EARN: Enhanced ADR With Coding Rate Adaptation in LoRaWAN

Junhyun Park, Kunho Park, Hyeongho Bae, and Chong-Kwon Kim

***Abstract*—As low-power wide-area (LPWA) networks emerge as a cost-effective choice of technologies for city-wide Internet-of-Things (IoT) applications, LoRaWAN, one of the most promising unlicensed band techniques, has received much attention from academia. LoRaWAN presents a set of tunable transmission parameters, along with an adaptive data rate (ADR) mechanism, to promote the best performance under the variable link state. But the performance of ADR, whose design neglects the complex correlation between such parameters, is yet to be practical in terms of both efficiency and scalability. In this article, we derive theoretical performance models of class A unconfirmed-mode LoRaWAN, focusing on the impact of coding rate (CR), a parameter that has not been explored in prior researches. Then, we present *EARN*, an enhanced greedy ADR mechanism with CR adaptation, to optimize the tradeoff between delivery ratio and energy consumption. In *EARN* design, we leverage the capture effect to increase the survival rate of colliding signals and introduce a concept of adaptive SNR margin to endure noisy link states. We validate our models and the feasibility of the CR adaptation with an empirical study, and large-scale simulations reveal that our method outperforms the conventional schemes.**

***Index Terms*—Adaptive data rate (ADR), coding rate (CR), Internet of Things (IoT), LoRaWAN, low-power wide-area (LPWA) networks, scalability.**

I. Introduction

THE ADVANCES in information and communication technology have brought the Internet-of-Things (IoT) paradigm, where a massive number of things connect to each other. Recently, IoT has been expanding its application from small scale to city-wide IoT, such as smart metering and smart environmental monitoring. To allow power-efficient and cost-effective wide-area connectivity, different low-power wide-area (LPWA) protocols emerged as promising solutions. The most representative protocols, LTE-M, NB-IoT, LoRaWAN, and Sigfox, have the common characteristics of connecting thousands of devices at low data rates, but they also have their own capabilities to meet specific application needs. Licensed LPWA technologies, such as LTE-M and NB-IoT, present reliable communications at relatively higher data rates owing to cellular infrastructures. However, the protocols to grant accesses only to authenticated users are often so complicated that it may bring about 2000 retransmissions and energy overheads. LPWA protocols on the unlicensed bands, such as LoRaWAN and Sigfox, require sophisticated interference control, yet enables the impromptu establishment of private and public networks at a much lower cost. In this article, we focus on LoRaWAN, whose simple use of protocols and high availability has attracted immense attention from both academia and industry.[1]

LoRaWAN, with its robust modulation scheme, is capable of providing reliable communications under harsh link conditions and, therefore, a higher chance of getting the combination of worthy performance and price. With efforts to develop practical solutions by embracing these benefits, many researchers started questioning the scalability that LoRaWAN can serve in real-world environments [1]–[3]. An experiment revealed that LoRaWAN adopting pure ALOHA is vulnerable to collisions in typical smart city deployments, and only 64 devices can achieve a transmission success rate of 0.9 or higher [1]. Some studies [4]–[6] mitigated collisions by distributing the traffic load of devices on parallel demodulation paths but instead reduced the network coverage to less than half of the maximum communication range claimed by the LoRaWAN standard. As such, highly constrained scalability of current LoRaWAN is only suitable for low-rate sparse applications. In this article, we aim to improve coverage and capacity of LoRaWAN, along with the efficient use of energy for battery-powered devices, so that it can meet the various needs of emerging future IoT applications.

In an environment, where one base station connects and serves a massive number of IoT devices, channel and radio resources should be utilized very efficiently. To this end, LoRaWAN presents various transmission parameters that manage the resources to regulate the link performance. Proper parameter configuration is arguably essential to provide stable link quality under different application scenarios, and fail to do so significantly drains device power and degrades network scalability. Typically, LoRaWAN devices can be configured to use different central frequencies (CF), bandwidth (BW), spreading factors (SFs), transmission powers (TP), and coding rates (CRs). These parameters, rather than being independent, are complexly correlated with each other to affect link

Manuscript received March 20, 2020; revised June 7, 2020; accepted June 16, 2020. Date of publication June 30, 2020; date of current version December 11, 2020. This work was supported in part by the Institute for Industrial System Innovation and the Institute of Engineering Research at Seoul National University, and in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MSIP) under Grant 2016R1A5A1012966. *(Corresponding author: Chong-Kwon Kim.)*

The authors are with the Department of Computer Science and Engineering, Seoul National University, Seoul 08826, South Korea (e-mail: jhpark041@snu.ac.kr; parkkh0812@snu.ac.kr; bhh214@snu.ac.kr; ckim@snu.ac.kr).

Digital Object Identifier 10.1109/JIOT.2020.3005881

[1] Sigfox networks are operated by Sigfox and its regional partners.

performance. Due to this nature, it is challenging to select one suitable configuration, which best fits a given link condition.

As a means to adjust some parameters, such as SF and TP, LoRaWAN exploits the adaptive data rate (ADR) mechanism. However, it assigns transmission parameters to devices with a very conservative policy in a way that the medium they share gets quickly saturated. Other ADR proposals [5], [6] that were suggested afterward tried to distribute the load evenly over different SFs, whose orthogonality is known to prevent inter-SF collisions, in efforts to optimize fairness between devices. But the fair distribution of SF did not account for energy efficiency, leaving edge devices to constantly suffer from short lifespan regardless of cell size and link condition. We also found that they are only practical in small cells where all SFs are available. After all, we leverage several facts that prior researches have overlooked to devise our own enhanced ADR scheme, *EARN*. First, there is no provision for proper CR assignment and its tradeoffs. The *capture effect* is also being neglected or misused, even though it can increase the survival rate of colliding signals. If the performance of devices, taking these into account, can closely be approximated, we will be able to adapt the parameters for a given link condition.

The main contributions of this article can be summarized as follows. We theoretically model the link performance of class A unconfirmed mode LoRaWAN in terms of PDR, goodput, and energy efficiency. In the meantime, we observe the tradeoffs of transmission parameters focusing on CR, which has been overlooked in prior works; the adoption of non-default CRs leads to a longer frame length but increases reliability to afford extended communication range. Then, we present *EARN*, an enhanced ADR mechanism that uses the developed models to predict link performances and assigns the best parameter set. In the *EARN* design, an LoRaWAN server maintains an aggregated load status for each SF and SNR to estimate collision probabilities within the impact range of collision and the capture effect. Then, by extension, *EARN* employs a concept of adaptive SNR margin to endure link variations, as a tight estimation of the link can severely degrade the performance in harsh environments. In the end, the thorough evaluations based on large-scale simulations, which we justify with our empirical pilot experiment, demonstrate that the proposed algorithm completely outperforms the conventional methods in all aspects.

The remainder of this article is organized as follows. Section II provides preliminaries and research trends on LoRa/LoRaWAN. In Section III, we formulate the link performance models and analyze the impacts of CR in detail. Then, we explain the *EARN* algorithm along with its extension in Section IV. Section V describes the simulation setup and compares *EARN* with other ADR schemes. Finally, in Section VI, we conclude this article and present our future directions.

II. Preliminaries

A. LoRa/LoRaWAN

LoRa is a PHY layer developed by Semtech, which adopts chirp spread spectrum (CSS) modulation for robust and long-range communications. LoRaWAN, an open standard maintained by the LoRa Alliance, operates on top of LoRa as the MAC layer protocol [7]. A LoRaWAN network uses a star-of-stars topology consisting of three entities: 1) network servers; 2) gateways (GW); and 3) end devices (ED). A network server can have multiple GW, each of which is connected via the IP network to the server and covers up to thousands of ED with LoRa wireless links.

In LoRaWAN, there are three types of ED: classes A, B, and C. Note that class A devices receive downlinks only after uplink transmissions. In contrast, class B and class C devices schedule extra downlink opportunities; class B exploits beacon-based synchronous receive windows, and class C continuously listens for downlinks. In each class, a *confirmed* message must be acknowledged by the receiver while an *unconfirmed* message does not require an acknowledgment.

