A Novel Autonomous Time-Slotted LoRa MAC Protocol with Adaptive Frame Sizes

Hanan Alahmadi, Fatma Bouabdallah
Faculty of Computing and Information Technology
King AbdulAziz University
Jeddah, Saudi Arabia
hjalahmadi1,fothman1@kau.edu.sa

Ahmed Al-Dubai, Baraq Ghaleb
School of Computing
Edinburgh Napier University
Edinburgh, UK
A.Al-Dubai, B.Ghaleb@napier.ac.uk

***Abstract*—Time-Slotted Medium Access Control protocols bring advantages to the scalability of LoRa networks as an alternative to the ALOHA access method. However, such Time-Slotted protocols require nodes synchronization and schedules dissemination under stringent duty cycles likely resulting in improper performance and limited scalability. One possible solution is to adopt decentralized approaches where nodes autonomously determine their schedules and other transmission parameters. Thus, this paper proposes a novel Time-Slotted MAC protocol, named Autonomous Adaptive Frame Size (AAFS-LoRa) protocol, that allows nodes to individually determine their transmission parameters without extensive downlink transmissions from the gateway. The proposed protocol can configure nodes by maintaining information, such as their location and the gateway location. The main contribution of the proposed protocol is the adoption of adaptive frame sizes that are large enough to accommodate nodes with common transmission parameters to mitigate collisions among them. The proposed protocol has been investigated under different operating conditions, and the experiments demonstrates that our protocol can effectively improve the network performance, in terms of latency as well as the capacity.**

***Index Terms*—LoRaWAN, IoT, TDMA, MAC protocols**

I. Introduction

LoRa, which stands for Long Range, provides long transmission range while maintaining a low-power consumption profile, enabling a wide range of Internet of Things (IoT) applications such as smart cities, smart monitoring, and smart agriculture [1]. Furthermore, LoRa has low deployment cost since it works on the unlicensed ISM band and has a high resistance to interference and massive connectivity support thanks to its novel physical features [2].

In technical terms, LoRa is a modulation technique derived from chirp spread spectrum (CSS) technology. Similar to cellular networks, LoRa adopts the star topology where a central hub called the gateway acts as a conduit to transmit messages from end nodes to a network server. LoRa physical layer provides different configurable transmission parameters that highly affect the performance of the overall network. Among these parameters are Channel Frequencies (CFs) and LoRa Spreading Factors (SFs). The number of the available CFs mainly depends on the deployed region. In Europe, the band assumed in this study, there are mainly eight uplink channels. SF represents the number of symbols per bit of payload, and it ranges from 7 to 12. Higher SFs means lower data rate and hence higher transmission time and energy consumption. On the other hand, higher SFs provides longer transmission ranges as gateways have higher sensitivity levels for higher SFs. Hence, farther nodes are obliged to use higher SFs to ensure the delivery of their packets. Furthermore, SFs are orthogonal [3], which means two or more packet transmissions on the same channel and encoded with different SFs can be received successfully at the gateway. Hence, the efficient use of the supported different Spreading Factors (SFs) and Channel Frequencies (CFs) could greatly enhance the capacity of LoRa networks.

LoRaWAN, a networking protocol, is built on top of the modulation technique of LoRa to provide networking capabilities between LoRa end devices and gateways. LoRaWAN mainly defines the Medium Access Control (MAC) based on ALOHA medium access protocol, which is known to result in high collisions under a large number of connected nodes, hence limiting the scalability of LoRa networks [4].

One promising way to enhance the network capacity is by controlling node's channel access to the shared medium. To this end, Time-Division Multiple Access (TDMA) has attracted many researchers to enhance LoRa networks capacity [5]. In other words, with appropriate combination of CF, SF, and time-slot per node, we can achieve collision-free transmissions. However, disseminating the schedule among nodes as well as synchronizing nodes with gateways could add further burden to LoRa network as nodes and gateways have limited Duty Cycle (DC).

