are assigned to $SF = j$ according to the thresholds defined in Table III with $j \in \{7, 8, 9, 10, 11, 12\}$.

TABLE II: Table of Symbols

Symbols	Notations
N	Number of nodes
N_i	Number of nodes using $SF = i$
T_i	Airtime for $SF = i$
p	Packet generation period (per node)
$M_{j,i}$	Nodes which can use $SF = j$ but shifted to $SF = i$
$M_{j,j}$	Nodes that keep their SF unchanged $SF = j$

Let $M_{j,i}$ be the number of nodes which can use $SF = j$ but shifted to $SF = i$ and $M_{j,j}$ number of nodes that keep their $SF = j$ unchanged.

For ease of presentation, we denote the total number of nodes that have been assigned a SF i, after the shifting procedure, as M_i. The shifting process is depicted in figure 2 and M_i can be computed as:

$$M_i = \sum_{j=7}^{i} M_{j,i} \tag{1}$$

B. SF Load

Let ρ_i be the load for each $SF = i$ after the SF shifting phase.

$$\rho_i = \frac{M_i \times T_i}{p} \tag{2}$$

ρ_i is computed using equation (2), where T_i is the time-on-air using $SF = i$ and p is the packet generation period per node.

Consequently, the Data Extraction Rate DER_i achieved for $SF = i$ with a load ρ_i can be calculated as:

$$DER_i = e^{-\rho_i} \tag{3}$$

C. Optimal SF Allocation

We formulate the SF allocation problem as an optimization problem. We consider the maximization problem described in equations (4a-4d).

$$\text{maximize} \quad \frac{1}{N} \sum_{i=7}^{12} DER_i \times M_i \tag{4a}$$

$$\text{subject to}$$

$$M_{j,i} \in \mathbb{N}, \tag{4b}$$

$$\sum_{j=i}^{12} M_{i,j} = N_i, \tag{4c}$$

$$M_{j,i} = 0 \qquad If \quad j > i \tag{4d}$$

The objective function given by the equation (4a) aims to maximize the global DER based on the DER_i of each $SF = i$ and the number of nodes using this SF after the load shifting phase is finished.

Equation (4b) makes a strong constraint to the optimization problem. Indeed, the number of nodes shifting their SF must be a positive integer. The shifting process is possible only if equation (4d) is satisfied. Indeed, shifting SF is one way

direction (from lower to higher SF), the other way is not possible.

The shifting process is controlled using equation (4c). This constraint guarantees that the total number of nodes shifted from $SF = i$ to higher ones must be equal to the number of nodes using $SF = i$ before the shifting process. This insures that $\sum_{i=7}^{12} M_i = N$.

The optimization problem presents a hard complexity. Indeed, on the one hand, we are facing an Integer Non-Linear Programming (INLP) [15] and on the other hand, the network size network is highly scalable.

Undoubtedly, an exhaustive search is not feasible due to the huge number of combinations, and no polynomial-time algorithm is known to be applicable in a general case, especially when the problem under optimization does not present a linear or convex form.

D. Load Shifting and SF Allocation Heuristics

In this section, we present a Load Shifting Strategy for Spreading Factor Allocation (L3SFA). The proposed Scheme is designed by default to optimally allocate SF to nodes to achieve better DER. It follows three main phases (See Algorithm 1) described in the following sections : 1) Initialization, 2) SF Allocation, 3) SF Adjustment.

Algorithm 1 L3SFA

Input : N : Nodes
Output : N with SF allocated, SF_Distribution

1: Initialization
2: SF Allocation
3: SF Adjustment

TABLE III: SNR and Sensitivity thresholds for 125 kHz Bandwidth in [dB]

SF	SNR	Sensitivity
7	-7.5	-126.5
8	-10	-127.25
9	-12.5	-131.25
10	-15	-132.75
11	-17.5	-133.25
12	-20	-134.5

1) Initialization: L3SFA starts with an initialization phase. Note that two actions are taken: 1) Since the SF allocation mechanism is incremental, all nodes are initialized with $SF = 7$. 2) Nodes are sorted according to their distance from the gateway and accordingly their RSSI. Obviously, this intuitive action allows to assign lower SF on priority basis to the closer nodes to the gateway or to the ones with the highest RSSI value. This is detailed by the lines [1,2] in Algorithm 2.

