Hash Access in Blockchain Radio Access Networks: Characterization and Optimization

Xintong Ling, *Member, IEEE*, Bowen Zhang, Hui Xie, Jiaheng Wang, *Senior Member, IEEE*, and Zhi Ding, *Fellow, IEEE*

***Abstract*—Blockchain radio access network (B-RAN) is a decentralized, trustworthy wireless networking paradigm spurred by distributed ledger technologies (DLTs). In B-RAN, even though the blockchain builds trust in upper layers, the absence of trust between client devices still causes the problem with open access, or the so-called Rogue's dilemma, and degrades the network performance. Therefore, Hash Access was proposed for B-RAN to address the trust issue between clients and enforce client devices to obey the grant-free access rule. However, the characteristics and performance of Hash Access in B-RAN remain unclear. In this work, we dive deep into the Rogue's dilemma from a game-theoretic model to emphasize the necessity of Hash Access. We establish an analytical model to comprehensively evaluate the performance of B-RAN using Hash Access regarding transmission success probability, access delay, and network throughput. Based on the analytical model, we further optimize the Hash Access protocol for network throughput and provide useful practical guidelines. Simulation results are presented to validate our proposed model and insights.**

***Index Terms*—Blockchain, Internet of Things (IoT), machine-to-machine communications, medium access control protocol, radio access network (RAN).**

I. Introduction

BLOCKCHAIN radio access network (B-RAN) is emerging as a decentralized, secure, and efficient wireless access paradigm by leveraging distributed ledger technologies (DLTs) for the upcoming sixth-generation (6G) era [1]–[5]. By establishing a trusted foundation, B-RAN facilitates collaboration among multiple parties and provides a unified framework for heterogeneous, distributed, and complex future Internet of Things (IoT) [6]–[8]. In B-RAN, the IoT devices are not limited to any specific subscribing service providers (SPs) but can receive resources and services from different subnetworks. With the help of blockchain and related critical components, B-RAN connects multiple untrusted parties, including SPs and clients and further forms a multisided platform (MSP) that enables direct interactions among them [1].

However, it is worth emphasizing that the process of establishing trust is not easy [1], [9]–[14]. That is why B-RAN requires a series of designs and modules, more than a simple blockchain, to work together through proper orchestration for a trustworthy environment [1], [15]. As highlighted in [1], there are several fundamental trust relationships, such as SP–client, SP–SP, and client–client, in such an interhost network platform. Among them, the trust between client devices is often overlooked. However, as stressed in [16], the lack of trust between the client devices may lead to an interesting phenomenon, named the Rogue's Dilemma.

To see the Rogue's Dilemma, let us consider the grant-free random access (GFRA) scenario, where a massive number of IoT devices share an open access medium without dedicated resources [17]–[21]. Due to the short packets with typically tens of bytes, GFRA enables efficient management with low signaling overhead and access delay and thus is favored. Traditionally, if all the devices either belong to a common operator or are shared authentication, they will strictly obey the random access (RA) protocols for the entire network's performance. For example, in the well-known Aloha protocol [22], [23], a random backoff is required before a packet transmission, to reduce the collision probability of the whole network. However, unlike such traditional networks, B-RAN includes a massive number of devices that may belong to multiple parties [1], [16]. These client devices do not trust each other and thus may ignore predefined protocols and compete for limited resources for self-interest. A rogue device may simply skip the required backoff to shorten the waiting time in its best interest; however, such behaviors harm honest device's interests, especially when the number of rogue devices increases. As illustrated in [16], even a few selfish devices may significantly compromise and undermine network efficiency and fairness. Clearly, the Rogue's Dilemma is directly caused by the absence of trust between clients. It can be viewed as the problem with open access or "the tragedy of the commons" [24] in the context of wireless communications.

Manuscript received March 16, 2021; revised June 23, 2021 and August 3, 2021; accepted August 27, 2021. Date of publication September 13, 2021; date of current version May 23, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1801103; in part by the National Natural Science Foundation of China under Grant 61901111, Grant 61971130, and Grant 61720106003; in part by the Natural Science Foundation of Jiangsu Province under Grant BK20190331; in part by the Jiangsu Province Basic Research Project under Grant BK20192002; in part by the Huawei Cooperation Project under Grant FA 2019051081-2021-01; and in part by the Fundamental Research Funds for the Central Universities. *(Corresponding author: Jiaheng Wang.)*

Xintong Ling and Jiaheng Wang are with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China, and also with the Pervasive Communication Research Center, Purple Mountain Laboratories, Nanjing 210023, China (e-mail: xtling@seu.edu.cn; jhwang@seu.edu.cn).

Bowen Zhang and Hui Xie are with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: bwzhang@seu.edu.cn; huixie@seu.edu.cn).

Zhi Ding is with the Department of Electrical and Computer Engineering, University of California at Davis, Davis, CA 95616 USA (e-mail: zding@ucdavis.edu).

Digital Object Identifier 10.1109/JIOT.2021.3111915

The Rogue's Dilemma implies that even if blockchain builds trust in the upper layers of an integrated network, the coordination may still fail because of the lack of trust between clients at the bottom. In B-RAN, a trustworthy grant-free access protocol named Hash Access was proposed in [16] for IoT environments to resolve the Rogue's Dilemma. Hash Access can enforce all the devices to obey the rule of access and prevent selfish behaviors of rogue devices, and can ultimately establish trust between clients in B-RAN. As revealed in [16], Hash Access can promote multiparty cooperation through cross-network integration and unbalanced traffic offloading. More recently, several variants, such as nonorthogonal hash access (NOHA) [25], were proposed based on Hash Access for further enhancement.

Nevertheless, some fundamental problems of the original Hash Access protocol remain unknown yet. First, the Rogue's Dilemma is straightforward; however, it has never been analytically characterized yet. More evidence and analysis are thus required to provide insightful intuitions for the Rogue's Dilemma. Second, the work in [16] demonstrated the performance of Hash Access but mainly relied on simulations. So far, there is a lack of models and methods to assess the behavior and performance of Hash Access. Third, due to the absence of modeling, the optimization of parameters in Hash Access is still open. Consequently, a proper analytical model is urgently called for Hash Access in B-RAN to analyze the key network performance indicators and give insights and guidelines for practical designs.

This article is intended to address the above issues. First, we interpret the Rogue's Dilemma from a game-theoretic model. We provide the detailed workflow of Hash Access in the B-RAN framework and illustrate how Hash Access enforces rogue devices to obey. To evaluate the performance of B-RAN using Hash Access, we analyze the characteristics of Hash Access systematically, which is quite challenging. Eventually, we establish an analytical model for Hash Access and optimize the protocol based on the proposed model. The key contributions of this work are listed as follows.

1) We demonstrate the Rogue's Dilemma from a simplified two-player game instead of empirical or pure intuitions. Based on the game model, we provide several interesting insights and even qualify the value of trust. The Rogue's Dilemma explains the fundamental motivation why we need Hash Access in B-RAN.
2) We establish an analytical model for Hash Access in the framework of B-RAN with a general traffic pattern, which, to the best of our knowledge, has never been developed before.
3) Based on the above model, we comprehensively assess the performance of Hash Access in terms of transmission success probability, network throughput, and access delay and provide several insightful guidelines for designs.
4) In the scenario with sparse IoT traffic, we present the network performance in closed-form solutions and derive the optimal access difficulty for the maximum throughput. We also take the impact of IoT device's storage space into consideration.
5) In the simulation, we evaluate several critical aspects of Hash Access and validate our analytical model and conclusions.

The remainder of this article is organized as follows. Section II introduces the game model of the Rogue's Dilemma. Section III describes Hash Access in detail and presents the traffic model. Section IV establishes an analytical model for B-RAN using Hash Access. Section V considers Bernoulli traffic and optimizes the access difficulty. Section VI presents the simulation results, and Section VII draws the conclusion.

TABLE I
PAYOFF MATRIX

	T	W
T	$-c_p-c_t, -c_p-c_t$	$g-c_p, -c_t$
W	$-c_t, g-c_p$	$-c_t, -c_t$

II. ROGUE'S DILEMMA

In this section, we would like to explain the Rogue's Dilemma from a game-theoretic model. We consider the case with two IoT devices as players that share a common medium but do not trust each other. In every slot, each device has two strategies: 1) T for transmitting and 2) W for waiting, belonging to the strategy set $\mathcal{S} = \{T, W\}$. Denote $s_i \in \{T, W\}$ as the strategy adopted by device $i = 0, 1$. Assume that both of them have packets to transmit. If they both attempt to transmit in a slot, i.e., $s_0 = s_1 = T$, a collision occurs, and no one transmits their packets successfully. If both of them wait, an available slot is wasted, resulting in longer access delays for both sides. One transmits and the other one waits, leading to successful transmissions. Therefore, the payoff function of device $i = 0, 1$ can be written as

$$u_i(s_i, s_{1-i}) = \begin{cases} g - c_p, & s_i = T, s_{1-i} = W \\ -c_t, & s_i = W \\ -c_p - c_t, & s_i = T, s_{1-i} = T \end{cases} \tag{1}$$

where g is the profit of one successful transmission, c_t is the waiting cost caused by transmission delays, and c_p is the transmission cost (e.g., power consumption). Compactly, we summarize (1) in the payoff matrix in Table I. Note that $g - c_p > -c_t$ always holds, implying one successful transmission is always better than waiting for a slot; otherwise, a successful access yields a negative payoff.