A. Contributions

This paper Proposes a TDMA MAC protocol that allows nodes to autonomously determine their appropriate transmission parameters without extensive downlink traffic from the gateway. Furthermore, the proposed protocol configures nodes to use the minimum SF whenever it is eligible. Eligibility in this context means using the minimum SF that ensures the transmitted packet will not be received below the sensitivity of the gateway. According to that, the proposed protocol has

The authors extend their appreciation to the Deputyship for Research and Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number (IFPRC-065-612-2020) and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

almost zero Packet Error Rate (PER) as long as all nodes use the appropriate SF that ensures the delivery of their packets. The main novelty of the proposed AAFS-LoRa protocol is the spatial partitioning of nodes to achieve $S \times K$ different group of nodes that use different SF and Channel, where S is the number of available SFs and K is the number of available channels. Hence, we only need relatively small frame sizes to regulate the channel access among the nodes of the same group. In fact, even with a very large number of connected nodes (4000 nodes) we can get a relatively small frame size (not exceeding 140 slots).

B. Motivation and related work

Many TDMA-based approaches have been proposed in the literature to address the main limitations of LoRaWAN [5]. The proposed solutions can be classified into three main categories: beaconic-based, synchronized-based, and autonomous. In beaconic-based approaches, the gateway initiates the transmission by sending beacons that hold control information to nodes at the beginning of each frame. Nodes use these beacons to set up their transmission parameters and their timeslots [6]–[8]. In these algorithms, the time is divided into frames and a specific duration is assigned at the beginning of each frame for receiving beacons from the server. Nodes during the beacon period will be in a listening mode waiting until receiving a beacon to initiate their transmissions. According to that, concerns regarding the limited Duty Cycle (DC) of the server are raised especially with high number of connected nodes and large beacon sizes. Moreover, it is not compatible with LoRaWAN class A devices as the transmissions in the latter are initiated by LoRa nodes not the server.

On the other hand, in synchronized-based approaches, nodes initiate the transmissions by sending synchronization requests during their synchronization phases, which is more compatible with LoRaWAN class A devices [9], [10]. After a node sends its synchronization request, it waits for a synchronization reply from the server that includes its transmission parameters. Possible shortages of the server's DC may be experienced when large number of connected nodes need to be synchronized in short period. Similar to beaconic-based approaches, more energy consumption is expected as nodes will be in the listening mode to receive the synchronization replies from the server.

In autonomous approaches, nodes determine their transmission parameters independently without a dictation from the server [11]–[13]. In [11] and TS-LoRa [12], nodes extract their timeslot ID from their MAC addresses and DeVAddr, respectively. However, both algorithms are not fully autonomous as nodes need the frame length to calculate their slot IDs, which is broadcast by the server. In SBTS-LoRa [13], nodes autonomously determine their transmission parameters including their timeslots with the help of their locations relative to the gateway.

To the best of our knowledge, the existing protocols have either fixed timeslot duration or fixed number of slots per frame. In both cases, there will be waste duration per slot or waste timeslots per frame due to the dynamic nature of LoRa networks. Moreover, none of the mentioned protocols adopt a dynamic frame size that reflects the actual need of the network.

The rest of this paper is structured as follows. Section II describes the proposed AAFS-LoRa MAC protocol. Simulation results are provided in Section III. Finally, Section IV concludes the work with directions for future work.

II. Protocol Description

Similar to LoRaWAN networks, we assume a star network topology, where nodes are randomly distributed in a field with radius R and a gateway that is located at the center of the field. In LoRa networks, the access to the shared medium is regulated by imposing Duty Cycles (DC). Duty cycle refers to the amount of time a node or a gateway is allowed to transmit in a period of time. Hence, if the DC is 1%, which is the case of the Europe band that is considered in this study, and a node transmits a packet with ToA_{SF}, which is the Time on Air of a packet using spreading factor SF, that node should wait at least $ToA_{SF} \times 99$ before transmitting again to respect its DC.