2) SF Allocation: After the initialization, L3SFA strategy begins the second phase of spreading factor pre-allocation. SF_LoRa Algorithm (lines 3 to 12) goes through the list of sorted nodes produced during the first phase and for each node a suitable SF is assigned function of the SNR and the RSSI values received on the gateway. These values are compared to RSSI and SNR thresholds given in table III. The SF allocation condition defined in line 7 is necessary and sufficient to guarantee a successful transmission. Undoubtedly, the RSSI higher than the sensitivity insures the frame reception by the gateway and combined to SNR higher than the SNR threshold, the frames will be decoded certainly.

3) SF Adjustment:
The SF allocation performed in phase two (SF Allocation) is not final. This allocation is controlled by the last phase of the L3SFA mechanism.

Before assigning definitely a SF to a node, SF_LoRa performs a load control on SF distribution (line 13). This control is performed with SF_Shifting heuristic.

Algorithm 3 and Fig. 2 give a description of the SF adjustment mechanism. SF_shifting needs to have a global viewpoint on the SF allocation system to perform the adjustments. Indeed, it takes as input the SF preassigned to a node, SF_Distribution (a vector of number of nodes using definitely $SF = i$ for $i \in \{7, 8, 9, 10, 11, 12\}$) and loads (a vector of number of nodes threshold for each $SF = i$) computed using equation 2.

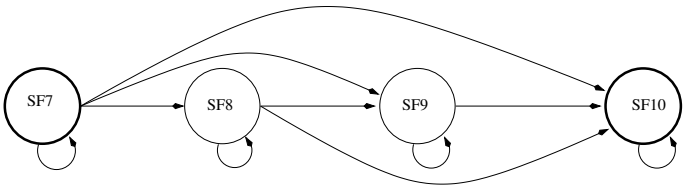


Fig. 2: Load Shifting Strategy for Spreading Factor Allocation Finite State Machine (only SF7 to SF10 are considered)

Without loss of generality, we illustrate in figure 2 how the SF adjustment scheme behaves. Let consider a simple system using only four SF values (from 7 to 10) where N nodes are sending frames to a single gateway. We suppose that SF Allocation phase allocates $SF = 7$ to a node k. SF_Shifting will check if this allocation is appropriate or it will be better to upgrade it. The finite state machine in Fig. 2 describes such a situation, a node with $SF = 7$ is facing two possible situations:

- conserves $SF = 7$ if the class of nodes this SF is not overloaded or yes it is but all the other classes are also overloaded.
- Upgrades to $SF = j$ with $j > 7$ if the class of nodes using $SF = 7$ is overloaded and the classes of nodes using $SF = j$ is not overloaded ($j \in \{8, 9, 10\}$ the first SF not overloaded)

Using such a strategy guarantees that each node will be assigned the suitable SF which is qualified as good for two main reasons: 1) SF allocation in phase two insures that a frame will be received and can be decoded if no-collision happens. 2) SF adjustment in phase three controls the SF load which reduces significantly the number of collisions.

As expected, shifting some nodes when possible to higher SF reduces the capture effect phenomena. Otherwise, having lower SF classes overloaded is less penalizing in term of DER than overloaded classes with higher SF. This is due to time-on-air of the frames which increases by increasing SF and consequently increases the collision probability.

V. Performance Analysis

In this section, we examine the performance of the proposed strategy through extensive simulations and implementation of the algorithms detailed in section IV.

Apart from the design of L3SFA strategy, we also developed a simulation tool LoRa-L3SFA to demonstrate the performance gains of our solution. LoRa-L3SFA simulation tool combines Simpy and the 3GPP Urban Macro path loss model. In addition to that, it is also based on LoRaSim [7] and LoRaFREE [2]. In fact, to highlight the applicability of our strategy on LoRaWAN systems, we extend LoRaSim with new features. The new LoRa-L3SFA tool features are as follows:

- It considers the imperfect orthogonality of spreading factors and the fading impact based on the interference thresholds reported in table IV) for frame collisions check.
- It integrates a 3GPP propagation model for random generation nodes. To the best of our knowledge, there is no other work that uses this model in LoRa performance studies.
- It also takes, as inputs, realistic RSSI and SNR values from an experimental recorded data-set.

