Delay and Stability Analysis of Connection-Based Slotted-Aloha

Huanhuan Huang, Tong Ye, *Member, IEEE*, Tony T. Lee, *Life Fellow, IEEE*, and Weiqiang Sun, *Senior Member, IEEE*

***Abstract*— In recent years, connection-based slotted-Aloha (CS-Aloha) has been proposed to improve the performance of random access networks. In this protocol, each node attempts to send a request to the access point (AP) before packet transmission. Once this attempt is successful, the node can transmit up to M packets to the AP. Previous works indicated that the CS-Aloha can achieve a higher throughput than the classical slotted Aloha (S-Aloha), if M is large enough. However, the impact of M on the delay performance and stability is still unknown. To solve this problem, we model each node of the CS-Aloha as a vacation queueing system with limited service discipline, where we consider each batch of packet transmissions as a busy period, and the attempt process between two successive busy periods as a vacation period. We derive the delay distribution, which is turned out to be a geometric distribution. From this result, we further obtain the mean delay, the delay jitter, and the bounded delay region. Our analysis shows that increasing M can accelerate the clean-up of the buffer in each node and thus decrease the attempt rate, which can reduce the average time needed by a node to make a successful attempt. As a result, a large M can decrease the mean delay and the delay jitter, and enlarge the bounded delay region. Also, we obtain the condition to achieve the minimum mean delay under different values of M.**

***Index Terms*— Slotted Aloha, connection-based, vacation model, random access.**

I. Introduction

SLOTTED Aloha (S-Aloha) is a medium access control (MAC) protocol designed for wireless random access networks. The S-Aloha is easy to implement and can provide low access delay when the traffic load is small [1]–[5]. Because of such advantages, the S-Aloha or Aloha-like protocols have been widely employed in different kinds of access networks [6]–[11].

Manuscript received February 24, 2020; revised August 12, 2020; accepted October 1, 2020; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor A. Banchs. Date of publication October 29, 2020; date of current version February 17, 2021. This work was supported in part by the National Science Foundation of China under Grant 61671286 and Grant 61571288, in part by the Shenzhen Science and Technology Innovation Committee under Grant JCYJ20180508162604311, and in part by the National Key Research and Development Program of China under Grant 2018YFB1800803. *(Corresponding author: Tong Ye.)*

Huanhuan Huang, Tong Ye, and Weiqiang Sun are with the State Key Laboratory of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: huanghuanhuan@sjtu.edu.cn; yetong@sjtu.edu.cn; sunwq@sjtu.edu.cn).

Tony T. Lee was with the State Key Laboratory of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, Shanghai 200240, China. He is now with the School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen), Shenzhen 518172, China (e-mail: tonylee@cuhk.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNET.2020.3029774, provided by the authors.

Digital Object Identifier 10.1109/TNET.2020.3029774

However, the throughput of the S-Aloha is low when the traffic load is moderate or high. In the S-Aloha, each backlogged node attempts to transmit a packet at the beginning of each time slot with probability r, called transmission probability in [12]. If only one node makes an attempt in this slot, this node can transmit the packet successfully; otherwise, multiple packets collide with each other and no transmissions are successful. No matter what happens in this slot, all the backlogged nodes will repeat the procedure in the next slot. Owing to frequent collisions, the maximum throughput of the S-Aloha is only e^{-1} [3].

Recently, connection-based slotted-Aloha (CS-Aloha) [13] was proposed to reduce the collision overhead incurred by channel competition. In this protocol, each node sends a short request with probability r to the access point (AP) before packet transmission. Once the request is transmitted successfully, the node can send up to M packets, where M is called *batch size* in this paper. It was shown in [13] that the throughput of the CS-Aloha increases with M, and will be larger than that of the S-Aloha if M exceeds a threshold. Owing to elegant throughput performance, the CS-Aloha has been gradually applied to underwater acoustic communication [14]–[18] and cellular-based Internet of things [19], [20].

Though the throughput performance of the CS-Aloha has been well studied, its delay performance and stability are still unknown. Intuitively, as M increases, a node may monopolize the channel for a long time once it succeeds, which may increase the packet delay of other nodes. This, in turn, increases the number of packets accumulated in the buffer before a node starts a busy period. It seems that increasing M will deteriorate the delay performance, such as mean delay and delay jitter. To understand this problem, it is necessary to study the impact of M on the delay performance of the CS-Aloha.

A. Previous Work

Up to now, most of the previous work focused on the throughput analysis of Aloha-like protocols. Only a few analyzed the delay performance of Aloha-like protocols.

Ref. [3], [4], [21]–[28] studied the throughput of the S-Aloha with a large node population under saturated conditions, where the buffer of each node is always non-empty. In the analysis, the number of attempts in each slot was assumed to be a Poisson random variable with mean value of G, called attempt rate. Ref. [3], [4], [22]–[28] showed that the maximum throughput of the S-Aloha was e^{-1} if the propagation delay between the nodes and the AP and the overhead of acknowledgement (ACK) were ignored.

Ref. [13] investigated the throughput of the CS-Aloha. The result in [13] showed that the throughput of the CS-Aloha increased with the batch size, and was larger than that of the S-Aloha if the batch size exceeded a threshold. However, Ref. [13] did not consider the queueing process of the packets in the buffer, and thus did not obtain the delay performance.

Ref. [25], [29]–[31] analyzed the delay performance of the S-Aloha in unsaturated networks. In particular, these works employed multi-dimensional Markov chains to delineate the dependency among the queues of the nodes. However, these models were too complex to be solved. For example, Ref. [29]–[31] modeled an n-node S-Aloha network as an n-dimensional Markov chain. When $n > 2$, the state space of the Markov chain was too large and the model became unsolvable.

Ref. [32] focused on the delay analysis of the S-Aloha, where all the nodes were statistically identical. Also, Ref. [32] assumed that the node population was large such that the dependency among different queues can be ignored. The model in [32] took advantage of the feature of the S-Aloha, where each head-of-line (HOL) packet experiences the same attempt process before it is successfully transmitted. Hence, Ref. [32] modeled each node as a Geo/G/1 queue, where the service time of each packet was defined as the duration from the slot when it became the HOL packet to the slot when it was successfully transmitted. This model devised a Markov chain to delineate the attempt process. Using this model, Ref. [32] derived the first and the second moments of the service time and the bounded delay region, which was defined as the value range of transmission probability r where the S-Aloha can achieve a bounded delay.

However, the model in [32] cannot be used to analyze the CS-Aloha. Unlike that in the S-Aloha, the node in the CS-Aloha can send multiple packets after each successful attempt, which means different packets have different experiences. If a packet becomes a HOL packet when the node does not take over the channel, it has to experience the attempt process before successful transmission. However, those packets that are waiting behind the HOL packet in the buffer when the node won the competition can be transmitted directly without any attempt. Therefore, the attempt process cannot be regarded as a part of the service of a packet in the analysis of the CS-Aloha.

In summary, there is no queueing model available for the delay performance analysis of the CS-Aloha.

B. Our Contributions

In this paper, we develop a vacation queueing model with limited-service discipline to analyze the CS-Aloha, where each node can transmit up to M packets once it succeeds in channel competition. Our goal is to study the impact of batch size M on the delay performance and the stable condition of the CS-Aloha.

Different from [32], we model each node in the CS-Aloha as a Geo/D/1 queue with vacation and limited service, where each batch of packet transmission is considered as a busy period and the attempt process between two successive busy periods is treated as a vacation period. We show that the vacation period is governed by the packet arrival process and the attempt process. Similar queueing systems with vacations controlled by the arrival process were considered in [33], in which the results demonstrated that the key to tackle the delay analysis is the distribution of the number of arrivals in a vacation period. Moreover, the number of attempts in each free slot is a Poisson random variable, and the successful probability of each attempt initiated by a node depends on the attempt rate. Thus, we first start our analysis with the derivation of the distribution of the number of arrivals during a vacation period, and then calculate the attempt rate. We derive this distribution using renewal equations, which are built up according to the feature of the attempt process, and attempt rate G using a Lindley's equation, which is established based on the limited service discipline of the CS-Aloha. We finally obtain the delay distribution, the mean delay, the delay jitter, and the bounded delay region. Interestingly, we find that, though the CS-Aloha adopts the limited-service discipline, its delay distribution is a geometric distribution when the number of nodes n is sufficiently large, say $n \geq 10$. This is attributed to the fact that the attempt process of each node in the CS-Aloha is essentially a series of Bernoulli trials.

Our analytical results indicate that increasing M can suppress the queue length of each node, accelerate the clean-up of the buffer and thus decrease the attempt rate, which can reduce the average time needed by a node to make a successful attempt. Therefore, the CS-Aloha with $M > 1$ can decrease the mean delay and the delay jitter, and enlarge the bounded delay region.

In summary, the contributions of this paper are as follows:

(1) A vacation queueing model is built up to derive the delay distribution of the CS-Aloha, which turns out to be a geometric distribution when n is sufficiently large.
(2) The stable conditions of the CS-Aloha are derived, from which we obtain the bounded delay region in terms of transmission probability.
(3) It is found that increasing M can remarkably improve the delay performance of the CS-Aloha and enlarge the bounded delay region.
(4) The optimum transmission probability is derived to achieve the minimum mean delay under different values of M.

The rest of this paper is organized as follows. Section II presents the working process of CS-Aloha and discusses its saturated throughput performance. Section III devises a vacation model to delineate the queueing behavior of each node, and analyzes the distribution of the number of arrivals during a vacation period. Section IV derives the attempt rate. Section V studies how the batch size M influences the mean delay and the bounded delay region. Section VI gives the delay distribution and studies how the delay jitter changes with M. Section VII demonstrates how our analytical model can be applied to the practical scenario. Section VIII concludes this paper.

II. Overview of CS-Aloha

The CS-Aloha network is a time slotted system, where n nodes communicate with an AP. As shown in Fig. 1, a time slot is defined as the time interval from the epoch when a node sends a request to the epoch when it receives the ACK from the AP. The node spends $1 - \delta$ portion of the time slot on request transmission and the remaining δ on receiving the

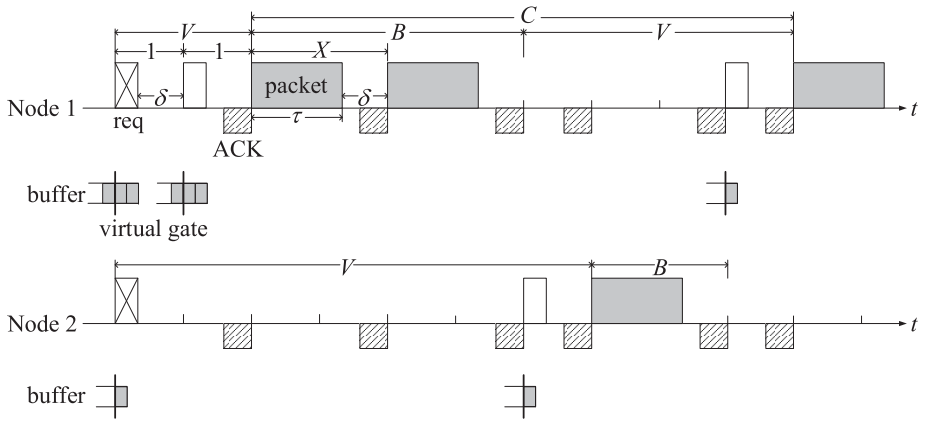


Fig. 1. Illustration of a two-node CS-Aloha with $M = 2$, where B, V, and C denote busy period, vacation period, and cycle time, respectively.

ACK, where $0 < \delta < 1$. The size of a packet is typically larger than that of a request. Let τ denote the transmission time of a packet, where $\tau \geq 1 - \delta$. Since the node can send the next packet only after it receives the ACK from the AP, we consider the interval from the time point when a packet is transmitted to that when it is acknowledged as the service time of the packet. Let X denote the service time of a packet. Clearly $X = \tau + \delta$.

Fig. 1 illustrates the working process of the CS-Aloha network. If the channel is free in a slot, each backlogged node attempts to send a request with transmission probability r. Each node maintains a virtual gate in the buffer. At the beginning of each slot in which the node makes an attempt, the node puts the first K packets inside the gate, where K is equal to the smaller of the queue length and a preset integer M. Also, the node appends this information to the request. If the AP does not receive the request successfully, the node will make an attempt with probability r in the next slot; otherwise, the AP will broadcast an ACK to all the nodes, such that all the other nodes can know that the following KX slots have been reserved. In other words, the successful node will take over the channel and send out K packets in the gate, while other nodes will keep silent in the next KX time slots. Once the AP receives a packet, it will feed back an ACK to this node to acknowledge the successful transmission of this packet. An example is plotted in Fig. 1, where $M = 2$.

To analyze the CS-Aloha, we adopt the following assumptions throughout this paper:

A1. The channel is error free.
A2. The number of nodes, denoted by n, is sufficiently large.
A3. All the nodes are statistically identical, which was widely adopted in the analysis of Aloha-like protocols [22], [32], [34]–[39].
A4. Each node has a packet arrival at the beginning of a slot with probability λ, and thus the aggregate arrival rate of all the nodes is $\hat{\lambda} = n\lambda$ packets/slot.
A5. The service time of each packet X is an integer.
A6. Each node has an infinite buffer and transmits the packets in a first-in-first-out manner.

According to A2, in a free slot that is not reserved by a node, the number of attempts is approximately a Poisson random variable [22], [32], [39], the mean of which is called the attempt rate and denoted by G.

The network throughput is defined as the fraction of the time that the channel spends on packet transmissions. Ref. [13] studied the saturated throughput of the CS-Aloha. When the network is saturated, each node always has packets to send, and thus $G = nr$. It follows that the probability that only one node makes an attempt in a free slot is $Ge^{-G} = nre^{-nr}$. After a node succeeds, the channel will transmit packets for this node in the next MX slots. Hence, the average number of time slots between two successive periods when the channel is busy transmitting is $1/(nre^{-nr})$. Clearly, these slots can be regarded as the collision overhead for packet transmissions. According to [13], the fraction of the slots during which the channel is busy is given by

$$\rho_{sat} = \frac{MX}{1/(nre^{-nr}) + MX}. \tag{1}$$

Remember that only time $M\tau$ in the MX slots is spent on packet transmissions. We thus have the saturated throughput as follows:

$$\eta_{sat} = \rho_{sat} \cdot \frac{M\tau}{MX} = \frac{\tau}{\frac{1}{Mnre^{-nr}} + X}, \tag{2}$$

where $1/(Mnre^{-nr})$ is the amortized collision overhead to transmit a packet. When $M = 1$, this overhead is significant, and the channel needs $1/(nre^{-nr})$ time slots for each packet transmission. When $M > 1$, the overhead is shared by multiple packets and thus the throughput is improved. Particularly when $M \to \infty$, the collision overhead can be ignored and the throughput approaches τ/X.

As [13] pointed out, if M is large enough, the CS-Aloha can achieve a higher throughput than the S-Aloha. Different from that in the CS-Aloha, each node in the S-Aloha competes for the channel by directly sending the HOL packet, instead of a request. If the node in the S-Aloha transmits the HOL packet successfully, it releases the channel and repeats channel competition in the next slot. That is, each HOL packet has to suffer an attempt process before it is successfully transmitted. Similarly, the probability that only one node makes an attempt in a slot is nre^{-nr}. It follows that the throughput of the S-Aloha is

$$\hat{\eta}_{sat} = nre^{-nr}\frac{\tau}{X}.$$

When batch size M is small, the throughput of the CS-Aloha may be less than that of the S-Aloha because the

CS-Aloha needs to send a request before packet transmission. For example, when $M = 1$ and $X = 1$, there is

$$\eta_{sat} = \frac{\tau}{\frac{1}{nre^{-nr}} + 1} < \hat{\eta}_{sat} = \frac{\tau}{\frac{1}{nre^{-nr}}}. \tag{3}$$

When M increases such that the volume of the data that the node can send each time is large enough to offset the overhead incurred by the request, the CS-Aloha achieves a higher throughput than the S-Aloha [13]. Comparing η_{sat} and $\hat{\eta}_{sat}$, we find that the CS-Aloha outperforms the S-Aloha if

$$M \geq \left\lceil \frac{1}{X(1 - nre^{-nr})} \right\rceil. \tag{4}$$

For example, $M \geq 2$ is large enough for the CS-Aloha to achieve a higher throughput than the S-Aloha if $X = 1$ and $nr = 1$, and $M \geq 1$ is enough if $X = 2$ and $nr = 1$.

