QoS Based Spreading Factor Assignment for LoRaWAN Networks in IoT Applications

Shobhit Aggarwal
Dept. of Electrical and Computer Engineering
University of North Carolina at Charlotte
Charlotte, USA
Email: saggarw4@uncc.edu

Asis Nasipuri
Dept. of Electrical and Computer Engineering
University of North Carolina at Charlotte
Charlotte, USA
Email: asis.nasipuri@uncc.edu

SoutheastCon 2022 | 978-1-6654-0652-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/SoutheastCon48659.2022.9764088

***Abstract*—This paper addresses the issue of meeting communication performance requirements for Internet of Things (IoT) applications using LoRaWAN, an emerging low–power wide–area networking (LPWAN) technology. The proposed approach involves the development of rules for the assignment of LoRa spreading factors to IoT devices to limit the number of packet collisions selectively for devices having high Quality of Service (QoS) requirements. Two different spreading factor assignment schemes are proposed that differ in the assignment rules applied. Performance results, obtained using mathematical models and simulations, indicate that the proposed schemes adequately distinguish between high priority and low priority devices and can meet QoS requirements within limitations of network size.**

***Index Terms*—LoRa, IoT, Scalability, QoS**

I. Introduction

Recent developments in technologies for IoT have led to an exponential increase in the number of IoT applications [1] such as smart wearables, smart homes, smart buildings, smart cities, and many others. IoT devices are typically embedded into the physical world and connected to the Internet, thereby enabling them to be accessed via the Internet. To facilitate the connection of these widely spread IoT devices to the Internet, backhaul networking technologies such as Wi-Fi, LPWANs, Zigbee, Cellular technologies, etc. are required. Most IoT devices are configured to run specific applications, thereby leading to a variety of different categories of IoT devices in the same network. The QoS requirements for different categories may be different. For example, one category of devices that consist of smoke sensors and buzzers can be running a fire alarm application with higher QoS requirements than another category of IoT devices consisting of temperature and humidity sensors for a heating, ventilation, and air conditioning (HVAC) control system. These QoS requirements may include packets success rates, latency, data rate, bandwidth, etc. Also, some applications may have strict QoS requirements [2]. The underlining network must satisfy these requirements in order for the application to run successfully on the IoT devices. The main focus of this paper is to provide communication support to IoT networks that require specific QoS guarantees using LoRaWAN, which is a popular LPWAN technology for smart city applications.

LPWAN technologies are gaining industrial interest as a viable choice for connecting IoT devices spread over a large geographical area. Low power consumption, single hop large communication range, and ability to connect thousands of devices using a single gateway are some of the main characteristics of LPWANs that makes LPWAN a great backhaul technology for smart city deployments. LoRa is the most prominent LPWAN standard due to its excellent performance in meeting the above requirements. LoRa uses a proprietory chirp spread spectrum technology with provision to use multiple spreading factors and power levels that provide robustness to interference as well as long–range connectivity with adaptive date rate control. These features make this technology best suited for IoT applications such as utility metering, environmental monitoring, smart appliances, etc. However, the LoRa network protocol suite developed by the LoRa Alliance called "LoRaWAN" does not differentiate between different types of devices. Consequently, the legacy LoRaWAN protocol does not guarantee any specific QoS requirements.

LoRaWAN employs an adaptive data rate (ADR) mechanism that improves the network performance by optimizing the data rate and power consumption for every device, utilizing a combination of appropriate spreading factors and transmit powers. However, this mechanism has the shortcoming of using a greedy approach. While assigning the configuration parameters, ADR only considers the performance of the device it is trying to optimize without consideration for other devices. Because of this, at higher traffic loads, collisions take place and the performance degrades. Hence, ADR can not guarantee to satisfy the QoS requirements..

The spreading factors in LoRa provide the mechanism to achieve trade-off between transmission range and data rate. A higher spreading factor provides a higher transmission range at a lower data rate and vice versa. Hence, lower spreading factors are assigned to devices that are close to the gateway and the ADR assigns highers spreading factors to those devices that are relatively far from the gateway or have low signal-to-interference ratios. However, given that the spreading factors are orthogonal to each other, spreading factors distribution may also be useful for *increasing network capacity*. A device that is in close vicinity to the gateway that is configured on a lower spreading factor can also be configured to a higher spreading factor if required. Hence, by assigning different spreading factors, the collision domain of devices can be changed and number of devices configured on the same spreading factor is reduced, thereby improving the performance.

In our previous work, we exploited this orthogonality property of spreading factors and calculated an optimum distribution of devices under different spreading factors that reduced the total number of collisions in the network [3]. Specifically, it was shown that in place of using the lowest spreading factor for all devices, allocating 36% of the devices to the next higher spreading factor with 64% of the remaining devices in the lower spreading factor improves the average packet success probability. However, it must be noted that the packet success probability depends on the number of devices. Hence, the optimum distribution only maximizes the average performance but does not guarantee to satisfy specific QoS requirements.

