# A Scheduling Algorithm for Improving Scalability of LoRaWAN

Junhee Lee, Wun-Cheol Jeong
University of Science and Technology,
Electronics and Telecommunications Research Institute
Daejeon, Korea
ljh8511@etri.re.kr, wjeong@etri.re.kr

Byeong-Cheol Choi
IoT Research Division
Electronics and Telecommunications Research Institute
Daejeon, Korea
bcchoi@etri.re.kr

***Abstract*— In this paper, we propose a scheduling algorithm to connect a massive number of IoT devices in LoRaWAN. The proposed algorithm supports time-synchronized transmissions to mitigate the problem of scalability due to random channel access used in LoRaWAN. Our proposed algorithm schedules spreading factors, frequency channels, and timeslots for wireless links connecting end devices and gateways. Also, the group acknowledgement is employed to improve channel efficiency by aggregating acknowledgements of uplink transmissions received simultaneously from multiple end devices. We evaluate the performance of the proposed algorithm and compare it with the performance of ALOHA used for uplink transmissions in LoRaWAN. Our numerical results show that the proposed algorithm improves the connectivity of end devices over ALOHA. Compared to ALOHA, the proposed algorithm provides more than 60% increase in the number of end devices connected to a gateway, whereas more than 90% data messages originating from all end devices are delivered successfully.**

***Keywords*— Low Power Wide Area; LPWAN; LoRaWAN; LoRa; Internet of Thing; IoT; scalability; scheduling;**

## I. Introduction

The number of internet of thing (IoT) application domains and deployments continues to increase due to growth of IoT market. Most of IoT applications require low-rate, long-range and delay-tolerant wireless communications at very low-energy usage. These requirements are difficult to achieve using existing Machine to Machine technologies such as wireless personal area networks (WPANs) and cellular networks. Low-power wide area (LPWA) networks represent a novel communication paradigm, which will complement existing technologies to meet requirements of IoT applications. LPWA technologies aim for supporting a massive number of low-cost, low-throughput devices, enabling battery-powered devices to be deployed for years. According to [1], it is estimated that the number of LPWA connections will grow over 1 billion by 2021.

LPWA technologies are characterized by exploiting the sub-GHz unlicensed, industrial, scientific, and medical (ISM) band, and licensed cellular band. Narrowband IoT (NB-IoT) standardized by the third generation partnership project (3GPP) supports requirements of IoT devices in the cellular band. This standard uses narrowband with the bandwidth of 180 kHz for uplink and downlink transmissions. The multiple access scheme used in the downlink is orthogonal frequency division multiple access (OFDMA) with 15 kHz sub-carrier spacing, while single carrier FDMA (SC-FDMA) is used for uplink transmissions. NB-IoT can be directly deployed in the global system for mobile communications (GSM) or long term evolution (LTE) band by three deployment types such as In-band, guard-band, and stand-alone. In-band deployment indicates that the narrowband is deployed inside a single LTE physical resource block (PRB). In guard-band deployment, the narrowband is deployed in unused PRBs between two adjacent LTE. Also, the narrowband can be deployed inside a single GSM carrier in stand-alone deployment [2].

Meanwhile, proprietary LPWA technologies used in unlicensed band are introduced. SIGFOX [3] offers LPWA connectivity solutions operated in sub-GHz ISM band. This technology uses ultra narrow band (UNB) to support long-range communications with low noise levels. The radio interface is optimized for uplink transmissions from battery-powered end devices. The end device wakes up only when uplink data message is generated, and sends the message over a narrow bandwidth of 100 Hz. Following the uplink transmission, the device opens a window of opportunity for receiving a downlink message. The uplink and downlink transmissions are unbalanced due to the unlicensed band regulation. The maximum number of uplink and downlink messages available per day is limited to 140 and 4, respectively. This implies that acknowledging every uplink transmission is not supported.

