The Impact of Access Probabilities on the Delay Performance of *Q-CSMA* Algorithms in Wireless Networks

Javad Ghaderi, *Student Member, IEEE*, and R. Srikant, *Fellow, IEEE*

***Abstract*—It has been recently shown that queue-based carrier sense multiple access (CSMA) algorithms are throughput-optimal. In these algorithms, each link of the wireless network has two parameters: a transmission probability and an access probability. The transmission probability of each link is chosen as an appropriate function of its queue length, however the access probabilities are simply regarded as some random numbers since they do not play any role in establishing the network stability. In this paper, we show that the access probabilities control the mixing time of the CSMA Markov chain and, as a result, affect the delay performance of the CSMA. In particular, we derive formulas that relate the mixing time to access probabilities and use these to develop the following guideline for choosing access probabilities: Each link i should choose its access probability equal to $1/(d_i+1)$, where d_i is the number of links that interfere with link i. Simulation results show that this choice of access probabilities results in good delay performance.**

***Index Terms*—Carrier sense multiple access (CSMA), Markov chain, scheduling, wireless network.**

I. Introduction

SCHEDULING in wireless networks is of fundamental importance due to the inherent broadcast property of the wireless medium. Two radios might not be able to transmit simultaneously because they create too much interference for each other causing the signal-to-noise-plus-interference ratio (SINR) at their corresponding receivers to go below the required threshold for successful decoding of the packets. Therefore, at each time, a *scheduling algorithm* (*MAC protocol*) is needed to schedule a subset of users that can transmit successfully at the same time.

The performance metrics used to evaluate a scheduling algorithm are *throughput* and *delay*. Throughput is characterized by the largest set of arrival rates under which the algorithm can stabilize the queues in the network. The delay performance of a scheduling algorithm can be characterized by the average delay experienced by the packets transmitted in the network. The design of efficient scheduling algorithms, to achieve maximum throughput and low delay, is the main objective of this paper. It is also essential for the scheduling algorithms to be distributed and have low complexity/overhead, since in many wireless networks there is no centralized entity and the resources at the nodes are very limited.

The wireless network is often modeled by its *conflict graph* (or interference model) to capture the interference constraints or technological ones (for example, a node cannot transmit and receive at the same time). In the conflict graph, two communication links form two neighboring nodes of the graph if they cannot transmit simultaneously. Therefore, at each time-slot, the active links should form an independent set of the conflict graph, i.e., no two scheduled nodes can share an edge in the conflict graph. The well-known result of Tassiulas and Ephremides [1] states that the *Maximum Weight Scheduling* (MWS) algorithm, where weights are functions of queue-lengths, is throughput-optimal in the sense that it can stabilize the queues in the network for all arrival rates in the capacity region of the network (without explicitly knowing the arrival rates). However, for a general network, MWS involves finding the maximum weight independent set of the conflict graph, with time-varying weights, in each time-slot, which requires the network to solve a complex combinatorial problem in each time-slot and, hence, is not implementable in practice. This has led to a rich amount of literature on design of approximate algorithms to alleviate the computational complexity of the MWS algorithm. For example, alternatives such as *Maximal Scheduling* and *Greedy Maximal Scheduling* have low complexity, but in general these algorithms can only guarantee stability for a fraction of the capacity region (see, e.g., [12]–[14]).

Carrier sense multiple access (CSMA)-type algorithms are an important class of scheduling algorithms due to their simplicity of implementation, and they have been widely used in practice, e.g., in WLANs (IEEE 802.11 Wi-Fi) or emerging wireless mesh networks. In these protocols, each user listens to the channel and can transmit, with some probability, only when the channel is not busy. In this paper, we consider design of CSMA algorithms in order to maximize throughput and improve delay performance.

From a local perspective, the CSMA algorithm might seem easy to understand, but, at a global perspective, interactions among different users might lead to a very complicated behavior that makes the performance characterization difficult. In recent years, fairly simple models have been proposed that are useful in predicting the throughput of the CSMA algorithm [2], [4]–[7]. A

Manuscript received February 26, 2012; revised July 11, 2012 and August 24, 2012; accepted August 24, 2012; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor E. Ekici. This work was supported by the NSF under Grant CNS 07-21286, the AFOSR MURI under Grant FA 9550-10-1-0573, and ARO MURI Projects W911NF-07-1-0287 and W911NF-08-1-0233. An earlier version of this paper appeared in the Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), Orlando, FL, March 25–30, 2012.

The authors are with the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA (e-mail: jghaderi@illinois.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNET.2012.2215964

more detailed representation of the IEEE 802.11 backoff mechanism can be found in [3]. While these models are idealized, experimental measurements in actual IEEE 802.11 networks match remarkably well with throughput estimates provided by these models [8].

Building upon the above-mentioned models, recently, it has been shown that it is possible to design CSMA algorithms that are throughput-optimal, e.g., see [9] and [15] for the continuous-time CSMA and [10] and [19] for the discrete-time CSMA. The common component in all these works is a Markov chain (called CSMA Markov chain) over the space of feasible schedules. The transition probabilities of the CSMA chain are controlled, by queue lengths or the differences between the average arrival rates and the average departure rates of the links, to make sure that a suitable schedule is selected at each time.

Essentially, the prior works on CSMA are mostly concerned with ensuring network stability. Their main focus is often on solving the maximum weight independent set problems in a distributed manner by using the so-called Glauber Dynamics. In CSMA algorithms, each user has two parameters: an *access probability* that controls how often the user tries to access the channel and a *transmission probability* that controls the length of the data transmission once the user acquires access to the channel. In the traditional *ALOHA* protocol, for a network of N users, the access probabilities $\{a_l\}_{l=1}^N$ are chosen to be $\frac{1}{N}$ in order to maximize the throughput and the maximum throughput per user is approximately $\frac{1}{Ne}$. However, in the CSMA schemes, as we will see in Section II, one of the parameters is fixed and the other parameter is controlled, as a function of the user's local information to achieve the maximum throughput.

Unfortunately, such maximum throughput guarantees have been established under a range of parameters that induce excessive backlogs and delays. This has triggered a strong interest in developing approaches for improving the delay performance of CSMA. Reference [17] proposes a modified CSMA scheme, called Unlocking CSMA, which requires all nodes to become silent periodically and operate as usual CSMA between such epochs. They show that such unlocking can yield optimal throughput and order-optimal delay in torus/grid graph topologies for uniform traffic patterns. The delay-order, as the number of nodes N in the network grows, has been characterized in [16], [21], and [22]. As shown in [16], for *general networks*, it may not be possible to design low-complexity scheduling algorithms that can achieve low delay (polynomial in the number of nodes) for a small fraction (depending on the number of nodes) of the capacity region. References [21] and [22] have reported some positive results but again for some *specific topologies*; they show that CSMA can achieve low delay ($O(\log N)$ or $O(1)$ as $N \to \infty$) when arrival rates are within a small fraction of the capacity region. Establishing or improving such asymptotic results is not the objective of the current paper. Instead, we consider the delay performance of CSMA for general arrival rates (inside the capacity region) and general network topologies and investigate the potential to improve the delay performance by optimizing over the range of parameters of CSMA. In particular, we consider *Queue-based CSMA* (Q-CSMA) schemes, where the access probabilities do not play a role in showing the stability/throughput optimality of CSMA because they do not appear in the steady-state distribution of the CSMA Markov chain. Hence, they have been simply regarded as some constants between zero and one. However, we will see that they do have a significant impact on the *mixing time* of the chain, i.e., the amount of time that it takes to reach close to the steady state starting from some initial condition. Therefore, the access probabilities control the rate at which CSMA responds to the queue dynamics and, hence, have a significant effect on the delay performance of the network. The relationship between the delay of the scheduling algorithm and the mixing time of the CSMA chain has been characterized in [21] where the expected queue length is bounded from above by a multiple of the mixing time (see the proof of [21, Theorem 6]).

A. Main Contributions and Organization

In this paper, we analyze the mixing time of the Q-CSMA Markov chain and develop guidelines to choose access probabilities that result in small mixing times. The main contributions of the paper are the following.

1) In the case of *collocated networks*, we show that access probabilities of the form $1/N$ yield mixing times that are within a constant factor of the optimal mixing time, i.e., the minimum mixing time assuming the global knowledge of the queues/weights of the network.
2) In *d-regular networks*, we show that access probabilities of the form $1/\chi$, when χ is the *chromatic number* of the graph, have the same kind of property when we replace the mixing time with a suitable upper bound on it. In general, $\chi \leq d+1$, nevertheless, replacing the chromatic number with the $d+1$ still yields similar result, but for a larger constant gap.
3) Based on these observations, in general graphs, we conjecture that access probabilities of the form $\{a_l = \frac{1}{d_l+1}\}_{l=1}^N$ should yield good performance, where d_l is the degree of the link l. Our simulation results show that the conjectured access probabilities have a good delay performance. Indeed, they seem to yield average queue lengths that are very close to the smallest queue lengths that can be obtained with any fixed access probabilities.

The remainder of the paper is organized as follows. In Section II, we give an overview the CSMA-type algorithms. In Section III, we briefly explain some preliminaries and definitions used in the proofs of the results. Section IV is devoted to the results for collocated networks. We extend the results to the general networks in Section V. Section VI contains the simulation results. Section VII contains concluding remarks and possible directions for future research. The proofs of the results are provided in the appendixes at the end of the paper.

II. Description of CSMA-Type Algorithms

In this section, we briefly overview the CSMA-type algorithms reported in the literature with more emphasis on the Q-CSMA algorithm, which is the one that has been considered in this paper. We first introduce the following notations.

Let $G(V,E)$ denote the conflict graph of the wireless network consisting of N communication links. Formally, a schedule can be represented by a vector $X = [x_s : s = 1, \dots, N]$ such that $x_s \in \{0,1\}$ and $x_i + x_j \leq 1$ for all $(i,j) \in E$. Let $\mathcal{M}$ denote the set of all feasible schedules and $C(i)$ denote the set of neighbors of i. Then, the basic idea of CSMA is to use *Glauber Dynamics* (to be described) to sample the independent sets of such a graph.

