Distributed Air-Time Reduction In Multi-Hop LoRa Networks With Multiple Spreading Factors

Chawinrod Chimma, Anan Phonphoem, Aphirak Jansang, Withawat Tangtrongpairoj and Chaiporn Jaikaeo*
Department of Computer Engineering, Kasetsart University, Bangkok, Thailand 10900
Email: chawinrod.chim@ku.ac.th, anan.p@ku.ac.th, aphirak.j@ku.ac.th, withawat.t@ku.ac.th, chaiporn.j@ku.ac.th*

2023 International Conference On Cyber Management And Engineering (CyMaEn) | 978-1-6654-9329-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/CyMaEn57228.2023.10050943

Abstract— **LoRa is a physical-layer communication technology usually used for Internet of Things (IoT). LoRa has one specific parameter not found in typical wireless technology, called spreading factors (SF). The lowest SF, SF7, has the highest data transfer rate, so it is common to configure all the end devices to use SF7. On the other hand, for a LoRa network that spans a large area, higher SFs can be used to cover larger distances. LoRa's radio modulation is based on the chirp spread spectrum technique which allows multiple devices using different SFs to transmit data in simultaneously. Even though LoRa supports long-distance data transmission (up to 15 km line-of-sight and 1.5 km in outdoor urban scenarios), single-hop networks do not support the coverage range required by certain scenarios such as smart cities, in which multi-hop networks are more appropriate. However, managing devices in multi-hop networks requires more effort. Centralized, manual configuration of devices become impractical in large-scale deployments. In this paper, we propose a distributed algorithm that improves the capacity of a multi-hop LoRa network. The algorithm arranges the end devices in a tree topology and assigns different SFs to different branches of the tree so that transmissions can be carried out in parallel at much as possible, hence minimizing total time on air. From simulation experiment, the algorithm can reduce time on air of the system up to 41%.**

Keywords-component; LoRa; Multi-Spreading Factor; Multi-Hop; Air-Time Reduction

I. Introduction

The Internet of Things (IoT) is used widely in a network of home appliances, smart cities, manufacturing transportation, and logistics. For the upcoming era of IoT, smart systems for applications are expected to be widely deployed across the world [1]. The IoT device market is growing considerably. Millions or billions of end devices are utilized to gather real-time data from the environment [2, 3]. Low-power wide area networks (LPWAN), such as LoRa, Sigfox and Weightless, are types of wireless telecommunication technologies designed for a wide area network with low power that have drawn much attention by offering connectivity over a large area. LPWAN technologies support low rate and robust modulation to achieve a long communication range with limited power sources. The market for LPWAN technologies is expected to be huge in the future [1] .

LoRa is the popular LPWAN IoT. LoRa supports communication ranges up to 15 km line-of-sight and 1.5 km in outdoor urban scenarios. LoRa uses the chirp spread spectrum technique utilizing wideband linear frequency modulation. Parameters such as spreading factors (SF) and coding rate (CR) can be customized. The SF setting has a direct impact on the communication performance of LoRa. Valid SF values are between 7 and 12. A larger SF increases the time on air and improves communication range but doubles transmission duration, which increases energy consumption and reduces the data rate. Unlike other communication technologies, LoRa allows simultaneous communication over the same frequency channel with different SF configurations that are orthogonal to each other. This technique could be used to expand the LoRa network to support more devices.

Previous work has shown that we can find an optimal combination of SF settings that maximizes the density of end devices with an acceptable success probability in a single hop or a star-topology network with a single gateway. However, with urban environments and wide deployment areas, single-hop networks cannot solve these communication range problems. Multi-hop networks can be deployed to overcome the limited communication range of a single-hop network [4]. However, the management of such networks gets more complicated as the number of end devices and/or gateways increases. Centralized, manual configuration of devices become impractical in large-scale deployments, so distributed techniques are preferable.

