Received 14 June 2021; revised 10 July 2021; accepted 12 July 2021. Date of publication 19 July 2021; date of current version 28 July 2021.

Digital Object Identifier 10.1109/OJCOMS.2021.3098197

Comparing Delay-Constrained ALOHA and CSMA: A Learning-Based Low-Complexity Approximate Approach

LEI DENG [1] (Member, IEEE), YOUZHI YU [1], YIJIN ZHANG [2] (Senior Member, IEEE), AND YUNGHSIANG S. HAN [3] (Fellow, IEEE)

[1]Shenzhen Key Laboratory of Digital Creative Technology and Guangdong Province Engineering Laboratory for Digital Creative Technology, College of Electronics and Information Engineering, Shenzhen University, Shenzhen 518060, China

[2]School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

[3]Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen 518110, China

CORRESPONDING AUTHOR: L. DENG (e-mail: ldeng@szu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61902256, Grant 62071236, and Grant 61671007; in part by Tencent "Rhinoceros Birds"-Scientific Research Foundation for Young Teachers of Shenzhen University; and in part by the Fundamental Research Funds for the Central Universities of China under Grant 30920021127. A preliminary version of this article was published in the 16th IEEE International Conference on Control & Automation, 2020 [1].

ABSTRACT Supporting delay-constrained traffic becomes more and more critical in multimedia communication systems, tactile Internet, networked control systems, and cyber-physical systems, etc. In delay-constrained traffic, each packet has a hard deadline. When it is not delivered before the hard deadline, it becomes useless and will be removed from the system. This feature is completely different from that of traditional delay-unconstrained traffic and brings new challenge to network protocol design. In this work, we study the widely-used (slotted) ALOHA and CSMA wireless access protocols but under the new delay-constrained setting. Our goal is to compare delay-constrained ALOHA and CSMA for different system settings and thus give network operators guidelines on protocol selection. We use two Markov chains to analyze delay-constrained ALOHA and CSMA, respectively. However, the number of states of Markov chains increases exponentially with respect to the number of users in the network. Therefore, we can only compare the exact performance of delay-constrained ALOHA and CSMA for small-scale networks. To address the curse of dimensionality, we design a single-user parameterized ALOHA (resp. CSMA) system, where the parameters are to be learned to approximate the original multi-user ALOHA (resp. CSMA) system. In addition, our low-complexity approach preserves the Markov-chain structure of the systems and thus enables us to compute some other interested performance metrics such as average delivery time. We use our low-complexity approach to reveal the conditions under which ALOHA (resp. CSMA) outperforms CSMA (resp. ALOHA) in the delay-constrained setting via extensive simulations.

INDEX TERMS Delay-constrained communications, ALOHA, CSMA, Markov chain, machine learning.

I. INTRODUCTION

DELAY-CONSTRAINED applications become widespread nowadays. Typical examples include multimedia communication systems such as real-time streaming and video conferencing [2], tactile Internet [3], [4], networked control systems (NCSs) such as remote control of unmanned aerial vehicles (UAVs) [5], [6], and cyber-physical systems (CPSs) such as medical tele-operations, X-by-wire vehicles/avionics, factory automation, and robotic collaboration [7]–[9]. In such applications, each packet has a hard deadline: if it is not delivered before the deadline, it becomes useless and will be removed from the system. On the other hand, wireless communication is ubiquitous because it can be easily deployed with low cost and low complexity. We focus on delay-constrained wireless communication in this paper. Many works designed centralized scheduling policies in the downlink [2], [10]–[12], while a few works investigated distributed wireless access protocols in the uplink for delay-constrained traffic [13]–[19].

ALOHA and carrier sense multiple access (CSMA) are two widely used random access protocols in traditional delay-unconstrained wireless communication. The advantage of ALOHA is that it is extremely simple. Since Abramson invented pure ALOHA in 1970 [20], a variety of other ALOHA-type protocols have been designed. Among them, one popular type is *slotted ALOHA*, where users are synchronized and can only transmit data at the beginning of a slot [21]. We focus on slotted ALOHA in the rest of this paper. For simplicity, we will sometimes call it ALOHA if there is no ambiguity through the context. There are also many types of extension for slotted ALOHA protocol, including slotted ALOHA for multi-packet reception [22], [23], framed slotted ALOHA [24]–[26], and coded slotted ALOHA for successive interference cancellation (SIC) [27], [28]. In addition, some works studied stability analysis for the slotted ALOHA systems [29]–[31]. CSMA is a more sophisticated wireless access protocol than slotted ALOHA, which is divided into non-persistent CSMA, p-persistent CSMA and 1-persistent CSMA [32]. Note that p-persistent CSMA is the most general one, which also has several versions, including a constant probability p, a uniform backoff strategy and a multi-stage backoff strategy, etc. Moreover, the behaviors of each such version can be closely approximated by each other (at least from the standpoint of maximum throughput) if the p value is selected to guarantee that the same average backoff interval is used [33], [34]. In the rest of this paper, we will consider CSMA with a uniform backoff strategy. Previous studies have shown that the design of parameters is important to improve the performance of CSMA [33], [35], [36]. There are also some works to compare ALOHA and CSMA under the delay-unconstrained setting [37], [38]. We also remark that the discrete-time Markov chain (DTMC) is an important tool to analyze ALOHA and CSMA. For example, [22], [26], [29], [31] used DTMC to analyze delay-unconstrained ALOHA and [33]–[36] used DTMC to analyze delay-unconstrained CSMA. In addition, currently there are also some works to use machine learning approaches to optimize MAC protocols for delay-unconstrained traffic, e.g., [39]–[42]. But all these works only focused on delay-unconstrained setting.

Due to the great success of ALOHA and CSMA, it deserves to investigate how they work in the delay-constrained setting. There are some existing works on delay-constrained ALOHA [15], [43], [44] and delay-constrained CSMA [45], [46]. Deng *et al.* in [15] analyzed the asymptotic performance of ALOHA system for frame-synchronized delay-constrained traffic pattern. Zhang *et al.* in [43], [44] studied the system throughput and optimal transmission probability of ALOHA under the saturated delay-constrained traffic. Campolo *et al.* in [45] analyzed the p-persistent CSMA for broadcasting delay-constrained traffic. Lu *et al.* in [46] proposed a frame-based CSMA algorithm which is shown to be asymptotically optimal for distributed scheduling of delay-constrained traffic in an ad hoc wireless network. However, to the best of our knowledge, currently no work compares ALOHA and CSMA under the delay-constrained setting. In this paper, we aim at theoretically providing a comprehensive comparison for delay-constrained ALOHA and delay-constrained CSMA protocols. We remark that there are many variants of ALOHA, CSMA, and other MAC protocols. We do not try to compare all of them in this paper. Instead, we only compare the conventional slotted ALOHA with a p-persistent strategy and conventional CSMA with a uniform backoff strategy in this paper. We also remark that previous studies [15], [43] assumed that the packet size $L = 1$, i.e., a packet can be delivered in one slot. However, in many applications, the packet size can be large enough such that it cannot be delivered in one slot but needs to be split into multiple slots to finish transmission [47]. To capture this case, in this paper we generalize L to be an arbitrary positive integer, and thus the delivery of a packet needs L slots. Partial delivery does not contribute to the throughput. We remark that we cannot simply enlarge the slot duration so that a large packet can be delivered in one slot and then we can reduce the problem of $L > 1$ to the well-studied problem of $L = 1$. The reason is that such reduction cannot capture the feature that partial delivery does not contribute to the throughput. Embedded with this new feature, we make the following contributions.

- For a given number of users N, hard delay D, packet size L, we construct two Markov chains to analyze delay-constrained ALOHA and delay-constrained CSMA. By analyzing the state distributions of the Markov chains, we obtain the exact theoretical system timely throughput for delay-constrained ALOHA and CSMA.
- The number of states of Markov chains in the previous exact characterization increases exponentially with respect to the number of users. Therefore, we can only compare the exact performance of delay-constrained ALOHA and CSMA for small-scale networks. We thus design a parameterized ALOHA (resp. CSMA) system in view of only one user where there are two parameters to be learned to approximate the original multi-user ALOHA (resp. CSMA) system. The numerical results show the effectiveness of this approximate approach.
- Since our proposed low-complexity approach preserves the Markov-chain structure of the systems, a by-product of our approach is that we can compute some other interested performance metrics. In this paper, we show how to theoretically compute the average delivery time of those packets that have been delivered successfully before expiration.
- Using our proposed low-complexity approximate approach, we compare delay-constrained ALOHA and CSMA for different system settings, and then summarize the conditions under which delay-constrained

ALOHA (resp. CSMA) outperforms delay-constrained CSMA (resp. ALOHA).

The rest of this paper is outlined as follows. Section II describes the system model. In Section III, we construct two Markov chains to analyze the exact performance of delay-constrained ALOHA and CSMA, respectively. In Section IV, we propose a learning-based low-complexity approximate approach. In Section V, we show how to compute the average delivery time based on our low-complexity approach. Section VI provides numerical results and summarizes the conditions under which delay-constrained ALOHA (resp. CSMA) outperforms delay-constrained CSMA (resp. ALOHA). Section VII concludes this paper.

II. SYSTEM MODEL

We consider a wireless network with N users who need to independently deliver delay-constrained traffic to a common receiver by competing for a shared wireless channel. We consider a slotted system indexed from 1. Similar to previous studies [2], [10], [11], [15], we consider the frame-synchronized traffic pattern. But we characterize the traffic pattern by parameters $L \in \mathbb{Z}^+$ and $D \in \mathbb{Z}^+$. More precisely, starting from slot 1, each user has a new packet arrival every D slots.[1] All packets are of size L and have a hard delay of D slots. A packet will be useless and removed from the system if it cannot be delivered within D slots after its arrival. We also call the duration from slot $(k-1)T+1$ to slot kT *frame* k or *period* k where $k = 1, 2, \ldots,$.

Note that existing studies [2], [10], [11], [15] assume that all packets are of unit size, i.e., $L = 1$. To capture more practical applications, we generalize packet size to a positive integer L, meaning that a user needs L slots to deliver a packet. In addition, a packet can be divided into L different units for transmission but we do not allow partial delivery. Thus, a packet can contribute to the system performance in terms of timely throughput, which will be described in (1) shortly, only when the whole packet of size L has been completely delivered within D slots after its arrival. Since sending a packet needs at least L slots and the hard delay is D, without loss of generality, we assume that $L \leq D$.