TABLE I
Required SNR to Demodulate Signal on Each SF

Spreading Factor	7	8	9	10	11	12
$\boldsymbol{SNR_{req}}$ ***(dB)***	-7.5	-10	-12.5	-15	-17.5	-20

B. Transmission Parameters

LoRaWAN EDs have several adjustable transmission parameters to adapt and best facilitate link performances; CFs are the central frequencies of LoRaWAN channels spread along a wide range of spectrum from 137 to 1020 MHz abided by regional regulations. BW is the bandwidth around CF and is directly proportional to the transmission capacity. BW of 125 kHz is typically used out of three configurable options, 125, 250, and 500 kHz. SF, ranging from 7 to 12, denotes the number of raw bits that can be encoded in a symbol. It also affects the symbol rate of LoRa, $\mathrm{BW}/2^{\mathrm{SF}}$, where a symbol is modulated as a chirp signal of 2^{SF} chips and transmitted at a given chip rate (i.e., BW). As can be inferred here, the data rate is nearly halved as SF increases. However, the higher SF makes the frame more robust to interference and introduces the lower $\mathrm{SNR}_{\mathrm{req}}$ [8], the minimum SNR needed to demodulate a signal (Table I). Here, note that LoRa ensures orthogonality between different SFs, enabling the simultaneous receptions of the frames. TP is the transmission power ranging from -4 to 20 dBm, but available power options for data rate adaptation are limited to a few by the regional regulations [9]. Here, the use of TP over 17 dBm is restricted by a 1% duty cycle. Using a higher TP extends the transmission range while consuming more energy. CR refers to the proportion of useful payload to total data bits, including the additional parity bits for forward error correction (FEC). LoRa adopts the Hamming FEC method and employs CRs of 4/5, 4/6, 4/7, and 4/8. We will thoroughly discuss the tradeoff of CR in Section III.

C. Adaptive Data Rate

The LoRaWAN employs the link-based ADR mechanism, which dynamically adjusts the transmission parameters of an ED to link states. The main purpose of ADR is to optimize data rates, but it also affects the energy consumption and capacity of the network. It consists of two different concurrently running algorithms, each on the server side [10] and ED side [11].

The ED-side mechanism, as complicated operations may drain a limited amount of energy, plays a one-sided role of simply lowering data rates to maintain or restore the connectivity to a GW. An ED maintains an uplink counter for ADR, and if it does not receive a downlink from a GW within a certain number of uplinks defined by *AdrAckLimit* and *AdrAckDelay*, TP or SF is increased by one step. Either case where a downlink arrives from the GW or the counter meets the limit, the above procedure repeats after initializing the counter back to zero. Here, note that SF is increased after TP is first raised to its maximum possible option.

The server-side ADR starts by measuring the link budget from uplink messages from each ED. $\mathrm{SNR}_{\mathrm{margin}}$ defines the link budget and is obtained by subtracting $\mathrm{SNR}_{\mathrm{req}}$ and *deviceMargin* of 10 dB from the measured SNR. A positive link budget implies that the data rate should be lowered and a negative value implies the opposite. A server also uses the margin to minimize SF first and then adjusts TP. When the configuration by the server is passed to an ED as a downlink control message called *LinkADRReq*, the ED can decide whether to accept it.

D. Capture Effect

The capture effect is a phenomenon between two colliding signals at the receiver, where only the stronger survives. When two signals are nearly equal in strength, both are lost. So it is a better strategy to keep one of the two colliding signals alive by giving one enough power to suppress the other. Bor *et al.* [1] experimentally demonstrated the capture effect in LoRa and noted its potential. Some studies [4]–[6], [12]–[14] to disentangle collisions in Section II-E can also take advantage of this for better performance.

E. Related Works

In the early stage of LoRaWAN studies, on-site measurements were prevalent to verify the coverage of the protocol. Although the transmission range has been found to be largely dependent on field characteristics, such as topography and temperature [15], [16], typical LoRaWAN deployments confirmed the wide transmission ranges of around 3 and 7 km in urban and suburban scenarios, respectively. However, a number of researches have since been introduced to question the applicability of the protocol to real-world use cases. Bor *et al.* [1] and Georgiou and Raza [2] condemned the collision, whose rate increases exponentially as the number of EDs grows, as a dominant cause of scalability constraints. It was shown in [1] that only 64 EDs can achieve a transmission success rate of 0.9 or higher in typical smart city deployments. Recently, Ghena *et al.* [3] defined a metric, bit flux, to quantify different LPWANs in terms of throughput over a coverage area, and evaluated LoRaWAN as suitable only for the low-rate sensing applications.

In this regard, many LoRaWAN studies nowadays aim to improve network connectivity with respect to coverage and density while ensuring a certain quality of service. Many researchers have tried to overcome the limited capabilities of LoRa hardware. Eletreby *et al.* [17] implemented Choir on GW to leverage radio imperfections in frequency, time, and phase to decode a large number of transmissions simultaneously. They also managed EDs to deliver messages in a collaborative manner to enable communication beyond their reach. Charm [18] used coherent combining, which pools signals from multiple GWs and decodes them in the cloud, to increase the resiliency of weak signals. From an energy perspective, backscattering techniques combined with LoRa significantly reduced power consumption [19]–[21]. However, the above methods require modifications to hardware, thereby increasing the deployment cost.

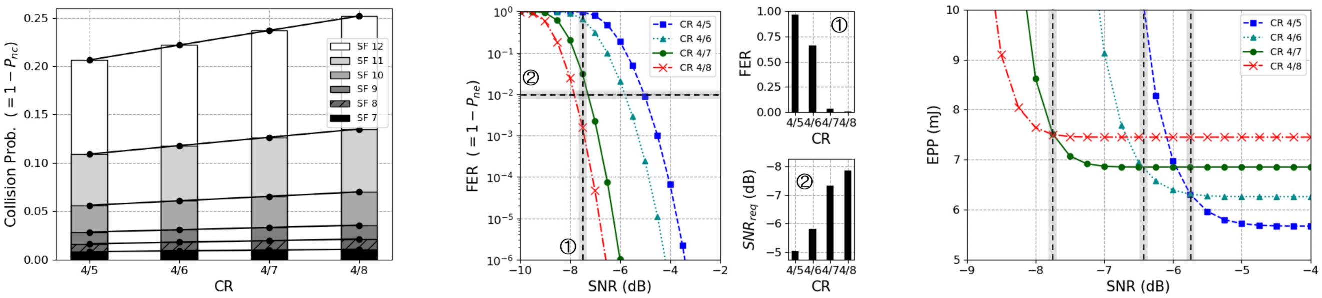
Next, there are studies to improve performance by efficiently utilizing the channel and radio resources. The optimal number of message replication to exploit the time diversity and enhance the reception rate was explored in [12], and different channel access mechanisms, such as slotted ALOHA [22] and CCA-based CSMA [23], were put on evaluations. However, these approaches [12], [22], [23] essentially increase uplink or downlink traffic and burden EDs with additional operations (e.g., time synchronization). Instead, the research on transmission parameter allocation, to which our study also belongs, have become popular due to its low cost and high effectiveness. In particular, we note that the parameter allocation is most fundamental in LoRa transmissions, and therefore, can be used in conjunction with many other aforementioned approaches to achieve synergistic performance gain.

The study of parameter allocation mainly focuses on the distribution of SF or, in addition, the ADR improvements. Many have taken approaches to properly disperse the high demands for the lowest SF to other SFs [5], [13]. EXPLoRa [5] defined a fixed SF distribution ratio and assigned SFs to EDs in order of distance to the GW so that all available SFs have the same air occupancy time; in the meanwhile, only the maximum TP was employed, and others suggested controlling TP together with SF [4], [6], [14]. FADR [6] used genetic algorithms to find the optimal SF distribution ratio that equalizes the collision probability among all available SFs. It then adjusted TP to fit in a given $\mathrm{SNR}_{\mathrm{req}}$ with a fixed SNR margin. But unfortunately, none of the prior parameter allocation schemes or ADR proposals have noted the impact of CR, and the capture effect has also been neglected or misused.