In practice, networks usually operate under the unsaturated condition, where the buffer of the node may sometimes be empty. To provide guaranteed quality of experience (QoE), the access network should operate in a stable state, where the mean delay is bounded. Note that the aggregate packet arrival rate of the network is $\hat{\lambda}$ and the capacity that the CS-Aloha can provide is ρ_{sat}/X. To keep the network stable, it is necessary to ensure

$$\hat{\lambda} < \frac{\rho_{sat}}{X} = \frac{M}{1/(nre^{-nr}) + MX}. \tag{5}$$

However, it is unknown that if Eq. (5) is also a sufficient condition for the stability of the CS-Aloha. Furthermore, the user expects the mean delay and the delay jitter to be as small as possible, which requires a careful selection of parameters, such as r and M. Hence, it is necessary to study the delay performance of the CS-Aloha.

III. Vacation Queueing Model for an Access Node

In the CS-Aloha, the working process of each node can be divided into cycles, each of which is defined as the duration between two consecutive time points that the node begins to send a batch of packets. As shown in Fig. 1, a cycle, denoted by C, consists of a busy period, denoted by B, followed by a vacation period, denoted by V. When the node successfully sends a request, it immediately starts a busy period, during which all the packets inside the virtual gate are sent out. After the busy period, the node releases the channel and a vacation period begins. This node will compete for the channel immediately after it releases the channel if its buffer is not empty; otherwise, it will do that only after a new packet arrives. The vacation period ends at the slot when the node succeeds in the channel competition again. This implies that the vacation period is influenced by packet arrival process.

Therefore, according to assumptions A4 and A5, each node in the CS-Aloha can be regarded as a Geo/D/1 queue with vacation period governed by the packet arrival process. The key to delay analysis for this kind of queueing system is to derive the distribution of the number of arrivals during a vacation period, similar to [33]. In the CS-Aloha, a vacation period of a node may include multiple busy periods of other nodes. As illustrated in Fig. 1, the first vacation period of node 2 includes one busy period of node 1. This implies that the distribution of the number of arrivals during the vacation period depends on the distribution of that during the busy period. In the following, we derive these two distributions.

A. Distribution of the Number of Arrivals During the Busy Period

Let L be the number of arrivals during a busy period. Given the busy period B, L is a binomial random variable according to assumption A4. We have $B = KX$ slots since K packets are served in a busy period. Define $k_j = Pr\{K = j\}$ as the probability that j packets are served in a busy period, where $j = 1, 2, \cdots, M$. The generating function of k_j is $K(z) \triangleq E[z^K] = \sum_{j=1}^{M} k_j z^j$. It follows that the probability generating function of the number of arrivals during the busy period is given by

$$\begin{aligned} L(z) &= E[z^L] \\ &= \sum_{j=1}^{M} E[z^L | K = j] k_j \\ &= \sum_{j=1}^{M} k_j \left[\sum_{i=0}^{jX} \binom{jX}{i} \lambda^i (1-\lambda)^{jX-i} z^i \right] \\ &= \sum_{j=1}^{M} k_j (1 - \lambda + \lambda z)^{jX} \\ &= K[(1 - \lambda + \lambda z)^X] \\ &= 1 + \lambda K'(1) X (z-1) \\ &\quad + \frac{\lambda^2}{2} [K''(1)X^2 + K'(1)X(X-1)](z-1)^2 + \cdots. \end{aligned} \tag{6}$$

With the increase of node population n, for a given $\hat{\lambda}$, the packet arrival rate of each node $\lambda = \frac{\hat{\lambda}}{n}$ decreases in the order of $o(\frac{1}{n})$. When n is sufficiently large, $L(z)$ can be expressed as

$$\begin{aligned} L(z) &= 1 - \lambda K'(1) X + \lambda K'(1) X z + o\left(\frac{1}{n}\right) \\ &= 1 - \lambda \overline{B} + \lambda \overline{B} z + o\left(\frac{1}{n}\right), \end{aligned} \tag{7}$$

where $\overline{B} = K'(1)X$ is the mean busy period. Equation (7) clearly indicates that the distribution of L approaches a Bernoulli distribution with the mean $\lambda\overline{B}$. In other words, when n is sufficiently large, λ is such small that the probability that more than one packet arrives at the node during the busy period is negligible.

$\overline{B}$ in (7) can be determined as follows. In an unsaturated network, the successful node will start a busy period of length B slots, where $B \leq MX$. Following the argument similar to (1), the fraction of the time slots during which the channel is busy serving packets, denoted by ρ, is given by:

$$\rho = \frac{\overline{B}}{1/(Ge^{-G}) + \overline{B}}. \tag{8}$$

Also, in a stable network, ρ is equal to the aggregate traffic load $\hat{\lambda}X$, which implies the mean busy period satisfies

$$\overline{B} = \frac{\hat{\lambda}X}{1 - \hat{\lambda}X} \cdot \frac{1}{Ge^{-G}}. \tag{9}$$

$U_1(z)$ {
- Event 1: Node 1 succeeds with probability $p_s=re^{-G}$; $E[z^{U_1}|\text{event 1}]=E[z^I]=I(z)$
- Event 2: One of the other nodes succeeds with probability $p_w=(1-r)Ge^{-G}$; $E[z^{U_1}|\text{event 2}]=E[z^{I+L+U_1}]=I(z)L(z)U_1(z)$
- Event 3: No nodes succeed with probability $p_c=1-re^{-G}-(1-r)e^{-G}$; $E[z^{U_1}|\text{event 3}]=E[z^{I+U_1}]=I(z)U_1(z)$

Fig. 2. Renewal process of the number of arrivals U_1 during vacation period Y_1.

As Section II states, Ge^{-G} is the probability that there is a successful attempt in a free slot. Intuitively, if Ge^{-G} is large, the nodes in the network will have more chances to send packets, and thus the number of packets backlogged in their buffers will be small, which will shorten the mean busy period. This point is clearly demonstrated by Eq. (9), in which $\overline{B}$ is inversely proportional to Ge^{-G}.

B. Distribution of the Number of Arrivals During the Vacation Period

When the busy period finishes, the node starts a vacation period. The buffer at this epoch may be empty or not. If the buffer is not empty, the node competes for the channel immediately. In this paper, we refer to the vacation period as a type-1 vacation, if the buffer is not empty at the beginning of this vacation; otherwise, it is a type-0 vacation.

Let Y_1 be the type-1 vacation and U_1 be the number of arrivals during Y_1 in this case. As shown in Fig. 2, there are three events that may occur in the first slot of the vacation period. Define the generating function of U_1 as $U_1(z) \triangleq E[z^{U_1}]$, which can be derived as follows.

E1: If node 1 sends a request with probability r while the other $n-1$ nodes do not make an attempt with probability e^{-G}, it will succeed and begin another busy period in the next slot. Thus, this event happens with probability $p_s = re^{-G}$. In this case, the vacation period is one slot. Let I be the number of arrivals in a slot. According to assumption A4, I is a Bernoulli random variable with mean λ, the generating function of which is $I(z) = 1 - \lambda + \lambda z$. Thus, if event 1 happens, the generating function of the number of arrivals during the vacation period is $E[z^{U_1}|\text{event 1}] = I(z)$.

E2: If node 1 does not make an attempt with probability $1-r$ while only one of the other $n-1$ nodes sends a request with probability Ge^{-G}, the node that sends the request will succeed and start a busy period at the beginning of the next slot. Thus, this event happens with probability $p_w = (1-r)Ge^{-G}$. In this case, node 1 will restart the channel competition after the busy period. The number of arrivals for node 1 during this slot and the following busy period is $I + L$, the generating function of which is $I(z)L(z)$. Since the competition process is memoryless, the generating function of the number of arrivals at node 1 during the remaining part of vacation period is $U_1(z)$. Thus, we have $E[z^{U_1}|\text{event 2}] = I(z)L(z)U_1(z)$.

E3: If no nodes or multiple nodes send requests, which occurs with probability $p_c = 1 - p_s - p_w$, no one will succeed in this slot and node 1 will compete for the channel in the next slot. The generating function of the number of arrivals during the first slot is $I(z)$ and that during the remaining vacation period is $U_1(z)$. We thus have $E[z^{U_1}|\text{event 3}] = I(z)U_1(z)$.

$U_0(z)$ {
- Event 1: One of the other nodes succeeds with probability Ge^{-G} and no packets arrive at node 1 during the $1+B$ slots with probability $(1-\lambda)l_0$; $E[z^{U_0}|\text{event 1}]=E[z^{U_0}]=U_0(z)$
- Event 2: One of the other nodes succeeds with probability Ge^{-G} and at least one packet arrives at node 1 during the $1+B$ slots with probability $1-(1-\lambda)l_0$; $E[z^{U_0}|\text{event 2}] = \sum_{j=1}^{\infty} \frac{(1-\lambda)l_j + \lambda l_{j-1}}{1-(1-\lambda)l_0} E[z^{j+U_1}] = \frac{[I(z)L(z)-(1-\lambda)l_0]U_1(z)}{1-(1-\lambda)l_0}$
- Event 3: No nodes succeed and no packets arrive at node 1 during the slot with probability $(1-\lambda)(1-Ge^{-G})$; $E[z^{U_0}|\text{event 3}]=E[z^{U_0}]=U_0(z)$
- Event 4: No nodes succeed and a packet arrives at node 1 during the slot with probability $\lambda(1-Ge^{-G})$; $E[z^{U_0}|\text{event 4}]=E[z^{1+U_1}]=zU_1(z)$

Fig. 3. Renewal process of the number of arrivals U_0 during vacation period Y_0.

The above analysis clearly indicates that the channel access procedure is a renewal process [40]. Conditioning on the event that occurs in the first slot, the probability generating function of U_1 satisfies the following equation:

$$U_1(z)=E[z^{U_1}]=p_sI(z)+p_wI(z)L(z)U_1(z)+p_cI(z)U_1(z),$$

which yields,

$$U_1(z) = \frac{p_sI(z)}{1-p_wI(z)L(z)-p_cI(z)} = \frac{p_s(1-\lambda+\lambda z)}{1-p_w(1-\lambda+\lambda z)L(z)-p_c(1-\lambda+\lambda z)}. \quad (10)$$

On the other hand, if the vacation is a type-0 vacation, the node competes for the channel after a new packet arrives. Let Y_0 be the type-0 vacation and U_0 be the number of arrivals during Y_0. As shown in Fig. 3, four events may happen in the first slot of the vacation period. In a way similar to that of $U_1(z)$, $U_0(z) \triangleq E[z^{U_0}]$ can be derived as follows:

$$U_0(z)=\frac{\lambda(1-Ge^{-G})z+Ge^{-G}[I(z)L(z)-(1-\lambda)l_0]}{1-(1-\lambda)(1-Ge^{-G})-Ge^{-G}(1-\lambda)l_0}U_1(z), \quad (11)$$

where

$$l_0 \triangleq Pr\{L=0\} = L(0) = K[(1-\lambda)^X]$$

is the probability that no packets arrive in a busy period.

The attempt process of a node is actually a series of Bernoulli trials, which terminate when the node makes a successful trial. Thus, the number of attempts needed by a node for success is geometrically distributed. If the buffer of the node is non-empty, it will make an attempt immediately when the vacation starts. As shown in Fig. 2, if the attempt is not successful, two events may occur: 1) no nodes succeed, and the duration time of this event is one slot and the number

of arrivals in a slot is a Bernoulli variable with mean λ; 2) one of other nodes succeeds, and the duration time of this event is $1+B$ slots and the number of arrivals during that period is $I+L$, which approaches a Bernoulli variable with mean $\lambda(1+\overline{B})$ if the number of nodes n is sufficiently large. In other words, the number of arrivals U_1 during the vacation period Y_1 is the sum of two kinds of Bernoulli random variables if n is sufficiently large, and the number of the Bernoulli random variables is a geometric random variable. Similarly, the same thing is true for the number of arrivals U_0 during the vacation period Y_0. This hints that, when n is sufficiently large, U_1 and U_0 may approach a geometric distribution.

Let U be the number of arrivals during the vacation period and $U(z)$ be the probability generating function of U. We have the following result.

Lemma 1: When the number of nodes n is sufficiently large, $U(z)$ can be expressed as

$$U(z)=\begin{cases} U_1(z), & \textit{if the buffer is not empty at the} \\ & \textit{start of a vacation} \\ U_0(z), & \textit{otherwise} \end{cases}$$

where

$$U_1(z) = \frac{\beta}{1-(1-\beta)z}(1-\lambda+\lambda z)+o\left(\frac{1}{n}\right), \quad (12)$$

$$U_0(z) = zU_1(z)+o\left(\frac{1}{n}\right) = \frac{\beta z}{1-(1-\beta)z}(1+\lambda+\lambda z)+o\left(\frac{1}{n}\right), \quad (13)$$

and

$$\beta = \frac{p_s}{p_s+\lambda p_c+\lambda(1+\overline{B})p_w}. \quad (14)$$

Proof: It is clear that $U=U_1$ if the buffer is not empty at the start of the vacation period; otherwise, $U=U_0$. When n becomes large, the distribution of the number of arrivals during a busy period $L(z)$ approaches a Bernoulli distribution, as (7) shows. Substituting (7) into (10), we obtain

$$U_1(z) = \frac{p_s(1-\lambda+\lambda z)}{1-p_c(1-\lambda+\lambda z)-p_w(1-\lambda+\lambda z)(1-\lambda\overline{B}+\lambda\overline{B}z)}+o\left(\frac{1}{n}\right) = \frac{\frac{p_s}{p_s+\lambda p_c+\lambda(1+\overline{B})p_w}}{1-\left(1-\frac{p_s}{p_s+\lambda p_c+\lambda(1+\overline{B})p_w}\right)z}(1-\lambda+\lambda z)+o\left(\frac{1}{n}\right).$$

Similarly, substituting (7) and

$$l_0 = K[(1-\lambda)^X] = 1-\lambda K'(1)X+o\left(\frac{1}{n}\right) = 1-\lambda\overline{B}+o\left(\frac{1}{n}\right)$$

into (11) produces $U_0(z)$ in (13). ∎

Recall that the node competes for the channel immediately when type-1 vacation Y_1 starts, meaning that Y_1 is completely governed by the channel competition process. Thus, the probability generating functions of Y_1 and U_1 have the following relationship:

$$U_1(z) = Y_1(1-\lambda+\lambda z) \quad (15)$$

It follows from Lemma 1 that

$$Y_1(z)=U_1\left(\frac{z-1+\lambda}{\lambda}\right)=\frac{\frac{\lambda\beta}{1-\beta+\lambda\beta}z}{1-\left(1-\frac{\lambda\beta}{1-\beta+\lambda\beta}\right)z}+o\left(\frac{1}{n}\right), \quad (16)$$

which implies that Y_1 is a geometric random variable. From (16), we immediately obtain

$$\overline{Y_1} = Y_1'(1) = \frac{1}{p_s}(1+\overline{B}p_w) = \frac{1}{re^{-G}}\cdot\frac{1-\hat{\lambda}Xr}{1-\hat{\lambda}X}, \quad (17)$$

where $\frac{1}{p_s}=\frac{1}{re^{-G}}$ is the average number of attempts made by a node before it succeeds, and $1+\overline{B}p_w=\frac{1-\hat{\lambda}Xr}{1-\hat{\lambda}X}$ is the average time that a node has to wait before it can reattempt after a failed attempt. Clearly, $1/p_s$ is proportional to attempt rate G, while $1+\overline{B}p_w$ does not change with G. This indicates the mean vacation period $\overline{Y_1}$ reduces with the decrease of G.

In type-0 vacation Y_0, the node competes the channel only after a new packet arrives, which means Y_0 also depends on the packet arrival process. It follows that $U_0(z)\neq Y_0(1-\lambda+\lambda z)$ [33]. Though the generating function of Y_0 cannot be derived from Lemma 1, we can obtain $\overline{Y_0}$ as follows:

$$\overline{Y_0} = \frac{U_0'(1)}{\lambda} = \frac{1}{\lambda}+\overline{Y_1} = \frac{1}{\lambda}+\frac{1}{re^{-G}}\cdot\frac{1-\hat{\lambda}Xr}{1-\hat{\lambda}X}. \quad (18)$$

The derivation of this section clearly shows that attempt rate G is the key parameter to determine $\overline{B}$, $\overline{U_1}$, $\overline{U_0}$, $\overline{Y_1}$, and $\overline{Y_0}$. Thus, we analyze G in the next section.