In this work, we propose spreading factor allocation schemes that satisfy the specific QoS requirements for special applications that need high QoS guarantees while still maximizing the QoS of other devices in the network. The rest of the paper is organized as follows. Section II provides the necessary background knowledge of LoRa technology. Section III presents the related works done to study the performance of LoRaWAN and the works done to improve its performance. Section IV dives deeper into the proposed methodology followed by performance evaluation in section V and section VI provides the conclusion and future prospects of the work.

II. LoRa Background

LoRa is physical layer technology that was developed by Semtech. It uses a modulation scheme that is derived from Chirp Spread Spectrum (CSS) modulation and operates on a license-free ISM band. In the US, LoRa operates on a frequency band centered at 915 MHz. The CSS modulation along with adaptive data rate mechanism enables LoRa devices to achieve transmission ranges of up to 5 miles in most open terrains [4]. Hence, LoRa devices can be configured in a star network with a single gateway covering several miles. The gateways are connected to the LoRaWAN network server forming a star-of-stars configuration.

A. LoRaWAN Architecture

The LoRaWAN network architecture is illustrated in Figure 1. At the lowest level are LoRa end devices consisting of sensors and/or actuators that are equipped with LoRa transceiver modules to send and receive data. The end devices are connected to one or more gateways using a single-hop LoRa RF communication link. The LoRa gateways are data concentrators that relay data from the end devices and the LoRa network server. The LoRa network server controls the entire network, including radio resources, security, and admission control. The application server runs and manages various applications. The network server delivers the data that it receives from the end devices to the application server for processing. The network server also takes data from the application server and transmits it to the end devices via the appropriate gateway [5]. Theoretically, each gateway is capable of serving hundreds of thousands of end devices [6].

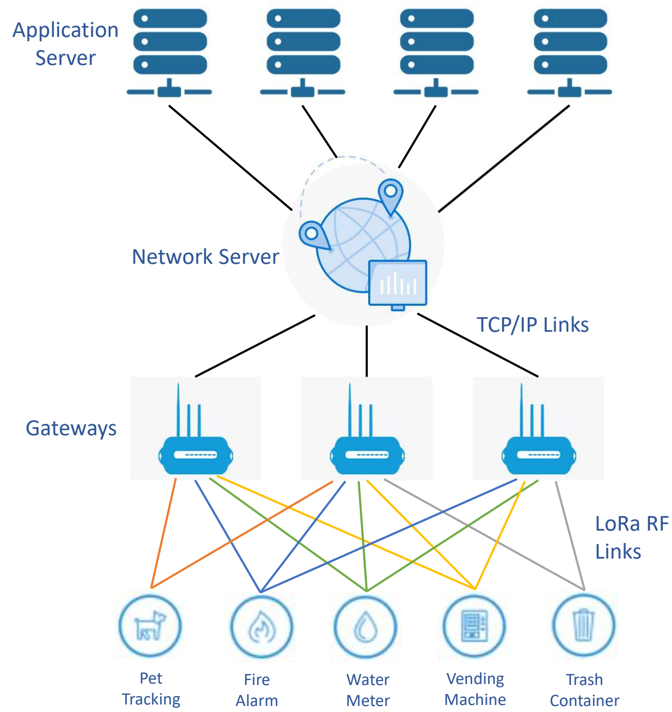


Fig. 1. LoRaWAN architecture

B. Configuration Parameters

LoRa modulation is specified by three major configurable parameters. They are:

- *Coding Rate (CR):* The CR determines the rate of the Forward Error Correcting code as $CR = \frac{4}{4+R}$, where 'R' ranges between 1 – 4.
- *Spreading Factor (SF):* SF determines the length of chirp symbol in time domain as $T_s = 2^{SF}/BW$. In the US, LoRa transmission can have SF ranging from 7–10 [7]. All the spreading factors are theoretically orthogonal to one another. A lower spreading factor indicates a shorter symbol duration, hence less time-on-air resulting in a higher data rate, and the higher the spreading factors, the lower will be the data rates. The spreading factors also affect the sensitivity of the receiver. A lower spreading factor corresponds to lower sensitivity, hence a lower range and vice versa. Table I [4] tabulates various data rates and typical transmission ranges associated with each spreading factor at 125 kHz bandwidth.
- *Bandwidth (BW):* In a given band, the BW specifies the allowed range of frequencies. The US allows 125 kHz and 500 kHz BW [7].

C. LoRaWAN MAC Layer

LoRa technology is a proprietary technology on the physical layer however, it is open to any protocol on the MAC layer. The LoRa Alliance [8] has developed an ALOHA-based MAC protocol for LoRaWAN networks. Under this MAC protocol, all devices can be categorized into three classes namely Class-A, Class-B, and Class-C.

Devices under the class-A mode of operation consume the least amount of energy. This category of devices opens two receive windows after every transmission and spends most of their time in sleep mode. However, the devices under the class-C mode of operation are the most power-hungry of all. Class-C devices have their radios always turned on in the receive mode, whenever they are not transmitting.

D. Adaptive data rate

The adaptive date rate mechanism is an essential feature of the LoRaWAN network. It seeks to optimize time-on-air, datarate, and energy consumption for the devices in the network [7]. This feature works best for the static end devices experiencing stable RF conditions.