Meanwhile, our ADR scheme, *EARN*, leverages those overlooked factors and fine tune the link performance of EDs. In Section V, we compare and evaluate *EARN* with the legacy ADR and FADR.

III. Link Performance Modeling

In this section, we model the link performance of a single gateway LoRaWAN. The network consists of class A EDs that transmit unconfirmed messages, assuming the most common



(a)

(b)

(c)

Fig. 1. Analysis of CR impact on LoRaWAN link performance. CR has a tradeoff between efficiency and scalability. (a) CR impact on collision probability. (b) CR impact on frame error rate. (c) *EPP* versus *SNR*.

application scenario. Unless otherwise stated, we assume the following parameters settings for the validation and analysis of the formulated models; 125 kHz BW, SF 7, CR 4/5, and (or) PHY payload size of 12 B, which is minimal due to the constraint on MAC message formats. The derived models are later used for link performance predictions to assign optimal parameters.

A. Collision Probability

As the first building block of the models, we derive the probability of having no collisions (P_{nc}). LoRaWAN adopts pure ALOHA as the medium access control mechanism and the probability is given as e^{-2G}, where G is the offered load to the network. G is obtained by multiplying *time-on-air* (T_{frame}), the amount of time it takes to send a frame, by the *arrival rate* (λ). Then, we get

$$P_{nc} \approx e^{-2\lambda T_{\text{frame}}}$$

which simply implies that the greater the number of frames arriving on the network is or the larger the payload is, the higher the likelihood of collisions becomes.

T_{frame} consists of the time taken for the preamble (T_{preamble}) and payload (T_{payload}) transmissions. Given $2^{\text{SF}}/\text{BW}$ as the symbol duration (T_{symbol}), T_{frame} equals the total number of symbols that make up the preamble and payload multiplied by T_{symbol}. The number of symbols in the preamble is $n_{\text{preamble}} + 4.25$, where n_{preamble} is 8 in most regions. The number of symbols needed to transmit the PHY payload with an explicit header, n_{payload}, is as follows [8]:

$$n_{\text{payload}} = 8 + \max\left(\left\lceil \frac{8\text{PL} - 4\text{SF} + 28 + 16C}{4(\text{SF} - 2DE)} \right\rceil (cr + 4), 0\right).$$

Here, PL is the number of payload bytes, $C = 1$ indicates the presence of the CRC payload, and DE is a data rate optimization option which is set to 1 when SF 11 and 12 are employed. The lowercase cr indicates a CR index ranging from 1 to 4, each of which corresponds to a rate from 4/5 to 4/8.

Obviously, CR changes the airtime of a modulated LoRa signal, thereby affecting P_{nc}. LoRa employs Hamming FEC as the coding scheme, and CRs of 4/5 and 4/6 each exploit 1 and 2 parity bit(s) for error detection. These rates are vulnerable to interference as they can only detect errors. A single-bit error, on the other hand, can be corrected by putting 3 or 4 additional bits at a CR of 4/7 or 4/8. LoRa partitions PHY payload into bit chunks corresponding to four symbols and appends parity bits to form cr additional symbol(s). Fig. 1(a) illustrates CR impact on the collision probability $(1 - P_{nc})$ over different SFs when $\lambda = 0.1$. The probability at SF 12 increases by nearly 22% as CR changes from 4/5 to 4/8 to contain additional parity bits. Employing a higher cr seems to be a bad choice as it always increases the collisions. However, higher cr has a significant benefit in that it prevents potential error under harsh link conditions.

B. BER and FER

The analytical expression for the bit error rate (BER) of CSS modulation can be derived from E_b/N_0, the energy per bit to noise power spectral density ratio, and the Q-function [24] as

$$\text{BER} = Q\left(\frac{\log_{12}(\text{SF})}{\sqrt{2}} \cdot \frac{E_b}{N_0}\right).$$

Then, we express E_b/N_0 as a function of SNR

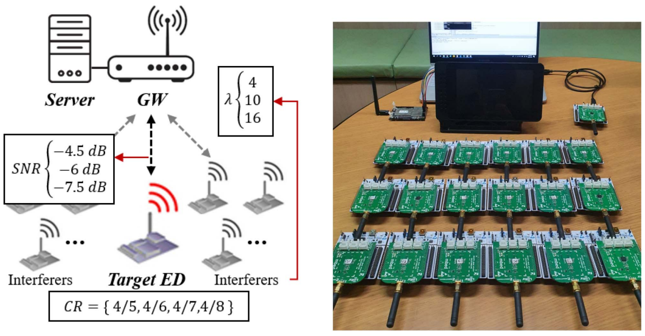
$$\frac{E_b}{N_0} = \text{SNR} - 10\log\left(\frac{R_s \cdot \text{SF} \cdot \text{CR}}{\text{BW}}\right)$$

to examine the BER change in terms of more familiar parameters.

In the previous part, we have found the total number of symbols in a modulated signal. Since each symbol encodes SF bits, a frame contains $\text{SF} \cdot (n_{\text{preamble}} + 4.25 + n_{\text{payload}})$ bits. Then, the probability of not having an error, P_{ne}, for a k-bit frame can be expressed as follows, reflecting the error detecting or correcting capability of each CR:

$$P_{ne} = \begin{cases} (1 - \text{BER})^k, & \text{if } cr = 1, 2 \\ (1 - \text{BER})^k + k & \\ \quad \cdot \text{BER} \cdot (1 - \text{BER})^{k-1}, & \text{if } cr = 3, 4. \end{cases}$$

Fig. 1(b) shows the frame error rates ($\text{FER} = 1 - P_{ne}$) of different CRs as functions of SNR. As stated, a frame with 12-B long PHY payload using SF 7 is taken into account. Note that each CR has a different tolerance for SNR but shows a common pattern in which FER sharply increases as SNR drops to and beyond SNR_{req}. The impact of CR on FER becomes noticeable near −7.5 dB, SNR_{req} of SF 7; FERs of CR 4/5 and 4/6 are over 0.65 while CR 4/7 and 4/8 manage to keep


Server
GW
$\lambda \begin{cases} 4 \\ 10 \\ 16 \end{cases}$
$SNR \begin{cases} -4.5\ dB \\ -6\ dB \\ -7.5\ dB \end{cases}$
Interferers
Target ED
Interferers
$CR = \{4/5, 4/6, 4/7, 4/8\}$

(a) (b)

Fig. 2. (a) Experimental setup and (b) devices used to demonstrate the necessity of CR adaptation.

the rates sufficiently low, thanks to single-bit error correction. For SNR_{req} that we theoretically derive for each CR to keep FER under 0.01, CR 4/8 endures the toughest link conditions up to nearly −8 dB.

C. Link Performance

We now model the link performance of LoRa on different criteria; PDR, goodput, and energy efficiency. First, we approximate PDR by combining the previously obtained P_{nc} and P_{ne} and it inherits the characteristics of both (PDR $\approx P_{nc} \cdot P_{ne}$). The goodput of an ED is obtained straightforwardly by multiplying PDR, physical payload length, and λ

$$\text{Goodput} = \text{PDR} \cdot \text{PL} \cdot \lambda.$$

Another important performance metric for battery-operated LoRaWAN EDs that are deployed over a wide area is energy efficiency. We define energy per packet (EPP), the energy consumed to successfully transfer a packet, and formulate it as follows:

$$\text{EPP} = \frac{V_{tx} \cdot I_{tx} \cdot T_{\text{frame}}}{\text{PDR}}.$$

LoRa's supply voltage, V_{tx}, is 3.3 V and current, I_{tx}, whose maximum value is 125 mA, is decided by TP [8]. Which metric to use depends on the application purpose, but EPP is what we use in our algorithm to capture the tradeoff between success rate and the energy consumed.

To observe the CR impact on the link performance, we plot the theoretical EPP of an ED in relation to SNR under the load of $\lambda = 0.1$ in Fig. 1(c). In the case of high SNR, it is better to employ CR 4/5 without having to endure overheads incurred by additional parity bits since bit errors are less likely to occur. If SNR falls to or beyond SNR_{req}, the loss at P_{ne} dominates the gain in P_{nc} by low cr. Therefore, EDs should sequentially choose a higher cr as SNR deteriorates. Down until SNR −5.8 dB, CR 4/5 achieves the best EPP. Then, it is followed by 4/6 until −6.4 dB, 4/7 until −7.8 dB, and 4/8 afterward. In the same context, one can also find the tradeoff of CRs in the λ domain; a high CR, such as 4/5 or 4/6, is preferred as a medium gets saturated. Therefore, it is often hard to determine which CR is the best, especially near the SNR_{req}.