IV. Attempt Rate

In a free slot, each backlogged node sends a request to the AP with probability r. Thus, the attempt rate is given by

$$G = np_{ne}r, \quad (19)$$

where p_{ne} is the probability that the buffer of a node is non-empty.

The buffer of a node may be emptied at the end of a busy period of this node. In this case, the node becomes backlogged when a new packet arrives. The average inter-arrival time of the packets is $1/\lambda$ slots. Let p_0 be the probability that the buffer of a node is empty at the end of a busy period. Since the average time that the buffer of a node remains empty in a cycle is p_0/λ, the probability that the buffer of a node remains empty is given by

$$1-p_{ne} = \frac{p_0/\lambda}{\overline{C}}. \quad (20)$$

In a stable network, the mean number of arrivals during a cycle is equal to the mean number of packets served in a busy period, which means

$$\lambda\overline{C} = \overline{K} = \frac{\overline{B}}{X}. \quad (21)$$

Combining (19)-(21), we obtain

$$G = nr\left(1-\frac{p_0}{\overline{K}}\right) = nr\left(1-\frac{p_0X}{\overline{B}}\right). \quad (22)$$

Thus, we need to solve p_0 to complete the derivation of attempt rate G.

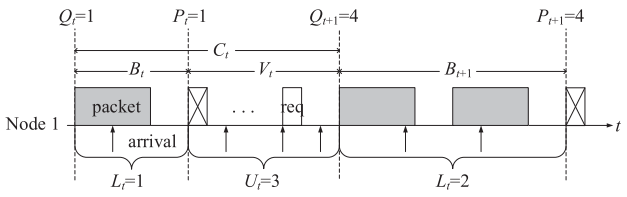


Fig. 4. Packet transmission process of node 1 where $M = 2$.

Let Q_t and P_t be the queue lengths at the start and the end of the t-th busy period of a node, respectively. Define $q_j \triangleq \lim_{t\to\infty} Pr\{Q_t = j\}$ and $p_j \triangleq \lim_{t\to\infty} Pr\{P_t = j\}$, and let $Q(z)$ and $P(z)$ be the probability generating functions of q_j and p_j, where $j = 0, 1, \cdots$.

On one hand, p_j depends on q_j. As shown in Fig. 4, if Q_t is less than M, all the Q_t packets will be transmitted in the busy period of cycle t; otherwise, only the first M packets will be sent. Thus, the number of packets served in the t-th busy period, denoted by K_t, is determined by

$$K_t = Q_t - (Q_t - M)^+, \tag{23}$$

where $x^+ \triangleq \max\{x, 0\}$. Let L_t denote the number of packets that arrive during B_t. The queue length at the end of the t-th busy period P_t is given by

$$P_t = Q_t - K_t + L_t = (Q_t - M)^+ + L_t. \tag{24}$$

In the steady state, we have

$$P = \lim_{t\to\infty} P_t = (Q - M)^+ + L. \tag{25}$$

Therefore, the probability generating function of P is

$$\begin{aligned} P(z) &\triangleq E[z^P] \\ &= E[z^{(Q-M)^+ + L}] \\ &= \sum_{j=1}^{\infty} E[z^{(Q-M)^+ + L} | Q = j, K = j - (j-M)^+] q_j \\ &= \sum_{j=1}^{M-1} E[z^L | K = j] q_j + \sum_{j=M}^{\infty} z^{j-M} E[z^L | K = M] q_j \\ &= \sum_{j=1}^{M-1} E[z^L | B = jX] q_j + \sum_{j=M}^{\infty} z^{j-M} E[z^L | B = MX] q_j \\ &= \sum_{j=1}^{M-1} (1-\lambda+\lambda z)^{jX} q_j + \sum_{j=M}^{\infty} z^{j-M} (1-\lambda+\lambda z)^{MX} q_j. \end{aligned} \tag{26}$$

Hence, the probability that the buffer of a node is empty at the end of a busy period is

$$p_0 = P(0) = \sum_{j=1}^{M} (1-\lambda)^{jX} q_j. \tag{27}$$

On the other hand, q_j is also determined by p_j. As depicted in Fig. 4, the queue length at the start of the $(t+1)$-th busy period Q_{t+1} is composed of the packets waiting in the buffer at the end of the t-th busy period P_t and the packets that arrive during the t-th vacation period, denoted by U_t, i.e.,

$$Q_{t+1} = P_t + U_t. \tag{28}$$

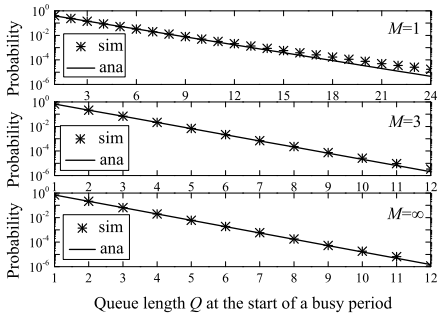


Fig. 5. Distribution q_j under different Ms, where $n = 20$, $\hat{\lambda} = 0.2$ packets/slot, $X = 2$ slots, and $r = 0.05$.

In the steady state, there is

$$Q = \lim_{t\to\infty} Q_{t+1} = P + U. \tag{29}$$

Accordingly, its probability generating function is given by

$$\begin{aligned} Q(z) &\triangleq E[z^Q] \\ &= E[z^{P+U}] \\ &= E[z^{P+U_0} | P = 0] p_0 + \sum_{j=1}^{\infty} E[z^{P+U_1} | P = j] p_j \\ &= p_0 E[z^{U_0}] + \sum_{j=1}^{\infty} E[z^{U_1}] E[z^P | P = j] p_j \\ &= p_0 U_0(z) + [P(z) - p_0] U_1(z). \end{aligned} \tag{30}$$

Equations (26) and (30) show that $P(z)$ and $Q(z)$ couple with each other and thus can only be solved numerically in general.

However, we demonstrate that $Q(z)$ approaches a geometric distribution in the case where the number of nodes n is sufficiently large. Since the vacation period of a node is typically much longer than the busy period, and thus the number of arrivals during the vacation period U constitutes the main body of Q. Lemma 1 shows that U is a geometric random variable when n is large. This hints that $Q(z)$ may also approach a geometric distribution.

Lemma 2: When n is sufficiently large, $Q(z)$ can be expressed as

$$Q(z) = \frac{\alpha z}{1 - (1-\alpha) z} + o\left(\frac{1}{n}\right), \tag{31}$$

where α is given by

$$\alpha = 1 - \frac{G}{nr}. \tag{32}$$

Proof: See APPENDIX A. ■

Fig. 5 verifies the analytical result in (31) via simulation when the number of nodes is $n = 20$ and the aggregate packet arrival rate is $\hat{\lambda} = 0.2$ packets/slot, the service time of a packet is $X = 2$ slots, and the transmission probability is $r = 0.05$. Fig. 5 clearly shows that $n = 20$ is already large enough to ensure the accuracy of the result in (31).

According to Lemma 2, we can easily obtain q_j when n is sufficiently large as follows:

$$q_j = \frac{1}{j!} \frac{d^j Q(z)}{dz^j}\Big|_{z=0} = \alpha(1-\alpha)^{j-1}, \quad j = 1, 2, \cdots \tag{33}$$

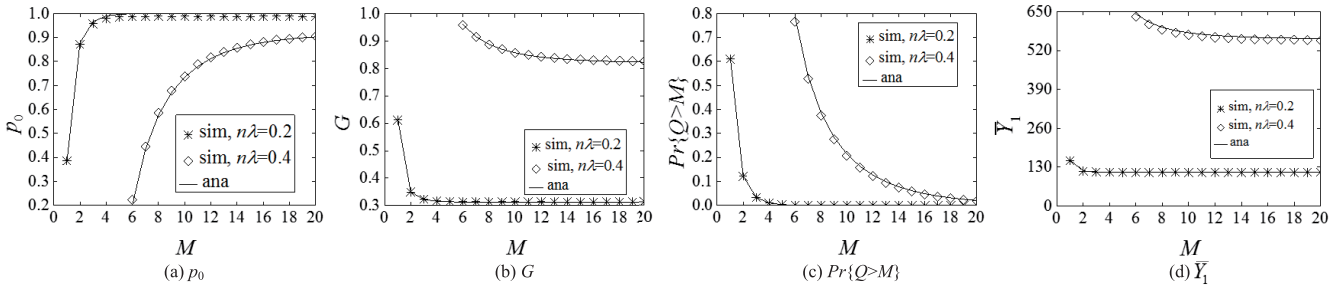


Fig. 6. p_0, G, $Pr\{Q > M\}$ and $\overline{Y_1}$ vary with M, where $n = 50$, $\hat{\lambda} = 0.2, 0.4$ packets/slot, $X = 2$ slots, and $r = 0.02$.

Substituting (33) into (27), we can obtain the probability that the buffer of a node is empty at the end of a busy period as follows

$$p_0 = \sum_{j=1}^{M}(1-\lambda)^{jX}\alpha(1-\alpha)^{j-1} = \frac{(1-\lambda)^X\alpha[1-(1-\lambda)^{XM}(1-\alpha)^M]}{1-(1-\lambda)^X(1-\alpha)}. \quad (34)$$

When n is sufficiently large, $(1-\lambda)^X = (1-\hat{\lambda}/n)^X = 1-\hat{\lambda}X/n+o\left(\frac{1}{n}\right)$, which approaches 1 since $\frac{\hat{\lambda}X}{n} < \frac{1}{n} \ll 1$. In this case, we have

$$p_0 \approx 1-(1-\alpha)^M = 1-\left(\frac{G}{nr}\right)^M. \quad (35)$$

With $\overline{B}$ in (9) and p_0 in (35), the attempt rate G can be derived as follows.

Theorem 1: Attempt rate G of the CS-Aloha with a sufficiently large n is the solution of the following equation:

$$\frac{1-\left(\frac{G}{nr}\right)^M}{1-\frac{G}{nr}} = \frac{\hat{\lambda}}{1-\hat{\lambda}X}\cdot\frac{1}{Ge^{-G}}. \quad (36)$$

Proof: Substituting (9) and (35) into (22), we can get Eq. (36) after some reconfigurations. ∎

Equation (36) shows that batch size M has a significant impact on attempt rate G. Intuitively, if M is large, the buffer of a node will be emptied in one busy period with high probability. This intuition is confirmed by Fig. 6, where G is numerically solved from (36) using the method presented in APPENDIX B. Fig. 6(a) clearly exhibits that the probability p_0 that a node has an empty buffer at the end of a busy period increases from 0.4 to 1 when M increases from 1 to 5 and $\hat{\lambda} = 0.2$ packets/slot. As a result, the number of backlogged nodes declines, and thus attempt rate G decreases, as depicted in Fig. 6(b).

Accordingly, increasing M can suppress the queue length Q at the start of the busy period. From (33), we can write down the probability that Q exceeds M as follows:

$$Pr\{Q > M\} = \sum_{j=M+1}^{\infty}\alpha(1-\alpha)^{j-1} = (1-\alpha)^M. \quad (37)$$

Eq. (32) indicates that α increases with M. It follows that $Pr\{Q > M\}$ decreases with M. This point is visualized by Fig. 6(c), where $Pr\{Q > M\}$ rapidly falls from 0.62 to 0 when M increases from 1 to 5 and $\hat{\lambda} = 0.2$ packets/slot. In other words, when $M > 5$, almost all of the packets waiting in the buffer at the beginning of a busy period can be cleaned up in this busy period with probability 1.

As a result, increasing M can shorten the attempt process of the node. As Section III mentions, on average, a node with a non-empty buffer needs $\overline{Y_1}$ slots to succeed in the channel competition. Also, $\overline{Y_1}$ reduces with the decrease of G. Thus, increasing M can lower $\overline{Y_1}$, as illustrated in Fig. 6(d).

V. Mean Delay and Bounded Delay Region

Our results in Section IV show that increasing M can accelerate the clean-up of the buffer, decrease the attempt rate, and thus reduce the average time needed by a node to recapture the channel. This hints that the CS-Aloha with larger M may achieve a better delay performance. In this section, we first derive the mean delay, denoted by $\overline{D}$, and the bounded delay region, and then study the effect of M on them.

Consider a packet, denoted by A, arrives at a node, say node 1 in Fig. 7, when there are N packets waiting in the buffer. Packet A may wait a residual time for the completion of a service or a vacation depending on if it arrives during a busy period or a vacation period. Also, packet A has to wait for the service completion of the N packets that stay ahead of it in the queue. Furthermore, because of the limitation of batch size, it may take node 1 several busy periods to send out all of the N packets. In this case, packet A has to experience multiple vacation periods in addition. In summary, the waiting time of packet A in queue consists of the following three components:

(a) The residual time, denoted by R, experienced by packet A;
(b) The total service time of N packets that are waiting in the queue upon the arrival of packet A;
(c) The total complete vacation periods experienced by packet A before transmission, which is denoted by H.

Thus, the mean waiting time of packets, denoted by $\overline{W}$, can be expressed as

$$\overline{W} = \overline{R} + \overline{N}X + \overline{H}, \quad (38)$$

where $\overline{N} = \lambda\overline{W}$ according to Little's law. In the following, we derive $\overline{R}$ and $\overline{H}$ to complete the derivation.

Since a node is busy with probability λX and is in vacation with probability $1-\lambda X$, the mean residual time is given by

$$\overline{R} = \lambda X\overline{R_s} + (1-\lambda X)\overline{R_v}, \quad (39)$$

where $\overline{R_s}$ is the mean residual service time and $\overline{R_v}$ is the mean residual vacation time. According to the graphic method described in [41], the mean residual service time $\overline{R_s}$ is given by

$$\overline{R_s} = \frac{X}{2}. \quad (40)$$

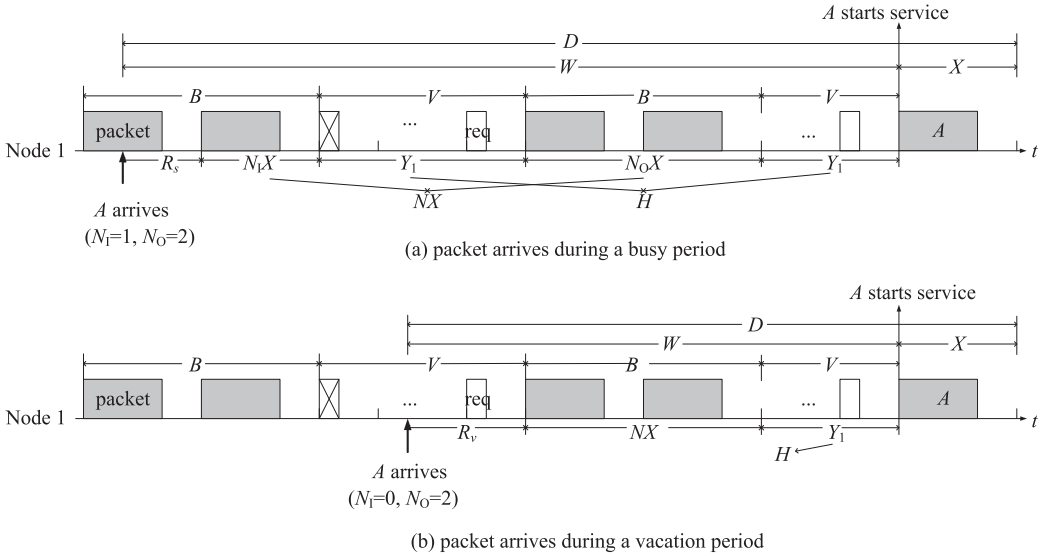


Fig. 7. Delay of packet A where $M = 2$.

There are two types of vacation periods, type-1 vacation (Y_1) and type-0 vacation (Y_0). Recall that Y_1 is a geometric random variable. If a packet arrives during Y_1, the residual vacation period of the packet is thus geometrically distributed. If a packet arrives during Y_0, the node will be eligible to compete the channel in the next available slot. Clearly, the channel competition process during the residual vacation period of the packet would be the same as that of Y_1. It follows that the residual vacation period of the packet has the same distribution as Y_1. Thus, the mean residual vacation period of a packet is

$$\overline{R_v} = \overline{Y_1}. \tag{41}$$

Combining (39)-(41), we obtain the mean residual time under the condition that n is sufficiently large as follows:

$$\overline{R} = \frac{\lambda X^2}{2} + (1 - \lambda X)\overline{Y_1}, \tag{42}$$

which clearly shows that $\overline{R}$ is proportional to $\overline{Y_1}$. Since $\overline{Y_1}$ decreases with M, increasing M can reduce $\overline{R}$.