In this work, nodes determine their appropriate transmission parameters without extensive downlink transmissions from the gateway. Indeed, allowing nodes to be aware of some characteristics of their environment will greatly help to make them smart enough to decide about their appropriate transmission parameters. Specifically, allowing each node to acquire only its own location and the one of the gateway will help to determine its transmission parameters such as the appropriate SF, frequency CF, and time-slot SID.

According to LoRa, each SF has a specific level of sensitivity at the gateway. The level of sensitivity is increasing with higher SFs. Hence, higher SFs have transmission ranges greater than the smaller ones. However, higher SFs have longer ToA compared to the smaller ones. Hence, using smaller SFs whenever eligible will result in shorter ToA, and hence lower collision rates and end-to-end delays. As a result, a higher network throughput could be achieved. Thus, in this work, we propose to divide the network field into six coronas that corresponds to the number of supported SFs as shown in Fig.1. As illustrated in Fig.1, each corona C_i is assigned a given SF according to the distance from the gateway. The range of each corona C_i represents the eligible range of the assigned SF. The eligible range of a given SF refers to the maximum distance from the gateway where nodes can securely use that SF such that their packet transmissions will not be received below the sensitivity level of the gateway. In other words, the radius R_i of each corona is actually the estimated maximum distance from the gateway where a node can use a given SF. Hence, packet transmissions from nodes that are located on a given corona will not be received below the sensitivity level of the gateway. Hence, predicting SF ranges is very important to determine the appropriate SF for each node. The following section explains the distribution of SFs among nodes.

A. SF ranges Determinations

In this paper, we assume that SFs are orthogonal, which means that simultaneous transmissions on the same frequency

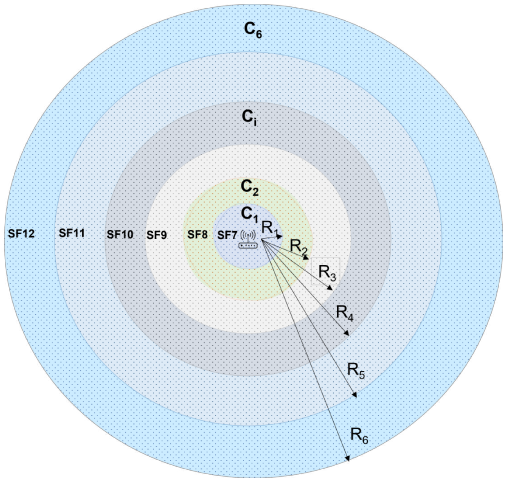

Fig. 1. Partitioning the field into overlapped coronas with different SFs.

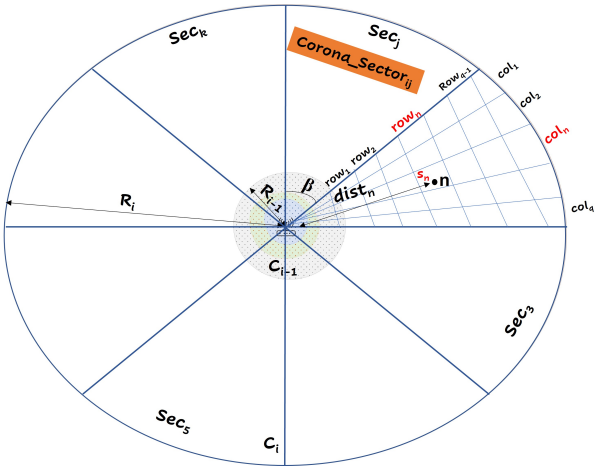

Fig. 2. Distributing CFs and slots among nodes

encoded with different SFs are successfully received at the gateway. According to that, we divide the network field into six coronas that corresponds to the number of the available SFs. Each corona C_i is assigned a given SF. The closest corona to the gateway is assigned the smallest SF, which is SF7. While we go farther from the gateway, coronas are assigned higher SFs. The radius R_i of a given corona C_i is actually the maximum eligible distance for a node to use a given SF such that its transmission is not received below the sensitivity level of the gateway.