Based on the above game model, we first consider a trustworthy case where two players strictly follow a given RA protocol with access probability q. The utility functions can be written as

$$u_i(T) \triangleq E_{s_{1-i}}\{u_i(s_i = T, s_{1-i})\} = g - c_p - (g + c_t)q$$
$$u_i(W) \triangleq E_{s_{1-i}}\{u_i(s_i = W, s_{1-i})\} = -c_t.$$

Hence, the expected payoff of device i is

$$\begin{aligned} u_i &= E_{s_i, s_{1-i}}\{u_i(s_i, s_{1-i})\} \\ &= qu_i(T) + (1-q)u_i(W) \\ &= -(g + c_t)q^2 + \big(g - c_p + c_t\big)q - c_t. \end{aligned} \tag{2}$$

To maximize the gain, the transmit probability q of the RA protocol should be set to $q^\star = ([g - c_p + c_t]/[2(g + c_t)])$ for

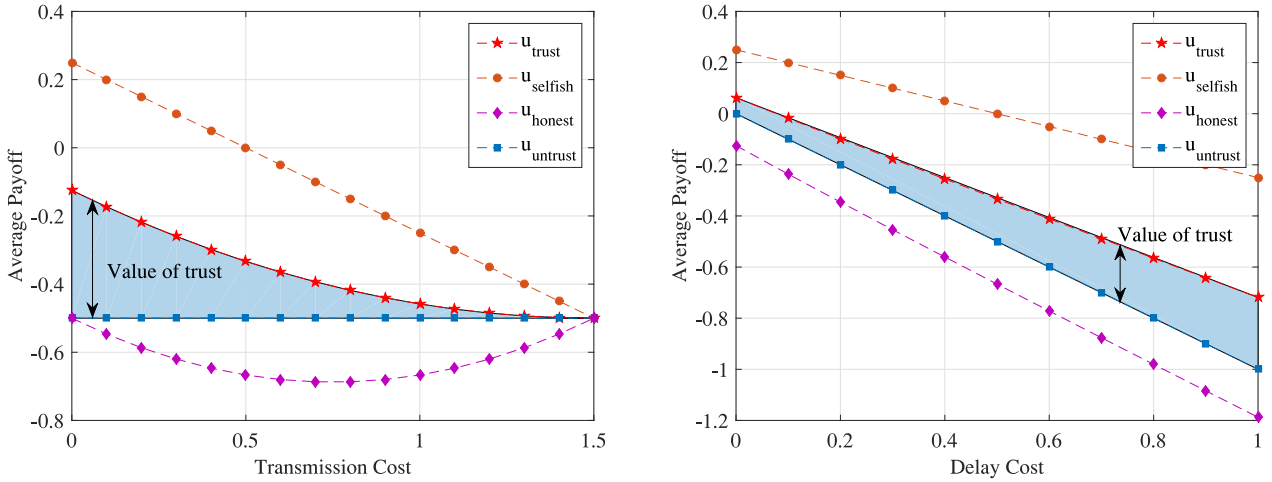


Fig. 1. Average payoffs of different strategies in the two-player game model. (The profit of a successful transmission is normalized as $g = 1$.) (a) $c_t = 0.5$. (b) $c_p = 0.5$.

both players, yielding

$$u_{\text{trust}} \triangleq \frac{(g + c_t - c_p)^2}{4(g + c_t)} - c_t. \tag{3}$$

Now, we consider the case that device 1 follows the RA protocol, i.e., $q_1 = q^\star$, and the rogue device 0 selfishly sets q_0 to maximize its own profit. The utility functions of device 0 are given by

$$u_0(T) = \frac{g - c_p - c_t}{2}$$
$$u_0(W) = -c_t.$$

The expected payoff of device 0 is

$$u_0 = \frac{g - c_p + c_t}{2} q_0 - c_t \tag{4}$$

which is maximized at $q_0 = 1$. In other words, a rogue device should always skip the backoff and occupy the link. The maximum payoff of device 0 is

$$u_{\text{selfish}} \triangleq \frac{g - c_p - c_t}{2}. \tag{5}$$

As a result, the expected payoff of device 1 becomes

$$u_{\text{honest}} \triangleq -\frac{g - c_p + c_t}{2(g + c_t)} c_p - c_t. \tag{6}$$

Obviously, the selfish strategy dominates the honest one.

Now, what if both devices behave selfishly? If both devices keep transmitting, it will tragically result in collisions in every slot, with payoff $(-c_p - c_t)$. Now, consider that they are smart enough to adjust their transmitting probability q_i based on the other's strategy q_{1-i} in their best interest. The network will converge to a Nash equilibrium at last in an uncoordinated manner. The utility function of device i is given by

$$u_i(T) = g - c_p - (g + c_t) q_{1-i}$$
$$u_i(W) = -c_t.$$

We calculate the mixed strategy equilibrium via the principle of indifference [26]. By letting $u_i(T) = u_i(W)$, we have

$$q_0 = q_1 = \frac{g - c_p + c_t}{g + c_t}. \tag{7}$$

The expected payoff is

$$u_{\text{untrust}} \triangleq -c_t. \tag{8}$$

In Fig. 1, we visualize the payoffs of different strategies in the game model with varying costs. Observing the above results in (3), (5), (6), and (8), we can obtain the relationship among them easily. Now, we formally summarize the Rogue's Dilemma based on the two-player game in the following.

Theorem 1 (Rogue's Dilemma): In an open access medium shared by two devices, the payoffs of different strategies have the following relationship:

$$u_{\text{honest}} \overset{(a)}{<} u_{\text{untrust}} \overset{(b)}{<} u_{\text{trust}} \overset{(c)}{<} u_{\text{selfish}} \tag{9}$$

indicating the situation where rogue behaviors for self-interests eventually result in low payoffs for everyone, even worse than the original trustworthy environment.

Proof: Inequality (a) is obvious since $g - c_p > -c_t$. Inequality (b) comes from the positivity of $([(g + c_t - c_p)^2]/[4(g + c_t)])$. Inequality (c) is because $u_{\text{selfish}} - u_{\text{trust}} = ([(g - c_p + c_t)(g + c_p + c_t)]/[4(g + c_t)]) > 0$. ■

Theorem 1 interprets how the Rogue's Dilemma occurs, and provides several interesting insights. First, if all the devices follow the access protocol honestly, the rogue behavior seems profitable to themselves due to the shorter delay and thus results in $u_{\text{selfish}} > u_{\text{trust}}$, but affects the honest device's interests ($u_{\text{honest}} < u_{\text{trust}}$). A device can easily conclude that it is in its best interests to skip the backoff and becomes a rogue device. That is how the selfish nature of devices comes. However, as more devices become rogues, all of them are losing out ($u_{\text{untrust}} < u_{\text{selfish}}$). The entire network will converge to an untrusted, uncoordinated Nash equilibrium with lower payoffs even than the original trusted state ($u_{\text{untrust}} < u_{\text{trust}}$).

Moreover, any single device that attempts to behave honestly will suffer by a lower payoff ($u_{\text{honest}} < u_{\text{untrust}}$) and cannot change the aggressive atmosphere. The gap between u_{trust} and u_{untrust}, i.e., $([(g + c_t - c_p)^2]/[4(g + c_t)])$, reflects the value of trust between clients, which is always positive. One can see that the relationships in Theorem 1 agree with the numerical results in Fig. 1. Essentially, the Rogue's Dilemma is the wireless communication version of "the tragedy of the commons." Even though Theorem 1 is based on a simplified two-player game model, the Rogue's Dilemma also holds for a network with multiple devices, which is verified by the simulations in Section VI.

Notably, B-RAN is a typical scenario, where the Rogue's Dilemma may occur. The rogue devices in B-RAN may violate the RA protocol and obtain gains in the short term; however, optimizing for the self in the short term is not optimal for all in the long term, and the cost of rogue behaviors is borne by all. The Rogue's Dilemma reveals the value of trust and stresses the necessity of establishing the trust between clients in B-RAN. As suggested by the Rogue's Dilemma, if we cannot guarantee every device to obey the protocol honestly, "the tragedy of the commons" could occur in B-RAN or any other shared open networks. That explains why Hash Access is indispensable.

III. Hash Access in B-RAN

A. Hash Access Protocol

With the help of blockchain and related designs, B-RAN integrates multilateral SPs and clients to form an interoperative network of subnetworks. As a trust builder, B-RAN pools diverse resources across sectors and connects network participants for multihost collaboration. Even so, the devices in B-RAN belonging to different parties without shared authentication could be dishonest or even selfish due to the absence of trust between them. Now, consider, in B-RAN, there are n_d untrusted and possibly selfish IoT devices sharing n_c open access channels. In other words, at most n_c IoT devices can access simultaneously. The value of n_c reflects the access capability of the shared network. As illustrated in Section II, the Rogue's Dilemma may occur in such a scenario. Therefore, we use Hash Access in B-RAN to avoid the Rogue's Dilemma caused by the lack of trust between devices.

According to the protocol of Hash Access, each IoT device shall perform hash queries before sending data packets until it finds a hash value below a given target value. Owing to the noninvertibility of the hash function, the access point (AP) can easily verify the validity of a solution, but it is difficult to find such a qualified solution. In this approach, an enforced backoff is automatically embedded into Hash Access, which can hardly be skipped. Different from the traditional RA protocol that simply assumes all the devices follow the protocol honestly, Hash Access enforces the protocol for every device and offers a feasible solution to the Rogue's Dilemma by rebuilding the trust between them.

Now, we demonstrate the Hash Access procedure in Fig. 2, with the detailed steps listed as follows.

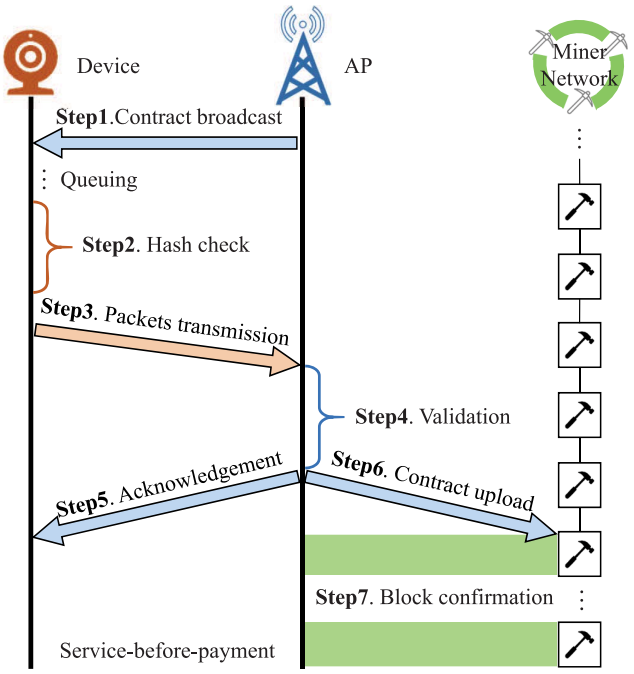


Fig. 2. Procedure of Hash access.

1) *Step 1 (Contract Broadcast):* In B-RAN, the AP periodically broadcasts access contracts to announce the availability of GFRA services. The access contracts are digitally signed by the AP and contain the service fee, a target hash, the timestamp, and control messages.
2) *Step 2 (Hash Check):* If a device has a packet to transmit, it starts to perform hash queries for access. The hash puzzle is formulated as

$$h = H(\mathcal{T} + \mathcal{I} + \mathcal{C}) \tag{10}$$

where h is the hash value, $H(\cdot)$ denotes a hash operation, $\mathcal{T}$ is the current timestamp, $\mathcal{I}$ is the device's unique identifier (ID),[1] and $\mathcal{C}$ is the access contract. This device uploads the packet, once the hash value is less than the given target hash, i.e.,

$$h = H(\mathcal{T} + \mathcal{I} + \mathcal{C}) < h_c \tag{11}$$

where h_c is the target hash value. Otherwise ($h \geq h_c$), this device is denied and has to wait and perform the hash query in the next slot (the timestamp is changing for each slot to keep the hash trials fresh and independent of each other). The device also needs to sign the smart contract $\mathcal{C}$ with its digital signature.
3) *Step 3 (Packet Transmission):* If a device passes the hash check, it transmits the data packet, the hash value, and the access contract to the AP in the current slot, and a contention resolution (CR) timer starts simultaneously. The data packet is successfully transmitted only if there is no packet collision.