After the packet experiences the residual time, it may wait several complete vacation periods. Let N_I and N_O be the number of packets waiting inside and outside the gate upon the arrival of a new packet, respectively. As shown in Fig. 7(a), if the new packet arrives in a busy period, it will experience $1 + \left\lfloor \frac{N_O}{M} \right\rfloor$ complete vacation periods before it can be served; otherwise, it will undergo $\left\lfloor \frac{N_O}{M} \right\rfloor$ complete vacation periods. Using the technique described in [42], APPENDIX D obtains the average number of complete vacation periods that a packet experiences as follows:

$$\overline{F} = \lambda X + \frac{\lambda \overline{W}}{M} - \frac{(1 + \lambda X)K''(1)}{2MK'(1)}, \tag{43}$$

where $K'(1)$ is the mean number of packets served in a busy period and $K''(1) = \overline{K^2} - \overline{K}$. $K'(1)$ and $K''(1)$ can be derived via the relationship between K and Q in (23) as follows:

$$K'(1) = \sum_{j=1}^{M-1} j q_j + \sum_{j=M}^{\infty} M q_j = \frac{1 - \left(\frac{G}{nr}\right)^M}{1 - \frac{G}{nr}}, \tag{44}$$

and

$$K''(1) = \sum_{j=1}^{M-1} j(j-1)q_j + \sum_{j=M}^{\infty} M(M-1)q_j = \frac{\frac{2G}{nr}\left[1 - M\left(1 - \frac{G}{nr}\right)\left(\frac{G}{nr}\right)^{M-1} - \left(\frac{G}{nr}\right)^M\right]}{\left(1 - \frac{G}{nr}\right)^2}. \tag{45}$$

Note that each vacation completely experienced by a packet must start with a non-empty buffer. It follows that the average duration of complete vacation periods experienced by a packet is given by

$$\overline{H} = \overline{FY_1} = \left[\lambda X + \frac{\lambda \overline{W}}{M} - \frac{(1 + \lambda X)K''(1)}{2MK'(1)}\right]\overline{Y_1}. \tag{46}$$

Intuitively, if M is large, more packets can be sent in a busy period and the number of complete vacations $\overline{F}$ experienced by a packet before transmission decreases. Also, the mean vacation period $\overline{Y_1}$ reduces with the growth of M. Thus, the average duration of complete vacations experienced by a packet $\overline{H}$ can be reduced by increasing M.

From (38), (42) and (46), it is clear that increasing M can reduce the mean delay, the expression of which is formally stated as follows.

Theorem 2: The mean delay of the CS-Aloha network with sufficiently large node population n is given by

$$\overline{D} = \overline{W} + X = \frac{\frac{\lambda X^2}{2} + \overline{Y_1}\left[1 - \frac{(1+\lambda X)K''(1)}{2MK'(1)}\right]}{1 - \lambda X - \frac{\lambda \overline{Y_1}}{M}} + X, \tag{47}$$

where $\overline{Y_1}$, $K'(1)$ and $K''(1)$ are respectively given by (17), (44) and (45). ■

Using (47), we plot the mean delay as a function of M in Fig. 8. As we can see, the mean delay is 410 slots when $M = 1$, and rapidly drops to 117 slots when M slightly increases from 1 to 5. Also, as displayed in Fig. 6(a), the buffer of a node can almost be emptied in a single busy period when $M = 5$ and $\hat{\lambda} = 0.2$ packets/slot. Thus, further increasing M does not reduce the mean delay any more. Fig. 8 indicates that a small M is enough for the reduction of the mean delay.

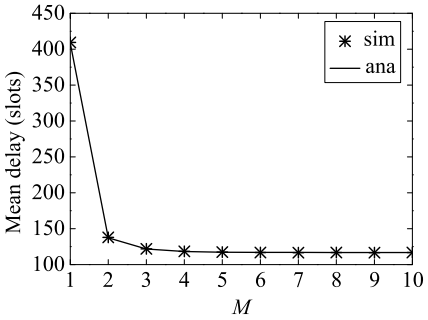


Fig. 8. Mean delay varies with M, where $n = 50$, $\hat{\lambda} = 0.2$ packets/slot, $X = 2$ slots and $r = 0.02$.

As a comparison, the S-Aloha cannot achieve a bounded delay since the aggregate packet arrival rate $\hat{\lambda} = 0.2$ is larger than $e^{-1}/X = 0.184$, that is the capacity that the S-Aloha can provide. This clearly indicates the CS-Aloha with small M is superior to the S-Aloha in terms of delay performance.

Stability is a key issue in the design of Aloha-like protocols. A network is unstable when its mean delay is unbounded. It is known that the stability of Aloha-like protocols is susceptible to the aggregate packet arrival rate $\hat{\lambda}$, the number of nodes in the system, and the transmission probability r [32], [39]. The result in Fig. 8 shows that the CS-Aloha with a large M can reduce the mean delay, which implies that increasing M can potentially make the network more stable. To demonstrate this point, we first investigate the stable condition of the CS-Aloha and the related design issues in Section V-A, and then discuss the effect of M on system performance in Section V-B.

A. Stable Condition and Network Design

From Theorem 2, we find that if $1 - \lambda X - \lambda\overline{Y_1}/M > 0$, the mean delay $\overline{D}$ is bounded. Recall that bounded delay requires that the network satisfies condition (5). Combining these two results, we have the following statement.

Proposition 1: To achieve the bounded delay, the random access network with the CS-Aloha should satisfy the following conditions

$$\begin{cases} 1 - \lambda X - \dfrac{\lambda}{Mre^{-G}} \cdot \dfrac{1 - n\lambda X r}{1 - n\lambda X} > 0 & (48) \\ n\lambda < \dfrac{M}{1/(nre^{-nr}) + MX} & (49) \end{cases}$$

■

Proposition 1 clearly indicates that the condition that the aggregate packet arrival rate is less than the capacity is not sufficient to ensure the bounded delay. This conclusion is consistent with that obtained in [39], which studied the delay performance of another kind of Aloha-like protocol, non-persistent CSMA.

The stable condition can be employed to design the network. One design issue in practice is to estimate the node population that can be supported by a network according to the packet arrival rate of each node λ, which is known by the operator before the network planning. Let $\hat{n}$ be the number of nodes that the network can support when transmission probability r is given. We have

$$n \leq \lfloor n_r \rfloor, \tag{50}$$

according to (48), where $n_r \in \mathbb{R}$ is the real-number root of variable n of the following simultaneous equations of n and G:

$$\begin{cases} 1 - \lambda X - \dfrac{\lambda}{Mre^{-G}} \cdot \dfrac{1 - n\lambda X r}{1 - n\lambda X} = 0 \\ \dfrac{1 - \left(\frac{G}{nr}\right)^M}{1 - \frac{G}{nr}} = \dfrac{n\lambda}{1 - n\lambda X} \cdot \dfrac{1}{Ge^{-G}} \end{cases}$$

and

$$n \leq \left\lfloor \frac{1}{\lambda X} - \frac{1}{r}\mathbb{W}_0\left(\frac{e^{\frac{r}{\lambda X}}}{MX}\right) \right\rfloor, \tag{51}$$

according to (49), where $\mathbb{W}_0(x) > -1$ is a principal branch of Lambert W function $\mathbb{W}(x)e^{\mathbb{W}(x)} = x$ [43]. Combining (50) and (51), the number of nodes that can be supported by the network is given by

$$\hat{n} = \min\left\{ \lfloor n_r \rfloor, \left\lfloor \frac{1}{\lambda X} - \frac{1}{r}\mathbb{W}_0\left(\frac{e^{\frac{r}{\lambda X}}}{MX}\right) \right\rfloor \right\}. \tag{52}$$

When the number of nodes n is determined, another issue of network design is to find a bounded delay region in terms of transmission probability r, such that the network can be stable. Again, this can be achieved by using Proposition 1. On one hand, condition (48) implies that the transmission probability should satisfy $r > r_0$, where r_0 is the root of variable r of the following equation:

$$1 - \lambda X - \frac{\lambda\overline{Y_1}}{M} = 1 - \lambda X - \frac{\lambda}{Mre^{-G}} \cdot \frac{1 - n\lambda X r}{1 - n\lambda X} = 0. \tag{53}$$

On the other hand, condition (49) yields

$$\frac{-\mathbb{W}_0(-S/M)}{n} < r < \frac{-\mathbb{W}_{-1}(-S/M)}{n}, \tag{54}$$

where $S = \frac{\hat{\lambda}}{1-\hat{\lambda}X}$, and $\mathbb{W}_{-1}(x) < -1$ is another principal branch of the Lambert W function. It is easy to prove $r_0 > \frac{-\mathbb{W}_0(-S/M)}{n}$. Thus, once the number of nodes n is given, the bounded delay region is given by

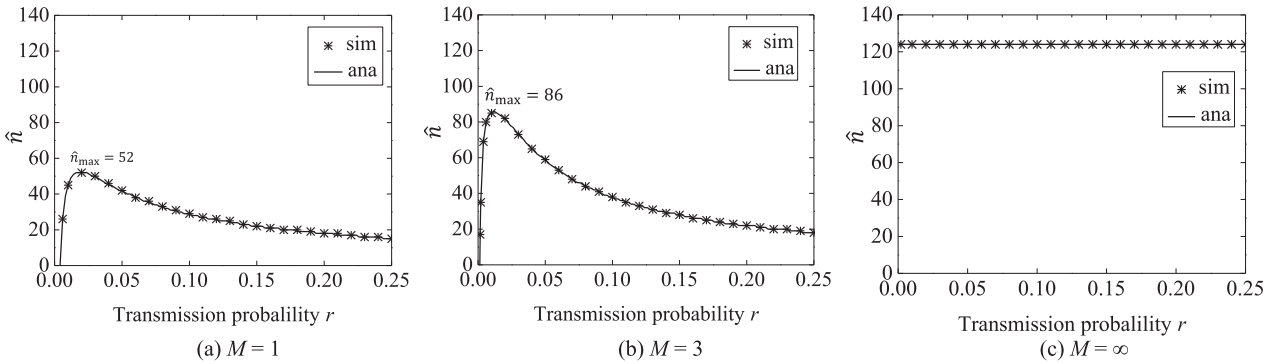
$$r \in \left(r_0, \frac{-\mathbb{W}_{-1}(-S/M)}{n}\right). \tag{55}$$

In the next part, we discuss how batch size M influences the node population that the network can support and the bounded delay region when the node population is given.

B. Effect of Batch Size M on System Performance

To facilitate our discussion, we fix the packet arrival rate per node $\lambda = 0.004$ packets/slot and the service time of a packet $X = 2$ slots.

1) $1 \leq M < \infty$: Fig. 9(a) plots the node population that the CS-Aloha with $M = 1$ can support $\hat{n}$ versus with the transmission probability r. When r is very small, the nodes send so few requests that the channel will be underutilized. In this case, the network can only provide a small capacity and thus accommodate a small number of nodes. With the increase of r, channel utilization is improved and the network can support more nodes. When r increases such that the total capacity the network can offer is maximized, $\hat{n}$ reaches its maximum value, denoted by $\hat{n}_{\max}$. However, if r becomes excessively large, a lot of nodes make attempts in the same slot, and collisions occur frequently. In this case, the network



(a) $M = 1$ (b) $M = 3$ (c) $M = \infty$

Fig. 9. The number of nodes supported by the CS-Aloha vs. transmission probability, where $\lambda = 0.004$ packets/slot and $X = 2$ slots.

capacity degrades, and $\hat{n}$ declines accordingly. For example, $\hat{n}$ in Fig. 9(a) first increases and reaches the maximum value $\hat{n}_{\max} = 52$ at $r = 0.02$, and then decreases after that.

Given the number of deployed nodes $n \leq \hat{n}_{\max}$, we can derive the mean delay using (47). When $M = 1$, the first and the second moments of the number of packets served in a busy period are $K'(1) = 1$ and $K''(1) = 0$, according to (44) and (45). As APPENDIX B shows, attempt rate $G_{M=1} = -\mathbb{W}_0(-S)$ is a constant as long as the network is stable. Thus, it follows from (47) that the mean delay is given by

$$\overline{D}_{M=1} = \frac{-\mathbb{W}_0(-S) - \hat{\lambda}X\left[-\mathbb{W}_0(-S) - \frac{\lambda X}{2}\right]r}{\hat{\lambda}[1 - \lambda X - \lambda X\mathbb{W}_0(-S)]r + \lambda\mathbb{W}_0(-S)} + X. \quad (56)$$

Fig. 10(a) plots $\overline{D}_{M=1}$ varying with transmission probability r, where $n = 30 < \hat{n}_{\max}$ and $n = 50 < \hat{n}_{\max}$. When $r \to 0$, the node sends few requests, thus a large number of packets accumulate in the buffer. In this case, the mean delay is quite large or unbounded. With the increase of r, the mean delay decreases. However, if r is excessively large, the collision probability of requests becomes prominent and the network will become unstable again. According to (55), when $M = 1$, the bounded delay region is:

$$r \in \left(\frac{-\mathbb{W}_0(-S)}{n[1 - \lambda X - \lambda X\mathbb{W}_0(-S)]}, \frac{-\mathbb{W}_{-1}(-S)}{n}\right). \quad (57)$$

Equation (56) indicates that $\overline{D}_{M=1}$ monotonically decreases with r in the bounded delay region. Therefore, $\overline{D}_{M=1}$ will approach its minimum value when $r \to r^*_{M=1}$, where

$$r^*_{M=1} = \frac{-\mathbb{W}_{-1}(-S)}{n}. \quad (58)$$

As shown in Fig. 10(a), in the network with $n = 30$, the mean delay tends to the minimum value 20 slots if $r \to 0.094$.

Also, as (57) shows, with the growth of node population n, the size of the bounded delay region shrinks and eventually vanishes. As demonstrated in Fig. 10(a), where the size of the bounded delay region reduces from 0.088 to 0.018 when n increases from 30 to 50. Specifically, the bounded delay region disappears and the network is inherently unstable when $n > 52$, which is the maximum node population that the CS-Aloha with $M = 1$ can support. This property is similar to that of the S-Aloha [32].

$\hat{n}$ as a function of r is further studied in Fig. 9(b) when $M = 3$. Comparing Fig. 9(a) and (b), it can be found that $\hat{n}$ at each r is improved when M increases from 1 to 3. In particular, $\hat{n}_{\max}$ increases from 52 to 86.

We then observe the mean delay versus r in Fig. 10(b), where $n = 30$ and $n = 50$. Similarly, we find that the bounded delay region becomes wider with the increase of M. For example, when $n = 50$ and M increases from 1 to 3, the bounded delay region is expanded from (0.012, 0.03) to (0.0025, 0.068). This confirms that increasing M can enlarge the node population capacity and the bounded delay region of r.

Similar to the case of $M = 1$, the bounded delay region of the CS-Aloha with $M = 3$ shrinks with the growth of n, as depicted in Fig. 10(b). When $n > \hat{n}_{\max}$, the network will never be stable no matter how r is tuned. Also, when the network is stable, $\overline{D}_{M=3}$ monotonically decreases with r. According to (55), the CS-Aloha with $M < \infty$ can tend to the minimum mean delay when $r \to r^*_{M<\infty}$, where

$$r^*_{M<\infty} = \frac{-\mathbb{W}_{-1}(-S/M)}{n}. \quad (59)$$

2) $M = \infty$: Recall that a node with a non-empty buffer can succeed in a free slot with probability re^{-G}, which is clearly larger than 0 for any r in the region (0, 1). Once the node takes over the channel, the unlimited batch size (i.e., $M = \infty$) allows the node to send out all the packets waiting in the buffer in the time slot when it succeeds. In other words, the queue length of each node will never increase to infinity, as long as the aggregate packet arrival rate $n\lambda$ is less than $1/X$, which is the capacity that can be offered by the CS-Aloha when $M = \infty$. This point can be verified by the following derivations.