We illustrate an example of $D = 3$ and $L = 2$ in Fig. 1. Each user has a new packet of size $L = 2$ at the beginning of slot 1, which will expire and be removed from the system if it cannot be delivered completely within $D = 3$ slots. Then, at the beginning of slot 4, each user has another new packet of size $L = 2$, which again has a hard delay of $D = 3$ slots. The process continues with the same behaviors.

It is straightforward to see that any user has at most one *non-expired* packet for delivery in any slot. For any user, if the packet arrived at the beginning of a period has been completely delivered successfully to the receiver, this user remains idle until the end of this period. Therefore, for our wireless access problem, we do not need to handle the packet queueing problem. In addition, since each packet is of size L and the transmission capacity of each slot is only one unit, we divide each packet into L units and call them unit 1, unit 2, $\cdots$, and unit L. In our wireless access problem, we sequentially transmit these L units. Namely, we keep transmitting unit i until it has been successfully delivered to the receiver. After that, we keep transmitting unit $i+1$. If the final unit L has been delivered successfully to the receiver before the end of this period, this packet is successfully delivered and can contribute to this user's timely throughput.

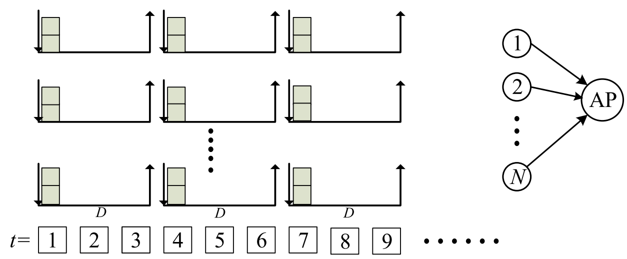


FIGURE 1. An example of the system model with $D = 3$ and $L = 2$.

In this paper, we adopt conventional wireless access setting with a collision channel—if two or more users send data in the same slot, a collision happens and no data can be delivered to the receiver; if only one user transmits data, it can be successfully delivered to the receiver. Same as previous studies on delay-constrained communications, we use *timely throughput* to characterize the system performance. The timely throughput of user i is defined as

$$R_i \triangleq \lim_{k\to\infty} \frac{L \cdot \mathbb{E}\left[\begin{array}{c}\text{number of packets of user } i \text{ delivered} \\ \text{before expiration from slot 1 to slot } kD\end{array}\right]}{kD}. \quad (1)$$

Our goal is to maximize the system timely throughput $R = \sum_{i=1}^{N} R_i$.

An Practical Example: We consider a WiFi network in an industrial automation system where N PLCs (Programmable Logic Controllers) needs to send control messages to remote I/O devices via a WiFi AP, as shown in Fig. 2. According to the on-going IEC/IEEE standard 60802 [48], we assume that the control messages are isochronous or cyclic-synchronous such that they arrive periodically (say with period D) and the hard delay of such packets are the same with period D. Thus, the control messages of all N PLCs form a frame-synchronized traffic patter as shown in Fig. 1. In WiFi networks, all PLCs (stations) need to compete the channel for uplink transmission via random-access schemes, which are typically CSMA/CA. We further remark that indeed, the next generation WiFi (IEEE 802.11be, aka, WiFi 7) aims at supporting time-sensitive applications [49]. Thus, our investigated model in this paper can be supported by this practical example.

1. We distinguish the two terminologies: *delay* and *deadline*. Delay refers to a time duration while deadline refers to a time instance. For example, if a packet arrives at the beginning of slot 10 and will expire at the end of slot 14, the packet has a *hard delay* of 5 slots and its *hard deadline* is the end of slot 14.

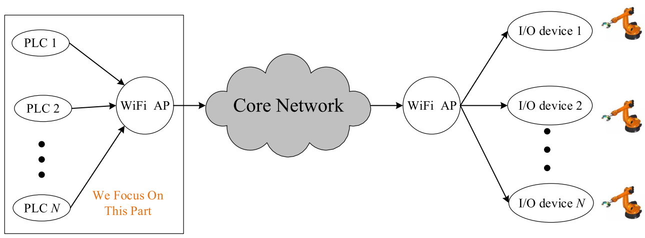


FIGURE 2. An application of our system model.

In the following, we will consider two widely used distributed wireless access protocols, slotted ALOHA and CSMA. As we mentioned in Section I, we sometimes use ALOHA to represent slotted ALOHA if there is no ambiguity through the context in the rest of this paper.

We remark that in this work, we follow traditional analysis for ALOHA and CSMA by only considering a single-channel scenario. Namely, we do not consider frequency-domain resource allocation in a multi-channel scenario, like OFDM-based 5G system. Instead, we focus on the time-domain random access scheme in a single-channel scenario, e.g., a Wi-Fi network with a single shared channel. But we should note that there are some works studying multi-channel resource allocation for delay-constrained traffic in 5G systems with flexible numerology [50]–[52], where both time-domain (with varying slot length) and frequency-domain resources need to be allocated and generally a complex integer program problem is involved. In this paper, we do not consider such joint time-domain and frequency-domain resource allocation problem. We would leave it as a future direction to compare delay-constrained ALOHA and CSMA is such a complex multi-channel system.

III. THEORETICAL ANALYSIS

A. ALOHA

Mechanism: In (slotted) ALOHA protocol, in each slot, if a user has a packet that has not yet delivered successfully before its deadline, this user transmits the current unit of this packet to the receiver with probability $p \in [0, 1]$; otherwise, if unit L of the packet has been delivered successfully, this user remains idle. Note that the transmission probability p needs to be optimized to maximize the system timely throughput. We assume that the transmission events of all users in any slot are independent as same as the traditional delay-unconstrained ALOHA protocol [20], [31].

Markov Chain Analysis: For the given number of users N, packet size L, and hard delay D, we construct a Markov chain to analyze the theoretical performance of ALOHA. In particular, the system state in any slot is denoted by

$$s = [(l_1, t), (l_2, t), \ldots, (l_N, t)], \tag{2}$$

where $l_i \in \{0, 1, \ldots, L\}$ is the number of units of the current (non-expired) packet that have been delivered successfully to the receiver before the current slot and $t \in \{1, 2, \ldots, D, D+1\}$ is the index of the current slot relative to the beginning of this period. Note that we construct a *virtual slot*, i.e., $t = D+1$, to indicate the end of the period, which is used for calculating the system timely throughput from the state distribution.

Let us consider an example with $N = 2$, $L = 2$ and $D = 3$. State $s = [(l_1, t), (l_2, t)] = [(0, 1), (0, 1)]$ means that this slot is the beginning (the first slot) of the period and no unit has been delivered successfully to the receiver before this slot, since a new packet just arrives at the system in the beginning of this slot. State $s = [(l_1, t), (l_2, t)] = [(0, 2), (1, 2)]$ means that this slot is the second slot of the period and no unit of user 1 has been delivered successfully to the receiver before this slot while unit 1 of user 2 has been delivered successfully to the receiver before this slot. State $s = [(l_1, t), (l_2, t)] = [(1, 4), (2, 4)]$ means that this slot is the virtual slot to indicate the end of the period and user 1 only transmits one unit while user 2 has delivered the whole packet to the receiver. Therefore, the packet of user 2 contributes to the timely throughput of user 2 while the packet of user 1 is discarded and does not contribute to the timely throughput of user 1. Note that the state space, denoted by $\mathcal{S}_{\mathsf{ALOHA}}$, is of size

$$[(L+1)]^N \cdot (D+1). \tag{3}$$

Now we construct the transition probabilities of the constructed Markov chain for ALOHA protocol. For state $s = [(l_1, t), (l_2, t), \ldots, (l_N, t)] \in \mathcal{S}_{\mathsf{ALOHA}}$, where $t \in \{1, 2, \ldots, D\}$, we divide all N users into two set $\mathcal{N}_1$ and $\mathcal{N}_2$, where $\mathcal{N}_1$ is the set of users who have not yet completely delivered the packet and $\mathcal{N}_2$ is the set of users who have already completely delivered the packet before this slot. That is,

$$\mathcal{N}_1 = \{i : l_i < L, i = 1, 2, \ldots, N\},$$

and

$$\mathcal{N}_2 = \{i : l_i = L, i = 1, 2, \ldots, N\}.$$

Thus, in this slot, all users in $\mathcal{N}_2$ remain idle and all users in $\mathcal{N}_1$ transmit one unit with probability p. We can compute the transition probabilities as follows,

$$\begin{aligned} &P\{[(l_1, t+1), \ldots, (l_i+1, t+1), \ldots, (l_N, t+1)]|s\} \\ &\quad = p(1-p)^{|\mathcal{N}_1|-1}, \forall i \in \mathcal{N}_1, \\ &P\{[(l_1, t+1), (l_2, t+1), \ldots, (l_N, t+1)]|s\} \\ &\quad = 1 - |\mathcal{N}_1| p(1-p)^{|\mathcal{N}_1|-1}, \end{aligned} \tag{4}$$

and the transition probabilities from state s to all other states are zero. Note that (4) follows from the fact that there exists at most one successful transmission in any slot.

Now we have a transition matrix $\boldsymbol{P} = [P(s'|s) : s, s' \in \mathcal{S}_{\mathsf{ALOHA}}]$. Next we give an initial state distribution (in the beginning of slot 1) $\boldsymbol{\pi}^1 = (\pi_s^1 : s \in \mathcal{S}_{\mathsf{ALOHA}})$ as

$$\pi_s^1 = \begin{cases} 1, & \text{If } s = [(0,1), (0,1), \ldots, (0,1)]; \\ 0, & \text{Otherwise.} \end{cases}$$

Then the state distribution in the beginning of slot 2 is

$$\boldsymbol{\pi}^2 = \boldsymbol{\pi}^1 \boldsymbol{P}.$$

Similarly, we can obtain the state distribution in the beginning of the virtual slot $D+1$,

$$\boldsymbol{\pi}^{D+1} = \boldsymbol{\pi}^1 \boldsymbol{P}^D. \tag{5}$$

Based on the state distribution $\boldsymbol{\pi}^{D+1}$, we can compute the timely throughput of user i by

$$R_i = \frac{L}{D} \cdot \left\{ \sum_{\substack{s=[(l_1,D+1),\ldots,(l_i,D+1),\ldots,(l_N,D+1)]:\\ l_i=L, l_j\in\{1,2,\ldots,L\},\forall j\neq i}} \pi_s^{D+1} \right\}.$$

Then we can obtain the exact theoretical system timely throughput $R = \sum_{i=1}^{N} R_i$.

