Optimal Distributed Scheduling under Time-varying Conditions: A Fast-CSMA Algorithm with Applications

Bin Li and Atilla Eryilmaz

Abstract

Recently, low-complexity and distributed Carrier Sense Multiple Access (CSMA)-based scheduling algorithms have attracted extensive interest due to their throughput-optimal characteristics in general network topologies. However, these algorithms are not well-suited for time-varying environments (i.e., serving real-time traffic under time-varying channel conditions in wireless networks) for two reasons: (1) the mixing time of the underlying CSMA Markov Chain grows with the size of the network, which, for large networks, generates unacceptable delay for deadline-constrained traffic; (2) since the dynamic CSMA parameters are influenced by the arrival and channel state processes, the underlying CSMA Markov Chain may not converge to a steady-state under strict deadline constraints and fading channel conditions.

In this paper, we attack the problem of distributed scheduling for time-varying environments. Specifically, we propose a Fast-CSMA (FCSMA) policy in fully-connected topologies, which converges much faster than the existing CSMA algorithms and thus yields significant advantages for time-varying applications. Then, we design optimal policies based on FCSMA techniques in four challenging and important scenarios in wireless networks for scheduling elastic/inelastic traffic with/without channel state information (CSI) over wireless fading channels.

I. Introduction

Efficient utilization of network resources requires careful interference management among simultaneous transmissions. Of particular interest in the efficient scheduling are Queue-Length-Based (QLB) schedulers (e.g., [2], [3], [4], [5], [6]) due to their provably optimal performance guarantees. Randomization is useful in allowing flexibilities in the design and implementation of such schedulers (e.g., [7]). However, it causes inaccurate operation and may be hurtful if it is not performed within limits (see [8] for more details). One of the most robust randomized schedulers is CSMA-based distributed scheduler (e.g., [9], [10], [11], [12]), whose stationary distribution of the underlying Markov chain has a product-form.

It is well-known that CSMA-based scheduler can maximize long-term average throughput for general wireless topologies. However, these results do not apply to time-varying environments (i.e., scheduling deadline-constrained traffic over wireless fading channels), since their throughput-optimality relies: (i) on the convergence time of the underlying Markov Chain to its steady-state, which grows with the size of the network; and (ii) on relatively stationary conditions in which the CSMA parameters do not change significantly over time so that the instantaneous service rate distribution can stay close to the stationary distribution. Both of these conditions are violated in time-varying environments. For example, packets of deadline constrained traffic are likely to be dropped before the CSMA-based algorithm converges to its steady-state, and the time-varying fading creates significant variations on the CSMA parameters, in which case the instantaneous service rate distribution cannot closely track the stationary distribution. To the best of our knowledge, there does not exist a work that can achieve provably good performance by using attractive CSMA principles under time-varying conditions.

While achieving low delay via distributed scheduling in general topologies is a difficult task (see [13]), in a related work [14] that focuses on grid topologies, the authors have designed an Unlocking CSMA

An earlier version of this paper has appeared in the workshop on Resource Allocation and Cooperation in Wireless Networks (RAWNET), Princeton, NJ, May, 2011 [1].

Bin Li and Atilla Eryilmaz ({lib,eryilmaz}@ece.osu.edu) are with the Department of Electrical and Computer Engineering at The Ohio State University, Columbus, Ohio 43210 USA.

(UCSMA) algorithm with both maximum throughput and order optimal average delay performance, which shows promise for low-delay distributed scheduling in special topologies. However, UCSMA also does not directly apply to deadline-constrained traffic since its measure of delay is on average. Moveover, it is not clear how existing CSMA or UCSMA implementations will perform under fading channel conditions. Thus, designing an optimal distributed scheduling algorithm in time-varying environments remains an open question.

With this motivation, in this work, we address the problem of distributed scheduling in fully connected networks (e.g., Cellular network, Wi-Fi network) for time-varying environments. We propose a Fast-CSMA (FCSMA) algorithm that, despite its similarity of name, fundamentally differs from existing CSMA policies in its design principle: rather than evolving over the set of schedules to reach a favorable steady-state distribution, the FCSMA policy aims to quickly reach one of a set of favorable schedules and stick to it for a duration related to time-varying scale of the application. While the performance of the former strategy is tied to the mixing-time of a Markov Chain, the performance of our strategy is tied to the hitting time, and hence, yields significant advantage for time-varying applications.

