# ALOHA with SIC-Aided Collision Resolution

Jun-Bae Seo, *Member, IEEE*, Yangqian Hu, Hu Jin, *Senior Member, IEEE*, and Swades De, *Senior Member*, *IEEE*

***Abstract*—ALOHA can be a viable solution as a light-weight medium access control (MAC) protocol in low power wide area networks (LPWAN) for Internet-of-Things (IoT). However, the maximum throughput of traditional ALOHA is too low to accommodate a large number of IoT devices. To address this limitation, this work proposes an enhanced ALOHA, where successive interference cancellation (SIC) aids in collision resolution. In the proposed system, each user measures the time interval from their transmission epoch to the end of a collision using a collision timer. Upon a collision, the access point (AP) with SIC and users' collision timer work jointly to resolve the collision. This work characterizes the throughput of the proposed system and further proposes an online backoff algorithm to maximize the throughput. Numerical results demonstrate that the proposed ALOHA with SIC-aided collision resolution (SACR) can offer significantly improved throughput compared to S-ALOHA and the other systems.**

***Index Terms*—IoT MAC protocol, successive interference cancellation (SIC), SIC-aided ALOHA, online backoff control, throughput maximization, collision timer.**

## I. INTRODUCTION

INTERNET-OF-THINGS (IoT) is an essential enabler for realizing smart cities, including smart grid, smart roads, smart lighting, smart parking, etc. IoT devices collect random event data and send it to remote servers to facilitate data-driven decision-making. As IoT applications grow enormously and expand rapidly, massive IoT devices are being deployed, leading to a tremendous increase in the volume of random access (RA) data. To support the massive connectivity required by IoT devices, increasing the throughput of RA systems becomes essential. Depending on the IoT applications, various wireless networks such as Bluetooth, Wi-Fi, and wireless cellular systems are employed to provide wireless connectivity. Among them, low power wide area networks (LPWANs) [1] have drawn recent research attention due to their wide coverage area of up to 15 km in rural settings, low power consumption, and low cost based on unlicensed bands. With battery-powered IoT devices for up to 10 years, it is thus most suitable for diverse industry applications such as smart agriculture, wildlife monitoring and tracking, critical infrastructure monitoring, and logistics sectors. LPWANs encompass various systems such as narrow-band IoT (NB-IoT) from Long-Term Evolution (LTE) with the licensed band, Long Range (LoRa), and Sigfox with the unlicensed band. NB-IoT adopts slotted ALOHA (S-ALOHA), whereas LoRa and Sigfox adopt (pure) ALOHA as the medium access control (MAC) protocol.

In slotted ALOHA (S-ALOHA), time is divided into fixed-size slots, each corresponding to one packet transmission. Devices transmit their packets within these slots, resulting in a *vulnerable* period of one slot [2], equivalent to one packet transmission time. In contrast, ALOHA operates asynchronously, allowing IoT devices to transmit at will without requiring high-precision synchronization hardware and algorithms to align with slot boundaries, regardless of their location or timing. This enables IoT devices to be small, inexpensive, and energy-efficient compared to systems that require high synchronization for global timing. However, the tradeoff is that the vulnerable period in ALOHA is twice the packet size. Consequently, for large populations, the maximum throughput of ALOHA is limited to 0.183 (packets per packet transmission time), which is half that of S-ALOHA. This low throughput restricts the allowable population size or coverage area. To support more IoT devices, it is crucial to enhance the access throughput of ALOHA while preserving its asynchronous nature. If the throughput of ALOHA can be improved at a reasonable cost, it would offer a highly viable and affordable solution in terms of achievable data rate per unit cost and the number of IoT devices per unit coverage area.

To overcome the throughput limit of S-ALOHA, a multi-packet reception (MPR) channel has been proposed [3], [4]. From a MAC layer perspective, the MPR channel can be seen as a mathematical abstraction of various $k$-out-of-$n$ RA channels, where $n$ packets are transmitted during a collision, and $k$ packets are successfully decoded, with $k \leq n$. On the physical layer, MPR channels can be realized using advanced signal processing technologies such as code-division multiple access (CDMA), successive interference cancellation (SIC), multi-input multi-output (MIMO) antenna systems, joint-detection (JD) among others [4], [5]. Materializing an MPR channel involves a cross-layer approach, where physical layer technology is integrated with a specific MAC protocol. Among various realizations of MPR channels, SIC has been widely applied to S-ALOHA [6]–[27], as have tree or splitting algorithms [28]–[36]. SIC enables the decoding of multiple packets from a collision within a single slot or across multiple collisions in different slots, by subtracting a packet successfully decoded from the other colliding packets transmitted together with the packet. This capability, absent in traditional S-ALOHA, led to a significant increase in throughput. Therefore, employing SIC in ALOHA is expected to overcome the throughput limit of 0.183. However, implementing SIC in ALOHA has been relatively underexplored [37]–[47], because colliding packets are not aligned within slots and may overlap across multiple packets, so the implementation of SIC becomes more challenging. Interestingly, while the slotted structure of S-ALOHA

J.-B. Seo is with the Department of Information and Communication Engineering, Gyeongsang National University, Tongyeong 53064, South Korea, (e-mail: jbseo@gnu.ac.kr).

Y. Hu and H. Jin are with the Department of Electrical and Electronic Engineering, Hanyang University, Ansan 15588, South Korea (e-mail: yijie86@hanyang.ac.kr, hjin@hanyang.ac.kr).

S. De is with the Department of Electrical Engineering and Bharti School of Telecommunication, IIT Delhi, New Delhi, India (e-mail: swadesd@ee.iitd.ac.in)

increases the cost and complexity compared to ALOHA, it also facilitates the implementation of MPR channels. This is because aligning colliding packets within slots simplifies the process of implementing SIC.

A common feature of previous approaches using SIC for S-ALOHA, ALOHA, and tree algorithms [6]–[47] is the direct application of SIC at the physical layer to decode packets involved in a collision. In contrast, this work proposes ALOHA with SIC-aided collision resolution (SACR), leveraging SIC to detect the transmission epoch of packets within collisions as long as the transmission epochs are at least greater than some time offset, say $\Delta$ ($\mu$sec) interval. This detection capability, combined with the users' collision timer, assists in determining which packet should be retransmitted next, thus aiding the collision resolution process at the MAC layer that starts immediately after the collision. The novelty of the proposed scheme lies in its ability to link the detection of transmission epochs in a collision at the physical layer (using SIC) to a sequence of retransmissions after the collision at the MAC layer, thereby improving throughput. Thus, with a suitable physical layer technology that reliably detects the transmission epochs of packets in a collision, our proposed scheme can be effectively applied to enhance system performance.

### A. Related work

In the extensive research focused on developing MPR channels with SIC, the majority of studies are based on S-ALOHA and tree algorithms, i.e., slotted RA systems. This section introduces the fundamental concepts of how SIC has been applied to slotted RA systems. As our proposed system is rooted in (unslotted) ALOHA, we provide a more detailed exploration of how SIC is utilized in ALOHA.

S-ALOHA systems (or tree algorithm) utilizing SIC can be divided into two cross-layer approaches: cross-slot and in-slot SICs. The cross-slot SIC for S-ALOHA is commonly referred to as irregular repetition S-ALOHA (IRSA) or coded S-ALOHA (CSA). The original scheme, known as contention resolution diversity S-ALOHA (CRDSA) [6], assumes a frame consisting of multiple slots. Devices randomly select two slots in a frame and transmit two copies of the same packet. If one copy of a packet is transmitted in a slot without any collisions, it is successfully decoded. The access point (AP) then leverages this successfully decoded copy to decode other packets that were transmitted in slots, where collisions occurred with the same packet copy.

Unlike CRDSA, where devices transmit only two copies, IRSA allows devices to transmit a variable number of copies [7] and optimizes the distribution of these copies using the bipartite graph technique, which is also used for forward error-correcting codes. Note that transmitting a variable or fixed number of copies of a packet has been proposed for a single-channel ALOHA as diversity ALOHA in [48] and multichannel ALOHA in [49]. CRDSA and IRA advance the concept of diversity ALOHA with the addition of SIC. Both approaches [6], [7] initially assumed a fixed length of a frame length. However, this assumption is relaxed in later studies [8], [9], allowing the length of the fame to be dynamically determined, e.g., based on the population size.

IRSA has been further generalized to CSA in [10], where a data packet is divided into $k$ segments, each of which is encoded by a segment-oriented linear block code. The user has $n$ encoded segments and transmits them instead of the same copies of the packet. The number of the resulting segments can be different since each user can choose one of the predefined codes. Thus, IRSA is a special case of CSA akin to a repetition code. CRDSA has continued to evolve as some aspects of the original scheme have been altered for higher performance. For instance, while the original CRDSA assumed equal transmit power for all copies, transmit power diversity is later considered in [11], where each copy can be transmitted with a different level of transmit power. The impact of imperfect SIC on CSA and IRSA has been examined in [12], [13], whereas it has been also studied the performance of CSA when some segments in CSA are lost in [14], [15]. S-ALOHA systems employing CRDSA, IRSA, and CSA need to store all packets transmitted in a frame to execute cross-slot SIC, whereas each copy of a packet should contain the information pointing to the locations of other copies. Furthermore, transmitting extra copies of a packet in advance may increase power consumption compared to retransmissions made only when necessary.

In contrast with the earlier S-ALOHA based on cross-slot SIC, non-orthogonal multiple access (NOMA) represents S-ALOHA with in-slot SIC, where packets in a collision are used for decoding others within the same slot. In particular, NOMA in [16]–[18] assumes that devices control their transmit power through channel inversion so that the received power of the packet can be one of several predefined target values. The AP then first decodes the packet with the highest target value by treating the other packets as noise. However, if more than one packet with the same target value is transmitted in a slot, none of those packets can be decoded, and additionally, packets with lower target values will also fail to be decoded. NOMA with a constant transmit power is also explored in [19]–[21], while an approach that gradually increases transmit power upon a collision is examined [22], [23]. Recently, some studies in [24]–[26] combine IRSA and NOMA.

Compared to the 0.368 throughput barrier of S-ALOHA, tree algorithms in [2] can achieve as high as 0.48 by allowing the colliding packets to be retransmitted successively with a coin-tossing rule. Both cross-slot and in-slot SICs have been integrated into the tree algorithm in [28]–[36] to get a higher throughput than 0.48 with the help of SIC. Tree algorithms with cross-slot SIC store the colliding packets as retransmissions by coin-tossing proceed over time [28], [32]. If some of them can be successfully retransmitted, they can be used for decoding the packets previously stored.

Turning to ALOHA, an MPR channel is realized by combining spread spectrum technology with ALOHA, resulting in what is known as spread spectrum ALOHA (SSA) [50]–[52]. In SSA, some packets involved in a collision can be decoded if their signal-to-interference-plus-noise ratio (SINR) exceeds a certain threshold. Building on this, an enhanced spread spectrum Aloha (E-SSA) has been proposed in [37], where the packet with the highest SINR is decoded and iteratively used for SIC to decode more packets in the collision. A comparative study for CRDSA, IRA, CSA, and E-SSA was conducted in [27].

CRDSA invented for slotted systems has been extended to unslotted systems as well [38]. In this variation, each device operates with its own frame structure, which is known as a virtual frame (VF) consisting of multiple slots. The VFs of each device, however, are not globally aligned at the AP, which results in an asynchronous system. This asynchronous CRDSA (ACRDSA) can be regarded as ALOHA with cross-slot SIC. Irregular repetition ALOHA (IRA) has been proposed for ALOHA [39], [40], allowing each user to transmit multiple replicas of the same packet uniformly within a frame duration of each user as in IRSA, while the header of each copy includes information on the locations of other copies within the frame. While SIC is primarily employed to decode some packets from those in a collision, some parts of the copies after SIC can be concatenated [39] or some combining techniques in the physical layer used in [40] to reconstruct the whole packet. ACRDSA is also integrated into a frequency-hopping SS in [41], where multiple copies of a packet are transmitted at different times and frequencies.

Similar to [50]–[52], interference-limited MPR channels, where a packet transmission is successful if the number of interfering packets in a collision is below a certain threshold, have been considered for ALOHA in [42], [43]. Specifically, a constant packet size was considered in [42] and a variable packet size in [43]. Although these studies provide performance models for the throughput of ALOHA with such MPR channels, the MPR channel they propose is a loose mathematical abstraction. It assumes that successful decoding based on the SINR threshold should depend on the number of interfering packets. This assumption poses challenges, particularly in extracting the beginning and end of packets when they are not perfectly aligned, as ALOHA involves mixed packets in the midst of collisions. The concept of packet capture, where a packet in a collision is successfully decoded if the SINR of a packet exceeds a threshold, has been studied for ALOHA in [53], and S-ALOHA without and with SIC in [54]–[56].