In addition, we would like to discuss how to calculate the D-th power of the transition matrix $\boldsymbol{P}$, more precisely, how to calculate the RHS of (5). It is nature that we first get the D-th power of $\boldsymbol{P}$, i.e., $\boldsymbol{P}^D$, which could have plenty of numerical methods. Anyway, $\boldsymbol{P}$ is a square matrix of size $|\mathcal{S}_{\text{ALOHA}}| \times |\mathcal{S}_{\text{ALOHA}}|$. We need to do matrix-multiplying-matrix $D-1$ times. However, what we need is $\boldsymbol{\pi}^1\boldsymbol{P}^D$ rather than $\boldsymbol{P}^D$. Note that π^1 is a vector of size $1 \times |\mathcal{S}_{\text{ALOHA}}|$. We do not need to do matrix multiplication, but instead we only need to do vector-multiplying-matrix D times. Namely, we first get $\boldsymbol{\pi}^2 = \boldsymbol{\pi}^1\boldsymbol{P}$, which involves one vector-multiplying-matrix operation. We then get $\boldsymbol{\pi}^3 = \boldsymbol{\pi}^1\boldsymbol{P}^2 = \boldsymbol{\pi}^2\boldsymbol{P}$, which again involves one vector-multiplying-matrix operation. We continue this process for D times. This computation process is much more faster than the one based on calculating $\boldsymbol{P}^D$. In this paper, we adopt this computation process.

Note that the achieved system timely throughput R depends on the number of users N, the hard delay D, the packet size L, and the transmission probability p. To show this dependance clearly, we denote $R_{\text{ALOHA}}(N, D, L, p)$ as the exact theoretical system timely throughput of ALOHA protocol for given N, D, L and p.

To maximize the system performance, we need to find the best transmission probability p, i.e.,

$$p^*(N, D, L) = \arg\max_{p\in[0,1]} R_{\text{ALOHA}}(N, D, L, p).$$

It is difficult to find closed-form $p^*(N, D, L)$. In this paper, we numerically search it with an adjustable step size to control the precision. We denote the maximum system timely throughput of ALOHA by $R^*_{\text{ALOHA}}(N, D, L)$.

We remark that although we use a brute-force-type searching algorithm to find $p^*(N, D, L)$, it is possible to use more efficient searching algorithm by exploiting the structure of $R_{\text{ALOHA}}(N, D, L, p)$. For example, we observes that $R_{\text{ALOHA}}(N, D, L, p)$ first increases and the decreases with respect to p, which enables more efficient searching algorithms. However, currently we lack of theoretical proofs for such structure. We thus leave it as a future direction to design more efficient searching algorithms.

B. CSMA

Mechanism: In CSMA/CA (carrier-sense multiple access with collision avoidance) protocol,[2] each node has a capability of carrier sensing to check whether the wireless channel is idle or not. If a user has a new packet arrival (i.e., in the beginning of a period), it randomly selects an integer value b from $[0, D-1]$ as the backoff time.[3] Then, in each slot, the user performs carrier sensing. If the channel is busy, the backoff-time value b is frozen; otherwise the backoff-time value b decreases by 1. If $b = 0$, there are two cases. If the packet has been delivered or the remaining number of units is larger than the remaining number of slots before expiration, the user remains idle in this slot; otherwise, the user transmits the current unit to the receiver in this slot. If the transmitted unit is delivered successfully, the user attempts to transmit the next unit in the next slot; otherwise, if the transmitted unit is not delivered successfully (i.e., a collision happens), the user restarts the backoff process by randomly selecting an integer value b from $[0, D-1]$ as the new backoff time. The detailed behavior of each user is shown in Algorithm 1. We remark that delay-constrained CSMA is more complicated than traditional delay-unconstrained CSMA. We thus present Algorithm 1 to explicitly show all the details of our delay-constrained CSMA protocol.

Markov Chain Analysis: For the given number of users N, packet size L, and hard delay D, we present a Markov chain to analyze the exact theoretical performance of CSMA. In particular, the system state in any slot is denoted by

$$s = [(b_1, l_1, t), (b_2, l_2, t), \ldots, (b_N, l_N, t)], \tag{6}$$

where $b_i \in \{0, 1, \ldots, D-1\}$ is the backoff-time value of user i at the current slot, $l_i \in \{0, 1, \ldots, L\}$ is the number of units that have been delivered successfully to the receiver before the current slot and $t \in \{0, 1, 2, \ldots, D, D+1\}$ is the index of the current slot relative to the beginning of this period. Similar to ALOHA, we construct a virtual slot $t = D+1$ to indicate the end of a period. In addition, we construct a virtual state

$$s_0 = [(b_1, l_1, t), \ldots, (b_N, l_N, t)] = [(0, 0, 0), \ldots, (0, 0, 0)]$$

to indicate the start of a period. This means that we construct another virtual slot $t = 0$ for each period. The state space of CSMA, denoted by $\mathcal{S}_{\text{CSMA}}$, is of size

$$1 + [D(L+1)]^N \cdot (D+1). \tag{7}$$

2. For simplicity, we use CSMA to represent CSMA/CA in the rest of this paper.

3. In fact, at some slot, if the packet has remaining $l_{\text{remaining}}$ units and it has experienced t_{elapsed} slots after its arrival, the backoff time b should not be chosen from $[D - t_{\text{elapsed}} - l_{\text{remaining}} + 1, D - t_{\text{elapsed}}]$. Otherwise, it is not possible to completely deliver all the remaining units of the packet before expiration. However, we do not strictly follow the feasible backoff region but enlarge the backoff region to $[0, D-1]$. The reason is that the backoff region $[0, D-1]$ allows some packets to be expired so as to reduce the competition. Our independent investigation finds that such design can improve the system performance. In addition, the backoff region $[0, D-1]$ is simple to implement. Thus, we adopt it in this paper.

Algorithm 1 CSMA Protocol of a User

```
Require: Hard delay D, packet size L
Set b = 0, u = 1
for t = 1, 2, ..., do
  if (t − 1) mod D = 0 then
    Randomly select an integer b from [0, D − 1]
    Set u = 1
    if b = 0 then
      Transmit unit u to the receiver in slot t
      if The unit is delivered successfully then
        Set u = u + 1
      else
        Randomly select an integer b from [0, D − 1]
      end if
    else
      Perform carrier sensing in slot t
      if The channel is idle then
        Set b = b − 1
      end if
    end if
  else
    if u = L + 1 or L − u + 1 > D − [(t − 1) mod D]
    then
      Remain idle
    else
      if b = 0 then
        Transmit unit u to the receiver in slot t
        if The unit is delivered successfully then
          Set u = u + 1
        else
          Randomly select an integer b from [0, D−1]
        end if
      else
        Perform carrier sensing in slot t
        if The channel is idle then
          Set b = b − 1
        end if
      end if
    end if
  end if
end for
```

For the initial state s_0, the transition probability to state

$$s = [(b_1, 0, 1), \ldots, (b_N, 0, 1)], \forall b_i \in \{0, 1, \ldots, D-1\}$$

is $(\frac{1}{D})^N$.

For any state $s = [(b_1, l_1, t), (b_2, l_2, t), \ldots, (b_N, l_N, t)]$ where $t \in \{1, 2, \ldots, D\}$, we divide the user set into three sets,

$$\mathcal{N}_1 = \{i : l_i = L, i = 1, \ldots, N\} \cup \{i : L - l_i > D - [(t-1) \mod D], i = 1, \ldots, N\},$$
$$\mathcal{N}_2 = \{i : b_i = 0, i \in \{1, 2, \ldots, N\} \backslash \mathcal{N}_1\},$$
$$\mathcal{N}_3 = \{i : b_i > 0, i \in \{1, 2, \ldots, N\} \backslash \mathcal{N}_1\}.$$

Any user $i \in \mathcal{N}_1$ either has already successfully delivered the packet or cannot complete the delivery even it transmits in all the rest of slots before the end of the period. Thus, any user $i \in \mathcal{N}_1$ remains idle in this slot. The backoff-time value of any user $i \in \mathcal{N}_2$ is 0, i.e., $b_i = 0$, and thus it transmits the current unit in this slot. The backoff-time value of any user $i \in \mathcal{N}_3$ is larger than 0, i.e., $b_i > 0$, and thus it performs carrier sensing.

Let us denote the state in the next slot by

$$s' = \left[\left(b'_1, l'_1, t+1\right), \left(b'_2, l'_2, t+1\right), \ldots, \left(b'_N, l'_N, t+1\right)\right],$$

which depends on the value of $|\mathcal{N}_2|$. We initialize $b'_i = b_i, l'_i = l_i, \forall i = 1, 2, \ldots, N$ in state s' and then discuss which of them will be changed (the changing parts of state s'). If $|\mathcal{N}_2| = 0$, no user transmits data in this slot. Thus, all users in $\mathcal{N}_3$ sense to know that the channel is idle and decrease the backoff-time value by 1. Thus, the updated portion of state s' is

$$b'_i = b_i - 1, \ \forall i \in \mathcal{N}_3.$$

The transition probability is $P(s'|s) = 1$. If $|\mathcal{N}_2| = 1$, only one user $i \in \mathcal{N}_2$ transmits unit $l_i + 1$ in this slot. Thus, no collision happens and the transmission is successful. In addition, since the channel is busy, the backoff-time values of all users in $\mathcal{N}_3$ are frozen. Thus, the updated portion of state s' is

$$l'_i = l_i + 1, \ i \in \mathcal{N}_2.$$

Again, the transition probability is $P(s'|s) = 1$. If $|\mathcal{N}_2| > 1$, all users (more than one) in $\mathcal{N}_2$ transmit data in this slot. Thus, a collision happens and all transmissions fail. Therefore, each of all users in $\mathcal{N}_2$ restarts the backoff process by randomly selecting a backoff-time value from $[0, D-1]$. In addition, since the channel is busy, the backoff-time values of users in $\mathcal{N}_3$ are frozen. Therefore, the updated portion of state s' is

$$b'_i \in \{0, 1, \ldots, D-1\}, \ \forall i \in \mathcal{N}_2.$$

There are in total $D^{|\mathcal{N}_2|}$ possibilities for state s' and the transition probability is $P(s'|s) = \frac{1}{D^{|\mathcal{N}_2|}}$ for each possible s'.