[1]The unique identifier can rely on the existing ones, e.g., the international mobile equipment identity (IMEI), and also hardware-dependent features, e.g., RF fingerprinting [27], [28].

4) *Step 4 (Validation):* After receiving the data packets and the access contract from the IoT device, the AP verifies the hash value and the access contract.
5) *Step 5 (Acknowledgment):* If the hash value is below the target, the AP will accept the data packets and send the acknowledgment (ACK) messages to the device. If the hash value or the access contract is invalid, the AP will ignore the corresponding packet. If the IoT device does not receive the ACK message before the CR timer expires, it regards the transmission as a failure and repeats *step 2* and *step 3*.
6) *Step 6 (Contract Upload):* As the ACK message is transmitted, the AP uploads the access contract to the miner network for the service fee.
7) *Step 7 (Block Confirmation):* The miners check the validity of the access contract and then commit the contract to the blockchain. After the access contract is accepted by the main chain, the service fee specified in the contract will be automatically transferred from the IoT device to the AP through the blockchain.

Indeed, it is true that Hash Access requires extra computation and energy for hash operations. However, in step 2, we design the hash query (10) and restrict that each device can compute the hash value once per slot, which largely saves the energy cost of Hash Access. As shown in [16], the energy consumption of hash operations per second is acceptable compared with that of the data transmission.

In the process of Hash Access, the AP sets a suitable target hash value h_c to determine how hard it is for client devices to transmit data packets. More literally, we introduce a straightforward term named access difficulty to measure the difficulty of finding a hash value below the current target. Borrowed from the concept of mining difficulty in bitcoin [29], [30], the access difficulty d is defined as

$$d \triangleq \frac{h_m}{h_c} \tag{12}$$

where h_m represents the maximum target hash. According to the definition of the access difficulty, the transmission probability of an IoT device in one slot, i.e., the probability that each IoT device calculates a valid hash below the target hash, can be expressed as

$$\Pr\{h < h_c\} = \frac{h_c}{h_m} = \frac{1}{d}. \tag{13}$$

To show it more straightforward, we give an example where the target hash is a 16-bit number. The maximum target hash is set to 0xffff in the hexadecimal version. If the current hash target is 0x1027, the access difficulty d is

$$d = \frac{\text{0xffff}}{\text{0x1027}} \approx 15.85. \tag{14}$$

Obviously, a larger target hash indicates less access difficulty, whereas a smaller target hash yields greater access difficulty. In principle, at a higher difficulty level, it is more difficult to pass the hash check and thus leads to a longer waiting time, whereas, at a lower difficulty level, it is easier to find a valid hash solution but results in a higher collision probability. Therefore, the value of access difficulty d should be set accordingly for traffic control in B-RAN. An analytical model is thus called for assessing the characteristics of Hash Access and optimizing the access difficulty in the protocol.

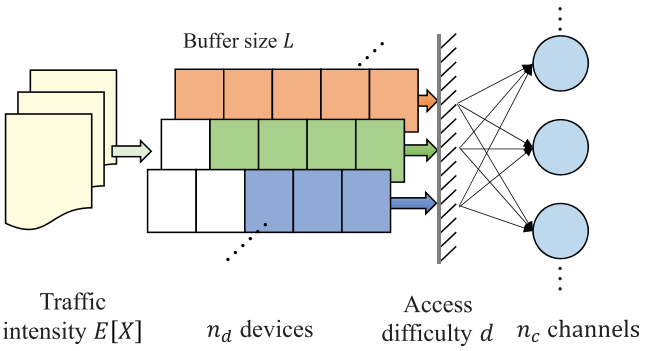


Fig. 3. Physical-layer model of B-RAN using Hash Access in GFRA scenarios.

B. Traffic Pattern

We adopt a general traffic model in this study. Every IoT device generates new packets following an independent, identical, stationary random process with an arbitrary probability mass function (PMF):

$$\Pr\{X = k\} = \xi_k,\ k = 0, 1, \ldots \tag{15}$$

where X is the random variable of newly generated packets within a slot duration τ_s for every IoT device. We define the expected value of $E[X]$ (packets/slot), or $E[X]/\tau_s$ (packets/s), as the offered load of an IoT device. We summarize the physical-layer model of B-RAN using Hash Access in GFRA scenarios in Fig. 3.

Based on the general traffic flow (15), we give several typical IoT traffic patterns as examples in the following.

1) *Bernoulli Traffic:* The arrivals of data packets follow a Bernoulli distribution [31] with probability ξ_1 for one packet and probability ξ_0 for none

$$\Pr\{X = k\} = \begin{cases} \xi_0, & k = 0 \\ \xi_1, & k = 1 \\ 0, & k = 2, 3, \ldots. \end{cases} \tag{16}$$

Bernoulli traffic implies sparse traffic, where at most one packet arrives in an IoT device in a slot. Hence, in the Bernoulli traffic, ξ_1 packet is expected to arrive in a slot.

2) *Sporadic Traffic:* If each IoT device sporadically generates packets (two packets cannot arrive at the same epoch but may arrive in the same slot), we can use a Poisson process to model the IoT traffic [32], [33] with the PMF

$$\Pr\{X = k\} = \xi_k = \frac{e^{-\lambda}(\lambda)^k}{k!},\ k = 0, 1, \ldots \tag{17}$$

where $\lambda \triangleq E[X]$ is defined as the average number of arrival packets in a slot.

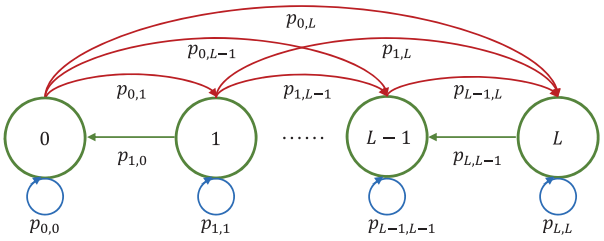


Fig. 4. State transition diagram of the proposed Markov chain.

IV. Performance Analysis of Hash Access

A. Markov Model

In this section, we will establish a Markov model to characterize the performance of Hash Access. Let the state of the Markov chain be the length of each IoT device's packet buffer, i.e., the number of data packets awaiting their turn to transmit in each device. Since an IoT device's storage space is limited, we set the maximum buffer size to be L such that the state space $\mathcal{Q}$ of an IoT device is given by $\mathcal{Q} = \{0, 1, \ldots, i, \ldots, L\}$. Define $\pi_i(t)$ as the probability that there are i packets in the buffer at time t for $i = 0, 1, \ldots, L$, and denote $\pi_i \triangleq \lim_{t\to\infty}\pi_i(t)$ as the steady-state probability of state i. The state transition diagram of Hash Access is shown in Fig. 4, where $p_{j,k}$ is the state transition probability from state j to state k. A state transition occurs as new packets' arrive or transmissions succeed (we assume that the ACK message can always be captured successfully before the CR timer expires). Mathematically, the state transition probability of the Markov chain can be represented as

$$p_{j,k} = \begin{cases} \xi_{k-j}\left(1-\frac{p_s}{d}\right) + \xi_{k-j+1}\frac{p_s}{d}, & 1 \le j \le k \le L-1 \\ \xi_0\frac{p_s}{d}, & j = 1, \ldots, L; k = j-1 \\ \sum_{i=k}^{\infty}\xi_{i-j}\left(1-\frac{p_s}{d}\right) & \\ \quad + \sum_{i=k}^{\infty}\xi_{i-j+1}\frac{p_s}{d}, & j = 1, \ldots, L-1; k = L \\ \xi_k, & j = 0; k = 0, \ldots, L-1 \\ \sum_{i=k}^{\infty}\xi_i, & j = 0; k = L \\ 1-\xi_0\frac{p_s}{d}, & j = L; k = L \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

where p_s is the transmission success probability. We denote the one-step state transition matrix as $\mathbf{P}$, of which the (j, k) element is given by the state transition probability $p_{j,k}$ in (18). Now, we can characterize and express the stationary distribution $\boldsymbol{\pi} = [\pi_0, \pi_1, \ldots, \pi_L]$ under Hash Access as

$$\boldsymbol{\pi} = \boldsymbol{\pi}\mathbf{P}. \tag{19}$$

Given the traffic model $\{\xi_k\}$ and the maximum buffer size L, $\boldsymbol{\pi}$ in (19) can be expressed by an implicit function of p_s/d, denoted by $\boldsymbol{\pi}(p_s/d)$. Essentially, the term p_s/d represents the probability that a device solves the hash puzzle and transmits the packet successfully, implying the probability of successful access. Even so, we are still far from solving the steady-state distribution via (19) since matrix $\mathbf{P}$ includes the transmission success probability p_s which is unknown yet.[2]

[2]The steady-state distribution may not exist if $d = 1$ and $n_c = 1$. The following discussions do not consider this case.

Naturally, we shall derive the transmission success probability p_s in the following.

B. Transmission Success Probability

The packet transmission success probability p_s is defined as the probability of successfully transmitting a packet to the AP with no collision. Obviously, $1 - p_s$ represents the probability of packet collision. We denote π^+ as the probability that an IoT device attempts to access to the AP, i.e., it has at least one data packet to transmit

$$\pi^+\left(\frac{p_s}{d}\right) \triangleq \sum_{i=1}^{L}\pi_i\left(\frac{p_s}{d}\right) = 1 - \pi_0\left(\frac{p_s}{d}\right). \tag{20}$$

We can hardly give the closed-form expression of the stationary distribution $\boldsymbol{\pi}$. Nevertheless, by using Cramer's rule [34], we can obtain the expression of $\pi_0(p_s/d)$ from (19)

$$\pi_0\left(\frac{p_s}{d}\right) = \frac{\left(\frac{\xi_0 p_s}{d}\right)^L}{\det\left(\left[(\mathbf{I}-\mathbf{P})_{1:L-1}, \mathbf{1}\right]\right)}$$

where $[(\mathbf{I}-\mathbf{P})_{1:L-1}, \mathbf{1}]$ is the matrix formed by replacing the Lth column of $\mathbf{I}-\mathbf{P}$ by an all-one column vector $\mathbf{1}$. Hence, we have

$$\pi^+\left(\frac{p_s}{d}\right) = 1 - \frac{\left(\frac{\xi_0 p_s}{d}\right)^L}{\det\left(\left[(\mathbf{I}-\mathbf{P})_{1:L-1}, \mathbf{1}\right]\right)}. \tag{21}$$

Consequently, the probability that only one specific device collides with the current device is $(1/dn_c)\pi^+(p_s/d)$. In a network with n_d devices, the transmission success probability is

$$\begin{aligned} p_s &= \left(1 - \frac{\pi^+\left(\frac{p_s}{d}\right)}{dn_c}\right)^{n_d-1} \\ &= \left(1 - \frac{1}{dn_c}\left(1 - \frac{\left(\frac{\xi_0 p_s}{d}\right)^L}{\det\left(\left[(\mathbf{I}-\mathbf{P})_{1:L-1}, \mathbf{1}\right]\right)}\right)\right)^{n_d-1}. \end{aligned} \tag{22}$$

We can calculate the value of p_s based on (22) by numerical methods. For example, we can use the fixed-point method and update p_s iteratively from an initial point until convergence.