LoRaWAN [4] is a LPWA technology in sub-GHz ISM band using chirp spread spectrum (CSS) technique, which makes it robust to interference and allow long transmission range. In LoRaWAN, a network is organized in a star-of-stars topology, in which gateways forward messages between end devices and a central network server. Gateways are connected to the network server through IP connections, while end devices use a simple random access such as ALOHA over a single-hop wireless links connected to gateways. LoRaWAN supports multiple spreading factors and frequency channels for communications between gateways and end devices. These multiple communication resources enable simultaneous uplink transmissions by allocating different spreading factor and frequency channel, which in turn maximizes overall network capacity. However, an acknowledgement message should be

This work was supported by ETRI internal R&D project [18HB2740, Space-time complex artificial intelligence blue-green algae prediction technology based on direct-readable water quality complex sensor and hyperspectral image].

individually transmitted to all devices sending data messages, which results in increasing overhead. The overhead may interfere with the reception of other uplink transmissions using ALOHA that end devices transmit the uplink traffic without doing any carrier sensing. Thus, it is difficult to support a large number of devices deployed a wide geographical area. This scalability problem in LPWA technologies has been seriously issued to meet the requirements of IoT applications [5-11].

In this paper, we propose a transmission scheduling algorithm to connect a massive number of IoT devices in LoRaWAN. The proposed algorithm provides time-synchronized uplink transmissions to mitigate the problem of scalability due to random channel access. Also, we employ group acknowledgement (GACK) to improve channel efficiency by aggregating acknowledgements of uplink transmissions received simultaneously. In the algorithm, spreading factors, frequency channels, and timeslots are considered as communication resources to be allocated for link transmissions. First of all, the algorithm allocates spreading factors to reduce the duration of transmission. Obviously, there is a trade-off between the duration of transmission and the receiver sensitivity. Low spreading factor reduces the duration of transmission, but long-range communication is not available due to high receiver sensitivity. The proposed algorithm minimizes the duration of transmission while supporting receiver sensitivity enough to receive the message from devices. Scheduling of timeslots and frequency channels is performed based on allocations of spreading factor. If communication links are allocated to the same spreading factor, the algorithm schedules link transmissions so that they send the uplink message simultaneously. Otherwise, simultaneous transmissions are not allowed by allocating different timeslots. By the scheduling strategy, a GACK message includes only acknowledgements for the uplink traffic transmitted with same spreading factor. This implies that the GACK message can be sent to only devices, which require the same spreading factor for downlink transmission. Thus, the proposed algorithm reduces the consumption of timeslots for the GACK transmission by avoiding the use of larger spreading factor than necessary for downlink transmission.

The rest of this paper is organized as follows. In section Ⅱ, we briefly introduce LoRaWAN, and some basic features of LoRaWAN are described. In section Ⅲ, a transmission schedule model in LoRaWAN is introduced to connect a massive number of IoT devices. We evaluate the network performance of the proposed algorithm and compare it with the performance of ALOHA in section Ⅳ. Finally, we conclude our study in section Ⅴ.

## II. Overview of LoRaWAN

LoRaWAN offers large coverage areas and long battery life operation for end devices. LoRa is the physical layer using CSS, which spreads a narrow band signal over a wider channel bandwidth. LoRa supports multiple spreading factors ranging from 7 to 12. The spreading factor is defined as $\log_2 W/R_b$ in [12], where $W$ is the bandwidth, and $R_b$ is the data rate. As the spreading factor increases, larger communication ranges can be supported due to decreased data rate. The different spreading factors are orthogonal. This means that multiple messages can be exchanged at the same time by using different spreading factors. Also, LoRa supports multiple frequency channels. Frequency hopping is exploited at each uplink transmission to mitigate wireless interferences. In US 902-928 MHz band, 64 channels with 125 kHz bandwidth and 8 channels with 500 kHz bandwidth are used for uplink transmissions. Spreading factors from 7 to 10 are allowed for uplink transmissions in channels with 125 kHz bandwidth, while all spreading factors are used in channels with 500 kHz bandwidth. Following the uplink channels, 8 channels with 500 kHz are used for downlink transmissions. Figure 1 shows the frequency channels in US 902-928MHz ISM band.