Now we have a transition matrix $\boldsymbol{P} = [P(s'|s) : s, s' \in \mathcal{S}_{\mathsf{CSMA}}]$. Next we give an initial state distribution (in the beginning of virtual slot 0) $\boldsymbol{\pi}^0 = (\pi_s^0 : s \in \mathcal{S}_{\mathsf{CSMA}})$ as

$$\pi_s^0 = \begin{cases} 1, & \text{If } s = [(0,0), (0,0), \ldots, (0,0)]; \\ 0, & \text{Otherwise.} \end{cases}$$

Then the state distribution in the beginning of slot 1 is

$$\boldsymbol{\pi}^1 = \boldsymbol{\pi}^0 \boldsymbol{P}.$$

Similarly, we can obtain the state distribution in the beginning of the virtual slot $D+1$ as

$$\boldsymbol{\pi}^{D+1} = \boldsymbol{\pi}^0 \boldsymbol{P}^{D+1}. \quad (8)$$

Based on the state distribution $\boldsymbol{\pi}^{D+1}$, we can compute the timely throughput of user i by

$$R_i = \frac{L}{D} \cdot \sum_{\substack{s=[(b_1,l_1,D+1),\ldots,(b_i,l_i,D+1),\ldots,(b_N,l_N,D+1)]:l_i=L,\\ l_j\in\{1,2,\ldots,L\},\forall j\neq i, b_j\in\{0,1,\ldots,D-1\},\forall j\in\{1,2,\ldots,N\}}} \pi_s.$$

Then we can obtain the exact theoretical system timely throughput $R = \sum_{i=1}^{N} R_i$.

Note that the achieved system timely throughput R depends on the number of users N, hard delay D, and packet size L. To show this dependance clearly, we denote $R_{\text{CSMA}}(N, D, L)$ as the exact theoretical system timely throughput of CSMA protocol for given N, D and L.

IV. A LEARNING-BASED LOW-COMPLEXITY APPROXIMATE APPROACH

In the previous section, we use Markov-chain analysis to obtain the exact system timely throughput for both ALOHA and CSMA protocols. However, when we solve the equations (5) and (8) to obtain the final-slot state distributions for ALOHA and CSMA, respectively, the computational complexity exponentially increases with respect to the total number of users, i.e., N. This is because the number of states for both ALOHA and CSMA exponentially increase with respect to N, as shown in (3) and (7). Therefore, we can only apply the exact approach in the previous section for small-scale networks. To compare ALOHA and CSMA broadly, we need to figure out how to evaluate the system timely throughput for practical medium-scale networks, say, up to 50 users.

In this section, we propose a learning-based low-complexity approach to obtain an approximate value of the exact system timely throughput for both ALOHA and CSMA. Note that the exact system timely throughput for ALOHA (resp. CSMA), i.e., $R^*_{\text{ALOHA}}(N, D, L)$ (resp. $R_{\text{CSMA}}(N, D, L)$), depends on three parameters, the number of users N, the hard delay D, and the packet size L. If we can learn functions $R^*_{\text{ALOHA}}(N, D, L)$ and $R_{\text{CSMA}}(N, D, L)$, based on given networks, we can predict the system performance for any input network. This is the basic idea of the proposed learning-based low-complexity approach. Note that the function types could be very arbitrary and we also have a large number of machine learning approaches.

We next propose our learning approach by leveraging Markov-chain structures of ALOHA and CSMA. The basic idea is as follows. The exponential complexity of the exact approach comes from the number of users N. We thus try to reduce the total number of users but find a way to maintain the structure of the multi-user Markov chains. Note that the behavior pattern of each user is homogeneous in our considered multi-user wireless access system. Thus, it is reasonable to use a single user's performance to mimic the system performance. Such a technique has already been adopted by some existing works [53]–[56]. More concretely, next we will respectively construct two parameterized single-user Markov chains to mimic the original multi-user Markov chains for ALOHA and CSMA where the parameters need to be learned.

A. ALOHA

For ALOHA, we construct a single-user Markov chain parameterized by a transmission probability $p_t \in [0, 1]$ and a success probability $p_s \in [0, 1]$. The single-user Markov chain behaves as follows. In each slot of a period, if the packet of the user in this period has not been successfully delivered to the receiver, the user transmits the current unit to the receiver with probability p_t. The transmitted unit is then delivered successfully with probability p_s. The transmission probability p_t mimics the transmission probability p in the original multi-user ALOHA system, while the success probability p_s mimics the potential collision from other users in the original multi-user ALOHA system, as shown in Section III-A. Clearly, we can construct a Markov chain for this particular user similar to Section III-A. We obtain the final-slot state distribution to calculate the timely throughput, denoted by $R^{\text{Approx}}_{\text{ALOHA}}(p_t, p_s)$. The total number of state is $(L+1)(D+1)$, which is linear with L and D, regardless of N. Thus, the computational complexity is exponentially reduced. Our goal is to estimate the achieved timely throughput of the original multi-user ALOHA system for given N, D, L, i.e., $R^*_{\text{ALOHA}}(N, D, L)$, from the achieved timely throughput of this single-user system. Namely, for given N, D, L, we aim at finding suitable p_t and p_s to minimize

$$\left| N \cdot R^{\text{Approx}}_{\text{ALOHA}}(p_t, p_s) - R^*_{\text{ALOHA}}(N, D, L) \right|.$$

The reason that we use two parameters p_t and p_s to construct the single-user Markov chain for ALOHA is because the combination of p_t and p_s reasonably mimics the original multi-user ALOHA system. Clearly, when a packet has at least one remaining unit, a packet will deliver one unit successfully if the user transmits a packet (with probability p_t) and the packet is transmitted successfully (with probability p_s). Thus, a packet will deliver one unit successfully with probability $p_t p_s$ if it has at least one remaining unit. The transition probability can be obtained by replacing p in (4) with $p_t p_s$. Thus, the transition matrix of the single-user Markov chain only depends on the product of p_t and p_s, but does not depend on the individual value of p_t or p_s. Therefore, the achieved timely throughput $R^{\text{Approx}}_{\text{ALOHA}}(p_t, p_s)$ also only depends on the product of p_t and p_s. Without loss of generality, next we assume that $p_t = 1$, i.e., the single user transmits the current unit in all slots for sure. We then denote the achieved timely throughput by $R^{\text{Approx}}_{\text{ALOHA}}(p_s)$.

Thus, for given N, D, L, we need to find suitable p_s to minimize

$$\left| N \cdot R^{\text{Approx}}_{\text{ALOHA}}(p_s) - R^*_{\text{ALOHA}}(N, D, L) \right|. \tag{9}$$

Namely, we need to learn function

$$p_s = f_1(N, D, L). \tag{10}$$

We construct the dataset as follows. For given D, L, and a small N, we obtain the exact system timely throughput, i.e., $R^*_{\text{ALOHA}}(N, D, L)$, by using the multi-user Markov-chain analysis in Section III-A. We find the best p_s to minimize (9) via the binary-search approach (since $R^{\text{Approx}}_{\text{ALOHA}}(p_t, p_s)$ strictly increases as p_s increases). Then we obtain a dataset

$$\left(N^i, D^i, L^i, p_s^i\right), \quad i = 1, 2, \ldots, K,$$

where K is the total number of data points and i is the index of a data point in the dataset. We apply machine learning approaches to predict function p_s as shown in (10).

Now given D, L, and a large N, we can use our learned model to predict p_s and then construct a single-user Markov chain to obtain $N \cdot R^{\text{Approx}}_{\text{ALOHA}}(p_s)$, which serves as an approximate value of $R^*_{\text{ALOHA}}(N, D, L)$.

B. CSMA

For CSMA, we construct a single-user Markov chain parameterized by a channel-busy probability $p_b \in [0, 1]$ and a collision probability $p_c \in [0, 1]$. The single user follows the same steps in Algorithm 1 except the followings. First, when the user performs carrier sensing as shown in lines 14 and 31 in Algorithm 1, the channel is busy with probability p_b. Second, when the user transmits data as shown in lines 7 and 24, a collision happens with probability p_c and thus the transmitted unit can be delivered successfully with probability $1 - p_c$. The channel-busy probability p_b and the collision probability p_c mimic the potential transmissions from other users in the original multi-user CSMA system, as shown in Section III-B. Clearly, we can construct a Markov chain for this single user similar to Section III-B. We remark that the notations p_b and p_c were also used for simplifying the analysis of delay-unconstrained CSMA [57], but our model here is for delay-constrained traffic. We then obtain the final-slot state distribution to calculate the timely throughput, denoted by $R^{\text{Approx}}_{\text{CSMA}}(p_b, p_c)$. The total number of state is $1 + D(L + 1)(D + 1)$, which is linear with L and quadratic with D, regardless of N. Again, the computational complexity is exponentially reduced. Our goal is to estimate the achieved timely throughput of the original multi-user CSMA system for given N, D, L, i.e., $R_{\text{CSMA}}(N, D, L)$, from the achieved timely throughput of this single-user system.

Thus, we need to predict two parameters p_b and p_c for given parameters N, D and L, i.e.,

$$p_b = f_2(N, D, L),$$

and

$$p_c = f_3(N, D, L).$$

The learning process for p_b and p_c is similar to that for p_s in ALOHA as shown in Section IV-A.

Now given D, L , and a large N, we can use our learned model to predict p_b and p_c and then construct a single-user Markov chain to obtain $N \cdot R^{\text{Approx}}_{\text{CSMA}}(p_b, p_c)$, which serves as an approximate value of $R^*_{\text{CSMA}}(N, D, L)$.

C. DIFFERENCES BETWEEN OUR APPROXIMATE APPROACHES AND BIANCHI'S WORK [33]

Bianchi's paper [33] is the seminal work for analyzing the performance of CSMA used in IEEE 802.11. Bianchi constructed a single-user system to approximate the original multi-user CSMA system with the key assumption that each packet collides with constant and independent probability p. This inspires our approximate approaches in this section. However, our approaches work for frame-synchronized traffic pattern, which is significantly different from the full-buffer delay-unconstrained traffic pattern in [33].

First, in our frame-synchronized pattern, each user has a new packet arrival every D slots. Thus, it is possible that a user does not have any packet to transmit in a slot. Once a packet has been delivered in a frame, the user has to remain idle in the remaining slots of the frame. That's why we added l_i to record the transmitted units of a packet in the state of the Markov chain (see Eq. (2) and Eq. (6)). However, in [33] with full-buffer traffic, there is no need to record whether there is a packet or not in the queue because each user always has content to transmit.

Second, under our delay-constrained traffic patter, a packet will expire and be removed from the system after its deadline. Thus, we need to record when a packet will expire. That's why we added t to record the elapsed time in the state of the Markov chain (see Eq. (2) and Eq. (6)). However, in [33] with delay-unconstrained traffic, there is no need to record how much time a packet has experienced because a packet can be kept in the queue for however long time.