From (22), we would like to discuss the impact of network parameters on p_s. Given d, p_s is increasing in the number of links n_c and decreasing in the number of devices n_d. It is straightforward because more uplink channels mean stronger access capability and massive devices cause excessive collisions. Given n_d and n_c, p_s is an implicit function of the access difficulty d determined by (22) and thus can be denoted by $p_s(d)$. Hence, $\pi^+(p_s/d)$ as a function of p_s/d, is essentially determined by d, which will be denoted as $\pi^+(d)$ in the following. Meanwhile, if we simply look at (22), the relationship between p_s and d is unclear. In principle, a larger d means a fewer devices allowed to access, implying fewer collisions and a higher successful transmission probability p_s. The monotonicity is verified by numerical and simulation results (see Fig. 5); however, the strict proof under a general traffic model is not easy. Hence, we would like to make a conjecture first.

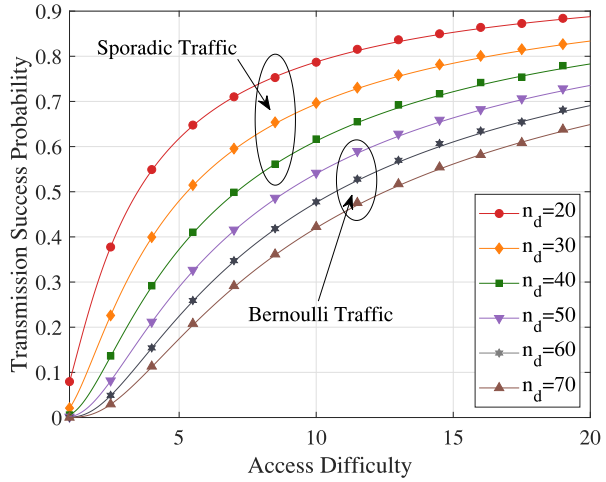


Fig. 5. Relationship between p_s and d with varying numbers of devices in different traffic patterns. ($n_c = 8$ and $E[X]/\tau_s = 40$packets/s).

Conjecture 1: Given n_c and n_d, p_s is monotonically increasing in d.

Section V will mathematically prove Conjecture 1 for the Bernoulli traffic as evidence of the general traffic pattern.

C. Network Throughput

Now, let us evaluate the network throughput of Hash Access, which is the critical performance measure in evaluating a network. We define the network throughput T as the average number of data packets successfully transmitted in a slot. The network throughput T is the product of the probability of successful transmission and the number of IoT devices that have packets to be transmitted in a slot, given by

$$\mathsf{T} = p_s(d) \cdot \frac{n_d \pi^+(d)}{d}. \tag{23}$$

According to (21) and (22), $p_s(d)$ and $\pi^+(d)$ are determined by the access difficulty d. Apparently, the access difficulty d plays a critical role in affecting the network throughput T.

Naturally, the access difficulty optimization for the network throughput arises. We formulate the optimization problem as

$$\begin{aligned} \underset{d}{\text{maximize}} \quad & \mathsf{T} \text{ in (23)} \\ \text{subject to} \quad & d \geq 1. \end{aligned} \tag{24}$$

The problem in (24) is challenging and intractable. We even do not have the closed-form expressions of $p_s(d)$ and $\pi^+(d)$, not to mention the objective T. However, by taking advantage of the problem structure, we have the following theorem.

Theorem 2: Given n_c and n_d, the network throughput is upper bounded by $\mathsf{T}_m = n_c(1 - 1/n_d)^{n_d - 1}$, and the optimal access difficulty $d^\star$ is given by

$$\begin{cases} p_s(d^\star) = \left(1 - \frac{1}{n_d}\right)^{n_d - 1}, & \text{if } p_s(1) \leq \left(1 - \frac{1}{n_d}\right)^{n_d - 1} \\ d^\star = 1, & \text{if } p_s(1) > \left(1 - \frac{1}{n_d}\right)^{n_d - 1}. \end{cases} \tag{25}$$

Proof: By substituting (22) into (23), the throughput can be rewritten as a function of p_s

$$\mathsf{T}(p_s) = n_c n_d p_s \left(1 - p_s^{\frac{1}{n_d - 1}}\right). \tag{26}$$

The first-order derivative of this function is given by

$$\mathsf{T}'(p_s) = n_c n_d \left(1 - \left(1 + \frac{1}{n_d - 1}\right) p_s^{\frac{1}{n_d - 1}}\right). \tag{27}$$

Hence, $\mathsf{T}(p_s)$ is monotonically increasing in p_s if $p_s < (1 - [1/n_d])^{n_d - 1}$ and is monotonically decreasing in p_s if $p_s > (1 - [1/n_d])^{n_d - 1}$. The maximum throughput is $n_c(1 - 1/n_d)^{n_d - 1}$, which is achieved at $p_s(d) = (1 - [1/n_d])^{n_d - 1}$. According to Conjecture 1, p_s is monotonically increasing in d and thus the minimum is $p_s(d) = p_s(1)$. Therefore, if $p_s(1) \leq (1 - [1/n_d])^{n_d - 1}$, we can always find an access difficulty d satisfying $p_s(d) = (1 - [1/n_d])^{n_d - 1}$. Meanwhile, if $p_s(1) > (1 - [1/n_d])^{n_d - 1}$, the access difficulty d satisfying $p_s(d) = (1 - [1/n_d])^{n_d - 1}$ does not exist. In this case, throughput T is decreasing in d, and the maximum is achieved at $d^\star = 1$. ∎

Theorem 2 indicates the optimal access difficulty $d^\star$ to maximize the network throughput T. As revealed by Theorem 2, the network has two states, depending on the value of $p_s(1)$. Since $p_s(d)$ is increasing in d, we have $p_s(1) \leq p_s(d) < 1$.[3] In the case $p_s(1) > (1 - [1/n_d])^{n_d - 1}$, we cannot find such d satisfying $p_s(d) = (1 - [1/n_d])^{n_d - 1}$ to achieve the upper bound T_m. In this case, the network throughput mainly depends on the offered traffic load, and the access channel is capable of handling the current traffic intensity. The optimal access difficulty $d^\star$ should be set to 1, suggesting that packets can be transmitted immediately once they arrive. This case corresponds to the scenario that the open access resources are sufficient for all the devices with packets to transmit so that backoff is unnecessary. Meanwhile, if $p_s(1) \leq (1 - [1/n_d])^{n_d - 1}$, we can always find proper $d^\star$ satisfying $p_s(d^\star) = (1 - [1/n_d])^{n_d - 1}$ for the upper bound T_m. Let us recall the Rogue's Dilemma in Section II. In this case, the network is saturated or even congested so that Hash Access is necessary to control the aggressive traffic and prevent selfish access.

In the saturated state using $d^\star$, both the transmission success probability $p_s(d^\star)$ and the maximum throughput T_m are monotonically decreasing in n_d. It accords with our intuition that more devices result in a higher collision probability and a lower throughput. In a network with massive devices $n_d \to \infty$, one finds that $\lim_{n_d \to \infty} \mathsf{T}_m = n_c(1 - 1/n_d)^{n_d - 1} = [n_c/e]$, which only depends on the number of uplinks n_c. Actually, the uplink channels represent the available resources of the network and determine the upper bound of the system performance. From Theorem 2, we always have $[n_c/e] < \mathsf{T}_m \leq [n_c/2]$ for any network with at least two devices.

[3] $\lim_{d \to +\infty} \pi^+(d)/d = 0$ yields $\lim_{d \to +\infty} p_s(d) = 1$, which is the supremum of $p_s(d)$.

D. Access Delay

Now, we would like to assess the delay of Hash Access based on the results of network throughput. According to Little's law [35], in a stable system, an item's waiting time equals the ratio of the average length of the waiting queue and the long-term average arrival rate (i.e., the throughput in our case). Hence, the access delay of Hash Access can be expressed as

$$\mathsf{D} = \frac{n_d \bar{L}}{\mathsf{T}} = \frac{d}{p_s(d)} \frac{\sum_{i=0}^{L} i\pi_i}{1-\pi_0} \tag{28}$$

where π_i is obtained from (19), $\bar{L} = \sum_{i=0}^{L} i\pi_i$ is the average packet number in a device's buffer, and the throughput T is given by (23). Equation (28) provides a general method to characterize the access delay of Hash Access for any traffic pattern described by (15). Remark that the access delay was previously discussed in [36] and derived by dividing into two parts: 1) the calculating delay and 2) queuing delay. However, Zhang *et al.* [36] assumed that the distribution (say Π_i) that the device buffer has i packets just prior to a packet arrival epoch[4] is the same as the stationary distribution π_i. However, such an assumption may not hold for any traffic patterns. Therefore, (28) provides a more accurate and general result for access delay analysis than the existing work [36].

As shown in Fig. 2, Hash Access is based on a service-before-payment mechanism to reduce the long confirmation delay caused by the blockchain. After receiving the packet from the IoT device, the AP directly provides the GFRA service and transmits the ACK messages to the IoT device if the hash value meets the requirement. Since Hash Access is grant-free, the IoT device does not have to wait for the access contract to be confirmed and accepted by the main chain, significantly reducing the access delay. The physical-layer safeguard mechanism guarantees that the AP cannot charge the service fee if the data packet is in collisions, since the data packet and the contract are packed together [16].