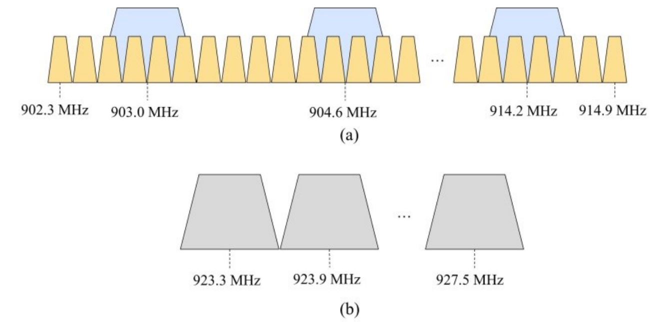


Figure 1. Illustrations of frequency channels in US 902-928MHz band. (a) Uplink channels. (b) Downlink channels.

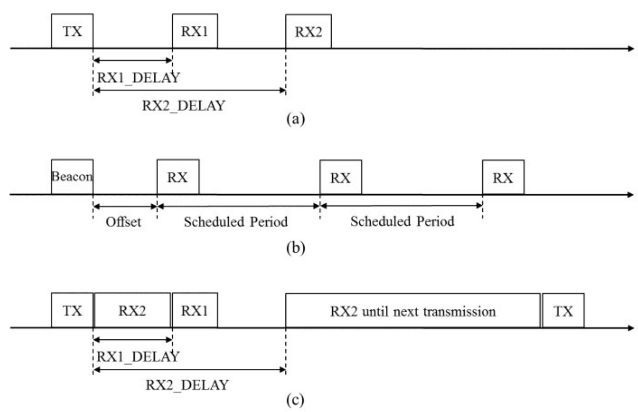


Figure 2. Timing structure of the uplink and downlink transmissions between the gateway and end devices. (a) class A. (b) class B. (c) class C.

LoRaWAN defines three different types of end devices with different capabilities and power requirements. Devices of class A consume least power among all types of devices in LoRaWAN. The device of class A turns off its transceiver until a data message to transmit is generated. If the uplink message is generated, the device sends the message by turning on the transceiver. After sending the message, the device waits for a short period to receive a downlink message from the gateway. Downlink transmission is only allowed at predetermined receive windows after a successful uplink transmission. On the other hand, end devices of class B open receive windows at scheduled timeslots for enabling a network server initiated downlink messages. In class B, devices are synchronized by periodic beacon sent from the gateway to schedule downlink transmissions without prior successful uplink transmissions. Finally, devices of class C are always listening to the channel with continuously open reception windows. This type offers flexibility on downlink transmissions, but power consumption arises. Figure 2 shows the timing structure of the uplink and downlink transmissions for devices of class A, B, and C.

## III. SCHEDULING ALGORITHM FOR LORAWAN

### A. System Model

In this section, we present the system model to schedule transmissions in LoRaWAN with single gateway environment. LoRa supports multiple spreading factors to flexibly adapt the data rate according to distance between a gateway and end devices. The proposed algorithm allocates spreading factor to minimize the duration of transmission, while providing receiver sensitivity to successfully receive the message. Let us denote that $P_i$ is the received signal power from link $i$, and $R(k)$ is the receiver sensitivity when spreading factor is $k$. Then, the spreading factor of transmission on link $i$, denoted by $s_i$, can be described as follows:

$$\boldsymbol{S(i)} = \{k | P_i \geq R(k) + \gamma\} \tag{1}$$

$$s_i = min(\boldsymbol{S(i)}), \tag{2}$$

where $\gamma$ is the positive threshold value. The proposed algorithm searches the set of spreading factors $\boldsymbol{S(i)}$, which support successful transmission on link $i$. Then, the minimum spreading factor in $\boldsymbol{S(i)}$ is selected to minimize the duration of transmission on link $i$. Table I shows the sensitivity of LoRa transceiver according to spreading factor when the bandwidth of channel is 125 kHz. If the power of signal received from link $i$ is -125 dBm, spreading factor of 9 is allocated for the transmissions on link $i$ when $\gamma$ is 3 dB.