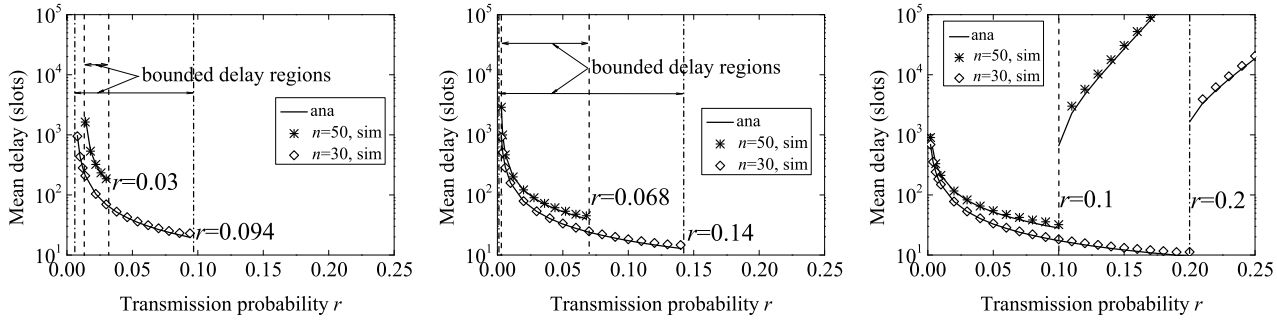
Substituting (17) into (47) and letting $M \to \infty$, we obtain the mean delay when $M = \infty$ as follows:

$$\overline{D}_{M=\infty} = \frac{\frac{\lambda X^2}{2} + \frac{1}{re^{-G}} \cdot \frac{1 - \hat{\lambda}Xr}{1 - \hat{\lambda}X}}{1 - \lambda X} + X, \quad (60)$$

where attempt rate G is determined by the following equation according to (36):

$$\frac{1}{1 - \frac{G}{nr}} = \frac{\hat{\lambda}}{1 - \hat{\lambda}X} \cdot \frac{1}{Ge^{-G}}. \quad (61)$$

Equation (60) clearly shows that the mean delay is bounded as long as $n\lambda < 1/X$ and $0 < r < 1$. This means that the number of nodes that can be supported by the CS-Aloha with $M = \infty$ is a constant $\hat{n} = \lfloor 1/(\lambda X) \rfloor$, and the network with $n \leq \lfloor 1/(\lambda X) \rfloor$ will be stable at any $r \in (0, 1)$, which is confirmed by Fig. 9(c) and Fig. 10.



(a) $M = 1$ (b) $M = 3$ (c) $M = \infty$

Fig. 10. Mean delay and bounded delay region of the CS-Aloha, where $\lambda = 0.004$ packets/slot and $X = 2$ slots.

Though $\overline{D}_{M=\infty}$ is finite for all the rs in the region $(0, 1)$ as long as $n \leq \lfloor 1/(\lambda X) \rfloor$, it could be very large when r is not properly set. For example, when $n = 50$ and $r > 0.1$, $\overline{D}_{M=\infty} > 10^3$ slots. This indicates that r should be carefully selected even when $M = \infty$. Substituting (61) into (60) to get rid of r, we have

$$\overline{D}_{M=\infty} = \frac{\lambda X^2}{2(1-\lambda X)} - \frac{1}{\lambda(1-\lambda X)} + \frac{n(1-\lambda XG)e^G}{(1-\lambda X)(1-\hat{\lambda} X)G} + X. \quad (62)$$

We show in APPENDIX C that $\overline{D}_{M=\infty}$ first decreases and then increases with the increase of G. We also demonstrate in APPENDIX B that G monotonically increases with r. This implies that $\overline{D}_{M=\infty}$ has a minimum value when r changes from 0 to 1. For example, the mean delay when $n = 50$ in Fig. 10 reaches the minimum value at $r = 0.1$. APPENDIX C further gives the optimum transmission probability $r^*_{M=\infty}$ in (C-5) and (C-7), and the minimum mean delay in (C-4) and (C-6).

VI. Delay Distribution and Delay Jitter

In wireless access networks, voice and video communications are popular applications, which are sensitive to delay jitter. Thus, it is necessary to evaluate the delay jitter of the CS-Aloha. The delay jitter, denoted by σ_D, is the standard deviation of the delay, which can be derived from the delay distribution. In this section, we first use the technique presented in [44] to solve the delay distribution and then calculate the delay jitter.

Consider the i-th packet that leaves the queue in a busy period of a node, where $i = 1, 2, \cdots, K$ and $K = \min\{Q, M\}$ is the number of packets served in a busy period. Let Π_i be the queue length left behind by the i-th packet and Φ_i be the number of arrivals during the service time of the i-th packet. We thus have

$$\Pi_i = Q + \Phi_1 + \Phi_2 + \cdots + \Phi_i - i, \quad (63)$$

where Q is the queue length at the start of a busy period. Since a packet arrives at the node in a slot with probability λ and the service time of a packet is X slots, the probability generating function of Φ_i is given by

$$\Phi_i(z) = (1 - \lambda + \lambda z)^X. \quad (64)$$

Let Π be the queue length of a node seen by a departure packet of this node and $\Pi(z) \triangleq E[z^{\Pi}]$ be the probability generating function of Π. It follows from (63) and (64) that $\Pi(z)$ can be shown in the following equation (65) at the bottom of the next page.

Note that Π is also the number of arrivals during the sojourn time D that the packet spends in the node. According to the description in Section V, the delay of a packet mainly consists of a residual time, several complete vacation periods, and several busy periods. Because a vacation period is much longer than a busy period, the number of arrivals during the vacation periods is the major components of Π. Recall that the number of arrivals during a vacation period U is a geometric random variable when node population n is sufficiently large. Following the argument similar to that of Q, it is reasonable to imagine that Π could also be geometrically distributed when n is large. We show that this is true as follows:

As Section IV mentions, $\lambda X = \frac{\hat{\lambda} X}{n} < \frac{1}{n} \to 0$ when n is sufficiently large, and thus we have $(1-\lambda+\lambda z)^X = 1-\lambda X+\lambda X z + o\left(\frac{1}{n}\right) \to 1$. Replacing $(1 - \lambda + \lambda z)^X$ with 1 in (65), we obtain

$$\Pi(z) = \frac{\alpha}{1-(1-\alpha)z}, \quad (66)$$

where $\alpha = 1 - \frac{G}{nr}$ as (32) shows. Equation (66) clearly indicates that Π approaches to a geometric random variable for a large n.

Let $D(z) \triangleq E[z^D]$ be the probability generating function of D. With the distribution of Π, the delay distribution can be derived.

Theorem 3: The generating function of the delay distribution of the CS-Aloha is given by

$$D(z) = \frac{\gamma}{1-(1-\gamma)z}, \quad (67)$$

where γ is given as follows:

$$\gamma = \frac{\lambda\alpha}{1-\alpha+\lambda\alpha} = \frac{\lambda(nr-G)}{G+\lambda(nr-G)}. \quad (68)$$

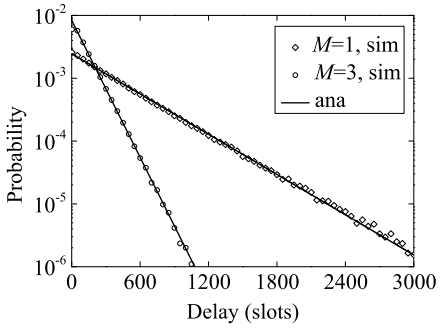
Proof: According to assumption A4 and following the relation between Π and D, we have

$$\Pi(z) = E[(1-\lambda+\lambda z)^D] = D(1-\lambda+\lambda z),$$

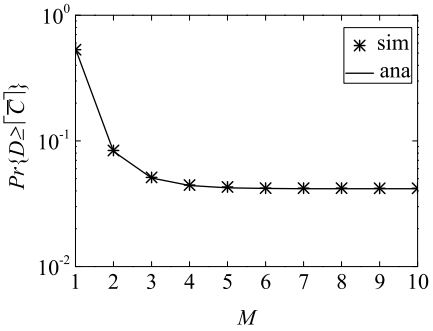
that is

$$D(z) = \Pi\left(\frac{z-1+\lambda}{\lambda}\right). \quad (69)$$

Substituting (66) into (69), we obtain (67). ■



(a) probability distribution



(b) $Pr\{D \geq \lceil\overline{C}\rceil\}$

Fig. 11. Delay distribution and $Pr\{D \geq \lceil\overline{C}\rceil\}$, where $n = 50$, $\hat{\lambda} = 0.2$ packets/slot, $X = 2$ slots, and $r = 0.02$.

Theorem 3 clearly indicates that the delay distribution is a geometric distribution when n is large. According to (67), we have

$$Pr\{D = j\} = \gamma(1-\gamma)^j. \tag{70}$$

The accuracy of (70) is confirmed by the simulation result in Fig. 11(a), which shows that D approaches a geometric distribution when $n = 50$. Furthermore, we can solve the mean delay from (67) as follows:

$$\overline{D} = D'(1) = \frac{1-\gamma}{\gamma} = \frac{G}{\lambda(nr-G)}. \tag{71}$$

It is easy to verify that (71) is numerically equal to (47), which confirms the validity of our results.

As Section III shows, the CS-Aloha with $M > 1$ can suppress the queue length, which implies that it can suppress the tail distribution of packet delay. Herein, we check the probability that the packet delay is larger than the mean cycle time, $Pr\{D \geq \lceil\overline{C}\rceil\}$. A small $Pr\{D \geq \lceil\overline{C}\rceil\}$ means that a packet can be transmitted in one cycle with high probability. According to (70), we have

$$Pr\{D \geq \lceil\overline{C}\rceil\} = \sum_{j=\lceil\overline{C}\rceil}^{\infty} \gamma(1-\gamma)^j = (1-\gamma)^{\lceil\overline{C}\rceil}, \tag{72}$$

which decreases with γ. Remember that $M > 1$ can reduce attempt rate G and thus increase γ. Hence, the result in (72) implies that, the CS-Aloha with $M > 1$ can lower $Pr\{D \geq \lceil\overline{C}\rceil\}$. As an example, Fig. 11(b) shows that $Pr\{D \geq \lceil\overline{C}\rceil\}$ declines very fast with the growth of M.

As a result, the fluctuation range of packet delay will shrink, and thus the delay jitter will decrease. According to (67), we calculate the delay jitter as follows:

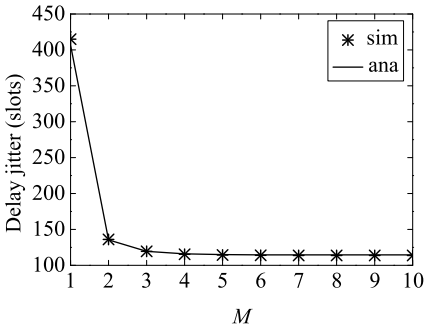
$$\begin{aligned}\sigma_D &= \sqrt{D''(1) + D'(1) - [D'(1)]^2} \\ &= \sqrt{\left[\frac{G}{\lambda(nr-G)}\right]^2 + \frac{G}{\lambda(nr-G)}}.\end{aligned} \tag{73}$$

Since G decreases with M, Eq. (73) indicates that the CS-Aloha with $M > 1$ can reduce σ_D. Fig. 12(a) plots the delay jitter versus M. Similar to the mean delay in Fig. 8, the delay jitter drops very fast with the growth of M and converges to 117 slots when $M = 5$. To be more intuitive, Fig. 12(b) further plots the queue length of a node varying with the time when $M = 1$ and 3. As we can see, the queue length fluctuates between 0 and 12 when $M = 1$, while it only varies between 0 and 3 when $M = 3$.

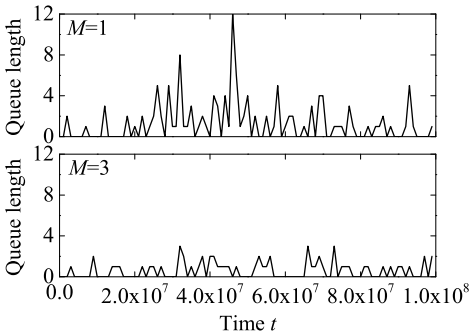
VII. Applications

An application of our model is underwater acoustic networks [14]–[18]. The underwater acoustic network is a wireless random access network with the following two features. First, the nodes are battery powered and battery replenishment is very difficult [14]–[17]. The node is discarded, if the battery runs out. For the purpose of power saving, how to enhance network throughput while reducing packet collisions is an important issue. In this context, the CS-Aloha based protocols emerged, e.g., T-Lohi [14], [15] and Slotted FAMA (S-FAMA) [16]–[18]. Second, the channel is error-prone due to fouling,

$$\begin{aligned}\Pi(z) &= \frac{E\left[\sum_{i=1}^{K} z^{\Pi_i}\right]}{E[K]} \\ &= \frac{E\left[\sum_{i=1}^{K} z^{Q+\Phi_1+\Phi_2+\cdots+\Phi_i-i}\right]}{E[K]} \\ &= \frac{\sum_{j=1}^{\infty} E\left[\sum_{i=1}^{K} z^{Q+\Phi_1+\Phi_2+\cdots+\Phi_i-i} | Q=j, \quad K=\min\{j,M\}\right] q_j}{\sum_{j=1}^{\infty} E\left[K=\min\{Q,M\} | Q=j\right] q_j} \\ &= \frac{\sum_{j=1}^{M}\sum_{i=1}^{j} z^{j-i}(1-\lambda+\lambda z)^{Xi}\alpha(1-\alpha)^{j-1} + \sum_{j=M+1}^{\infty}\sum_{i=1}^{M} z^{j-i}(1-\lambda+\lambda z)^{Xi}\alpha(1-\alpha)^{j-1}}{\sum_{j=1}^{M} j\alpha(1-\alpha)^{j-1} + \sum_{j=M+1}^{\infty} M\alpha(1-\alpha)^{j-1}} \\ &= \frac{\alpha}{1-(1-\alpha)^M} \cdot \frac{(1-\lambda+\lambda z)^X[1-(1-\alpha)^M(1-\lambda+\lambda z)^{XM}]}{z-(1-\lambda+\lambda z)^X} \left\{\frac{\alpha z}{1-(1-\alpha)z} - \frac{\alpha(1-\lambda+\lambda z)^X}{1-(1-\alpha)(1-\lambda+\lambda z)^X}\right\}\end{aligned} \tag{65}$$



(a) delay jitter varies with M



(b) queue length changes with time

Fig. 12. (a) Delay jitter varies with M and (b) queue length changes with time, where $n = 50$, $\hat{\lambda} = 0.2$ packets/slot, $X = 2$ slots, and $r = 0.02$.

corrosion, and plentiful underwater noises [15], [45], [46]. According to [15], the typical bit error rate of underwater acoustic networks is 10^{-5}. Thus, the transmission failure of requests and data packets should be taken into consideration in the design of the network.

We take S-FAMA in [16], [17] as an example to demonstrate how our model can be applied to modelling underwater acoustic networks. As plotted in Fig. 13, each node in S-FAMA sends a request-to-send (RTS) to the AP before data transmission. If the RTS is sent successfully, the AP will return a clear-to-send (CTS). Thus, one time slot in S-FAMA includes the transmission time of the RTS and the waiting time for the CTS, as illustrated in Fig. 13. After receiving the CTS, the node monopolizes the channel to transmit packets. Each packet is followed by an ACK if it is transmitted without error. In the case where the channel is error-prone, the transmission of RTS/CTS and packet/ACK may fail. An RTS (or a packet) is transmitted successfully only if the node receives the CTS (or ACK) from the AP. If the node does not receive the CTS from the AP, it reattempts to send the RTS in the next slot with probability r. Similarly, the node keeps on retransmitting the packet if the packet is not acknowledged. Clearly, S-FAMA is quite similar to CS-Aloha, except that the channel is error-prone. In this section, we extend our model to analyze S-FAMA, assuming that the channel conditions in the downlink direction (from AP to nodes) are the same and the CTS/ACK can or cannot be received correctly by all the nodes at the same time.