In order for a node to determine to which corona it belongs and hence its appropriate SF for its transmissions, it will compare its distance with the radius R_i of each corona C_i starting from the smallest corona. A node n considers that it is located on corona C_i if it satisfies the following condition: $R_{i-1} < dist_n \leq R_i$, where $dist_n$ is the distance of node n to the gateway. In other words, a node will use the assigned SF for corona C_i that has an R_i greater than its distance. In order for a node to calculate its distance to the gateway, we assume that each node knows its coordinates as well as the ones of the associated gateway. Furthermore, each node maintains a vector of R_i values that is used for comparison with its distance. It is worth pointing out that, the process of determining the corresponding corona for nodes, and hence the appropriate SF, is done only once for static nodes during their joining phase to the network, which is the case of most IoT scenarios. Furthermore, SF ranges or corona radii are not fixed values and it greatly depends on the environment. In other words, rural environments with possible Line of Sight (LoS) have longer SF ranges than urban environments. After a node has recognized its corona C_i and hence its appropriate SF, it will determine its Channel Frequency CF and timeslot SID as explained in the following section.

B. CF and SID Determinations

In order to fairly assign nodes to the available number of CFs K, we divide the network field into K sectors with angle β and assign a channel CF_j for each sector Sec_j. Fig.2 shows a demonstrative example of this partitioning. All nodes located in sector Sec_j will use channel CF_j that is assigned for that sector. In other words, based on node's angle θ_n to the sector's angle β, nodes find their sector IDs Sec_j as follows

$$Sec_j = \frac{\theta_n}{\beta} \tag{1}$$

where $\beta = 2\pi/K$, θ_n is the node's angle to the gateway, and K is the number of the available CFs.

By doing that, the network field is partitioned into sectors with identical areas. Hence, the traffic load is balanced among the channels. The intersection between corona C_i and sector Sec_j is called $Corona - Sector_{ij}$. All nodes belonging to the same $Corona - Sector_{ij}$ will use the same SF_i that is assigned to corona C_i and the same channel CF_j that is assigned to sector Sec_j. To regulate the channel access among them, we further divide each $Corona - Sector_{ij}$ into a grid of $q_{ij} = \sqrt{m_{ij}}$ rows and columns, where m_{ij} is the number of timeslots per time frame. We assume that each square, which is the intersection between a row and a column, is labeled with an identifier that starts from 1 till m_{ij}, which represents the time-slot ID SID. m_{ij} is calculated as in Eq.2

$$m_{ij} = \begin{cases} 100, & \text{if } m_{ij} < 100 \\ \frac{\beta}{2}(R_i^2 - R_{i-1}^2) \times d, & \text{if} m_{ij} \geq 100 \end{cases} \tag{2}$$

where R_i and R_{i-1} are the radii of corona C_i and C_{i-1} respectively and d is the network density. As shown in Eq.2, the number of needed timeslots per frame or the frame size m_{ij} mainly depends on the node density of a given $Corona - Sector_{ij}$. Hence, frame sizes are dynamic and associated to the network density of a given area with common transmission parameters. However, when the network density d of a given $Corona - Sector_{ij}$ is not large, short frame sizes m are resulted. If this is the case, we may get frame sizes that is shorter than the DC limit between consecutive transmissions. To overcome that, frame sizes m_{ij} has a lower limit, which is

100 slots per frame, to respect the duty cycle of nodes, which is 1%. In AAFS-LoRa protocol, we take advantage of the waiting time between the successive transmissions by dividing it into timeslots and assign them to nodes that belong to a specific $Corona-Sector_{ij}$. In order for a node to determine its timeslot ID SID, it needs firstly to determine in which square it is located. Hence each node calculates its row_n and col_n as in Eq.3

$$\begin{aligned} row_n &= \frac{q_{ij}(dist_n - R_{i-1})}{R_i - R_{i-1}} \\ col_n &= \theta_n \mod \frac{\beta}{q_{ij}} \end{aligned} \tag{3}$$