A. Propagation Model

A transmission in LoRaWAN systems is successfully received if the received signal at the gateway P_{rx} is higher than receiver sensitivity. Equation 5 gives the formula of P_{rx} :

$$P_{rx} = P_{tx} - PL + GL \tag{5}$$

where P_{tx} is the transmission power, PL is the path loss and GL combines all general gains and losses.

Several models exist to describe path loss PL for different environments (urban, suburban, rural, free space) such as: log-distance [18], okumora-hata [22] and 3GPP models [1].

We use in this work the 3GPP Urban Macro path loss model [1] which is commonly used to model deployments in urban macro-cells areas. We choose this model as it matches the environments where LoRa networks are expected to be deployed. Using this path loss model depends on the communication distance d and can be described as:

$$\begin{aligned} PL = {} & 44.9 - 6.55 \times \log_{10}(GH) \times \log_{10}(\frac{d}{1000}) \\ & + 45.5 \\ & + 35.46 \times 1.1 \times NH \times \log_{10}(f_c) \\ & - 13.82 \times log_{10}(GH) \\ & + 0.7 \times NH \\ & + \kappa \end{aligned} \tag{6}$$

Algorithm 2 SF_LoRa

Input : N : Nodes
Output : N with SF allocated, SF_Distribution

```
Sort Nodes by distance d to the gateway;
Init all nodes with SF =7 and SF_Distribution to zeros
for n in N do
    SF = n.sf
    Get SNR and RSSI thresholds for the corresponding SF from the table III
    for s in range (SF,12) do
        if n.snr < SNR_sf or n.rssi < sensibility_sf then
            if n.sf < 12 then
                n.sf = n.sf +1
            end if
            Get SNR and RSSI thresholds for the corresponding SF from the table III
        end if
        n.sf = SF_Shifting(n.sf,SF_Distribution)
    end for
    SF_Distribution[n.sf] = SF_Distribution[n.sf] + 1
end for
```

Algorithm 3 SF_Shifting

Input : SF, SF_Distribution, load
Output : SF

```
for s in range (SF,12) do
    if SF_Distribution[s] < load[s] then
        return SF
    else
        if SF_Distribution[s] ≥ load[SF] and SF_Distribution[s+1] < load[s+1] then
            if s < 12 then
                return s+1
            end if
        end if
    end if
end for
```

where GH is the gateway antenna height in meters, NH the node antenna height in meters, f_c the carrier frequency in MHz, d is the distance between the gateway and the node in meters, and κ is a constant factor ($\kappa = 0\ dB$ for suburban macro and $\kappa = 3\ dB$ for urban macro).

B. Experimental Platform For Link Behavior

In order to better understand how the RSSI behaves according to the distance between the end modules and the LoRa Gateway we conducted the following experimental study. The objective is also to validate and support the chosen path loss model. On the hardware side, we used for the experiments a MultiTech Conduit Gateway and a RN2483 microchip node scratched on a Raspberry Pi3. In addition, we equipped this platform with an Xbee GPS module (See Fig. 3). The LoRa

TABLE IV: Interference Thresholds [dbm]

	7	8	9	10	11	12
7	6	-8	-9	-9	-9	-9
8	-11	6	-11	-12	-13	-13
9	-15	-13	6	-13	-14	-15
10	-19	-18	-17	6	-17	-18
11	-22	-22	-21	-20	6	-20
12	-25	-25	-25	-24	-23	6

node is attached to a bicycle and transmits the GPS coordinates every 20 seconds under the following parameters: $SF = 12$, $BW = 125kHz$, $Ptx = 14dBm$ and $CR = 4/5$.

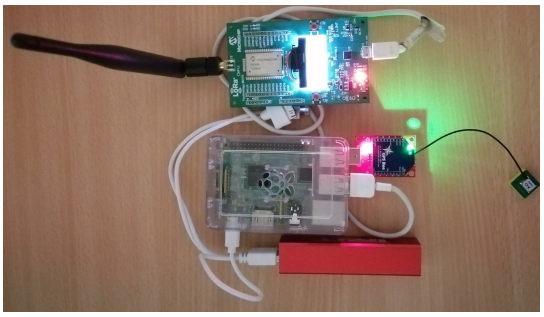


Fig. 3: Experimentation Setup

Fig. 5 shows the experimentation circuit traced using the GPS coordinates sent by the RN2483 microchip node. The aim of this experiment is to compare the propagation models with measurements from a real environment and using a real hardware. The results of this comparative study are represented in Fig. 4.