V. Hash Access With Bernoulli Traffic

A. Analytical Model

This section focuses on the Bernoulli traffic model, where $\xi_k = 0$ for any $k > 1$. In this case, at most one packet arrives in an IoT device in a slot, implying very sparse IoT traffic. The state transition probability is thus simplified as

$$p_{j,k} = \begin{cases} \xi_k, & j=0; k=0,1 \\ \xi_1\left(1-\frac{p_s}{d}\right), & j=1,\ldots,L-1; k=j+1 \\ \xi_0\left(1-\frac{p_s}{d}\right)+\xi_1\frac{p_s}{d}, & j=1,\ldots,L-1; k=j \\ 1-\xi_0\frac{p_s}{d}, & j=L; k=j \\ \xi_0\frac{p_s}{d}, & j=1,\ldots,L; k=j-1 \\ 0, & \text{otherwise.} \end{cases} \tag{29}$$

Now, the steady-state probabilities are expressed as

$$\begin{cases} \pi_0 = \pi_0 p_{0,0} + \pi_1 p_{1,0} \\ \pi_i = \pi_{i-1} p_{i-1,i} + \pi_i p_{i,i} + \pi_{i+1} p_{i+1,i}, \ i = 1,\ldots,L-1 \\ \pi_L = \pi_{L-1} p_{L-1,L} + \pi_L p_{L,L}. \end{cases} \tag{30}$$

[4]In other words, the distribution Π_i represents the viewpoint of the arriving packet.

Combining the above equations yields

$$\pi_i = \begin{cases} \pi_0, & i=0 \\ \pi_0 \prod_{j=0}^{i-1}\left(\frac{p_{j,j+1}}{p_{j+1,j}}\right), & i=1,\ldots,L. \end{cases} \tag{31}$$

We normalize the distribution $\{\pi_i\}$ and obtain

$$\pi_0 = \left(1 + \sum_{i=1}^{L}\prod_{j=0}^{i-1}\left(\frac{p_{j,j+1}}{p_{j+1,j}}\right)\right)^{-1}. \tag{32}$$

To simplify the expression, we introduce an auxiliary variable $a(p_s/d)$ as the ratio of $p_{j,j+1}$ and $p_{j+1,j}$

$$a\left(\frac{p_s}{d}\right) \triangleq \frac{p_{j,j+1}}{p_{j+1,j}} = \frac{\xi_1}{\xi_0}\left(\frac{d}{p_s}-1\right), \ j=1,\ldots,L-1. \tag{33}$$

Let us first consider the case $\xi_1 \neq (p_s/d)$ or equivalently $a \neq 1$. We can express the steady-state probabilities as

$$\pi_0 = \left(1 + \sum_{i=1}^{L}\frac{p_{0,1}}{p_{1,0}}a^{i-1}\right)^{-1} = \left(1 + \frac{\xi_1 d}{\xi_0 p_s}\frac{1-a^L}{1-a}\right)^{-1} \tag{34}$$

$$\pi_i = \left(1 + \frac{\xi_1 d}{\xi_0 p_s}\frac{1-a^L}{1-a}\right)^{-1}\frac{\xi_1 d}{\xi_0 p_s}a^{i-1}, i=1,\ldots,L. \tag{35}$$

Hence, $\pi^+(p_s/d) = 1-\pi_0$ yields

$$\pi^+\left(\frac{p_s}{d}\right) = \frac{1-a^L}{\frac{p_s}{d\xi_1}-a^L}. \tag{36}$$

If $\xi_1 = (p_s/d)$, i.e., $a = 1$, then we have the steady-state distribution as

$$\pi_0 = \left(1 + \sum_{i=1}^{L}\frac{p_{0,1}}{p_{1,0}}a^{i-1}\right)^{-1} = \frac{\xi_0}{L+\xi_0}$$

$$\pi_i = \frac{1}{L+\xi_0}, i=1,\ldots,L.$$

Hence

$$\pi^+ = \frac{L}{L+\xi_0}. \tag{37}$$

Compactly, we can write $\pi^+(p_s/d)$ as

$$\pi^+\left(\frac{p_s}{d}\right) = \begin{cases} \frac{1-a^L}{\frac{p_s}{d\xi_1}-a^L}, & \xi_1 \neq \frac{p_s}{d} \\ \frac{L}{L+\xi_0}, & \xi_1 = \frac{p_s}{d}. \end{cases} \tag{38}$$

Now, in the Bernoulli traffic, we have obtained the steady-state distribution in a more explicit form. This analytical model is helpful to the optimization of Hash Access.

B. Optimal Access Difficulty

In this section, we aim to find the optimal access difficulty to maximize the network throughput with the Bernoulli traffic. First, it seems that $\pi^+(p_s/d)$ is not a continuous function due to the discontinuity at $\xi_1 = (p_s/d)$; however, it is not true. In fact, as the value at $\xi_1 = (p_s/d)$ is specifically defined by $(L/[L+\xi_0])$, $\pi^+(d, p_s)$ is a continuous function, proved by the following proposition.

Proposition 1: $\pi^+(p_s/d)$ is a continuous function.

Proof: We only need to prove the function is continuous at $(p_s/d) = \xi_1$. From the definition of $a(p_s/d)$, we can easily

obtain $\lim_{(p_s/d)\to\xi_1} a = 1$. According to L'Hospital's rule, we have

$$\lim_{\frac{p_s}{d}\to\xi_1} \pi^+\left(\frac{p_s}{d}\right) = \frac{-La^{L-1}a'\left(\frac{p_s}{d}\right)}{\frac{1}{\xi_1} - La^{L-1}a'\left(\frac{p_s}{d}\right)} = \frac{L}{L+\xi_0} = \pi^+\left(\frac{p_s}{d}\right)|_{\frac{p_s}{d}=\xi_1}. \quad (39)$$

Therefore, $\pi^+(p_s/d)$ is a continuous function. ∎

Substituting (38) into (22) yields

$$p_s = \begin{cases} \left(1 - \frac{1-a^L}{dn_c\left(\frac{p_s}{d\xi_1} - a^L\right)}\right)^{n_d-1}, & \xi_1 \neq \frac{p_s}{d} \\ \left(1 - \frac{L}{dn_c(L+\xi_0)}\right)^{n_d-1}, & \xi_1 = \frac{p_s}{d}. \end{cases} \quad (40)$$

Obviously, $p_s(d)$ is also a continuous function. Given the access difficulty d, the transmission success probability $p_s(d)$ is determined by the implicit equation (40). In Section IV, we conjecture that p_s is increasing in d without strict proof. With the Bernoulli traffic, we can strictly prove the monotonicity of $p_s(d)$, summarized by the following theorem.

Theorem 3: Given n_c and n_d, p_s is monotonically increasing in $d \geq 1$.

Proof: First, we would like to prove $p_s(d)$ is injective, i.e., if $d_1 \neq d_2$, then $p_s(d_1) \neq p_s(d_2)$. We show this by contradiction. We assume that there exist $1 \leq d_1 < d_2$ but $p_s(d_1) = p_s(d_2) = \hat{p}_s$. We consider the case where $d \neq (\hat{p}_s/\xi_1)$ first. According to the expression of p_s in (40), we have

$$n_c\left(1 - \hat{p}_s^{\frac{1}{n_d-1}}\right) = \frac{1-a^L(d_1)}{\frac{\hat{p}_s}{\xi_1} - d_1a^L(d_1)} = \frac{1-a^L(d_2)}{\frac{\hat{p}_s}{\xi_1} - d_2a^L(d_2)}. \quad (41)$$

Now, we define

$$f(d) \triangleq \frac{1-a^L(d)}{\frac{\hat{p}_s}{\xi_1} - da^L(d)} - n_c\left(1 - \hat{p}_s^{\frac{1}{n_d-1}}\right). \quad (42)$$

$f(d)$ is undefined at the point $d = (\hat{p}_s/\xi_1)$. Redefine $f(d)|_{d=(\hat{p}_s/\xi_1)} = \lim_{d\to(\hat{p}_s/\xi_1)} f(d)$ so that $f(d)$ is continuous for any $d \geq 1$ and thus the case $d = (\hat{p}_s/\xi_1)$ is included. The derivative of $f(d)$ with respect to d is

$$f'(d) = -\frac{a^{L+1}(d) - a(d)(L+1) + L}{a^{1-L}(d)\left(\frac{\hat{p}_s}{\xi_1} - da^L(d)\right)^2} = \frac{g(a)}{a^{1-L}(d)\left(\frac{\hat{p}_s}{\xi_1} - da^L(d)\right)^2} \quad (43)$$

where $g(a) \triangleq -a^{L+1} + (L+1)a - L$. The derivative of $g(a)$ is $g'(a) = -(L+1)a^L + L + 1$, yielding

$$\begin{cases} g'(a) > 0, & 0 < a < 1 \\ g'(a) < 0, & a > 1. \end{cases}$$

Hence, $g(a)$ is upper bounded by $g(1) = 0$ and thus $g(a) < 0$ if $a \neq 1$. That means, $f'(d) < 0$ for $d \neq \hat{p}_s/\xi_1$ and $f'(d) = 0$ at $d = \hat{p}_s/\xi_1$. Therefore, $f(d)$ is monotonically decreasing in $d \geq 1$ (except the point $d = \hat{p}_s/\xi_1$ at which the derivation is 0). Due to the monotonicity of $f(d)$, if $f(d_1) = 0$, then we always have $f(d_2) < 0$ for any $d_2 > d_1$, implying $p_s(d_2) \neq \hat{p}_s$. This contradicts the assumption that $p_s(d_1) = p_s(d_2)$, indicating that $p_s(d)$ is injective. $p_s(d)$ is both continuous and injective and thus is strictly increasing or decreasing. By checking the derivation of $p_s(d)$ at some points, we can prove $p_s(d)$ is monotonically increasing in $d \geq 1$. ∎

Theorem 3 agrees with our intuition, but we should not take it for granted. In fact, even though the traffic pattern is specified, it is not easy to prove Theorem 3 (the monotonicity can hardly be proved directly and thus we use an alternative approach to complete the proof). Theorem 3 verifies Conjecture 1 as a piece of evidence with the Bernoulli traffic. As pointed out by Theorem 3, with a greater access difficulty, the devices are less likely to pass the hash check, resulting in a lower collision probability and a higher transmission success probability. Meanwhile, if the access difficulty is too large, very few devices can pass the hash check, resulting in negative effects on the network throughput.

Note that Theorem 3 is helpful for the access difficulty optimization problem. According to Section V-A, the network throughput can be expressed as

$$\mathsf{T} = \frac{n_d\pi^+\left(\frac{p_s}{d}\right)p_s(d)}{d} = \begin{cases} n_d\xi_1\frac{1-a^L}{1-\frac{d\xi_1}{p_s}a^L}, & \xi_1 \neq \frac{p_s}{d} \\ n_d\frac{\xi_1 L}{\xi_0+L}, & \xi_1 = \frac{p_s}{d}. \end{cases} \quad (44)$$

The throughput optimization problem is formulated the same as (24), and yields the same optimal solution to (25) given by Theorem 2. However, Theorem 2 and the optimality of $d^\star$ rely on the monotonicity of $p_s(d)$ based on Conjecture 1. From Theorem 3, we can guarantee the optimality of $d^\star$ with the Bernoulli traffic. Also, according to the monotonicity of $p_s(d)$, we can find the optimal $d^\star$ via, e.g., the bisection method.