ADR algorithm runs in two parts, on the network server as well as the end devices. It also requires the network server to send some feedback to end devices after every pre-configured "l" number of uplink transmissions. The algorithm starts all the end devices with the highest spreading factor. The network server makes SNR and RSSI measurements over the uplink transmissions. A "margin" is calculated from these measurements as

$$SNR_{margin} = SNR_{measured} - SNR_{req} - margin \quad (1)$$

where, '$SNR_{measured}$' is the SNR measured from the uplink, 'SNR_{req}' is the required SNR for the demodulation of a message and the 'margin' is an preconfigured margin that is kept to consider the fluctuations in the RF medium. The data rate is increased or decreased based on the calculated margin [7] [9]. Once the data rate is configured, the algorithm tries to optimize transmit power in order to save energy.

If the end device does not receive any feedback from the server after sending the 'l' number of uplink transmissions, the end device will send next 'l' uplink transmission by increasing it spreading factor by one. This cycle repeats itself till the highest spreading factor. If the device is configured with highest spreading factor and still does not receive any feedback, it starts the re-joining procedure.

III. Related Works

LoRa technology has attracted a lot of attention from researchers lately. This section briefly discusses some of the existing works done on the QoS requirements for IoT applications, performance evaluation, and spreading factor allocation for LoRa networks.

The authors of [2] and [10] present the QoS requirements by IoT applications and network deployments that follow those requirements. In [2], the authors present a generic QoS architecture for IoT applications. The paper categorizes all the IoT applications into three categories namely: inquiry task, control task, and monitoring task. They further state that the task of inquiry (eg. status of smart logistics) focuses on timeliness and reliability. The control task (eg. remotely controlling some actuator) relies on timeliness and reliability of the information, and the monitoring task may require real-time or non-real time data. Different applications have different QoS requirements.

An analytical model for IoT applications is proposed in [11]. The authors classified the network traffic into two levels of priorities i.e. low priority (normal traffic) and high priority(emergency traffic). They considered the low priority traffic can only be served in absence of high priority traffic and low priority traffic can be entirely sacrificed if required in order to serve the delay–sensitive high priority traffic. The service to low priority traffic is immediately preempted upon arrival of high priority traffic.

There are several studies done to evaluate the performance of the LoRa networks. The authors of [12] provides insights about the capabilities and limitations of LoRa networks. The packet success probability was analyzed with different number of total devices (250, 500, 1000 and 5000) in the network. The study shows that even with only 250 devices the success probability of 14.01% was achieved. Also, the probability of collision increases with the length of the payload. A comparative survey of various LPWAN technologies is presented in [13]. The authors also analyzed the various performance parameters of the LoRa network such as practical transmission range, scalability and power consumption. The study shows that a non-line-of-sight transmission range of 3 miles was achieved. At this transmission range more than 53% of transmitted packets were successfully received. Both [12] [13] confirm that the network performance degrades as the number of devices in the network increases. Packet collisions play a vital role in this performance degradation and gives rise to the scalability issues in LoRa networks.

Significant amount of work has also been reported on the assignment of spreading factors in order to improve the performance of LoRaWAN networks. The authors of [14] proposed a machine learning based approach to assign spreading factors to the devices. The mechanism starts by learning the transmission behaviour of the nodes in the network. The network server then assigns the spreading factors to nodes based on its prediction regarding collisions. This prediction model requires the location of each node. Hence, for this mechanism to work at least three gateways are required to triangulate the nodes' position. [15] proposes a lightweight spreading factor assignment scheme. According to their approach, the devices are initially assigned spreading factors based on the received power of the transmissions. It compares the received power

TABLE I
LoRa spreading factors

Spreading factor (UL for 125 KHz)	Physical bit rate (bits/sec)	Transmission Range (Depends on Terrain)
SF7	5470	2 km
SF8	3125	4 km
SF9	1760	6 km
SF10	980	8 km
SF11	440	10 km
SF12	250	12 km

against the gateway sensitivity values. The spreading factor is assigned based on this comparison. If the received power is below the sensitivity threshold of gateway then the highest spreading factor is assigned to the end device. After this initial assignment of the spreading factors, if any collision occurs among two nodes with same spreading factors, a new spreading factor is assigned to one of the nodes such that the relation between the received power and sensitivity is maintained. The authors of [16] proposes an approach to allocate the spreading factors, channels and transmit power to the end devices. The approach works by sorting and grouping all end devices according to their distance from the gateway. Parameters are assigned in such a way that the network performance is improved. The approach was able to achieve a significant improvement for devices on the edge of the cell but has a high complexity and also does not guarantee QoS requirements. Another approach to allocate the spreading factors is discussed in [3]. The main principle of this approach was to distribute the end devices on various spreading factors. By doing so the collision domain gets changed reducing the number of collisions. The authors calculated an optimum fraction of devices to be distributed among spreading factors, maximizing the performance of the network.