Table I. Sensitivity of LoRa transceiver [13]

| $k$ | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|
| $R(k)$[-dBm] | -123 | -126 | -129 | -132 | -133 | -136 |

Timeslots and frequency channels are scheduled based on allocations of spreading factor. In the proposed algorithm, simultaneous transmissions are not allowed if links are allocated to different spreading factor. This scheduling policy reduces the duration of GACK transmission. Figure 3 shows the difference in the duration of GACK transmission depending on how uplink transmissions are scheduled. A star network topology consisting of three end devices (A, B, and C) and one gateway (G) is considered as shown in figure 3(a). We assume that device B and C are assigned to the same spreading factor, but the different spreading factor is allocated for the transmission of device A. Figure 3(b) describes link transmissions scheduled regardless of allocations of spreading factor. In the example, timeslots and frequency channels are scheduled so that device A and C send data messages simultaneously. Since the GACK message should be transmitted to not only device A but also device C located far from the gateway, a spreading factor high enough to deliver the message to device C is required. This implies that more timeslots are consumed than the number of timeslots required to deliver the downlink transmission for device A. However, the duration of GACK transmission can be reduced if simultaneous transmissions on links assigned different spreading factors are not allowed, as shown in figure 3(c).

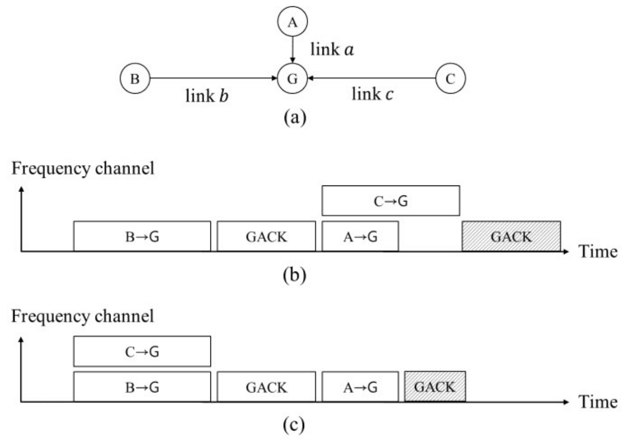


Figure 3. An example of uplink transmissions and GACK transmissions. (a) Network topology. (b) Transmission schedules regardless of spreading factor allocations. (c) Transmission schedules based on allocations of spreading factor.

From the scheduling strategy of the proposed algorithm, we realize that the assignment of timeslots and frequency channels depends on whether two links are assigned to the same spreading factor. Let us denote that $\pi_i^u$, $f_i^u$, and $\Delta_i^u$ are timeslot index, frequency channel index, and duration of uplink transmission for link $i$ in timeslots. We also define $\pi_i^g$, $f_i^g$, and $\Delta_i^g$, which are notations for timeslot index, frequency channel, and duration of GACK transmission for link $i$ in timeslots. Then, the conditions for scheduling of timeslots and frequency channels are described as follows:

$$\Delta_i^u + \Delta_i^g \leq \pi_j^u - \pi_i^u + o_{ij}N \leq N - (\Delta_j^u + \Delta_j^g),$$
$$s_i \neq s_j, \quad \forall i \in \boldsymbol{I}, \forall j \in \boldsymbol{N(i)} \tag{3}$$

$$f_i^u \neq f_j^u, \quad s_i = s_j, \ \forall i \in \boldsymbol{I}, \forall j \in \boldsymbol{N(i)} \tag{4}$$

$$\pi_i^u + \Delta_i^u \leq \pi_i^g, \quad \forall i \in \boldsymbol{I} \tag{5}$$