A. Continuous-Time CSMA

In the continuous-time CSMA, each link l has two parameters λ_l and μ_l. The parameter λ_l determines the attempt rate, and μ_l determines the transmission length. In other words, the link l senses the channel at the end of exponentially distributed backoff intervals with the parameter λ_l, and if it detects no ongoing transmissions (the channel is idle), it will transmit for an exponentially distributed amount of time with the mean μ_l. Note that if the links were independent, i.e., no interference constraints, then the stationary probability of a schedule X would be proportional to $\prod_{l\in X}\lambda_l\mu_l$. Now consider a collection of N links with some scheduling constraints captured by a conflict graph. Then, the corresponding Markov chain will be truncation of the previous Markov chain to the set of feasible schedules $\mathcal{M}$. As the result, we have a reversible Markov chain over $\mathcal{M}$ with the stationary distribution

$$\pi(X)=\frac{\prod_{l\in X}\lambda_l\mu_l}{\sum_{Y\in\mathcal{M}}\prod_{l\in Y}\lambda_l\mu_l}\qquad \forall X\in\mathcal{M}.$$

By choosing $\lambda_l\mu_l=e^{w_l}$ where w_l is the weight of the link l, i.e., an appropriate function of its queue length, the stationary distribution will be in the form of

$$\pi(X)=\frac{1}{Z}\exp\left(\sum_{i\in X}w_i\right),\qquad X\in\mathcal{M} \tag{1}$$

where Z is the normalizing constant. Hence, when the weights are large, the algorithm picks the maximum weight schedule with high probability in steady state. Therefore, the algorithm is throughput-optimal [20] if we make sure that the instantaneous probability distribution and the stationary distribution are close enough. To get a faster mixing time, one can let λ_l grow very large (and $\mu_l=\exp(w_l)/\lambda_l$). However, this does not make sense since, in practice, the carrier sensing is performed using energy detection (and hence, cannot be instantaneous) and the backoff interval cannot be smaller than a certain mini-slot. Similarly, the data transmission slot cannot be made arbitrarily small. Moreover, this model is based on a perfect carrier sense assumption and does not consider the collisions due to propagation delays. Thus, in the rest of the paper, we consider the discrete-time CSMA algorithm proposed in [10] called *Q-CSMA*.

B. Q-CSMA [10], [18]

Time is slotted, and arrival process to each link is assumed to be discrete-time, where $A_l(t)$ is the number of packets arriving at link l in time-slot t. For example, $\{A_l(t)\}_{t=0}^{\infty}$, for $l=1,\dots,N$, are independent Bernoulli processes with parameter $\lambda=[\lambda_l; l=1,\dots,N]$. In each time-slot, one packet can be successfully transmitted over a link if there are no other transmissions from the neighboring links.

Each link l is associated with a queue q_l, where the queue dynamics are given by

$$q_l(t)=(q_l(t-1)-x_l(t))^{+}+A_l(t)$$

for $t\geq 0$ and $l=1,\dots,N$. Recall that $x_l(t)=1$ if the link l has been chosen, by Q-CSMA, to be part of the data transmission schedule $X(t)$. The vector of queue lengths is denoted by $q(t)=[q_l(t): l=1,\dots,N]$.

In Q-CSMA, each link l has two parameters a_l and p_l. The parameter a_l is the access probability and chosen to be constant, and p_l is the transmission probability and chosen to be

$$p_l(t)=\frac{e^{w_l(t)}}{1+e^{w_l(t)}} \tag{2}$$

where w_l is an appropriate function of q_l (the queue length at link l) [10], [15], [18]. Let

$$\bar{p}_l(t):=1-p_l(t)=\frac{1}{1+e^{w_l(t)}}.$$

Each time-slot is divided into a control slot and a data slot. In the control slot, each link l that wishes to become part of the data transmission schedule transmits a short control message called an INTENT message with probability a_l. Those links that transmit INTENT messages and do not hear any INTENT messages from the neighboring links constitute a decision schedule.[1] In the data slot, each link l that is included in the decision schedule can transmit a data packet with probability p_l only if none of its neighbors have been transmitting in the previous data slot (see the description of the following algorithm).

Algorithm 1: Q-CSMA in Time-Slot t

1: In the control slot, randomly select a decision schedule $m(t)\subseteq\mathcal{M}$ by using access probabilities $\{a_l\}_{l=1}^{N}$.
2: —$\forall\, i$ in $m(t)$:
If no links in $C(i)$ were active in the previous data slot, i.e., $\sum_{j\in C(i)}x_j(t-1)=0$:
- $x_i(t)=1$ with probability $p_i(t)$, $0<p_i(t)<1$;
- $x_i(t)=0$ with probability $\bar{p}_i(t)=1-p_i(t)$.

Else $x_i(t)=0$.
—$\forall i\notin m(t)$: $x_i(t)=x_i(t-1)$.
3: In the data slot, use $X(t)$ as the transmission schedule.

If the weights are constant, then the above algorithm is the discrete-time version of the Glauber dynamics with multiple-site updates that generates the independent sets of G. Thus, the state space $\mathcal{M}$ consists of all independent sets of G. Q-CSMA algorithm uses a time-varying version of the Glauber dynamics, where the weights change with time. This yields a time-inhomogeneous Markov chain, but for the proper choice of weights that are log-type function of queue-lengths (to be discussed more in Section II-C), it behaves similarly to the Glauber dynamics. By [10, Proposition 1], the stationary distribution with fixed transmission probabilities, i.e., when weights are fixed and do not change with time, is given by

$$\pi(X)=\frac{1}{Z}\prod_{i\in X}\frac{p_i}{\bar{p}_i}\qquad \forall X\in\mathcal{M}.$$

One can check the detailed balanced equations to show that this is the true stationary distribution and the chain is reversible (see

[1]More accurately, it is the transmitter of the link that transmits an INTENT message, and it is the receiver of the link that hears INTENT messages. The node-based implementation of the algorithm is obtained by a trivial modification of the one considered in [11, Appendix] by choosing the control slot to be of size 1. Hence, we do not pursue this issue here and refer the interested reader to [11, Appendix] for all the details.

the proof of Proposition 1 in [10] or [11]). By choosing the transmission probabilities to be in the form of (2), the stationary distribution will be the same as (1) and therefore can pick the maximum-weight schedule with high probability as the queue sizes in the network grow.

Definition 1: The capacity region of the network is defined to be the set of all arrival rates λ that can be supported by the network, i.e., for which there exists a scheduling algorithm that can stabilize the queues. It is known, e.g., [1], that the capacity region is given by

$$\Lambda = \{\lambda \geq 0 : \exists \mu \in Co(\mathcal{M}),\ \lambda < \mu\}$$

where $Co(\cdot)$ is the convex hull operator.[2]

Definition 2: A scheduling algorithm is throughput-optimal if it can stabilize the network for any arrival rate in Λ.

C. Throughput Optimality

The proof of throughput optimality of CSMA algorithms follows from a time-scale separation assumption, i.e., the Markov chain evolves much faster than the rate of changes in the weights (due to queue dynamics in the network) such that the chain always remains close to its stationary distribution. This time-scale separation is justified in [15] and [18]. More precisely, for Q-CSMA, it is shown in [18] that throughput optimality is preserved with weight functions of the form $w(q) = \log(1+q)/g(q)$, where $g(q)$ can be a function that increases arbitrarily slowly, e.g., $w(q) = (\log(1+q))^{1-\epsilon}$ for any small positive ϵ. Roughly speaking, by choosing any function slower than such a $w(\cdot)$, the rate of changes in the weights will be much smaller than the rate at which the CSMA chain responds to these changes, although, for the sake of delay, we will always choose the fastest weight function possible.

Remark 1: It turns out that the mixing time of the CSMA Markov chain plays a fundamental role in establishing the throughput optimality property. We will see later that the mixing time is related to the Second Largest Eigenvalue Modulus (SLEM) of the transition probability matrix of the Markov chain.

Remark 2: Reference [19] develops another version of the CSMA, called preemptive CSMA, which is slightly different from Q-CSMA. In the preemptive CSMA, again each link has two parameters a_l and p_l but there is no exchange of control messages. Each link l grabs the channel with some probability a_l if it does not sense any transmissions from its neighbors. Once it grabs the channel, it transmits a packet. It can continue data transmission with some probability p_l in the subsequent slots. Again the throughput optimality is established under a time-scale separation assumption, while the exact proof can be followed using the techniques in [15] and [18]. Note that under preemptive CSMA, if there is a collision due to propagation delays, the whole data packet will be lost, while in the Q-CSMA, the control messages will be lost which are of smaller size. On the other hand, there is no exchange of the control messages in the preemptive CSMA.

III. Preliminaries

Before we state the main results, some preliminaries regarding the mixing time of Markov chains is needed.

[2]When dealing with vectors, inequalities are interpreted componentwise.

Consider a time-homogenous discrete-time Markov chain over the finite state space $\mathcal{M}$. For simplicity, we index the elements of $\mathcal{M}$ by $1, 2, \ldots, r$, where $r = |\mathcal{M}|$. Assume the Markov chain is irreducible and aperiodic, so that a unique stationary distribution $\pi = [\pi(1), \ldots, \pi(r)]$ always exists.

A. Distance Between Probability Distributions

First, we introduce two convenient norms on $\mathbb{R}^r$ that are linked to the stationary distribution [23]. Let $\ell^2(\pi)$ be the real vector space $\mathbb{R}^r$ endowed with the scalar product

$$\langle z, y\rangle_\pi = \sum_{i=1}^{r} z(i)y(i)\pi(i).$$

Then, the norm of z with respect to π is defined as

$$\|z\|_\pi = \left(\sum_{i=1}^{r} z(i)^2\pi(i)\right)^{1/2}.$$

We shall also use $\ell^2(\frac{1}{\pi})$, the real vector space $\mathbb{R}^r$ endowed with the scalar product

$$\langle z, y\rangle_{\frac{1}{\pi}} = \sum_{i=1}^{r} z(i)y(i)\frac{1}{\pi(i)}$$

and its corresponding norm. For any two strictly positive probability vectors μ and π, the following relationship holds:

$$\|\mu - \pi\|_{\frac{1}{\pi}} = \left\|\frac{\mu}{\pi} - 1\right\|_\pi \geq 2\|\mu - \pi\|_{TV} \tag{3}$$

where $\|\pi - \mu\|_{TV}$ is the total variation distance

$$\|\pi - \mu\|_{TV} = \frac{1}{2}\sum_{i=1}^{r} |\pi(i) - \mu(i)|.$$

Note that the inequality in (3) is just a result of Cauchy–Schwarz inequality.