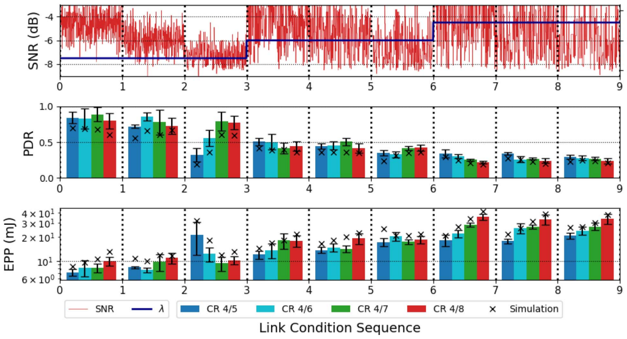


Fig. 3. LoRaWAN link performances of different CRs under a varying link condition (SNR and λ).

D. Pilot Experiment on the Impact of CR

To suggest the necessity and effectiveness of CR adaptation on the ADR mechanism, while at the same time demonstrating that our theoretical models are also valid in real-world environments, we designed and conducted a pilot experiment as in Fig. 2(a). We constructed a small LoRaWAN cell with a GW connected to a server and 20 EDs. Here, one target ED was chosen to measure the performances of each CR under different link conditions, and the other EDs operated as interferers to transmit noise signals. Fig. 2(b) shows the commercially available LoRaWAN devices we used. We built a GW using the RAK831 board attached to a Raspberry Pi 3, which also took the role of a server. We implemented EDs with SX1272MB2DAS communication modules attached to NUCLEO-L073RZ boards. The GW was located on the third floor of an office building, and the EDs were placed at different positions on the sixth- and seventh-floor corridors of the same building about 50 m apart from the GW with many walls in between. Only the position of the target ED was relocated depending on the SNR level we tried to emulate, and interferers were fixed in their place to generate noise of constant strength. The payload length and message rates of the interferers were controlled depending on the λ level we tried to emulate, while the target ED maintained its settings; 180 messages of 12 B payload on each 15 min long link condition. As impacts from different SF are inherently minimal due to their orthogonality, we assumed single SF scenarios where all EDs were configured to use SF 7. Then, we fixed TP to the default value of 14 dBm. As it is proven in many research of orthogonality of different SFs, we make it simple and all EDs were configured to use fixed SF of 7 and TP of 14 dBm throughout the whole experiment and changed CR on each scenario. In each round, the server recorded the PDR and EPP of the target node based on the number of successfully received packets. We conducted multiple runs of the test at late nights where the impacts of human obstacles are minimized.

The top of Fig. 3 shows the nine different link conditions experienced by the target ED; −7.5, −6.0, −4.5 dB for low, medium, and high SNR and arrival rates (λ) of 4, 10, 16 under low, medium, and high traffic load, on average. The center of Fig. 3 shows the averaged PDR of the target ED for each CRs under different link conditions. For a clear and concise interpretation of the experimental results, we compare the three-link conditions in the front, which have low traffic loads, and the three from the back, which has heavy traffic loads. A common characteristic observed in all cases is that, naturally, PDR degrades with SNR. But an important thing to note is that the degree of performance degradation depends on CR the target ED uses. When the channel is lightly loaded, and therefore, the impact of collisions is small, SNR is the dominant factor that affects PDR. Under stable link conditions with SNR of −4.5 and 6 dB, all CRs guarantee a relatively high PDR. However, as SNR drops to −7.5 dB, PDRs of high CRs (4/5 and 4/6) are significantly reduced. In the meanwhile, low CRs (4/7 and 4/8) manage to keep their PDRs, confirming that the error correction of the coding scheme actually benefits the packet delivery on the bad link. The situation is slightly different when the traffic load is high. Low CRs no longer benefit from error correction under the bad link condition as before but rather worsen the PDR. This is because the low CR introduces a higher chance of collisions with the enlarged frame length under heavy traffic scenarios, where the collision rate acts as a dominant factor affecting PDR.

Another interesting observation can be derived from the bottom of Fig. 3 depicting EPP measurements of each CR. Even the CR with the highest PDR may not be the best option in terms of energy efficiency. As shown in the first link condition, CR 4/5 has the lowest EPP while CR 4/7 has the highest PDR. This implies that the CR 4/5, when energy is taken into account together, is the most efficient option. In Fig. 3, we also put the simulation results, which are driven based on the performance models we formulated above, to justify the evaluations that follow in Section V. Although there are small performance gaps between the two, the tradeoff behavior of CR depending on the link condition is similar; and the best CR option in each link condition generally coincides with each other.

The above results confirmed the impact of CR on target metric optimization and inspired us to design an ADR algorithm that adaptively manages CR. In general, a high CR allows EDs to utilize energy and network capacity efficiently. Conversely, given the assumption that the link is not congested, a low CR improves the network coverage by helping EDs that are located in the distance or have poor link conditions to reach a GW. The tradeoff is similar to that of SF, but the scale is smaller, allowing precise calibration of the link performance.

Algorithm 1 *EARN* Algorithm

```
Input: Uplink info and the last $\{SF_i, SNR_i, G_i\}$ of $ED_i$, and $AggregatedLoad$
Output: $\{SF, TP, CR\}$ configuration for $ED_i$
# Initialize and load the previous config info
$EPP_{best} \leftarrow \emptyset$
$Conf \leftarrow \{\emptyset, \emptyset, \emptyset\}$
$\{SF_i, SNR_i, G_i\} \leftarrow$ Last $\{SF_i, SNR_i, G_i\}$ of $ED_i$
# Retrieve the load portion of $ED_i$
$AggregatedLoad[SF_i][SNR_i] -= G_i$
# Update the arrival rate and path loss info of $ED_i$
$\lambda_i \leftarrow$ avg($FCntDiff/TimeGap$ of 20 recent frames)
$PathLoss_i \leftarrow$ avg($PathLoss$ of 20 recent frames)
for $sf = 7, 8 \ldots 12$ do
    for $tp = 2, 5 \ldots 14$ do
        for $cr = 1, 2 \ldots 4$ do
            $T_{frame} \leftarrow getTframe(PL_i, BW_i, sf, cr)$
            $G_{temp} \leftarrow T_{frame} \cdot \lambda_i$
            $SNR_{temp} \leftarrow getSNR(PathLoss_i, tp)$
            $G_{col} \leftarrow G_{temp}$
            # Sum up the load that may collide with $ED_i$
            for $j = SNR_{temp} - 6 \ldots SNR_{max}$ do
                $G_{col} += AggregatedLoad[sf][j]$
            end for
            $P_{nc} \leftarrow PncModel(G_{col})$
            $P_{ne} \leftarrow PneModel(PL_i, BW_i, sf, cr, SNR_{temp})$
            $PDR \leftarrow P_{nc} \cdot P_{ne}$
            $Energy \leftarrow getEnergy(T_{frame}, tp)$
            $EPP_{temp} \leftarrow Energy/PDR$
            # Keep the best config for $ED_i$
            if $EPP_{best} == \emptyset$ or $EPP_{best} > EPP_{temp}$ then
                $EPP_{best} \leftarrow EPP_{temp}$
                $Conf \leftarrow \{sf, tp, cr\}$
                $\{SF_i, SNR_i, G_i\} \leftarrow \{sf, SNR_{temp}, G_{temp}\}$
            end if
        end for
    end for
end for
# Update the aggregated load info
$AggregatedLoad[SF_i][SNR_i] += G_i$
return $Conf$
```

IV. *EARN*: Proposed Algorithm

In this section, we explain our enhanced ADR algorithm, *EARN* (Algorithm 1), to optimize the tradeoff between success rate and energy consumption. *EARN* takes into account an additional transmission parameter, CR, upon data rate assignments. It also leverages the capture effect to distribute the TP of EDs. The LoRaWAN server infers configurations and path loss of all EDs from uplink messages without requiring them to explicitly hand over extra data. Upon every uplink arrivals from EDs, *EARN* predicts link performances by substituting all possible combinations of transmission parameters into the models we designed in Section II and chooses the best among them. Our objective function aims to minimize EPP, the energy efficiency metric, and is as follows:

$$\underset{x}{\text{minimize}} \quad \text{EPP}(x)$$

where x is the set of target parameters, $\{\text{SF}, \text{TP}, \text{CR}\}$. The server sends a downlink control message in a receive slot of an ED in response to the uplink, only when the computed setting is better than that of the ED. In *EARN*, the server takes over all the complex computations, and therefore, does not risk the lifespan of EDs. Rather, the parameters, which are chosen to minimize EPP, improve the energy efficiency of EDs to help LoRaWAN keep its promise of years-long battery life.

