Optimal CSMA Scheduling with Look Ahead Mechanism for Multihop Wireless Networks

Jin-Ghoo Choi and Do Young Eun

***Abstract*—Optimal CSMA scheduling algorithms such as Q-CSMA suffer from large delay, mainly due to the strong correlations residing in consecutive link schedules. Some previous works have remedied this issue with multiple instances of the scheduler, but the baseline scheduler (Q-CSMA) itself remains untouched. By noticing the inherent inefficiency in the contention mechanism of Q-CSMA, we propose LA-CSMA algorithm in which each link looks ahead (LA) its state update in advance and utilizes this free information during the contention stage. We show that our algorithm achieves optimal throughput with reduced correlation in the service process, thereby leading to significantly smaller delay without any additional overhead.**

***Index Terms*—Optimal CSMA, distributed link scheduling, low delay, throughput optimality, multihop wireless network.**

I. Introduction

Recently, we have witnessed the rapid spread of various personal information devices with communication functionality such as smart phones and smart watches. In order to handle huge traffic from these devices, we must be able to fully utilize the limited wireless spectrum as efficiently as possible. Among many approaches over the last two decades, the so-called optimal CSMA (Carrier Sense Multiple Access), a class of link scheduling algorithms for general multihop wireless networks, has been shown to achieve the maximal capacity (or throughput) in a distributed manner [1].

While optimal CSMA algorithms achieve maximal throughput, they suffer from large delay [2]–[4] mainly due to the strong correlations in the consecutive link schedules, inherent in the underlying Glauber dynamics governing how each link updates its activity state under interference constraints. To tackle this issue, [3] has adopted multiple instances of schedulers in parallel on virtual channels. Recently, [4] has proposed a delayed-CSMA algorithm that emulates the multiple schedulers by employing time-delayed link schedules on a single channel. In all these algorithms, the focus has been on reducing the correlations for smaller delay, but the original baseline scheduler (called Q-CSMA [2]) still remains intact.

In this paper, we show that there is a way to reduce the correlations of link schedules in the Q-CSMA itself – the original optimal CSMA algorithm in a discrete time setting, without sacrificing the maximal throughput. Within each slot, Q-CSMA selects a set of links, called a decision set, through a randomized procedure and then probabilistically updates the state of those chosen links based on local information only. It is well known that, as long as each link is chosen in the decision set with positive probability in the long run, the stationary distribution of the resulting schedules obeys a certain product form, leading to the maximal throughput. Specifically, we propose LA-CSMA (Look Ahead CSMA) algorithm in which each link looks ahead its state update in advance and judiciously incorporates this information in the randomized procedure for the decision set. We first show that our algorithm possesses the identical product form stationary distribution with that of Q-CSMA for maximal throughput, and then demonstrate how the correlations in the link service process can be reduced using our approach over the standard Q-CSMA. We also derive the optimal access probability for each link in terms of the local network topological information as well as the queue-length at that link. We show both in theory and simulation that our proposed LA-CSMA significantly outperforms the existing Q-CSMA with access parameters proposed in [5] and even the delayed-CSMA, recently proposed variant of Q-CSMA with low delay [4]. More importantly, our proposed algorithm reaps all the benefits without any additional overhead, and can be made complementary to all the previous schemes [3], [4] with a 'better-performing' baseline scheduling algorithm.

This work was supported in part by the 2014 Yeungnam University research grants and by National Science Foundation under CNS-1217341 and CNS-1423151.

J. Choi is with Department of Information and Communication Engineering, Yeungnam University, Korea (email: jchoi@yu.ac.kr) and D. Eun is with Department of Electrical and Computer Engineering, North Carolina State University, USA (email: dyeun@ncsu.edu).