Finally, the slot duration in Bianchi's paper could be either the constant backoff slot time (e.g., the duration of slot 8, 7, 6, 4, 3, 2, 1 in [33, Fig. 1]) or the variable time in which a packet is transmitted (e.g., the duration of slot 5 in [33, Fig. 1]). Bianchi did not differentiate these two cases and constructed a Markov chain (i.e., [33, Fig. 4]). In other words, in Bianchi's Markov chain, a state transition could experience a constant backoff slot time or a variable packet-transmission time. Such modeling is applicable to delay-unconstrained setting but not applicable to our delay-constrained setting. In our delay-constrained setting, we need to know exactly how much time a packet has experienced. That's why we discretize the system into equal-length slots and the state of our constructed Markov chain is captured at the beginning of each slot. In addition, in our constructed Markov chain for the approximate CSMA system in Section IV-B, we use two parameters, p_b and p_c, to characterize the system behaviors. Parameter p_c is the collision probability, which is exactly parameter p in Bianchi's work [33]. But we have an extra parameter p_b, which is the channel-busy probability. It can model the busy behaviour of slot 5 in [33, Fig. 1] but in the granularity of constant backoff slot time.

Therefore, we can see that our constructed Markov chain is significantly different from that in Bianchi's paper [33]. Under the full-buffer delay-unconstrained traffic patter in [33], Bianchi obtained the closed-form stationary distribution and then got the closed-form of the transmission probability $\tau = \tau(p)$, i.e., [33, eq. (7)]. After that, Bianchi connected the single-user approximate system to the original multi-user system by [33, eq. (9)]. Solving this equation, Bianchi obtained the collision probability p for the approximate single-user system. However, under our frame-synchronized delay-constrained traffic patter, we cannot obtain the closed-form stationary distribution of our constructed Markov chain, which is significantly different from Bianchi's Markov chain in [33]. As a result, we cannot obtain the parameters of our approximate single-user systems by solving an equation like [33, Eq. (9)]. Instead, in this paper, we resort to use data-driven approaches to estimate such parameters, as shown in Section VI-B .

V. AVERAGE DELIVERY TIME

As we analyzed in Section IV, our proposed learning-based approximate approach can exponentially reduce the complexity of our considered delay-constrained ALOHA and CSMA systems. Furthermore, another good property of our approach is that we still preserve the Markov-chain structure of the systems. This enables us to compute more performance metrics in addition to the system average timely throughput. One important metric is the average delivery time of those packets that have been delivered successfully before expiration. Even though we consider a delay-constrained system where a packet remains valid if it is delivered before its expiration, smaller delivery time is still preferred in many applications. For example, in wireless NCSs, the remote controller can control the plant more accurately if the deliver time of the control message is smaller.

Since our ALOHA and CSMA systems are frame-by-frame stationary, we focus on a particular frame. Denote random variable X by the delivery time of the packet in this frame, which is the number of slots from its arrival to its departure. The range of X is $1, 2, \cdots, D, D+1, \cdots$. Note that we allow $X > D$, which means that the packet has not delivered successfully before expiration. Namely, we have

$$\sum_{k=1}^{D} P(X=k) + P(X>D) = 1. \tag{11}$$

Note that we compute the average delivery time of those packets that have been delivered successfully before expiration. Mathematically, this means that we need to compute

$$\begin{aligned}\text{Average delivery time} &= \mathbb{E}[X|X \le D] \\ &= \sum_{k=1}^{D} k \cdot P[X=k|X\le D] \\ &= \sum_{k=1}^{D} k \cdot \frac{P(X=k)}{\sum_{k'=1}^{D} P(X=k')}. \end{aligned} \tag{12}$$

For both ALOHA and CSMA systems, we can obtain $P(X=k)$ based on $\boldsymbol{\pi}^t, t = 1, 2, \ldots, D, D+1$ which is the state distribution of slot t of our constructed single-user

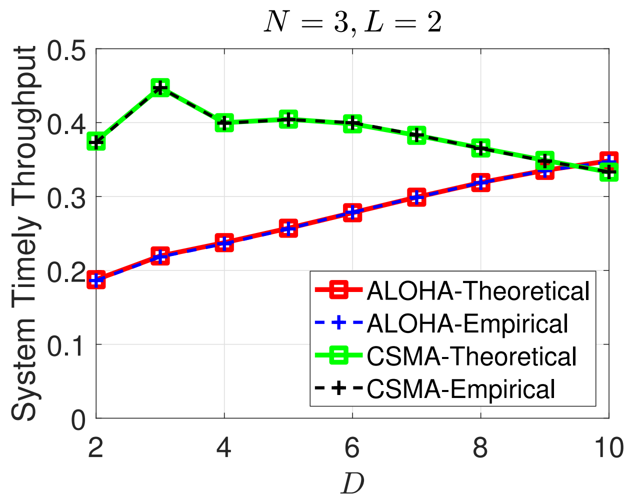


FIGURE 3. The system timely throughput of $N = 3$ and $L = 2$.

(approximate) Markov chain in Section IV. In particular, for the single-user ALOHA system, as shown in (2), the state is denoted as (l, t) which means that l units of the packet have been delivered before slot t. Thus, the delivery time X is not greater than k if and only if the whole packet has been delivered before slot $k+1$, i.e.,

$$P(X \le k) = \pi^{k+1}_{(L,k+1)}. \tag{13}$$

For CSMA system, as shown in (6), the state is denoted as (b, l, t) which means that the backoff-time value of the packet is b and l_i units of the packet has been delivered before slot t. Again, the delivery time X is not greater than k if and only if the whole packet has been delivered before slot $k+1$, i.e.,

$$P(X \le k) = \sum_{b=0}^{D-1} \pi^{k+1}_{(b,L,k+1)}. \tag{14}$$

Based on $P(X \le k)$ as shown in (13) for ALOHA and (14) for CSMA, we can compute

$$P(X=k) = P(X\le k) - P(X \le k-1), \ \forall k = 1, 2, \ldots, D \tag{15}$$

Then we can obtain the average delivery time according to (12).

VI. NUMERICAL RESULTS

In this section, we conduct extensive simulations to illustrate the performance of our proposed learning-based low-complexity approach. We implement all algorithms and evaluate their performances using MATLAB and Python languages. All evaluations are conducted in a computer with two CPUs (Intel Xeon E5-2678 v3), one GPU (NVIDIA GeForce GTX 2080 Ti), and 64GB memory, running Ubuntu 16.04.6 LTS. All source code and dataset are publicly available in https://github.com/yuyouzhi/compare.dc.aloha.and.csma.

TABLE 1. The cost value for predicting parameters in ALOHA and CSMA under different machine learning approaches.

Cost	p_s for ALOHA			p_c for CSMA		
	Linear Regression	Neural Network	SVR Regression	Linear Regression	Neural Network	SVR Regression
$L=1$	0.5717	0.0059	3.481×10^{-4}	0.1783	0.0026	4.790×10^{-4}
$L=2$	0.0123	0.0030	7.384×10^{-4}	0.1791	0.0022	1.748×10^{-4}
$L=3$	0.0060	0.0053	9.431×10^{-5}	0.1327	0.0031	1.455×10^{-4}
$L=4$	0.3044	0.0044	2.424×10^{-5}	0.1844	0.0034	9.012×10^{-5}
$L=5$	0.2638	0.0035	4.175×10^{-5}	0.0608	0.0046	6.090×10^{-5}

A. CONFIRM OUR EXACT THEORETICAL ANALYSIS

In Section III, we used Markov-chain analysis to obtain the theoretical system timely throughput for both ALOHA and CSMA. We need to confirm the correctness of this exact approach. We consider $N = 3, L = 2$ and vary D from 2 to 10. We first use the proposed exact approach to get the theoretical system timely throughput. We then simulate a multi-user ALOHA system and a multi-user CSMA system to get the empirical system timely throughput by running 100,000 periods. The results are shown in Fig. 3. As we can see, the theoretical values match well with the empirical values for both ALOHA and CSMA. This confirms the correctness of our exact theoretical analysis in Section III.

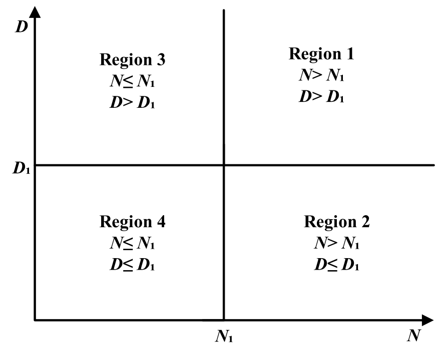


FIGURE 4. The four regions to divide the (N, D) plane.

B. LEARNING RESULTS OF THE APPROXIMATE APPROACH

In our approximate approach, we need to learn parameter p_s for ALOHA and learn parameters p_b and p_c for CSMA with respect to N, D, and L. We have collected 9,065 data points for both ALOHA and CSMA. Note that due to the limitations of our computation resources and the exponential complexity of the exact theoretical analysis, we can only get the theoretical system timely throughput for small-scale networks. To enlarge the dataset, we further use the empirical system timely throughput via simulation to collect more data points for practical medium-scale networks. We remark that we assume that the network has at most $N = 50$ users. It is a relatively moderate-size network and can capture a wide range of practical scenarios. For example, although most WiFi APs can theoretically support up to 250 users, it is generally suggested that at most 50 users connect to the same WiFi AP simultaneously in typical scenarios to ensure reasonable user experience.[4,5] In fact, in Bianchi's seminal work [33] for analyzing the performance of the DCF mechanism of IEEE 802.11, all simulations were also conducted in a network with at most 50 users; see [33, Figs. 7–10]. We then randomly choose 80% of the dataset to be the training dataset and leave the rest 20% as the test dataset. We remark that the packet size is generally not very large due to the maximum transmission unit (MTU) limit. For example, the MTU of a MAC packet of Wi-Fi is 2304 bytes [58, Table 9–25]. For this reason, we assume that $L \leq 5$ in this paper. Since there are only 5 possible values for the packet size L, next we learn the parameters in the approximate approach for each $L \in \{1, 2, 3, 4, 5\}$ separately.