where q_{ij} is the number of rows and columns in $Corona - Sector_{ij}$, $dist_n$ is the distance between the node n and the associated gateway, R_i and R_{i-1} are the radii of corona C_i and C_{i-1} respectively . After that, time-slot SID can be retrieved as given in Eq.4

$$SID = (row_n - 1) \times q_{ij} + col_n. \tag{4}$$

Algorithm 1 shows the steps a node considers to autonomously determine its SF, CF, and SID. It is worth pointing out that in AAFS-LoRa protocol, even with large number of connected nodes, the frame size remains reasonable and it is not proportional to the total number of the connected nodes. This is mainly because of the efficient distribution of SFs and CFs as there will be relatively small number of connected nodes with the same SF and CF and hence need different timeslots. In other words, the frame sizes are not corresponding to the total number of connected nodes, as it is the case of most of the related work. Alternatively, the frame sizes in the proposed algorithm corresponds to the number of nodes that have common transmission parameters and hence need different timeslots to avoid collisions.

Algorithm 1 AAFS-LoRa MAC protocol

Input: node coordinates (X_n,Y_n), Gateway coordinates (X_G, Y_G), number of channels K, and node density d
Output: The Spreading Factor SF_i, the Channel CF_j, and the timeslot SID for a given node n
$CFs \leftarrow [CF_1, CF_2, .., CF_K]$
$R_i \leftarrow [R_1, R_2, .., R_i]$
$dist_n \leftarrow euclideanDistance(X_n, Y_n, X_G, Y_G)$
# Determine node's SF SF_i
for $i \leftarrow 1$ to 7 **do**
    if $dist_n \leq R[i]$ **then**
        $SF_i \leftarrow (i-1) + 7$
        $break$
    end if
end for
# Calculate sector's angle β
$\beta \leftarrow 2\pi/K$
Calculate node's theta to the gateway θ_n
$\theta_n \leftarrow \arctan(Y_n - Y_G / X_n - X_G)$
Find node's sector ID Sec_j
$Sec_j \leftarrow \theta_n / \beta$
$CF_j \leftarrow CFs[Sec_j]$
Calculate the frame size m_{ij}
$m_{ij} \leftarrow (\beta/2)(R_i^2 - R_{i-1}^2) \times d$
Apply lower limit to m_{ij} to respect node's DC
$m_{ij} \leftarrow (m_{ij} < 100)?100 : m_{ij}$
$q_{ij} \leftarrow \sqrt{m_{ij}}$
Calculate the row_n and col_n ▷ Eq.3
$SID \leftarrow (row_n - 1) \times q_{ij} + col_n$

III. Performance Evaluation

In this section, a simulation-based assessment for AAFS-LoRa MAC protocol is conducted using OMNET++ simulator [14] under FLoRa framework [15]. The simulation was run under a different number of nodes (1000 to 4000 nodes) that are randomly distributed around the gateway with a maximum distance corresponding to the field radius R. The radius of the field as well as the radii of SF coronas R_i are field-dependent, which means that they are varying depending on the deployed environment. It is worth stating that this simulation assumes a sub-urban environment that uses the well-known log-distance path loss model with shadowing [16]. Table I shows the values used in the simulation experiment for these parameters. The proposed protocol assumes one packet transmission per frame with a 20-byte packet length. In order to determine the maximum eligible distance for a given SF , R_i values, we conduct an experiment using OMNET++ simulator. Table II shows the maximum eligible distance for each SF using a transmission power equals 14 dBm and the sensitivity levels that is configured at the gateway as specified in [17].

TABLE I
Simulation Parameters

Parameter	Value	Comments
CF	{868.1, 868.3, 868.5, 867.1, 867.3, 867.5, 867.7, 867.9}	Carrier Frequencies (MHz)
SF	7 to 12	Spreading factors
TP	14 dBm	Transmission powers
CR	4/5	Coding rate
BW	125kHz	Bandwidth
R	8921m	Field radius
N	1000 - 4000	Number of nodes
Simulation time	11	Days

The following sections demonstrate the performance of AAFS-LoRa MAC protocol compared to the legacy LoRaWAN with Adaptive Data Rate (ADR) enabled.