II. System model

We consider a multihop wireless network modeled by a conflict graph $G(\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of links and $\mathcal{E}$ is the set of edges representing conflict relationship between links. If an edge $(i,j) \in \mathcal{E}$ exists between links i and j, the two links cannot be *active* at the same time to transmit packets. We define an indicator function $s_i \in \{0,1\}$ to represent the activity of link i, i.e., $s_i = 1$ if link i is active and $s_i = 0$ if inactive. Their vector $\mathbf{s} = (s_i)_{i\in\mathcal{N}}$ is called a link schedule. For link i, we denote the set of its neighbors as $N(i) \triangleq \{j \in \mathcal{N} \mid (i,j) \in \mathcal{E}\}$ with its degree $\delta_i \triangleq |N(i)|$. Similarly, for a set of links L, its neighbor set is written as $N(L) = \cup_{i\in L} N(i)$. We say that a link schedule $\mathbf{s} \in \mathcal{F}$ is feasible if it does not incur any transmission collisions, i.e., $s_i + s_j \leq 1$ for every links i, j with $(i,j) \in \mathcal{E}$, where $\mathcal{F}$ is the set of all feasible schedules.

Each link i consists of consecutive time slots, where each time slot is further divided into a control subslot and a data subslot. Link schedulers use the control subslot to determine which links to activate, and the active links transmit packets in the following data subslot. Associated with each link i is a queue with exogenous arrival process $A_i(t)$, $i.i.d.$ over time t with $E\{A_i(t)\} = \alpha_i$, and independent of the service process $s_i(t)$ governed by the scheduling algorithm. When link i is activated by the scheduler, one packet from the queue is transmitted (or served) in a FIFO manner and leaves the network. The queue-length $q_i(t)$ at link i then satisfies $q_i(t+1) = \max\{0, q_i(t) + A_i(t) - s_i(t)\}$ at time t.

Finally, we define the capacity region of a network as $\big\{ \sum_{\mathbf{s}} \sigma_{\mathbf{s}} \mathbf{s} \mid \sum_{\mathbf{s}} \sigma_{\mathbf{s}} = 1, \sigma_{\mathbf{s}} \geq 0, \mathbf{s} \in \mathcal{F} \big\}$, where $\sigma_{\mathbf{s}}$ denotes the probability of schedule $\mathbf{s} \in \mathcal{F}$ being chosen by a link scheduler. If a link scheduler can accommodate any traffic with rate vector $(\alpha_i)_{i \in \mathcal{N}}$ within the capacity region such that the resulting queue-length at each link remains asymptotically finite, the scheduler is said to attain the maximal throughput (or throughput-optimal).

III. Proposed Scheduling Algorithm

A. LA-CSMA: Algorithm Description

We begin with a brief review on the operation of Q-CSMA: Every time slot t, each link retains its activity state of time slot $t-1$ in principle. If a link i intends to change the activity at time slot t, it should be chosen in the decision set $D(t)$ first at time t, constructed every time slot. By construction, the decision set is not allowed to contain conflicting links at the same time, i.e., if $i \in D(t)$, $j \notin D(t)$ for all $j \in N(i)$. Hence, each link i should contend with neighbor links in $N(i)$ to be included in the decision set, by accessing the shared channel with probability a_i (called access probability) in the control subslot. Once the link i wins the contention so that $i \in D(t)$, it has the chance to be active and transmit a packet with probability v_i (called activation probability). We here note that each link must win the contention to be included in the decision set, before having any chance to change its state, regardless of the queue-length. Thus, if a currently inactive link with large queue-length has many contending neighbors, it will be less likely to be chosen in the decision set to turn itself on, leading to even longer queueing delay at that link.

Our observation is that the contention mechanism of Q-CSMA is not very efficient, since a link $i \in D(t)$ does not always change the activity. Specifically, for an inactive link $i \in D(t)$, consider a neighboring link $j \in N(i)$ with large $q_j(t)$. If the link i had not joined the contention in the first place (as it will remain silent with probability $1-v_i$), the link j with large queue-length might have been included in $D(t)$ more likely (due to less contention with i) and activated in the end, contributing to smaller delay. To remedy this issue, we propose Look Ahead CSMA (LA-CSMA) algorithm in which each link looks ahead its *potential* activity at time slot t, and participates in the contention only when this potential activity is different from its activity in the previous time slot. In other words, each link prefetches a random number for the activity and if it draws a random number dictating not to change activity, it doesn't even participate in the contention in the first place, thereby giving more opportunities to some of its neighbors to win the contention instead. Thus, in our approach, a link is expected to have higher chance of winning the contention.