$$\Delta_i^u = \Delta_j^u, \ \forall i \in \boldsymbol{I}, \forall j \in \boldsymbol{N(i)}, \tag{6}$$

where $N$ is the number of timeslots in the beacon interval, $\boldsymbol{I}$ is the set of all links in the network, and $\boldsymbol{N(i)}$ is the set of all links except for link $i$. Also, $o_{ij}$ is 0 if $\pi_j^u > \pi_i^u$, and 1 if $\pi_j^u < \pi_i^u$. The first condition (3) indicates that simultaneous transmissions are not allowed if two links are allocated to the different spreading factor. The second condition (4) implies that simultaneous transmissions are available by allocating different frequency channel if two links are allocated to the same spreading factor. The last condition (5) means that the uplink data message cannot send during the GACK message is transmitted.

## B. Proposed Scheduling Algorithm

In this section, we propose a scheduling algorithm to connect a massive number of devices in LoRaWAN. In the proposed algorithm, a network server schedules spreading factors, frequency channels, and timeslots for all links in the network. The scheduling algorithm runs when a message indicating a request of scheduling is received from the device. First of all, the spreading factor is allocated to the link considering the received signal power from the device as described in (1)-(2). After allocating spreading factor for the link, the network server schedules frequency channels and timeslots. If other links have already been scheduled prior to receive the request message, the network server allocates timeslots and frequency channels so that multiple uplink messages on links assigned to the same spreading factor can be delivered simultaneously. These procedures minimize the length of scheduling, which in turn maximizes the number of devices to be scheduled. We describe scheduling of timeslots and frequency channels in the proposed algorithm as follows:

$$\min_{\pi_i^u, \pi_i^g, f_i^u} \varphi \tag{7}$$

subject to (3)-(6),

where $\varphi$ is the length of scheduling for all links connecting a gateway and end devices requesting transmission schedules.

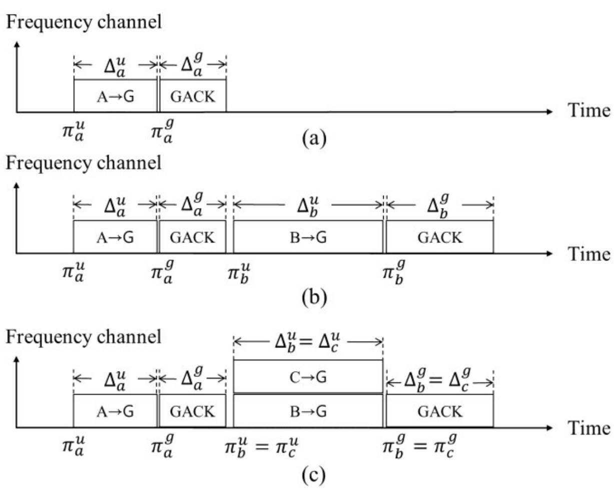


Figure 4. An example of uplink and GACK transmissions for three end devices. (a) Transmission schedules for device A. (b) Transmission schedules for device A and B after a request of scheduling from device B. (c) Transmission schedules for all devices.

Figure 4 shows the example for procedures of scheduling in a network configuration described in figure 3(a). We consider allocations of spreading factor described in the previous example with figure 3 ($s_a \neq s_b = s_c$). We assume that device A requests a scheduling first among all devices, and device C is scheduled lastly. First of all, the network server schedules the uplink transmission and the GACK transmission for the link connecting the gateway and device A. Since the scheduling request from device A is the first time, the uplink transmission is allocated to $\pi_a^u$ without any constraints as shown in figure 4(a). After allocating the uplink transmission, the GACK transmission is allocated to $\pi_a^g$ satisfying constraint (5). Figure 4(b) shows the result of link scheduling after a request of scheduling from device B. Since the spreading factor allocated for device B differs from the spreading factor of device A, different timeslots are assigned for device B by constraint (3). Finally, the uplink transmission of device C is assigned to the same timeslots with device B by allocating different frequency channel as shown in figure 4(c). This is because that both device B and C are allocated to the same spreading factor. The GACK transmission for device B and C is scheduled so that the downlink message can be sent after receiving uplink messages from two devices.