A. The end-to-end delay

Fig.3 shows the end-to-end delay as a function of the number of connected nodes. As shown in the figure, the proposed protocol has much lower end-to-end delay compared to ADR-LoRaWAN. In fact, the end-to-end delay is greatly affected by the used SF. Smaller SFs have shorter ToA and hence lower delay. Accordingly and from delay point of view, it is better to

TABLE II
MAX ELIGIBLE DISTANCE FOR EACH SPREADING FACTOR SF [17]

SF	MAX eligible distance (m)	Sensitivity (S)
7	2450	-124 dBm
8	3306	-127 dBm
9	4450	-130 dBm
10	5998	-133 dBm
11	7316	-135 dBm
12	8921	-137 dBm

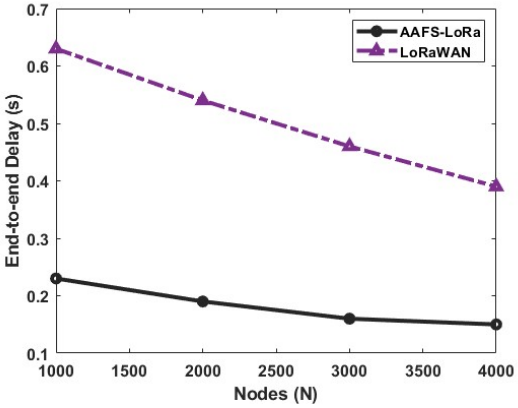

Fig. 3. The end-to-end delay

use lower SFs whenever they are eligible. Moreover, the end-to-end delay is only considered for the successfully received packets by the gateway. Hence, the end-to-end delay of both protocols is decreasing with the increased N as collisions are increased.

B. Network throughput

Fig.4 shows the network throughput as a function of the number of connected nodes. The network throughput is greatly affected by the end-to-end delay and the collision rate. As shown in the figure, the proposed protocol achieves higher throughput than that of the ADR-LoRaWAN thanks to the efficient distribution of the transmission parameters that allows different parallel transmissions to be successfully received at the gateway. In fact, the throughput of AAFS-LoRa MAC protocol has a convex behavior as collisions increase with the increase of N. Furthermore, since the AAFS-LoRa MAC protocol has relatively small frame sizes with large number of connected nodes, we get small waiting time and hence higher throughput. In fact, with $N = 4000$, the maximum frame size in AAFS-LoRa MAC protocol is only 132 timeslots.

C. AAFS-LoRa MAC protocol with different frame sizes

In this section, we investigate the performance of AAFS-LoRa MAC protocol under different frame sizes m ranging from 100 to 500 timeslots and under different number of connected nodes N ranging from 1000 to 4000 nodes. Please note that this experiment was run for 5 simulation days. The goal of the experiment is to find the optimal frame size m