To serve the requirements of future IoT systems we aim to improve the efficiency of end devices and gateways in multi-hop LoRa networks in terms of channel occupancy. Thus, we propose a distribution algorithm where end devices are self-configured with SF settings and path to a gateway to reduce the overall data packets' time on air.

The rest of the paper is organized as follows. Section II reviews the LoRa technology and previous work attempting to improve single-hop and multi-hop LoRa networks. Section III describes our model and how it works. Section IV evaluates and discusses the results of our algorithm. Section V concludes our work with future research directions.

II. LITERATURE REVIEW

LoRa is a physical-layer specification patented by Semtech. It integrates Forward Error Correction (FEC) with a chirp spread spectrum modulation scheme. Its design allows for long communication ranges, reaching up to 15 km in line-of-sight rural areas and 1.5 km in outdoor urban scenarios [5]. A LoRa device has several configurable parameters: transmission power (TP), bandwidth (BW), carrier frequency (CF), spreading factor (SF), and coding rate (CR), which can be chosen to respond to the transmission requirement, energy consumption, transmission range, and resilience to noise [6]. SF is one of the most important parameters of LoRa. It is the ratio between the symbol rate and chip rate. The number of chips per symbol is calculated as 2^{SF}. For example, at SF7, the value of chips per symbol is 2^7, or 128. Each increment in SF halves the transmission rate but doubles the transmission duration and ultimately energy consumption. Zorbas et al. [7] presents an optimization model to maximize end device capacity by assigning end devices with different SF configurations. It can increase the number of nodes in the system up to 700% compared to the case where only a single SF is used.

Multi-hop communication provides flexibility in terms of network connectivity and coverage but has great impact on throughput and latency. Specifically, the capacity at the gateway for each of the SF used by end devices in a single-hop network is determined only by the number of end devices and their data rates. In a multi-hop LoRa network, other factors must also be taken into consideration such as loop prevention. Azhari *et al.* [8] proposes a simple protocol for multi-hop LoRa networks that prevent forwarding loops by enforcing data to be sent from a higher-level end device to a lower-level end device node only. Farooq *et al.* [9] presents a cluster-based layering for multi-hop LoRa networks for uplink multi-hop communication. This work also relies on the leveling approach to manage the topology. However, the capacity of the entire network highly depends on the topology, which has not yet been considered in these works.

LoRa network capacity can be improved when multiple communication parameters are properly adjusted. Hence, previous efforts have investigated this resource allocation issue under different LoRa network architectures and assumptions, specifically, SF assignment in LoRa networks [10]. Based on previous work on multi-hop LoRa networks with single SF and multiple SF, it has been found that the LoRa characteristics of multi-hop with multiple SF can be greatly beneficial in terms of energy consumption [11], coverage, and increasing node density [12].

Zhu et al. [12] proposes an algorithm that can decrease the time on air in multi-hop LoRa networks by employing the parallel transmission characteristics of LoRa. This work presents a centralized algorithm to manage topology and SF configuration from the fixed SF setting. The result of the algorithm shows that using multiple SF can greatly decrease the time on air of the system. However, such a centralized algorithm where a gateway is responsible for all computation and decision is not suitable for large-scale deployments. Our work proposes a distributed algorithm that allows end devices to make decisions based on partial information received from the gateway.

III. METHODOLOGY

A. *System Model*

Our LoRa network model consists of a single gateway and a set of end devices connected to the gateway in a multi-hop fashion with a tree topology. The gateway and end devices are stationary and configured with the same carrier frequency, bandwidth, and coding rate. As long as packets are transmitted in the configured frequency channel, the gateway is capable of receiving and decoding multiple packets of different SF simultaneously. Different SF may be assigned to different end devices but each of the end devices can only receive packets over the same SF it is configured to use. The entire system must be reachable via SF7 communication, either directly or indirectly, which is initially used by the gateway to collect information from and send SF reconfiguration commands to the end devices. Eventually the network will end up with multi-SF configurations where end devices belonging to the same subtree from the gateway will be configured with the same SF, as shown in Figure 1.