The proposed LA-CSMA is described in Algorithm 1, where $d_i(t) \in \{0,1\}$ is the indicator function with $d_i(t) = 1$ if $i \in D(t)$ and $d_i(t) = 0$ otherwise. If $s_i(t) = 1$ at the end of the control subslot, link i transmits a packet in the subsequent data subslot, all happening within the same time slot t.

- *Initialization* (line 1): $i \notin D(t)$ initially.

Algorithm 1: Proposed algorithm at link i (LA-CSMA).

1 $d_i(t) \leftarrow 0$;
2 $s_i(t) \leftarrow 1$ with probability v_i, and $s_i(t) \leftarrow 0$ otherwise;
3 **if** $s_i(t) \neq s_i(t-1)$ **then**
4   link i accesses the channel with probability a_i;
5   **if** it wins the contention **then**
6         $d_i(t) \leftarrow 1$;
7 **if** $d_i(t) = 0$ **then**
8     $s_i(t) \leftarrow s_i(t-1)$;
9 **if** $\sum_{j \in N(i)} s_j(t-1) \geq 1$ **then**
10     $s_i(t) \leftarrow 0$;

- *Activity look-ahead* (line 2): Link i sets the potential activity as 'on' with activation probability $v_i \in (0,1)$ and accordingly as 'off' with probability $\overline{v_i}$ (we use the notation $\overline{x} \triangleq 1-x$). This activity is just tentative and may change.
- *Contention* (lines 3-6): Link i participates in the contention with access probability $a_i \in (0,1]$ only if it is supposed to change its activity state. If link i wins the contention, it is placed in the decision set.
- *Activity rollback* (lines 7-8): If link i does not win the contention, the link rolls back its activity to the previous activity in the time slot $t-1$.
- *Enforced deactivation* (lines 9-10): Like Q-CSMA, links with active neighbors in time slot $t-1$ cannot be active in time slot t, which ensures the feasibility of the resulting link schedule with the construction of the decision set not containing the mutually conflicting links.

***Remark** 1:* In Q-CSMA, a link i accesses the channel with probability a_i during the contention. This is no longer the case in LA-CSMA. Indeed, at time slot t, link i participates in the contention with probability $v_i a_i$ if $s_i(t-1) = 0$, and with probability $\overline{v_i} a_i$ if $s_i(t-1) = 1$. □

B. Stationary Distribution of Link Schedules

Under LA-CSMA, the link schedule process $\mathbf{s}(t) \in \mathcal{F}$ evolves as a discrete time Markov chain since the current schedule $\mathbf{s}(t)$ is determined by its previous schedule $\mathbf{s}(t-1)$ and other random events that occur in time slot t at each link. In this subsection, we show that the Markov chain $\mathbf{s}(t)$ has its stationary distribution

$$\pi(\mathbf{s}) = \frac{1}{Z} \prod_{\{i \mid s_i = 1\}} \lambda_i, \tag{1}$$

where Z is the normalization constant and $\lambda_i \triangleq \frac{v_i}{1-v_i}$ is a parameter called fugacity, a function of the activation probability v_i of link i. It is well known in the literature that when a scheduling algorithm has the stationary distribution of the form in (1), it achieves the maximal throughput with appropriate fugacities, e.g., $\lambda_i = \exp(f_i(q_i(t)))$, where $f_i(q)$ is a slowly increasing function of q such as $\log\log(1+q)$. We refer to [2] or [6] for the detail.

The link schedule process of Q-CSMA has been shown to be a reversible Markov chain if $D(t)$ is constructed independently of $\mathbf{s}(t-1)$. Such independent construction of $D(t)$ in every time slot t is crucial for the stationary distribution as in (1) and for the throughput optimality [2]. On the other hand, in our LA-CSMA, the decision set $D(t)$ now depends on $\mathbf{s}(t-1)$ since each link i participates in the contention with different probabilities depending on the value of $s_i(t-1)$ as noted in Remark 1. Hence, the following proposition, whose proof is deferred to Appendix, is not trivial. Indeed it is essential for the optimal throughput of LA-CSMA.

***Proposition** 1:* Under LA-CSMA, the link schedule process $\mathbf{s}(t)$ is a reversible Markov chain with its stationary distribution given by (1).