A. Throughput, Mean Delay, and Delay Jitter

Let b be the bit error rate (BER), and s_R, s_P, and s_{ACK} be the size of a request, a packet, and an ACK, respectively. Define θ_R and θ_P as the probability of successful request transmission and that of successful packet transmission, given that no collisions occur. We have

$$\theta_R = (1-b)^{s_R+s_{ACK}}, \tag{74}$$

and

$$\theta_P = (1-b)^{s_P+s_{ACK}}. \tag{75}$$

Let G_e be the attempt rate when the channel is error-prone. Owing to the request collisions and the channel errors, the probability that a node can make a successful attempt in a free slot is $\theta_R G_e e^{-G_e}$. After the node succeeds, each packet transmission may fail with probability $1-\theta_P$. Let T be the service time of a packet, which is defined as the duration from the time when a node begins to send the packet to the time when the node receives the ACK for the packet. Clearly, $T \geq X$ due to a non-ideal channel. In particular, the distribution of service time can be written down as follows

$$Pr\{T = jX\} = \theta_P(1-\theta_P)^{j-1}, \quad j = 1, 2, \cdots,$$

of which the generating function is

$$T(z) = \sum_{j=1}^{\infty} z^{jX}\theta_P(1-\theta_P)^{j-1} = \frac{\theta_P z^X}{1-(1-\theta_P)z^X}. \tag{76}$$

Thus, the first and second moments of service time are given by

$$\overline{T} = T'(1) = \frac{X}{\theta_P}, \tag{77}$$

and

$$T''(1) = \frac{2X^2 - X(X+1)\theta_P}{\theta_P^2}. \tag{78}$$

Following the method in Sections II through IV, we can derive the equation for attempt rate G_e as follows:

$$\frac{1-\left(\frac{G_e}{nr}\right)^M}{1-\frac{G_e}{nr}} = \frac{\hat{\lambda}}{1-\hat{\lambda}\overline{T}} \cdot \frac{1}{\theta_R G_e e^{-G_e}}. \tag{79}$$

Accordingly, we can obtain the following results of the case where the channel is error-prone:

(1) Saturated throughput

$$\eta_{sat,e} = \frac{\tau}{\frac{1}{M(1-b)^{s_R+s_{ACK}}nre^{-nr}} + \frac{X}{(1-b)^{s_P+s_{ACK}}}}. \tag{80}$$

(2) Mean delay

$$\overline{D_e} = \frac{\frac{\lambda T''(1)}{2} + \overline{Y_{1,e}}\left[1 - \frac{(1+\lambda\overline{T})K_e''(1)}{2MK_e'(1)}\right]}{1-\lambda\overline{T}-\frac{\lambda\overline{Y_{1,e}}}{M}} + \overline{T}, \tag{81}$$

where

$$K_e'(1) = \frac{1-\left(\frac{G_e}{nr}\right)^M}{1-\frac{G_e}{nr}} \tag{82}$$

and

$$K_e''(1) = \frac{\frac{2G_e}{nr}\left[1 - M\left(1-\frac{G_e}{nr}\right)\left(\frac{G_e}{nr}\right)^{M-1} - \left(\frac{G_e}{nr}\right)^M\right]}{\left(1-\frac{G_e}{nr}\right)^2} \tag{83}$$

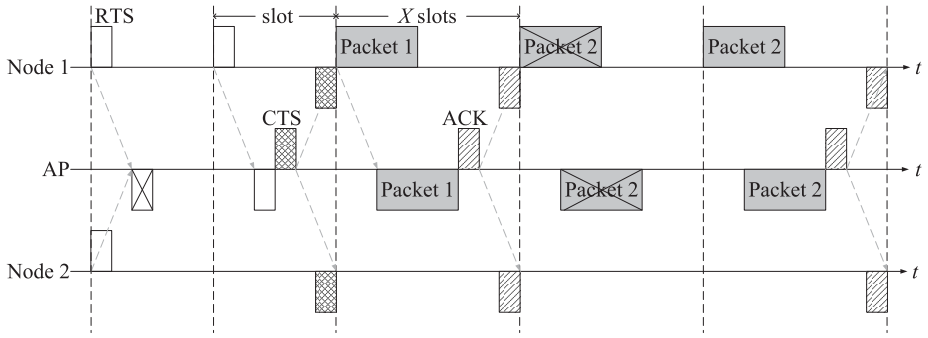


Fig. 13. Illustration of the S-FAMA.

are the first and second moments of the number of packets served in a busy period, and

$$\overline{Y_{1,e}} = \frac{1}{\theta_R r e^{-G_e}} \cdot \frac{1-\hat{\lambda}\overline{T}r}{1-\hat{\lambda}\overline{T}} \tag{84}$$

is the mean vacation period of a node that starts with a non-empty buffer.

(3) Number of nodes that can be supported by the network

$$\hat{n}_e = \min\left\{\lfloor n_{r,e}\rfloor, \left\lfloor \frac{1}{\lambda\overline{T}} - \frac{1}{r}\mathbb{W}_0\left(\frac{e^{\frac{r}{\lambda\overline{T}}}}{\theta_R M\overline{T}}\right)\right\rfloor\right\}, \tag{85}$$

where $n_{r,e} \in \mathbb{R}$ is the real-number root of n for the following simultaneous equations:

$$\begin{cases} 1-\lambda\overline{T} - \dfrac{\lambda}{M\theta_R r e^{-G_e}} \cdot \dfrac{1-n\lambda\overline{T}r}{1-n\lambda\overline{T}} = 0 \\ \dfrac{1-\left(\frac{G_e}{nr}\right)^M}{1-\frac{G_e}{nr}} = \dfrac{n\lambda}{1-n\lambda\overline{T}} \cdot \dfrac{1}{\theta_R G_e e^{-G_e}} \end{cases} \tag{86}$$

(4) Delay jitter

$$\sigma_{D_e} = \sqrt{\left[\frac{G_e}{\lambda(nr-G_e)}\right]^2 + \frac{G_e}{\lambda(nr-G_e)} + T''(1) + T'(1) - [T'(1)]^2}. \tag{87}$$

It is easy to verify that when bit error rate $b = 0$, $\eta_{sat,e}$ in (80), $\overline{D_e}$ in (81), $\hat{n}_e$ in (85), and σ_{D_e} in (87) will change to η_{sat} in (2), $\overline{D}$ in (47), $\hat{n}$ in (52), and σ_D in (73), respectively.

B. Performance of S-FAMA

We evaluate the performance of underwater acoustic networks, using the parameters listed in Table I [16]. According to these parameters, we have:

(1) The duration of a time slot of the underwater acoustic network using S-FAMA protocol is 1.53s. In particular, the transmission time of RTS is 0.1s, and the waiting time for the CTS is 1.43s. This implies $1-\delta = 0.1/1.53 = 0.07$ and $\delta = 1.43/1.53 = 0.93$.

(2) The packet arrival rate of each node is $\lambda = 1/300 \cdot 1.53 \approx 0.005$ packets/slot.

(3) The service time of a packet is 4.43s, including the 3-s transmission time of the packet and the 1.43-s waiting time for the ACK, and thus $X = 4.43/1.53 \approx 3$ slots.

Herein, we only consider the S-FAMA protocol with infinite batch size.

TABLE I
PARAMETERS EMPLOYED IN OUR STUDY

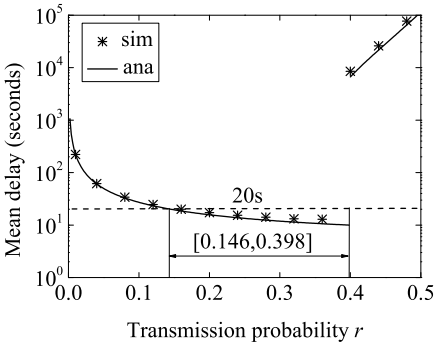
Packet	3000 bits
RTS	100 bits
CTS/ACK	100 bits
Channel rate	1000 bits/s
Transmission distance	1000m
Acoustic speed	1500 m/s
Packet arrival rate	1/300 packets/s
Number of nodes	16

A protocol for underwater acoustic networks is flexible and can be adaptive to different application scenarios if it can support more nodes at the same time. We thus first check the number of nodes $\hat{n}_e$ that can be supported by the protocol. According to Eq. (85), we can easily know that S-FAMA can support $\hat{n}_e = \lfloor \frac{\theta_P}{\lambda X} \rfloor = \left\lfloor \frac{(1-10^{-5})^{3100}}{0.005 \cdot 3} \right\rfloor = 64$ nodes, which is independent of the transmission probability r. As a comparison, the classical S-Aloha can only support 23 nodes.

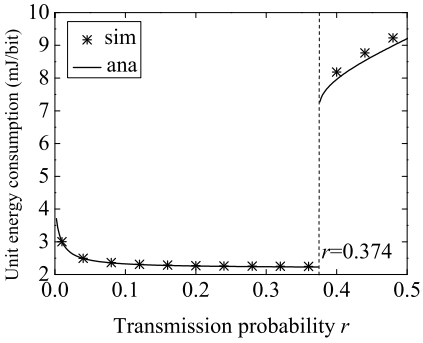
Given $n \le \hat{n}_e$, transmission probability r is a key parameter that influences network performance. As Section V mentions, $M = \infty$ can make the network stable for any $r \in (0, 1)$. If r is very small, the network is underutilized and will suffer a high mean delay. If r is excessively large, RTS collisions happen frequently, which will not only lead to a large delay but also waste a lot of energy. Thus, r should be carefully selected to meet the requirements on the mean delay and the node lifetime.

In the example defined by Table I, suppose that the requirement of the mean delay is $\overline{D_e} \le 20$ seconds. It is clear from Eq. (81) that transmission probability r should be adjusted in the region [0.146, 0.398], as shown in Fig. 14(a).

To evaluate the energy consumption of S-FAMA, we define unit energy consumption as the average energy consumed per bit. Clearly, given the battery size of a node, it can transmit more data before the battery runs out, if the unit energy consumption is low. The power consumption of an acoustic modem is 2W of power when it is transmitting data, and 0.02W when it is idle or receiving the CTS/ACK [14]. Let Δ be the duration of a time slot. Recall that a non-empty node on average spends $\overline{Y_{1,e}}$ slots on channel competition before it succeeds. In particular, the node makes attempts in $1/(\theta_R e^{-G})$ slots, and stays idle in $\overline{Y_{1,e}} - 1/(\theta_R e^{-G})$ slots during channel competition. If a node sends a request in a slot, it consumes $2(1-\delta)\Delta + 0.02\delta\Delta$ J; otherwise, it consumes 0.02Δ J. On the other hand, a node on average transmits $\overline{K_e}$ packets during the busy period after it succeeds. The average service time of a packet is $\overline{T}$ slots, in which the node spends



(a) mean delay



(b) unit energy consumption

Fig. 14. Mean delay and unit energy consumption vary with transmission probability.

$\frac{\overline{T}(X-\delta)}{X}$ slots on packet transmission and waits $\frac{\overline{T}\delta}{X}$ slots for the ACK. Thus, a node consumes $\left[\frac{2\Delta\overline{T}(X-\delta)}{X}+\frac{0.02\Delta\overline{T}\delta}{X}\right]\overline{K_e}$ J on average during a busy period. As a result, we have the unit energy consumption as follows:

$$\begin{aligned}\overline{E_P} &= \frac{[2(1-\delta)\Delta+0.02\delta\Delta]\cdot\frac{1}{\theta_R e^{-G}}+0.02\Delta\left(\overline{Y_{1,e}}-\frac{1}{\theta_R e^{-G}}\right)}{\overline{K_e}s_P} \\ &\quad+\frac{\left[\frac{2\overline{T}(X-\delta)\Delta}{X}+\frac{0.02\overline{T}\delta\Delta}{X}\right]\overline{K_e}}{\overline{K_e}s_P} \\ &= \frac{\overline{Y_{1,e}}}{\overline{K_e}}\cdot\frac{0.02\Delta}{s_P}+\frac{1.98(1-\delta)(1-\hat{\lambda}\overline{T})G\Delta}{\hat{\lambda}s_P}+\frac{(2X-1.98\delta)\overline{T}\Delta}{Xs_P},\end{aligned} \tag{88}$$

where $\frac{\overline{Y_{1,e}}}{\overline{K_e}}$ is the amortized collision overhead to successfully transmit a packet. It is obvious that the smaller the overhead is, the lower the unit energy consumption is. In the current example, the unit energy consumption achieves its minimal value 2.23 mJ/bit when $r = 0.374$, as depicted in Fig. 14(b). As a comparison, if S-Aloha is applied to underwater acoustic networks [47], [48], the minimum unit energy consumption will be 3.5 mJ/bit, which is 1.5 times as high as that of S-FAMA with $M = \infty$.

VIII. Conclusion

In this paper, we developed a vacation queueing model with limited-service discipline to study the delay performance and the stable condition of the CS-Aloha with batch size M. Based on this model, we derive the delay distribution, the mean delay, the delay jitter and the bounded delay region. Our analysis shows that the attempt process of each node in the CS-Aloha is essentially a series of Bernoulli trials and thus the delay distribution is a geometric distribution when node population is sufficiently large. Also, our analytical results demonstrate that increasing M can accelerate the clean-up of the buffer in each node, and thus decrease the attempt rate in each slot, which increases the successful probability of each node in attempt process. As a result, the CS-Aloha with $M > 1$ can decrease the mean delay and the delay jitter, and enlarge the bounded delay region. In particular, when M is infinity, the network can be stable for all the transmission probability rs in the region (0, 1), as long as the aggregate packet arrival rate $n\lambda$ is smaller than $1/X$. We also derived the optimum transmission probability that is required to achieve the minimum mean delay under different values of M.

References

[1] S. Lam and L. Kleinrock, "Packet switching in a multiaccess broadcast channel: Dynamic control procedures," *IEEE Trans. Commun.*, vol. 23, no. 9, pp. 891–904, Sep. 1975.

[2] L. Kleinrock and S. Lam, "Packet switching in a multiaccess broadcast channel: Performance evaluation," *IEEE Trans. Commun.*, vol. 23, no. 4, pp. 410–423, Apr. 1975.

[3] L. Kleinrock and S. S. Lam, "Packet-switching in a slotted satellite channel," in *Proc. Nat. Comput. Conf. Expo. (AFIPS)*, 1973, pp. 703–710.

[4] N. Abramson, "Packet switching with satellites," Hawaii Univ., Honolulu, HI, USA, Tech. Rep. B73-2, 1973.

[5] D. Vukobratovic, "Massive machine-type communications and revival of ALOHA," in *Proc. INFOTEH-JAHORINA*, vol. 16, 2017, pp. 1–4.

[6] Q. Song, X. Lagrange, and L. Nuaymi, "An analytical model for S-ALOHA performance evaluation in M2M networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–7.

[7] H. Wu, C. Zhu, R. J. La, X. Liu, and Y. Zhang, "FASA: Accelerated S-ALOHA using access history for event-driven M2M communications," *IEEE/ACM Trans. Netw.*, vol. 21, no. 6, pp. 1904–1917, Dec. 2013.

[8] D. Wang, X. Hu, F. Xu, H. Chen, and Y. Wu, "Performance analysis of P-CSMA for underwater acoustic sensor networks," in *Proc. Oceans (Yeosu)*, May 2012, pp. 1–6.

[9] Y. Zhou, K. Chen, J. He, and H. Guan, "Enhanced slotted aloha protocols for underwater sensor networks with large propagation delay," in *Proc. IEEE 73rd Veh. Technol. Conf. (VTC Spring)*, May 2011, pp. 1–5.

[10] S. Rashwand, J. Misic, and V. B. Misic, "Analysis of CSMA/CA mechanism of IEEE 802.15.6 under non-saturation regime," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 5, pp. 1279–1288, May 2016.

[11] *IEEE Standard for Local and Metropolitan area Networks—Part 15.6: Wireless Body Area Networks*, IEEE Standard 802.15.6, 2012, pp. 1–271.

[12] 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.

[13] Y. Gao and L. Dai, "Random access: Packet-based or connection-based?" *IEEE Trans. Wireless Commun.*, vol. 18, no. 5, pp. 2664–2678, May 2019.

[14] A. A. Syed, W. Ye, and J. Heidemann, "Comparison and evaluation of the T-Lohi MAC for underwater acoustic sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 9, pp. 1731–1743, Dec. 2008.

[15] A. A. Syed, W. Ye, and J. Heidemann, "T-lohi: A new class of MAC protocols for underwater acoustic sensor networks," in *Proc. 27th Conf. Comput. Commun. (IEEE INFOCOM)*, Apr. 2008, pp. 231–235.

[16] M. Molins and M. Stojanovic, "Slotted FAMA: A MAC protocol for underwater acoustic networks," in *Proc. OCEANS (Asia–Pacific)*, May 2006, pp. 1–7.

[17] P. Casari, B. Tomasi, and M. Zorzi, "A comparison between the tone-lohi and slotted FAMA MAC protocols for underwater networks," in *Proc. OCEANS*, Sep. 2008, pp. 1–8.

[18] W. Huang, M. Liu, and S. Zhang, "SFAMA-MM: A slotted FAMA based MAC protocol for multi-hop underwater acoustic networks with a multiple reception mechanism," in *Proc. 37th Chin. Control Conf. (CCC)*, Jul. 2018, pp. 7315–7321.

