Energy-Efficient Multi-hop LoRaWAN with Transmission and Synchronization Control

Hiroto Shida, Aoto Kaburaki, and Koichi Adachi
Advanced Wireless & Communication Research Center (AWCC)
The University of Electro-Communications
1-5-1 Chofugaoka Chofu, Tokyo, Japan 182-8585, Email: {shida, kaburaki, adachi}@awcc.uec.ac.jp

***Abstract*—In LoRaWAN, each node transmits packets autonomously, so packet collisions occur when multiple nodes transmit packets at the same time and frequency. However, the clocks of inexpensive LPWAN nodes are generally not highly accurate, and synchronization errors can occur between devices over time. In addition, in single-hop LoRaWAN, it is not possible to achieve high data rates and a wide communication range at the same time. By using multi-hop communication, it is possible to achieve a wide communication range while maintaining a high data rate. However, LoRaWAN multi-hop communication suffers from the hidden node problem, throughput degradation due to the inability to transmit and receive packets simultaneously, and power consumption due to the need to constantly open the receive window. This paper proposes an autonomous distributed adaptive resource allocation method to solve the above problems. Specifically, we show that by assigning packet transmission slot decisions based on LoRaWAN packet information, the transmitting and receiving sides can share transmission and reception locations, avoid packet collisions, and reduce power consumption by avoiding unnecessary opening of reception windows.**

***Index Terms*—Multi-hop communication, Resource allocation, LPWAN, LoRaWAN**

I. Introduction

In recent years, research and development of the Internet-of-things (IoT), in which various devices are connected to the Internet for information exchange, has been underway [1]. Some IoT nodes constituting a wireless sensor network (WSN) are required to be battery-powered for their flexible deployment. In addition, IoT nodes are deployed over a wide area in smart cities and smart agriculture. Therefore, there is an increasing demand for communication standards that enable low-power consumption and long-distance transmission. A low-power wide area network (LPWAN) is a generic communication technology that meets the aforementioned requirements [2]. Although the transmission data rate of LPWAN is low, it can provide a wide communication area with low-power consumption. Since LPWAN uses unlicensed bands, the duty cycle (DC) is specified on each frequency channel for a gateway (GW) and end nodes (ENs) to realize frequency sharing with other systems [3]. DC prevents each EN from using a specific frequency channel frequently. The DC, defined by the current standard, is specified for each EN and limits so that time spent in all frequency channels per unit time is below the specified value [4]. In the future, as communication systems become more diverse, it is expected to be relaxed so that the occupied time of each frequency channel per unit time is less than the specified value [5].

A type of LPWAN called long range wide area network (LoRaWAN) is widely used due to its low deployment cost and use of unlicensed bands. Long range (LoRa) modulation based on chirp spread spectrum (CSS) modulation is used in the physical layer of LoRaWAN to achieve long-range and low power consumption. In LoRa modulation, the number of bits transmitted per symbol is defined as the spreading factor (SF), which can be set in the range of 7 to 12. Increasing the SF improves noise immunity and enables long-range communication, but there is a trade-off in that it reduces the data rate. In other words, a high data rate and long-range communication cannot be achieved simultaneously in single-hop node-GW communication.

The medium access control (MAC) layer in LoRaWAN uses a simple ALOHA method to reduce power consumption[1]. However, each EN selects one of the pre-allocated frequency channels and transmits packets in an autonomous distributed manner. Therefore, the ALOHA method is unable to detect simultaneous transmissions on the same frequency, leading to packet collisions. In contrast, the carrier sense multiple access/collision avoidance (CSMA/CA) method, though more complex, can prevent packet collisions and enhance transmission properties by performing carrier sense (CS) [7].

Other collision avoidance methods include resource allocation methods based on time synchronization such as time division multiple access (TDMA). However, the clocks of low-cost ENs in LoRaWANs are not highly accurate, and synchronization deviations occur among devices over time [8]. This makes it difficult for LoRaWAN to adopt resource allocation methods that require strict time synchronization. Additionally, the overhead for the network, such as control signals, increases for synchronization.