On the other hand, Wang and Fapojuwo in [44] proposed a specific method for an AP to decode packets using SIC in an ALOHA. In their approach, the AP employs a receive window to capture incoming packets for a specific period of time. The duration of this window is at least twice the packet transmission time, matching the vulnerable period for ALOHA. These receive windows are designed to partially overlap, allowing packets captured in one window to be leveraged in the subsequent window. SIC is then iteratively performed in each window up to the maximum number of iterations. It can achieve a maximum throughput of up to 0.45 (packets/packet transmission time). In fact, this method bears a strong similarity with ACRDSA [38].

Leveraging the correlation property of the chirp spread spectrum technique adopted by LoRa, [45]–[47] realized an MPR channel in ALOHA-based LoRa. More specifically, they identify and separate the packets in a collision when the transmission epochs of the packets can be at least greater than some time offset by detecting the correlation peak of a preamble, which is part of a packet.

The similarities and differences between our work and previous research can be summarized as follows. First, direct packet decoding is performed by identifying packets through preamble detection in a collision in [45]–[47], provided that the transmission epochs of the packets are separated by a time offset $\Delta$. Similarly, our work also relies on the assumption that the transmission epochs should be $\Delta$ apart for correct detection; however, instead of directly decoding the packets with SIC, we utilize it to detect the transmission epochs of the packets. This detection is then used to aid in the contention resolution process following the collision. While our current focus is on using SIC, our proposed scheme could be adapted to incorporate other physical layer technologies that serve the same function. As exploring the application of alternative physical layer technologies in place of SIC is beyond our scope, we leave it as a future work.

Second, our proposed system bears a similarity with tree algorithms in the sense that resolving the collision of initial $n$ users is focused on, while the transmissions from the users not involved with the collision are not allowed. In most of the tree algorithms, the sequence of retransmissions is determined by a coin-tossing except Gallager's first-come-first-serve (FCFS) algorithm [2]. In our proposed system, however, the users' collision timer, with the help of SIC, assists in determining the sequence of retransmissions. Note that in [57], we developed a collision resolution method without SIC to identify the first and last packets transmitted in a collision. Each user employs a collision timer and starts to measure the interval from their transmission epoch to the end of the collision. Upon a collision, if the AP announces the entire collision period and the one packet transmission time sequentially, the users who transmitted the first and last packets in the collision can match this information with their collision timer. They then retransmit sequentially immediately after the collision. Since two packets can be successfully retransmitted after a collision, this collision resolution method in the MAC layer brought significant throughput gain. We extend this method with SIC to allow the retransmission of packets in the middle of the collision, not just the first and last packets, under a certain condition.

Third, tree algorithms often adopt some estimation methods for batch size upon a collision to resolve it as fast as it can, or minimize future collisions [32], [34]–[36]. Similarly, in the proposed ALOHA-SACR, we introduce an online backoff algorithm aimed at maximizing the throughput, which is based on estimating the backlog size.

### *B. Contributions*

Compared to the previous works [39]–[47], our main contribution is the proposal of a novel ALOHA system with SIC-aided collision resolution (SACR). In this system, the users make an initial transmission of a packet as in traditional ALOHA, but the AP is capable of distinguishing the colliding packets if their transmission epochs are at least $\Delta$ (msec) apart from the other packets with the help of SIC. The value of $\Delta$ can present a system sensitivity parameter. As [45] demonstrated the feasibility of this capability at the physical layer with preambles in CoLoRa protocol, we expect that the proposed RA scheme can be integrated with any physical technology that plays the same role as SIC in our system. We characterize the throughput performance of ALOHA with SACR and demonstrate through simulations that the throughput of the proposed ALOHA can be better than that of S-ALOHA, depending on the system parameters, e.g., $\Delta$. In implementing this system, the users are only equipped with a collision timer, whereas the AP is capable of SIC.

As another contribution, in order to achieve the maximum throughput of the proposed system in practice, this work proposes an online backoff algorithm based on backlog size estimation. While the backlogged users form an underlying Markov process, which is not observable to the AP, the AP estimates its size using the length of each idle period and broadcasts an optimal backoff interval based on it. Finally, we compare the proposed ALOHA system with the two existing SIC-based RA systems: a tree algorithm with SIC in [32] and the other is IRA in [39], which are slotted and unslotted systems, respectively.

### C. Organization

The paper is organized as follows: Section II-A introduces the system model of ALOHA with SACR, and Section II-C shows how the proposed system works in detail. The analytical framework for the system throughput analysis is introduced. The main results are presented in Section III. Detailed analysis is given as lemmas in Section III-A. More extended results of the throughput are shown in Section III-B. The proposed online backoff algorithm is introduced in Section III-D. Some lengthy derivations are placed in Appendices for readability. Section IV discusses the numerical results that validate our analysis. Finally, concluding remarks are given in Section V.

## II. The Proposed ALOHA system with SIC

### A. System Model

Suppose an ALOHA system with time-division duplex (TDD), where an AP is centered within its coverage area, and users are randomly distributed, maintaining wireless connections with the AP. Although the ALOHA system with SACR can work in both TDD as well as frequency division duplex (FDD) modes, TDD is more effective for SIC due to its reciprocity of uplink and downlink channels in low mobility scenarios. For convenience, the terms “devices” and “users” are often used interchangeably.

Let us assume $N$ saturated users, meaning that each user always has a packet to transmit. The length of each packet is $T$ (msec) long. The users obey the following backoff procedure before transmitting their packets: User $i$ draws a random backoff time, say $\tau_i$, from an exponential distribution with a mean of $1/\beta$ (sec). The cumulative distribution function (CDF) and the probability density function (PDF) of the backoff time are given by:

$$F_\tau(x) = 1 - e^{-\beta x} \quad \text{and} \quad f_\tau(x) = \beta e^{-\beta x}. \tag{1}$$

After drawing the backoff time, user $i$ sets its backoff timer for $\tau_i$: Before this timer expires, it continuously monitors

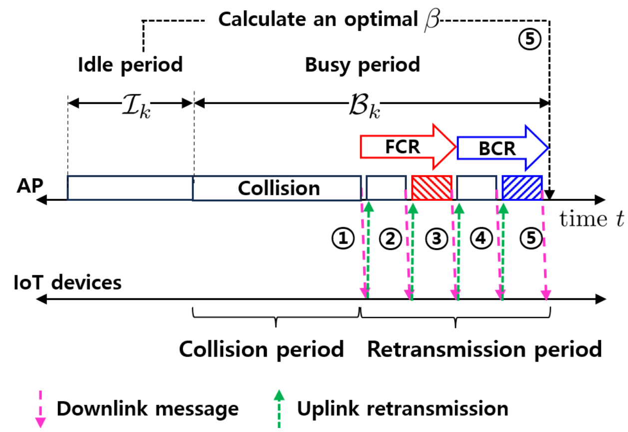



Fig. 1. Flow diagram of Events.

the downlink broadcast message to see if the AP is in the *retransmission period*. If the user $i$, whose backoff timer will expire in the middle of the retransmission period, finds that the AP is in the retransmission period, it draws a new backoff time again and repeats the previous backoff procedure. Otherwise, the user triggers a *collision* timer and transmits its packet. Each transmitting user uses a collision timer to measure the time interval between the beginning of the packet transmission time and the end of the collision period. We assume a strong channel coding such that any channel error could be corrected upon a single packet transmission with no processing delay. We thus focus only on performance degradation due to collisions.

As shown in Fig. 1, where four small rectangles in the busy period depict packets, the channel alternates between idle and busy periods, and $\mathcal{I}_k$ and $\mathcal{B}_k$ denote the $k$th idle and busy periods, respectively. The idle period indicates the time period of no (re)transmission to the channel. The busy periods consist of either a successful single packet transmission or a combination of a collision period and a subsequent retransmission period. The retransmission period starts right after a collision period.

In the proposed system, only the packets involved in the collision period are allowed to be retransmitted during this period. The collision resolution process consists of two phases: The forward collision resolution (FCR) phase and the backward collision resolution (BCR) phase. The system transitions from FCR to BCR upon the occurrence of a specific event known as a $\Delta$-collision. However, the AP may choose to abort the entire retransmission period or may not need the BCR phase if it determines that the BCR phase is not beneficial or necessary. Details on how users retransmit during this period, including the FCR, BCR phases, and the definition of $\Delta$-collision, are provided in Section II-C. At the end of either a single packet transmission or the collision period, the AP broadcasts the downlink (broadcast) message. The circled numbers from 1 to 5 indicate the downlink message following the collision period. This message conveys three key pieces of information. The first two pieces of information are the channel outcome (success or collision) and the length of the *remaining* collision period. During the retransmission period, as shown in Fig. 1, the AP transmits downlink messages numbered 1 to 5.

One of the users involved in a collision, whose collision timer matches to the remaining collision period specified in the downlink message, will retransmit its packet during this period. Consequently, each downlink message triggers an uplink retransmission. The retransmission occurs in response to the downlink message without the need to synchronize with a slot boundary in S-ALOHA. In other words, some latency is permissible between the downlink message and the corresponding retransmission. The third information given in the downlink message is related to facilitating SIC. Specifically, the AP informs the transmit power of the downlink broadcast message, the predefined received power that the users should satisfy when transmitting their packet, and pilot signals. The users then perform a transmit power control as follows: They estimate the channel gain and path-loss from the downlink message with pilot signals and transmit their packet with transmit power as much as the channel gain to meet the predefined received power.

### B. Online Backoff Algorithm

In the previous section, we assume $N$ backlogged users in the system, i.e., representing the number of users with a packet to send. In practice, the backlog size $N$, fluctuates over time. The backoff rate $\beta$ should be adjusted such that the mean (access) rate $G = N\beta$ can be controlled to maximize system throughput. However, directly knowing $N$ at any given time is challenging, as users are continuously joining and leaving the backlog. The proposed backoff algorithm addresses this by enabling the system to estimate the mean of fluctuating $N$, denoted by $\overline{m}$, and then broadcasts an optimized $\beta$ at the end of the busy period to maximize throughput, e.g., the downlink message 5 in Fig. 1. The derivation for this algorithm is given in Appendix B, and we introduce the detail of its algorithmic procedure in Section III-D. This section gives a brief overview of the proposed algorithm.

The underlying idea is that the AP determines the optimal backoff rate $\beta$ by monitoring the duration of the idle period $\mathcal{I}_k$, as illustrated in Fig. 1. The idle period $\mathcal{I}_k$ ends when the shortest backoff time of backlogged users expires. In other words, $\mathcal{I}_k$ is the minimum backoff time across $N$ backlogged users. Note that the idle period is our observation, while the number of backlogged users is a hidden Markov process. Generally, as the number of backlogged users $N$ increases, the idle period becomes shorter. Note that newly arriving users during the retransmission period, the users involved in a collision, and the users whose backoff timers expire during the retransmission period will adopt the newly broadcast $\beta$. On the other hand, users who have already set their backoff time will continue using their previously set value. Although this leads to backlogged users having different mean backoff rates $\beta$ over time, the simulations will demonstrate that our backoff algorithm is robust to this variation.

### C. Collision Resolution with SIC

Fig. 2 depicts a timing diagram of ALOHA with SACR, where the rectangles denote a packet of length $T$ (msec), and the numbers inside them denote their transmission order in the collision; that is, '1' indicates the first packet. Note that Fig. 1 is a simplification of the event after $t_3$ in Fig. 2.

For ongoing packet transmission, if there is no interference by other packets, e.g., from $t_1$ to $t_1 + T$ in Fig. 2, the packet transmission is successful. On the other hand, if any user transmits its packet during someone else's packet transmission (e.g., at time $t_2$ and $t_3$), it causes a collision such that none of the packets in transmission can be decoded. After the collision period, the retransmission period *can* follow, in which the retransmissions of the colliding packets take place. We denote the length of the collision period and that of the retransmission period by $\mathcal{C}_N$ and $W_N$ for $N$ saturated users, respectively.

Let us introduce how the users in a collision should retransmit in the retransmission period with their collision timer: In Fig. 2 $t_{c_i}$ denotes the collision timer of the user who transmits packet $i$ when the collision period $\mathcal{C}_N$ is over. For simplicity of the presentation, the collision timers are presented only for the $k$th busy period $\mathcal{B}_k$ in Fig. 2.