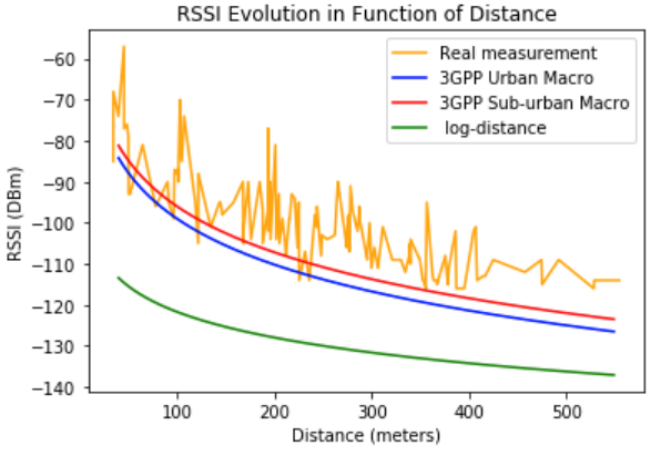


Fig. 4: Received Signal Strength Indicator: Mathematical Models vs Real Measurements

As all the path loss models depend on the distance d between the node and the base station, we plotted the evolution of the RSSI as a function of the distance for log-distance model, 3GPP models and RSSI values from experimentation (distances are computed using GPS coordinates). We can see

Fig. 5: GPS Coordinates Transmitted With RN2483 Equiped With Xbee GPS Module

that all the curves have the same appearance. In fact, the value of the RSSI decreases while moving away from the gateway.

$$PL = PL(d_0) + 10\gamma log_{10}(\frac{d}{d_0}) + \chi \quad (7)$$

Log-distance model (equation 7) is very penalizing and gives overestimated RSSI values compared to the empirical results obtained through the experiments. This is due to an overestimation of the path loss value at a reference distance d_0, where $PL(d_0)$ at distance $d_0 = 40$ meters is supposed to be $127.41dB$. γ is the Path Loss Exponent with a value of 2.08 and χ is a zero-mean Gaussian distributed random variable (in dB) with standard deviation σ. This variable is used only when there is a shadowing effect. If there is no shadowing effect, then this variable is zero (we suppose that $\chi = 0$).
3GPP models as shown in Fig. 4 are suitable for LoRaWAN networks, we notice that the 3GPP models are not too penalizing in terms of path loss and reflects results obtained by experimentation.
As a conclusion, we decide to use 3GPP Urban macro in our simulations because it gives RSSI values which correspond to real environments where LoRaWAN networks are the most adopted.

C. Simulation Scenarios

We simulate and compare results from L3SFA Load Shifting for SF Allocation approach and SF Allocation without load control (the allocation derived only by the LoRaWAN specifications). In all scenarios, we consider an application that generates packets of 20 $Bytes$ according to a Poisson process. We also assume that generation period p is the same for all nodes.

Before the packet transmission phase, the L3SFA scheme starts and follows three different steps that we describe in section IV. Indeed, let remind that L3SFA initializes all nodes

with $SF = 7$ and sorts them according to the distance d or their $RSSI$, allocates the SFs while controlling the load generated on the different SF classes. Hence, these operations produce a set of nodes with SF assignment. We simulate packets transmission by N nodes using SFs allocated previously for 2 hours. We examine the scalability and the performance of our approach in terms of Data Extraction Rate (DER) metric. Indeed, in an effective LoRa deployment all transmitted messages should be received at the gateway level, the higher the DER the effective and efficient is the deployment.