The above works discuss QoS requirements for the IoT applications and the performance issues of LoRaWAN networks. Meeting QoS requirements gets more challenging as the number of devices increases in the network. The ADR mechanism is also not able to improve the network performance. According to the ADR mechanism, all devices start with the highest spreading factor and at higher traffic loads, this will add to the number of collisions. The algorithm will either take too long to converge or it will fail to converge due to the huge number of collisions. Different SF assignment schemes are also discussed to improve the performance of the networks but the QoS requirements are not addressed. The authors in [17] discuss the research gap in LoRaWAN technology regarding the QoS requirements. Our proposed approach tries to fill this gap and improve the performance of LoRa networks while satisfying the QoS requirements posed by IoT applications.

IV. QoS Based SF Allocation

We now describe the proposed mechanism for assigning spreading factors according to the QoS requirement. The main idea is to assign spreading factors to devices based on the application QoS requirements as opposed to using the default ADR mechanism that allocates higher SFs to end-devices at increasing distances from the gateway to meet the required RSSI and SNR thresholds [18].

A. Defining QoS Requirements

For ease of understanding, we define two levels of QoS requirements considering two different types of IoT applications running on the application server:

1) ***High priority (HP) applications:*** Applications that require a guaranteed QoS such as those that are characterised by the maximum latency or minimum delivery rate. Examples of such applications include fire and intrusion alarms.
2) ***Low priority (LP) applications:*** Applications that have relaxed latency and delivery requirements such as environmental monitoring sensors used for HVAC control, smart agriculture, etc.

Both types of applications consist of sensing devices that transmit their status to the server via gateway through LoRa link. We assume that the QoS requirements can be achieved by *a minimum probability of success for packet delivery* for the HP devices. This assumption is analogous to the assumptions made in [11], which preempts the low priority traffic to serve the high priority traffic. Preempting the low priority traffic to serve the high priority traffic means sacrificing all the low priority traffic for the sake of high priority traffic. Instead, we consider the high priority traffic must have a certain probability of success as the strict QoS requirement but for the low priority traffic there is no such strict QoS requirements.

B. Network Model

We assume a network comprising N devices in the network that are uniformly distributed over a circular area of radius r, as shown in Figure 2. A fraction β of total devices consist of high priority devices and the remaining are low priority devices. A gateway is placed at the center of this disc. The radius is small enough that a transmission from any device configured with any spreading factor can reach the gateway. It is assumed that the gateway has enough resources available to demodulate any number of received transmissions that are above the sensitivity threshold. The traffic in the network is generated according to Poisson distribution.

C. Principle of Operation

Due to the proximity of devices to the gateway in the assumed LoRa cell, the fastest data rate or the lowest spreading factor i.e., SF7 can be assigned to all the end devices. However, as observed in studies such as [13], [3], [12], the probability of collisions may be reduced by distributing the devices over multiple spreading factors. The spreading factors being orthogonal to each other do not collide with one another and by configuring devices on different spreading factors changes the collision domain of the devices.

Assuming a Poisson arrival model with average arrival rate of λ, the packet success probability P_s can be calculated using the analysis applied to the ALOHA protocol

$$P_s = P(k = 0) = e^{-2G} \tag{2}$$

where $P(k)$ represents the Poisson probability of k transmissions per packet duration, and G is the average number of transmission attempts per frame time. In our case, G can be calculated as

$$G = \frac{D * ToA}{Total\ time\ period} \tag{3}$$

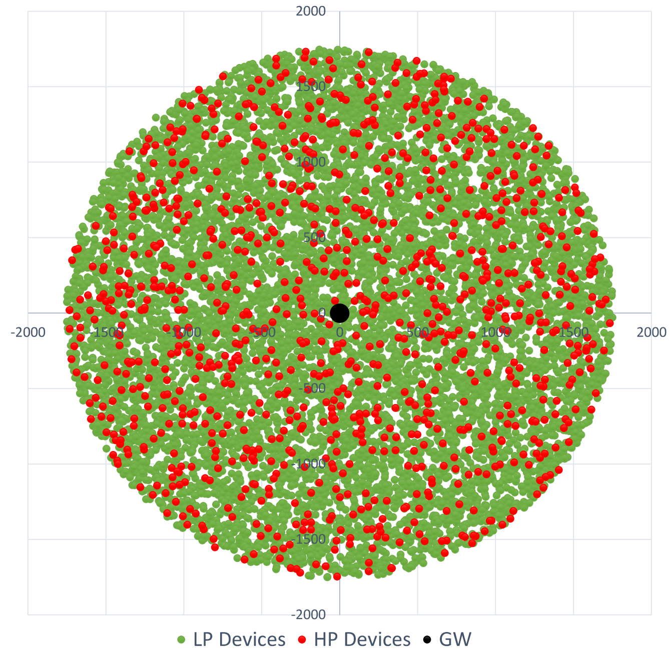


Fig. 2. Proposed network model with single gateway serving a circular area of radius 1700 meters

TABLE II
TIME ON AIR FOR VARIOUS SFS FOR A PAYLOAD OF 8 BYTES WITH 8 PREAMBLE SYMBOLS.