[19] 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.
[20] 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.
[21] N. Abramson, "THE ALOHA SYSTEM: Another alternative for computer communications," in *Proc. FJCC*, 1970, pp. 281–285.
[22] L. Dai, "Toward a coherent theory of CSMA and aloha," *IEEE Trans. Wireless Commun.*, vol. 12, no. 7, pp. 3428–3444, Jul. 2013.
[23] N. Abramson, "The throughput of packet broadcasting channels," *IEEE Trans. Commun.*, vol. 25, no. 1, pp. 117–128, Jan. 1977.
[24] L. Barletta, F. Borgonovo, and I. Filippini, "The S-Aloha capacity: Beyond the e^{-1} myth," in *Proc. IEEE INFOCOM*, Apr. 2016, pp. 1–9.
[25] L. Barletta, F. Borgonovo, and I. Filippini, "The throughput and access delay of slotted-aloha with exponential backoff," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 451–464, Feb. 2018.
[26] R.-H. Gau, "Performance analysis of slotted aloha in interference-dominating wireless ad-hoc networks," *IEEE Commun. Lett.*, vol. 10, no. 5, pp. 402–404, May 2006.
[27] Y. Li and L. Dai, "Maximum sum rate of slotted aloha with capture," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 690–705, Feb. 2016.
[28] A. Munari, M. Heindlmaier, G. Liva, and M. Berioli, "The throughput of slotted aloha with diversity," in *Proc. 51st Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Oct. 2013, pp. 698–706.
[29] W. Luo and A. Ephremides, "Stability of n interacting queues in random-access systems," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1579–1587, Jul. 1999.
[30] R. R. Rao and A. Ephremides, "On the stability of interacting queues in a multiple-access system," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 918–930, Sep. 1988.
[31] B. S. Tsybakov and V. A. Mikhailov, "Ergodicity of a slotted ALOHA system," *Problemy Peredachi Informatsii*, vol. 15, no. 4, p. 73–87, 1979.
[32] L. Dai, "Stability and delay analysis of buffered aloha networks," *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2707–2719, Aug. 2012.
[33] X. Pan, T. Ye, T. T. Lee, and W. Hu, "Power efficiency and delay tradeoff of 10GBase-T energy efficient Ethernet protocol," *IEEE/ACM Trans. Netw.*, vol. 25, no. 5, pp. 2773–2787, Oct. 2017.
[34] X. Sun and L. Dai, "Performance optimization of CSMA networks with a finite retry limit," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 5947–5962, Sep. 2016.
[35] X. Sun and L. Dai, "Backoff design for IEEE 802.11 DCF networks: Fundamental tradeoff and design criterion," *IEEE/ACM Trans. Netw.*, vol. 23, no. 1, pp. 300–316, Feb. 2015.
[36] J.-B. Seo, H. Jin, and V. C. M. Leung, "Throughput upper-bound of slotted CSMA systems with unsaturated finite population," *IEEE Trans. Commun.*, vol. 61, no. 6, pp. 2477–2487, Jun. 2013.
[37] L. Dai and X. Sun, "A unified analysis of IEEE 802.11 DCF networks: Stability, throughput, and delay," *IEEE Trans. Mobile Comput.*, vol. 12, no. 8, pp. 1558–1572, Aug. 2013.
[38] J.-W. Cho, J.-Y. Le Boudec, and Y. Jiang, "On the asymptotic validity of the decoupling assumption for analyzing 802.11 MAC protocol," *IEEE Trans. Inf. Theory*, vol. 58, no. 11, pp. 6879–6893, Nov. 2012.
[39] 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.
[40] S. M. Ross, *Introduction to Probability Models*, 9th ed. New York, NY, USA: Academic, 2006.
[41] D. P. Bertsekas, R. G. Gallager, and P. Humblet, *Data Networks*, vol. 2. Upper Saddle River, NJ, USA: Prentice-Hall, 1992.
[42] H. Huang, T. Ye, and T. T. Lee, "Optimum transmission window for EPONs with limited service," *IEEE Access*, vol. 7, pp. 57956–57971, 2019.
[43] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth, "On the LambertW function," *Adv. Comput. Math.*, vol. 5, no. 1, pp. 329–359, Dec. 1996.
[44] H. Takagi, *Queueing Analysis: A Foundation of Performance Evaluation*, vol. 1. Amsterdam, The Netherlands: North Holland, 1991.
[45] S. Jiang, "State-of-the-Art medium access control (MAC) protocols for underwater acoustic networks: A survey based on a MAC reference model," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 96–131, 1st Quart., 2018.
[46] R. B. Manjula and S. S. Manvi, "Issues in underwater acoustic sensor networks," *Int. J. Comput. Electr. Eng.*, vol. 3, no. 1, pp. 101–110, 2011.
[47] P. Mandal, S. De, and S. S. Chakraborty, "Characterization of aloha in underwater wireless networks," in *Proc. Nat. Conf. Commun. (NCC)*, Jan. 2010, pp. 1–5.
[48] A. A. Syed, W. Ye, J. Heidemann, and B. Krishnamachari, "Understanding spatio-temporal uncertainty in medium access with ALOHA protocols," in *Proc. 2nd Workshop Underwater Netw. (WuWNet)*, 2007, pp. 41–48.

APPENDIX A
PROOF OF LEMMA 2

Equations (26) and (30) in Section IV clearly indicate that the probability generation function of the queue length at the beginning of the busy period $Q(z)$ and that of the queue length at the end of the busy period $P(z)$ couple with each other. Owing to such a coupling relationship, it is hard to solve $Q(z)$ in the general case. In this appendix, however, we prove that Q is geometrically distributed when the number of nodes n is sufficiently large.

Assume that Q obeys a geometric distribution with parameter α, that is

$$q_j = \alpha(1-\alpha)^{j-1}, j = 1, 2, \cdots \tag{A-1}$$

where $\alpha \in (0, 1)$. Thus, the generating function of Q is

$$Q(z) = \frac{\alpha z}{1-(1-\alpha)z}. \tag{A-2}$$

In the following, we prove Lemma 2 by checking if (30) holds under this assumption.

Substituting (A-1) into (26) and (27), $P(z)$ and p_0 are respectively given by

$$P(z) = \frac{\alpha(1-\lambda+\lambda z)^X \left[1-(1-\alpha)^{M-1}(1-\lambda+\lambda z)^{X(M-1)}\right]}{1-(1-\alpha)(1-\lambda+\lambda z)^X} + \frac{\alpha(1-\alpha)^{M-1}(1-\lambda+\lambda z)^{MX}}{1-(1-\alpha)z}, \tag{A-3}$$

and

$$p_0 = P(0) = \frac{\alpha(1-\lambda)^X[1-(1-\alpha)^M(1-\lambda)^{XM}]}{1-(1-\alpha)(1-\lambda)^X}. \tag{A-4}$$

Substituting (A-3), (A-4), (12) and (13) into the right-hand side of (30), we obtain

$$\begin{aligned}
&p_0U_0(z) + [P(z)-p_0]U_1(z)\\
=&\left[P(z)-p_0+p_0z\right]U_1(z) + o\left(\frac{1}{n}\right)\\
=&\left\{\frac{\alpha(1-\lambda+\lambda z)^X\left[1-(1-\alpha)^{M-1}(1-\lambda+\lambda z)^{X(M-1)}\right]}{1-(1-\alpha)(1-\lambda+\lambda z)^X}\right.\\
&+\frac{\alpha(1-\alpha)^{M-1}(1-\lambda+\lambda z)^{XM}}{1-(1-\alpha)z}\\
&\left.+\frac{\alpha(1-\lambda)^X\left[1-(1-\alpha)^M(1-\lambda)^{XM}\right]}{1-(1-\alpha)(1-\lambda)^X}(z-1)\right\}\\
&\times\frac{\beta}{1-(1-\beta)z}(1-\lambda+\lambda z) + o\left(\frac{1}{n}\right).
\end{aligned} \tag{A-5}$$

In the following, we show that β can be expressed as a function of α. Recall that $Q = P + U$, as (29) shows. Thus, $\overline{Q}$ satisfies,

$$\overline{Q} = \overline{P} + \overline{U} = \overline{P} + p_0\overline{U_0} + (1-p_0)\overline{U_1}, \tag{A-6}$$

where

$$\overline{P} = P'(1) = \frac{(1-\alpha)^M + \lambda X[1-(1-\alpha)^M]}{\alpha}, \tag{A-7}$$

and

$$\overline{U_0} = 1 + \overline{U_1} = \frac{1}{\beta} + \lambda. \tag{A-8}$$

According to (A-2), $\overline{Q}$ is also given by

$$\overline{Q} = Q'(1) = \frac{1}{\alpha}. \tag{A-9}$$

Combining (A-4) and (A-6)-(A-9), the parameter β can be expressed as a function of α as follows:

$$\beta = \frac{1}{1-\lambda+\frac{(1-\lambda X)[1-(1-\alpha)^M]}{\alpha} - \frac{\alpha(1-\lambda)^X[1-(1-\alpha)^M(1-\lambda)^{XM}]}{1-(1-\alpha)(1-\lambda)^X}}. \tag{A-10}$$

As Section IV mentions, $\lambda X \to 0$ when n is sufficiently large, which implies $(1-\lambda+\lambda z)^X \to 1$, $(1-\lambda)^X \to 1$, and $1-\lambda X \to 1$ for large n. Substituting (A-10) into (A-5) and replacing $(1-\lambda+\lambda z)^X$, $(1-\lambda)^X$, and $1-\lambda X$ with 1, we can simplify (A-5) as follows

$$\begin{aligned}
&p_0U_0(z) + [P(z)-p_0]U_1(z)\\
=&\left\{\frac{\alpha(1-\alpha)^{M-1}}{1-(1-\alpha)z} + \left[1-(1-\alpha)^M\right]z - \alpha(1-\alpha)^{M-1}\right\}\\
&\times\frac{\frac{\alpha}{1-(1-\alpha)^{M+1}}}{1-\left(1-\frac{\alpha}{1-(1-\alpha)^{M+1}}\right)z}\\
=&\frac{\alpha}{1-(1-\alpha)z}\\
=&Q(z)
\end{aligned} \tag{A-11}$$

This indicates that Lemma 2 is established.

In the following, we determine the value of parameter α using (22). In the CS-Aloha, only the first M packets can be transmitted if the queue length at the start of the busy period is larger than M; otherwise, all packets in the buffer can be served. Thus, according to (A-1), the mean number of packets served in a busy period is given by

$$\overline{K} = \sum_{j=1}^{M-1} jq_j + \sum_{j=M}^{\infty} Mq_j = \frac{1-(1-\alpha)^M}{\alpha}. \tag{A-12}$$

On the other hand, applying the condition $(1-\lambda)^X \to 1$ when n is sufficiently large, we can obtain from (A-4) the probability that the buffer is empty at the end of the busy period as follows

$$p_0 \approx 1-(1-\alpha)^M. \tag{A-13}$$

Substituting (A-12) and (A-13) into (22), we obtain

$$G = nr(1-\alpha), \tag{A-14}$$

which implies

$$\alpha = 1 - \frac{G}{nr}.$$

APPENDIX B
SOLUTION OF ATTEMPT RATE G

In this appendix, we discuss how to solve the attempt rate G from (36) in Theorem 1.

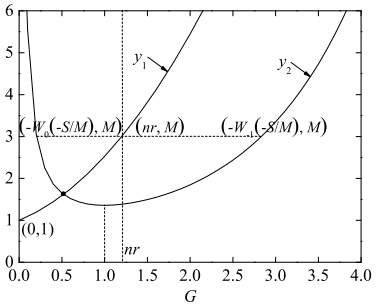


Fig. 1. Solution of attempt rate G where $n = 20$, $\hat{\lambda} = 0.25$ packets/slot, $X = 2$ slots, $r = 0.06$, and $M = 3$.

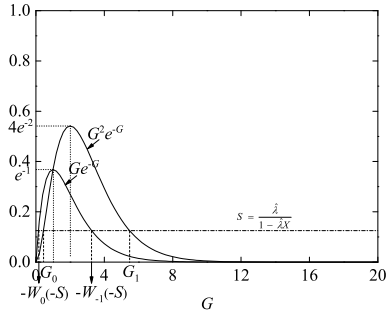


Fig. 2. G^2e^{-G} and Ge^{-G} versus attempt rate G, where $\hat{\lambda} = 0.1$ packets/slot, $X = 2$ slots, and $n = 20$.

A. $M = 1$

When $M = 1$, Eq. (36) changes to

$$\frac{\hat{\lambda}}{1-\hat{\lambda}X} \cdot \frac{1}{Ge^{-G}} = 1.$$

This equation has two roots, $-\mathbb{W}_0(-S)$ and $-\mathbb{W}_{-1}(-S)$, where $S = \hat{\lambda}/(1-\hat{\lambda}X)$. As [1] shows, $-\mathbb{W}_{-1}(-S)$ is an unstable equilibrium at which the network is transiently stable, while $-\mathbb{W}_0(-S)$ is a stable point at which the network is stable in the long term. In other words, though G may temporarily stay at $-\mathbb{W}_{-1}(-S)$, it will eventually converge to $-\mathbb{W}_0(-S)$ in the steady state.

B. $1 < M < \infty$

In this case, the root of (36) is the value of G at the cross point of curves $y_1 = \frac{1-\left(\frac{G}{nr}\right)^M}{1-\frac{G}{nr}}$ and $y_2 = \frac{S}{Ge^{-G}}$ in the region $0 < G < nr$ since $G = np_{ne}r$. As shown in Fig. 1, with the increase of G, y_1 monotonically increases from 1 and goes through points $(0,1)$ and (nr, M), while y_2 first decreases from ∞ and then climbs up after it reaches the minimum value at $G = 1$. Also, $y_2\left(-\mathbb{W}_0(-S/M)\right) = y_2\left(-\mathbb{W}_{-1}(-S/M)\right) = M$. In a stable network, the aggregate packet arrival rate $\hat{\lambda}$ should be less than $\frac{M}{\frac{1}{nre^{-nr}}+MX}$, which implies

$$-\mathbb{W}_0(-S/M) < nr < -\mathbb{W}_{-1}(-S/M).$$

Therefore, y_1 and y_2 have only one cross point in the region $G \in (0, nr)$, as illustrated in Fig. 1. This means (36) has only one desirable solution of attempt rate G when $1 < M < \infty$.

C. $M = \infty$

When $M = \infty$, Eq. (36) can be rewritten as follows:

$$\frac{SG}{n(S-Ge^{-G})} = r, \tag{B-1}$$

where $S = \hat{\lambda}/(1-\hat{\lambda}X)$. The roots of (B-1) are the values of G at the cross points of two curves $y_1(G) \triangleq \frac{SG}{n(S-Ge^{-G})}$ and $y_2(G) = r$. Clearly, the number of cross points depends on the monotonicity of y_1. Taking the derivative of y_1 with respect to attempt rate G, we have

$$y_1' = \frac{S(S-G^2e^{-G})}{n(S-Ge^{-G})^2}, \tag{B-2}$$

the sign of which is determined by $S - G^2e^{-G}$. As plotted in Fig. 2, G^2e^{-G} is a bell-shape curve. With the growth of G, G^2e^{-G} first rises from 0 to the maximum value $4e^{-2}$ at $G = 2$, and then gradually falls to 0 again. When $S < 4e^{-2}$, i.e., $0 < \hat{\lambda} < 4/(e^2+4X)$, y_1 has three monotone intervals:

(a) $(0, G_0)$: y_1 is monotonically increasing since $y_1' > 0$,
(b) $[G_0, G_1]$: y_1 is monotonically decreasing since $y_1' \leq 0$,
(c) (G_1, n): y_1 is monotonically increasing since $y_1' > 0$,

where $G_0 = -2\mathbb{W}_0\left(-\sqrt{S}/2\right)$ and $G_1 = -2\mathbb{W}_{-1}\left(-\sqrt{S}/2\right)$ are two roots of $S - G^2e^{-G} = 0$. When $4/(e^2+4X) \leq \hat{\lambda} < 1/X$, $y_1' > 0$ and thus y_1 is a monotonous increasing function for all Gs. This indicates that the monotonicity of y_1 depends on the value of $\hat{\lambda}$.