In LoRaWAN communication, multi-hop communication is a method to achieve simultaneous high data rate and long-range communication [9]. By inserting a repeater between a node and a GW, multi-hop communication can maintain a high data rate while achieving a wide communication area. However, LoRaWAN multi-hop communication has several problems. First, multi-hop communication is prone to the

[1]In certain countries and regions, it is compulsory to conduct carrier sense before transmission

979-8-3503-7099-7/24/$31.00 ©2024 IEEE

hidden node problem, in which packet collisions occur due to simultaneous transmissions from nodes outside the CS range. Specifically, in the case of multi-hop communication from the transmitter to relay 1, relay 2, and GW in the order, the transmission of relay 2 cannot be detected by the transmitter, and packets transmitted by the transmitter and relay 2 may collide. In addition to packet collisions, LoRaWAN nodes cannot transmit and receive packets simultaneously due to half-duplex communication, resulting in a decrease in overall system throughput compared to single-hop communication. To improve throughput, it is necessary to relay packets frequently, but this requires a lot of power for the relay. In addition, the receiver side does not know when a packet will arrive, so the receive window must always be open, which causes a battery consumption problem due to relaying. Furthermore, signals for synchronization can only be sent to one hop ahead in the communication range at a time, and synchronization with two or more hops ahead requires additional time overhead.

This paper proposes an autonomous decentralized adaptive resource allocation method to solve the above problems of conventional multi-hop LoRaWAN. The proposed method dynamically allocates resources for packets to be transmitted to achieve high interference tolerance and to apply to a larger number of systems. Specifically, packet transmission slots are allocated using the index of the hop order and the packet counter of the transmitting node in the header of the LoRaWAN packet [10]. This allows the receiver and transmitter sides to share the time and frequency of packets to be transmitted, avoiding packet collisions and reducing power consumption without opening unnecessary receive windows.

The remainder of this paper is organized as follows. Section II describes the system model. Section III explains the proposed method. Section IV provides the simulation results. Section V concludes this paper.

II. SYSTEM MODEL

A. Transmission and Reception Processes

This paper assumes a multi-hop communication system consisting of M devices ($\mathcal{M} = \{0, \ldots, m, \ldots, M-1\}$) where data packets are transmitted from a transmitter ($m = 0$) to a GW ($m = M-1$) through $M-2$ relays ($m \in \{1, 2, \ldots M-2\}$). Each device is assigned an index $m \in \mathcal{M}$ that determines the order of hops, and these indices are assumed to be known among the devices. The transmitter sequentially transmits $N_{\rm pckt}$ packets of a fixed time length $T^{\rm pckt}$ [sec]. The ith packet ($i \in \mathcal{I} = \{0, \ldots, N_{\rm pckt}-1\}$) contains a packet counter $D_i^{\rm pcnt} = i$. The relays and GW ("receiving devices $m_{\rm r} \in \mathcal{M}\backslash\{0\}$") will be in the standby state with the receive windows of all frequency channels ("all receive windows") opened until reception of the first packet from the transmitter and their previous relay ("transmitting devices $m_{\rm t} \in \mathcal{M}\backslash\{M-1\}$"). The relays are able to transmit a packet after receiving the packet from its previous transmitting device with a smaller index than its own. As shown in Fig. 1, the system divides the continuous time into multiple *frames* of frame time length $T^{\rm frame}$ [sec], which is further split into

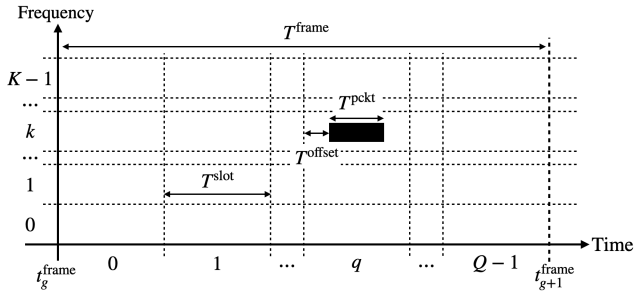

Fig. 1. Composition of packet transmission/reception frames

Q time slots of $T^{\rm slot}$ [sec]. The start time of the gth frame is denoted by $t_g^{\rm frame}$. The system uses K frequency channels. Each transmitter selects one of the combinations of time slot and frequency channel, i.e., *radio resource*. Thus, the number of available resources within a frame is $Q \times K$.