C. Optimal Access Probability for Smaller Delay

In our LA-CSMA, access probabilities $\mathbf{a}=(a_i)_{i\in\mathcal{N}}$ do not affect the resulting stationary distribution of link schedules (thus the throughput), as shown in Proposition 1 and eq. (1). However, the choice of $\mathbf{a}$ significantly influences the correlation structure of the service process $s_i(t)$ of link i, which in turns impacts the queue-length $q_i(t)$. We here analyze and compare the correlations of LA-CSMA with those of Q-CSMA to show how to configure $\mathbf{a}$ to improve the delay performance. Unfortunately, the exact analysis of the whole correlations is intractable even for the standard Q-CSMA. Instead we focus on the dominant term, the lag-1 auto-covariance $R_i(\mathbf{a}) \triangleq E\{s_i(t)s_i(t+1)\}-(E\{s_i(t)\})^2$ of each service process $s_i(t)$, given the access probabilities $\mathbf{a}$. From now on, we use the superscripts 'LA' and 'Q' to represent LA-CSMA and Q-CSMA, respectively. We then obtain the following.

***Proposition** 2:* Under the same choice of $\mathbf{a}$ and $(v_i)_{i\in\mathcal{N}}$,

$$R_i^{LA}(\mathbf{a}) \le R_i^{Q}(\mathbf{a}),\ \forall i\in\mathcal{N}. \tag{2}$$

Proof: First, we observe that

$$\begin{aligned} E\{s_i(t)s_i(t+1)\} &= \Pr\{s_i(t)=1\}E\{s_i(t)s_i(t+1)|s_i(t)=1\} \\ &\quad + \Pr\{s_i(t)=0\}E\{s_i(t)s_i(t+1)|s_i(t)=0\} \\ &= \Pr\{s_i(t)=1\}E\{s_i(t+1)|s_i(t)=1\} \\ &= E\{s_i(t)\}(1-\phi_i(\mathbf{a})), \end{aligned} \tag{3}$$

where $\phi_i(\mathbf{a})$ denotes the probability of active link i at time t turning off at the next time slot $t+1$, for given access probabilities $\mathbf{a}$. Since the marginal distribution of $s_i(t)$ is identical for LA-CSMA and Q-CSMA, we only need to compare the term $\phi_i(\mathbf{a})$ under both algorithms.

For Q-CSMA, in order for link i to change its state from on to off, it should win the contention by accessing the channel with probability a_i while all its neighbors in $N(i)$ do not, and then turns itself off with probability $\overline{v_i}$. In other words, $\phi_i^Q(\mathbf{a}) = \left(a_i\prod_{j\in N(i)}\overline{a_j}\right)\cdot\overline{v_i}$. On the other hand, for LA-CSMA, link i first decides to turn itself off (look-ahead) with probability $\overline{v_i}$, and if this is the case, it participates in the contention with probability a_i, while every neighbor link j does not participate with probability $\overline{v_ja_j} = 1-v_ja_j$. Thus, $\phi_i^{LA}(\mathbf{a}) = \overline{v_i}a_i\prod_{j\in N(i)}\overline{v_ja_j}$. Then, by noting $\overline{v_ja_j} \ge \overline{a_j}$, $\phi_i^{LA}(\mathbf{a}) \ge \phi_i^{Q}(\mathbf{a})$ follows. This completes the proof. ■

While Proposition 2 assures that our LA-CSMA always produces smaller (no larger) lag-1 auto-covariance than Q-CSMA for the service process at every link i, there still exists a room for further improvement by judiciously choosing the access probabilities. Indeed, for a *given* link i, we could always minimize the auto-covariance $R_i(\mathbf{a})$ of its service process trivially by choosing $a_i=1$ and $a_j=0$ for $j\in N(i)$, at the cost of increased auto-covariance of other links. However, we are more interested in jointly reducing the auto-covariances of the service processes over *all* the links in the network. To this end, since $R_i(\mathbf{a})$ is decreasing in $\phi_i(\mathbf{a})$ from (3), we set out to solve the following optimization problem:

$$\max_{\mathbf{a}\in(0,1]^{|\mathcal{N}|}} \prod_i \phi_i(\mathbf{a}). \tag{4}$$