On the other hand, since $0 < y_2 = r < 1$, y_1 and y_2 have cross points only when $y_1 > 0$ (i.e., S should be larger than Ge^{-G}). Consider the fact that the maximum value of Ge^{-G} is e^{-1}. As displayed in Fig. 2, if $S > e^{-1}$, i.e., $\hat{\lambda} > 1/(e+X)$, $S > Ge^{-G}$ holds if $G \in (0, n)$; otherwise, $S > Ge^{-G}$ only holds if $G \in (0, -\mathbb{W}_0(-S) \cup (-\mathbb{W}_{-1}(-S), n)$. Thus, whether (B-1) has solutions also depends on the value of $\hat{\lambda}$.

Thus, we should discuss the solutions of (B-1) in the three regions of $\hat{\lambda}$: $(0, 1/(e+X)]$, $(1/(e+X), 4/(e^2+4X)]$, and $(4/(e^2+4X), 1/X)$.

1) $0 < \hat{\lambda} \leq 1/(e+X)$*:* In this case, y_1 and y_2 have cross points only when attempt rate G is in $(0, -\mathbb{W}_0(-S) \cup (-\mathbb{W}_{-1}(-S), n)$, and y_1 has three monotone intervals: $(0, G_0)$, $[G_0, G_1]$, and (G_1, n).

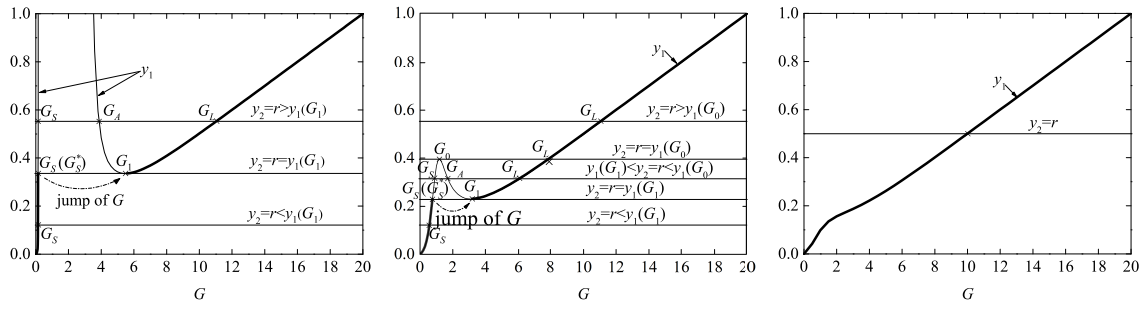
As plotted in Fig. 2, $Ge^{-G} > G^2e^G$ when $G < 1$, and $Ge^{-G} < G^2e^{-G}$ when $G > 1$, which implies the boundaries of different regions of G have the following relationship:

$$\begin{aligned} &-\mathbb{W}_0(-S) < G_0 = -2\mathbb{W}_0(-\sqrt{S}/2) < 1 < \\ &-\mathbb{W}_{-1}(-S) < G_1 = -2\mathbb{W}_{-1}(-\sqrt{S}/2). \end{aligned} \tag{B-3}$$

Thus, Eq. (B-1) has solutions in the following three monotone intervals:

(a) $(0, -\mathbb{W}_0(-S)$: increasing interval;
(b) $(-\mathbb{W}_{-1}(-S), G_1]$: decreasing interval;
(c) (G_1, n): increasing interval.

Fig. 3(a) illustrates these monotone intervals. Since $y_1'(G_1) = 0$, $y_1(G_1)$ is a local minimum of y_1, as depicted in Fig. 3(a). Table I gives the values of G at the cross points of y_1 and y_2 when r is in a different region, where G_S, G_A and G_L are



(a) $\hat{\lambda} = 0.1$ (b) $\hat{\lambda} = 0.23$ (c) $\hat{\lambda} = 0.3$

Fig. 3. y_1 and y_2 v.s. G under different aggregate packet arrival rates, where $X = 2$ slots and $n = 20$.

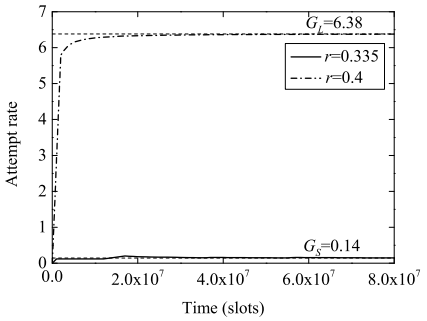


Fig. 4. Attempt rate changes with time, where $\hat{\lambda} = 0.1$ packet/slot, $X = 2$ slots, and $n = 20$.

the values in monotone intervals (a), (b) and (c), respectively. These values can be obtained by numerically solving (B-1). From Table I, we can see that

(1) If $0 < r < y_1(G_1)$, Eq. (B-1) has only one root G_S and the attempt rate will converge to this root in a stable network.
(2) If $r = y_1(G_1)$, there are two roots G_S and $G_A = G_L = G_1$. Using the drift analysis in [1], it is easy to show that the attempt rate converges to G_S when the network reaches its stable state.
(3) If $y_1(G_1) < r < 1$, there are three roots G_S, G_A and G_L, and the attempt rate will converge to G_L.

Thus, as shown in Fig. 3(a), when $0 < \hat{\lambda} \leq 1/(e + X)$, the value range of G at the steady state is a non-contiguous region $(0, G_S^*] \cup (G_1, n)$, where G_S^* is the value of G_S when $r = y_1(G_1)$. In other words, though G can be adjusted by tuning r when $M = \infty$, it will experience a sudden change from G_S^* to G_1 at $r = y_1(G_1)$. We use the thick lines in Fig. 3(a) to indicate the changing trajectory of attempt rate G if we tune r from 0 to 1 in a stable network.

As an example, we consider a network with $n = 20$, $\hat{\lambda} = 0.1$ packets/slot and $X = 2$ slots. In this case, $G_1 = -2\mathbb{W}_{-1}(-\sqrt{S}/2) = 5.48$. We set $r = y(G_1) = 0.335$ and $r = 0.4 > y(G_1) = 0.335$, which correspond to the cases displayed in the second row and the third row of Table I. Our simulation results in Fig. 4 show that the attempt rate converges to $G_S = 0.14$ if $r = 0.335$ and to $G_L = 6.38$ if $r = 0.4$, as the network reaches the steady state.

2) $1/(e + X) < \hat{\lambda} \leq 4/(e^2 + 4X)$*:* In this case, y_1 and y_2 have cross points in the following monotone intervals of y_1:

(a) $(0, G_0)$: increasing interval;
(b) $[G_0, G_1]$:decreasing interval;
(c) (G_1, n): increasing interval.

It follows that $y_1(G_0)$ and $y_1(G_1)$ are, respectively, the local maximum and the local minimum of y_1, as plotted in Fig. 3(b). TABLE II gives the roots for Eq. (B-1). Similar to the previous case, we find that

(1) If $r \leq y_1(G_1)$, the attempt rate converges to G_S,
(2) otherwise, the attempt rate converges to G_L,

when the network reaches the stable state. This indicates that, when $1/(e+X) < \hat{\lambda} \leq 4/(e^2+4X)$, the value range of G at the steady state is also a non-contiguous range $(0, G_S^*] \cup (G_1, n)$, the same with the situation in the case of $0 < \hat{\lambda} \leq 1/(e + X)$. We use the thick line in Fig. 3(b) to display the trajectory of G if we tune r from 0 to 1 in a stable network.

3) $4X/(e^2 + 4X) < \hat{\lambda} < 1/X$*:* In this situation, y_1 monotonically increases when G increases from 0 to n. As shown in Fig. 3(c), y_1 and y_2 have only one cross point such that (B-1) has only one desirable root and G at the steady state changes in a continuous region $(0, n)$.

TABLE II
THE ROOTS FOR EQUATION (B-1) WHEN $1/(e + X) < \hat{\lambda} \leq 4/(e^2 + 4X)$

	$G \in (0, G_0)$	$G \in [G_0, G_1]$	$G \in [G_1, n)$
$0 < r < y_1(G_1)$	G_S	-	-
$r = y_1(G_1)$	G_S	$G_A = G_1$	$G_L = G_1$
$y_1(G_1) < r < y_1(G_0)$	G_S	G_A	G_L
$r = y_1(G_0)$	$G_S = G_0$	$G_A = G_0$	G_L
$y_1(G_0) < r < 1$	-	-	G_L

APPENDIX C
MINIMUM MEAN DELAY WHEN $M = \infty$

To obtain the minimum value of $\overline{D}_{M=\infty}$ given by (62), we take the derivative of $\overline{D}_{M=\infty}$ with respect to G as follows:

$$\frac{d\overline{D}_{M=\infty}}{dG} = -\frac{\hat{\lambda} X e^G \left(G - \frac{1+\sqrt{1-4\lambda X}}{2\lambda X}\right)\left(G - \frac{1-\sqrt{1-4\lambda X}}{2\lambda X}\right)}{(1 - \lambda X)(1 - \hat{\lambda} X) G^2}. \tag{C-1}$$

This implies

$$\begin{cases} \dfrac{d\overline{D}_{M=\infty}}{dG} < 0, & \text{if } G < \frac{1-\sqrt{1-4\lambda X}}{2\lambda X} \text{ or } G > \frac{1+\sqrt{1-4\lambda X}}{2\lambda X} \\ \dfrac{d\overline{D}_{M=\infty}}{dG} > 0, & \text{otherwise.} \end{cases} \tag{C-2}$$

TABLE I
THE ROOTS FOR EQUATION (B-1) WHEN $0 < \hat{\lambda} < e^{-1}$

	$G \in (0, -\mathbb{W}_0(-S))$	$G \in (-\mathbb{W}_{-1}(-S), G_1]$	$G \in [G_1, n)$
$0 < r < y_1(G_1)$	G_S	-	-
$r = y_1(G_1)$	G_S	$G_A = G_1$	$G_L = G_1$
$y_1(G_1) < r < 1$	G_S	G_A	G_L

On the other hand, though attempt rate G monotonically increases with transmission probability r, its value range is different when $\hat{\lambda}$ is different. As APPENDIX B shows,

(a) when $0 < \hat{\lambda} \leq 4/(e^2 + 4X)$, G is in the range $(0, G_S^*] \cup (G_1, n)$;
(b) when $4/(e^2 + 4X) < \hat{\lambda} < 1/X$, G is in the range $(0, n)$.

It is easy to show that

$$G_S^* < \frac{1 - \sqrt{1 - 4\lambda X}}{2\lambda X} < 2 < G_1 < n < \frac{1 + \sqrt{1 - 4\lambda X}}{2\lambda X}. \tag{C-3}$$

Thus, we can derive the minimum mean delay under different ranges of $\hat{\lambda}$.

(a) $0 < \hat{\lambda} \leq 4/(e^2+4X)$: As APPENDIX B mentions, though G can be adjusted by tuning r, it will experience a non-contiguous change from G_S^* to G_1 at $r = y_1(G_1)$ if $0 < \hat{\lambda} \leq 4/(e^2 + 4X)$ and $M = \infty$. This implies that $\overline{D}_{M=\infty}$ may also have a sudden change at $r = y_1(G_1)$ in this case, as we show below. According to (C-2) and (C-3), $\overline{D}_{M=\infty}$ is a decreasing function when $G \in (0, G_S^*]$ and an increasing function when $G \in (G_1, n)$. It is easy to show that the mean delay at G_S^* is less than that at G_1. Thus, $\overline{D}_{M=\infty}$ reaches its minimum

$$\overline{D}^*_{M=\infty} = \frac{\lambda X^2}{2(1-\lambda X)} - \frac{1}{\lambda(1-\lambda X)} + \frac{n(1-\lambda X G_S^*)e^{G_S^*}}{(1-\lambda X)(1-\hat{\lambda}X)G_S^*} + X, \tag{C-4}$$

at G_S^*. According to (B-1), the corresponding transmission probability is given by

$$r^*_{M=\infty} = y_1(G_S^*) = \frac{\lambda G_S^*}{\hat{\lambda} - (1 - \hat{\lambda}X)G_S^* e^{-G_S^*}}. \tag{C-5}$$

When r is slightly larger than $r^*_{M=\infty}$, the mean delay will suddenly change from $\overline{D}^*_{M=\infty}$ to

$$\overline{D}_{M=\infty} = \frac{\lambda X^2}{2(1-\lambda X)} - \frac{1}{\lambda(1-\lambda X)} + \frac{n(1-\lambda X G_1)e^{G_1}}{(1-\lambda X)(1-\hat{\lambda}X)G_1} + X.$$

For example, $\overline{D}_{M=\infty}$ in Fig. 10(c) jumps from 28 slots to 661 slots after $r = r^*_{M=\infty} = 0.1$, if $n = 50$.

(b) $4/(e^2 + 4X) < \hat{\lambda} < 1/X$: According to (C-2) and (C-3), $\overline{D}_{M=\infty}$ first falls and then rises when G increases from 0 to n. $\overline{D}_{M=\infty}$ achieves its minimum

$$\overline{D}^*_{M=\infty} = \frac{\lambda X^2}{2(1-\lambda X)} - \frac{1}{\lambda(1-\lambda X)} + \frac{n\left(1 - \frac{1-\sqrt{1-4\lambda X}}{2}\right)e^{\frac{1-\sqrt{1-4\lambda X}}{2\lambda X}}}{(1-\lambda X)(1-\hat{\lambda}X)\frac{1-\sqrt{1-4\lambda X}}{2\lambda X}} + X, \tag{C-6}$$

at

$$G = \frac{1 - \sqrt{1 - 4\lambda X}}{2\lambda X}.$$

According to (B-1), the corresponding transmission probability is given as

$$r^*_{M=\infty} = \frac{\frac{1-\sqrt{1-4\lambda X}}{2X}}{\hat{\lambda} - (1 - \hat{\lambda}X)\frac{1-\sqrt{1-4\lambda X}}{2\lambda X} e^{-\frac{1-\sqrt{1-4\lambda X}}{2\lambda X}}}. \tag{C-7}$$

APPENDIX D
MEAN NUMBER OF COMPLETE VACATION PERIODS EXPERIENCED BY A PACKET

Consider packet A in Fig. 7. If packet A arrives in a busy period, it will experience $1 + \left\lfloor \frac{N_O}{M} \right\rfloor$ complete vacation periods before it can be served; otherwise, it will undergo $\left\lfloor \frac{N_O}{M} \right\rfloor$ complete vacation periods, where N_O is the number of packets waiting outside the virtual gate of buffer when it arrives. Since a node is busy with probability λX and is in vacation with probability $1 - \lambda X$, the mean number of complete vacation periods experienced by packet A is given by

$$\begin{aligned} \overline{F} &= \lambda X \left(1 + E\left[\left\lfloor \frac{N_O}{M} \right\rfloor\right]\right) + (1 - \lambda X) E\left[\left\lfloor \frac{N_O}{M} \right\rfloor\right] \\ &= \lambda X + E\left[\left\lfloor \frac{N_O}{M} \right\rfloor\right]. \end{aligned} \tag{D-1}$$

Let Ω be the number of packets transmitted before packet A in the busy period that it is served. Clearly, $\Omega = N_O - M\left\lfloor \frac{N_O}{M} \right\rfloor$. Thus, we have

$$E\left[\left\lfloor \frac{N_O}{M} \right\rfloor\right] = \frac{E[N_O] - E[\Omega]}{M}. \tag{D-2}$$

Notice that packet A sees N packets queued in the buffer when it arrives, which includes N_I packets waiting inside the gate and N_O packets waiting outside the gate. It follows that

$$E[N_O] = E[N] - E[N_I] = \lambda \overline{W} - \lambda E[\Omega] X, \tag{D-3}$$

where we apply Little's law for the second equality and $E[\Omega]$ can be derived as follows. Given packet A is transmitted in a busy period that j packets are served, the mean number of packets transmitted ahead of it is $\frac{j-1}{2}$. It is easy to show that the probability that packet A is sent in such a kind of busy period is $\frac{jk_j}{\sum_{i=1}^{M} ik_i}$, where $k_j \triangleq Pr\{K = j\}$ is the probability that j packets are served in a busy period. Thus, we have

$$E[\Omega] = \sum_{j=1}^{M} \frac{j-1}{2} \cdot \frac{jk_j}{\sum_{i=1}^{M} ik_i} = \frac{K''(1)}{2K'(1)}. \tag{D-4}$$

Combining (D-1) and (D-4), we obtain

$$\overline{F} = \lambda X + \frac{\lambda \overline{W}}{M} - \frac{(1 + \lambda X)K''(1)}{2MK'(1)}$$

REFERENCES

[1] L. Dai, "Stability and delay analysis of buffered Aloha networks," *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2707–2719, 2012.