Spreading Factors	Time on Air (sec)
SF7	0.0361
SF8	0.06195
SF9	0.1239
SF10	0.24781

where D is the total number of devices attempting to transmit during the total time period, and ToA is the time-on-air, i.e., the duration for which a packet stays over the transmission medium.

Using the values of ToA from Table II and equations (2) & (3), and setting value of probability according to the QoS requirements of high priority traffic, the value D can be calculated. The value D in this case will represent the maximum number of devices that can be assigned to a specific spreading factor satisfying the QoS requirements. The values calculated for a 90% probability of success and are tabulated in Table III.

From Table III it can be inferred that the maximum number of devices that can be configured on a spreading factor decreases as we move from lower spreading factor to higher. This corresponds to the fact that as we move from lower spreading factor to higher spreading factor the packet length increases and that increases the probability of collision.

TABLE III
MAX. NUMBER OF DEVICES THAT CAN BE CONFIGURED ON VARIOUS SFS TRANSMITTING PAYLOAD OF 8 BYTES WITH MORE THAN 90% PROBABILITY OF SUCCESS.

Spreading Factors	Max. number of devices
SF7	870
SF8	500
SF9	250
SF10	120

D. Rules for SF allocation

In this section we propose the rules for allocating the spreading factors to high priority traffic devices and low priority traffic devices. Two different Spreading Factor Allocation (SFA) schemes are considered, which are

- ***SFA-1:*** Under this scheme the network server will begin with allocating higher spreading factors to high priority traffic devices and the remaining to low priority traffic devices.
- ***SFA-2:*** This scheme is reverse of SFA-1 and will begin with allocating lower spreading factors to high priority traffic devices and the remaining to low priority traffic devices.

Both schemes try to maintain the optimum ratio of 64:36 of devices between two adjacent spreading factors [3].

Let $p = \beta N$ and $q = N - p$ be the number of high and low priority devices in the network, respectively. We consider the QoS requirements for the HP application be defined by the minimum threshold of the packet success probability x. Consider that the number of devices supported by SF7, SF8, SF9, and SF10 to meet this QoS requirement are denoted by a, b, c, and d, respectively. It can be inferred that $d < c < b < a$.

With these, the rules for SFA-1 are as follow:

1) For a given N and β, if 36% of $p < d$ then p devices will be distributed according to 64:36 ratio between SF9 and SF10 respectively, and the q devices will be distributed according to 64:36 ratio between SF7 and SF8, respectively.
2) For a given N and β, if 36% of $p > d$ then d number of devices out of p devices will be configured on SF10 and remaining $p - d$ devices will be configured on SF9, given that $p - d < c$. The q devices will be distributed between SF7 and SF8 according to 64:36 ratio.
3) If $p > c + d$ but $p < b + c + d$ then all the low priority devices will be configured on SF7 and high priority devices will be configured on SF8, SF9 and SF10.
4) If $p > b + c + d$, decrease the x and find the values of b, c, and d for this new x.
5) Configure all p devices on SF8, SF9, and SF10 in optimum fraction. The thresholds b, c, and d must be maintained at all times.

As mentioned earlier, SFA-2 is a reversed version of SFA-1, it operates as:

1) For a given N and β, if 64% of $p < a$ then p devices will be distributed between SF7 and SF8 according to 64:36 ratio, and the q devices will be distributed between SF9 and SF10 according to 64:36 ratio.
2) For a given N and β, if 64% of $p > a$ then a number of devices out of p devices will be configured on SF7

and remaining $p-a$ devices will be configured on SF8, given that $p-a<b$. The q devices will be distributed between SF9 and SF10 according to 64:36 ratio.
3) If $p>a+b$ but $p<a+b+c$ then all the low priority devices will be configured on SF10 and high priority devices will be configured on SF7, SF8 and SF9.
4) If $p>a+b+c$, decrease the x and find the values of a, b, and c for this new x.
5) Configure all p devices on SF7, SF8, and SF9 in optimum fraction. The thresholds a, b, and c must be maintained at all times.

V. Performance Evaluation

The performance of the proposed allocation schemes have been evaluated mathematically as well as via network simulations. We assume the QoS requirement for the high priority devices to be defined by a minimum packet success rate of 90% or above. For comparison, we evaluate the packet success probabilities of the high priority and low priority devices as evaluated with the proposed SFA-1 and SFA-2 schemes with that of the legacy LoRaWAN network as well as that using the optimum distribution of spreading factors as presented in [3].

A. Mathematical Evaluation

The packet success probabilities of the proposed approaches can be evaluated using the Poisson's distribution equation. In order to find the performance of proposed approaches, end devices need to be configured on the different spreading factors according to the rules stated in section IV D. Once the configuration is complete for any number of total devices, the probability of success for each spreading factor can be found using equations 2 and 3. A weighted average of probability of successes for different spreading factors on which high priority device are configured gives the probability of success for high priority devices. The same process is followed to get the probability of success for low priority devices. This process was done for different number of total devices in the network.