In addition, we assume that every end device always has measurement data to be sent towards the gateway in fixed-size packets.

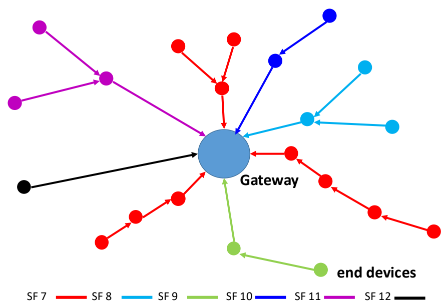

Figure 1. System model

B. *Design Concept of Algorithm*

Our objective is to minimize the overall time on air of the system by using characteristics of LoRa that allow parallel transmissions of multiple SFs. The idea is to separate all the transmissions into different SFs to balance time on air. The design of our algorithm relies on three mechanisms. First, as the algorithm has been designed as a distributed algorithm, each end device must make a decision based on partial information from the gateway and information learned from other surrounding devices. This also implies that when an end device's configuration is changed, the device must update its neighbors with the new information. Second, to prevent loops, every end device will be assigned a level number representing the number of hops from the gateway for each SF being used. Each end device is allowed to send packets to another end device in a lower level only. And third, as mentioned in the system model, end devices are unable to receive packets of different SF than what they are using. The algorithm will assure that end devices belonging to the same branch of the tree topology are assigned with the same SF.

C. Balanced-SF Model

We know that SF7 is the best setting for the highest bitrate. However, with the concept of parallel transmissions, transmitting packets with higher SF can help reduce the overall time on air. For example, if the network requires ten SF7 transmissions, these transmissions must be carried out sequentially. Alternatively, we could use eight SF7 transmissions along with two SF8 transmissions so that some transmissions can be performed in parallels, reducing the overall transmission time. TABLE I. compares the time-on-air characteristics of a 25-byte packet transmitted at bandwidth of 125 kHz over different SF settings [13].

TABLE I. TIME ON AIR AND RATIO WITH RESPECT TO SF7 WHEN THE PACKET SIZE IS FIXED AT 25 BYTES

SF	Time on air (ms)	Airtime ratio w.r.t. SF7 (c_i)
7	36.6	1
8	64	1.749
9	113	3.087
10	204	5.573
11	372	10.164
12	682	18.634

It can be observed that when increasing SF by 1, the airtime will increase by roughly a factor of 2. Hence, the number of transmissions in SF n+1 should be approximately half of that of SF n. Figure 2. illustrates a situation where some end device transitions from SF7 to SF8, which benefits from a longer distance and a parallel transmission. The total airtime of configuration (a) is 36.6×5 = 183 seconds, whereas the total airtime of configuration (b) is **max**(36.6×3, 64×1) = 109.8 seconds.

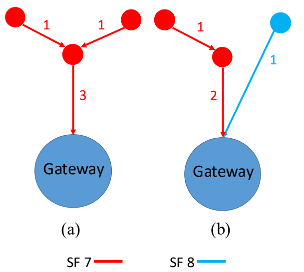

Figure 2. Example of SF balancing: (a) 5 sequential SF7 transmissions, and (b) one device transitioning to SF8, resulting in 3 sequential SF7 transmissions and 1 SF8 transmission in parallel

To minimize the total airtime in the system, we formulate the total airtime, T_i, of all sequential transmissions of SF$_i$ as

$$T_i = n_i t_i, \tag{1}$$

where i = 7, 8, ..., 12, n_i is the total number of SF$_i$ transmissions in the system, and t_i is airtime used by SF$_i$ to transmit a single packet.