In addition, for approximate CSMA, we need to predict two parameters p_b and p_c. Note that p_b and p_c jointly affect the system timely throughput $R^{\mathsf{Approx}}_{\mathsf{CSMA}}(p_b, p_c)$. We then observe that it is possible that $R^{\mathsf{Approx}}_{\mathsf{CSMA}}(p_b, p_c) = R^{\mathsf{Approx}}_{\mathsf{CSMA}}(p'_b, p'_c)$ even if $p'_b \neq p_b$ and $p'_c \neq p_c$. This results in ambiguity for predicting p_b and p_c. In fact, we have used the dataset to learn both p_b and p_c. However, due to the aforementioned ambiguity, the prediction error is relatively large. We thus preprocess the dataset as follows. For each $L \in \{1, 2, 3, 4, 5\}$, we divide the (N, D) plane into four regions with separation lines $N = N_1$ and $D = D_1$, as shown in Fig. 4. For each region $i \in \{1, 2, 3, 4\}$, we select a best $p_b \in [0, 1]$ to minimize

$$\delta_i(N_1, D_1) = \sum_{(N,D) \text{ in region } i} \min_{p_c \in [0,1]} \times \left| N \cdot R^{\mathsf{Approx}}_{\mathsf{CSMA}}(p_b, p_c) - R_{\mathsf{CSMA}}(N, D, L) \right|,$$

where $R^{\mathsf{Approx}}_{\mathsf{CSMA}}(p_b, p_c)$ is the achieved timely throughput of the approximate single-user CSMA system with parameters p_b and p_c, and $R_{\mathsf{CSMA}}(N, D, L)$ is the achieved system timely throughput of the original multi-user CSMA system, as described in Section IV-B. For each separation (N_1, D_1), we can get a total cost $\delta(N_1, D_1) = \sum_{i=1}^{4} \delta_i(N_1, D_1)$. We then iteratively find the best separation (N_1, D_1) to minimize $\delta(N_1, D_1)$. After this preprocessing, for each L, we have fixed the separation and also fixed the parameter p_b for each of the four regions in Fig. 4. Then, we only need to learn parameter p_c for each region. This avoids the aforementioned ambiguity and improves the prediction accuracy.

4. https://www.lifewire.com/how-many-devices-can-share-a-wifi-network-818298

5. https://www.fusionconnect.com/blog/blog-archive/too-many-devices-on-wifi-how-to-identify-and-correct-limited-wifi-connections

TABLE 2. The cost value for predicting the theoretical system timely throughput under our best approach for both ALOHA and CSMA schemes.

Scheme	Cost of Our Best Approach
ALOHA	1.146×10^{-4}
CSMA	6.643×10^{-4}

After the preprocessing, we can use many possible machine learning approaches. In this paper, we consider three classic approaches: linear regression, neural network, and support vector regression (SVR) [59]–[61]. The cost function is defined as the mean square error. Taking p_s as an example, the cost function is

$$J = \frac{1}{K}\sum_{i=1}^{K}\left[\bar{p}_s^i - p_s^i\right]^2, \tag{16}$$

where K is the total number of data points in the test dataset, $\bar{p}_s^i$ is the predicted value, and p_s^i is the true value. All three machine learning approaches could have different configurations. For linear regression, we consider both the first-order linear regression and the second-order linear regression. For neural network, we range the number of layers from one to three. For support vector regression (SVR), we consider three kernels: linear kernel, polynomial kernel, and radial basis function (RBF) kernel. We report the best-configuration results of these different learning approaches in Table 1. We remark that it is possible to apply high-order linear regressions and multi-layer neural networks (even deep learning with many layers). However, based on our investigation, such approaches do not have benefits but lead to over-fitting. From Table 1, we can see that the SVR is the best learning approach for both predicting p_s for ALOHA and predicting p_c for CSMA.

Although we have predicted parameters for ALOHA and CSMA, they are used to approximate the theoretical system timely throughput. Thus, we need to evaluate the accuracy of our proposed approximate approach. Again, we use the mean square error as the performance metric. Given N, D, L, for ALOHA (resp. CSMA) system, we use the single-user Markov chain parameterized by the best-learned success probability $\bar{p}_s$ (resp. the best channel-busy probability $\bar{p}_b$ and the best-learned collision probability $\bar{p}_c$) to obtain the timely throughput, N times of which serves as the predicted (approximate) value for the theoretical system timely throughput of the multi-user ALOHA (resp. multi-user CSMA) system. The results are shown in Table 2. As we can see, our prediction is very accurate with mean-square error in the order of 10^{-4} for both ALOHA and CSMA. This shows the effectiveness of our approach.

C. COMPARE ALOHA AND CSMA

Our goal of this paper is to compare ALOHA and CSMA for a broad number of system settings. With the help of our learning-based low-complexity approximate approach, we can compare ALOHA and CSMA for different N, D, L. It would be difficult to plot and view the three-dimension

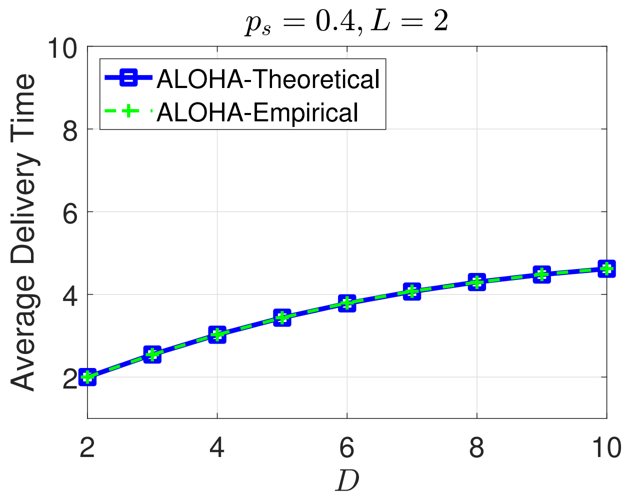


FIGURE 5. The average delivery time of ALOHA when $p_s = 0.4$ and $L = 2$.

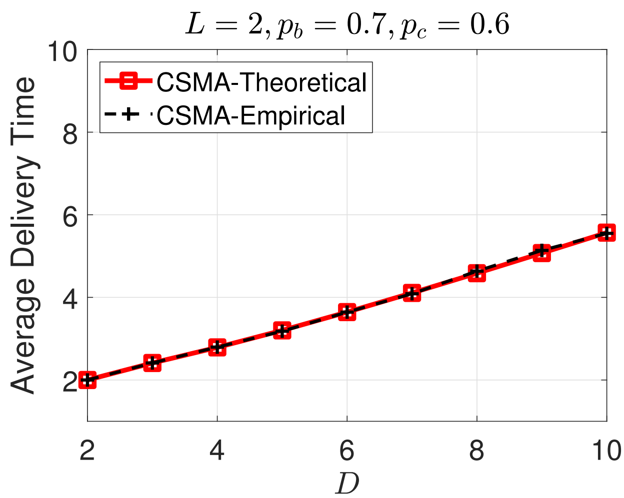


FIGURE 6. The average delivery time of CSMA when $p_b = 0.7$, $p_b = 0.6$ and $L = 2$.

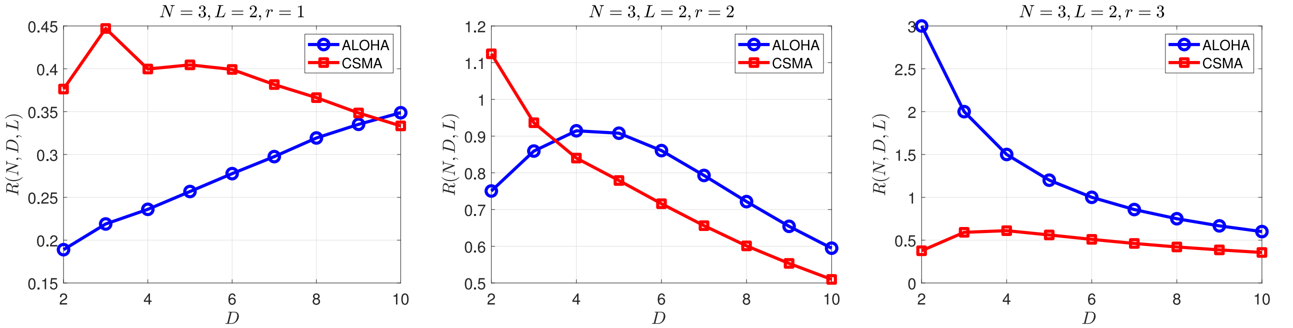
results. As we mentioned in Section VI-B, we consider $L \leq 5$ in this paper. We thus plot the results in the (N, D) plane for each $L \in \{1, 2, 3, 4, 5\}$. The results are shown in Fig. 9. We have the following observations.

For $L = 1$, ALOHA outperforms CSMA when N is small ($N \leq 25$), while CSMA generally outperforms ALOHA when N is large ($N > 25$) except a small region when $30 \leq N \leq 35$ and $D \leq 20$. Overall, we can see that when $L = 1$, the number of users, i.e., N, has bigger impact than the delay D.

For $L > 1$, CSMA outperforms ALOHA in the majority of cases. Only when N is small and D is large, ALOHA is better than CSMA. In addition, as L increases, the percentage of cases that CSMA is better also increases. This suggests that CSMA benefits from large packet size L.

D. THE AVERAGE DELIVERY TIME OF ALOHA AND CSMA

Based on our proposed low-complexity approach, we have showed how to theoretically calculate the average delivery



(a) $\gamma = 1$ (b) $\gamma = 2$ (c) $\gamma = 3$

FIGURE 7. **The system timely throughput of delay-constrained ALOHA and CSMA under MPR channel model.**

time for both ALOHA and CSMA system in Section V. In this part, we first confirm the analysis in Section V by comparing its result with empirical result. The results for ALOHA and CSMA are shown in Fig. 5 where $p_s = 0.4$ and $L = 2$ and Fig. 6 where $p_b = 0.7$, $p_c = 0.6$, $L = 2$, respectively. For both figures, we can see that indeed our theoretical analysis matches well with the empirical result. This confirms the correctness of our analysis in Section V. In addition, we can see that the average delivery time increases as the hard delay D increases. This is an intuitive observation.

In addition, our analysis in Section V can be applied to compare the average delivery time of ALOHA and CSMA. In Fig. 8, we fix $N = 4$ and $L = 4$ and vary D from 5 to 29. We compare the average delivery time of ALOHA under the optimal p_s and CSMA under the optimal p_c and p_s. We can observe that the average delivery time of ALOHA is larger than that of CSMA.