In this work, we apply FCSMA techniques in four main scenarios: (1) scheduling with channel state information (CSI) over wireless fading channels; (2) scheduling without CSI over wireless fading channels; (3) deadline-constrained scheduling with CSI over wireless fading channels; (4) deadline-constrained scheduling without CSI over wireless fading channels. The most challenging and important application is the deadline-constrained scheduling, since wireless networks are expected to serve real-time traffic, such as video or voice applications, generated by a large number of users over potentially fading channels. These constraints and requirements, together with the limited shared resources, generate a strong need for distributed algorithms that can efficiently utilize the available resources while maintaining high quality-of-service for the real-time applications. Yet, the strict short-term deadline constraints and long-term throughput requirements associated with most real-time applications complicate the development of provably good distributed solutions.

All existing works in scheduling over wireless fading channels (e.g., [3], [15]) and in deadline-constrained scheduling (e.g., [16], [17], [18], [19]) assume centralized controllers, and hence are not suitable for distributed operation in large-scale networks. Our main contributions in this paper are:

• In Section II, we propose a FCSMA algorithm and discuss its difference from existing CSMA policies. Then, we generalize the Lyapunov Drift Theorem in [20].

• We design an optimal policy based on FCSMA techniques in scheduling with/without CSI over wireless fading channels in Section III and Section IV, respectively.

• We design an optimal policy based on FCSMA techniques in scheduling deadline-constrained traffic with/without CSI over wireless fading channels in Section V and Section VI, respectively.

II. The principle of Fast-CSMA design

We consider a fully-connected network topology where L links contend for data transmission over a single channel. We assume a time-slotted system, where all links start transmission at the beginning of each time slot. Due to the interference constraints, at most one link can transmit in each slot. We call a schedule where at most one link is active in each slot as a *feasible schedule*.

Randomized schedulers (e.g., [7], [9], [21], [22], [23] and [24]) are widely studied due to their flexibilities in development of low-complexity and distributed implementations. The most promising and interesting randomized schedulers are distributed CSMA-based algorithms. We give the definition of continuous-time CSMA algorithm for completeness.

Definition 1 (CSMA Algorithm): Each link l independently generates an exponentially distributed random variable with rate $R_l[t]$ and starts transmitting after this random duration unless it senses another transmission before. If link l senses the transmission, it suspends its backoff timer and resumes it after the completion of this transmission. The transmission time of each link is exponentially distributed with mean 1. ◇

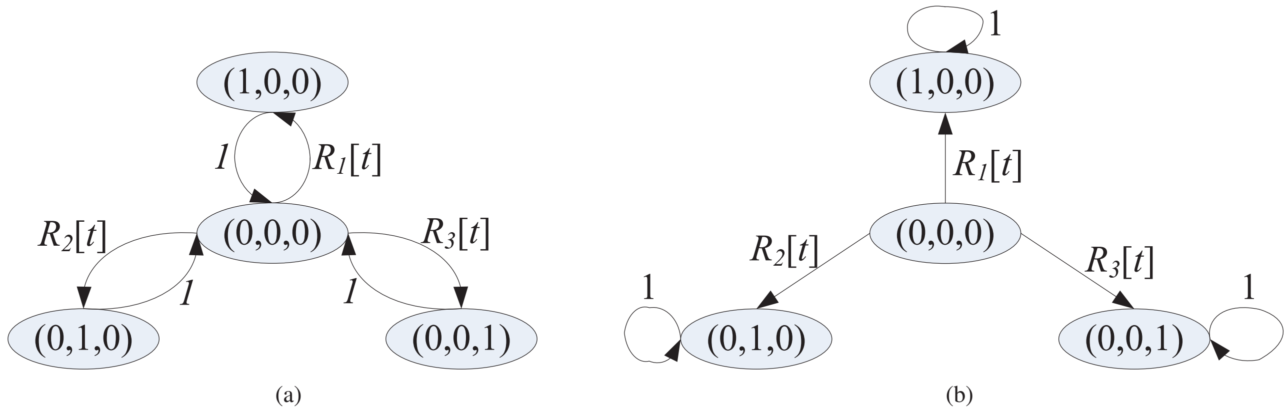

Fig. 1: (a) Markov chain for a CSMA algorithm (b) Markov chain for a FCSMA algorithm