We want to distribute all the transmissions among every SF. Let SF$_i$ be assigned the fraction R_i of all the transmissions. The total airtime for each SF$_i$ becomes

$$T_i = n_i t_i = nR_i t_i, \tag{2}$$

where n is the total number of transmissions in the system. Using the airtime ratios, c_i, from TABLE I. we can write t_i in terms of SF7's airtime, t_7, as

$$t_i = c_i t_7. \tag{3}$$

To balance all SF's airtime, we want the following equation to hold:

$$T_7 = T_8 = \cdots = T_{12}. \tag{4}$$

It then follows that:

$$nR_7 t_7 = nR_8\, c_8 t_{7_8} = \cdots = R_{12} c_{12} t_7. \tag{5}$$

The last constraint is that the ratios of all SF must sum up to one, i.e.,

$$R_7 + R_8 + R_9 + R_{10} + R_{11} + R_{12} = 1. \tag{6}$$

Solving (5) and (6) results in transmission ratios of all SF that balance and minimize the overall airtime. These values are not fixed but vary with packet size, bandwidth, and coding rate, so they need to be calculated for each deployment. TABLE III. presents an example of these ratios with the settings used by our simulation experiments.

D. Initialization Phase

This phase serves as an initial phase to determine the level of each end device as a mechanism to avoid loops in packet forwarding. Each end device also learns the maximum number of hops to reach the gateway. The algorithm starts from the gateway with level equal to 0. The gateway starts broadcasting an announcement in SF7 to all the surrounding end devices. Upon receiving the announcement, an end device records the sender as its parent, increases the level by 1, and rebroadcasts. All announcements containing levels equal to or greater than each end device's level will be ignored. Once a device does not hear any further rebroadcast, it starts sending transmission count, which is one for a leaf node, toward its parent. The parent combines all the transmission counts and relays the information upward the tree branch until the gateway is reached. Doing so allows the gateway to obtain information about the aggregate transmission number for each branch connected to itself. The gateway starts the whole process with the higher SF until SF12 is reached. This process also allows each end device to learn about its immediate neighbors using different SFs. Figure 3. illustrates level assignments as a result of this phase.

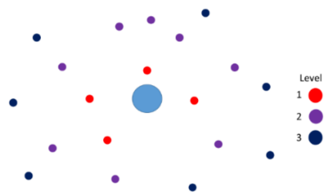

Figure 3. Levels assigned to end devices after Initialization Phase

E. SF-Balancing Phase

This phase relies on the principle of the balanced-SF model to branch nodes into other SFs according to the ratios we have calculated. At the end of the initialization phase, the gateway has learned about the number of transmissions required by each tree branch if SF7 were to be used by every device. It then applies each SF's transmission ratio, R_i, to obtain the ideal transmission count for each SF. Based on these ratios, certain branches may be requested by the gateway to switch to a higher SF. Otherwise, all the ratios and transmission counts are passed to the root of each branch. Upon receiving this information, each device recursively performs the same operation on its subtree. Because a higher SF means a longer communication range, some branches that already switch to a higher SF may also change the connectivity to another branch closer to the gateway. Figure 4. illustrates SF settings and connectivity before and after executing this phase.

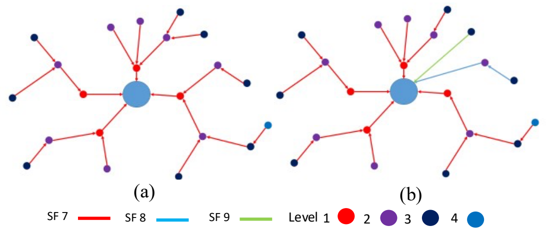

Figure 4. Example of before and after Algorithm

F. Schedule Phase

To ensure no collision, two packets of the same SF must not be transmitted at the same time. The gateway assigns each branch with a range of time slots. Each device located down the branch applies the same process to assign time slots in each of its branches. Eventually, all the devices know the time slots they are allowed to transmit a packet. Figure 5. illustrates the process.

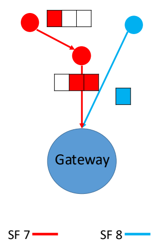

Figure 5. Gateway and end devices assigning time slots to their subtrees

IV. Evaluation

We have conducted a simulation experiment to evaluate our proposed algorithm.