To achieve higher accuracy of simulation results, two kinds of scenarios were considered:

1) Scenario 1 - Random node generation:

TABLE V: Simulation Parameters

parameters	Values
Nodes	100-10000 Randomly Scattered
Bandwidth	125 [kHz]
Coding Rate	4/5
Transmission Power (P_{tx})	14 [dBm]
Spreading Factor	7, .., 12
Channels	3 up-links [868.1, 868.3, 868.5]
Concurrent Receptions	8
SNR Thresholds	Table III
Sensitivity	Table III
Inter-SF and Intra-SF Interferences	Table IV
p	100, 300, 600 ($second$)
Packet Length	20 [bytes]
Path Loss [1]	3GPP Urban Macro
Node Height (NH)	1 [meter]
Gateway Height (GH)	15 [meter]

In this approach, the nodes are randomly generated and positioned in a 2-dimensional space forming a cell of $radius = 600$ meters. Several simulation runs are achieved for each scenario with different seed values in order to ensure the accuracy of the results.

We run 2400 simulations for N nodes, $N \in [500, 10000]$ with a step of 500, under different configurations. All simulation settings, propagation model variables, transmission and L3SFA parameters are summarized in table V

For this scenario, we plotted in Fig. 6d, to 6f the spreading factor distribution after SF allocation with and without load control for a LoRa network of 5000 nodes. Similarly, we highlighted the impact of the Sf redistribution on providing better global DER in Fig. 6a to 6c. It is also worth noting that, the traffic load generated decreases by using respectively packet generation rate $\frac{1}{100}$, $\frac{1}{300}$, and $\frac{1}{600}$.

We observe from Fig. 6a to 6c that L3SFA leads to better results in terms of DER. Indeed, shifting some nodes to higher SF lightens the system load and reduces collisions due to the capture effect, hence it increases LoRa network capacity.

For instance, for a packet generation ratio of $\frac{1}{600s}$ and to achieve a DER higher than 80%, L3SFA supports 8500 nodes for just 6000 nodes using L3SFA without SF adjustment phase (there is no SF classes load control).

2) Scenario 2 - Dataset based node generation:

In Scenario 2, we used a real dataset of LoRaWAN messages obtained in the city center of Antwerp [4]. Simulated nodes positions are picked from actual positions of nodes in the dataset. This dataset holds 130 430 messages which are collected from November 17^{th}, 2017 until July 19^{th}, 2019.

This data-set is a wealth of information which contains node payloads combined to meta-data describing the channel state (RSSI, SNR) and the transmission parameters (SF, BW, CR, etc). Thus, we used these RSSI and SNR values as simulation inputs to emulate nodes with real radio parameter like in a real deployment. All other simulation parameters are the same as the first scenario (See Table V).

Similarly as in random generating study, we evaluate the simulation done with real emulated nodes to demonstrate the effectiveness of our solution and improve the accuracy of our evaluation. The obtained results are shown in Fig. 7. Fig. 7a clearly shows that shifting some nodes to higher SFs(See Fig. 7b) in order to lighten SF classes suffering from high load leads to better global DER. Hence, using L3SFA increases the network capacity. Indeed, we report from Fig. 7 that to achieve a DER higher than 85%, the network can support 6000 nodes versus only 4500 without SF adjustment.

VI. Testbed Setup

In this section, we address some practical issues, and present an end-to-end testbed setup for LoRaWAN experiments. Indeed, to validate our SF allocation approach we set up a real experiment platform composed of a Multitech Conduit Gateway and RN2483 Microchip based nodes.

On the Backend, we deployed the ChirpStack [9] solution. ChirpStack is an open source LoRaWAN stack which provides open-source software components (cf. Fig. 8) for setting up LoRaWAN networks. Theses components are described as follows:

- Gateway Bridge: This component is used for format conversion to handle communications between the gateway and the network server.
- Network Server: This component is the ChirpStack core. It is responsible for receiving frames and provides: authentication, MAC layer, downlink scheduling and insures communications with the Application Server.
- Application Server: This component is responsible on node management, handling and encryption of application payloads and offers integration with external services.