B. Mixing Times of Markov Chains

Starting from some initial distribution μ_0, the convergence to steady-state distribution is geometric with a rate equal to the *second largest eigenvalue modulus* (SLEM) of the transition matrix [23] as it is described next.

Lemma 1: Let P be an irreducible, aperiodic, and reversible transition matrix on the finite state space $\mathcal{M}$ with the stationary distribution π. Then, the eigenvalues of P are ordered in such a way that

$$\lambda_1 = 1 > \lambda_2 \geq \ldots \geq \lambda_r > -1$$

and for any initial probability distribution μ_0 on $\mathcal{M}$, and for all $n \geq 1$

$$\|\mu_0\mathbf{P}^n - \pi\|_{\frac{1}{\pi}} \leq \sigma^n\|\mu_0 - \pi\|_{\frac{1}{\pi}} \tag{4}$$

where $\sigma = \max\{\lambda_2, |\lambda_r|\}$ is the SLEM of P.

Therefore, if we define the mixing time as

$$\tau(\epsilon) = \inf\{n : \|\mu_0\mathbf{P}^n - \pi\|_{1/\pi} \leq \epsilon\}$$

for some small $\epsilon > 0$, then a simple calculation reveals that

$$(T-1)\log(\|\mu_0 - \pi\|_{1/\pi}/\epsilon) \leq \tau(\epsilon) \leq T\log(\|\mu_0 - \pi\|_{1/\pi}/\epsilon).$$

where $T = \frac{1}{1-\sigma}$. We will see that for Q-CSMA algorithm, T is exponential in number of links or the maximum weight of the network. Therefore, T is approximately proportional to $\tau(\epsilon)$, and by abusing terminology, we will also sometimes refer to T as the mixing time.

C. *Characterization of the Eigenvalues*

Let $\beta_i = 1 - \lambda_i$, so $0 = \beta_1 < \beta_2 \cdots \leq \beta_r < 2$. For any vector $\theta \in \mathbb{R}^{|\mathcal{M}|}$, define the Dirichlet form $\mathcal{E}_\pi(\theta, \theta)$ as

$$\mathcal{E}_\pi(\theta,\theta) = \langle (I-P)\theta, \theta\rangle_\pi$$

and also the variance

$$\mathrm{Var}_\pi(\theta) = \|\theta\|_\pi^2 - \langle \theta, 1\rangle_\pi^2.$$

Lemma 2 (Raleigh Theorem [23]): Let P be an irreducible, aperiodic, and reversible transition matrix on a finite state space $\mathcal{M}$, then for $j \geq 2$

$$\beta_j = \inf_{\theta \neq 0} \left\{ \frac{\mathcal{E}_\pi(\theta,\theta)}{\mathrm{Var}_\pi(\theta)} : \langle \theta, v_i\rangle_\pi = 0 \text{ for } 1 \leq i \leq j-1 \right\}$$

where v_is are the right eigenvectors of P. Moreover, any vector θ achieving the infimum is an eigenvector of P corresponding to the eigenvalue $\lambda_j = 1 - \beta_j$.

Expanding the inner product, and using reversibility of the Markov chain, reveals that

$$\mathcal{E}_\pi(\theta,\theta) = \frac{1}{2} \sum_{i,j \in \mathcal{M}} \pi(i) p_{ij} (\theta_j - \theta_i)^2.$$

To characterize the SLEM σ, we need to find λ_2 and λ_r. When solving the minimization in Lemma 2 is difficult, one can still use the result of the geometric convergence rate, Lemma 1, by finding good bounds on λ_2 and λ_r. In these cases, the following lemmas are useful [23], [24]. First, for a nonempty set $B \subset \mathcal{M}$, define the following:

$$\pi(B) = \sum_{i \in B} \pi(i)$$

and

$$F(B) = \sum_{i \in B, j \in B^c} \pi(i) p_{ij}.$$

Then, the *conductance* of an irreducible, aperiodic, and reversible transition matrix P is defined as

$$\phi(P) = \inf_{B: \pi(B) \leq 1/2} \frac{F(B)}{\pi(B)}.$$

Lemma 3 (Cheeger's Inequality):

$$1 - 2\phi(P) \leq \lambda_2 \leq 1 - \frac{\phi^2(P)}{2}.$$

Lemma 4 (Gershgorin's Bound): Let $P = [p_{ij}]$ be a finite $r \times r$ matrix. Then, for any eigenvalue λ and all $k \in [1, r]$,

$$|\lambda - p_{kk}| \leq \min(r_k, s_k)$$

where $r_k = \sum_{j \neq k} |p_{kj}|$ and $s_k = \sum_{j \neq k} |p_{jk}|$.

IV. Main Results for Collocated Networks

In prior works, the access probabilities are chosen to be some constant numbers strictly between zero and one to guarantee the irreducibility of the CSMA Markov chain. In this section, we will see that access probabilities affect the mixing time of the CSMA chain that, in turn, controls the delay of the scheduling algorithm. We aim to optimize the access probabilities in order to minimize the mixing time of the chain. Recall that the mixing time is proportional to $\frac{1}{1-\sigma}$ where σ is the SLEM of the transition probability matrix. Therefore, we need to choose access probabilities to maximize $1 - \sigma$.

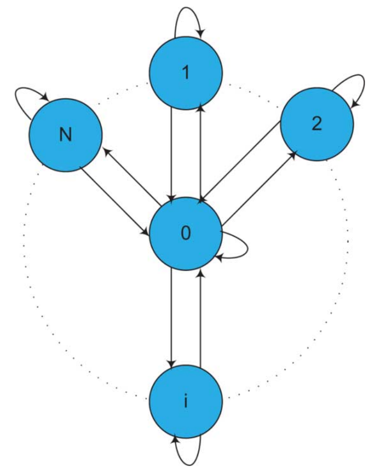

Fig. 1. Q-CSMA Markov chain for a collocated network with N links.

Consider a collocated network under Q-CSMA where every link interferes with all the other links, i.e., the conflict graph is complete. In this case, we can index the feasible schedules by $0, 1, 2, \ldots, N$, as in Fig. 1, where 0 shows the empty schedule and nonzero indices show the active link number. Every link i, $1 \leq i \leq N$, can change its state, i.e., becomes active or silent, if and only if it is selected in the decision schedule. Let X be the Q-CSMA Markov chain as in Fig. 1 and p_{ij} denote the probability of transition from state i to state j, i.e.,

$$p_{ij} = \mathbb{P}(X(t+1) = j \,|\, X(t) = i).$$

Link i is selected in the decision schedule, when it sends an INTENT message and nobody else transmits INTENT messages, which happens with probability

$$\alpha_i = a_i \prod_{\substack{j=1 \\ j \neq i}}^{N} (1 - a_j)$$

where a_i is the access probability of link i, $1 \leq i \leq N$, as defined earlier. Therefore, it follows that the transition probabilities of the Q-CSMA Markov chain are given by

$$\begin{aligned}
p_{0i} &= a_i p_i \prod_{j \neq i} (1 - a_j), \qquad i \neq 0 \\
p_{ii} &= 1 - a_i \bar{p}_i \prod_{j \neq i} (1 - a_j), \qquad i \neq 0 \\
p_{i0} &= a_i \bar{p}_i \prod_{j \neq i} (1 - a_j), \qquad i \neq 0 \\
p_{00} &= 1 - \sum_{i=1}^{N} p_{0i}
\end{aligned}$$

where $\bar{p}_i = 1 - p_i$.

Calculating λ_2 and λ_r ($r = |\mathcal{M}|$) directly from the transition probability matrix, especially when N is large and weights are different, is not an easy task. Instead, we use the Raleigh Lemma (Lemma 2) to calculate λ_2. Solving the exact minimization in Lemma 2 is possible, but it does not yield a closed-form expression for $\beta_2 = 1 - \lambda_2$ (see Appendix E; β_2 is expressed as a zero of a complex polynomial). Hence, we do not present the exact solution here, and instead present the following more useful result about the upper and lower bounds on λ_2. The proof is provided in Appendix A.

Lemma 5: For a collocated network of $N \geq 2$ links, and given a set of access probabilities $\{a_i\}_{i=1}^N$ and a set of transmission probabilities $\{p_i\}_{i=1}^N$, $\beta_2^{\text{low}} \leq \beta_2 \leq \beta_2^{\text{up}}$, where

$$\beta_2^{\text{low}} = \min_{1 \leq i \leq N} \bar{p}_i a_i \prod_{j \neq i} (1 - a_j) \tag{5}$$

$$\beta_2^{\text{up}} = 2 \min_{i: \pi_i \leq 1/2} \bar{p}_i a_i \prod_{j \neq i} (1 - a_j). \tag{6}$$

Note that in the case of $N = 1$, trivially, no scheduling is needed and $a_1^* = 1$. Thus, we can assume that there are at least two links in the network. Next, we use the Gershgorin's bound (Lemma 4) to find a lower bound on all the eigenvalues. We state the result as a lemma, whose proof is given in Appendix B.

Lemma 6: For a collocated network of $N \geq 2$ links, under equal access probabilities and any set of transmission probabilities, all the eigenvalues are nonnegative, i.e., $\lambda_r \geq 0$.

Note that for general access probabilities, $T = \frac{1}{1-\sigma} \geq \frac{1}{\beta_2}$. However, in the case of equal access probabilities, by Lemma 6, SLEM is dominated by λ_2 and, hence, $T = 1/\beta_2$.

We will use the following result in bounding the smallest possible mixing time.

Lemma 7: The optimal access probabilities that maximize β_2^{low}, in Lemma 5, are in the form of $a_i^* = \frac{k}{k+\bar{p}_i}$, where the constant k is chosen such that $\sum_{i=1}^N a_i^* = 1$.