The input to *EARN*, as in Algorithm 1, includes the aggregated load status of an LoRaWAN, ED$_i$'s uplink frame, and ED$_i$'s $\{\text{SF}_i, \text{SNR}_i, G_i\}$ measurements at the previous *EARN* operation. These are configured SF, estimated SNR and load which are kept by the server for ED$_i$, and the default value of G_i is zero. An LoRaWAN server that operates *EARN* maintains an aggregated load status for each SF and SNR to estimate collision probability within the impact range of collision and the capture effect. To be specific, the server can tell the network demand of all EDs whose SF is 7, and SNR at the GW is, on average, around -4 dB. After initialization, it subtracts from aggregated load the portion of ED$_i$, if there is any ED$_i$'s contribution from the past operation (line 4). Then, *EARN* can obtain parameter settings, arrival time, frame count, and SNR from the uplink frame, which are then translated into the input for *EARN* to run a set of thorough substitutions to the performance models. The message rate (λ) is computed by dividing the frame count difference of two successive frames from an ED by arrival time gap (line 5). The path loss of an ED is inferred from SNR measured at the GW (line 6). The λ and path-loss estimation may not be accurate and fluctuate from time to time. So, *EARN* takes the averaged values of up to 20 latest measurements to mitigate the error. After all the ingredients are ready, *EARN* predicts link performance by try feeding all SF, TP, and CR combinations to the model sequentially. For each iteration, we first compute T_{frame} whose length varies by SF and CR and multiply it with λ_i to get G_{temp}, the load of ED$_i$ (lines 10 and 11). Then, we also estimate SNR for the given TP (line 12). These are used to consolidate the loads (G_{col}) within the SNR range by which ED$_i$ is affected (lines 13–16). Then, the rest procedures to estimate EPP are straightforward. P_{nc}, the probability that no collision occurs, can be estimated by substituting G_{col} to the P_{nc} model. P_{ne}, the probability that no error occurs, can be estimated by substituting PL, BW, SF, CR, and SNR$_{\text{temp}}$ to the P_{ne} model. T_{frame} and TP are used to compute the energy consumed on transmitting a packet. *EARN* assigns to ED$_i$ the SF, TP, and CR values with the lowest expected EPP (lines 22–25). Finally, the corresponding SF, SNR, and load estimates for ED$_i$ are recorded to the latest network load status of the server (line 30).

The parameters assigned by *EARN* to satisfy SNR$_{\text{req}}$ and achieve minimal EPP work well in ideal link conditions. However, the tight parameter match to a predicted link condition is not always the best choice, as noise and fluctuation in path loss adversely affect the *EARN*'s link estimation in the real-world scenarios; there is a higher chance of losing frames that have lower SNRs than the expected ones. So we make up for the weaknesses that allow *EARN* to adapt to dynamic channel conditions. Unlike the legacy ADR that assigns parameters by placing a huge fixed-size margin of 10 dB on a given link budget, we let *EARN* employ an *adaptive margin* of variable size. In *EARN*, the size of the margin to adjust the link budget measurement is decided by averaging the path-loss standard deviations from up to 20 latest frames. The above revision can be easily reflected to the original *EARN* by subtracting the marginal value from the estimated SNR which is passed as an input to the P_{ne} model in line 20. We named the algorithm *EARN-AM* (adaptive margin), and it brings a protective margin against channel noise, just like the legacy ADR, but alters the size in response to network conditions to make better use of communication resources.

TABLE II
SIMULATION PARAMETERS

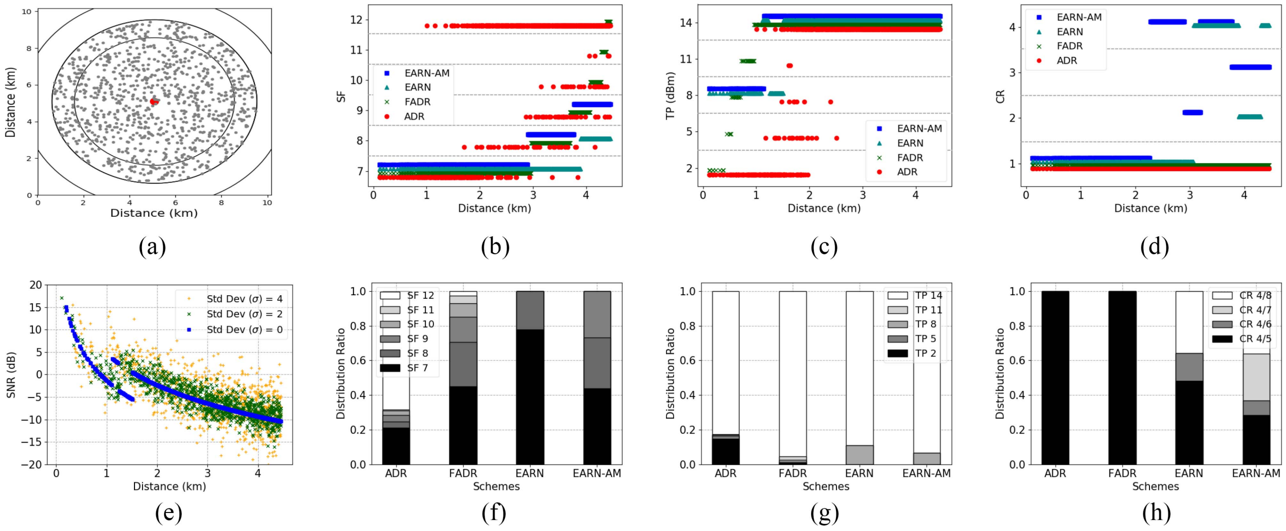
Parameters	***Values***
Carrier frequency (CF)	922.0 MHz
Bandwidth (BW)	125 kHz
Spreading factor (SF)	7, 8, 9, 10, 11, 12*
Transmission power (TP)	2, 5, 8, 11, 14* dBm
Coding rate (CR)	$4/5^*$, 4/6, 4/7, 4/8
Number of nodes (N)	1,000
PHY payload (PL)	12 bytes
Traffic load (λ)	0.28, 0.56, 0.83* $\cdots$ 2.22
Cell radius (R)	3.5, 4.5*, 5.7, 7.3, 9.4, 12 km
Path loss Std Dev (σ)	0, 1, 2, 3*, 4

V. EVALUATIONS

In this section, we compare the performance of *EARN* with legacy ADR and FADR [6]. FADR is a well-established recent ADR proposal whose mechanism is described in our preliminaries. In all schemes, the ED-side ADR mechanism operates as before, but FADR and *EARN* replace the server-side ADR operation with their own. We evaluated the performances in terms of goodput, EPP, and fairness under three different scenarios, each of which varies the noise level (i.e., path-loss standard deviation), cell radius, and traffic load. Each scenario averages the results of 30 episodes of 24-h long simulation whose network topology is regenerated every time.