ChirpStack provides a high level of modularity which makes it flexible and extendable. Note, however, that as other existing LoRaWAN projects, ChirpStack does not allow to send personalized downlink control messages (MAC commands) for over the air node configuration. Indeed, ChirpStack is designed to send only MAC commands when ADR (Adaptive Data Rate) is activated. In an other term, sending customized control commands in the payload by positioning the "Port" to 0 or using header fields "fOpts" and "fOptsLen" is not possible. In the end, sending mac commands in the downlink turns out to be a tedious task which complicates testing new approaches

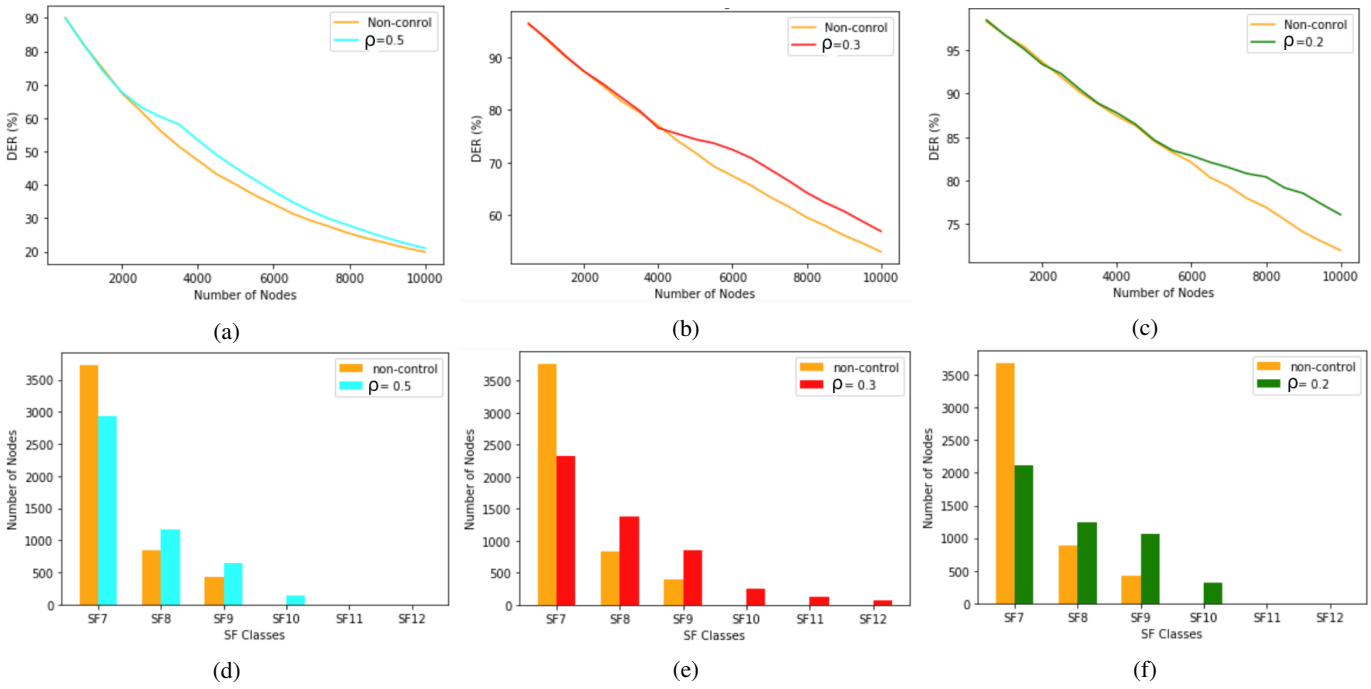


Fig. 6: L3SFA Random Generation Performance Study. [a-c] DER as a function of the number of nodes for a packet generation rate respectively $\frac{1}{100s}$, $\frac{1}{300s}$, $\frac{1}{600s}$. [d-f] SF Distribution for 5000 nodes

that propose a gateway-centric node control.

Our objective is to exploite the modularity of ChirpStack to extend it with a new flexible and modular module called "L3SFA Extention" (See Fig. 8). This new module provides very interesting features. In fact, it is composed of two main parts: *1) Core Engine:* this is the modular part of the module, it implements LoRa node control intelligence and make possible to LoRawan developers to inject their proposed mechanisms. For instance, L3SFA is provided in the core engine and other mechanisms can be added. *2) Flask Server:* exposes endpoints to interface with ChirpStack Network server. It provides MAC command sequences computed by the core engine and insert them in the network server down-link queue for scheduling. We plan to use this platform in our future work to evaluate L3SFA in real environment and explore downlink issues.