LoRaWAN nodes generally operates in a half-duplex mode, which means that transmission and reception cannot be performed simultaneously. Therefore, for example, the transmission timing of a transmitter ($m = 0$) and a relay ($m = 1$) must be different. In this paper, the nodes with even index m for each node are assumed to transmit on even-numbered frames, and the nodes with odd indexes are assumed to transmit on odd-numbered frames, alternately. In a transmission frame, each node transmits a packet in one slot within the frame. The packets are transmitted with an offset time $T^{\rm offset}$ [sec] from the starting time in the slot. Conversely, the non-transmitting frame is the receiving frame, and the packet is received.

B. Clock Drift

Since the clocks of inexpensive LPWAN devices are generally not highly accurate, time deviations occur between devices when there is no external input such as global positioning system (GPS). Such time discrepancy accumulates over time. This paper defines the clock drift as the relative time deviation of device $m \in \mathcal{M}\backslash\{0\}$ from device $m = 0$.

The clock drift accumulated during a certain period. For example, during the period $T^{\rm frame}$[sec] at device m, $T_m^{\rm frame}$ [sec], is given by

$$T_m^{\rm frame} = T^{\rm frame} + \int_0^{T^{\rm frame}} \Delta T_m^{\rm d}(t) dt, \tag{1}$$

where $\Delta T_m^{\rm d}$ represents the normalized clock drift of device m normalized by unit time and $\Delta T_m^{\rm d} \sim \mathcal{N}(\mu_m, \sigma_m^2)$ with $\mathcal{N}(\mu_m, \sigma_m^2)$ representing a Gaussian distribution with mean μ_m and variance σ_m^2 [11]. Similarly, $T^{\rm slot}$,$T^{\rm offset}$,and $T^{\rm pckt}$ are set to $T_m^{\rm slot}$,$T_m^{\rm offset}$, and $T_m^{\rm pckt}$, taking into account the clock drift at device m. When $m = 0$, $\Delta T_m^{\rm d} = 0$ due to the clock drift as the relative time deviation of device $m \in \mathcal{M}\backslash\{0\}$ from device $m = 0$.

III. PROPOSED SCHEME

A. Overview

This subsection describes the flow of the proposed method. The transmitter sets the start time as the start time of the

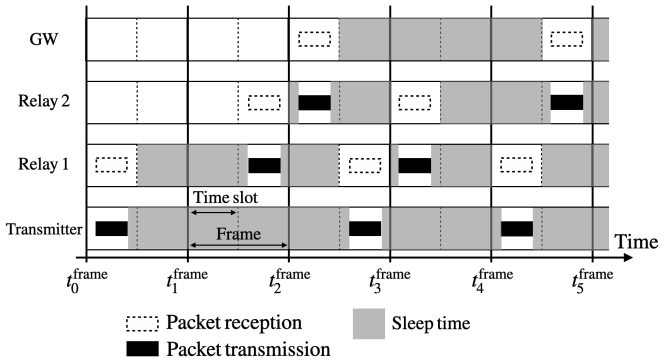

Fig. 2. Proposed scheme when each device is synchronized ($K = 1, Q = 2$).

0th frame, t_0^{frame} [sec], and transmits subsequent packets according to the packet transmission procedure described in Sect. III-B at an even number of frames.

Once a relay receives a data packet, it calculates the frame time to relay packet and to receive subsequent packets according to the procedure in Sect. III-C. By doing this, the time for transmission and reception has been calculated, the sleep time is set according to the procedure in Sect. III-D. Packet transmission is performed along the same way as the transmitter along the procedure in Sect. III-B. Figure 2 shows an example of the proposed scheme when each device is synchronized and number of time slots is $Q = 2$. In this case, transmitter $m = 0$ and relay $m = 2$ transmit in the same even frames, but in different time slots by following the procedure in Section III-B. Thus, relay $m = 1$ can receive packets from transmitter $m = 0$ without the interference from relay $m = 2$.