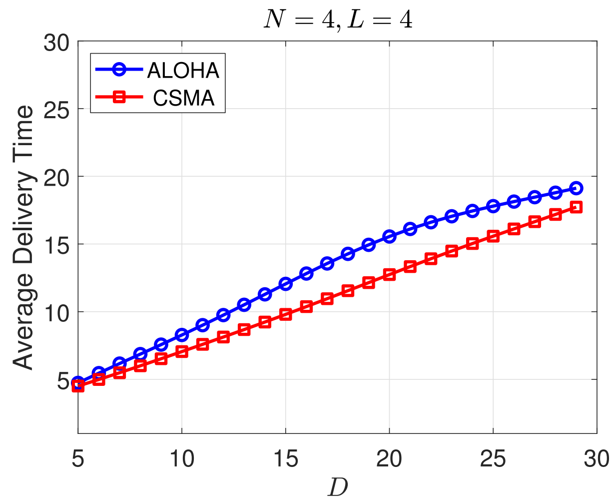


FIGURE 8. **The average delivery time of CSMA and ALOHA when $N = 4$ and $L = 4$.**

E. MPR CAPABILITY

In this paper, we only consider the traditional model of a single-packet reception (SPR) channel. Namely, a packet can be correctly received if and only if there is no other packet transmissions during its transmission. However, the new physical (PHY) layer techniques enables a single PHY channel to accommodate multiple concurrent transmissions, which is called multiple-packet reception (MPR) capability [36], [43], [62]. According to our previous research on random access protocols with MPR capability [36], [43], [62], the random access protocol design with MRP is very different from that with SPR. Thus, as a first step to compare ALOHA and CSMA under the new delay-constrained settings, we follow traditional SPR model in this paper.

In this subsection, to gain some basic understandings on MRP model, we show some simulation results. We assume that the MPR capability is denoted by parameter γ, which is the maximal number of collided packets that can be received/decoded by the AP. We compare the achieved system timely throughput of both ALOHA and CSMA for $\gamma = 1$ (i.e., our current SPR model), $\gamma = 2$ and $\gamma = 3$, as shown in Fig. 7. We can see that when the MRP capability increases, both ALOHA and CSMA achieve better system timely throughput but ALOHA achieves higher improvement than CSMA. It deserves further investigations to explain such observations and study MPR model comprehensively, and we will leave it as a future direction.

VII. CONCLUSION

In this paper, we have applied Markov chains to analyze the theoretical system performance for delay-constrained ALOHA and CSMA systems. Since the number of states of Markov chains grows exponentially, the exact approach can only be applied for small-scale networks. We have thus exploited the structures of ALOHA and CSMA protocols and proposed a learning-based low-complexity approximate approach. Our numerical results have shown the effectiveness of our proposed approximate approach. We have further used this approximate approach to compare ALOHA and CSMA for a broad number of system settings and summarized the conditions under which each of them outperforms the other one. We have also summarized the different settings under which ALOHA or CSMA is better as shown in

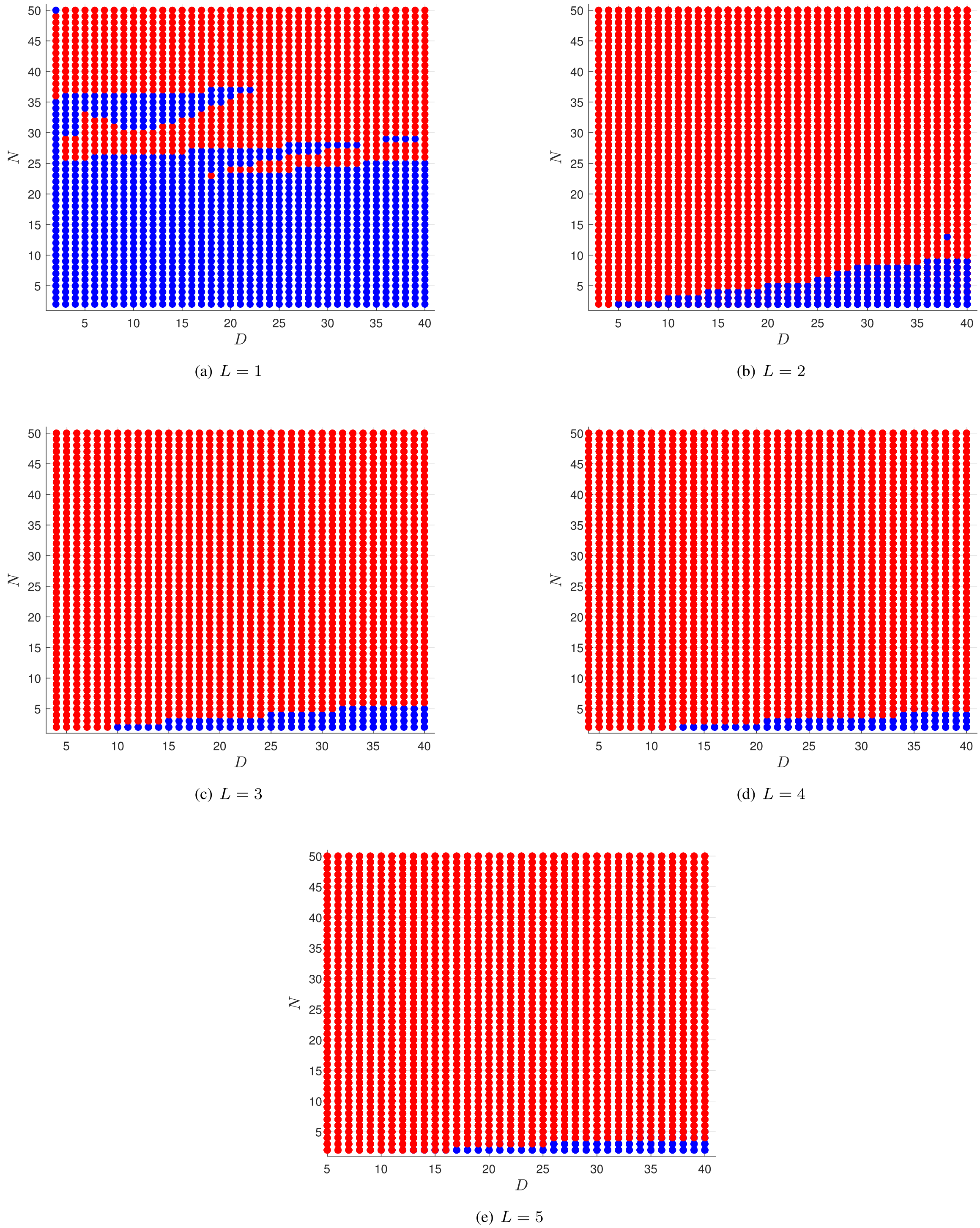


FIGURE 9. **Comparing ALOHA and CSMA for different settings where the red bullet (●) means that CSMA outperforms ALOHA and the blue bullet (●) means that ALOHA outperforms CSMA. We can see that for $L = 1$, ALOHA is better when N is small, and CSMA is better when N is large; for $L > 1$, ALOHA is better when N is small and D is large, and CSMA is better in other cases.**

Fig. 9. In the future, it is interesting and important to relax the frame-synchronized traffic pattern.

ACKNOWLEDGMENT

The authors thank the anonymous reviewers for their constructive suggestions to improve the quality of this paper.

REFERENCES

[1] Y. Yu, L. Deng, Y. Zhang, and Y. S. Han, "Comparing delay-constrained ALOHA and CSMA: A learning-based low-complexity approximate approach," in *Proc. IEEE 16th Int. Conf. Control Autom. (ICCA)*, Singapore, 2020, pp. 1346–1352.

[2] L. Deng, C.-C. Wang, M. Chen, and S. Zhao, "Timely wireless flows with general traffic patterns: Capacity region and scheduling algorithms," *IEEE/ACM Trans. Netw.*, vol. 25, no. 6, pp. 3473–3486, Dec. 2017.

[3] G. P. Fettweis, "The tactile Internet: Applications and challenges," *IEEE Veh. Technol. Mag.*, vol. 9, no. 1, pp. 64–70, Mar. 2014.

[4] M. Simsek, A. Aijaz, M. Dohler, J. Sachs, and G. Fettweis, "5G-enabled tactile Internet," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 3, pp. 460–473, Mar. 2016.

[5] J. Baillieul and P. J. Antsaklis, "Control and communication challenges in networked real-time systems," *Proc. IEEE*, vol. 95, no. 1, pp. 9–28, Jan. 2007.

[6] B. Demirel, Z. Zou, P. Soldati, and M. Johansson, "Modular design of jointly optimal controllers and forwarding policies for wireless control," *IEEE Trans. Autom. Control*, vol. 59, no. 12, pp. 3252–3265, Dec. 2014.

[7] K.-D. Kim and P. R. Kumar, "Cyber–physical systems: A perspective at the centennial," *Proc. IEEE*, vol. 100, no. Special Centennial Issue, pp. 1287–1308, May 2012.

[8] K. Kang, K.-J. Park, L. Sha, and Q. Wang, "Design of a crossbar VOQ real-time switch with clock-driven scheduling for a guaranteed delay bound," *Real-Time Syst.*, vol. 49, no. 1, pp. 117–135, 2013.

[9] L. Deng, W. S. Wong, P.-N. Chen, Y. S. Han, and H. Hou, "Delay-constrained input-queued switch," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2464–2474, Nov. 2018.

[10] I.-H. Hou, V. Borkar, and P. R. Kumar, "A theory of QoS for wireless," in *Proc. IEEE INFOCOM*, Rio de Janeiro, Brazil, 2009, pp. 486–494.

[11] I.-H. Hou and P. R. Kumar, "Utility maximization for delay constrained QoS in wireless," in *Proc. IEEE INFOCOM*, San Diego, CA, USA, 2010, pp. 1–9.

[12] S. Lashgari and A. S. Avestimehr, "Timely throughput of heterogeneous wireless networks: Fundamental limits and algorithms," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8414–8433, Dec. 2013.

[13] P. Popovski *et al.*, "Wireless access for ultra-reliable low-latency communication: Principles and building blocks," *IEEE Netw.*, vol. 32, no. 2, pp. 16–23, Mar./Apr. 2018.

[14] P.-C. Hsieh and I.-H. Hou, "A decentralized medium access protocol for real-time wireless ad hoc networks with unreliable transmissions," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Vienna, Austria, 2018, pp. 972–982.

[15] L. Deng, J. Deng, P.-N. Chen, and Y. S. Han, "On the asymptotic performance of delay-constrained slotted ALOHA," in *Proc. 27th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Hangzhou, China, 2018, pp. 1–8.

[16] E. Fountoulakis, T. Charalambous, N. Nomikos, A. Ephremides, and N. Pappas, "Information freshness and packet drop rate interplay in a two-user multi-access channel," in *Proc. IEEE Inf. Theory Workshop (ITW)*, 2020, pp. 1–5.

[17] N. Nomikos, N. Pappas, T. Charalambous, and Y.-A. Pignolet, "Deadline-constrained bursty traffic in random access wireless networks," in *Proc. IEEE 19th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Kalamata, Greece, 2018, pp. 1–5.