The proof for Lemma 7 is provided in Appendix C. As a special case, when all the p_i's are equal, i.e., weights are equal, simple calculation reveals that the optimal access probabilities in Lemma 7 are all equal to $1/N$. Therefore, for such a choice of access probabilities, the equality $T = 1/\beta_2$ holds and therefore, $T \leq 1/\beta_2^{\text{low}}$. Hence, in the case of equal weights, the access probabilities of the traditional *ALOHA* protocol, i.e., $a_i = \frac{1}{N}$, minimize the upper bound $1/\beta_2^{\text{low}}$.

In general, the ALOHA access probabilities are not optimal for the queue-based random access protocols, and finding the optimal access probabilities requires the knowledge of all the weights in the network, which might not be feasible in practice. In this case, one might be interested in a suboptimal solution that does not require the global knowledge and the *mixing time ratio*, i.e., the ratio of the optimal solution to the suboptimal solution, remains bounded, i.e.,

$$1 < \frac{T(\text{subopt.})}{T(\text{opt.})} < M \tag{7}$$

for some constant M *independent of the network size* N. It suffices to find a suboptimal solution such that

$$1 < \frac{T^{\text{up}}(\text{subopt.})}{T^{\text{low}}(\text{opt.})} < M \tag{8}$$

where $T^{\text{up}}(\text{subopt.})$ is the upper bound on the suboptimal solution and $T^{\text{low}}(\text{opt.})$ is the lower bound on the optimal solution. Equivalently, for a suboptimal solution with equal access probabilities, we need to show that

$$1 < \frac{\beta_2^{\text{up}}(\text{opt.})}{\beta_2^{\text{low}}(\text{subopt.})} < M \tag{9}$$

where β_2^{up} and β_2^{low} were defined in Lemma 5. To show such a property, we need to consider an appropriate distribution of $\bar{p}_i$'s as the number of nodes N grows. Here, we assume that there exist m types of weights, such that a constant fraction α_k of the nodes have the weight $\bar{p}_k$ for $k = 1, \ldots, m$. Note that in such a setting, if there exists a state/link l with $\pi_l > 1/2$, then since p_l is one of the m possible weights, there must exist $\alpha_k N$ links with the same transmission probability p_l, and all of them should have stationary probabilities greater than 1/2 which is impossible since $\sum_{i=0}^N \pi_i = 1$. Therefore, all the states have the stationary probability less than 1/2, and $\beta_2^{\text{up}} = 2\beta_2^{\text{low}}$. Hence, the access probabilities that maximize β_2^{up}, i.e., yield the smallest lower bound on the mixing time, are given by Lemma 7. Then, it is easy to see that the optimal k^* in Lemma 7 is in the form of $\frac{c}{N}$ for some constant $\bar{p}^{\min} < c < 1/2$, where $\bar{p}^{\min} = \min_i \bar{p}_i$. Thus, we have

$$\beta_2^{\text{up}}(\text{opt.}) = 2k^* \prod_i \frac{\bar{p}_i}{k^* + \bar{p}_i} = \frac{2c}{N} \prod_{k=1}^m \left(\frac{\bar{p}_k}{\bar{p}_k + c/N} \right)^{\alpha_k N}.$$

Putting everything together, the suboptimal access probabilities of the form $a_i = \frac{1}{N}$ yield a bounded mixing time ratio independent of N because

$$\beta_2^{\text{low}}(\text{subopt.}) = \frac{\bar{p}^{\min}}{N} \left(1 - \frac{1}{N}\right)^{N-1} \tag{10}$$

and

$$\lim_{N \to \infty} \frac{\beta_2^{\text{up}}(\text{opt.})}{\beta_2^{\text{low}}(\text{subopt.})} = \frac{2c}{\bar{p}^{\min}} \exp\left(1 - c \sum_{k=1}^m \frac{\alpha_k}{\bar{p}_k}\right) < \infty. \tag{11}$$

Therefore the mixing time ratio is bounded for all values of N. The importance of the above ratio is that it guarantees that the mixing time of the suboptimal solution is within a constant multiple of the optimal mixing time, independent of the network size. Furthermore, *choosing access probabilities independent of* N *results in unbounded mixing time ratio*. This is because, by (5), any choice of equal access probabilities a, for some $0 < a < 1$, yields $\beta_2^{\text{low}} = \bar{p}^{\min} a (1-a)^{N-1}$, and it is easy to check that $\beta_2^{\text{up}}(\text{opt.})/\beta_2^{\text{low}}$ grows unboundedly if a does not scale as $1/N$.

The same analysis is possible for other kinds of weight assignment as well. Essentially, since there exists at most one link l with $\pi_l > 1/2$, we can prove that this does not change the asymptotics. For example, for the weight assignment that there are N different weights, following the exact line of arguments as above shows that

$$\beta^{\text{up}}(\text{opt.}) \leq \frac{2c}{N-1} \prod_{i \neq l} \left(1 - \frac{1}{1 + \frac{\bar{p}_i}{c}(N-1)}\right)$$

where l is the link with $\bar{p}_l = \bar{p}^{\min}$ and c is a constant satisfying $\bar{p}^{\min} < c < 1/2$. Hence, as $N \to \infty$, $\beta^{\text{up}}(\text{opt.}) \leq \frac{2c}{N-1} \exp(-1/2c)$ and (10) still holds. Therefore, the mixing time ratio is bounded by $\frac{2c}{\bar{p}^{\min}} \exp(1 - \frac{1}{2c})$.

V. Main Results for General Networks

The extension of results to general networks is more difficult since the corresponding CSMA Markov chain is much more complex than the Markov chain of collocated networks, hence finding the second largest eigenvalue by solving the optimization in Lemma 2 is cumbersome. Instead, we find an upper bound on the SLEM based on the conductance bound (Lemma 3).

Assume the current schedule is $X(t) = X$, for some $X \in \mathcal{M}$, and the CSMA Markov chain makes a transition to the next state/schedule $X(t+1) = Y$. Note that $X \backslash Y = \{l : x_l = 1, y_l = 0\}$ is the set of links that change their states from 1 (active) to 0 (silent). Similarly $Y \backslash X = \{l : x_l = 0, y_l = 1\}$ is the set of links that change their states from 0 to 1. From the scheduling algorithm, it is clear that a link can change its state only when it belongs to the decision schedule. Therefore, X can make a transition to Y when $X \Delta Y \subseteq m$, for some $m \in \mathcal{M}$, where $X \Delta Y := (X \backslash Y) \cup (Y \backslash X)$. Now, given that such a decision schedule m has been chosen, with some probability $\alpha(m)$, we can divide the links of m into five cases.

1) $l \in X \backslash Y$: Link l decides to change its state from 1 to 0; this occurs with probability $\bar{p}_l$.
2) $k \in Y \backslash X$: Link k decides to change its state from 0 to 1; this occurs with probability p_k.
3) $i \in m \cap (X \cap Y)$: Link i decides to keep its state 1; this occurs with probability p_i.
4) $e \in m \backslash C(X)$ where $C(X) = \cup_{l \in X} C(l)$: Link e has to keep its state 0; this occurs with probability 1.
5) $j \in m \backslash (X \cup Y) \backslash C(X)$: Link j decides to keep its state 0; this occurs with probability $\bar{p}_j$.

Note that $m \backslash C(Y \backslash X) = \emptyset$ because $Y \backslash X \subseteq m$, so we have $m \backslash (X \cup Y) \backslash C(X) = m \backslash (X \cup Y) \backslash C(X \cup Y)$. Then, it is not hard to argue that $P(X,Y)$, the probability of transition from the schedule X to the schedule Y, is given by

$$P(X,Y) = \sum_{m \in \mathcal{M}: X \Delta Y \subseteq m} \alpha(m) \prod_{l \in X \backslash Y} \bar{p}_l \prod_{k \in Y \backslash X} p_k \times \prod_{i \in m \cap (X \cap Y)} p_i \prod_{j \in m \backslash (X \cup Y) \backslash C(X \cup Y)} \bar{p}_j. \quad (12)$$

Recall the mechanism for generating a decision schedule m by transmitting the INTENT messages based on access probabilities $\{a_i\}_{i=1}^N$. All the links that are included in m should have transmitted INTENT messages and have not heard any INTENT messages from their interfering neighbors. Hence, the probability of generating a decision schedule $\alpha(m) = \alpha(m; G)$ in the graph G can be characterized by

$$\alpha(m, G) = \prod_{i \in m} a_i \prod_{j \in C(m)} (1 - a_j) \alpha(\emptyset; G \backslash (m \cup C(m)))$$

where $\alpha(\emptyset; G \backslash (m \cup C(m)))$ is the probability that no nodes are included in the decision schedule in graph $G \backslash (m \cup C(m))$, i.e., the graph obtained by removing all the links in m and $C(m)$ from G. The expression for $\alpha(\emptyset; G')$ could be quite complicated since it has to account for all the events that yield a $\emptyset$ schedule due to either not transmitting INTENT messages or collision between INTENT messages transmitted by the nodes of G'. Nevertheless, the mixing time can be upper-bounded by[3]

$$T = \frac{1}{\beta_2} \leq \frac{2}{\tilde{\phi}^2(P)} \quad (13)$$

where $\tilde{\phi}(P)$ is an approximate conductance defined in the following lemma. The proof is presented in Appendix D.

Lemma 8: In a general network, under the Q-CSMA with transition probability matrix P, the conductance $\phi(P)$ is lower-bounded by $\tilde{\phi}(P)$, where

$$\tilde{\phi}(P) = \min_{m_0 \in \mathcal{M}_0} P(m_0, \emptyset),$$

$\mathcal{M}_0 \subset \mathcal{M}$ is the set of all maximal schedules, and $P(m_0, \emptyset)$ is the probability of transition from the maximal schedule m_0 to the empty schedule $\emptyset$.

Therefore, we can try to find optimal access probabilities that maximize $\tilde{\phi}(P)$. In this case, the optimal access probabilities are the solution to

$$\max_{\{a_i\}} \min_{m_0 \in \mathcal{M}_0} \prod_{i \in m_0} a_i \bar{p}_i \prod_{j \notin m_0} (1 - a_j). \quad (14)$$

Solving the above optimization needs some global knowledge of the network. Hence, we investigate possible suboptimal solutions with the bounded mixing-time ratio (7) when we use the upper bounds on the mixing times, based on (13), instead of the exact values.