After receiving a packet, the GW calculates the frame time to receive subsequent packets according to the packet reception procedure described in Section III-C.

As mentioned above, devices are subject to time deviations due to clock drift. Therefore, this paper proposes a synchronization algorithm at each device based on the information in the header of the LoRaWAN packet in a distributed manner. When a device receives the first packet, it carries out initial synchronization as described in Section III-C1, and performs sequential synchronization as in Section III-C2. This compensates for time deviations due to clock drift and enables a transmission/reception method that avoids packet collisions.

B. Packet transmission procedure

Transmitting device $m_{\text{t}} \in \mathcal{M}\backslash\{M-1\}$ determines time slot q and frequency channel k in which it transmits the D_i^{pcnt}th packet as follows.

$$q(m_{\text{t}}, D_i^{\text{pcnt}}) = \text{mod}(f_{\text{q}}(m_{\text{t}}, D_i^{\text{pcnt}}), Q), \quad (2)$$

$$k(m_{\text{t}}, D_i^{\text{pcnt}}) = \text{mod}(f_{\text{k}}(m_{\text{t}}, D_i^{\text{pcnt}}), K), \quad (3)$$

where $\text{mod}(\cdot,\cdot)$ is the remainder operation, $f_{\text{q}}(m_{\text{t}}, D_i^{\text{pcnt}})$ and $f_{\text{k}}(m_{\text{t}}, D_i^{\text{pcnt}})$ is an arbitrary function uniquely determined by index m_{t} of the transmitting device and the packet counter D_i^{pcnt}. D_i^{pcnt} can be obtained by receiving packet, m_{t} and Eqs. (2) and (3) are known by all the devices, and the transmission resources can be shared among all devices. This paper designs functions $f_{\text{q}}(m_{\text{t}}, D_i^{\text{pcnt}})$ and $f_{\text{k}}(m_{\text{t}}, D_i^{\text{pcnt}})$ for collision avoidance only. The packet transmission start time, $t_{m_{\text{t}},j}^{\text{tx}}$ [sec], is expressed as follows:

$$t_{m_{\text{t}},i}^{\text{tx}} = t_{m_{\text{t}},i}^{\text{txframe}} + q(m_{\text{t}}, D_i^{\text{pcnt}}) \times T_{m_{\text{t}}}^{\text{slot}} + T_{m_{\text{t}}}^{\text{offset}}, \quad (4)$$

where transmission frame start time $\tilde{t}_{m_{\text{t}},i}^{\text{txframe}}$ [sec] is obtained by (6) in case of relay. The transmitter calculates $t_{m_{\text{t}},i}^{\text{tx}}$ with its own clock from the start time of the 0th frame t_0^{frame}.

C. Packet reception procedure

1) Initial synchronization: Suppose that receiving device $m_{\text{r}} \in \mathcal{M}\backslash\{0\}$ receives a packet D_i^{pcnt} from transmitting device $m_{\text{t}} = m_{\text{r}} - 1$ at time $t_{m_{\text{r}},i}^{\text{recv}}$ [sec] for the first time, it sets the synchronization time as $t_{m_{\text{r}}}^{\text{sync}} = t_{m_{\text{r}},i}^{\text{recv}}$. Then, the start time of the g-th frame $\tilde{t}_g^{\text{frame}}$ [sec] is calculated as

$$\tilde{t}_g^{\text{frame}} = t_{m_{\text{r}}}^{\text{sync}} - T_{m_{\text{r}}}^{\text{offset}} - q(m_{\text{r}} - 1, D_i^{\text{pcnt}}) \times T_{m_{\text{r}}}^{\text{slot}} + (g - 2 \cdot D_i^{\text{pcnt}} - m_{\text{r}} + 1) \times T_{m_{\text{r}}}^{\text{frame}}. \quad (5)$$

For a relay, start time $\tilde{t}_{m_{\text{r}},i}^{\text{txframe}}$ of the frame transmitting received packet D_i^{pcnt} is calculated as

$$\tilde{t}_{m_{\text{r}},i}^{\text{txframe}} = \tilde{t}_{m_{\text{r}}+2i}^{\text{frame}}. \quad (6)$$