For LA-CSMA, the objective function can be written as $\prod_i \phi_i^{LA}(\mathbf{a}) = \prod_i \left(\overline{v_i}a_i\prod_{j\in N(i)}\overline{v_ja_j}\right) = \prod_i \overline{v_i}a_i(\overline{v_ia_i})^{\delta_i}$ with $\delta_i=|N(i)|$, where the second equality holds since for each link i, every neighbor $j\in N(i)$ contributes the term $\overline{v_ia_i}$ to link i. Taking logarithm on the objective function yields $\Psi(\mathbf{a})\triangleq\sum_i\left(\log\overline{v_i}+\log a_i+\delta_i\log(1-v_ia_i)\right)$, which is concave in $\mathbf{a}=(a_i)_{i\in\mathcal{N}}$. Thus, by solving $\nabla\Psi(\mathbf{a})=0$, the optimal solution becomes $a_i=\frac{1}{v_i(1+\delta_i)}$. Since $a_i\in(0,1]$, the optimal access probability is given by $a_i^*=\min\{1,(v_i(1+\delta_i))^{-1}\}$.

***Remark** 2:* We can also derive the optimal access probabilities for Q-CSMA by solving (4). For Q-CSMA, the objective function to maximize can be written as $\log\left(\prod_i\phi_i^Q(\mathbf{a})\right) = \log\prod_i\left(a_i\prod_{j\in N(i)}\overline{a_j}\right)\overline{v_i} = \sum_i\left(\log\overline{v_i}+\log a_i+\delta_i\log(1-a_i)\right)$, which is again concave in $\mathbf{a}$. Hence, by following similar lines as above, the optimal access probability of Q-CSMA under our framework becomes $a_i^*=(1+\delta_i)^{-1}$ for link i, which coincides with the suggestion in [5]. Note that the authors of [5] derived these optimal access probabilities with the goal of minimizing the second largest eigenvalue of the transition matrix of underlying Markov chain $\mathbf{s}(t)$ only under limited graph topologies such as collocated (or complete) or regular graphs. In contrast, our framework reproduces the same result in a totally different setting, by jointly minimizing the covariance of the service process. More importantly, our framework is for any arbitrary graph topology. In this sense, our result corroborates the conjecture in [5]. □

Our last result below asserts that LA-CSMA with its optimal access probabilities still produces smaller lag-1 auto-covariance than that of Q-CSMA with its optimal access probabilities, at *every* link $i\in\mathcal{N}$. Let $\mathbf{a}^*=(a_i^*)_{i\in\mathcal{N}}$ with $a_i^*=\min\{1,(v_i(1+\delta_i))^{-1}\}$ and $\tilde{\mathbf{a}}^*=(\tilde{a}_i^*)_{i\in\mathcal{N}}$ with $\tilde{a}_i^*=(1+\delta_i)^{-1}$ to distinguish the optimal access probabilities of each algorithm.

***Proposition** 3:* Under the same choice of $(v_i)_{i\in\mathcal{N}}$,

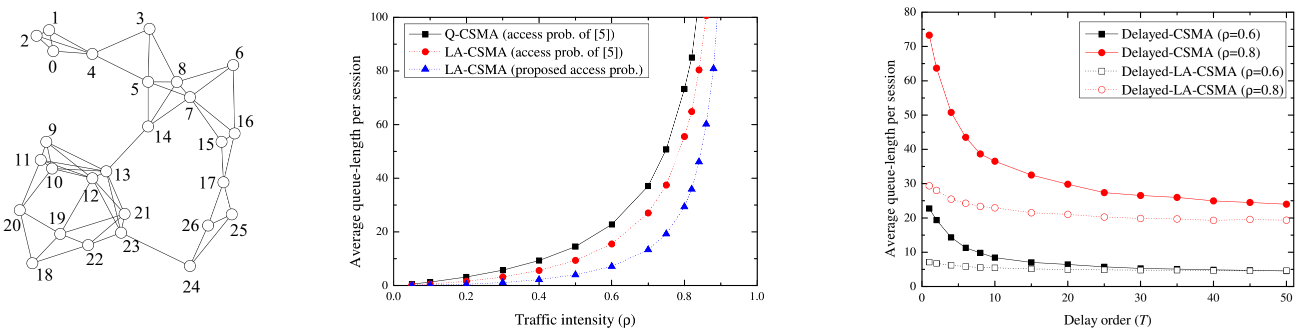
$$R_i^{LA}(\mathbf{a}^*) \le R_i^{Q}(\tilde{\mathbf{a}}^*),\ \forall i\in\mathcal{N}. \tag{5}$$