*1) FCR phase:* When the users receive the information on the length of the collision period $\mathcal{C}_N$ at time $t_2 + \mathcal{C}_N$, the user transmitting packet 1 finds that its collision timer $t_{f_1}$ is equal to $\mathcal{C}_N$. Then, the user should retransmit its packet first in the retransmission period $W_N$ right after the collision period while the other users wait. When receiving packet 1 successfully in the retransmission period $W_N$, the AP subtracts this packet from the previous collided packets by SIC to find where the second packet transmission began in the collision period. In other words, through the SIC, the AP can extract the information on the remaining collision period from the beginning of the second packet transmission. This procedure, referred to as *the FCR phase*, identifies the remaining collision period by subtracting the packets retransmitted during the retransmission period from the colliding packets decoded by SIC, and it also enables the retransmission of packets whose collision timers are equal to the remaining collision period. It will continue until the $\Delta$-collision occurs, which will be introduced shortly. When the AP broadcasts the information on the remaining collision period after the retransmission of packet 1 at time $t_2 + \mathcal{C}_N + T$, the user with packet 2 can find its collision timer $t_{c_2}$ equal to the remaining collision period. This user then retransmits right away, i.e., at time $t_2 + \mathcal{C}_N + T$. Thus, packets 1 and 2 can be successfully retransmitted in the retransmission period. Subtracting two packets retransmitted, the AP can find the remaining collision period with SIC and broadcast this information. Finally, the user with packet 3 retransmits in the retransmission period. The AP broadcasts the end of the retransmission period at time $t_2 + \mathcal{C}_N + 3T$.

*2) $\Delta$-collision and $s$-collision:* Each user starts its collision timer when transmitting a packet, which includes the propagation delay of the user. For example, for an AP with coverage radius of 5 km, the propagation delay can be $5 \times 10^3/(3 \times 10^8) = 16.6$ ($\mu$sec). When the packet length is on the order of msec, the overall latency takes a small fraction of the packet transmission time. When the (remaining) collision period, say $C_r$, is announced, the users that meet the condition $t_{c_i} - C_r \leq \Delta$ ($\mu$sec) only retransmit. If more than one user meet this condition, another collision occurs in the retransmission period. We now define a $\Delta$-collision as follows:

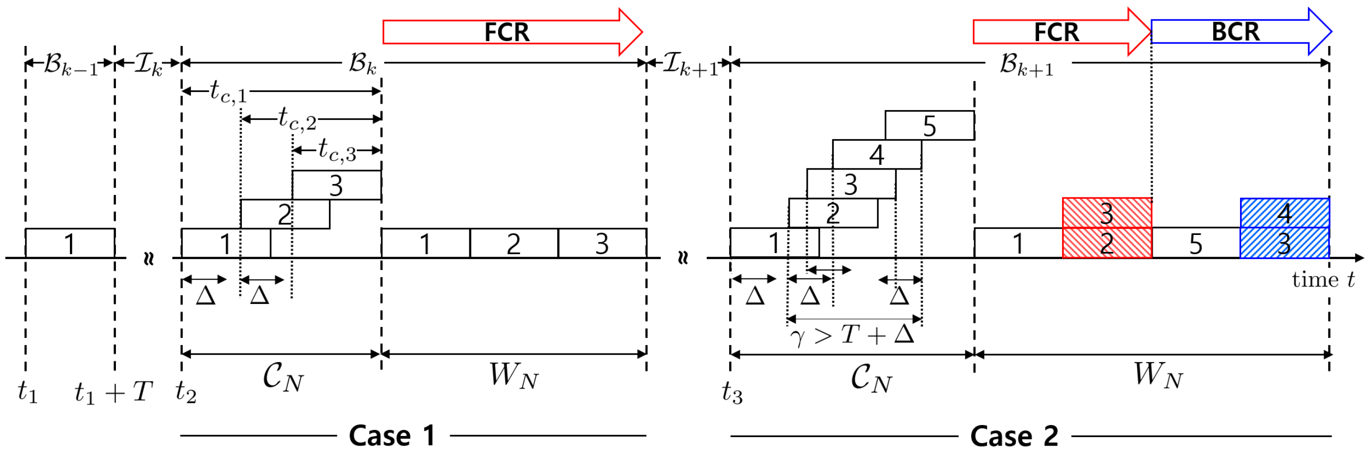



Fig. 2. Timing diagram of ALOHA system with SACR.

**Definition 1.** *A $\Delta$-collision is defined as a collision in which the transmission epochs of at least two packets are separated by no more than $\Delta$ ($\mu$sec) for $\Delta \ll T$.*

When the transmission epochs of the packets are spaced at least $\Delta$ apart, even if they collide, according to the proposed scheme, each packet can be separately retransmitted, one by one, during the retransmission period. In addition to the $\Delta$-collision, we define an $s$-collision:

**Definition 2.** *An $s$-collision is defined as a collision in which the beginning of two packets is separated by at least $\Delta$ ($\mu$sec) for $\Delta \ll T$.*

When an $s$-collision occurs, only one user's collision timer will match the remaining collision period announced by the AP, resulting in a successful retransmission during the retransmission period.

Since each user utilizes a collision timer to measure its own collision period, the interval $\Delta$ serves as a buffering time to accommodate latency, including the propagation delay of each user. It also functions to compensate for any inaccuracies in the time measured by the collision timer. The three packets transmitted between $t_2$ and $t_2 + \mathcal{C}_N$ are distinguishable by SIC since each of their transmission epochs is more than $\Delta$ apart in Fig. 2. For a system with $\Delta \to 0$, which is called *the ideal system*, the AP with SIC can separate the beginning of transmission epochs of any packets in a collision.

Now, let us consider what happens upon a $\Delta$-collision in the collision period starting from $t_3$ in Fig. 2, i.e., case 2. Since the transmission epoch of the first packet is separated more than $\Delta$ (sec) from that of the second packet, it is successfully retransmitted in the subsequent retransmission period. This results from the fact that only packet 1's collision timer matches the collision period. However, the second and third packets experience a $\Delta$-collision; that is, their transmission epochs are separated by no more than $\Delta$ ($\mu$sec) within the collision period. When the AP subtracts the first packet using SIC and broadcasts the remaining collision period, the packets involved with the $\Delta$-collision are retransmitted at the same time. In other words, the users transmitting the second and third packets find that their collision timers align with the remaining collision period given by the AP. This results in another collision in the retransmission period. The AP realizes a $\Delta$-collision when a collision occurs in the retransmission period. Because the users involved in the collision are only allowed to retransmit in the retransmission period, the collision in the retransmission period implies a $\Delta$-collision in the collision period. The BCR phase does not proceed on two occasions. First, there is no $\Delta$-collision in the FCR phase, as shown in case 1 of Fig. 2, where all colliding packets are in $s$-collision. Second, when a $\Delta$-collision does occur, it happens during the last retransmission in the FCR, as illustrated in Case 3 of Fig. 3.

*3) BCR phase:* Once a $\Delta$-collision occurs in the FCR phase, and if the $\Delta$-collision is not the last transmission during the collision period, the AP broadcasts the remaining collision period, which is equal to one packet transmission time $T$. This is the beginning of the BCR phase. Those users whose collision timer is equal to $t_{c_i} - C_r \leq \Delta$ will retransmit their packet. These users are those who believe their packet is the last one in the collision period. In Fig. 2, when the remaining collision time is $T$ in case 2, the user transmitting packet 5 finds its collision timer equal to it. Furthermore, packet 5 is separated more than $\Delta$ from packet 4, it is retransmitted successfully. In the BCR phase, the AP removes the last packet from the remaining collided packets, which allows it to determine the end of the collision period. Additionally, by subtracting one packet transmission time from this point, the AP can identify the start of the second-to-last packet's transmission.

For Fig. 2, suppose that the AP broadcasts the remaining collision period of the second-to-last packet in Case 2. Since packets 3 and 4 are also in the $\Delta$-collision, they are retransmitted simultaneously, which ends up with a collision. In the proposed system, the AP then terminates the retransmission period when a $\Delta$-collision is encountered during the BCR phase. It is important to note that the AP will not perform the FCR phase when the collision period is less than $T + \Delta$, because the colliding packets are in a $\Delta$-collision, and the AP knows that they are inseparable.

*4) Analytical framework for throughput:* Before proceeding further, we introduce how to obtain the throughput. Let $\mathcal{Z}_k$ denote the length of the renewal process (or period), whereas $R_N$ denotes the number of successfully transmitted packets for the system with $N$ users during the renewal period, i.e., reward. Based on the renewal reward theorem [58], the system throughput, which is denoted by $S_N$ for population size $N$, can be expressed as

$$S_N = \frac{\mathbb{E}[R_N]}{\mathbb{E}[\mathcal{Z}_k]}, \tag{2}$$

where in Fig. 2, a renewal period consists of an idle period followed by a busy period, it follows that

$$\mathbb{E}[\mathcal{Z}_k] = \mathbb{E}[\mathcal{I}_k] + \mathbb{E}[\mathcal{B}_k]. \tag{3}$$

Since the system has $N$ saturated users, the idle period above is the mean of the minimum of $N$ independently identically distributed exponential random variables with mean $1/\beta$:

$$\mathbb{E}[\mathcal{I}_k] = \frac{1}{N\beta}. \tag{4}$$

The average busy period of this system in Fig. 2 is expressed as

$$\mathbb{E}[\mathcal{B}_k] = \mathbb{E}[\mathcal{C}_N] + \mathbb{E}[W_N]. \tag{5}$$

To find $\mathbb{E}[R_N]$, let $V_N$ be the number of packets successfully decoded at the end of the retransmission period for $N$ saturated users. When $q_1$ denotes the probability that a single packet is transmitted in a busy period, i.e., one successful transmission, the numerator of (2) can be obtained as

$$\mathbb{E}[R_N] = 1 \cdot q_1 + \mathbb{E}[V_N]. \tag{6}$$

The throughput of the system depends on how many successful retransmissions $V_N$ are made during the retransmission period $W_N$. To analyze the system throughput, we need to calculate $\mathbb{E}[V_N]$, $\mathbb{E}[\mathcal{C}_N]$ and $\mathbb{E}[W_N]$. The following four cases describe the conditions under which a pair of $V_N$ and $W_N$ occur. These cases are depicted in Figs. 2 and 3, and depend on whether the FCR phase alone or both the FCR and BCR phases are necessary. In Case 1, only the FCR phase is needed because all the packets are in $s$-collision. In Case 3, although the FCR phase is required, the last transmission within it experiences a $\Delta$-collision. Case 2 represents a scenario where the AP proceeds with both the FCR and BCR phases, and a $\Delta$-collision occurs in each phase. Finally, in Case 4, both FCR and BCR phases were executed as in Case 2, but a $\Delta$-collision occurs only in the FCR phase. This $\Delta$-collision is subsequently found in the BCR phase.

1) Case 1: All the packets are in $s$-collision during the collision period. None of the packets are in $\Delta$-collision. The FCR phase is enough for the AP to get all the packets retransmitted successfully. This represents the collision that occurred at time $t_2$ in Fig. 2. In this case, we have $V_N \in \{2, 3, \ldots, N\}$, since at least two packets are involved in a collision and all the packets involved with the collision can be successfully retransmitted. The length of the retransmission period is $W_N = V_N \cdot T$.
2) Case 2: Suppose that a $\Delta$-collision in the FCR phase occurs, the AP initiates the BCR phase, because the $\Delta$-collision is not the last transmission in the collision period. This is the event depicted at time $t_3$ in Fig. 2. During the BCR phase, another $\Delta$-collision can occur, but this one is different from the $\Delta$-collision in the FCR phase. After the $\Delta$-collision is found in the BCR phase, the AP terminates the retransmission period. Fig. 1 depicts this case, too. As in Fig. 2, when three packets are in $\Delta$-collision, at most $N-3$ packets can be successfully retransmitted. We thus have $V_N = \{0, 1, \ldots, N-3\}$ and $W_N = (V_N + 2)T$.
3) Case 3: Suppose that the first $l$ packets are in $s$-collision out of $k$ colliding packets in total for $1 \le l \le k-2$, while the remaining $k-l$ packets are in $\Delta$-collision. This is depicted as case 3 in Fig. 3 with $k = 5$ and $l = 2$; the first two packets are successfully retransmitted in the FCR phase. After that, the AP finds a $\Delta$-collision, which is the last transmission. Notice that for $l = 0$, this case includes all the packets in a single $\Delta$-collision so that the AP allows no retransmissions, i.e., no FCR phase. On the other hand, the maximum number of packets successfully decoded is up to $N-2$ since a $\Delta$-collision destroys at least two packets. Thus, we have $V_N \in \{1, \ldots, N-2\}$ and $W_N = (V_N + 1)T$ for $l \ge 1$; for $l = 0$, $V_N = 0$ and $W_N = T$.
4) Case 4: Suppose that a $\Delta$-collision takes place during the FCR phase, and it is not the last transmission. The AP then initiates the BCR phase so that the last packet in the collision period can be successfully retransmitted. However, this case is the one in which a $\Delta$-collision occurs during the BCR, which is found identical to the $\Delta$-collision that the AP encountered in the FCR. Fig. 3 shows this case that the third, fourth, and fifth packets are in a $\Delta$-collision in the middle. In particular, the $\Delta$-collision in the FCR phase can occur as soon as it starts, whereas at least one packet can be successfully retransmitted in the BCR phase. Therefore, we have $V_N \in \{1, 2, \ldots, N-2\}$ and add one to $W_N$, i.e., $W_N = (V_N + 1)T$ due to one $\Delta$-collision.