A. Simulation Setup

We implemented a large-scale LoRaWAN simulation testbed using Simpy [25], a discrete event simulator on Python. The simulation environment imitates the real-world behavior of LoRaWAN as much as it can. All factors that can possibly bring about a collision, such as co/inter-SF interference, the capture effect, and overlap durations of all up/downlink frames, as well as transmission parameters, are taken into account. Table II shows the parameter settings we used on the experiments, where the values with asterisks(*) are set by default; 1000 EDs transmitting 12 B of payload once every 20 min ($\lambda = 0.83$) are randomly scattered within the range of 4.5 km around a GW. We exploited the log-distance path-loss model with the following parameters as in [15]: 128.95 dB as the mean path loss of 1-km distance, 2.32 as the path-loss exponent, and 3.0 as the standard deviation of the path loss. Although the maximum transmission range of an LoRaWAN can be as far as 12 km with SF 12 in our configuration that reflects a suburban environment, we limit the cell radius to 4.5 km by default, which is the communication range of an ED with SF 8, for the fair comparison with conventional schemes. FADR requires EDs to be placed close enough to a GW so that the fixed SF distribution ratio can be kept. In LoRaWAN's basic operation, multiple channels and BWs can be employed. However, for the sake of simplicity, only 125 kHz wide single-channel communication is tested, as the use of *EARN* and other ADR schemes can be easily extended to multichannel and multiband scenarios.



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 4. Comparison on different parameter distribution behaviors of ADR schemes: ADR, FADR, *EARN*, and *EARN-AM*. (a) GW and ED deployment. (b) SF distribution. (c) TP distribution. (d) CR distribution. (e) SNR at the GW with and without fluctuation in *EARN-AM*. (f) SF distribution ratio. (g) TP distribution ratio. (h) CR distribution ratio.

B. Parameter Distribution

We first observe the different behaviors of ADR schemes in terms of SF, TP, and CR distribution. Fig. 4(a) shows an example of a simulation environment with 1000 EDs randomly placed within a range of 4.5 km around a GW. The circled borderlines present the maximum communication ranges defined by SNR_{req} for each SF. For different ADR schemes, Fig. 4(b)–(d) shows the distributions of SF, TP, and CR, where each point corresponds to an ED, and Fig. 4(b)–(d) shows the distribution ratios of SF, TP, and CR.

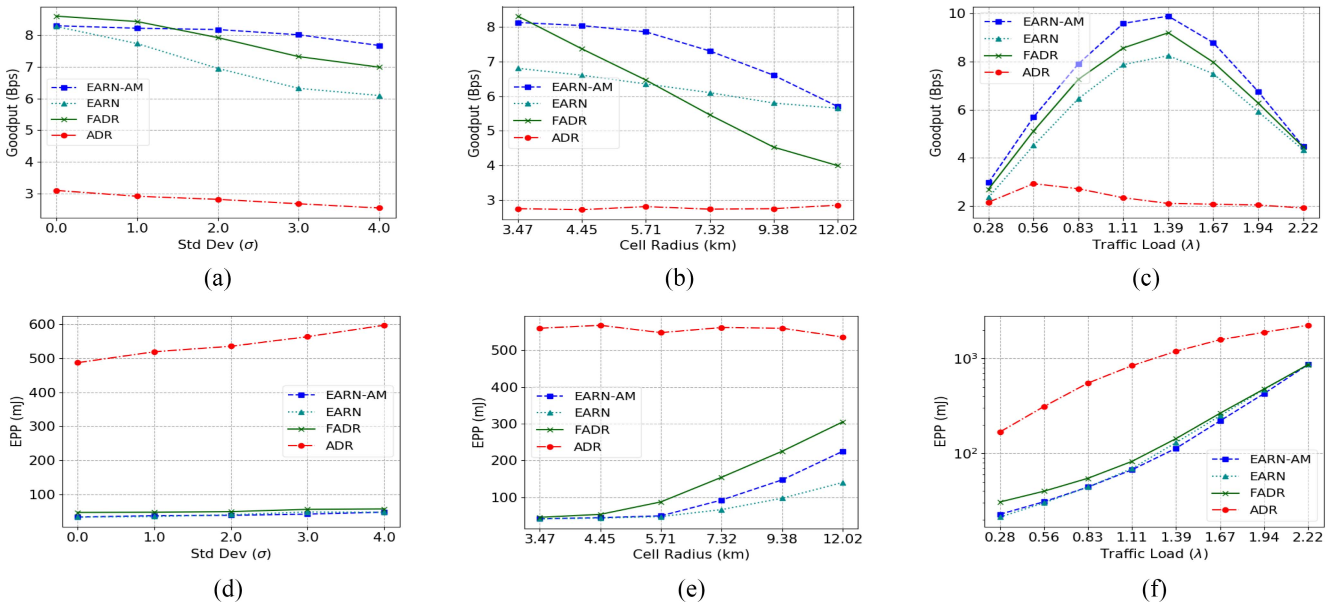
Generally, the farther away an ED is from the GW, the higher SF and TP are adopted. ADR estimates the link budget with a generous margin of 10 dB to SNR_{req}, and therefore, allocates especially higher SF and TP than those of which an actual communication requires. FADR allocates SF to EDs based on fixed ratios with the purpose of equally balancing the packet airtime or collision probability among different SFs; the ratio of each SF from 7 to 12 is {0.449, 0.257, 0.144, 0.08, 0.044, 0.024}. EDs out of those ratios are eventually allotted SF 12, aggravating the load in the lowest data rate. FADR can work intact on LoRaWAN with the cell radius of up to 3.5 km, but they may not function properly on larger cells depending on the ED deployments. The SF distribution of *EARN* and *EARN-AM*, on the other hand, is not tied to any ratio and consists only of SF 7, 8, and 9. Since the cell is small and the default network load can be sufficiently accommodated with these high data rates, SF 10, 11, and 12, which consume unnecessarily much energy, are not employed at all. The rate of allocation of SF 7 in *EARN*, without taking path-loss deviation into account, is significantly higher than in *EARN-AM*.

Unlike other methods that sequentially assign higher TPs based on the distance of an ED to a GW, *EARN* and *EARN-AM* assigns TP of 8 dBm to EDs near the GW and TP of 14 dBm to the others. This is to evenly distribute the collision probability on the SNR domain among EDs with the same SF. If TPs of some EDs, no matter how close they are to the GW, are equally configured to SNR_{req}, their messages will all be lost upon collision due to the capture effect. The power chosen by *EARN*, 8 dBm, is neither too weak to lose in competition nor too strong to waste energy as in Fig. 4(a).

Moreover, in Fig. 4(b), it is important to note that the wider communication ranges to each SF by *EARN* and *EARN-AM* is the benefits from the suitable CR values. Our methods, unlike the other ADR methods, is capable of reducing the error rates of EDs with lower CRs, especially near the SNR_{req}. The lower CR constructs a slightly larger frame, but it obtains an improved tolerance to the path loss and noise. If we observe the *EARN*'s CR allocation in Fig. 4(d), EDs close to a GW put a priority on efficiency and exploit the highest CR of 4/5, and EDs far from the GW start to adopt a lower CR to prioritize stability. The CRs allocated by distance are sequentially 4/5, 4/8, 4/6, 4/8, 4/7, and 4/8. The same pattern is exhibited in all subsequent experiments. The CR adaptation of *EARN-AM* follows this pattern with a distance imposed by the adaptive margin.

C. Noise Level

The path loss in a signal can be varied due to factors, such as shadowing and noise. We analyze the impact of the variation in path loss on performance in Fig. 5(a) and (d), to justify *EARN-AM*, the extended version of *EARN*. In Fig. 5(a), the goodput of all ADR schemes decreases as the deviation increases from 0 to 4. This is because the transmission parameters were assigned to satisfy the link budget estimated by the SNR measurements. If the actual SNR an uplink is lower than the predicted due to the fluctuations, the message will likely be lost. As mentioned earlier, ADR employs a generous SNR margin of 10 dB to respond to such deviation. Although it results in the worst goodput and EPP, no further performance degradation will occur unless the SNR variation is greater than 10 dB. FADR shows the highest goodput in the smallest



(a) (b) (c)

(d) (e) (f)

Fig. 5. Link performance comparisons of different ADR schemes by varying path loss std dev., cell radius, and traffic load (λ). (a) Goodput versus std dev. (b) Goodput versus cell radius. (c) Goodput versus traffic load (λ). (d) EPP versus std dev. (e) EPP versus cell radius. (f) EPP versus traffic load (λ).

cell, as collision probabilities are evenly distributed to different SFs. Still, there is also a fast performance degradation as the deviation of the path loss becomes more substantial. *EARN*, without noise, achieves great baseline performance in terms of goodput but is susceptible to the link variation. This is because the *EARN* makes the best use of the estimated link budget by tightly adjusting the SF, TP, and CR to fine tune the performance. The parameters become impractical when the actual link condition does not match the prediction, and the penalty in the performance is the largest among the ADR schemes. Nonetheless, it provides the best energy efficiency with *EARN-AM*. *EARN-AM*, in particular, uses statistics on channel conditions to respond to path-loss deviations. As shown in Fig. 5(a), *EARN-AM* performs as good as *EARN* when the standard deviation is zero. The employment of the *adaptive margin* has notably eased the performance penalty by high path-loss deviation, and EPP of *EARN-AM* is also the lowest of all.