Receiving device $m_{\text{r}} \in \mathcal{M}\backslash\{0\}$ calculates the start time of the receiving slot for next packet D_j^{pcnt} $(j > i)$ as

$$\tilde{t}_{m_{\text{r}},j}^{\text{rxSlot}} = \tilde{t}_{m_{\text{r}}-1+2j}^{\text{frame}} + q(m_{\text{r}} - 1, D_j^{\text{pcnt}}) \times T_{m_{\text{r}}}^{\text{slot}}. \quad (7)$$

2) Sequential synchronization (Clock drift compensation): Receiving device $m_{\text{r}} \in \mathcal{M}\backslash\{0\}$ updates the synchronization time to $t_{m_{\text{r}}}^{\text{sync}} = t_{m_{\text{r}},j}^{\text{recv}}$ once it receives the second or later packet D_j^{pcnt} $(j > i)$ at time $t_{m_{\text{r}},j}^{\text{recv}}$ [sec]. Then, based on the updated synchronization time $t_{m_{\text{r}}}^{\text{sync}}$, it uses (5)∼(7) to update $\tilde{t}_g^{\text{frame}}$ [sec], $\tilde{t}_{m_{\text{r}},j}^{\text{txframe}}$, and $\tilde{t}_{m_{\text{r}},l}^{\text{rxSlot}}(l > j)$.

D. Sleep time control

Each device reduces its power consumption by switching to sleep during periods when it is not transmitting or receiving. Receiving device $m_{\text{r}} \in \mathcal{M}\backslash\{0\}$ enters the sleep state and closes all receive windows as soon as the first packet is received from the transmitting device $m_{\text{t}} \in \mathcal{M}\backslash\{M-1\}$. Then, at $\tilde{t}_{m,i}^{\text{tx}}$ [sec], which is derived using Eq. (4), the device comes out sleep, forwards the packet, and switchs back into sleep. Then, at $\tilde{t}_{m,i}^{\text{rxSlot}}$ [sec], which is derived using Eq. ((7)), the Receiving device releases sleep for T [sec] and waits to receive packets. Sleep and transmission/reception are repeated in the same manner.

When the second and subsequent packets are received, due to clock drift, the start time of packet reception $\tilde{t}_{m_{\text{r}},j}^{\text{recv}} = \tilde{t}_{m_{\text{r}},j}^{\text{rxSlot}} + T_{m_{\text{r}}}^{\text{offset}}$ [sec] assumed by the receiving device and the actual start time of reception $t_{m_{\text{r}},j}^{\text{recv}}$ [sec] are different. In order to receive packets with consideration of this time discrepancy, the receiving device wakes up from sleep at the start time $\tilde{t}_{m,i}^{\text{rxSlot}}$ [sec] of the receive slot and waits for the reception of packets in the entire receive slot.

TABLE I
SIMULATION PARAMETERS

Parameters	Values
Number of devices M	4
Frame length T^{frame}	2.825 [sec]
Packet length T^{pckt}	$\{72, 123, 226\}$[msec]
Number of frequency channels K	4
Number of time slots Q	$2 \sim 40$
Duty cycle Δ_{DC}	0.01
Mean $[\mu_{\min}, \mu_{\max}]$	$[-1.91 \times 10^{-3}, 0.28 \times 10^{-3}]$
Variance $[\mu_{\min}, \mu_{\max}]$	$[9.59 \times 10^{-11}, 3.19 \times 10^{-10}]$
Power consumption (transmission) W_{tx}	99×10^{-3} [W]
Power consumption (reception) W_{rx}	18.15×10^{-3} [W]
Power consumption (sleep) W_{slp}	2.97×10^{-6} [W]

IV. SIMULATION AND RESULTS

The performance of the proposed method is evaluated using computer simulations. Table I shows the simulation parameters. The frame length, T^{frame} [sec], is determined so that it satisfies duty cycle Δ_{DC} for each frequency channel at each transmitting device as

$$T^{\mathrm{frame}} = T^{\mathrm{pckt}} / \left(2K \cdot \Delta_{\mathrm{DC}}\right). \tag{8}$$