In the above analysis, p_s/d plays an important role in Hash Access. Again, p_s/d represents the probability that a packet passes the hash check and is successfully transmitted with no collision, reflecting the network's access capability. If $\xi_1 \geq (p_s/d)$, the network is saturated and congested; otherwise, the network is capable of supporting the given offered traffic load. In the next section, we will provide more discussions on these two network statuses.

C. Infinite Buffer Size

This section considers the case that the buffer size is infinite or large enough. In this case, (38) can be written as

$$\lim_{L\to\infty} \pi_+\left(\frac{p_s}{d}\right) = \begin{cases} \frac{d\xi_1}{p_s}, & \xi_1 < \frac{p_s}{d} \\ 1, & \xi_1 \geq \frac{p_s}{d}. \end{cases} \quad (45)$$

In light traffic $\xi_1 < (p_s/d)$, we find that p_s is determined by the root of the following equation:

$$p_s = \left(1 - \frac{\xi_1}{p_s n_c}\right)^{n_d-1}. \quad (46)$$

Observe that the access difficulty d is not involved in (46). That means, no matter what the value of d is, the transmission success probability is constant in this case. We denote the root of (46) as p_s^*, indicating the transmission success probability in light traffic.

Substituting (45) into (44), we can obtain the expression of the throughput as

$$\mathsf{T} = n_d\xi_1.$$

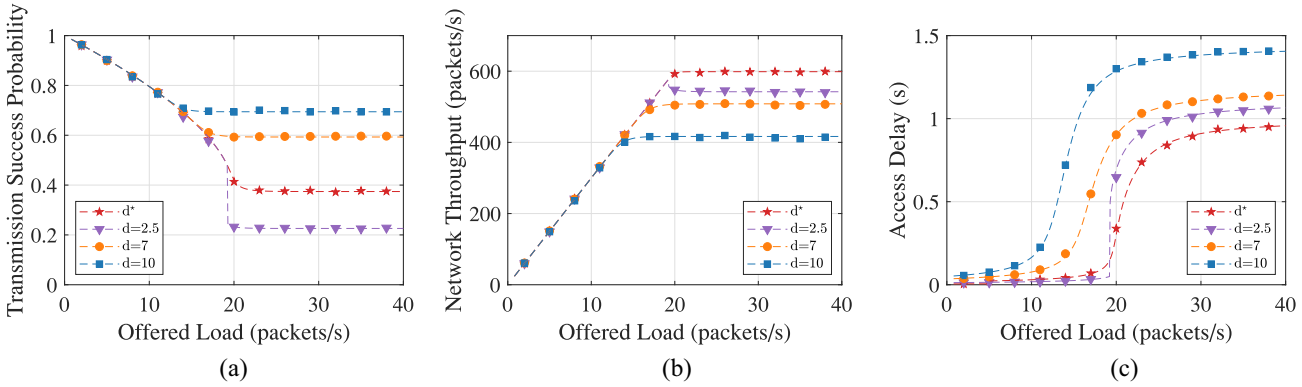


Fig. 6. Performance of Hash access in B-RAN with sporadic traffic. ($n_c = 8$ and $n_d = 30$). (a) Transmission success probability. (b) Network throughput. (c) Access delay.

Interestingly, in this case, the network throughput with low traffic is unrelated to any network parameters, such as p_s, d, and n_c. It is not surprising because the devices using infinite buffers would not reject any packet, and hence the network throughput merely depends on the offered load. Consequently, we can simply set the access difficulty d to 1, which means every device can transmit immediately if necessary. We name the case with $\xi_1 < p_s/d$ as the traffic-limited status, since the network throughput only depends on the traffic.

In heavy traffic $\xi_1 \geq (p_s/d)$, we have $\lim_{L\to\infty}\pi^+ = 1$, which means almost every device has packets to transmit. Substituting (45) into (22) yields

$$p_s = \left(1 - \frac{1}{dn_c}\right)^{n_d-1}. \tag{47}$$

Hence, the network throughput is given by

$$\mathsf{T} = \frac{n_d}{d}\left(1 - \frac{1}{dn_c}\right)^{n_d-1} \tag{48}$$

which is maximized at $\mathsf{T_m} = n_c(1 - [1/n_d])^{n_d-1}$ with $d^\star = (n_d/n_c)$. It is consistent with the results of Section V-B. Under high traffic $\xi_1 \geq (p_s/d)$, both p_s and T are irrelevant to the traffic intensity ξ_1. Therefore, we say the system is in the network-limited status. The access difficulty d has to be carefully selected and optimized, and the maximum throughput only depends on the network parameters n_c and n_d.

Now, we are interested in the exact value of the threshold traffic intensity, denoted by ξ_1^{th}. At the threshold, we have

$$\mathsf{T_m} = n_c\left(1 - \frac{1}{n_d}\right)^{n_d-1} = n_d\xi_1^{\text{th}}$$

yielding

$$\xi_1^{\text{th}} = \frac{n_c}{n_d}\left(1 - \frac{1}{n_d}\right)^{n_d-1}.$$

With the traffic intensity ξ_1^{th}, the transmission success probability $p_s = (1-[1/dn_c])^{n_d-1}$ given by (47), is also the solution of (46), which verifies the correctness of ξ_1^{th}. In conclusion, if the traffic is above ξ_1^{th}, then the network is limited by itself and we should set $d^\star = [n_d/n_c]$ to maximize the throughput; if the traffic is below ξ_1^{th}, then the network throughput is constrained by the traffic and we should set $d^\star = 1$. In the traffic-limited case, we do not have to worry about the Rogue's Dilemma since the shared channel is sufficient for the given traffic.

VI. Simulation Results

In this section, we present the simulation results to verify our proposed analytical model. We consider the GFRA scenario of B-RAN using Hash Access where n_d possibly selfish IoT devices share n_c open access links. The slot duration is set to 5 ms according to the RA channel (RACH) setup in long-term evolution (LTE) [22]. In all the figures (except Fig. 10), markers and lines represent the simulation and analytical results, respectively.

The monotonicity of $p_s(d)$ plays an important role in the analytical model and optimization of Hash Access. We first demonstrate the monotonic relationship between p_s and d in Fig. 5 to verify Conjecture 1 and Theorem 3. We consider Bernoulli and sporadic traffic patterns with different numbers of devices. As strictly proved by Theorem 3, $p_s(d)$ is monotonically increasing in d with the Bernoulli traffic. Meanwhile, with sporadic traffic, one can also find the monotonic relationship for different n_d, which provides evidence for supporting Conjecture 1.

Fig. 6 illustrates the performance of B-RAN using Hash Access ($n_c = 8$ and $n_d = 30$) with sporadic traffic in terms of transmission success probability, network throughput, and access delay. One can see that the analytical results (dashed lines) fit well with the simulations (markers), implying that our proposed model can characterize the B-RAN using Hash Access accurately.

Fig. 6(a) shows the relationship between the transmission success probability and the offered load. As the traffic intensity increases, the transmission success probability decreases because more packets compete for the limited uplink and cause more collisions. A greater access difficulty can control the number of IoT devices transmitting in a slot and reduce the collision probability. Using the optimal access difficulty, we can see that the transmission success probability is close to $p_s(d^\star) = (1 - [1/n_d])^{n_d-1} \approx (1/e)$.

Fig. 6(b) visualizes the network throughput with varying access difficulties. In light traffic, the network is capable of

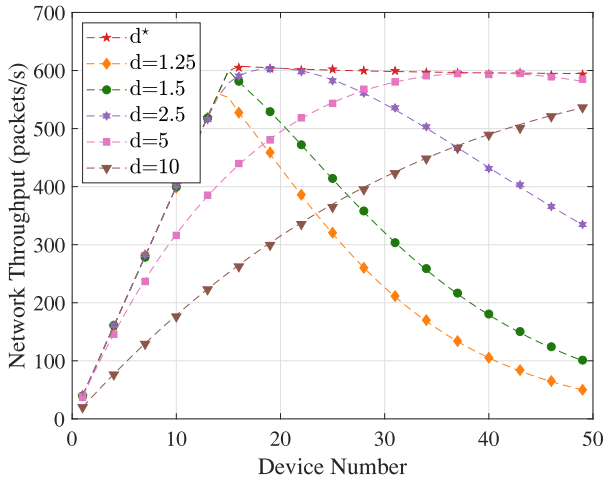


Fig. 7. Network throughput of Hash Access versus the number of devices in the Bernoulli traffic. ($n_c = 8$ and $E[X]/\tau_s = 40$packets/s).

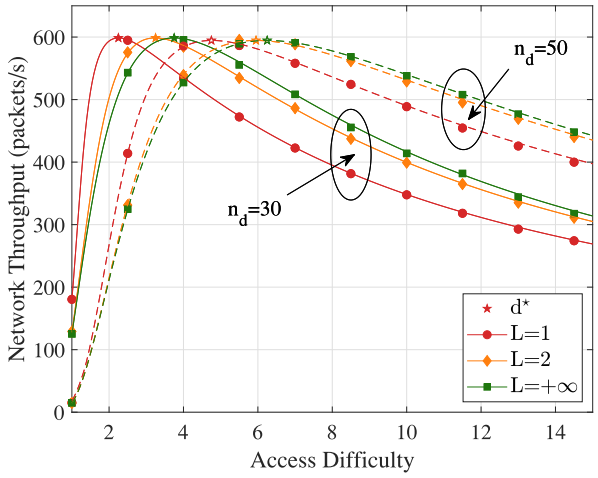


Fig. 8. Network throughput of Hash Access versus access difficulty in the Bernoulli traffic. ($n_c = 8$ and $E[X]/\tau_s = 40$packets/s).

supporting the offered load and thus the network throughput depends on the traffic intensity. In heavy traffic, the network is saturated and limited by its capacity. If d is too more difficult than necessary, the channels are not fully utilized; otherwise, the traffic is not effectively controlled, causing catastrophic collisions. As one can see, the optimal access difficulty $d^\star$ derived by Theorem 2 achieves the maximum network throughput.

We present the access delay of Hash Access at different difficulty levels in Fig. 6(c). As the offered load increases, the access delay rises slightly initially and becomes significant once the network is congested. The delay is relatively long for a greater access difficulty since it needs more time to find a valid hash solution. Meanwhile, for a little difficulty, the access delay is short with light traffic and increases dramatically with heavy traffic due to excessive collisions. The best access difficulty, although is not optimized for latency, still exhibits satisfactory performance in general.