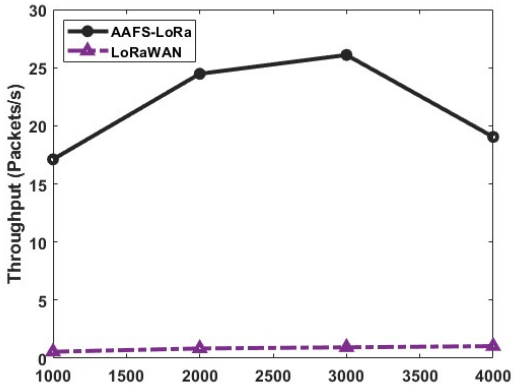

Fig. 4. The network throughput

for a given number of connected nodes N. Fig.5 shows the performance of the AAFS-LoRa MAC protocol with different frame sizes. As shown in the figure, the lowest collision rate was achieved under a frame size of $m = 500$ and a number of nodes of $N = 1000$. This is mainly because when the number of connected nodes is relatively small, collisions are mitigated. Furthermore, with large frame sizes, the number of available squares per $Corona - Sector_{ij}$ becomes larger and the area of each square becomes smaller. Hence, the probability of having two or more nodes located on the same square, and hence have the same SID is decreased. Accordingly, the lowest end-to-end delay was achieved when $N = 4000$ nodes and $m = 300$ timeslots since the end-to-end delay is only calculated for the successfully received packets at the gateway. Hence, we achieve lowest delay with this high number of connected nodes as it has the highest collision rate. On the other hand, the highest throughput was obtained with frame size $m = 100$ timeslots and $N = 3000$ nodes when the traffic rate equals almost 29 packets/s. Regarding energy efficiency, the lowest energy consumption was obtained with $m = 500$ and $N = 1000$. As expected, with relatively few number of connected nodes and large frame sizes, collisions will be mitigated and hence less energy will be consumed. To sum up, as shown in Fig.5, there is a different optimal m for each N and performance objective. For example, large m is recommended for applications that target to lower the collision rates and hence the energy consumption. However, small frame sizes are preferable when the main objective is to increase the network throughput.

IV. CONCLUSION

This paper proposes AAFS-LoRa protocol that allows nodes to autonomously configure their transmission parameters without the need for a costly synchronization process. The paper presents a method that allows the nodes to determine their optimal configurations in terms of SFs, channels, and timeslots based on the location information of the nodes and their gateway. Each node is configured with the minimum eligible SF that ensure the successful delivery of their packets. Furthermore,

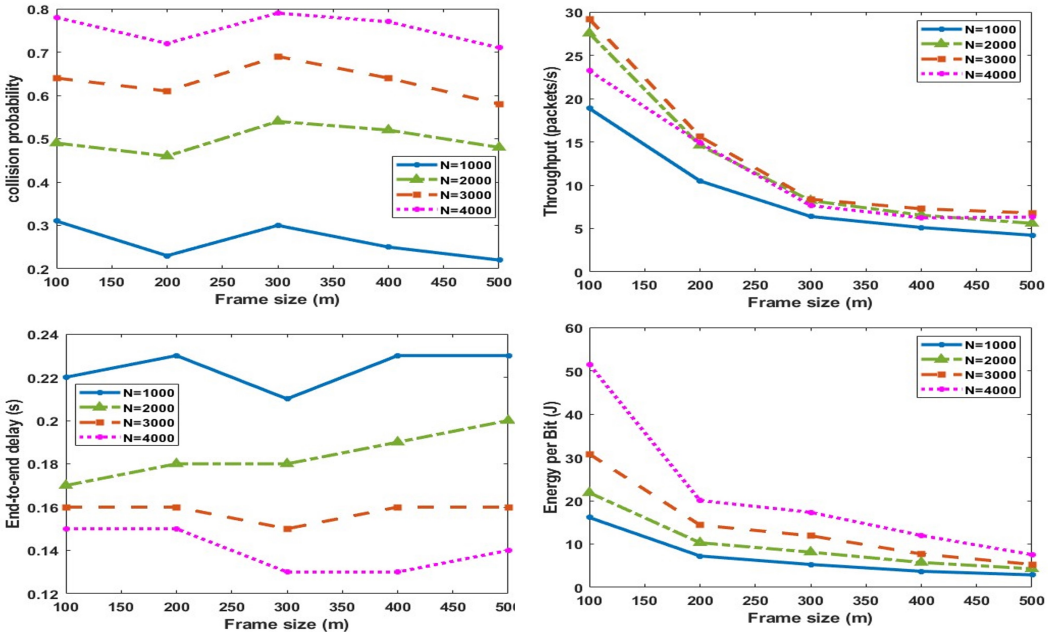

Fig. 5. AAFS-LoRa MAC protocol with different frame sizes (m)

AAFS-LoRa protocol ensures the load balancing among the channels to avoid channel congestion. A novel aspect of the proposed TDMA-based protocol is that the frame size is large enough to accommodate only the nodes that have the same channel and SF to mitigate collisions between them. Hence, small frame sizes are achieved even with a large number of connected nodes. Further investigation could be applied to mitigate collisions by ensuring a unique slot-ID for all nodes that use common SF and channel.