B. Simulation model

The Network Simulator-3 (NS-3) [19] simulator was used for the analysis. Details of the simulation model are presented in [20]. To simulate the real-life scenario, the model considers uniformly spread end devices on a circular disc of radius 1700 meters. Selection of HP devices as well as assignment of spreading factors are done randomly, using uniform distributions. The network traffic is generated according to Poisson distribution. The simulation is based on the configuration parameters tabulated in Table IV.

The simulations were conducted for number of end devices varying from 1000–10,000 devices increasing in steps of 500 devices. We measure the average probability of success for high priority and low priority devices individually from each simulation.

TABLE IV
Configuration parameters for simulation

Number of end devices, N	1000-10,000 in increments of 500
Number of Gateways	1
Number of Channels	1
Spreading factors	7, 8, 9, 10
LoRa cell	Circular disc of radius 1700 meters
Simulation time	600 Seconds
Application Payload	8 bytes
Packet generation model	Poisson Distribution
Transmit Power	Default power 14dBm
Mean packet arrival time	$600/N$ Seconds

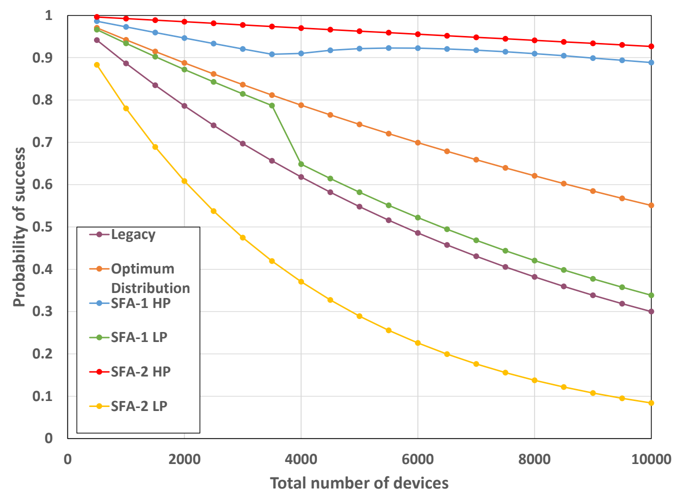


Fig. 3. Probability of success for various approaches with increasing number of total devices in the network as obtained from the mathematical model with $\beta = 0.1$

C. Results and discussions

This section presents the performance results obtained mathematically as well as from simulations. The legacy LoRaWAN configures the end devices on spreading factors based on the RSSI and SNR, whereas the optimum distribution approach distributes spreading factors that maximizes the average packet success rate. Both of these schemes consider all transmission to have same level of priority.

Figure 3 shows the results from the mathematical evaluation of the packet success rates for the proposed SFA approaches as well as those obtained for the legacy network and the optimum distribution approach. It can be observed from the plot that the legacy approach achieves success rates of about 95% for fewer number of total devices (500 devices). Even for 1000 devices in the network the legacy approach does not seem to satisfy the QoS requirement. The optimum distribution approach achieves better performance than the legacy approach. It achieves a success rate of 97% for 500 devices in the network but this approach too fails to satisfy the QoS requirements once the number of devices exceeds 1500 devices. The SFA-1 and SFA-2 both are able to satisfy the QoS requirements. The SFA-1 in addition to satisfy the QoS requirements was also able to achieve better performance for low priority devices. It can also be observed that for SFA-1, the probability of success for

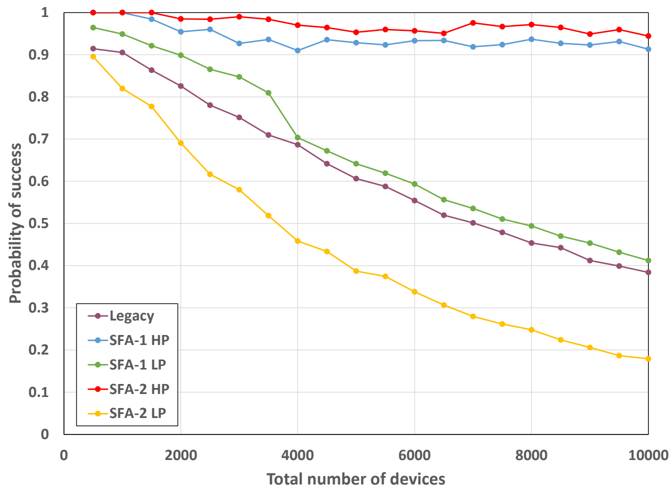


Fig. 4. Probability of success as obtained from simulation of proposed and legacy approaches with $\beta = 0.1$.

low priority devices with increasing number of total devices is decreasing linearly, except for 4000 total number of devices, where a sudden degradation in performance is observed. This sudden degradation occurred due to fact that SF10 and SF9 combined could not serve the increasing high priority devices and hence, according to the rules, they need to be configured on SF8 in addition to SF10 and SF9, limiting low priority devices to SF7 only. This resulted in the observed sudden degradation in probability of success.