A. Simulation Environment and Parameters

In this study we used LoRaSim, a discrete event simulator implemented with Python and SimPy [14]. We modified the original code to match our model with 868 MHz ISM band and the path loss model for smart city as presented in [15, 16]. Parameters used in the simulation are shown in TABLE II. .

TABLE II. Simulation Parameters

Parameter	Value
Carrier frequency	868 MHz
Bandwidth (BW)	125 kHz
Spreading factor (SF)	7-12
Packet length	25 bytes
Transmission power	14 dBm
Number of nodes	225 and 400
Terrain size	225 and 400 km^2

B. Running Simulation

In our model, the algorithm starts with the initialization phase to make each end device know the information of other surrounding devices and learn the level from the gateway at each SF. An example of this phase is shown in Figure 6.

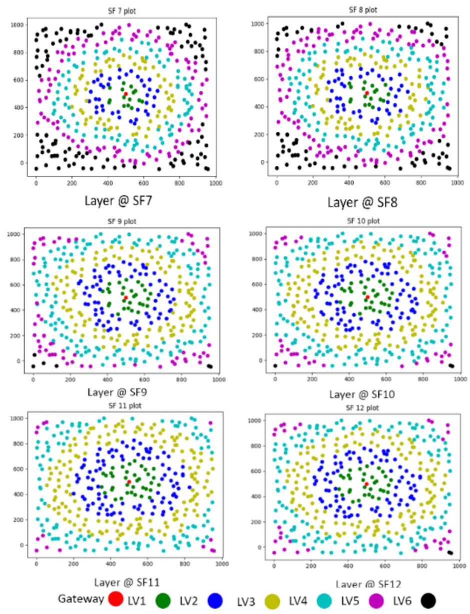

Figure 6. Results After Initialization Phase

Based on the simulation parameters, the gateway also solves the equations (5) and (6) to obtain the SF ratios, whose values are shown in TABLE III.

TABLE III. Result of transmission ratios

SF	*Transmission Ratio (R_i)*	SF	*Transmission Ratio (R_i)*
7	0.4490	10	0.0806
8	0.2568	11	0.0442
9	0.1455	12	0.0239

Every node that receives a command from the gateway uses this ratio to decide whether to separate their child branch into new branch with higher SF, as shown in Figure 7.

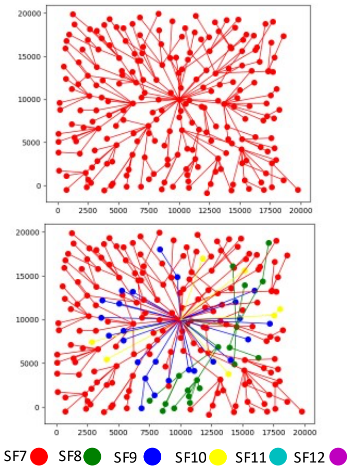

Figure 7. Illustration of end device connectivity after SF balancing phase

C. *Results and Discussion*

The results from the simulation are summarized in TABLE IV. As expected, the overall time on air has been reduced significantly, especially when there are a lot of devices or the device density is high. As we obtained 0.4490 for R_7, ideally the transmission time could be decreased up to 55.1%. However, the result shows that in some case we can decrease the time on air of the system by only about 30%. The reason is that in such scenarios some branch already has the number of transmissions higher than the given ratio, so they can't be separated from the parent.

TABLE IV. RESULT OF THE EXPERIMENT

Number of devices	**Terrain size (km^2)**	**Result compared with single SF7**	
		Average time on air (%)	***S.D.***
225	225	64.94	2.06
400	225	67.78	2.60
225	400	60.21	4.06
400	400	58.59	2.73

V. CONCLUSION

We can use LoRa's orthogonality characteristic to the benefit of parallel transmission to reduce overall time on air in multi-hop LoRa networks. The simulation results show that up to 41% of airtime could be reduce. For future directions, the algorithm's overhead and performance in realistic scenarios such as device failures should be investigated. Techniques to further optimize the results should also be explored. For instance, as some devices switch to higher SFs, the transmission range increases, thus reducing the total number of transmissions. This fact has not yet been considered when computing each SF's transmission ratio.