As a special case, consider a d-regular network with N links, i.e., each link has exactly d interfering neighbors. Furthermore, assume that the weights are equal, i.e., $\bar{p}_1 = \ldots = \bar{p}_N$. It is easy to show that, in this case, in the optimization (14), we need to consider the minimization over the maximal schedules with the maximum size, i.e., over the set of nodes with the same color in a valid node coloring of the graph. Let χ denote the chromatic number of the corresponding graph. Note that since there is no unique way of constructing a d-regular graph with N nodes, the chromatic number depends on the construction, but we know that

$$\text{maximum clique size } \leq \chi \leq d + 1.$$

Since the graph is symmetric, all the access probabilities must be equal and the maximum size of the maximal schedule is $s = \lceil \frac{N}{\chi} \rceil$. Then, the optimal access probabilities in (14) are all equal and simply the solution to

$$\max_a a^s (1-a)^{N-s}$$

i.e., $a = \frac{s}{N} = \frac{1}{N} \lceil \frac{N}{\chi} \rceil$ or

$$\frac{1}{\chi} \leq a \leq \frac{1}{\chi} + \frac{1}{N}. \quad (15)$$

This suggests using $1/\chi$ as the access probability. Since, in general, the chromatic number of the network might not be known,

[3]Here, we assume that the SLEM is λ_2. To ensure this, one may have to modify the CSMA Markov chain slightly to make it a *lazy chain* [25] by adding self-loops of probability at least 1/2 to each state. This will not change the steady-state distribution but changes the transition probability matrix to $P' = \frac{1}{2}(P + I)$, and hence, $\lambda'_i = \frac{1}{2}(\lambda_i + 1)$. This shows that all the eigenvalues of P' are nonnegative and the lazy chain is at most twice slower than the original CSMA chain.

our conjecture is that $\frac{1}{d+1}$ is a good candidate for the access probabilities when each node only knows the number of its interfering neighbors. We validate this conjecture through simulations later.

Next, consider a more general case of a d-regular network with different weights. Although $1/\chi$ or $1/(d+1)$ are not the optimal access probabilities, we argue that they yield a bounded gap between the upper bound (13) on the mixing time of the optimal access probabilities and the corresponding upper bound on the suboptimal solution. To prove such a property, similar to the collocated network, we need to consider an appropriate scaling of the network and a weight assignment as we add more nodes to the network. For the assignment of transmission probabilities/weights, we consider the worst assignment that is possible for the suboptimal solution: Consider transmission probabilities $\bar{p}_1 \leq \bar{p}_2 \ldots \leq \bar{p}_\chi$, and then assign $\bar{p}_i$ to all the links in the ith maximal schedule, for $i = 1, \ldots, \chi$. It is clear that the following optimization gives an upper bound on (14):

$$\tilde{\phi}^{\rm up}(P) = \max_{\{a_i\}} \min_{1\leq i\leq\chi} \bar{p}_i^s a_i^s \prod_{j\neq i}(1-a_j)^s \tag{16}$$

where s is the maximum size of a maximal schedule. The rest of calculations follows in parallel with those of the complete graph. The optimal access probabilities, maximizers of (16), are given by

$$a_i^* = \frac{k}{k+\bar{p}_i} \qquad \sum_{i=1}^{\chi} a_i^* = 1. \tag{17}$$

Next, we prove that the suboptimal solution has a bounded mixing time property, i.e.,

$$\frac{T^{\rm up}(\text{subopt.})}{T^{\rm up}(\text{opt.})} = \left(\frac{\tilde{\phi}(\text{opt.})}{\tilde{\phi}(\text{subopt.})}\right)^2 \leq \left(\frac{\tilde{\phi}^{\rm up}(\text{opt.})}{\tilde{\phi}(\text{subopt.})}\right)^2 < \infty.$$

To show such a property, we need to consider an appropriate scaling of the network as the number of nodes N grows. We assume that the degree d grows uniformly for all the nodes in G as N increases, i.e., the number of interfering neighbors of each node increases uniformly. Therefore, the chromatic number grows linearly in N ($\chi = \chi_N$), and the maximum size of a schedule remains constant s. Moreover, there are m types of weights such that α_k fraction of maximal schedules have the transmission probability p_k for $k = 1, \ldots, m$. Noting that constant k is in the form of $k^* = \frac{c}{\chi}$ for some $\bar{p}^{min} \leq c \leq 1/2$, the optimal $\tilde{\phi}^{\rm up}$ is given by

$$\begin{aligned}\tilde{\phi}^{\rm up}(\text{opt.}) &= \left(k^*\prod_{i=1}^{N}\frac{\bar{p}_i}{k^*+\bar{p}_i}\right)^s\\ &= \left(\frac{c}{\chi_N}\prod_{k=1}^{m}\left(\frac{\bar{p}_k}{\bar{p}_k+c/\chi_N}\right)^{\alpha_k\chi_N}\right)^s\\ &\simeq \left(\frac{c}{\chi_N}\right)^s \exp\left(-cs\sum_{k=1}^{m}\frac{\alpha_k}{\bar{p}_k}\right)\end{aligned}$$

where "$\simeq$" shows the asymptotic as $N \to \infty$. The suboptimal $\tilde{\phi}$ is

$$\tilde{\phi}(\text{subopt.}) = \left(\bar{p}^{min}\frac{s}{N}\left(1-\frac{s}{N}\right)^{N/s-1}\right)^s \simeq \left(\bar{p}^{min}\frac{s}{eN}\right)^s$$

and hence

$$\lim_{N\to\infty}\frac{\tilde{\phi}^{\rm up}(\text{opt.})}{\tilde{\phi}(\text{subopt.})} = \left(\frac{ce}{\bar{p}^{min}}\right)^s \exp\left(-cs\sum_{k=1}^{m}\frac{\alpha_k}{\bar{p}_k}\right). \tag{18}$$

The importance of the above ratio is that it guarantees that the gap between the upper bounds on the mixing times of the suboptimal solution and the optimal solution is a constant independent of the network size.

Similarly, using $1/(d_N+1)$ as the suboptimal access probability still yields a bounded ratio, but the corresponding ratio will be

$$\begin{aligned}\lim_{N\to\infty}\frac{\tilde{\phi}^{\rm up}(\text{opt.})}{\tilde{\phi}(\text{subopt.})} &= \left(\frac{c}{\bar{p}^{min}}\frac{d_N+1}{\chi_N}e^{\frac{\chi_N}{d_N+1}}\right)^s\\ &\quad\times\exp\left(-cs\sum_{k=1}^{m}\frac{\alpha_k}{\bar{p}_k}\right)\end{aligned} \tag{19}$$

which is greater than (18) (since $\frac{d_N+1}{\chi_N} \geq 1$ and $xe^{1/x} \geq e$ for all $x \geq 1$.).

VI. Simulation Results

In this section, we evaluate the performance of different access probabilities via simulations. For this purpose, we have considered different topologies for the wireless network. In the algorithm, we have selected the transmission probabilities at time t as $p_l(t) = \frac{e^{w_l(t)}}{1+e^{w_l(t)}}$ where $w_l(t) = \log(q_l(t)+1)$, $1 \leq l \leq N$. This choice makes the network stable [18] and yields the best delay performance.

The first example is a collocated network of $N = 8$ links. To choose the arrival rates, choose a point $v = (v_1 \cdots v_8)$ on the boundary of the capacity region which satisfies $\sum_{i=1}^{8} v_i < 1$ and $v_i \geq 0$, and consider the arrival rates of the form $\lambda = \rho v$ for a load $0 < \rho < 1$. Note that, as $\rho \to 1$, λ approaches a point on the boundary of the capacity region. For example, we have chosen $v_1 = \cdots = v_4 = 3/16$, $v_5 = \cdots = v_8 = 1/16$ and $\rho = 0.8$. The queue length behavior (averaged over the links) for two access probabilities $\frac{1}{N}$ and $\frac{3}{N}$ have been depicted in Fig. 2. Fig. 3 shows the average queue length (averaged over time and over the links) for different values of the access probability a (all the links have the same access probability). The simulation time is set to 10^6 time-slots since, as it is seen from Fig. 2, the queues show a steady-state behavior in this duration. For each access probability, we have repeated the experiment five times and averaged over the experiments. It can be seen that $a = 1/N(= 1/8)$ yields the smallest average queue size.

The second example is a 5×5 grid conflict graph consisting of $N = 25$ links in Fig. 4. Consider the point $v = (1/2, \ldots, 1/2)$ on the boundary of the capacity region and the the load $\rho = 0.7$ so the arrival rates are all equal to 0.35. Note that v is a combination of two maximal schedules: one consisting of all the links with even indices and one consisting of all the odd links. Fig. 5 shows the performance of different access probabilities, where we choose equal access probabilities for all the links. Again, $a_i = 1/(4+1) = 1/5$, $i = 1, \ldots 25$, gives the smallest average queue length. Moreover, its average queue size was almost identical to the one obtained by using $a_i = 1/(d_i+1)$ as the access probabilities.

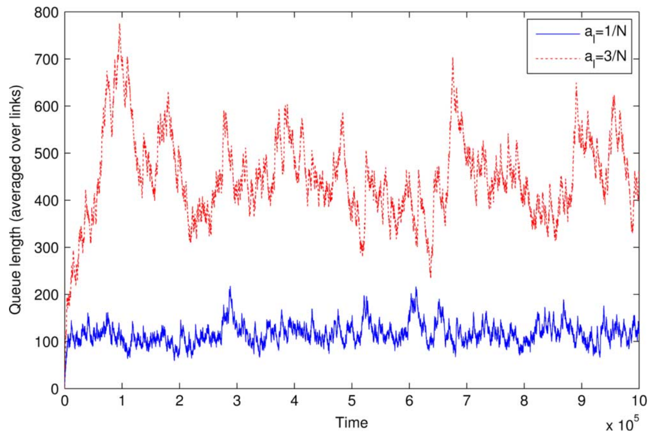

Fig. 2. Queue behavior for two access probabilities, $1/N$ and $3/N$, in a collocated network of $N = 8$ links.