# IV. Peformance Evaluation

In this section, we evaluate the performance of the proposed algorithm and compare it with that of ALOHA in LoRaWAN. We employ packet success ratio as a performance metric by varying the number of devices deployed in the network to investigate how many devices can be connected to a gateway. The performance is evaluated in a star network topology, which consists of a gateway and multiple end devices connected to the gateway. Devices are randomly deployed a geographical area within 4 km from the gateway. We consider devices of class B to enable beacon synchronized uplink and GACK transmissions in the proposed algorithm. In the configuration, each device generates a data message periodically, and sends the uplink data message to the gateway. We assume that the type of uplink data is confirmed message, which require receiving an acknowledgement message from the gateway. The spreading factor, frequency channel, and bandwidth for US 902-928 MHz ISM band described in [4] are considered in the simulation. Table II shows the lists of simulation parameters.

Table II. Simulation parameters

| Parameters | Value |
|---|---|
| Number of devices | 1000 – 10000 |
| Terrain dimensions | 4 km from a gateway |
| Device placement | Random |
| Tx Power | 20 dBm |
| Bandwidth | 125 kHz |
| Antenna gain | 0 dBm |
| Antenna height | 1 m |
| Pathloss model | Two-Ray |
| Beacon interval | 128 sec |
| Coding rate | 4/5 |

Figure 5 illustrates the packet success ratio of both ALOHA and proposed algorithm by varying the number of end devices when each device generates a data message every 10 minutes, 30 minutes, and 1 hour. We assume that 8 frequency channels are available for link transmissions. It is observed that the packet success ratio of both algorithms decreases as the number of devices increases. However, the performance of the proposed algorithm is higher than that of ALOHA regardless of the number of devices deployed in the network. The difference in packet success ratio results between two algorithms is noticeable for a large number of devices, where the performance of ALOHA approaches to zero. This is mainly because the increase in the number of collisions of uplink messages as the number of devices increases. Meanwhile, the simulation results show that the proposed algorithm supports 8,000 devices to be connected to a gateway if target packet success ratio is 90%, while 5,000 devices can be connected when each device transmits a data message every 1 hour using ALOHA. The performance of ALOHA decreases as the transmission interval of the data message becomes shorter. If a data message is generated every 30 minute, only 2,000 devices can be connected to a gateway. Moreover, ALOHA makes it difficult to connect 1,000 devices when the device generates a data message every 10 minutes. On the other hand, the transmission interval does not affect the performance of the proposed algorithm. Since each device has opportunity to send a data message in scheduled timeslots every beacon interval, the data message can be delivered successfully within the beacon interval if the transmission interval of the message is longer than beacon interval.

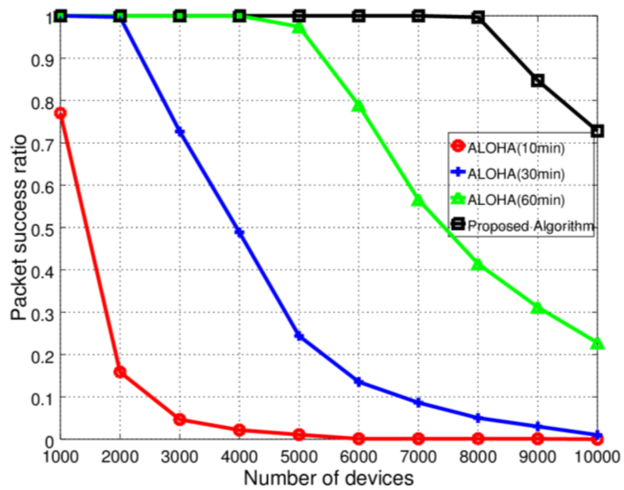


Figure 5. Packet success ratio according to the transmission interval of the data message.