D. Cell Radius

In Fig. 5(b) and (e), we observe the performance change due to the cell radius, the maximum distance from the GW where an ED can be placed. The tested distances are the maximum communication ranges of each SF based on $\mathrm{SNR}_{\mathrm{req}}$ with default parameter settings; respectively, 3.5, 4.5, 5.7, 7.3, 9.4, and 12 km. In the figures, ADR seems to be resistant to distance changes. However, the performances are so bad that ADR stands far behind other methods in terms of both efficiency and scalability. The conservative strategy of ADR always assigns high SF and TP to most EDs except for the few adjacent to the GW, easily saturating the medium in low data rates. FADR shows high goodput and low EPP in narrow cells, but performance degrades rapidly as the cell radius broadens. The fixed SF distribution ratio of FADR, which evenly portion out the collision probability to SF, can be complied with only in small cells. The goodput of *EARN-AM* in the smallest cell is slightly lower than that of FADR of about 2.2%, but the performance reduction over radius change is modest. The goodput of *EARN* constantly suffers from the path-loss deviation. In the meanwhile, it suffices for its objective function and achieves the best EPP in all cases. It is not tied to a certain SF distribution ratio, but rather it immediately assigns parameters that are most suitable for an ED in a given network condition. Besides, the stability achieved by CR adaptation significantly enhances the capacity of all SFs by lowering $\mathrm{SNR}_{\mathrm{req}}$. FADR, in particular, is imperfect in its design in that it only assumes the narrow range cells where all SFs are available. They do not properly reflect the low-power and long-range characteristics of LoRaWAN. *EARN-AM* assigns link-adaptive tight SF, TP, and CR, exhibiting a reliable performance over a wide range of environments and application scenarios.

E. Traffic Load

We observe performance changes by varying traffic loads in Fig. 5(c) and (f), as the traffic capacity of a network is a metric that is closely related to the scalability. The evaluation assumes general application scenarios of LoRaWAN and starts from the initial load of about arrival rate (λ) 0.28, which is the load of 1000 EDs transmitting messages once an hour. Then, the traffic is gradually increased by shortening the message period to 30, 20, 15 min, and so on. The goodput of all ADR methods in Fig. 5(c) shows a pattern similar to that of the famous ALOHA performance function; goodput increases to a certain level and then begins to decrease as the collision rate grows exponentially. The legacy ADR is the first to reach its maximum goodput near $\lambda = 0.56$ and enters the decreasing phase. This is followed by *EARN*, FADR, and *EARN-AM* near $\lambda = 1.39$. *EARN-AM* is ahead of other methods regardless of the load degree, showing about 7.5% better performance than

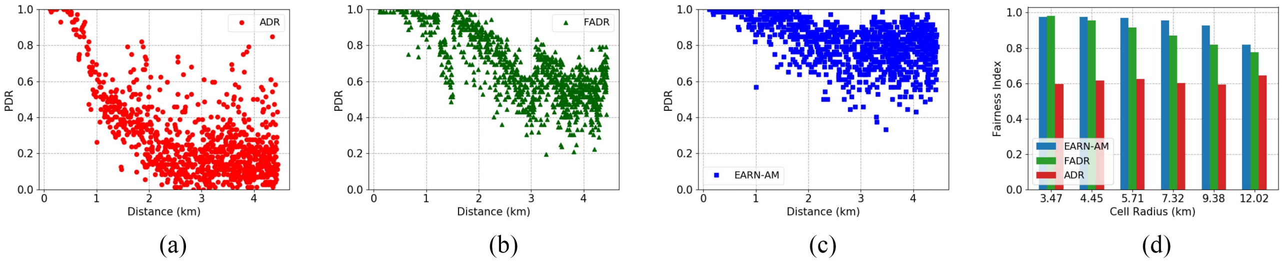


Fig. 6. PDR distributions of (a) ADR, (b) FADR, and (c) *EARN-AM*. *EARN-AM* ensures the overall high PDR to EDs. The fairness comparison with Jain's fairness index in (d) also confirms that *EARN-AM*, regardless of the cell radius, outperforms the others.

FADR, the second-best ADR method, at their maximum goodput. We can refer that *EARN-AM* makes the best use of channel capacity to sustain the highest load with the greatest stability. However, we found that collisions between uplink frames are not the sole factor that degrades the performance. When we tracked down the cause of frame losses, the processing capacity of a GW acted as a bottleneck under heavy traffic loads. An LoRaWAN GW, which can simultaneously decode up to eight frames, could not send downlink control messages, which contains the new parameter settings from the server, back to EDs when there were too many frames arriving. EDs, failing to receive any downlinks from the GW, gradually lowered their data rates and intensified the performance degradation.

In the meantime, *EARN* and *EARN-AM* maintained the lowest EPP under any loads as shown in Fig. 5(f), indicating that the link prediction and subsequent parameter assignment were successful. The adoption of different CRs lowered SNR_{req} and increased the number of EDs that an SF can sustain, which in turn resulted in better P_{ne} and P_{nc}. If our models additionally take into account the processing power of a GW and the impact of downlink messages, or if we find an effective way to secure the downlink control messages, further improvements are expected in link performance prediction and parameter assignment.

F. Fairness

Finally, we compare the fairness of different ADR schemes. The fairness that we use in our evaluation is the fairness in results (i.e., PDR), as the transmission opportunities of EDs are already properly controlled by the existing duty cycle limit of LoRaWAN. Fig. 6(a)–(c) shows the PDRs of EDs when ADR, FADR, and *EARN-AM* are applied, respectively. In the figures, each point corresponds to an ED and demonstrates the distance from a GW. In the legacy ADR, most EDs that are not close enough to the GW are easily congested with one another and exhibit low PDR overall with wide variations. FADR, with an improved SF and TP allocation strategy, shows a better PDR distribution compared to ADR. However, it does not adapt CR and therefore is susceptible to frame error induced by path loss. The CR impact is evident where an ED is about 3 km away from the GW near SNR_{req}, and in Fig. 6(b), we observe the huge gaps between PDRs of EDs placed at the transmission range boundaries limited by each SF. Furthermore, the low PDRs from EDs 1.5 km away from the GW is due to the bad TP allocation practice. They are in the middle of near and far EDs in SF 7, and their frames are lost in collisions. *EARN-AM*, on the other hand, fine tunes the link performance of EDs by adapting CR as well as SF and TP. As a result, the decrease in PDR with distance is moderate, and the deviation is also not significant as in Fig. 6(c).

We also evaluated the fairness of each method with Jain's fairness index based on PDR as in [6], and the formula is as follows:

$$\zeta = \frac{\left(\sum_{i=1}^{N} \text{PDR}_i\right)^2}{N \sum_{i=1}^{N} \text{PDR}_i^2}.$$

Here, N is the number of EDs located within an LoRaWAN cell, and PDR_i denotes the PDR of an ED i. The fairness index has a value ranging from 0 and 1, where the higher value indicates the higher PDR fairness between the EDs. Fig. 6(d) shows the fairness index of each method under different cell sizes. The result is truly meaningful in that *EARN-AM*, which optimized EPP, outperforms FADR that prioritized fairness. FADR shows good fairness between most EDs in the smallest cell. However, as the cell size increases, the fixed SF distribution ratios cannot be complied with and thus, the fairness is lost. In *EARN-AM*, all EDs independently obtain high PDR by estimating the link condition and searching for the best parameter set. This, in turn, ensures the overall high PDR throughout the network, even to the EDs far away from the GW. The fairness of *EARN-AM* is similar to FADR in the smallest cell and surpasses the others in the larger cell scenarios.