Figure 1a shows the state transition diagram of the underlying Markov Chain for CSMA algorithm when there are 3 links at time t, where each state stands for a feasible schedule. It is easy to see that the stationary distribution of this Markov Chain is

$$P_l = \frac{R_l[t]}{Z[t]}, \forall l, \tag{1}$$

where $Z[t] = \sum_{l=1}^{L} R_l[t]$. Since $\mathbf{R}[t] = (R_l[t])_{l=1}^{L}$ is chosen as a function of network state information (e.g., queue length, channel state information, arrivals) in wireless networks, the underlying Markov Chain for CSMA algorithm is inhomogeneous. Intuitively, the CSMA parameters $\mathbf{R}[t]$ should change slowly such that the instantaneous service rate distribution can stay close to the stationary distribution. Indeed, for the application of scheduling over time-invariant channels (i.e., the transmission rate of each link does not change over time), such mapping has been observed to be optimal (e.g., [11] and [12]) if the CSMA parameter $R_l[t]$ of each link l can take certain functional forms (i.e., $\log\log(\cdot)$) of its queue length at time t. Note that the queue length will change slowly when it is large enough. The purpose of choosing the slowly increasing function is further to make the CSMA parameters as a function of queue length do not change significantly over time.

However, for the application of scheduling over wireless fading channels, the CSMA parameters $\mathbf{R}[t]$ need to be chosen as a function of channel sate information to yield good performance. In such case, no matter what function we choose for the channel state, $\mathbf{R}[t]$ will change significantly as the fading state fluctuates and thus the instantaneous service distribution is not expected to track the stationary distribution. More generally, extending CSMA solutions to stochastic network dynamics or sophisticated application requirements (e.g., serving traffic with strict deadline constraints over wireless fading channels) is difficult for two reasons: 1) the mixing time of the underlying CSMA Markov chain grows with the size of the network, which, for large networks, generates unacceptable delay for deadline-constrained traffic; 2) since the dynamic CSMA parameters $\mathbf{R}[t]$ are influenced by the arrival and channel state process, the underlying CSMA Markov chain may not converge to its steady-state under strict deadline constraints and fading channel conditions.

Thus, designing an optimal and distributed scheduling algorithm for stochastic networks becomes quite challenging. In this paper, we propose a Fast-CSMA strategy that provides provably good performance under time varying conditions. Our approach fundamentally differs from existing CSMA solutions in that our FCSMA policy exploits the fast convergence characteristics of ”hitting times” instead of ”mixing times”.

Definition 2 (Fast-CSMA (FCSMA) Algorithm): At the beginning of each time slot t, each link l independently generates an exponentially distributed random variable with rate $R_l[t]$, and starts transmitting after this random duration unless it senses another transmission before. The link that captures the channel transmits its packets[1] until the end of the slot and the whole process restarts at the completion of the transmission. $\diamond$

Remarks: (1) The operation of FCSMA algorithm resembles that of UCSMA algorithm (see [14]). The difference lies in that UCSMA restarts the CSMA algorithm to achieve both maximum throughput and order optimal average delay in grid network topologies over time-invariant channels by carefully choosing the running period. However, it is unclear whether UCSMA can still work well in time-varying applications.

(2) By choosing the running period for FCSMA algorithm the same as the time scale of network dynamics (i.e., the block length for block fading or maximum allowable deadline), we can show in later sections that FCSMA algorithm exhibits excellent performance in time-varying applications.

Figure 1b gives the state transition diagram of underlying Markov Chain for FCSMA algorithm when there are 3 links, where each state represents a feasible schedule. The convergence time of FCSMA algorithm is tied to the hitting time, while the convergence time of CSMA algorithm is dominated by the mixing time of Markov chain, which generally is large. The hitting time of FCSMA algorithm at slot t is exponentially distributed with mean $\frac{1}{Z[t]}$, which quickly becomes negligibly small as we demonstrate later. Due to its small hitting time, FCSMA policy yields significant advantages over existing CSMA policies evolving slowly to the steady-state and may work well in more challenging environments, i.e., scheduling real-time traffic over wireless fading channels.

In FCSMA, the probability of serving the link l in each slot t will be:

$$\pi_l[t] = \frac{R_l[t]}{Z[t]}(1 - \frac{1}{Z[t]}) \tag{2}$$

In equation (2), $\frac{R_l[t]}{Z[t]}$ is the probability that link l successfully grabs the channel; while $1 - \frac{1}{Z[t]}$ is the average remaining time for serving the packet at each slot given that link l grabs the channel. Let $W_l[t] = \log(R_l[t])$ and $W^*[t] = \max_l W_l[t]$. The following lemma establishes the fact that FCSMA policy picks a link with the weight close to maximum weight with high probability when the maximum weight $W^*[t]$ is large enough at each slot t.