Now, let us look at Bernoulli traffic. Fig. 7 verifies the optimality of $d^\star$ in networks with different numbers of devices. One can see that the optimal access difficulty can always achieve the maximum network throughput. A greater access difficulty unduly restricts traffic flow and negatively affects the network throughput. In a network with a few devices, the transmission resources are adequate and, hence, a low access difficulty can reach a close-to-optimal performance; meanwhile, with more devices, the network throughput falls rapidly due to aggressive competition.

Fig. 8 shows the network throughput versus the access difficulty. Given the buffer size L and the number of devices n_d, one can see that the optimal access difficulty $d^\star$ given by Theorem 2 does maximize the throughput T. From the figure, as the number of devices grows, we should set a greater $d^\star$ to control the traffic, since more devices cause severe collisions. Meanwhile, we also demonstrate the impact of buffer size on the network throughput. If the buffer size is small, a high proportion of data packets may be rejected directly due to the limited buffer size. Hence, the actual traffic intensity is lower than the offered traffic load, requiring a lower optimal access difficulty level, just as illustrated by Fig. 8. From the

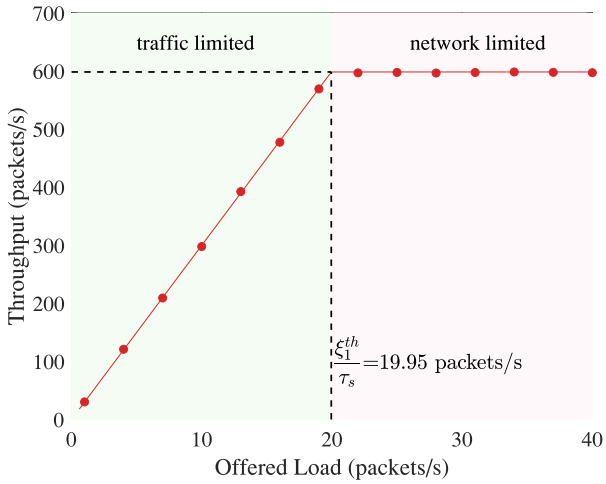


Fig. 9. Network throughput of Hash access with infinite buffer size in the Bernoulli traffic. ($n_c = 8$ and $n_d = 30$).

simulation, if the buffer size is larger enough (say $L > 3$), its impact on the network is negligible.

Fig. 9 presents the network behaviors with infinite buffer size. According to Section V-C, the system could be traffic-limited or network-limited, depending on the offered load. In the traffic-limited status, the uplink is able to support the offered traffic load and thus the throughput is determined by the traffic intensity. The optimal access difficulty is $d^\star = 1$ so that every device can access directly. In the network-limited status, the network resources are scarce so that the throughput is constrained by the number of channels and devices (n_c and n_d). Every device is required to pass the hash check with $d^\star = (n_d/n_c)$. These two statuses are divided by $\xi_1^{\text{th}} = (n_c/n_d)(1 - [1/n_d])^{n_d - 1}$, which is consistent to the conclusion in Section V-C.

Fig. 10 demonstrates the throughput of Hash Access and Aloha in the presence of rogue nodes. The Aloha protocol in a trustworthy environment is shown as a benchmark. The CR based on a uniform backoff algorithm is adopted in the Aloha with a maximum window of 60 slots [37]. From Fig. 10(a), Hash Access using the optimal access difficulty achieves a slightly higher throughput than the Aloha protocol. However, in the existence of selfish devices, the network throughput of

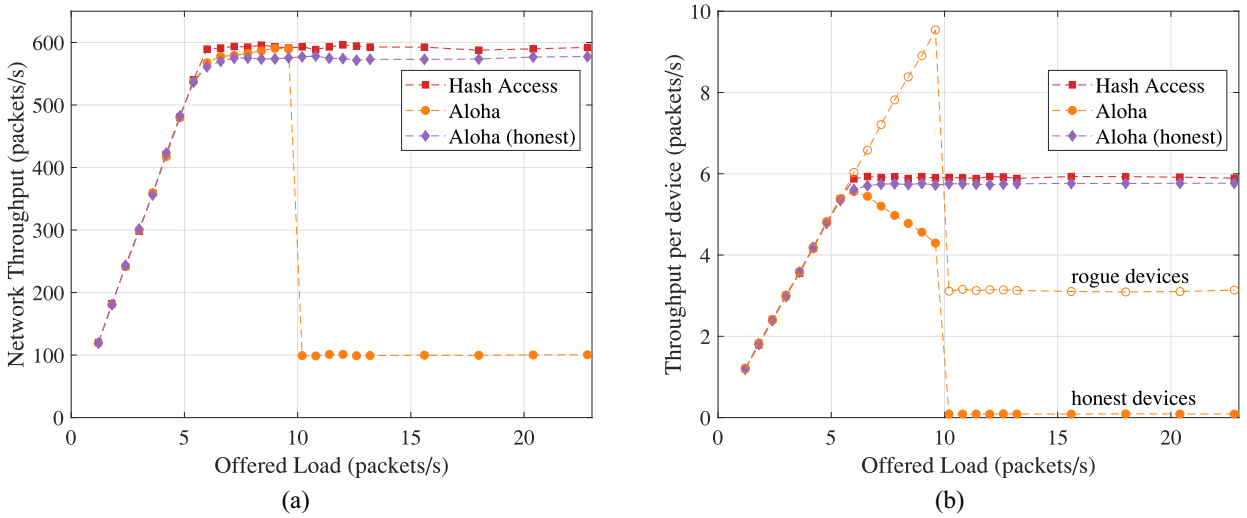


Fig. 10. Network throughput in the presence of rogue devices with sporadic traffic. ($n_c = 8$ and $n_d = 100$). (a) Point of view of the network. (b) Point of view from an individual device.

the Aloha protocol drops sharply once the shared link cannot support so many rogue devices. Hash Access enforces the backoff and thus effectively avoids such a tragic situation.[5] Furthermore, we visualize the throughput per device in Fig. 10(b) to show the interesting phenomenon of the Rogue's Dilemma. Now, look at the performance of the Aloha protocol with 30% rogue devices. With low traffic, the network is not significantly affected, whereas, with high traffic, the rogue devices occupy the uplink for higher individual throughput, and harm the honest devices' interests. As the network congests more severely, even the rogue devices' throughput falls rapidly. Eventually, every device's throughput is worse than the trustworthy case that everyone obeys the given RA protocol.

Finally, let us dive deeper into the Rogue's Dilemma in Fig. 11 by showing the average access delay and the number of transmission attempts per packet. In these two aspects, Hash Access is close to the Aloha protocol in a trustworthy environment, even in the presence of a few rogue devices (5%). Anyway, these rogue devices can obtain negligible access delays so that more devices become rogues due to their selfish nature. Now, with more rogue devices (30%), both honest and rogue devices have to suffer much longer access delays and much more transmission attempts for every packet. The network is highly congested by selfish access, and the trust between client devices is severely damaged. If Hash Access is not adopted in B-RAN or other untrustworthy environments, every device would optimize for the self in the short term, which, ironically, precipitates the network collapse. Fig. 11 indicates the profound implications of Hash Access for avoiding the Rogue's Dilemma and establishing the trust between clients.

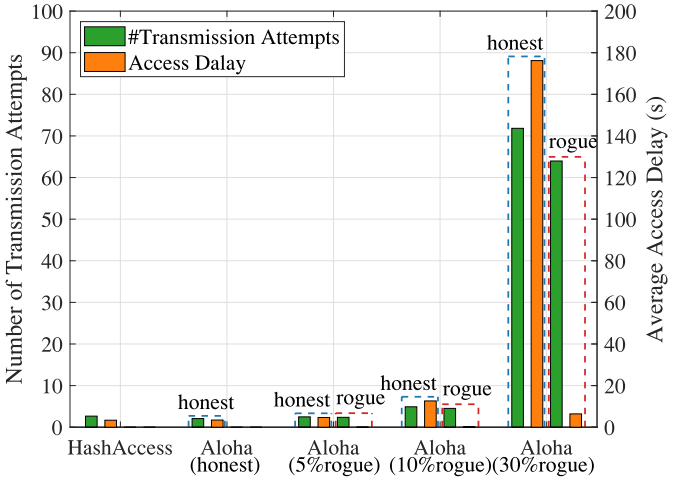


Fig. 11. Impact of rogue nodes on the network regarding the average number of transmission attempts per packet and average access delay. ($n_c = 8$, $n_d = 100$, and $E[X]/\tau_s = 40$packets/s).

VII. Conclusion

In this study, we started from the Rogue's Dilemma via a simplified two-player game. The game model showed how such rogue devices affected the entire network and qualified the value of trust between clients established by Hash Access. After demonstrating Hash Access in detail, we built a Markov model for general traffic patterns to quantitatively evaluate the performance of B-RAN using Hash Access in different aspects. Based on the analytical model, we further optimized Hash Access in terms of throughput and pointed out the relationships between the access difficulty and the key performance indicators, such as network throughput and transmission success probability. Also, we characterized the impact of the buffer size and assessed the infinite buffer case with closed-form solutions. Simulation results illustrate the effectiveness of our established analytical model.

[5]Note that Fig. 10(a) shows a different result from [16, Fig. 5(a)], since the traffic load in [16] includes the packets retransmitted due to the collisions, which is different from the definition of offered load in this study.

References

[1] X. Ling, J. Wang, Y. Le, Z. Ding, and X. Gao, "Blockchain radio access network beyond 5G," *IEEE Wireless Commun.*, vol. 27, no. 6, pp. 160–168, Dec. 2020.

[2] X. Ling, J. Wang, T. Bouchoucha, B. C. Levy, and Z. Ding, "Blockchain radio access network (B-RAN): Towards decentralized secure radio access paradigm," *IEEE Access*, vol. 7, pp. 9714–9723, Jan. 2019.

[3] J. Wang, X. Ling, Y. Le, Y. Huang, and X. You, "Blockchain enabled wireless communications: A new paradigm towards 6G," *Nat. Sci. Rev.*, vol. 8, no. 9, Apr. 2021, Art. no. nwab069.

[4] X. Ling, Y. Le, J. Wang, Z. Ding, and X. Gao, "Practical modeling and analysis of blockchain radio access network," *IEEE Trans. Commun.*, vol. 69, no. 2, pp. 1021–1037, Feb. 2021.

[5] H. Xu, L. Zhang, E. Sun, and C. I, "BE-RAN: Blockchain-enabled open RAN with decentralized identity management and privacy-preserving communication," Jan. 2021. [Online]. Available: arXiv:2101.10856.

[6] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, Jun. 2016.