VII. Conclusion

In this paper, we investigated the problem of Spreading Factor allocation in large-scale LoRaWAN systems with low duty-cycle. We demonstrated that conventional SF assignment strategies suffer from poor performance or simply fail to handle the network traffic load. We showed that it is equivalent to a load balancing problem, and accordingly presented a novel approach for SF allocation which consists on finding the best distribution of the nodes among the different SF classes by orchestrating an effective load balancing. Our solution has also motivated an experimental study that relies on the correlation between RSSI and distance. We examined the performance of our strategy under various network configurations. Our results suggest that Load Shifting leads to better performance in terms of DER (Date Extraction Rate) while guaranteeing good scalability on the network size and density.

In the future, we aim to improve the performance of our strategy. Besides, we would like to deploy our algorithms on our LoRaWAN network server, and conduct experiments to investigate how it behaves in real infrastructure especially on the down-link. We are also interested in using the macro-diversity in reception to extend our strategy to a multi-gateways solution.

References

[1] 3GPP, "Etsi tr 125 996 v10.0.0," 2011. [Online]. Available: https://www.etsi.org/deliver/etsi_tr/125900_125999/125996/10.00.00_60/tr_125996v100000p.pdf

[2] K. Q. Abdelfadeel, D. Zorbas, V. Cionca, and D. Pesch, "$free$ —fine-grained scheduling for reliable and energy-efficient data collection in lorawan," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 669–683, 2020.

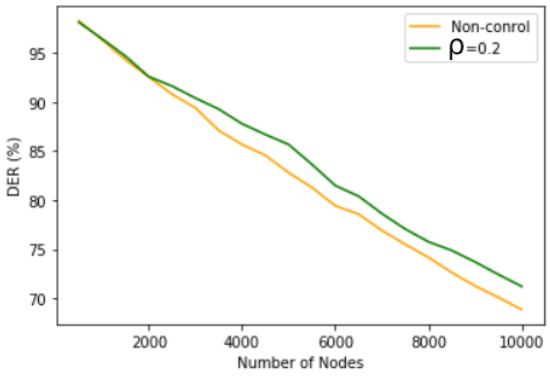
[3] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of lorawan," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, 2017.

[4] M. Aernouts, R. Berkvens, K. Vlaenderen, and M. Weyn, "Sigfox and lorawan datasets for fingerprint localization in large urban and rural areas," *Data*, vol. 3, p. 13, 04 2018.

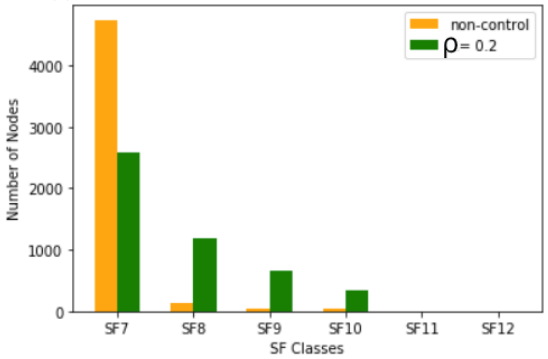
[5] M. Asad Ullah, J. Iqbal, A. Hoeller, R. D. Souza, Alves, and Hirley, "K-means spreading factor allocation for large-scale lora networks," *Sensors (Basel, Switzerland)*, vol. 19, no. 21, p. 4723, Oct 2019, 31671700[pmid]. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31671700

[6] D. Bankov, E. Khorov, and A. Lyakhov, "Mathematical model of lorawan channel access," in *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2017, pp. 1–3.

[7] M. Bor, U. Roedig, T. Voigt, and J. Alonso, "Do lora low-power wide-area networks scale?" in *Proc. of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWIM'16*, 11 2016, p. 59–67.