Both the legacy and the optimum distribution approaches does not differentiate devices on priority levels and hence all the devices whether high priority or low priority will have the same probability of success. The optimum distribution performs better than the legacy because it distributes devices on all spreading factors according to a optimum fraction that improve the performance as compared to legacy.

The proposed allocation schemes considers low priority and high priority devices differently. As the SFA-2 starts configuring the high priority devices from lowest spreading factor and significantly higher number of low priority devices on high spreading factors, the high priority devices enjoys very high probability of success while the performance of low priority devices substantially degrades.

The higher spreading factors can serve lesser number of devices as compared to lower spreading factors for the same performance levels, hence by configuring high priority devices in the network, that are fewer in number as compared to low priority devices, to the higher spreading factors and low priority devices on lower spreading factors, the SFA-1 approach tries to optimize the performance of both the high and low priority devices.

As the spreading factors are not perfectly orthogonal, some collisions can occur due to imperfection in orthogonality among spreading factors [21]. In order to take these collisions into account both of the proposed approaches were simulated. The results of the simulation, as plotted in figure 4, indicate a slightly better performance for all schemes as the simulator considers the power capture effect that is not included in the mathematical model.

Figure 5 shows the performance comparison of SFA-1 and SFA-2 for higher values of β. For both of the allocation schemes, it can be observed that whenever the number of high priority increases more than the respective thresholds, $b+c+d$ in case of SFA-1 and $a+b+c$ in case of SFA-2, neither of the allocation scheme is able to satisfy the QoS requirements, and hence x was reduced. It is also intuitive that for a fixed N, p is greater when $\beta = 0.3$ as compared to when $\beta = 0.2$, resulting in lesser number of q devices in the former case. Hence, the performance of low priority devices should improve and the same can be observed for SFA-1 but in case of SFA-2, as N exceeds 5000 devices, the performance starts to degrade even further. This degradation stems from the fact that in SFA-2, lower spreading factors that have higher thresholds, are allocated to p devices that are lesser in number as compared to q devices. SFA-2 provides better QoS for high priority devices when compared with SFA-1 by almost sacrificing the performance of low priority devices. However, SFA-1 tries to optimize the performance of low priority devices while trying to maintain requirements of high priority devices. Hence, SFA-1 approach is better than SFA-2 in terms of success rates for low priority devices while maintaining QoS for high priority devices.

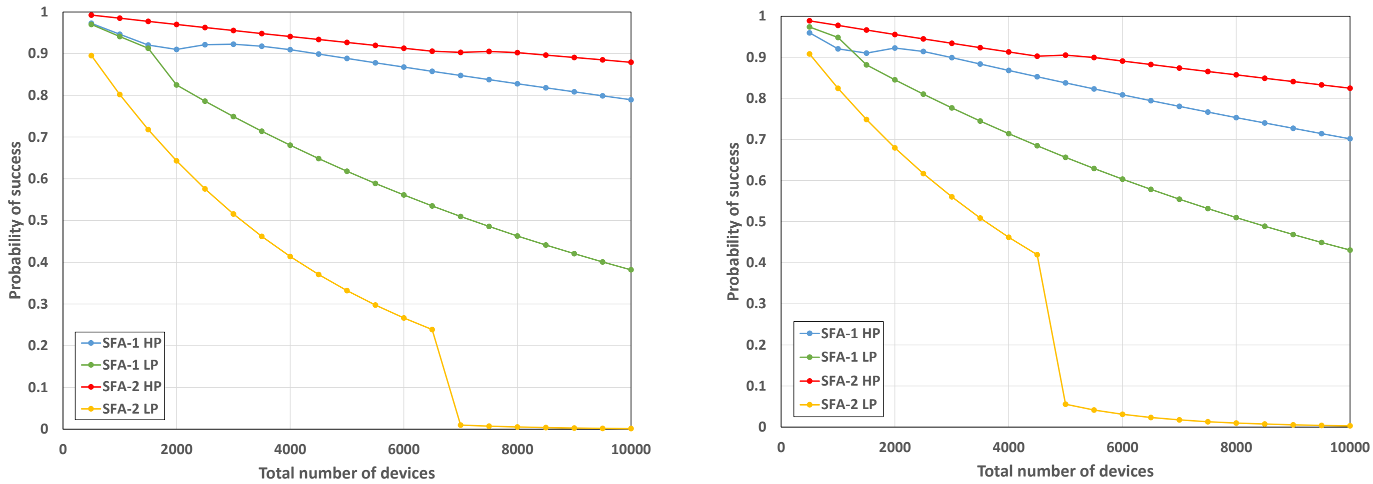
This work has considered devices within 1700 meters radius but full coverage potential of LoRa gateway can also be considered. In that case, the devices that are farther away from the gateway can only be configured to higher spreading factors. Hence, the high priority devices that are at the maximum coverage distance can only be configured to SF10, but according to SFA-2, high priority devices will be configured on lower spreading factors so high priority devices that are farther from the gateway will not be served and hence the QoS requirements may not be satisfied. However, SFA-1 configures high priority devices on higher spreading factors and low priority devices on lower spreading factors hence, will be able to scale to full coverage potential of the gateway.