Using the four cases above, let us formulate $\mathbb{E}[W_N]$ in (5) and $\mathbb{E}[V_N]$ in (6). Let $v_k(l)$ denote the probability that $l$ packets are successfully retransmitted for Case $k \in \{1, 2, 3, 4\}$. The detailed derivation of $v_k(l)$ is given in Section III-A. With $v_k(l)$, we can write $\mathbb{E}[V_N]$, i.e., the average number of packets successfully transmitted during a busy period, is expressed as

$$\mathbb{E}[V_N] = \sum_{l=2}^{N} l v_1(l) + \sum_{l=0}^{N-3} l v_2(l) + \sum_{n=0}^{N-2} l v_3(l) + \sum_{l=1}^{N-2} l v_4(l). \tag{7}$$

The lower and upper limits of each summation are explained in each case above.

To find the denominator of (2), we consider the average of

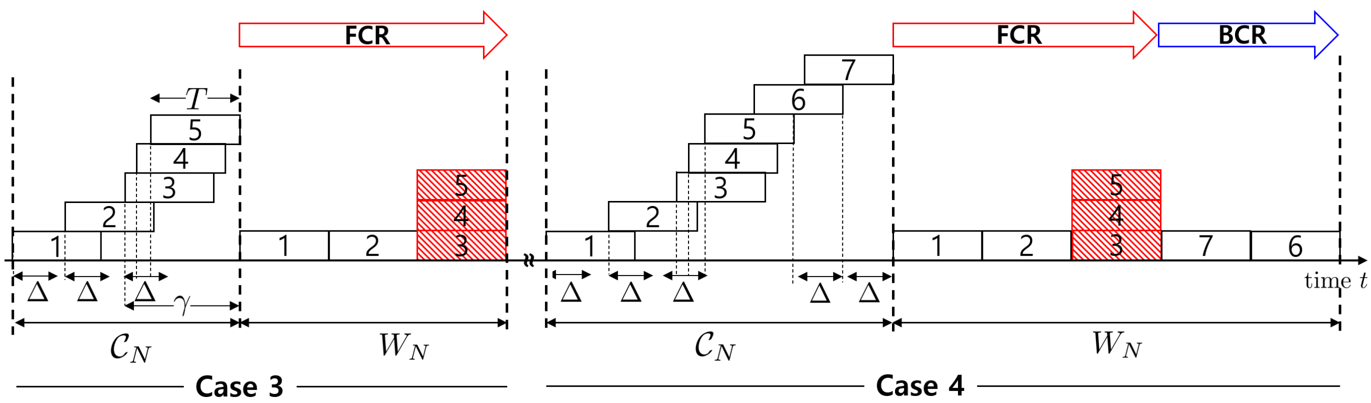



Fig. 3. Timing diagram of Cases 3 and 4.

the retransmission periods, $\mathbb{E}[W_N]$, as

$$\mathbb{E}[W_N] = \left[ \sum_{l=2}^{N} l v_1(l) + \sum_{l=0}^{N-3} (l+2) v_2(l) + \sum_{l=1}^{N-2} (l+1) v_3(l) + \sum_{l=1}^{N-2} (l+1) v_4(l) \right] T. \quad (8)$$

So far we discussed $\mathbb{E}[W_N]$ in (5) and $\mathbb{E}[V_N]$ in (6). The next section examines $\mathbb{E}[\mathcal{C}_N]$ in (5).

### D. Implementation Issues

Before we proceed to the system performance analysis in Section III, let us briefly discuss the overhead and cost associated with implementing this system.

First, the AP should be capable of performing SIC, whereas users need a collision timer and be aware of the retransmission period. While the proposed system achieves higher throughput than traditional ALOHA, this comes with the trade-off of increased energy consumption, as users must listen to the downlink message to determine the retransmission period. It is important to note that LoRaWAN, which employs traditional ALOHA, categorizes devices into three classes: A, B, and C. Class C devices continuously listen to downlink messages, whereas Class A devices spend most of their time in sleep mode. For improved throughput and performance, devices with higher energy consumption are often considered in practice. Thus, while the proposed scheme is well-suited for Class C devices, we should focus on low energy consumption in the future. Second, as mentioned in Section II-C2, the value of $\Delta$ can absorb the propagation delay and latency between devices and the AP such that a potential mismatch in the remaining collision period measured by devices and the AP due to such delays can be resolved. Finally, the downlink message from the AP should deliver the information of the remaining collision period and the indicator of the retransmission period.

## III. Analysis and Algorithm Design

It is important to notice that the proposed system does not alter how a collision occurs or the collision period $\mathcal{C}_N$ in the traditional ALOHA system. The proposed system adds the retransmission period, in which SIC can be efficiently used to determine the sequence of retransmissions. Thus, regarding $\mathbb{E}[\mathcal{C}_N]$, we utilize the following two lemmas for the traditional ALOHA system [59].

**Lemma 1.** *Let $q_j$ be the probability that while $j$ users have been transmitting, none of $N-j$ users (re)transmits during the packet transmission time of the $j$th user, i.e., $T$ (sec):*

$$q_j = e^{-(N-j)\beta T}. \quad (9)$$

*Proof:* This is the probability that $j$ packets are involved in a collision. See Lemma 2 in [59]. ■

**Lemma 2.** *Let $\varphi_{N-j}$ denote the time interval from the beginning of the $j$-th packet transmission to that of the $(j+1)$th packet in a collision period, when $j+1$ packets are in a collision. The average of the collision period $\mathbb{E}[\mathcal{C}_N]$ is obtained as*

$$\mathbb{E}[\mathcal{C}_N] = T + \sum_{j=1}^{N-1} \mathbb{E}\left[\varphi_{N-j}\right] \prod_{i=1}^{j} (1-q_i), \quad (10)$$

*where $\mathbb{E}\left[\varphi_{N-j}\right]$ is expressed as*

$$\mathbb{E}\left[\varphi_{N-j}\right] = \frac{1-[1+(N-j)\beta T]e^{-(N-j)\beta T}}{\left(1-e^{-(N-j)\beta T}\right)(N-j)\beta}. \quad (11)$$

*Proof:* See Lemma 3 in [59]. ■

According to (2), we get the throughput of this system as follows.

**Theorem 1.** *The throughput of ALOHA with SACR is expressed as*

$$S_N = \frac{e^{-(N-1)\beta T} + \mathbb{E}[V_N]}{\frac{1}{N\beta} + \mathbb{E}[\mathcal{B}_k]}, \quad (12)$$

*where two lengthy expressions of $\mathbb{E}[V_N]$ in (7) and $\mathbb{E}[W_N]$ in (8) can be obtained once $v_i(l)$ is obtained in the next section.*

*Proof:* According to Lemma 1, $q_1$ is one successful single transmission; that is, $q_1 = e^{-(N-1)\beta T}$. Using (4), (5), and (6), we obtain the result. ■

### A. Derivations of $v_k(l)$

This section investigates $v_k(l)$ for $\mathbb{E}[V_N]$ in (7) and $\mathbb{E}[W_N]$ in (8). As $v_k(l)$ depends on $\Delta$-collision and $s$-collision probabilities, Lemmas 3 and 4 examine them, respectively. Lemmas 5 to 8 analyze $v_k(l)$ for $k = 1, 2, 3,$ and $4$, respectively.

Let us recall that user $i$ adopts a backoff algorithm whose interval $\tau_i$ is exponentially distributed with mean $1/\beta$. In addition, $t_i$ denotes the remaining backoff time of user $i$ after an event. Due to the memoryless property of the exponential distribution, $t_i$ follows the same exponential distribution of $\tau_i$. Now, we define the ordered distribution of backoff times $\tau_1$, $\tau_2, \ldots, \tau_N$ as follows.

**Definition 3.** *Let $\tau_{(r)}$ and $t_{(r)}$ denote the $r$-th order of $\tau_i$'s and $t_i$'s, respectively. Since $\tau_i$ and $t_i$ follow the same distribution, $\tau_{(r)}$ and $t_{(r)}$ have the same distribution. If there are a total of $N$ independently and identically distributed exponential backoff intervals (chosen by $N$ users), the cumulative distribution function (CDF) and probability density function (PDF) of the $r$-th order backoff interval are respectively expressed as [58]*

$$F_{\tau_{(r)}}(x;N) = \sum_{n=r}^{N} \binom{N}{n} [F_\tau(x)]^n [1 - F_\tau(x)]^{N-n} \tag{13}$$

*and*

$$f_{\tau_{(r)}}(x;N) = \frac{N! f_\tau(x) [F_\tau(x)]^{r-1} [1 - F_\tau(x)]^{N-r}}{(r-1)!(N-r)!}, \tag{14}$$

*where* (1) *is used.*

*It is worth noting that the PDF and CDF of $t_{(r)}$ follow* (13) *and* (14)*, respectively, owing to the assumption on the exponential distribution for $\tau_i$'s.*

**Lemma 3.** *($\Delta$-collision probability) Let $g_j$ be the probability that the $j$th packet is in $\Delta$-collision with the $(j+1)$th packet. This is obtained as*

$$g_j = F_{\tau_{(1)}}(\Delta; N - j). \tag{15}$$

*Proof:* The $\Delta$-collision between the $j$th packet and the $(j+1)$th packet occurs when the shortest backoff time of $N-j$ users, i.e., $\tau_{(1)}^{(N-j)}$, is less than $\Delta$, i.e.,

$$g_j = \Pr\left[\tau_{(1)}^{(N-j)} < \Delta\right] = F_{\tau_{(1)}}(\Delta; N - j), \tag{16}$$

which completes the proof. ■

**Lemma 4.** *($s$-collision probability) Let $h_j$ denote the $s$-collision probability between the $j$-th and the $(j+1)$th packet, which can be obtained as*

$$h_j = e^{-(N-j)\beta\Delta} - e^{-(N-j)\beta T}. \tag{17}$$

*Proof:* When one of $N-j$ users with the shortest backoff time transmits, it is the $(j+1)$th user. If the shortest backoff time falls into the interval between $\Delta$ and $T$, the $s$-collision occurs. Thus, we can get $h_j$ as

$$\begin{aligned} h_j &= \Pr\left[\Delta \le \tau_{(1)}^{(N-j)} \le T\right] \\ &= F_{\tau_{(1)}}(T; N-j) - F_{\tau_{(1)}}(\Delta; N-j) \\ &= e^{-(N-j)\beta\Delta} - e^{-(N-j)\beta T}. \end{aligned} \tag{18}$$

■

**Corollary 1.** *Given that $l$ packets for $l \ge 2$ are in collision, we get the probability that $l$ packets are in $s$-collision as*

$$\mathcal{H}_1^{(l-1)} \triangleq \prod_{i=1}^{l-1} h_i. \tag{19}$$

*Proof:* Since $h_1$ is the probability that the first and second packets are in $s$-collision, multiplying $h_i$ up to $l-1$ yields the result. In particular, we have $\mathcal{H}_1^{(0)} = 1$. ■

In the following four lemmas, we examine $v_k(l)$ for $k = 1, 2, 3,$ and $4$.