Proof: Since the marginal distribution of $s_i(t)$ is identical for both LA-CSMA and Q-CSMA under the same v_i's, it suffices to compare the term $\phi_i(\mathbf{a})$ in (3). Observe that $a_i^*\ge\tilde{a}_i^*$ and $v_ja_j^*\le\tilde{a}_j^*$. Thus, $\phi_i^{LA}(\mathbf{a}^*)=\overline{v_i}a_i^*\prod_{j\in N(i)}(1-v_ja_j^*)\ge\left(\tilde{a}_i^*\prod_{j\in N(i)}(1-\tilde{a}_j^*)\right)\cdot\overline{v_i}=\phi_i^Q(\tilde{\mathbf{a}}^*)$, and the result follows. ■

IV. Simulation Results

For simulations, we consider a realistic network topology as shown in Fig. 1a. We use the following 6 (maximal) schedules $\mathbf{s}^{(k)}, k=1,...,6$ to generate input traffic, whose active links therein are respectively

$$S^{(1)}=\{2,5,10,16,19,24\},\quad S^{(2)}=\{4,8,9,17,20,21,24\},$$



(a) Conflict graph for simulations.

(b) Average queue-length per session.

(c) Comparison with delayed-CSMA.

Fig. 1. Simulation Results.

$S^{(3)}=\{0,3,7,12,18,26\}$, $S^{(4)}=\{1,8,11,15,19,23,25\}$,
$S^{(5)}=\{4,7,13,20,22,25\}$, $S^{(6)}=\{4,6,10,14,17,19,24\}$.

In each time slot t, we inject $A_i(t)$ packets to each link i in which $A_i(t) \in \{0,1,...\}$ is a geometric random variable with $E\{A_i(t)\}=\alpha_i=\frac{\rho}{6}\sum_{k=1}^{6} s_i^{(k)}$. Traffic intensity ρ scales the arrival rate vector $(\alpha_i)_{i\in\mathcal{N}}$, and varies from 0 to 1. The fugacity of link i is configured as $\lambda_i = \exp(f_i(q_i(t)))$ with $f_i(q)=\frac{1}{2}\log q$. We run each simulation for 10^6 time slots.

Fig. 1b shows the average queue-length (over the whole network) for different algorithms considered, as the traffic intensity ρ increases from 0 to 1. The average queue-length is closely related to the average delay by Little's law. For Q-CSMA, link i has the access probability $\tilde{a}_i^* = (1+\delta_i)^{-1}$ as in [5]. For LA-CSMA, we test under $\tilde{a}_i^*$, the same access probability as Q-CSMA, as well as under our proposed optimal access probabilities $a_i^* = \min\{1, (v_i(1+\delta_i))^{-1}\}$. As expected from our analysis in Section III-C, the queue-length dramatically decreases under LA-CSMA when compared to Q-CSMA, by about 36% at $\rho=0.5$ with the same choice of the access probability $\tilde{a}_i^*$ in [5], and by 73% with our proposed optimal access probability a_i^*.

Delayed-CSMA, a recent variant of Q-CSMA as proposed in [4], is known to reduce the correlations of consecutive link schedules to attain low delay by constructing link schedule $\mathbf{s}(t)$ based on $\mathbf{s}(t-T)$ rather than $\mathbf{s}(t-1)$, where $T \geq 1$ is the delay order. We can easily incorporate this scheme into our LA-CSMA (and call it delayed-LA-CSMA), in which LA-CSMA generates link schedule $\mathbf{s}(t)$ from $\mathbf{s}(t-T)$ when delay order is T. Fig. 1c shows the average queue-lengths of these two algorithms while varying T, for $\rho=0.6$ and 0.8, respectively. Although the benefit of the 'delayed' approach is marginal for large T, our delayed-LA-CSMA outperforms the original delayed-CSMA with smaller delay, for every choice of T. The improvement for small T is of highly practical relevance since the delayed-CSMA is known to suffer slow-mixing and queue-length overshoot for large T [4].