Figure 6 shows the performance behavior of packet success ratio according to the number of available frequency channels. We consider 2, 4, and 8 frequency channels used for scheduling of links connecting end devices and the gateway. We assume that each device generates a data message every 1 hour. We observe that the number of devices to be connected to a gateway increases as the number of available frequency channels increases. However, the performance of the proposed algorithm is lower than that of ALOHA when a small number of frequency channels are available for link transmissions. If 2 frequency channels are used for scheduling of links, the proposed algorithm supports 2,000 devices to be connected to a gateway when target packet success ratio is 90%, while 5,000 devices can be connected in ALOHA. In the proposed algorithm, the number of simultaneous transmissions cannot exceed the number of frequency channels. This limitation causes low performance of the proposed algorithm in a small number of frequency channels. However, the performance of the proposed algorithm shows higher than that of ALOHA when the number of devices increases to 8,000 or more. Since devices do not listen to the channel before sending data messages, collisions of the messages incur frequently, which results in degrading the performance of ALOHA. Meanwhile, the proposed algorithm provides higher performance when 8 frequency channels are used for transmissions. As shown in the figure 6, the proposed algorithm supports link transmissions between a gateway and 8,000 devices, whereas more than 90% data messages are delivered successfully. However, only 40% of data messages originating from 8,000 devices are delivered successfully in ALOHA. This is because that the number of communication resources to be allocated for link transmissions increases by employing a large number of frequency channels.

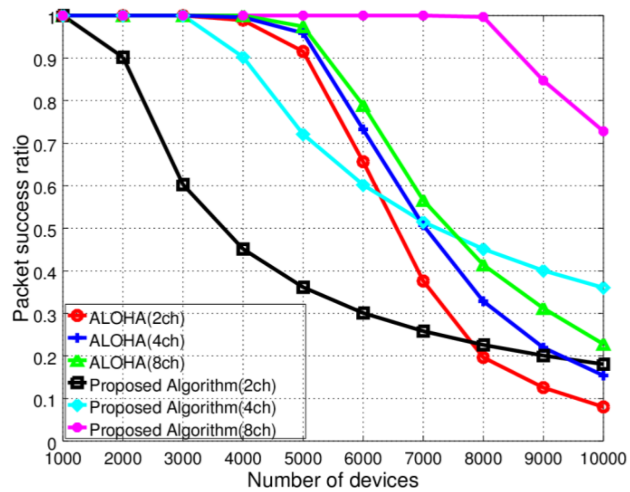


Figure 6. Packet success ratio according to the channel bandwidth

Figure 7 illustrates the packet success ratio of two algorithms by varying MAC payload size. We consider 15, 25, and 35 bytes as the payload size of uplink message. In the simulation, 8 frequency channels are employed and messages are generated from each device every 1 hour. It is observed that the performance of two algorithms decreases as the payload size increases. This stems from the fact that larger payload size of the message increases the duration of message. Especially, it is more likely that the medium has already been occupied by other transmissions with long duration when the device attempts to access the medium via ALOHA. Meanwhile, the performance of the proposed algorithm is related with the number of timeslots to be allocated for link transmissions. As the duration of message increases, more timeslots are required to schedule the transmission. However, the number of timeslots that can be allocated for link transmissions is limited. Thus, the performance of the proposed algorithm decreases as the payload size increases. The simulation results show that approximately 4,000 devices can be connected to a gateway if the target packet success ratio is 90%, when each device transmits a data message with payload size of 35 bytes using ALOHA. However, our proposed algorithm supports 50% increase in the number of devices connected to the gateway compared to ALOHA.