**Lemma 5.** *(Case 1) Let $v_1(l)$ denote the probability that $l$ packets are in collision, but they all are successfully retransmitted in the retransmission period. For $2 \le l \le N$, we get $v_1(l)$ as*

$$v_1(l) = q_l \mathcal{H}_1^{(l-1)}. \tag{20}$$

*Proof:* Notice that $q_l$ is the probability that $l$ packets are in collision. For these $l$ packets to be successfully retransmitted in the retransmission period, all of them should be in $s$-collision with probability $\mathcal{H}_1^{(l-1)}$. ■

**Lemma 6.** *(Case 2) Let $v_2(l)$ for $2 \le l \le N$ denote the probability that the AP finds two $\Delta$-collisions in the FCR and BCR phases each, whereas they are not duplicated. This can be found as*

$$\begin{aligned} v_2(l) = \sum_{k=0}^{l} \mathcal{H}_1^{(k)} \sum_{n=l+3}^{N} \Bigg( & \underbrace{g_{k+1} \prod_{m=k+2}^{n-l+k-2} (1 - q_m) g_{n-l+k-1}}_{(*)} \\ & - F_{\tau_{(n-(l+1))}}(\Delta; N-(k+1)) \Bigg) \mathcal{H}_{n-l+k}^{(n-1)} q_n. \end{aligned} \tag{21}$$

*Proof:* The first term $\mathcal{H}_1^{(k)}$ is the probability that the first $k$ packets in $s$-collision can be retransmitted successfully in the FCR phase when there are more than $k$ packets. Here, we assume that a $\Delta$-collision occurs at the $(k+1)$th packet in the FCR phase and the $(n-l+k)$th packet in the BCR phase. The term indicated by $(*)$ is the probability that the $(k+1)$th packet is the first packet involved in the $\Delta$-collision, while the $(n-l+k)$th packet is the last packet in the $\Delta$-collision, with $n-l-2$ collided packets in between. In Lemma 8, we find the probability that after the beginning of the $(k+1)$th packet transmission, all of the next $n-(l+1)$ packets are in $\Delta$-collision with the $(k+1)$th packets, i.e., $F_{\tau_{(n-(l+1))}}(\Delta, N-k-1)$. By subtracting this from the term marked by $(*)$, we get the probability of two $\Delta$-collisions, but not identical one. During the BCR phase, $l-k$ packets are successfully retransmitted with probability $\mathcal{H}_{n-l+k}^{(n-1)}$. This results in successful retransmissions of $l$ packets in total when $n$ packets are in collision with probability $q_n$. ■

**Lemma 7.** *(Case 3) Let $v_3(l)$ for $1 \le l \le N-2$ denote the probability that the first $l$ packets are in $s$-collision, while the* *remaining packets are transmitted with $T+\Delta$, i.e., $\Delta$-collision. This can be found as*

$$v_3(l) = \mathcal{H}_1^{(l)} \sum_{n=l+2}^{N} F_{\tau_{(n-(l+1))}}(\Delta; N-(l+1))q_n. \quad (22)$$

*Proof:* The case 3 corresponds to a collision of $l+2+i$ packets for $0 \le i \le N-2-l$, in which the first $l$ packets are in $s$-collision and the remaining $2+i$ packets are transmitted during $T+\Delta$ (sec). The first term $\mathcal{H}_1^{(l)}$ indicates that the first $l$ packets are in $s$-collision when there are more than $l$ packets. Then, all of the next $n-l-1$ packets transmitted after the $(l+1)$th packet are in $\Delta$-collision with the $(l+1)$th packet. This occurs with probability $F_{\tau_{(n-(l+1))}}(\Delta; N-(l+1))$. Finally, probability $q_n$ ensures that $n$ packets are in collision, and after the beginning of the $n$-th packet transmission for $n \ge l+2$, no packet collides with it. ■

**Lemma 8.** *(Case 4) Let $v_4(l)$ denote the probability that $l$ packets are successfully retransmitted in the FCR and BCR phases when the packets in the middle of a collision are in a single $\Delta$-collision. We get $v_4(l)$ as*

$$\begin{aligned} v_4(l) = &\sum_{k=0}^{l-1} \mathcal{H}_1^{(k)} \sum_{n=l+2}^{N} F_{\tau_{(n-(l+1))}}(\Delta; N-(k+1)) \\ &\times \mathcal{H}_{n-l+k}^{(n-1)} q_n, \end{aligned} \quad (23)$$

*where $\mathcal{H}_{n-l+k}^{(n-1)} = 1$ for $l = k$.*

*Proof:* As before, $\mathcal{H}_1^{(k)}$ means the probability that the first $k$ packets are in $s$-collision so they can be successfully retransmitted in the FCR phase for $0 \le k \le l-1$. After the beginning of the $(k+1)$th packet transmission, all of the next $n-(l+1)$ packets for $l+2 \le n \le N$ are in $\Delta$-collision with the $(k+1)$th packet. This occurs with probability $F_{\tau_{(n-(l+1))}}(\Delta; N-(k+1))$ for $0 \le k \le l-1$. After the beginning of the $(n-l+k)$th packet transmission, $l-k$ packets can be successfully retransmitted in the BCR phase with probability $\mathcal{H}_{n-l+k}^{(n-1)} q_n$. ■

### B. System with Infinite Population Model

Suppose that the users with a packet to transmit arrive at the system according to the Poisson process with mean rate $G$. In this system, we assume that $N\beta = G$ as $N \to \infty$ and $\beta \to 0$. This is called an infinite population model.

**Theorem 2.** *The throughput of ALOHA with SACR for the infinite population model is expressed as*

$$\begin{aligned} S_\infty(\Delta) &= \lim_{N\to\infty} \frac{\mathbb{E}[R_N]}{\mathbb{E}[\mathcal{Z}_k]} \quad (24) \\ &= \frac{G\left\{\mathsf{q}(1-\mathsf{h})^2 + \mathsf{h}\left(2\mathsf{g} + \mathsf{q}(2-\mathsf{h})\right)\right\}}{e^{GT}(1-\mathsf{h})^2 + GT\left[2\mathsf{g} + \mathsf{q}\mathsf{h}(2-\mathsf{h})(1+G\Delta) - 2\mathsf{q}G\Delta\right]}, \end{aligned}$$

*in which $\mathsf{g}$ and $\mathsf{h}$ respectively denote the $\Delta$-collision probability and the $s$-collision probability under the infinite population model:*

$$\begin{aligned} \mathsf{g} &= \lim_{N\to\infty} g_j \quad (25) \\ &= \lim_{N\to\infty} \left(1 - e^{-(N-j)\beta\Delta}\right) = 1 - e^{-G\Delta} \end{aligned}$$

*and*

$$\begin{aligned} \mathsf{h} &= \lim_{N\to\infty} h_j \\ &= \lim_{N\to\infty} \left(e^{-(N-j)\beta\Delta} - e^{-(N-j)\beta T}\right) \\ &= e^{-G\Delta} - e^{-GT}. \end{aligned} \quad (26)$$

*Finally, $\mathsf{q}$ corresponds to the probability that $j$ packets are in a collision and $N-j$ packets are not engaged in the collision, i.e., $q_j$ under the same infinite population assumption:*

$$\mathsf{q} = \lim_{N\to\infty} q_j = \lim_{N\to\infty} \left(e^{-(N-j)\beta T}\right) = e^{-GT}, \quad (27)$$

*which is the probability that no packet is transmitted during one packet transmission time $T$ according to the Poisson process with mean rate $G$.*

*Proof:* See Appendix A. ■

### C. Ideal Systems: $\Delta \to 0$

So far, we have assumed that the packets in a collision are distinguishable when their transmission epochs are at least $\Delta$ apart. Let us consider the system with $\Delta \to 0$, where any packets in the FCR phase can be distinguished by SIC, no matter how closely the transmission epochs of the packet are overlapped. Thus, no $\Delta$-collision occurs in the FCR phase, and the BCR phase is not needed. We call this system an *ideal* system, where the FCR phase is enough to resolve the colliding packets. To analyze the throughput of this ideal system, let us consider the following.

**Corollary 2.** *Let $\tilde{v}_i(l)$ denote the probability of case $i$ for $\Delta \to 0$: We get*

$$\tilde{v}_1(l) = \prod_{j=1}^{l-1} \left(1 - e^{-(N-j)\beta T}\right) e^{-(N-l)\beta T}, \quad (28)$$

*and $\tilde{v}_i(l) = 0$ for $i = 2, 3,$ and $4$.*

*Proof:* Setting $\Delta = 0$ into $v_1(l)$ yields (28). Notice that $g_j = 0$ and $h_j = 1 - e^{-(N-j)\beta T}$ for $\Delta = 0$, whereas $q_j$ is independent of $\Delta$. ■

**Theorem 3.** *Let $\widetilde{S}_N$ denote the throughput of the ideal ALOHA with SACR. It can be expressed as*

$$\widetilde{S}_N = \frac{e^{-(N-1)\beta T} + \mathbb{E}[\mathcal{V}_N]}{\frac{1}{N\beta} + \mathbb{E}[\mathcal{C}_N] + \mathbb{E}[\mathcal{W}_N]}, \quad (29)$$

*in which $\mathbb{E}[\mathcal{V}_N]$ and $\mathbb{E}[\mathcal{W}_N]$ denote the average number of successfully transmitted packets during the retransmission period and the mean length of the retransmission period, respectively, for the ideal system. They are given in the proof.*

*Proof:* It is notable that letting $\Delta \to 0$ does not affect the length of the collision period $\mathcal{C}_N$. Since it makes $\tilde{v}_i(l) = 0$ for $i = 2, 3, 4$ shown in Corollary 2, we can write $\mathbb{E}[\mathcal{V}_N]$ and $\mathbb{E}[\mathcal{W}_N]$ only with $\tilde{v}_1(l)$ as

$$\mathbb{E}[\mathcal{V}_N] = \sum_{l=2}^{N} l\tilde{v}_1(l), \quad \text{and} \quad \mathbb{E}[\mathcal{W}_N] = T\sum_{l=2}^{N} l\tilde{v}_1(l). \quad (30)$$

**Algorithm 1** Online control algorithm for $\Delta$-collision

1: Initialize $\hat{\lambda} = 0.2$ at epoch $k = 0$.
2: **if** either success or collision begins **then**
3: $\quad \hat{\lambda} = \theta\hat{\lambda} + (1-\theta)\left[\frac{1}{\mathcal{I}+\hat{\mathcal{B}}_s}\mathbb{I}(S) + \frac{V}{\mathcal{I}+\hat{\mathcal{B}}_c}\mathbb{I}(C)\right]$.
4: $\quad$ **if** success **then**
5: $\quad\quad \overline{m} = \overline{m}e^{-\beta\mathcal{I}} + \hat{\lambda}\cdot\hat{\mathcal{B}}_s$.
6: $\quad$ **else**
7: $\quad\quad \overline{m} = \max(\overline{m}e^{-\beta\mathcal{I}} + 1 - V, \epsilon) + \hat{\lambda}\cdot\hat{\mathcal{B}}_c$.
8: $\quad$ **end if**
9: **end if**
10: Broadcast $\beta = \kappa/(\overline{m}\cdot T)$ at the end of busy period.

This completes the proof. ■

Let us examine the throughput of the ideal system under the infinite population model.

**Corollary 3.** *The throughput of the ideal system under the infinite population assumption can be obtained as*

$$\lim_{\Delta\to 0} S_\infty(\Delta) = \frac{G}{1+GT(1-e^{-2GT})}. \tag{31}$$

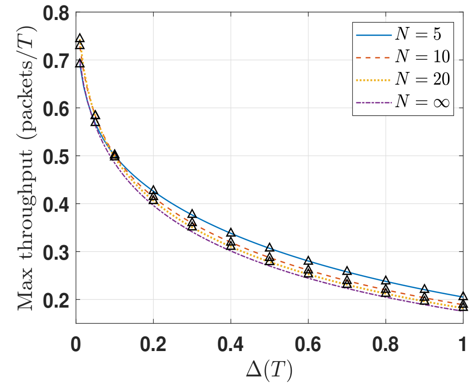
*Proof:* We can get (31) by letting $\Delta \to 0$ in (24). Notice that $\mathsf{g} = 0$ and $\mathsf{h} = 1 - e^{-GT}$. ■

As $G$ increases, we can see that $\lim_{\Delta\to 0} S_\infty(\Delta) = \frac{1}{T}$ in (31).

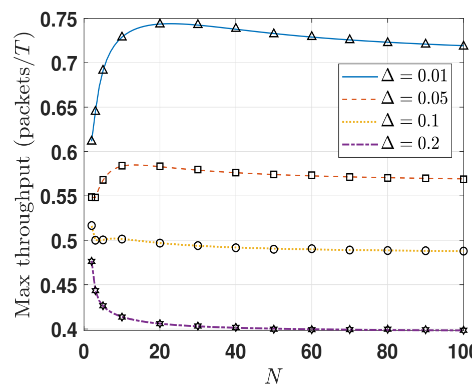
### D. Online Backoff Algorithm

To start with, we assume that the users with a new packet arrive at the system according to Poisson process with mean rate $\lambda$ (packets/packet transmission time) and that users can hold only one packet to transmit. The user with a new packet to transmit schedules its transmission time according to an exponential distribution with mean $1/\beta$ that the AP broadcasts. The notations in Algorithm 1 are explained as follows. First, $\hat{\lambda}$ denotes the new packet arrival rate estimated by the AP and $\mathcal{I}$ is the length of an idle period that the AP observes. The lengths of busy period with a single packet transmission and with multiple packet transmissions are denoted by $\hat{\mathcal{B}}_c$ and $\hat{\mathcal{B}}_s$, respectively. Thus, $\mathcal{I} + \hat{\mathcal{B}}_c$ or $\mathcal{I} + \hat{\mathcal{B}}_s$ indicates the end of busy period (collision or success) after the preceding idle period ends. In line 3, $\mathbb{I}(x)$ is an indicator function that takes one if $x$ is true; otherwise, it is zero, whereas $\theta \in (0,1)$ is a weighting factor of the first-order autoregressive (AR) model for estimation on $\hat{\lambda}$. In line 7, $V$ denotes the number of packets successfully transmitted after the retransmission period.