Hence, the SFA-1 approach achieves better performance than other approaches.

VI. Conclusion and Future Works

In this paper, we present design considerations for a LoRaWAN network to meet specific QoS requirements for IoT applications. Specifically, we propose spreading factors assignment schemes that try to meet a minimum packet success rate for end devices that require high priority service while still providing adequate packet success rate for other devices, i.e., those for which the packet success rate is not highly critical. Such differentiated QoS requirements are typical of IoT applications in smart city environments.

Legacy LoRaWAN networks typically assign spreading factors depending on the signal to noise ratios signal of end devices. A higher spreading factor enables communication at a lower SINR at the cost of lower data rates. Here, we



(a) β=0.2 (b) β=0.3

Fig. 5. Probability of success with both proposed allocation schemes with (a) β=0.2, and (b) β=0.3 with varying total number of devices

propose spreading factor assignment schemes that utilize their orthogonality to limit the number of contending end devices using the same spreading factor. Two different spreading factor allocation schemes are introduced that assign all high priority devices to spreading factors such that the packet success probability is within acceptable limits. The approaches were evaluated mathematically as well as using a network simulator. It was observed that by assigning the spreading factors according to the proposed SFA-1 approach the network was able to satisfy the QoS requirements of the IoT applications that can be deployed in smart cities.

This work considers transmissions on single channel. In a multi-channel deployment the end devices performs frequency hopping for every transmission. They can select transmission channels randomly. In our future works, we will be working towards an approach to assign transmission channels according to the QoS requirements.

References

[1] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, “Long-range communications in unlicensed bands: the rising stars in the iot and smart city scenarios,” *IEEE Wireless Communications*, vol. 23, no. 5, pp. 60–67, 2016.

[2] R. Duan, X. Chen, and T. Xing, “A qos architecture for iot,” in *2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*, 2011, pp. 717–720.

[3] S. Aggarwal and A. Nasipuri, “Improving scalability of lorawan networks by spreading factor distribution,” in *SoutheastCon 2021*, 2021, pp. 1–7.

[4] “Lora and lorawan: Technical overview,” [Online]. Available: https://lora-developers.semtech.com/library/tech-papers-and-guides/lora-and-lorawan/.

[5] S. Abboud, N. el Rachkidy, A. Guitton, and H. Safa, “Gateway selection for downlink communication in lorawan,” in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, 2019, pp. 1–6.

[6] “Sx1301 - datasheet (accessed on 10/24/2021).” [Online]. Available: https://www.semtech.com/products/wireless-rf/lora-core/sx1301.

[7] LoRaWAN Specification v1.0, LoRa Alliance, Inc. 2400 Camino Ramon, Suite 375 San Ramon, CA 94583, 2015.”.

[8] “Lora alliance (accessed on 10/31/2021).” [Online]. Available: https://lora-alliance.org/about-lora-alliance.

[9] S. Li, U. Raza, and A. Khan, “How agile is the adaptive data rate mechanism of lorawan?” in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 206–212.

[10] A. Dvornikov, P. Abramov, S. Efremov, and L. Voskov, “Qos metrics measurement in long range iot networks,” in *2017 IEEE 19th Conference on Business Informatics (CBI)*, vol. 02, 2017, pp. 15–20.

[11] I. Awan, M. Younas, and W. Naveed, “Modelling qos in iot applications,” in *2014 17th International Conference on Network-Based Information Systems*, 2014, pp. 99–105.

[12] 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.

[13] S. Aggarwal and A. Nasipuri, “Survey and performance study of emerging lpwan technologies for iot applications,” in *2019 IEEE 16th International Conference on Smart Cities: Improving Quality of Life Using ICT IoT and AI (HONET-ICT)*, 2019, pp. 069–073.

[14] T. Yatagan and S. Oktug, “Smart spreading factor assignment for lorawans,” in *2019 IEEE Symposium on Computers and Communications (ISCC)*, 2019, pp. 1–7.

[15] S. Farhad, M. A. Lodhi, W. U. Khan, and F. Masood, “An adaptive and lightweight spreading factor assignment scheme for lorawannetworks,” in *2020 14th International Conference on Open Source Systems and Technologies (ICOSST)*, 2020, pp. 1–6.

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

[17] A. Dvornikov, P. Abramov, S. Efremov, and L. Voskov, “Qos metrics measurement in long range iot networks,” in *2017 IEEE 19th Conference on Business Informatics (CBI)*, vol. 02, 2017, pp. 15–20.

[18] “Lorawan – simple rate adaptation recommended algorithm (accessed on 10/24/2021).” [Online]. Available: https://www.thethingsnetwork.org/forum/uploads/default/original/2X/7/7480e044aa93a54a910dab8ef0adfb5f515d14a1.pdf.

[19] “Network simulator-3,” [Online]. Available: https://www.nsnam.org/.

[20] “Lorawan simulation model for ns-3 (accessed on 01/24/2021).” [Online]. Available: https://github.com/signetlabdei/lorawan.

[21] 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.