Since T^{pckt} [sec] depends on bandwidth W and spreading factor SF, $W = 125$ [kHz] and spreading factor $SF = \{7, 8, 9\} = \{72, 123, 226\}$ [msec] are used for calculating T^{pckt} [sec] [12]. The offset time, T^{offset} [sec], is set to $T^{\mathrm{offset}} = (T^{\mathrm{slot}} - T^{\mathrm{pckt}})/2$ to transmit a data packet at the center of the time slot. Mean $\mu_{m_{\mathrm{r}}}$ and variance $\sigma^2_{m_{\mathrm{r}}}$ used for the clock drift value of each node are determined from uniform random numbers taking the range $[\mu_{\min}, \mu_{\max}]$ and $[\sigma^2_{\min}, \sigma^2_{\max}]$ of the minimum and maximum values obtained from experimental evaluation [11]. The arbitrary function used in the proposed method is $f_{\mathrm{q}}(m_{\mathrm{t}}, D_i^{\mathrm{pcnt}}) = f_{\mathrm{k}}(m_{\mathrm{t}}, D_i^{\mathrm{pcnt}}) = m_{\mathrm{t}} + D_i^{\mathrm{pcnt}}$. This paper does not consider any retransmissions. Power consumption W_{tx}, W_{rx}, W_{slp} are determined based on the parameters of 920 MHz band radio module [12].

If multiple packets are transmitted simultaneously at the same frequency channel and can be received by a receiving node, we assume that the packets lost, i.e., no capture effect is taken into account. If a receiving node can receive a packet during the estimated receive slot, it is considered that the packet reception is unsuccessful.

A. PDR

Figure 3 shows the packet delivery rate (PDR) performance as a function of elapsed time. Number of time slots Q is set to 2, which is the minimum number for the proposed method to operate. The PDR is determined by $1 - N_u^{\mathrm{fail}}(t)/N_u^{\mathrm{pckt}}(t)$ where $N_u^{\mathrm{pckt}}(t)$ denotes the total number of packets transmitted by the transmitter up to a certain time t. $N_u^{\mathrm{fail}}(t)$ denotes the total number of packets cannot be received by the GW until a certain time t. The PDR performance with the proposed sequential synchronization (clock drift compensation) is denoted as "w/ comp." and the case without sequential synchronization is denoted as "w/o comp". By using the proposed sequential synchronization method, the PDR

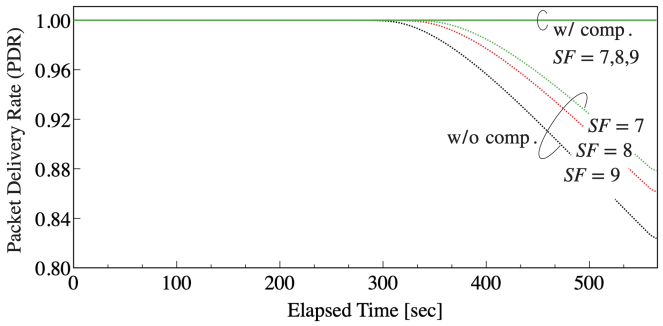

Fig. 3. PDR performance as a funtion of elapsed time

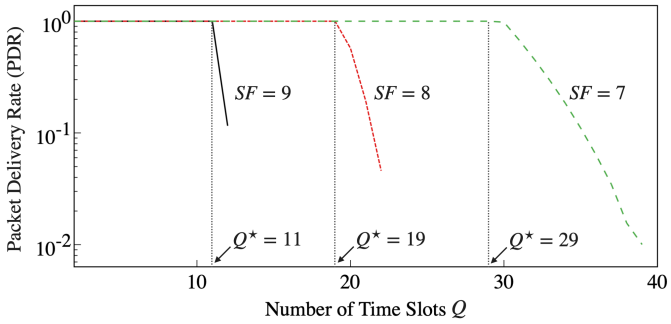

Fig. 4. The impact of number of time slots Q on PDR performance.