The new packet arrival rate $\hat{\lambda}$ is estimated in line 3 based on the AR model under the assumption that for a stable system, the mean rate of new packet arrivals should be equal to the mean output rate. In estimating the mean backlog size $\overline{m}$, it is important to note that since the length of the idle period is a function of backlog size $N$, the observation on the length of the idle period $\mathcal{I}$ and the backlog size $N$ form a hidden Markov process. Thus, the AP estimates $\overline{m}$ after observing the idle period. As shown in lines 5 and 7, it can be seen that $\overline{m}$ exponentially decreases as the idle period $\mathcal{I}$ increases. Notice that 'success' in line 4 means a single packet transmission, whereas line 6 shows the end of the retransmission period. In line 10, $\kappa$ is a maximizer of (24), which is numerically obtained. From $\kappa = \overline{m}\beta T \triangleq G$, we have the optimal $\beta = \frac{\kappa}{\overline{m}T}$.




(a) Maximum throughput versus $\Delta$




(b) Maximum throughput versus $N$

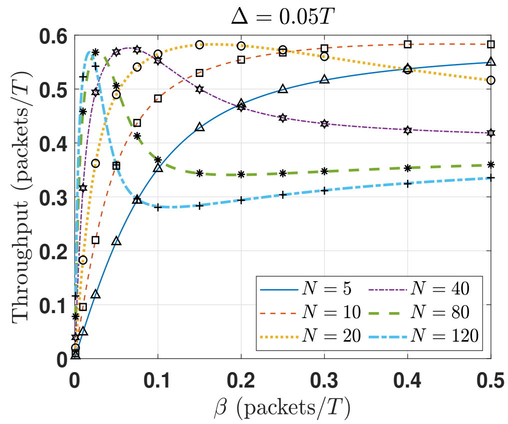
Fig. 4. Maximum throughput of ALOHA with SACR versus $\Delta$.

## IV. Numerical Results

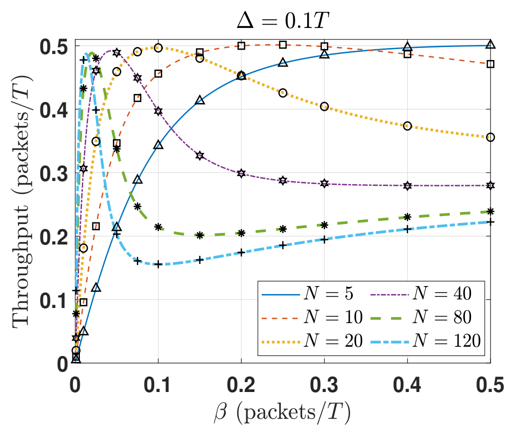
In all the figures in this section, the lines and symbols show the analytical and simulation results, respectively. We build a simulation program with Matlab. The run time of simulation is set to $3\times 10^6$ (in packet transmission time unit), and its time-average is obtained. Unless otherwise stated, we set one packet transmission time $T = 1$ (msec). In Fig. 4, we examine the maximum (achievable) throughput with optimal $\beta$ as either $\Delta$ or $N$ varies. Fig. 5 shows how the throughput varies as the backoff rate $\beta$ changes. In Fig. 6, we compare the proposed system with two schemes in [32] and [39].It is notable that, as in our proposed system requirement (see Section II-D), the competitive approaches also require Class C devices for the SIC operation, thus making the benchmark comparison fair.

### A. Achievable Throughput

Figs. 4(a) and 4(b) illustrate the maximum throughput $S_N$ of the proposed system, where we find numerically the backoff rate $\beta$ that maximizes the throughput and apply it. In Fig. 4(a), the maximum throughput is presented with respect to $\Delta = c{\cdot}T$ for $0 \le c \le 1$. Firstly, the maximum throughput seems more

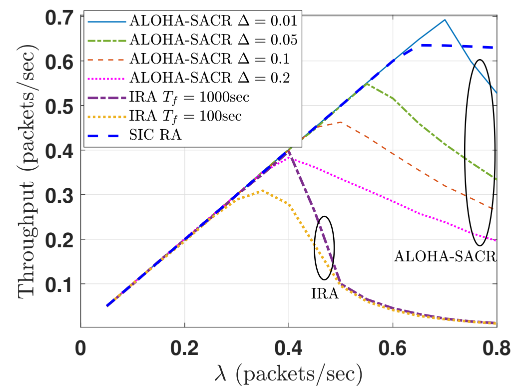



(a) Throughput versus $\beta$

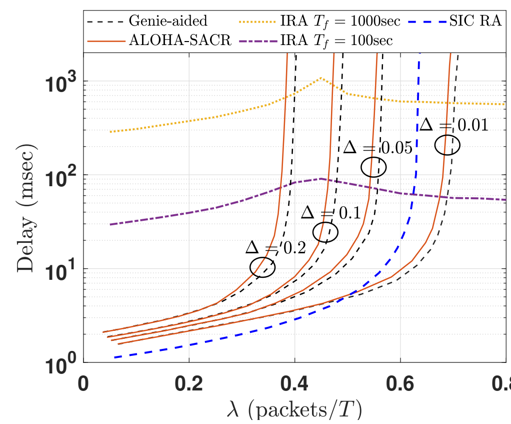



(b) Throughput versus $\beta$

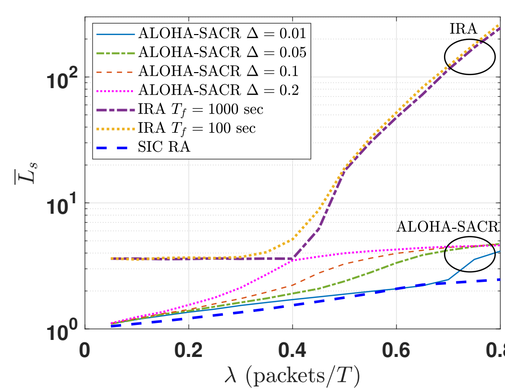
Fig. 5. Throughput versus $\beta$ for $\Delta = 0.05T$ and $0.1T$.




(a) Throughput comparison




(b) Comparison of average access delay




(c) Comparison of $\overline{L}_s$

Fig. 6. Comparison with two existing competitive schemes, SIC-RA [32] and IRA [39].

sensitive to $\Delta$ rather than the population size $N$ when $\beta$ is optimally chosen. Secondly, as long as $\Delta$ is no larger than $0.3T$ (around one-quarter of one packet transmission time $T$), the proposed ALOHA with SACR can offer throughput of more than $0.35$ even for large population size. This can also be observed in Fig. 4(b), which shows the maximum throughput for various $\Delta$'s as $N$ increases. Since it can be seen that the smaller the $\Delta$, the higher the throughput $S_N$, the ideal system ($\Delta \to 0$) provides the upper bound of $S_N$.

## *B. Effect of Backoff Rate $\beta$*

Figs. 5(a) and 5(b) show the throughput $S_N$ for two values of $\Delta$ ($\Delta = 0.05T$ and $0.1T$) as $\beta$ and $N$ increase. We can make the following three observations. First, although we use two different values of $\Delta$ in each figure, $S_N$ versus $N$ behaves similarly; the difference is the maximum throughput. More specifically, as observed in Fig. 4(b), the throughput in Fig. 5(a) does not grow more than 0.6 for $\Delta = 0.05T$ as $N$ increases beyond 10. Likewise, the throughput in Fig. 5(b) does not exceed 0.51 for $\Delta = 0.1T$ for $N > 10$. The second observation is that in Figs. 5(a) and 5(b), there exists a throughput-optimal $\beta$ of maximizing $S_N$ for each population size $N$. In other words, there exists a throughput optimal $G = N\beta$ for each $\Delta$. Finally, as $N$ increases in both figures, the throughput-optimal $\beta$ gets smaller, i.e., the mean of backoff interval should be longer as $N$ becomes large.

## *C. Performance Comparison with the Competitive Approaches*

Fig. 6 presents a comparison of the proposed scheme against three others in terms of throughput, delay, and the number of (re)transmissions for success. As the first benchmark, we consider a genie-aided system [62]. In contrast with the proposed backoff algorithm that estimates the mean number of backlogged users, the AP in the genie-aided system knows the exact backlog size every time instant so that the throughput-optimal control can be realized. Thus, the RA delay of the proposed system cannot be better than that of the genie-aided system. Note that we use $\kappa = 2.729$, 1.614, 1.302, and 1.016 in Algorithm 1 for $\Delta = 0.01$, 0.05, 0.1, and 0.2. These are numerically obtained and are also depicted in Fig. 4(a) for $N = \infty$. The second scheme is SIC resolve algorithm (SIC RA) [32], which adopts a cross-slot SIC: In the SIC RA, the AP is capable of detecting if the number of packets in a collision (at a slot) is no more than $M$. If it finds that the packets are no more than $M$ in a collision slot, the AP initiates a SIC-enabled resolution procedure (SRP), for which the packets in the collision are retransmitted and decoded by SIC. Otherwise, the packets are retransmitted according to the proposed backoff algorithm without an SRP. The $M$ indicates the maximum number of packets decoded by SIC, which might be an abstraction for the limitation of physical layer technology. It is not mentioned in [32] how the AP can detect more than $M$ packets in a collision. Here, we consider $M = 3$ for the SIC resolve algorithm. As mentioned in Section I, the comparison of unslotted ALOHA and slotted RA systems may sound unfair. However, while some similar performance is required, time-slotting is a major issue for implementation, but one of them can be chosen. It should be noted that the SIC RA makes use of SIC to decode all packets from a collision, whereas the proposed algorithm detects the beginning of the transmission epochs in a collision using SIC. The third scheme is an unslotted RA system called IRA [39], which has been introduced in Section I-A. We use the user degree distribution $\Lambda_8(x) = 0.5x^2 + 0.28x^3 + 0.22x^8$, where $\Lambda_d x^d$ indicates that a user transmits $d$ replicas with probability $\Lambda_d$.

In Fig. 6(a), the throughput of the proposed ALOHA for $\Delta = 0.2$ is as good as that of IRA. As $\lambda$ increases, the throughput of IRA abruptly degrades, but the proposed scheme gracefully decreases. For $\Delta = 0.01$, the proposed scheme is also as good as SIC RA. As $\Delta$ becomes smaller than 0.2, the throughput of the proposed system becomes much higher than the IRA. In the IRA system, a user transmits $d$ replicas of the packet, including the packet itself, regardless of the success of the original packet. Thus, each user consumes $d - 1$ times more transmit power. However, users only retransmit when a retransmission is necessary upon a collision in the proposed system. Furthermore, in the IRA system, the users specify the location of each replica within frame length $T_f$. It is not only hard to implement it but also increases the overhead. Incorrect location information on other replicas or an error can destroy the overall SIC performance. However, our system does not need such an overhead, and the concept of $\Delta$-collision can absorb some timing errors. As the AP in the IRA system should store all the packets during multiple $T_f$s after the first replica transmission of each packet during one frame $T_f$, this requires huge memory. But our system does not need it.

Fig. 6(b) shows the average RA delay when the proposed ALOHA with SACR adopts the proposed backoff algorithm for various $\Delta$ values. In the genie-aided system, the average RA delay for each $\Delta$ grows unbounded as $\lambda$ goes to the maximum throughput found in Fig. 4(b). The proposed system is noted to have the RA delay performance close to that of the genie-aided system. On the other hand, the delay of IRA heavily depends on one frame length $T_f$, because the success of a packet transmission can be known at the end of $T_f$. Moreover, it is no larger than $T_f$, because if none of the transmitted copies for a packet is not successfully decoded during $T_f$, the packet is dropped. Compared to SIC RA, the proposed system shows a better RA delay as $\Delta$ gets smaller, e.g., $\Delta = 0.01$.

The transmission power consumption of devices is proportional to the number of (re)transmissions for successful packet transmission, denoted by $\overline{L}_s$. Fig. 6(c) illustrates $\overline{L}_s$. As opposed to the explosive increase in transmissions observed in IRA as $\lambda$ rises, the proposed scheme keeps retransmissions to no more than five.