[7] B. Cao *et al.*, "When Internet of Things meets blockchain: Challenges in distributed consensus," *IEEE Netw.*, vol. 33, no. 6, pp. 133–139, Nov./Dec. 2019.

[8] Z. Xiong, Y. Zhang, N. C. Luong, D. Niyato, P. Wang, and N. Guizani, "The best of both worlds: A general architecture for data management in blockchain-enabled Internet-of-Things," *IEEE Netw.*, vol. 34, no. 1, pp. 166–173, Jan./Feb. 2020.

[9] H. Xu, P. V. Klaine, O. Onireti, B. Cao, M. Imran, and L. Zhang, "Blockchain-enabled resource management and sharing for 6G communications," *Digit. Commun. Netw.*, vol. 6, no. 3, pp. 261–269, Aug. 2020.

[10] H. Xu, L. Zhang, Y. Liu, and B. Cao, "RAFT based wireless blockchain networks in the presence of malicious jamming," *IEEE Wireless Commun. Lett.*, vol. 9, no. 6, pp. 817–821, Jun. 2020.

[11] Z. Xiong, S. Feng, W. Wang, D. Niyato, P. Wang, and Z. Han, "Cloud/fog computing resource management and pricing for blockchain networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4585–4600, Jun. 2019.

[12] S. Guo, Y. Qi, Y. Jin, W. Li, X. Qiu, and L. Meng, "Endogenous trusted DRL-based service function chain orchestration for IoT," *IEEE Trans. Comput.*, early access, Jan. 18, 2021, doi: 10.1109/TC.2021.3051681.

[13] Y. Li *et al.,* "Direct acyclic Graph-Based ledger for Internet of Things: Performance and security analysis," *IEEE/ACM Trans. Networking*, vol. 28, no. 4, pp. 1643–1656, Aug. 2020.

[14] B. Cao *et al.*, "Performance analysis and comparison of PoW, PoS and DAG based blockchains," *Digit. Commun. Netw.*, vol. 6, no. 4, pp. 480–485, Nov. 2020.

[15] Y. Le, X. Ling, J. Wang, and Z. Ding, "Prototype design and test of blockchain radio access network," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Shanghai, China, May 2019, pp. 1–6.

[16] X. Ling, Y. Le, J. Wang, and Z. Ding, "Hash access: Trustworthy grant-free IoT access enabled by blockchain radio access networks," *IEEE Netw.*, vol. 34, no. 1, pp. 54–61, Jan./Feb. 2020.

[17] G. Berardinelli *et al.*, "Reliability analysis of uplink grant-free transmission over shared resources," *IEEE Access*, vol. 6, pp. 23602–23611, May 2018.

[18] Y. Wei, F. R. Yu, M. Song, and Z. Han, "Joint optimization of caching, computing, and radio resources for fog-enabled IoT using natural actor–critic deep reinforcement learning," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2061–2073, Apr. 2019.

[19] B. Cao, M. Li, L. Zhang, Y. Li, and M. Peng, "How does CSMA/CA affect the performance and security in wireless blockchain networks," *IEEE Trans. Ind. Inf.*, vol. 16, no. 6, pp. 4270–4280, Jun. 2020.

[20] L. Zhang, H. Xu, O. Onireti, M. A. Imran, and B. Cao, "How much communication resource is needed to run a wireless blockchain network?" Jan. 2021. [Online]. Available: arXiv:2101.10852.

[21] Z. Li, W. Wang, and Q. Wu, "Blockchain-based dynamic spectrum sharing for 5G and beyond wireless communications," in *Proc. Int. Conf. Blockchain Trustworthy Syst.*, Nov. 2020, pp. 575–587.

[22] A. Laya, L. Alonso, and J. Alonso-Zarate, "Is the random access channel of LTE and LTE-A suitable for M2M communications? A survey of alternatives," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 4–16, 1st Quart., 2014.

[23] N. Jiang, Y. Deng, A. Nallanathan, X. Kang, and T. Q. S. Quek, "Analyzing random access collisions in massive IoT networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6853–6870, Oct. 2018.

[24] G. Hardin, "The tragedy of the commons," *Science*, vol. 162, no. 3859, pp. 1243–1248, Dec. 1968.

[25] J. Farhat, J. F. Grybosi, G. Brante, R. D. Souza, and J. L. Rebelatto, "Non-orthogonal hash access for grant-free IoT blockchain radio access networks," *IEEE Wireless Commun. Lett.*, vol. 10, no. 5, pp. 1066–1070, May 2021.

[26] M. Osborne and A. Rubinstein, *A Course in Game Theory*. Cambridge, MA, USA: MIT Press, 1994.

[27] G. Li, J. Yu, Y. Xing, and A. Hu, "Location-invariant physical layer identification approach for WiFi devices," *IEEE Access*, vol. 7, pp. 106974–106986, Aug. 2019.

[28] J. Yu, A. Hu, G. Li, and L. Peng, "A robust RF fingerprinting approach using multisampling convolutional neural network," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6786–6799, Aug. 2019.

[29] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Bitcoin, Rep., Oct. 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[30] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2084–2123, 3rd Quart., 2016.

[31] W. Zhan and L. Dai, "Massive random access of machine-to-machine communications in LTE networks: Modeling and throughput optimization," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2771–2785, Apr. 2018.

[32] P. Osti, P. Lassila, S. Aalto, A. Larmo, and T. Tirronen, "Analysis of PDCCH performance for M2M traffic in LTE," *IEEE Trans. Veh. Technol.*, vol. 63, no. 9, pp. 4357–4371, Nov. 2014.

[33] E. Soltanmohammadi, K. Ghavami, and M. Naraghi-Pour, "A survey of traffic issues in machine-to-machine communications over LTE," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 865–884, Dec. 2016.

[34] R. A. Horn and C. R. Johnson, *Matrix Analysis*. New York, NY, USA: Cambridge Univ. Press, 1985.

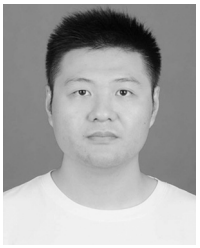
[35] L. Kleinrock, *Theory, Volume 1, Queueing Systems*. New York, NY, USA: Wiley-Intersci., 1975.

[36] B. Zhang, X. Ling, Y. Le, J. Wang, C. Cai, and Z. Tang, "Analysis and evaluation of hash access for blockchain radio access networks," in *Proc. 12th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Nanjing, China, Oct. 2020, pp. 62–67.

[37] M. E. Rivero-Angeles, D. Lara-Rodriguez, and F. A. Cruz-Perez, "Gaussian approximations for the probability mass function of the access delay for different backoff policies in S-ALOHA," *IEEE Commun. Lett.*, vol. 10, no. 10, pp. 731–733, Oct. 2006.

Xintong Ling (Member, IEEE) received the B.E. and Ph.D. degrees in electrical engineering from Southeast University, Nanjing, China, in 2013 and 2018, respectively.

He is currently an Associate Professor with the National Mobile Communications Research Laboratory, Southeast University and also with the Purple Mountain Laboratories, Nanjing. From 2016 to 2018, he was a visiting Ph.D. student with the Department of Electrical and Computer Engineering, University of California at Davis, Davis, CA, USA. His current research interests focus on future-generation wireless communications and networks, including blockchain technologies, distributed systems, machine learning, optical wireless communications, and signal processing.



Bowen Zhang received the B.E. and M.S. degrees in electrical engineering from Southeast University, Nanjing, China, in 2018 and 2021, respectively.

His current research interests include wireless communications, Internet of Things, and blockchain.

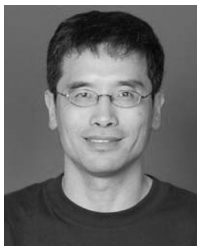
Hui Xie received the B.E. degree in communication engineering from North China Electric Power University, Beijing, China, in 2020. He is currently pursuing the M.S. degree with the School of Information Science and Engineering, National Mobile Communications Research Laboratory, Southeast University, Nanjing, China.

His research interests include medium access control protocol, Internet of Things, and blockchain.

Jiaheng Wang (Senior Member, IEEE) received the B.E. and M.S. degrees from Southeast University, Nanjing, China, in 2001 and 2006, respectively, and the Ph.D. degree in electronic and computer engineering from Hong Kong University of Science and Technology, Hong Kong, in 2010.

He is currently a Full Professor with the National Mobile Communications Research Laboratory, Southeast University and also with the Purple Mountain Laboratories. From 2010 to 2011, he was with the Signal Processing Laboratory, KTH Royal Institute of Technology, Stockholm, Sweden. He also held visiting positions with the Friedrich Alexander University Erlangen–Nürnberg, Nürnberg, Germany, and the University of Macau, Macau, China. He has published more than 150 articles on international journals and conferences. His research interests are mainly on communication systems and networks.

Dr. Wang was a recipient of the Humboldt Fellowship for Experienced Researchers and the best paper awards of IEEE GLOBECOM 2019, ADHOCNETS 2019, and WCSP 2014. From 2014 to 2018, he served as an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS. In 2018, he served as a Senior Area Editor for the IEEE SIGNAL PROCESSING LETTERS.



Zhi Ding (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Cornell University, Ithaca, NY, USA, in 1990.

He is a Professor of Electrical and Computer Engineering with the University of California at Davis, Davis, CA, USA. From 1990 to 2000, he was a Faculty Member of Auburn University, Auburn, AL, USA, and later, University of Iowa, Iowa City, IA, USA. He has held visiting positions with Australian National University, Canberra, ACT, Australia; Hong Kong University of Science and Technology, Hong Kong; NASA Lewis Research Center, Cleveland, OH, USA; and USAF Wright Laboratory, Wright-Patterson AFB, OH, USA. He has active collaboration with researchers from several countries, including Australia, China, Japan, Canada, Taiwan, South Korea, Singapore, and Hong Kong. He has coauthored the text: *Modern Digital and Analog Communication Systems* (4th ed., Oxford University Press, 2009).

Prof. Ding received the 2012 IEEE Wireless Communication Recognition Award from the IEEE Communications Society. He has been an active volunteer, serving on technical programs of several workshops and conferences. He was an Associate Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING from 1994 to 1997 and from 2001 to 2004, and IEEE SIGNAL PROCESSING LETTERS from 2002 to 2005. He was a member of technical committee on Statistical Signal and Array Processing and member of Technical Committee on Signal Processing for Communications from 1994 to 2003. He was the Technical Program Chair of the 2006 IEEE Globecom. He is also an IEEE Distinguished Lecturer of Circuits and Systems Society from 2004 to 2006 and Communications Society from 2008 to 2009. He served as an IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS Steering Committee Member from 2007 to 2009 and its Chair from 2009 to 2010.