(a) DER as a funtion of the number of nodes



(b) SF Distribution for 5000 nodes

Fig. 7: L3SFA Dataset Generation Performance Study

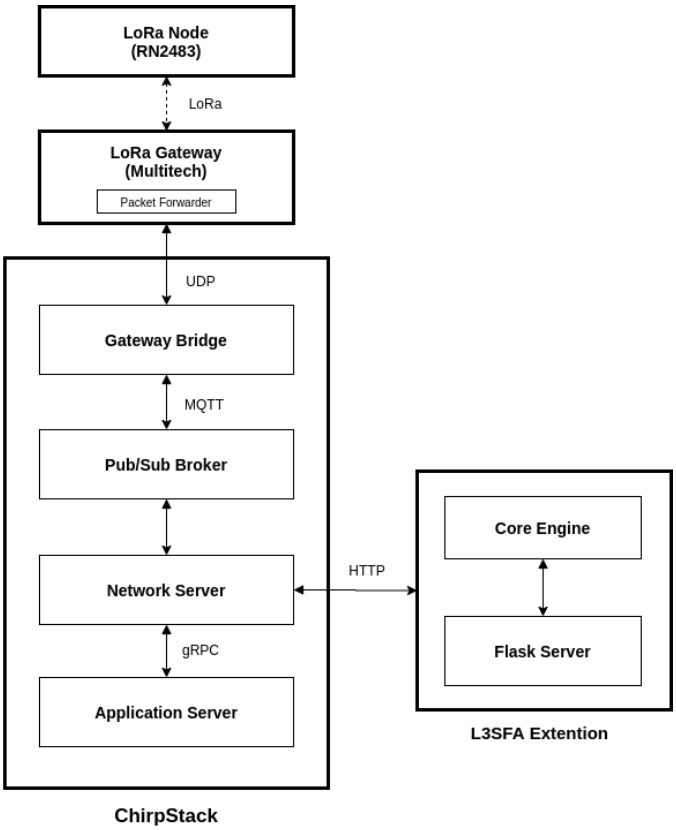


Fig. 8: Extended ChirpStack Architecture

[8] P. S. Cheong, J. Bergs, C. Hawinkel, and J. Famaey, "Comparison of lorawan classes and their power consumption," 11 2017.

[9] ChirpStack, "Chirpstack: open-source lorawan network server stack." [Online]. Available: https://www.chirpstack.io/

[10] F. Cuomo, M. Campo, A. Caponi, G. Bianchi, G. Rossini, and P. 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)*, 2017, pp. 1–8.

[11] F. Cuomo, J. C. C. Gámez, A. Maurizio, L. Scipione, M. Campo, A. Caponi, G. Bianchi, G. Rossini, and P. Pisani, "Towards traffic-oriented spreading factor allocations in lorawan systems," in *2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, 2018, pp. 1–8.

[12] A. Farhad, D.-H. Kim, and J.-Y. Pyun, "Resource allocation to massive internet of things in lorawans," *Sensors*, vol. 20, p. 20, 05 2020.

[13] 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.

[14] J. Haxhibeqiri, E. De Poorter, I. Moerman, and J. Hoebeke, "A survey of lorawan for iot: From technology to application," *Sensors*, vol. 18, p. 3995, 11 2018.

[15] R. Hemmecke, M. Köppe, J. Lee, and R. Weismantel, "Nonlinear integer programming," in *50 Years of Integer Programming*, 2010.

[20] R. M. Sandoval, D. Rodenas-Herraiz, A. Garcia-Sanchez, and J. Garcia-

[16] J. Liando, A. Gamage, A. Tengourtius, and M. Li, "Known and unknown facts of lora: Experiences from a large-scale measurement study," *ACM Transactions on Sensor Networks*, vol. 15, pp. 1–35, 02 2019.

[17] T. T. Network, "Lorawan architecture," Jun 2020. [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/architecture.html

[18] T. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. USA: Prentice Hall PTR, 2001.

[19] R. M. Sandoval, A. Garcia-Sanchez, and J. Garcia-Haro, "Optimizing and updating lora communication parameters: A machine learning approach," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 884–895, 2019.
Haro, "Deriving and updating optimal transmission configurations for lora networks," *IEEE Access*, vol. 8, pp. 38 586–38 595, 2020.

[21] R. Sandoval, A.-J. Garcia-Sanchez, and J. Garcia-Haro, "Performance optimization of lora nodes for the future smart city/industry," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, 12 2019.

[22] Y. Singh, "Comparison of okumura, hata and cost-231 models on the basis of path loss and signal strength," *International Journal of Computer Applications*, vol. 59, pp. 37–41, 12 2012.