V. Conclusion

We have proposed a new link scheduling algorithm by enhancing the contention mechanism of the existing Q-CSMA algorithm through the look-ahead, without any additional overhead. Our algorithm substantially outperforms Q-CSMA and even the delayed-CSMA, a state-of-the-art variant of Q-CSMA with low delay.

Appendix

Proof of Proposition 1: Consider two arbitrary states (or link schedules) $\mathbf{s}$ and $\mathbf{s}'$ with transition probability $P(\mathbf{s},\mathbf{s}')>0$. For $\mathbf{s}=(s_i)_{i\in\mathcal{N}}$, $S\triangleq\{i|s_i=1,\forall i\in\mathcal{N}\}$ denotes the set of active links therein. And, given two sets S and S', we define $S\Delta S'\triangleq(S\backslash S')\cup(S'\backslash S)$. We note the set of all links in $\mathcal{N}$ can be partitioned into four disjoint sets, $S\backslash S'$, $S'\backslash S$, $N(S\Delta S')$ and $[(S\Delta S')\cup N(S\Delta S')]^c$. By Algorithm 1, state $\mathbf{s}$ makes a transition to $\mathbf{s}'$ when the links jointly behave as follows.

- $i\in N(S\Delta S')$: $s_i(t-1)=0$, and link i should not participate in the contention at time slot t (with probability $\overline{v_i a_i}$).
- $i\in S\setminus S'$: $s_i(t-1)=1$ and $s_i(t)=0$ (with probability $\overline{v_i}a_i$).
- $i\in S'\setminus S$: $s_i(t-1)=0$ and $s_i(t)=1$ (with probability $v_i a_i$).
- $i \in [(S\Delta S')\cup N(S\Delta S')]^c$: $s_i(t)=s_i(t-1)$. If every link $j\in N(S\Delta S')$ does not participate in the contention, the occurrence of this event depends only on the random events, i.e., channel access and activity decision, at links in $[(S\Delta S')\cup N(S\Delta S')]^c$. This conditional probability is denoted as $\theta(S,S')$ without explicit expression. By symmetry, we have $\theta(S,S')=\theta(S',S)$, which is necessary to ensure the reversibility of Markov chain $\mathbf{s}(t)$.

We then obtain the transition probability from $\mathbf{s}$ to $\mathbf{s}'$ as

$$P(\mathbf{s},\mathbf{s}')=\theta(S,S')\prod_{i\in S\backslash S'}\overline{v_i}a_i\prod_{i\in S'\backslash S}v_i a_i\prod_{i\in N(S\Delta S')}\overline{v_i a_i},$$

and likewise, the transition probability from $\mathbf{s}'$ to $\mathbf{s}$ is

$$P(\mathbf{s}',\mathbf{s})=\theta(S',S)\prod_{i\in S'\backslash S}\overline{v_i}a_i\prod_{i\in S\backslash S'}v_i a_i\prod_{i\in N(S'\Delta S)}\overline{v_i a_i}.$$

It is straightforward to see that the detailed balance equation $\pi(\mathbf{s})P(\mathbf{s},\mathbf{s}')=\pi(\mathbf{s}')P(\mathbf{s}',\mathbf{s})$ holds with $\pi(\mathbf{s})=\frac{1}{Z}\prod_{i\in S}\frac{v_i}{\overline{v_i}}$. Thus, the result follows by noting that $v_i=\frac{\lambda_i}{\lambda_i+1}$.

References

[1] S. Yun, Y. Yi, J. Shin and D. Eun, "Optimal CSMA: A survey," *IEEE ICCS*, 2012.

[2] 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 INFOCOM*, 2010.

[3] P. Huang and X. Lin, "Improving the delay performance of CSMA algorithms: A virtual multi-channel approach," *IEEE INFOCOM*, 2013.

[4] J. Kwak, C. Lee, and D. Eun, "A high-order Markov Chain based scheduling algorithm for low delay in CSMA networks," *IEEE INFOCOM*, 2014.

[5] J. Ghaderi and R. Srikant, "Effect of access probabilities on the delay performance of Q-CSMA algorithms," *IEEE INFOCOM*, 2012

[6] S. Rajagopalan, D. Shah, and J. Shin, "Network adiabatic theorem: An efficient randomized protocol for contention resolution," *ACM SIGMETRICS*, 2009.