VI. Conclusion

In this article, we formulated LoRaWAN link performance models that reflect the complex correlation between transmission parameters, emphasizing the need for the adaptive employment of CR in the ADR mechanism. Then, we proposed an enhanced ADR called *EARN*, which takes into account CR and the capture effect, to fine tune the link performance of EDs. In the end-to-end LoRaWAN simulator we developed, *EARN* outperformed the conventional schemes in terms of goodput, EPP, and fairness, refining LoRaWAN as a scalable and practical IoT solution.

In the future, we would like to further sophisticate our models to reflect the processing power of a gateway, the impact of downlink messages, and other unknowns. In this regard, we plan on applying deep learning techniques to design an ADR that learns the better use of transmission parameters from rich empirical and theoretical LoRaWAN data. We also envision

ED-side ADR improvements to address the case where an ED should autonomously adjust its parameters as the downlink control messages from the server are lost.

REFERENCES

[1] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "Do LoRa low-power wide-area networks scale?" in *Proc. 19th ACM Int. Conf. Model. Anal. Simulat. Wireless Mobile Syst.*, 2016, pp. 59–67.

[2] O. Georgiou and U. Raza, "Low power wide area network analysis: Can LoRa scale?" *IEEE Wireless Commun. Lett.*, vol. 6, no. 2, pp. 162–165, Apr. 2017.

[3] B. Ghena, J. Adkins, L. Shangguan, K. Jamieson, P. Levis, and P. Dutta, "Challenge: Unlicensed LPWANs are not yet the path to ubiquitous connectivity," in *Proc. 25th Annu. Int. Conf. Mobile Comput. Netw.*, 2019, pp. 1–12.

[4] B. Reynders, W. Meert, and S. Pollin, "Power and spreading factor control in low power wide area networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2017, pp. 1–6.

[5] F. Cuomo, M. Campo, A. Caponi, G. Bianchi, G. Rossini, and P. Pisani, "ExpLoRA: Extending the performance of LoRa by suitable spreading factor allocations," in *Proc. IEEE 13th Int. Conf. Wireless Mobile Comput. Netw. Commun. (WiMob)*, 2017, pp. 1–8.

[6] K. Q. Abdelfadeel, V. Cionca, and D. Pesch, "Fair adaptive data rate allocation and power control in LoRaWAN," in *Proc. IEEE 19th Int. Symp. World Wireless Mobile Multimedia Netw. (WoWMoM)*, 2018, pp. 14–15.

[7] "A technical overview of LoRa and LoRaWAN," LoRa Alliance, San Jose, CA, USA, White Paper, 2015.

[8] Semtech. (2015). *SX1272/73—860 MHz to 1020 MHz Low Power Long Range Transceiver*. [Online]. Available: https://www.semtech.com/uploads/documents/SX1272_DS_V4.pdf

[9] *LoRaWAN^{TM} 1.1 Regional Parameters*, LoRa Alliance, San Jose, CA, USA, 2017.

[10] The Things Network. (2017). *The Thing Network Wiki: Adaptive Data Rate*. [Online]. Available: https://www.thethingsnetwork.org/wiki/LoRaWAN/ADR

[11] *LoRaWAN^{TM} 1.1 Specification*, LoRa Alliance, San Jose, CA, USA, 2017.

[12] A. Hoeller, R. D. Souza, O. L. A. López, H. Alves, M. de Noronha Neto, and G. Brante, "Exploiting time diversity of LoRa networks through optimum message replication," in *Proc. IEEE 15th Int. Symp. Wireless Commun. Syst. (ISWCS)*, 2018, pp. 1–5.

[13] J.-T. Lim and Y. Han, "Spreading factor allocation for massive connectivity in LoRa systems," *IEEE Commun. Lett.*, vol. 22, no. 4, pp. 800–803, Apr. 2018.

[14] M. Bor and U. Roedig, "LoRa transmission parameter selection," in *Proc. 13th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, 2017, pp. 27–34.

[15] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo, "On the coverage of LPWANs: Range evaluation and channel attenuation model for LoRa technology," in *Proc. IEEE 14th Int. Conf. ITS Telecommun. (ITST)*, 2015, pp. 55–59.

[16] M. Cattani, C. A. Boano, and K. Römer, "An experimental evaluation of the reliability of LoRa long-range low-power wireless communication," *J. Sensor Actuator Netw.*, vol. 6, no. 2, p. 7, 2017.

[17] R. Eletreby, D. Zhang, S. Kumar, and O. Yağan, "Empowering low-power wide area networks in urban settings," in *Proc. Conf. ACM Special Interest Group Data Commun.*, 2017, pp. 309–321.

[18] A. Dongare *et al.*, "Charm: Exploiting geographical diversity through coherent combining in low-power wide-area networks," in *Proc. 17th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, 2018, pp. 60–71.

[19] V. Talla, M. Hessar, B. Kellogg, A. Najafi, J. R. Smith, and S. Gollakota, "LoRa backscatter: Enabling the vision of ubiquitous connectivity," *Proc. ACM Interact. Mobile Wearable Ubiquitous Technol.*, vol. 1, no. 3, pp. 1–24, 2017.

[20] A. Varshney, O. Harms, C. Pérez-Penichet, C. Rohner, F. Hermans, and T. Voigt, "LoRea: A backscatter architecture that achieves a long communication range," in *Proc. 15th ACM Conf. Embedded Netw. Sensor Syst.*, 2017, pp. 1–14.

[21] Y. Peng *et al.*, "PLoRa: A passive long-range data network from ambient LoRa transmissions," in *Proc. Conf. ACM Special Interest Group Data Commun.*, 2018, pp. 147–160.

[22] T. Polonelli, D. Brunelli, and L. Benini, "Slotted aloha overlay on LoRaWAN—A distributed synchronization approach," in *Proc. IEEE 16th Int. Conf. Embedded Ubiquitous Comput. (EUC)*, 2018, pp. 129–132.

[23] T.-H. To and A. Duda, "Simulation of LoRa in NS-3: Improving LoRa performance with CSMA," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2018, pp. 1–7.

[24] B. Reynders, W. Meert, and S. Pollin, "Range and coexistence analysis of long range unlicensed communication," in *Proc. IEEE 23rd Int. Conf. Telecommun. (ICT)*, 2016, pp. 1–6.

[25] Simpy. (2016). *Event Discrete Simulation for Python*. [Online]. Available: https://simpy.readthedocs.io

Junhyun Park received the B.S. degree in computer science from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 2013. He is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Seoul National University, Seoul, South Korea.

His current research interests include wireless networks, specifically IEEE 802.11 WLAN and Internet of Things, and social networks security.

Kunho Park received the B.S. degree in information and communication engineering from Sejong University, Seoul, South Korea, in 2018. He is currently pursuing the M.S. degree with the School of Computer Science and Engineering, Seoul National University, Seoul.

His current research interests include wireless network, sensor network, and Internet of Things.

Hyeongho Bae received the B.S. degree in computer science and engineering from the University of Inha, Incheon, South Korea, in 2019. He is currently pursuing the M.S. degree with the School of Computer Science and Engineering, Seoul National University, Seoul, South Korea.

His current research interests include wireless networks, video streaming, and localization and network optimization using DRL.

Chong-Kwon Kim received the B.S. degree in industrial engineering from Seoul National University, Seoul, South Korea, in 1981, the M.S. degree in operations research from the Georgia Institute of Technology, Atlanta, GA, USA, in 1982, and the Ph.D. degree in computer science from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 1987.

In 1987, he joined Bellcore, Piscataway, NJ, USA, as a Technical Staff Member, where he was involved with broadband ISDN and ATM. Since 1991, he has been with Seoul National University, as a Professor with the School of Computer Science and Engineering. His current research interests include wireless and mobile networking, social network analysis, SNS security, and performance evaluation.