## V. Conclusion

This work proposed a pure ALOHA system with SACR and a novel online backoff algorithm to maximize its throughput. In the proposed system, the users utilize a collision timer upon the initial transmission of their packet to measure the collision period. The AP's SIC can extract the information from each user's transmission epoch to the end of the collision period so that the users involved with the collision can estimate the order of retransmissions in the retransmission period by comparing their collision timer with the information. To make it work in practice, motivated by the technology capability demonstrated in [45], it was considered that the packets in collision are distinguishable by SIC when the transmission epochs of their packet are at least $\Delta$ apart. Our analysis showed that the maximum throughput asymptotically approaches one for a large population size as $\Delta \to 0$, i.e., ideal system. It was further shown that the proposed ALOHA system with SACR performs better than S-ALOHA if $\Delta$ is at most less than one-third of the packet transmission time. Although we proposed the collision resolution based on SIC, the key element of the MAC layer for the throughput improvement is the design of how to determine the retransmission sequence immediately after the collision period based on the users' collision timers and the information regarding the remaining collision period. As future work, we are interested in developing a similar ALOHA system using preamble detections.

## Appendix A
## Proof of Theorem 2

First, using the Poisson (arrival) process with the mean rate $N\beta = G$, we can write the idle and collision periods under the infinite population model:

$$\lim_{N\to\infty} \mathbb{E}[\mathcal{I}_k] = \frac{1}{G} \quad \text{and} \quad \lim_{N\to\infty} \mathbb{E}[\mathcal{C}_N] = \frac{e^{GT}-1}{G}. \tag{32}$$

In order to find the expressions of $\mathbb{E}[R_N]$ and $\mathbb{E}[W_N]$ in (12) as $N \to \infty$, let $\nu_i(l) = \lim_{N\to\infty} v_i(l)$ under the Poisson arrival assumption. We obtain $\nu_1(l)$ as

$$\begin{aligned}\nu_1(l) &= \lim_{N\to\infty} v_1(l) = \lim_{N\to\infty} q_l \mathcal{H}_1^{(l-1)} \\ &= \mathsf{q}\prod_{i=1}^{l-1}\mathsf{h} = \mathsf{q}\mathsf{h}^{l-1} \\ &= e^{-GT}\left(e^{-G\Delta} - e^{-GT}\right)^{l-1}, \end{aligned} \tag{33}$$

where we have used (26). Note that as $N \to \infty$, we have

$$\mathcal{H}_j^{(k)} \to \mathsf{h}^{k-j+1}. \tag{34}$$

We get $\nu_2(l)$ as

$$\begin{aligned}\nu_2(l) &= \lim_{N\to\infty} v_2(l) \\ &= \lim_{N\to\infty}\left[\sum_{k=0}^{l}\mathcal{H}_1^{(k)}\sum_{n=l+3}^{N}\left(g_{k+1}\prod_{l=k+2}^{n-l+k-2}(1-q_l)g_{n-l+k-1}\right.\right. \\ &\quad \left.\left. - F_{\tau(n-l-1)}(\Delta; N-k-1)\right)\mathcal{H}_{n-l+k}^{(n-1)}q_n\right] \\ &= \mathsf{q}\mathsf{h}^l\sum_{n=l+3}^{\infty}\left(\mathsf{g}^2(1-\mathsf{q})^{n-l-3} - \sum_{j=n-l-1}^{\infty}\Phi_{G\Delta}(j)\right) \\ &= \mathsf{q}(l+1)\mathsf{h}^l\left(\mathsf{g}^2\frac{1}{\mathsf{q}} - \sum_{j=2}^{\infty}(j-1)\Phi_{G\Delta}(j)\right) \\ &= (l+1)\mathsf{h}^l\left[\mathsf{g}^2 - \mathsf{q}\left(G\Delta + e^{-G\Delta} - 1\right)\right].\end{aligned}$$

We can also write $\nu_3(l)$ as

$$\begin{aligned}\nu_3(l) &= \lim_{N\to\infty} v_3(l) \\ &= \lim_{N\to\infty}\left[\mathcal{H}_1^{(l)}\sum_{n=l+2}^{N}F_{\tau(n-l-1)}(\Delta; N-l-1)q_n\right] \\ &= \mathsf{q}\mathsf{h}^l\sum_{n=l+2}^{\infty}\sum_{j=n-l-1}^{\infty}\Phi_{G\Delta}(j) \\ &= \mathsf{q}\mathsf{h}^l\sum_{j=1}^{\infty}\sum_{n=1}^{j}\Phi_{G\Delta}(j) = \mathsf{q}\mathsf{h}^l\sum_{j=1}^{\infty}j\Phi_{G\Delta}(j) = \mathsf{q}\mathsf{h}^l G\Delta,\end{aligned} \tag{35}$$

where we have used $\Phi_x(j) \triangleq \frac{x^j}{j!}e^{-x}$. Notice that $\nu_3(l) = G\Delta\mathsf{h}\nu_1(l)$. Additionally, $\nu_4(l)$ is found as

$$\begin{aligned}\nu_4(l) &= \lim_{N\to\infty} v_4(l) \\ &= \lim_{N\to\infty}\left[\sum_{k=0}^{l-1}\mathcal{H}_1^{(k)}\sum_{n=l+2}^{N}F_{\tau(n-l-1)}(\Delta; N-k-1)\right. \\ &\quad \left.\times\mathcal{H}_{n-l+k}^{(n-1)}q_n\right] \\ &= \mathsf{q}\sum_{k=0}^{l-1}\mathsf{h}^k\sum_{n=l+2}^{\infty}\sum_{j=n-l-1}^{\infty}\Phi_{G\Delta}(j)\mathsf{h}^{l-k} = l\mathsf{q}\mathsf{h}^l G\Delta.\end{aligned}$$

We can find that $\nu_4(l) = l\nu_3(l)$. We can find

$$\begin{aligned}\lim_{N\to\infty}\mathbb{E}[V_N] &= \sum_{l=2}^{\infty}l\nu_1(l) + \sum_{l=0}^{\infty}l\nu_2(l) + \sum_{l=0}^{\infty}l\nu_3(l) + \sum_{l=1}^{\infty}l\nu_4(l) \\ &= \frac{2\mathsf{h}}{1-\mathsf{h}} - \frac{\mathsf{q}\mathsf{h}^2}{(1-\mathsf{h})^2}.\end{aligned} \tag{36}$$

Using (36), we get the average reward

$$\begin{aligned}\lim_{N\to\infty}\mathbb{E}[R_N] &= \lim_{N\to\infty}\mathbb{E}[V_N] + \mathsf{q} \\ &= \frac{\mathsf{q}(1-\mathsf{h})^2 + \mathsf{h}\left(2\mathsf{g} + \mathsf{q}(2-\mathsf{h})\right)}{(1-\mathsf{h})^2}.\end{aligned} \tag{37}$$

Additionally, we obtain

$$\begin{aligned}\lim_{N\to\infty}\mathbb{E}[W_N] &= \lim_{N\to\infty}\left[\sum_{l=2}^{N}lv_1(l) + \sum_{l=0}^{N-3}(l+2)v_2(l)\right. \\ &\quad \left. + \sum_{l=0}^{N-1}lv_3(l) + \sum_{l=1}^{N-2}(l+1)v_4(l)\right]T \\ &= \left[\sum_{l=2}^{\infty}l\nu_1(l) + \sum_{l=0}^{\infty}(l+2)\nu_2(l) + \sum_{l=1}^{\infty}(l+1)\nu_3(l)\right. \\ &\quad \left. + \sum_{l=1}^{\infty}(l+1)\nu_4(l)\right]T \\ &= \left[\frac{2\mathsf{g} + \mathsf{q}\mathsf{h}(2-\mathsf{h})(1+G\Delta) - 2\mathsf{q}G\Delta}{(1-\mathsf{h})^2}\right]T.\end{aligned} \tag{38}$$

We then get

$$\begin{aligned}&\lim_{N\to\infty}\mathbb{E}[\mathcal{Z}_k] \\ &= \frac{e^{GT}(1-\mathsf{h})^2 + GT\left[2\mathsf{g} + \mathsf{q}\mathsf{h}(2-\mathsf{h})(1+G\Delta) - 2\mathsf{q}G\Delta\right]}{G(1-\mathsf{h})^2},\end{aligned} \tag{39}$$

where we have used

$$\lim_{N\to\infty}\mathbb{E}[\mathcal{Z}_k] = \lim_{N\to\infty}\left(\mathbb{E}[\mathcal{I}_k] + \mathbb{E}[\mathcal{C}_N] + \mathbb{E}[W_N]\right). \tag{40}$$

Using (37) and (39), we have the result.

## Appendix B
## Derivations of Online Control Algorithm

The algorithm updates the estimation on the mean number of the backlogged users $\overline{m}$ at the end of each busy period. The AP's estimation is based on the observation on the length of the idle period prior to a busy period. When the system has $X = k$ backlogged users, the PDF of the idle period is expressed as

$$f_{\mathcal{I}}(t|X=k) = k\beta e^{-k\beta t}. \tag{41}$$

To begin with, let $X$ denote the number of backlogged users. Since the AP does not have this information *a priori*, it assumes that $X$ follows a Poisson distribution with mean $\overline{m}$:

$$\Pr[X=k] = \frac{\overline{m}^k}{k!}e^{-\overline{m}}, \tag{42}$$

which is called the *a prior* distribution. Once the duration of an idle period is over, the AP can get the a posterior distribution of $X$ as follows.

$$\Pr[X=k|\mathcal{I}=t] = \frac{f_{\mathcal{I}}(t, X=k)}{f_{\mathcal{I}}(t)} = \frac{(\overline{m}e^{-\beta t})^{k-1}}{(k-1)!}e^{-\overline{m}e^{-\beta t}}. \tag{43}$$

It is noteworthy that the a posteriori distribution in (43) is also a Poisson distribution with mean $\overline{m}e^{-\beta t}$. Then, the conditional expectation of $X$ given idle time period $t$ is obtained as

$$\begin{aligned}\mathbb{E}[X|\mathcal{I}=t] &= \sum_{k=0}^{\infty} k \Pr[X=k|\mathcal{I}=t] \qquad (44)\\ &= \frac{\sum_{k=0}^{\infty} k \Pr[X=k, \mathcal{I}=t]}{f_{\mathcal{I}}(t)} = 1+\overline{m}e^{-\beta t}.\end{aligned}$$

More details on (43) and (44) are found in [60], where the algorithm for the traditional ALOHA was developed. There are two differences between Algorithm 1 and the one in [60]: The first one is the number of packets successfully retransmitted and decoded $V$ in line 7 and the second one is the constant $\kappa$ in line 10.

Finally, let us return to line 5 of Algorithm 1: At the end of a single packet transmission, i.e., $\hat{\mathcal{B}}_s$, the AP needs to subtract one user from the estimated backlog, i.e., $1+\overline{m}e^{-\beta t}$ and add the number of the packets newly joining to the backlog for $\hat{\mathcal{B}}_s$; that is $\lambda \cdot \hat{\mathcal{B}}_s$. In line 7, $V$ is subtracted from the estimated backlog since $V$ users transmit successfully at the end of the retransmission period. Subsequently, the number of new packet arrivals during $\hat{\mathcal{B}}_c$ is added.

## References


[1] J. P. S. Sundaram, W. Du, and Z. Zhao, “A survey on LORA networking: research problems, current solutions, and open issues,” *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 371-387, 1st Quart., 2020.

[2] D. Bertsekas and R. Gallager, *Data Networks*, 2nd Ed., Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992.

[3] S. Ghez, S. Verdu, and S. C. Schwartz, “Stability properties of slotted Aloha with multipacket reception capability,” *IEEE Trans. Autom. Control*, vol. 33, no. 7, pp. 640-649, Jul. 1988.

[4] L. Tong, Q. Zhao, and G. Mergen, “Multipacket reception in random access wireless networks: from signal processing to optimal medium access control,” *IEEE Commun. Mag.*, vol. 39, no. 11, pp. 108–112, 2001.

[5] G. Bartoli, N. C. Beaulieu, R. Fantacci, and D. Marabissi, “An effective multiuser detection scheme for MPR random access networks,” IEEE Trans. Commun., vol. 65, no. 3, pp. 1119-1130, Mar. 2017.

[6] E. Casini, R. De Gaudenzi, and O. del Rio Herrero, “Contention resolution diversity slotted aloha (CRDSA): an enhanced random access scheme for satellite access packet networks,” *IEEE Trans. Wireless Commun.*, vol. 6, no. 4, pp. 1408–1419, Apr. 2007.

[7] G. Liva, “Graph-based analysis and optimization of contention resolution diversity slotted aloha,” *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 477–487, Feb. 2011.