[18] Z. Chen, N. Pappas, M. Kountouris, and V. Angelakis, "Throughput with delay constraints in a shared access network with priorities," *IEEE Trans. Wireless Commun.*, vol. 17, no. 9, pp. 5885–5899, Sep. 2018.

[19] Y. H. Bae and J. W. Baek, "Performance analysis of delay-constrained traffic in a cognitive radio network with RF energy harvesting," *IEEE Commun. Lett.*, vol. 23, no. 12, pp. 2177–2181, Dec. 2019.

[20] N. Abramson, "The ALOHA system: Another alternative for computer communications," in *Proc. AFIPS Fall Joint Comput. Conf.*, 1970, pp. 281–285.

[21] L. G. Roberts, "ALOHA packet system with and without slots and capture," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 5, no. 2, pp. 28–42, 1975.

[22] V. Naware, G. Mergen, and L. Tong, "Stability and delay of finite-user slotted ALOHA with multipacket reception," *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2636–2656, Jul. 2005.

[23] 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.

[24] S.-R. Lee, S.-D. Joo, and C.-W. Lee, "An enhanced dynamic framed slotted ALOHA algorithm for RFID tag identification," in *Proc. 2nd Annu. Int. Conf. Mobile Ubiquitous Syst. Netw. Serv. (MobiQuitous)*, San Diego, CA, USA, 2005, pp. 166–172.

[25] Z. G. Prodanoff, "Optimal frame size analysis for framed slotted ALOHA based RFID networks," *Comput. Commun.*, vol. 33, no. 5, pp. 648–653, 2010.

[26] J. Yu and L. Chen, "Stability analysis of frame slotted ALOHA protocol," *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1462–1474, May 2017.

[27] 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.

[28] E. Casini, R. De Gaudenzi, and O. D. R. Herrero, "Contention resolution diversity slotted ALOHA (CRDSA): An enhanced random access schemefor satellite access packet networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 4, pp. 1408–1419, Apr. 2007.

[29] W. Rosenkrantz and D. Towsley, "On the instability of slotted ALOHA multiaccess algorithm," *IEEE Trans. Autom. Control*, vol. AC-28, no. 10, pp. 994–996, Oct. 1983.

[30] B. Hajek and T. van Loon, "Decentralized dynamic control of a multiaccess broadcast channel," *IEEE Trans. Autom. Control*, vol. AC-27, no. 3, pp. 559–569, Jun. 1982.

[31] R. R. Rao and A. Ephremides, "On the stability of interacting queues in a multiple-access system," *IEEE Trans. Inf. Theory*, vol. IT 34, no. 5, pp. 918–930, Sep. 1988.

[32] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*, 6th ed. Boston, MA, USA: Addison-Wesley, 2013.

[33] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[34] F. Cali, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," *IEEE/ACM Trans. Netw.*, vol. 8, no. 6, pp. 785–799, Dec. 2000.

[35] P. K. Wong, D. Yin, and T. T. Lee, "Analysis of non-persistent CSMA protocols with exponential backoff scheduling," *IEEE Trans. Commun.*, vol. 59, no. 8, pp. 2206–2214, Aug. 2011.

[36] Y. Zhang, A. Gong, Y.-H. Lo, J. Li, F. Shu, and W. S. Wong, "Generalized p-persistent CSMA for asynchronous multiple-packet reception," *IEEE Trans. Commun.*, vol. 67, no. 10, pp. 6966–6979, Oct. 2019.

[37] L. Dai, "Toward a coherent theory of CSMA and ALOHA," *IEEE Trans. Wireless Commun.*, vol. 12, no. 7, pp. 3428–3444, Jul. 2013.

[38] X. Sun and L. Dai, "To sense or not to sense: A comparative study of CSMA with ALOHA," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 7587–7603, Nov. 2019.

[39] Y. Yu, T. Wang, and S. C. Liew, "Deep-reinforcement learning multiple access for heterogeneous wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1277–1290, Jun. 2019.

[40] B. Yang, X. Cao, Z. Han, and L. Qian, "A machine learning enabled MAC framework for heterogeneous Internet-of-Things networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 7, pp. 3697–3712, Jul. 2019.

[41] H. B. Pasandi and T. Nadeem, "Unboxing MAC protocol design optimization using deep learning," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Austin, TX, USA, 2020, pp. 1–5.

[42] F. De Rango, N. Cordeschi, and F. Ritacco, "Applying Q-learning approach to CSMA scheme to dynamically tune the contention probability," in *Proc. IEEE 18th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, 2021, pp. 1–4.

[43] Y. Zhang, Y.-H. Lo, F. Shu, and J. Li, "Achieving maximum reliability in deadline-constrained random access with multiple-packet reception," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5997–6008, Jun. 2019.

[44] Y. Zhang, F. Guan, Y.-H. Lo, F. Shu, and J. Li, "Optimal multichannel slotted ALOHA for deadline-constrained unicast systems," *IEEE Syst. J.*, vol. 13, no. 2, pp. 1308–1311, Jun. 2019.

[45] C. Campolo, A. Molinaro, A. Vinel, and Y. Zhang, "Modeling prioritized broadcasting in multichannel vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 61, no. 2, pp. 687–701, Feb. 2012.

[46] N. Lu, B. Li, R. Srikant, and L. Ying, "Optimal distributed scheduling of real-time traffic with hard deadlines," in *Proc. IEEE 55th Conf. Decis. Control (CDC)*, Las Vegas, NV, USA, 2016, pp. 4408–4413.

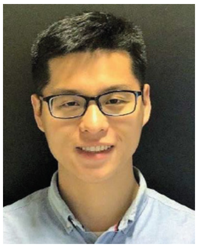
[47] *Evolved Universal Terrestrial Radio Access (E-UTRA); Evolved Universal Terrestrial Radio Access (E-UTRA); Layer 2—Measurements, Version 16.0.0*, 3GPP Standard (TS) 36.314, JUl. 2020. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/36_series/36.314/36314-g00.zip

[48] IEC/IEEE. (2021). *IEC/IEEE 60802 TSN Profile for Industrial Automation*. [Online]. Available: https://1.ieee802.org/tsn/iec-ieee-60802/

[49] IEEE. (2021). *IEEE P802.11-Task Group Be (EHT) Meeting Update*. [Online]. Available: https://www.ieee802.org/11/Reports/tgbe_update.htm

[50] L. You, Q. Liao, N. Pappas, and D. Yuan, "Resource optimization with flexible numerology and frame structure for heterogeneous services," *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2579–2582, Dec. 2018.

[51] L. Marijanović, S. Schwarz, and M. Rupp, "Multiplexing services in 5G and beyond: Optimal resource allocation based on mixed numerology and mini-slots," *IEEE Access*, vol. 8, pp. 209537–209555, 2020.
[52] W. Sui, X. Chen, S. Zhang, Z. Jiang, and S. Xu, "Energy-efficient resource allocation with flexible frame structure for hybrid eMBB and URLLC services," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 1, pp. 72–83, Mar. 2021.
[53] D. Malone, K. Duffy, and D. Leith, "Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions," *IEEE/ACM Trans. Netw.*, vol. 15, no. 1, pp. 159–172, Feb. 2007.
[54] G. Bianchi, "IEEE 802.11-saturation throughput analysis," *IEEE Commun. Lett.*, vol. 2, no. 12, pp. 318–320, Dec. 1998.
[55] T. R. Park, T. H. Kim, J. Y. Choi, S. Choi, and W. H. Kwon, "Throughput and energy consumption analysis of IEEE 802.15.4 slotted CSMA/CA," *Electron. Lett.*, vol. 41, no. 18, pp. 1017–1019, 2005.
[56] B. Jang and M. L. Sichitiu, "IEEE 802.11 saturation throughput analysis in the presence of hidden terminals," *IEEE/ACM Trans. Netw.*, vol. 20, no. 2, pp. 557–570, Apr. 2012.
[57] E. Ziouva and T. Antonakopoulos, "CSMA/CA performance under high traffic conditions: Throughput and delay analysis," *Comput. Commun.*, vol. 25, no. 3, pp. 313–321, 2002.
[58] *IEEE Standard for Information Technology–Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks–Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard Std 802.11-2020, 2021.
[59] S. Weisberg, *Applied Linear Regression*, vol. 528. Hoboken, NJ, USA: Wiley, 2005.
[60] N. R. Draper and H. Smith, *Applied Regression Analysis*, vol. 326. New York, NY, USA: Wiley, 1998.
[61] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Stat. Comput.*, vol. 14, no. 3, pp. 199–222, 2004.
[62] Y. Zhang, Y.-H. Lo, W. S. Wong, and F. Shu, "Protocol sequences for the multiple-packet reception channel without feedback," *IEEE Trans. Commun.*, vol. 64, no. 4, pp. 1687–1698, Apr. 2016.



LEI DENG (Member, IEEE) received the B.Eng. degree from the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2012, and the Ph.D. degree from the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong, in 2017. In 2015, he was a Visiting Scholar with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA. He is currently an Assistant Professor with College of Electronics and Information Engineering, Shenzhen University, China. His research interests are real-time (delay-constrained) networking and graph-theoretic networking problems.

YOUZHI YU received the B.Eng. degree from the Department of Electronic Science and Technology, Hunan Institute of Engineering, China, in 2018, and the master's degree from the College of Electronics and Information Engineering, Shenzhen University, China, in 2021. Her research interest is timely network communications.

YIJIN ZHANG (Senior Member, IEEE) received the B.S. degree in information engineering from the Nanjing University of Posts and Telecommunications, China, in 2004, the M.S. degree in information engineering from Southeast University, China, in 2007, and the Ph.D. degree in information engineering from the Chinese University of Hong Kong in 2010. In 2011, he joined the Nanjing University of Science and Technology, China, where he is currently a Professor with the School of Electronic and Optical Engineering. His research interests include sequence design and resource allocation for communication networks.

YUNGHSIANG S. HAN (Fellow, IEEE) was born in Taipei, Taiwan, in 1962. He received the B.Sc. and M.Sc. degrees in electrical engineering from the National Tsing Hua University, Hsinchu, Taiwan, in 1984 and 1986, respectively, and the Ph.D. degree from the School of Computer and Information Science, Syracuse University, Syracuse, NY, USA, in 1993. He is currently with the Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, China. He is also a Chair Professor with National Taipei University from February 2015. His research interests are in error-control coding, wireless networks, and security. He was a winner of the 1994 Syracuse University Doctoral Prize. One of his papers won the prestigious 2013 ACM CCS Test-of-Time Award in cybersecurity.