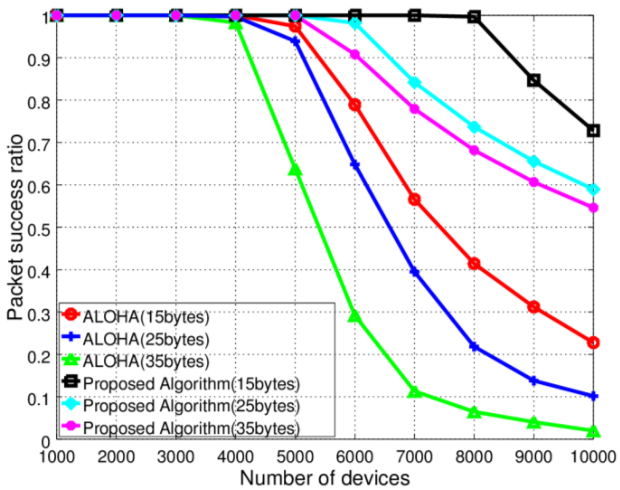


Figure 7. Packet success ratio according to the payload size

## V. CONCLUSION

In this paper, we propose a scheduling algorithm to connect a massive number of IoT devices in LoRaWAN. The proposed algorithm schedules spreading factor, frequency channel, and timeslot to minimize the length of scheduling for all wireless links connected to a gateway. Also, we employ GACK including multiple acknowledgements of uplink transmissions to improve channel efficiency. We evaluate the performance of the proposed algorithm and compare it with ALOHA used in uplink transmissions for LoRaWAN. The performance is evaluated by varying uplink traffic, channel bandwidth, and payload size. Simulation results show that the proposed algorithm provides more than 60% increase in the number of end devices connected to a gateway than ALOHA. The proposed algorithm supports maximally 8,000 devices connected to a gateway, while more than 90% of data messages originating from all devices are delivered successfully.

## REFERENCES


[1] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021," Technical Report, 2017. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.pdf

[2] A.Ratilainen, "NB-IoT Characteristics," IETF, draft-ratilainen-lpwan-nb-iot-00, 2016.

[3] Sigfox. [Online]. Available: http://www.sigfox.com/

[4] N.Sornin, M.Luis, T.Eirich, and T.Kramp, " LoRa Specification 1.0," LoRa Alliance, 2015.

[5] U.Raza, P.Kulkarni, and M.Sooriyabandara, "Low Power Wide Area Networks: An Overview," IEEE Communications Surveys & Tutorials, vol.19, pp.855-873, 2017.

[6] 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, pp.34-40, 2017.

[7] B.Reynders, Q.Wang, P.Tuset-Peiro, X.Vilajosana, and S.Pollin, "Improving Reliability and Scalability of LoRaWANs Through Lightweight Scheduling," IEEE Internet of Things Journal, vol.5, pp.1830-1842, 2018.

[8] F.V.Abeele, J.Haxhibeqiri, I.Moerman, and J.Hoebeke, "Scalability Analysis of Large-Scale LoRaWAN Networks in ns-3," IEEE Internet of Things Journal, vol.4, pp.2186-2198, 2017.

[9] S.Barrachina-Munoz, B.Bellalta, T.Adame, and A.Bel, "Multi-hop Communication in the Uplink for LPWANs," Computer Networks, vol.123, pp.153-168, 2017.

[10] J.Haxhibeqiri, F.V.Abeele, I.Moerman, and J.Hoebeke, "LoRa Scalability: A Simulation Model Based on Interference Measurements," Sensors, vol.17, pp.1193, 2017.

[11] O.Georgiou and U.Raza, "Low Power Wide Area Network Analysis: Can LoRa Scale?," IEEE Wireless Communication Letters, vol.6, pp.162-165, 2017.

[12] Semtech, "SX1272/3/6/7/8 LoRa Modem Design Guide," [Online]. Available: https://www.semtech.com/uploads/documents/LoraDesign Guide_STD.pdf.

[13] Semtech, "SX1276/77/78/79 – 137 MHz to 1020 MHz Low Power Long Range Transceiver," [Online]. Available: https://www.semtech.com/ uploads/documents/DS_SX1276-7-8-9_W_APP_V5.pdf.