[8] Č. Stefanović, P. Popovski, and D. Vukobratovic, “Frameless Aloha protocol for wireless networks” *IEEE Commun. Lett.*, vol. 16, no. 12, pp. 2087–2090, Dec. 2012.

[9] Č. Stefanović, and P. Popovski, “Aloha random access that operates as a rateless code,” *IEEE Trans. Commun.*, vol. 61, no. 11, pp. 4653-4662, Nov. 2013.

[10] E. Paolini, G. Liva, and M. Chiani, “Coded slotted Aloha: a graph-based method for uncoordinated multiple access,” *IEEE Trans. Inf. Theory*, vol. 61, no. 12, pp. 6815–6832, Dec. 2015.

[11] S. Alvi, S. Durrani, and X. Zhou, ”Enhancing CRDSA with transmit power diversity for machine-type communication,” *IEEE Trans. Vehicular Technol.*, vol. 67, no. 8, pp. 7790-7794, Aug. 2018.

[12] C. Dumas, L. Salaün, I. Hmedoush, C. Adjih, and C. S. Chen, “Design of coded slotted ALOHA with interference cancellation errors,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 12, pp. 12742-12757, Dec. 2021.

[13] S. -L. Shieh, and S. -H. Yang, “Enhanced irregular repetition slotted Aloha under sic limitation,” *IEEE Trans. Commun.*, vol. 70, no. 4, pp. 2268-2280, Apr. 2022.

[14] Z. Sun, Y. Xie, J. Yuan, and T. Yang, “Coded slotted ALOHA for erasure channels: design and throughput analysis,” *IEEE Trans. Commun.*, vol. 65, no. 11, pp. 4817–4830, Nov. 2017.

[15] Z. Zhang, K. Niu, and J. Dai, “Performance bounds of coded slotted Aloha over erasure channels,” *IEEE Trans. Vehicular Technol.*, vol. 71, no. 11, pp. 12338-12343, Nov. 2022.

[16] J. Choi, “NOMA-based random access with multichannel Aloha,” *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2736–2743, Dec. 2017.

[17] J.-B. Seo, B. C. Jung, and H. Jin, “Performance analysis of NOMA random access,” *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2242-2245, Aug. 2018.

[18] J.-B. Seo, B. C. Jung, and H. Jin, “Non-orthogonal random access for 5G networks,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7867-7871, Aug. 2018.

[19] S. A. Tegos, P. D. Diamantoulakis, A. S. Lioumpas, P. G. Sarigiannidis, and G. K. Karagiannidis, “Slotted Aloha with NOMA for the next generation IoT,” *IEEE Trans. Commun.*, vol. 68, no. 10, pp. 6289-6301, Oct. 2020.

[20] A. A. Tegos, S. A. Tegos, D. Tyrovolas, P. D. Diamantoulakis, P. Sarigiannidis, and G. K. Karagiannidis, “Breaking orthogonality in uplink with randomly deployed sources,” *IEEE Open J. Commun. Soc.*, vol. 5, pp. 566-582, 2024.

[21] J. F. Grybosi, J. L. Rebelatto, and G. L. Moritz, “Age of Information of SIC-aided massive IoT networks with random access,” *IEEE Internet of Things J.*, vol. 9, no. 1, pp. 662-670, Jan., 2022.

[22] Y. Liang, X. Li, J. Zhang, and Z. Ding, “Non-orthogonal random access for 5G networks,” *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4817–4831, Jul. 2017.

[23] Y. Wang, T. Wang, Z. Yang, D. Wang, and J. Cheng, “Throughput-oriented non-orthogonal random access scheme for massive MTC networks,” *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1777–1793, Mar. 2020.

[24] X. Shao, Z. Sun, M. Yang, S. Gu, and Q. Guo, “NOMA-based irregular repetition slotted ALOHA for satellite networks,” *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 624-627, Apr. 2019.

[25] F. Babich, G. Buttazzoni, F. Vatta, M. Comisso, “Energy-constrained uncoordinated multiple access for next-generation networks”, *IEEE Open J. Commun. Soc.*, vol.1, pp.1808-1819, 2020.

[26] Y. -C. Huang, S. -L. Shieh, Y. -P. Hsu, and H. -P. Cheng, “Iterative collision resolution for slotted Aloha with NOMA for heterogeneous devices,” *IEEE Trans. Commun.*, vol. 69, no. 5, pp. 2948-2961, May 2021.

[27] A. Mengali, R. De Gaudenzi, and Č. Stefanović, “On the modeling and performance assessment of random access with SIC,” *IEEE J. Selected Areas Commun.*, vol. 36, no. 2, pp. 292-303, Feb. 2018.

[28] Y. Yu and G. B. Giannakis, “High-throughput random access using successive interference cancellation in a tree algorithm,” *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4628-4630, Dec. 2007.

[29] X. Wang, Y. Yu, and G. B. Giannakis, “A robust high-throughput tree algorithm using successive interference cancellation,” *IEEE Trans. Commun.*, vol. 55, no. 12, pp. 2253-2256, Dec. 2007.

[30] R. Yim, N. B. Mehta, A. F. Molish, and J. Zhang, “Dual power multiple access with multipacket reception using local CSI,” *IEEE Trans. Wirel. Commun.*, vol. 8, no. 8, pp. 4078-4088, Aug. 2009.

[31] A. D. Gore and A. Karandikar, “Power-controlled FCFS splitting algorithm for wireless networks,” *IEEE Trans. Veh. Technol.*, vol. 59, no. 2, pp.842–856, Feb. 2010.

[32] S.-W. Jeon and H. Jin, “Online estimation and adaptation for random access with successive interference cancellation,” *IEEE Trans. Mob. Comput.*, vol. 22, no. 9, pp. 5418-5433, Sept. 2023.

[33] Č. Stefanović, Y. Deshpande, H. M. Gürsu, and W. Kellerer, “Tree-algorithms with multi-packet reception and successive interference cancellation,” *IEEE Trans. Commun.*, vol. 71, no. 11, pp. 6287-6300, Nov. 2023.

[34] I. Cidon, and M. Sidi. “Conflict multiplicity estimation and batch resolution algorithms,” *IEEE Trans. Inf. Theory*, vol. 34, no.1, pp. 101-110, Jan. 1988.

[35] P. Popovski, F. H. P. Fitzek, and R. Prasad. “Batch conflict resolution algorithm with progressively accurate multiplicity estimation,” *in Proceedings of the 2004 Joint workshop on Foundations of mobile computing*, pp. 31-40, Oct. 2004.

[36] A. Zanella, "Adaptive batch resolution algorithm with deferred feedback for wireless systems," *IEEE Trans. Wireless Commun.* vol. 11, no.10, pp. 3528-3539, Oct. 2012.

[37] O. Del Rio Herrero, and R. De Gaudenzi, "High efficiency satellite multiple access scheme for machine-to-machine communications," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 4, pp. 2961-2989, Oct. 2012.

[38] R. De Gaudenzi, O. Del Río Herrero, G. Acar, and E. Garrido Barrabés, "Asynchronous contention resolution diversity ALOHA: making CRDSA truly asynchronous," *IEEE Trans. Wireless Commun.*, vol. 13, no. 11, pp. 6193–6206, Nov. 2014.

[39] T. Akyıldız, U. Demirhan and T. M. Duman, "Energy harvesting irregular repetition Aloha with replica concatenation," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 955-968, Feb. 2021.

[40] F. Clazzer, C. Kissling and M. Marchese, "Enhancing contention resolution ALOHA using combining techniques," *IEEE Trans. Commun.*, vol. 66, no. 6, pp. 2576-2587, Jun. 2018.

[41] J. M. de Souza Sant'Ana, O. d. S. Neto, A. Hoeller, J. L. Rebelatto, R. D. Souza, and H. Alves, "Asynchronous contention resolution-aided ALOHA in LR-FHSS networks," *IEEE Internet of Things J.*, vol. 11, no. 9, pp. 16684-16692, May, 2024.

[42] I. B. Arun, and T. G. Venkatesh, "Order statistics based analysis of pure ALOHA in channels with multipacket reception," *IEEE Commun. Letters*, vol. 17, no. 10, pp. 2012-2015, Oct. 2013.

[43] A. Baiocchi, and F. Ricciato, "Analysis of pure and slotted ALOHA with multi-packet reception and variable packet size," *IEEE Commun. Letters*, vol. 22, no. 7, pp. 1482-1485, Jul. 2018.

[44] H. Wang, and A. O. Fapojuwo, "Design and performance evaluation success interference cancellation-based pure aloha for internet-of-things networks," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6578-6592, Aug. 2019.

[45] S. Tong, Z. Xu, and J. Wang, "CoLoRa: enabling multi-packet reception in LoRa networks," *IEEE Tran. Mob. Comput.*, vol. 22, no. 6, pp. 3224–3240, Jun. 2023.

[46] X. Wang, L. Kong, L. He and G. Chen, "mLoRa: A Multi-Packet Reception Protocol in LoRa networks," *IEEE 27th International Conference on Network Protocols (ICNP)*, 2019, pp. 1-11, doi: 10.1109/ICNP.2019.8888038.

[47] L. You, Z. Tang, P. Wang, Z. Wang, H. Dai, and L. Fu, "Quick and reliable LoRa data aggregation through multi-packet reception," *IEEE/ACM Trans. Network.*, vol. 32, no. 2, pp. 1616-1630, April 2024.

[48] G. Choudhury, and S. Rappaport, "Diversity ALOHA - a random access scheme for satellite communications," *IEEE Trans. Commun.*, vol. 31, no. 3, pp. 450-457, Mar. 1983.

[49] Y. Birk, and Y. Keren, "Judicious use of redundant transmissions in multichannel ALOHA networks with deadlines," *IEEE J. Select. Areas Commun.*, vol. 17, no. 2, pp. 257-269, Feb. 1999.

[50] N. Abramson, "Multiple access in wireless digital networks," *Proc. IEEE*, vol. 82, no. 9, pp. 1360-1370, Sept. 1994.

[51] T. Sato, H. Okada, T. Yamazato, M. Katayama, and A. Ogawa, "Throughput analysis of DS/SSMA unslotted ALOHA system with fixed packet length," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 750–756, May 1996.

[52] J.-W. So, Il Han, B.-C. Shin, and D.-H. Cho, "Performance analysis of DS/SSMA unslotted ALOHA system with variable length data traffic," *IEEE J. Selec. Areas Commun.*, vol. 19, no. 11, pp. 2215-2224, Nov. 2001.

[53] M. Heusse, C. Caillouet, and A. Duda, "Performance of unslotted ALOHA with capture and multiple collisions in LoRaWAN," in IEEE Internet of Things Journal, vol. 10, no. 20, pp. 17824-17838, 15 Oct.15, 2023.

[54] M. Zorzi, and R. Rao, "Capture and retransmission control in mobile radio," *IEEE J. Sel. Areas Commun.*, pp. 1289–1298, Oct. 1994.

[55] A. Zanella, and M. Zorzi, "Theoretical analysis of the capture probability in wireless systems with multiple packet reception capabilities," *IEEE Trans. Commun.*, vol. 60, no. 4, pp. 1058-1071, Apr. 2012.

[56] F. Babich, and M. Comisso, "Impact of segmentation and capture on slotted Aloha systems exploiting interference cancellation," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2878-2892, Mar. 2019.

[57] Y. Hu, H. Jin, and J. -B. Seo, "Asynchronous random access systems with immediate collision resolution for low power wide area networks," *IEEE Trans. Vehicular Technol.*, vol. 73, no. 2, pp. 2755-2770, Feb. 2024.

[58] S. M. Ross, *Stochastic Processes*, 2nd ed., John Wiley & Sons, Inc.

[59] J.-B. Seo, Y. Hu, S.-H. Pack, and H. Jin, "Comprehensive throughput analysis of unslotted Aloha for low power wide area networks," *IEEE Internet Things J.*, vol. 9, no. 17, pp. 15800-15813, Sep. 2022.

[60] J.-B. Seo, B.-C. Jung, and H. Jin, "Modeling and online adaptation of ALOHA for low power wide area networks (LPWAN)," *IEEE Internet of Things Journal*, vol. 8, no. 20, pp. 15608-15619, Oct. 2021.

[61] J.-B. Seo, S. De, and S.-Y. Kim, "Half-duplex Aloha systems for low power wide area networks," *in Proc. of IEEE Vehicular Technology Conference*, Jul. 2018.

[62] A. Ephremides, and B. Hajek, "Information theory and communication networks: An unconsummated union," *IEEE Trans. Inf. theory*, vol. 44, no. 6, pp. 2416-2434, Oct. 1998.