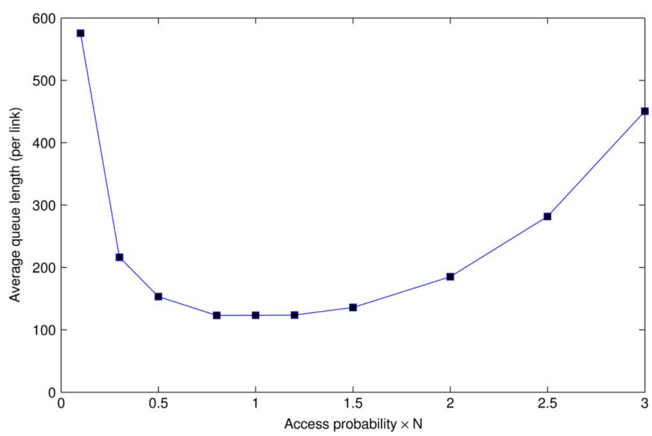

Fig. 3. Average queue size versus access probability in a collocated network with $N = 8$ links.

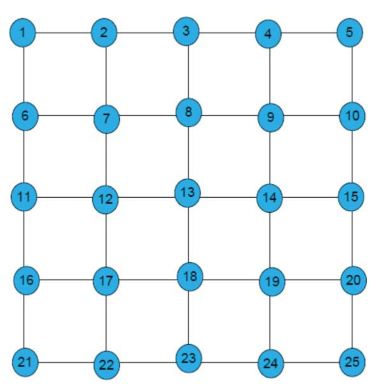

Fig. 4. Conflict graph where links consist a 5×5 grid. Links are indexed from 1 to 25.

Next, consider the grid network of Fig. 6. Note that this is the actual network not its conflict graph. The network has 16 nodes and 24 links. We consider a one-hop interference constraint, i.e., two links interfere if they are adjacent (share a node in the network). Consider the following maximal schedules:

$$M_1 = \{1, 3, 8, 10, 15, 17, 22, 24\}$$
$$M_2 = \{4, 5, 6, 7, 18, 19, 20, 21\}$$
$$M_3 = \{1, 3, 9, 11, 14, 16, 22, 24\}$$
$$M_4 = \{2, 4, 7, 12, 13, 18, 21, 23\}.$$

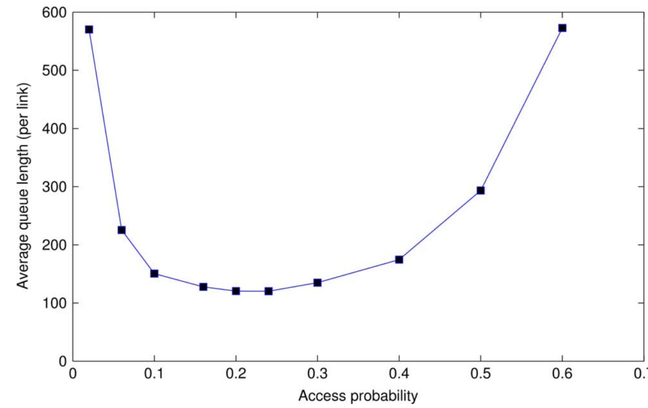

Fig. 5. Average queue size versus access probability for the network of Fig. 4.

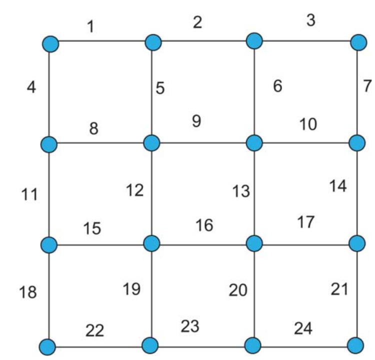

Fig. 6. Grid network consisting of 16 nodes and 24 links.

With a little abuse of notation, let M_i also be a vector whose ith element is 1 if $i \in M_i$, and 0 otherwise. We consider arrival rates that are a convex combination of the above maximal schedules scaled by $\rho = 0.8$, e.g.,

$$\lambda = \rho \sum_{i=1}^{4} c_i M_i, \qquad c = [0.2, 0.3, 0.2, 0.3].$$

Our conjecture is that access probabilities of the form $\{1/(d_i + 1)\}_{i=1}^{N}$ (d_i is the number of interfering links of the link i) should result in good performance. We compare the performance of the network under this choice of access probabilities to the performance of the network under equal access probabilities, i.e., $a_i = a$, $1 \leq i \leq 24$, for some constant a between zero and one. Fig. 7 verifies our conjecture where the dashed line at the bottom of the figure is the average queue size resulted by using $1/(d_i+1)$ as the access probabilities. Next, to make the topology more asymmetric, we remove a random set S of links from the network. In the simulation shown here, $S = \{3, 4, 12, 14, 23\}$. Fig. 8 shows the resulting network. For the arrival rates, we consider the same convex combination of 4 maximal schedules M_1 to M_4 with links of the set S removed from the maximal schedules, with different loads $\rho = 0.5, 0.8$, and 0.9. For each load ρ, the average queue size for the equal access probabilities, ranging from 0.05 to 0.55, and also for probabilities of the form $1/(d_i + 1)$, the dashed line, have been depicted in Figs. 9–11. Again, we see that the choice of $a_i = 1/(d_i+1)$ performs nearly as well as the best choice of fixed access probabilities. However, it is important to note that *the choice* $1/(d_i + 1)$ *adapts itself to*

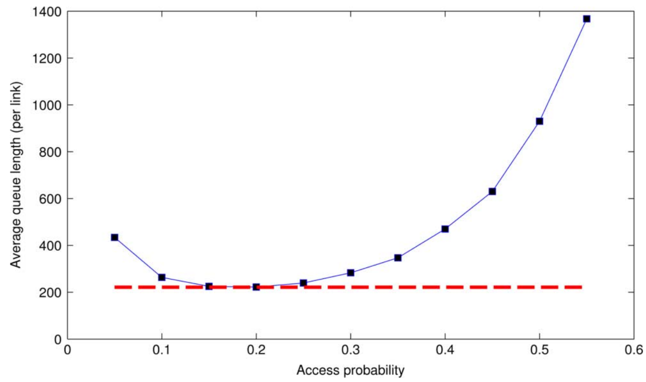

Fig. 7. Average queue size versus access probability for the grid network of Fig. 6.

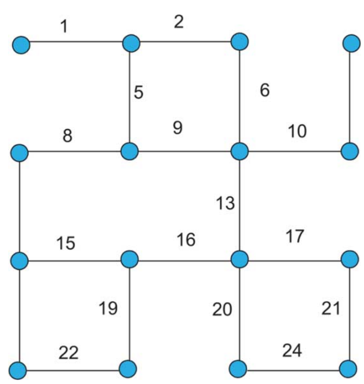

Fig. 8. Network obtained, from the grid of Fig. 6, by removing links 3, 4, 12, 14, 23.

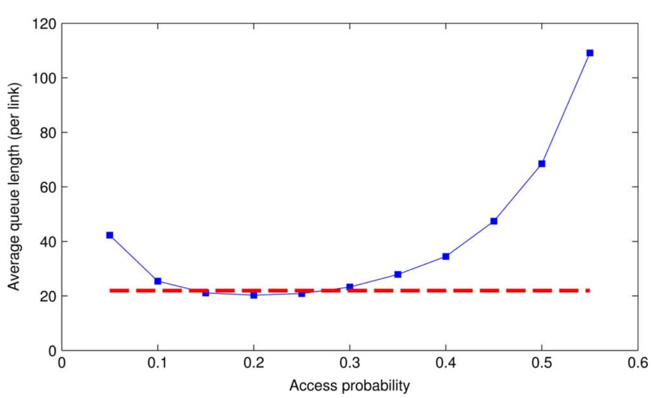

Fig. 9. Average queue size versus access probability for the network of Fig. 8, $\rho = 0.5$.

the topology of the network, whereas it is not clear how one can choose the best fixed access probabilities a priori for a network.

VII. Concluding Remarks

Access probabilities affect the mixing time of the CSMA Markov chain, which, in turn, has a significant impact on the delay performance of the algorithm. It turns out that characterizing the optimal mixing time, as a function of access probabilities, in general, is a formidable task. Even if we are able to characterize the optimal mixing time and find the optimal access probabilities, they will depend on global knowledge of the network and, thus, will not be suitable for the distributed operation of the CSMA algorithm. Instead, we have shown that access probabilities of the form $1/(d_l+1)$, where d_l is the number of interfering neighbors of link l, can yield mixing times that are within a constant factor of the optimal mixing time, independent of the number of nodes in the network. This was proved for fully connected networks and d-regular networks. We conjecture that, in general topologies, such access probabilities should have good delay performance. This conjecture is verified through extensive simulations, some of which are shown in the paper. It would be interesting to prove such a conjecture for general networks as future work.

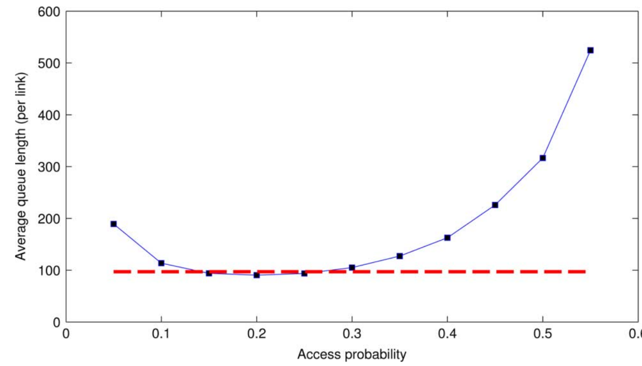

Fig. 10. Average queue size versus access probability for the network of Fig. 8, $\rho = 0.8$.

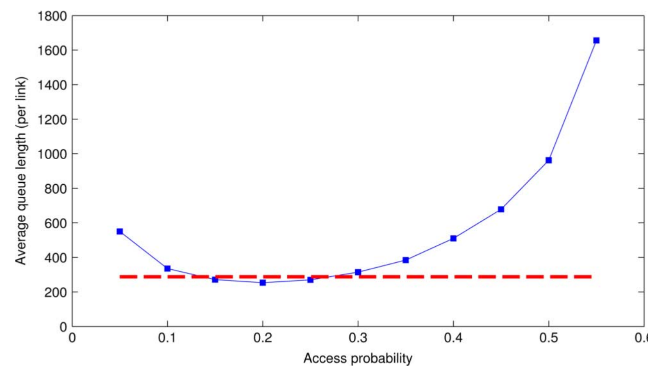

Fig. 11. Average queue size versus access probability for the network of Fig. 8, $\rho = 0.9$.