REFERENCES

[1] Z. Sun, H. Yang, K. Liu, Z. Yin, Z. Li, and W. Xu, "Recent Advances in LoRa: A Comprehensive Survey," *ACM Transactions on Sensor Networks*, Jun. 2022, just Accepted. [Online]. Available: https://doi.org/10.1145/3543856

[2] A. Furtado, J. Pacheco, and R. Oliveira, "PHY/MAC Uplink Performance of LoRa Class A Networks," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6528–6538, Jul. 2020, conference Name: IEEE Internet of Things Journal.

[3] "About LoRaWAN® | LoRa Alliance®." [Online]. Available: https://lora-alliance.org/about-lorawan

[4] T. Polonelli, D. Brunelli, A. Marzocchi, and L. Benini, "Slotted ALOHA on LoRaWAN-Design, Analysis, and Deployment," *Sensors*, vol. 19, no. 4, p. 838, Feb. 2019. [Online]. Available: http://www.mdpi.com/1424-8220/19/4/838

[5] D. Zorbas and X. Fafoutis, "Time-Slotted LoRa Networks: Design Considerations, Implementations, and Perspectives," *IEEE Internet of Things Magazine*, vol. 4, no. 1, pp. 84–89, Mar. 2021, conference Name: IEEE Internet of Things Magazine.

[6] L. Leonardi, F. Battaglia, and L. L. Bello, "RT-LoRa: A Medium Access Strategy to Support Real-Time Flows Over LoRa-Based Networks for Industrial IoT Applications," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 812–10 823, Dec. 2019, conference Name: IEEE Internet of Things Journal.

[7] A. Triantafyllou, P. Sarigiannidis, T. Lagkas, I. D. Moscholios, and A. Sarigiannidis, "Leveraging fairness in LoRaWAN: A novel scheduling scheme for collision avoidance," *Computer Networks*, vol. 186, p. 107735, Feb. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128620313232

[8] A. Triantafyllou, D. Zorbas, and P. Sarigiannidis, "Time-slotted LoRa MAC with variable payload support," *Computer Communications*, vol. 193, pp. 146–154, Sep. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366422002444

[9] J. Haxhibeqiri, I. Moerman, and J. Hoebeke, "Low Overhead Scheduling of LoRa Transmissions for Improved Scalability," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3097–3109, Apr. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8516298/

[10] 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, Jan. 2020, conference Name: IEEE Internet of Things Journal.

[11] D. Zorbas and B. O'Flynn, "Autonomous Collision-Free Scheduling for LoRa-Based Industrial Internet of Things," in *2019 IEEE 20th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*. Washington, DC, USA: IEEE, Jun. 2019, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/document/8792975/

[12] D. Zorbas, K. Abdelfadeel, P. Kotzanikolaou, and D. Pesch, "TS-LoRa: Time-slotted LoRaWAN for the Industrial Internet of Things," *Computer Communications*, vol. 153, pp. 1–10, Mar. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0140366419314677

[13] H. Alahmadi, F. Bouabdallah, and A. Al-Dubai, "A novel time-slotted LoRa MAC protocol for scalable IoT networks," *Future Generation Computer Systems*, vol. 134, pp. 287–302, Sep. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X22001261

[14] OMNeT++ Discrete Event Simulator, 2018. [Online]. Available: https://omnetpp.org/

[15] M. Slabicki, G. Premsankar, and M. Di Francesco, "Adaptive configuration of lora networks for dense IoT deployments," in *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*. Taipei, Taiwan: IEEE, Apr. 2018, pp. 1–9. [Online]. Available: https://ieeexplore.ieee.org/document/8406255/

[16] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, Mar. 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128601003024

[17] S. Corporation, "Sx1272/73 datasheet rev 3.1," 2017. [Online]. Available: shorturl.at/klBDI