performance can be 1 regardless of SF. On the other hand, the PDR performance begins to deteriorates after a certain period of time. This is becuase of the accumulated clock drift, which prevents packets from fitting into the expected receive slots at the receiving device, resulting in a drop in PDR. In addition, the PDR begins to drop earlier when the SF is large, i.e., when T^{pckt} [sec] is large, because it is more susceptible to the effects of clock drift. Next, the effect of the number of time slots Q on the PDR performance of the proposed method is shown in Fig 4. The PDR is determined by $N_u^{\mathrm{suc}}/N_{\mathrm{pckt}}$ where N_{pckt} denotes the total number of packets transmitted and N_u^{suc} denotes the number of packets successfully received by the GW. Increasing number of time slots Q leads to more sleep time, but it shortens time slot length T^{slot}[sec] and time offset length T^{offset} within that slot, which makes it impossible to absorb the effect of clock drift. In LoRa, the larger SF becomes, the longer T^{pckt} [sec] becomes. Thus, the larger SF, the system becomes more susceptible to clock drift and the PDR performance starts to drop earlier. In particular, for the considered simulation parameters, $SF = 9$ can keep PDR of 1 up to $Q^\star = 11$, $SF = 8$ up to $Q^\star = 19$, and $SF = 7$ up to $Q^\star = 29$. If Q is increased beyond these values, the clock drift value cannot be absorbed and the PDR will start to drop.

B. The power consumption

Finally, this section provides the power saving effect of the proposed method. Since relays do not know the reception timing of the data packetin multi-hop communication, it is necessary to keep the receive window open at all times so that it can receive the data packet. Thus, the scenario when the relays always open the receive window is considered

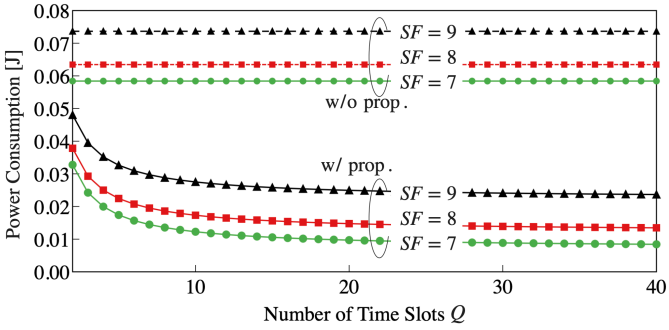

Fig. 5. The power consumption for the number of time slots Q

for comparison, i.e., *comparison method*. On the other hand, the relay can estimate the packet transmission time from its corresponding transmitter, so it can reduce power consumption by switching to sleep mode. First, the power consumption of a transmission frame $J_{\rm tx}$ can be calculated as

$$J_{\rm tx} = W_{\rm slp} \times (T^{\rm frame} - T^{\rm pckt}) + W_{\rm tx} \times T^{\rm pckt}, \quad (9)$$

where each relay becomes sleep mode except for packet transmission $T^{\rm pckt}$ [sec] during a transmission frame. This value is same for both the proposal and comparison method.

The comparison method always keeps the receive window open during the entire reception frame, so, the power consumption during the reception frame of the comparison method, $J_{\rm rx,conv}$, is expressed as follows

$$J_{\rm rx,conv} = W_{\rm rx} \times T^{\rm frame}. \quad (10)$$

The power consumption during the reception frame of the proposed method, $J_{\rm rx,prop}$, is calculated as

$$J_{\rm rx,prop} = W_{\rm slp} \times (T^{\rm frame} - T^{\rm slot}) + W_{\rm rx} \times T^{\rm slot}. \quad (11)$$

The power consumption required to forward one packet at each relay for the comparison method, $J_{\rm conv}$, and that for the proposed method, $J_{\rm prop}$, are respectively calculated by

$$J_{\rm conv} = J_{\rm tx} + J_{\rm rx,conv}, \quad (12a)$$
$$J_{\rm prop} = J_{\rm tx} + J_{\rm rx,prop}. \quad (12b)$$