ACKNOWLEDGMENT

This research is supported in part by the Master's degree scholarship from Faculty of Engineering, Kasetsart University and IWING laboratory research fund.

REFERENCES

[1] K. Mikhaylov, J. Petäjäjärvi., and T. Hänninen, "Analysis of the capacity and scalability of the LoRa wide area network technology",in European Wireless (EW). 2016: Oulu, Finland.

[2] A. Emmanuel, X. Fernando, F. Hussain and W. Farjow, "Optimization of spreading factor distribution in High Density LoRa Networks", in Vehicular Technology Conference (VTC). 2020, Antwerp, Belgium.

[3] F. Peper , K. Leibnitz, C. Tanaka, K. Honda and M. Hasegawa, "High density resource restricted pulse based IoT networks". IEEE Transactions on Green Communications and Networking, 2021. 5(4): pp. 1856 - 1868.

[4] T. Yatagan and S. Oktug,"Smart spreading factor assignment for LoRaWANs", in IEEE Symposium on Computers and Communications (ISCC). 2019, Barcelona, Spain.

[5] D. Magrin, M. Capuzzo and A. Zanella,"A thorough study of LoRaWAN performance under different parameter settings". IEEE Internet of Things Journal, 2019. **7**(1): pp. 116 - 127.

[6] M. Bor, J. Vidler, U. Roedig,"LoRa for the internet of things", in International Conference on Embedded Wireless Systems and Networks (EWSN). 2016. Graz, Austria

[7] D. Zorbas, G. Papadopoulosy, P. Mailley, N. Montavonty and C. Douligeris, "Improving LoRa network capacity using multiple spreading factor configurations",in International Conference on Telecommunications (ICT). 2018, Saint-Malo, France

[8] R. Azhari, E. Firmansyah, A. Bejo, "Simple protocol design of multi-hop network in LoRa", in International Seminar on Research of Information Technology and Intelligent Systems (ISRITI). 2019, Yogyakarta, Indonesia.

[9] M. Farooq, "Clustering based layering approach for uplink multi-hop communication in LoRa networks". IEEE Networking Letters, 2020. 2(3): pp. 132 – 135

[10] S. Kim, H. Lee and S. Jeon, "An adaptive spreading factor selection scheme for a single channel LoRa modem". Sensors, 2020. 20(4): pp. 1008.

[11] S. Fahmida, V. Modekurthy, M. Rahman, A. Saifullah and M. Brocanelli, "Long-Lived LoRa: prolonging the lifetime of a LoRa network", in International Conference on Network Protocols (ICNP). 2020, Madrid, Spain.

[12] G. Zhu, C. Liao, T. Sakdejayont, I. Lai, Y. Narusue and H. Morikawa, "Improving the capacity of a mesh LoRa network by spreading factor based network clustering". IEEE Access, 2019. 7: pp. 21584 - 21596.

[13] O. Georgiou and U. Raza, "Low power wide area network analysis: can LoRa scale?" IEEE Wireless Communications Letters, 2017. 6(2): pp. 162-165.

[14] M. Bor, U. Raza, T. Voigt and J. Alonso, "Do LoRa low power wide area networks scale?", in International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM). 2016, Malta

[15] P. Jörke, S. Böcker, F. Liedmann and C. Wietfeld, "Urban channel models for smart city IoT-networks based on empirical measurements of LoRa-links at 433 and 868 MHz", in IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC). 2017, Montreal, Canada.

[16] J. Pettissalo, K. Mikhaylov, A. Roivainen, T. Hänninen and M. Pettissalo, "On the coverage of LPWANs: range evaluation and channel attenuation model for LoRa technology", in International Conference on ITS Telecommunications (ITST). 2015, Copenhagen, Denmark.