Appendix A
Proof of Lemma 5

Note that $v_1 = 1$, because $\pi P = 1\pi$, and $\mathcal{E}_\pi(\theta,\theta) = \mathcal{E}_\pi(\theta - c\mathbf{1}, \theta - c\mathbf{1})$, and $\mathrm{Var}_\pi(\theta) = \mathrm{Var}_\pi(\theta - c\mathbf{1})$ for any constant vector $c\mathbf{1}$. Therefore, without loss of optimality, we can only consider zero mean vectors θ. Then, it is easy to show that the minimization in Lemma 2 can be written as

$$\beta_2 = \frac{1}{2}\inf_{\theta} \sum_{i,j=0}^{N} \pi(i) p_{ij} (\theta_j - \theta_i)^2 \tag{20}$$

subject to the constraints

$$\theta \neq 0 \quad \sum_{i=0}^{N} \pi(i)\theta_i = 0 \quad \sum_{i=0}^{N} \pi(i)\theta_i^2 = 1. \tag{21}$$

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Note that, for the complete graph

$$\pi(i) = \frac{1}{Z}\frac{p_i}{\bar{p}_i} = \frac{1}{Z}\exp(w_i)$$

for $i = 0, 1, \ldots, N$, where we have defined $p_0 = 1/2 = \bar{p}_0$. Hence, the optimization can be written as

$$\beta_2 = \frac{1}{2Z}\inf_{\theta}\left\{\sum_{j=0}^{N} p_{0j}(\theta_j - \theta_0)^2 + \sum_{i=1}^{N}\frac{p_i}{\bar{p}_i}p_{i0}(\theta_0 - \theta_i)^2\right\}$$

subject to the constraints in (21). Using the reversibility of the Markov chain, we have $\frac{p_i}{\bar{p}_i}p_{i0} = p_{0i}$, and therefore

$$\beta_2 = \frac{1}{Z}\inf_{\theta}\sum_{i=1}^{N} p_{0i}(\theta_i - \theta_0)^2 \tag{22}$$

subject to the constraints in (21).

Although solving the above optimization is possible, it does not yield a closed-form expression for β_2 (see Appendix E). Instead, we try to find upper and lower bounds on β_2. By a change of variable $y_i = \theta_i - \theta_0$, $i = 1, \ldots, N$, the optimization problem simplifies to

$$\beta_2 = \frac{1}{Z}\inf_{y,\theta_0}\sum_{i=1}^{N} y_i^2 a_i p_i \prod_{j\neq i}^{N}(1 - a_j) \tag{23}$$

subject to

$$\sum_{i=1}^{N}\frac{p_i}{\bar{p}_i}y_i^2 = Z(1+\theta_0)^2 \quad \sum_{i=1}^{N}\frac{p_i}{\bar{p}_i}y_i = -\theta_0 Z. \tag{24}$$

To get a lower bound on β_2, we ignore the last constraint, i.e., the lower bound is the solution to

$$\beta_2^{\text{low}} = \frac{1}{Z}\inf_{u}\sum_{i=1}^{N} u_i^2 \bar{p}_i a_i \prod_{j\neq i}^{N}(1 - a_j) \tag{25}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} u_i^2 = Z(1+\theta_0)^2. \tag{26}$$

where we have used a change of variable $u_i = \sqrt{\frac{p_i}{\bar{p}_i}}y_i$. Then, it is clear that $\theta_0^* = 0$, and the optimal value, which is a lower bound on β_2, is given by (5) for any set of access probabilities.

Next, we prove the upper bound. By Lemma 3, $\beta_2 \leq 2\phi(P)$, so it remains to calculate the conductance

$$\phi(P) = \inf_{B:\pi(B)\leq 1/2}\frac{\sum_{i\in B, j\in B^c}\pi(i)P_{ij}}{\sum_{i\in B}\pi(i)}.$$

Consider the set B to only contain singletons. Then, the optimal B does not contain 0 since $\pi(0) \leq \pi(i)$ for any $i \neq 0$, and, by the reversibility, $\pi(i)p_{i0} = \pi(0)p_{0i}$ for any $i \neq 0$. Thus, considering only the singletons, we have

$$\phi(P) \leq \min\{p_{i0} : \pi(i) \leq 1/2, i \neq 0\} \tag{27}$$

$$\leq \min_{1\leq i\leq N}\left\{\bar{p}_i a_i \prod_{j\neq i}(1 - a_j) : \pi(i) \leq 1/2\right\}. \tag{28}$$

Equation (28) is obviously an upper bound on the actual conductance. Nevertheless, it yields the statement of lemma. Note that if there is only one link in the network, the set (27) is empty.

APPENDIX B
PROOF OF LEMMA 6

By Gershgorin's bound, Lemma 4, $\lambda_r \geq -1 + 2\min_i p_{ii}$, where we have used the fact that $\sum_{j\neq i} p_{ij} = 1 - p_{ii}$. Note that since $\bar{p}_i \leq 1/2$, $p_{ii} \geq 1/2$ for $i \neq 0$. Furthermore

$$\begin{aligned} p_{00} &= 1 - \sum_{i=1}^{N} a_i p_i \prod_{j\neq i}(1 - a_i) \geq 1 - \sum_{i=1}^{N} a \prod_{j\neq i}(1 - a) \\ &\geq 1 - \sum_{i=1}^{N}\frac{1}{N}\left(1 - \frac{1}{N}\right)^{N-1} \\ &= 1 - \left(1 - \frac{1}{N}\right)^{N-1} \geq 1/2 \end{aligned}$$

where we used the assumption that there are at least two links in the network. Hence, $p_{ii} \geq 1/2$ for all $0 \leq i \leq N$, and therefore $\lambda_r \geq 0$.

APPENDIX C
PROOF OF LEMMA 7

First, we argue that, at the optimal solution, all the values $p_{i0}(a) = \bar{p}_i a_i \prod_{j\neq i}^{N}(1 - a_j)$, for $i = 1, \ldots, N$, are equal. Assume that this is not true, then let $S \subseteq \{1, 2, \ldots, N\}$ be the set of indices that are minimizers of $\min_i p_{i0}(a)$. Therefore, $p_{i0}(a) < p_{j0}(a)$ for every $i \in S$ and every $j \in S^c$. However, decreasing the values of a_j, $j \in S^c$, increases the values of $p_{i0}(a)$, $i \in S$, which is a contradiction. Hence, this means that

$$\prod_{j=1}^{N}(1 - a_j^*)\bar{p}_i\frac{a_i^*}{1 - a_i^*} = k', \qquad \text{for } i = 1, \ldots, N$$

for some constant k' independent of a. Therefore, the optimal access probabilities are in the form of

$$a_i^* = \frac{k}{k + \bar{p}_i}. \tag{29}$$

Returning to the maximization problem, the constant k must be chosen as the solution to

$$\max_k k\prod_{i=1}^{N}\frac{\bar{p}_i}{k + \bar{p}_i}.$$

Taking the $\log(\cdot)$ and finding the zeros of the derivative yields

$$\frac{1}{k^*} - \sum_{i=1}^{N}\frac{1}{k^* + \bar{p}_i} = 0.$$

Putting everything together, the optimal access probabilities must satisfy

$$a_i^* = \frac{k}{k + \bar{p}_i} \qquad \sum_{i=1}^{N} a_i^* = 1.$$

APPENDIX D
PROOF OF LEMMA 8

Let m_0 be a maximal schedule such that $X\Delta Y \subseteq m_0$, then it is clear that

$$\begin{aligned}P(X,Y) &\geq \alpha(m_0)\prod_{l\in X\setminus Y}\bar{p}_l\prod_{k\in Y\setminus X}p_k\\ &\quad\times\prod_{i\in m_0\cap(X\cap Y)}p_i\prod_{j\in m_0\setminus(X\cup Y)\setminus C(X,Y)}\bar{p}_j\\ &\geq\prod_{i\in m_0}a_i\prod_{j\notin m_0}(1-a_j)\prod_{l\in m_0}\bar{p}_l = P(m_0,\emptyset)\end{aligned}$$

where we have used the fact $p_i \geq \bar{p}_i$, for all i, due to (2). The conductance can be lower-bounded by

$$\phi(P) = \inf_{B:\pi(B)\leq 1/2}\frac{F(B)}{\pi(B)} \geq \inf_{B\neq\emptyset}\frac{F(B)}{\pi(B)}$$

since if $B = \emptyset$, we can replace B with B^c and get a smaller conductance because then $\pi(B^c)\geq\pi(B)$ and $F(B) = F(B^c)$ by reversibility. Note that there is a transition between X and Y whenever $X\Delta Y$ is a valid schedule, therefore a direct transition from the empty schedule $\emptyset$ to any schedule X, and vice versa, is possible. Hence, the conductance can be further lower-bounded as follows:

$$\begin{aligned}\phi(P) &\geq \inf_{B\neq\emptyset}\frac{\sum_{X\in B,Y\in B^c}\pi(X)P(X,Y)}{\pi(B)}\\ &\geq\inf_{B\neq\emptyset}\frac{\sum_{X\in B}\pi(X)P(X,\emptyset)}{\sum_{X\in B}\pi(X)}\\ &=\min_{X\neq\emptyset}P(X,\emptyset).\end{aligned}$$

Hence

$$\phi(P)\geq\min_{m_0\in\mathcal{M}_0}P(m_0,\emptyset) = \tilde{\phi}(P) \tag{30}$$

where $\mathcal{M}_0$ denotes the set of all maximal schedules in $\mathcal{M}$.