The power consumption required to forward one packet at each relay for the number of time slots Q is shown in Fig 5. The comparison method is denoted as "w/o prop." and the proposed method as "w/ prop.", and the evaluation is performed at each spreading factor SF. "w/o prop." requires a constant amount of power consumption regardless of the number of time slots, because the receive window must always be open in the reception frame (10). However, the smaller SF is, the shorter the packet transmission time $T^{\rm pckt}$ [sec] is, the smaller power consumption $J_{\rm tx}$ of the transmission frame becomes, and overall power consumption $J_{\rm conv}$ is smaller. On the other hand, "w/ prop.", time slot length $T^{\rm slot}$ [sec] can be reduced by increasing the number of time slots Q, and power consumption of the received frame $J_{\rm rx,prop}$ is reduced. This means that the larger the number of time slots Q is, the lower the overall power consumption $J_{\rm prop}$ becomes. However, Fig. (4) shows that if number of time slots Q is increased to a certain number, packets with a time length $T^{\rm pckt}$ [sec] cannot be received within time slot length $T^{\rm slot}$ [sec], resulting in a significant decrease in PDR. In other words, there is a limit to the amount of power consumption that can be reduced while maintaining a high PDR in a system using the proposed method. Therefore, the power consumption $J_{\rm prop}$ (Fig. 5) when the number of time slots Q with PDR of 1 in Fig. 4 is the performance limit of the system. Considering these limits, especially for the present simulation parameters, the proposed scheme reduces the energy consumption by about 84.7% (at $Q^\star = 29$) for $SF = 7$, 76.5% (at $Q^\star = 19$) for $SF = 8$, 63.3% (at $Q^\star = 11$) for $SF = 9$, compared to the system without proposed scheme.

V. CONCLUSION

This paper proposed an autonomous decentralized adaptive resource allocation method for LoRaWAN multi-hop communication using the information in the header of the LoRaWAN packet. In particular, the proposed method maps transmission resources using the index of the order of hops and the packet counter. The proposed method can share the transmission resources between the transmitting and receiving sides, avoiding packet collisions and eliminating the time wasted in opening the receive window. Future work includes the incorporation and evaluation of the proposed method using actual equipment.

REFERENCES

[1] A. Lavric, et. al., "Internet of things software defined radio technology for LoRaWAN wireless communication: a survey," in *Proc. Int. Symp. Adv. Topics in Elect. Eng. (ATEE)*, pp. 1–4, Mar. 2021.
[2] P. Mutescu, et. al., "Wireless communications for IoT: energy efficiency survey," in *Proc. Int. Symp. Adv. Topics in Elect. Eng. (ATEE)*, pp. 1–4, May 2021.
[3] K. Mikhaylov, "On the uplink traffic distribution in time for duty-cycle constrained LoRaWAN networks," in *Proc. Int. Congress on Ultra Modern Telecommun. and Control Syst. and Workshops (ICUMT)*, pp. 1–6, Dec. 2021.
[4] *920MHz-Band Telemeter, Telecontrol and Data Transmission Radio Equipment*, ARIB STD-T108, Apr. 2021.
[5] Ministry of Internal Affairs and Communications, "Revision of Technical Standards for sophistication of 920 MHz band Power Saving Radio Systems (Draft)" https://www.soumu.go.jp/main_content/000452569.pdf, Accessed: Oct. 19, 2022.
[6] LoRa Alliance Technical Committee, "LoRaWAN™1.1 specification". [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/lorawantm_specification_-v1.1.pdf
[7] J. Ortin, et. al., "How do ALOHA and Listen Before Talk Coexist in LoRaWAN ?" in *Proc. IEEE Int. Symp. on Personal, Indoor and Mobile Radio Commun.*, 2018, pp. 1–7.
[8] J. Haxhibeqiri, et. al., "Low overhead scheduling of lora transmissions for improved scalability," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3097–3109, 2019.
[9] M. O. Farooq, "Multi-hop communication protocol for LoRa with software-defined networking extension," *Internet of Things*, 2021.
[10] LoRa Alliance Technical Committee, "LoRaWAN™1.1 Specification". [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/lorawantm_specification_-v1.1.pdf
[11] K. Tsurumi, et. al., "Simple clock drift estimation & compensation for packet-level index modulation and its implementation in lorawan," *IEEE Internet Things J.*, 2022.
[12] *Specified Low Power Radio Module ES920LR2 Data Sheet Version 1.05*, EASEL CO., Ltd., Jan. 19, 2022.