APPENDIX E
EXACT β_2 FOR COLLOCATED NETWORKS

As we saw from the proof of Lemma 5 in Appendix A, in the case of collocated networks, the exact β_2 is the optimal value of the following optimization:

$$\frac{1}{Z}\inf_\theta\sum_{i=1}^N p_{0i}(\theta_i-\theta_0)^2, \qquad \text{s.t. } \mathrm{Var}_\pi(\theta) = 1$$

where $\theta\in\mathbb{R}^{N+1}$. Here, we show that β_2 can be expressed as a zero of a polynomial. First, since shift by a constant does not change the variance, the above optimization is equivalent to

$$\frac{1}{Z}\inf_y\sum_{i=1}^N p_{0i}y_i^2, \qquad \text{s.t. } \mathrm{Var}_\pi(y) = 1.$$

Note that since $y_0 = 0$

$$\mathrm{Var}_\pi(y) = \left(\sum_{i=1}^N\pi_i y_i^2\right) - \left(\sum_{i=1}^N\pi_i y_i\right)^2$$

so we can consider the minimization over $y = (y_1\cdots y_N)^T\in\mathbb{R}^N$. It is not hard to show that $\mathrm{Var}_\pi(y) = y^TAy$, where $A_{N\times N}$ is a symmetric positive definite matrix such that $A_{ii} = \pi_i(1-\pi_i)$ and $A_{ij} = -\pi_i\pi_j$ for $j\neq i$. Moreover, define $D_{N\times N}$ to be a diagonal matrix consisting of diagonal elements $p_{0i}/Z = p_{0i}\pi_0$. Then, the optimization can be written as

$$\inf_y y^TDy, \qquad \text{s.t. } y^TAy = 1.$$

Since A is a real symmetric matrix, by using eigen decomposition, it can be written as $A = U\Lambda U^T$, where where U is an orthonormal matrix (the columns of which are eigenvectors of A), and Λ is diagonal (having the eigenvalues of A on the diagonal). Since A is positive definite, all Λ_{ii}'s are positive. Consider a change of variable $w = \Lambda^{1/2}U^Ty$, then the optimization can be written as

$$\beta_2 = \inf_{w:w^Tw=1} w^T\Lambda^{-1/2}U^TDU\Lambda^{-1/2}w.$$

Hence, β_2 is the smallest eigenvalue of the matrix

$$\Lambda^{-1/2}U^TDU\Lambda^{-1/2}$$

i.e., the smallest zero of the following determinant:

$$\begin{aligned}|\Lambda^{-1/2}U^TDU\Lambda^{-1/2}-\mu I| &= |D-\mu U\Lambda U^T|\\ &= |D-\mu A|.\end{aligned}$$

Define $\tilde{\pi} = (\pi_1\cdots\pi_N)^T$. Note that π_0 is not an element of $\tilde{\pi}$ so $\tilde{\pi}$ is not a probability vector. Then, A can be written

$$A = \mathrm{diag}(\tilde{\pi}_i) - \tilde{\pi}\tilde{\pi}^T$$

where $\mathrm{diag}(\tilde{\pi}_i)$ is a $N\times N$ diagonal matrix with elements of $\tilde{\pi}$ on the diagonal. However

$$\begin{aligned}|D-\mu A| &= |D-\mu\mathrm{diag}(\tilde{\pi})+\mu\tilde{\pi}\tilde{\pi}^T|\\ &= |\mathrm{diag}(\pi_0 p_{0i}-\mu\tilde{\pi}_i)+\mu\tilde{\pi}\tilde{\pi}^T|\\ &= |\mathrm{diag}(\pi_0 p_{0i}-\mu\tilde{\pi}_i)\\ &\quad\times(I+\mu\tilde{\pi}\tilde{\pi}^T(\mathrm{diag}(\pi_0 p_{0i}-\mu\tilde{\pi}_i)^{-1}))|.\end{aligned}$$

From the Sylvester's determinant theorem, we know $|I+AB| = |I+BA|$ for any two matrices A and B. Hence

$$\begin{aligned}|D-\mu A| &= |\mathrm{diag}(\pi_0 p_{0i}-\mu\tilde{\pi}_i)|\\ &\quad\times\left(1+\mu\tilde{\pi}^T\mathrm{diag}\left(\frac{1}{\pi_0 p_{0i}-\mu\tilde{\pi}_i}\right)\tilde{\pi}\right).\end{aligned}$$

Therefore, β_2 is the smallest μ that satisfies

$$\left(\prod_{i=1}^N(\pi_0 p_{0i}-\mu\pi_i)\right)\left(1+\mu\sum_{i=1}^N\frac{\pi_i^2}{\pi_0 p_{0i}-\mu\pi_i}\right) = 0.$$

References

[1] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximal throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[2] R. R. Boorstyn, A. Kershenbaum, B. Maglaris, and V. Sahin, "Throughput analysis in multihop CSMA packet radio networks," *IEEE Trans. Commun.*, vol. COM-35, no. 3, pp. 267–274, Mar. 1987.

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

[4] X. Wang and K. Kar, "Throughput modeling and fairness issues in CSMA/CA based ad-hoc networks," in *Proc. IEEE INFOCOM*, 2005, vol. 1, pp. 23–34.

[5] M. Durvy, O. Dousse, and P. Thiran, "Modeling the 802.11 protocol under different capture and sensing capabilities," in *Proc. IEEE INFOCOM*, 2007, pp. 2356–2360.

[6] M. Durvy and P. Thiran, "A packing approach to compare slotted and non-slotted medium access control," in *Proc. IEEE INFOCOM*, 2006, pp. 1–12.

[7] M. Garetto, T. Salonidis, and E. W. Knightly, "Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 4, pp. 864–877, Aug. 2008.

[8] S. C. Liew, C. H. Kai, J. Leung, and B. Wong, "Back-of-the-envelope computation of throughput distributions in CSMA wireless networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 9, pp. 1319–1331, Sep. 2010.

[9] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," in *Proc. Allerton Conf. Commun., Control, Comput.*, 2008, pp. 1511–1519.

[10] J. Ni, B. Tan, and R. Srikant, "Q-CSMA: Queue length-based CSMA/CA algorithms for achieving maximum throughput and low delay in wireless networks," in *Proc. IEEE INFOCOM Mini-Conf.*, 2010, pp. 1–5.

[11] J. Ni, B. Tan, and R. Srikant, "Q-CSMA: Queue length-based CSMA/CA algorithms for achieving maximum throughput and low delay in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 3, pp. 825–836, Jun. 2012.

[12] M. Leconte, J. Ni, and R. Srikant, "Improved bounds on the throughput efficiency of greedy maximal scheduling in wireless networks," in *Proc. ACM MobiHoc*, May 2009, pp. 165–174.

[13] A. Brzezinski, G. Zussman, and E. Modiano, "Distributed throughput maximization in wireless mesh networks via pre-partitioning," *IEEE/ACM Trans. Netw.*, vol. 16, no. 6, pp. 1406–1419, Dec. 2008.

[14] X. Lin and N. B. Shroff, "The impact of imperfect scheduling on cross-layer congestion control in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 14, no. 2, pp. 302–315, Apr. 2006.

[15] S. Rajagopalan, D. Shah, and J. Shin, "Network adiabatic theorem: An efficient randomized protocol for contention resolution," in *Proc. ACM SIGMETRICS/Perform.*, 2009, pp. 133–144.

[16] D. Shah, D. N. C. Tse, and J. N. Tsitsiklis, "Hardness of low delay network scheduling," *IEEE Trans. Inf. Theory*, vol. 57, no. 12, pp. 7810–7817, Dec. 2011.

[17] M. Lotfinezhad and P. Marbach, "Throughput-optimal random access with order-optimal delay," in *Proc. IEEE INFOCOM*, 2011, pp. 2867–2875.

[18] J. Ghaderi and R. Srikant, "On the design of efficient CSMA algorithms for wireless networks," in *Proc. IEEE Conf. Decision Control*, 2010, pp. 954–959.

[19] T. Kim, J. Ni, R. Srikant, and N. Vaidya, "On the achievable throughput of CSMA under imperfect carrier sensing," in *Proc. IEEE INFOCOM*, 2011, pp. 1674–1682.

[20] A. Eryilmaz, R. Srikant, and J. R. Perkins, "Stable scheduling policies for fading wireless channels," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 411–424, Apr. 2005.

[21] L. Jiang, M. Leconte, J. Ni, R. Srikant, and J. Walrand, "Fast mixing of parallel Glauber dynamics and low-delay CSMA scheduling," in *Proc. IEEE INFOCOM Mini-Conf.*, 2011, pp. 371–375.

[22] V. Subramanian and M. Alanyali, "Delay performance of CSMA in networks with bounded degree conflict graphs," in *Proc. IEEE ISIT*, 2011, pp. 2373–2377.

[23] P. Bremaud, *Markov Chains, Gibbs Fields, Monte Carlo Simulation, and Queues*, 2nd ed. New York: Springer-Verlag, 1999, 2001.

[24] M. Jerrum and A. Sinclair, "Approximating the permanent," *SIAM J. Comput.*, vol. 18, pp. 1149–1178, 1989.

[25] D. Randall, "Rapidly mixing Markov chains with applications in computer science and physics," *Comput. Sci. Eng.*, vol. 8, no. 2, pp. 30–41, Mar. 2006.

Javad Ghaderi (S'09) received the B.S. degree in electrical engineering from the University of Tehran, Tehran, Iran, in 2006, and the M.S. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2008, and is currently pursuing the Ph.D. degree in electrical and computer engineering at the University of Illinois at Urbana–Champaign.

His research interests include wireless networks, network control and optimization, and network information theory.

R. Srikant (S'90–M'91–SM'01–F'06) received the B.Tech. degree from the Indian Institute of Technology, Madras, India, in 1985, and the M.S. and Ph.D. degrees from the University of Illinois at Urbana–Champaign in 1988 and 1991, respectively, all in electrical engineering.

He was a Member of Technical Staff with AT&T Bell Laboratories, Holmdel, NJ, from 1991 to 1995. He is currently with the University of Illinois at Urbana-Champaign, where he is the Fredric G. and Elizabeth H. Nearing Professor with the Department of Electrical and Computer Engineering and a Research Professor in the Coordinated Science Laboratory. His research interests include communication networks, stochastic processes, queueing theory, information theory, and game theory.

Prof. Srikant was an Associate Editor of *Automatica*, the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, and the IEEE/ACM TRANSACTIONS ON NETWORKING. He has also served on the Editorial Boards of special issues of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS and IEEE TRANSACTIONS ON INFORMATION THEORY. He was the Chair of the 2002 IEEE Computer Communications Workshop in Santa Fe, NM, and a Program Co-Chair of IEEE INFOCOM 2007. He is a Distinguished Lecturer for the IEEE Communications Society for 2011–2012.