Lemma 1: Given $\epsilon > 0$ and $\zeta > 0$, $\exists \overline{W} \in (0, \infty)$, such that if $W^*[t] > \overline{W}$, then FCSMA policy picks a link l satisfying

$$P\{W_l[t] \geq (1-\epsilon)W^*[t]\} \geq 1 - \zeta. \tag{3}$$

Proof: The proof is similar to that in [25] and [8], and thus is omitted here for brevity. ■

In the rest of paper, we mainly consider two types of traffic: elastic and inelastic traffic, where the inelastic traffic means that each packet has a maximum delay requirement while the elastic traffic does not have such a requirement. We apply the FCSMA technique in four challenging scenarios: scheduling elastic/inelastic traffic with/without Channel State Information (CSI) over wireless fading channels. In each application, we need to carefully design the FCSMA parameters $\mathbf{R}[t] = (R_l[t])_{l=1}^L$ at each slot t to yield optimal performance. To facilitate the flexibility in the design and implementation of the FCSMA algorithm, we define a set of functions (also see [8]):

$\mathcal{F}$:= set of non-negative, nondecreasing and differentiable functions $f(\cdot) : \mathbb{R}_+ \to \mathbb{R}_+$ with $\lim_{x\to\infty} f(x) = \infty$.

$\mathcal{B} := \{f \in \mathcal{F}$: $\lim_{x\to\infty} \frac{f(x+a)}{f(x)} = 1$, for any $a \in \mathbb{R}\}$.

The examples of functions that are in class $\mathcal{B}$ are $f(x) = \log x$, $f(x) = x$ or $f(x) = e^{\sqrt{x}}$. Note that $f(x) = e^x$ does not belong to class $\mathcal{B}$.

[1]If there are no packets awaiting in the link l, it transmits a dummy packet to occupy the channel.

Next, we briefly review same basic concepts in queueing theory and generalize the Lyapunov Drift Theorem in [20], which lays the foundation in establishing the optimality of the proposed algorithms. We use $Q_l[t]$ to denote the queue length of link l at time t. We say that the queue l is *f-stable* (also see [8]) for a function $f \in \mathcal{F}$ if it satisfies

$$\limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[f(Q_l[t])] < \infty. \tag{4}$$

We say that the queue l is *mean rate stable* if $\lim_{T\to\infty} \frac{\mathbb{E}[Q_l[T]]}{T} = 0$.

Theorem 1: (*Generalized Lyapunov Drift*) Consider the Lyapunov function

$$Y(\mathbf{Q}[t]) = \sum_{l=1}^{L} h(Q_l[t]), \tag{5}$$

where $\mathbf{Q}[t] = (Q_l[t])_{l=1}^L$, $h'(x) = f(x)$, and $f \in \mathcal{F}$. Assume $\mathbb{E}[Y(\mathbf{Q}[0])] < \infty$. Suppose there are constants $B > 0$, $\epsilon \geq 0$ such that the following drift condition holds for all slots $t \in \{0, 1, 2, ...\}$ and all possible $\mathbf{Q}[t]$:

$$\mathbb{E}[Y(\mathbf{Q}[t+1]) - Y(\mathbf{Q}[t])|\mathbf{Q}[t] = \mathbf{Q}] \leq -\epsilon \sum_{l=1}^{L} f(Q_l) + B. \tag{6}$$

Then:
(1) If $\epsilon > 0$, then all queues $\mathbf{Q}[t]$ are f-stable;
(2) If $\epsilon \geq 0$, then all queues $\mathbf{Q}[t]$ are mean rate stable.

Remark: In [20], the author established the above theorem by using the quadratic Lyapunov function. We generalize this to the functional space $\mathcal{F}$. It is interesting that the mean rate stability is not sensitive to the functional choice.

Proof: See Appendix A for the proof. ■

Now, we are ready to develop optimal FCSMA algorithms in four challenging applications: scheduling elastic/inelastic traffic with/without CSI over wireless fading channels.

III. Scenario 1: Scheduling elastic traffic with CSI over wireless fading channels

A. Basic Setup

We assume that the wireless channel is independently block fading at each link. We capture the channel fading over link l via $C_l[t]$, which measures the maximum amount of service available in slot t, if scheduled. We assume that $\mathbf{C}[t] = (C_l[t])_{l=1}^L$ are independently distributed random variables over links and identically distributed over time with $C_l[t] \leq C_{\max}, \forall l, t$, for some $C_{\max} < \infty$. We use a binary variable $S_l[t]$ to denote whether the link l is served at slot t, where $S_l[t] = 1$ if the link l can be served at slot t and $S_l[t] = 0$, otherwise. Let $\mathbf{S}[t] = (S[t])_{l=1}^L$ be a feasible schedule. We use $\mathcal{S}$ to denote the collection of feasible schedules. Recall that at most one link can be active in a feasible schedule, due to the fully connected network topology we consider in this paper. We assume that each link l serves its own traffic and maintains them in a queue with $Q_l[t]$ denoting its queue length at the beginning of slot t. Let $A_l[t]$ denote the amount of data arriving at link l in slot t that are independently distributed over links and identically distributed over time with mean λ_l, and $A_l[t] \leq A_{\max}$ for some $A_{\max} < \infty$. Let $U_l[t]$ be the unused service for queue l at the end of slot t, which is upper-bounded by $C_{\max}$. Based on above setup, the evolution of queue l is described as follows:

$$Q_l[t+1] = Q_l[t] + A_l[t] - C_l[t]S_l[t] + U_l[t], \forall l. \tag{7}$$

In this and next section, we consider two main scenarios: known channel state and unknown channel state. For the known/unknown channel state, we mean that each link know/does not know the channel sate information (CSI) at the beginning of each slot. Consider the class of stationary policies $\mathcal{P}_1$ that select $\mathbf{S}[t]$ as a function of $(\mathbf{Q}[t], \mathbf{C}[t])$ for the known channel state case and as a function of $\mathbf{Q}[t]$ for the unknown channel state case, where $\mathbf{Q}[t] = (Q_l[t])_{l=1}^{L}$. We say the system is *$f$-stable* if all queues are f-stable. We call an algorithm *f-optimal* if it can make the system f-stable for any arrival rate vector within the capacity region, which is characterized in Lemma 2 for the known channel state case and Lemma 3 for the unknown channel state case. Next, we propose an optimal FCSMA algorithm with CSI for elastic traffic over wireless fading channels.

B. FCSMA algorithm implementation

To show the optimality of FCSMA algorithm, we first characterize the capacity region and then show that the FCSMA algorithm can support any arrival rate vector within the capacity region. In [15], the authors gave the capacity region $\mathbf{\Lambda}_1(\mathbf{C})$ over a fading channel $\mathbf{C}$. We rewrite it for completeness.

Lemma 2: The capacity region $\mathbf{\Lambda}_1(\mathbf{C})$ is a set of arrival rate vectors $\boldsymbol{\lambda}$ such that there exist non-negative numbers $\alpha(\mathbf{c};\mathbf{s})$ satisfying

$$\sum_{\mathbf{s}\in\mathcal{S}} \alpha(\mathbf{c};\mathbf{s}) = 1, \forall \mathbf{c}, \tag{8}$$

$$\lambda_l < \sum_{\mathbf{c}} P_{\mathbf{C}}(\mathbf{c}) \sum_{\mathbf{s}\in\mathcal{S}} \alpha(\mathbf{c};\mathbf{s}) c_l s_l, \forall l, \tag{9}$$

where $\mathbf{s} = (s_l)_{l=1}^{L}$ and $P_{\mathbf{C}}(\mathbf{c}) = P(\mathbf{C}[t] = \mathbf{c})$. In (9), the right-hand-side (RHS) is the total average service provided for each link and the left-hand-side (LHS) is just the mean arrival rate. Thus, to stabilize the queue, (9) should be satisfied.

We are now ready to develop an optimal FCSMA Algorithm with CSI that can support any arrival rate vector within the capacity region $\mathbf{\Lambda}_1(\mathbf{C})$.

FCSMA Algorithm with CSI for Scheduling Elastic Traffic over Wireless Fading Channels:

At each time slot t, choose the rates

$$R_l[t] = f(Q_l[t])^{C_l[t]}, \forall l, \tag{10}$$

where $f \in \mathcal{F}$.

Theorem 2: If $\log f \in \mathcal{B}$ and $f(0) \geq 1$, the FCSMA Algorithm with CSI for scheduling elastic traffic is $(\log f)$-optimal over wireless fading channels, i.e., for any arrival rate $\lambda \in \mathbf{\Lambda}_1(\mathbf{C})$, it makes the system $(\log f)$-stable.

The proof is a special case of that in Theorem 6 which deals with the deadline-constrained traffic over wireless fading channels. The main difference lies in that the proof in Theorem 2 considers stabilizing data queues, instead of virtual queues in the proof for Theorem 6. However, the argument is almost the same as that in Theorem 6 and thus is omitted here for brevity. Note that there are lots of flexibilities in choosing the function f. The FCSMA algorithm with less steep function is easier to be implemented in practice. However, the higher steepness of function f in FCSMA algorithm leads to smaller average queue length, as we will see in the following simulations.

C. Simulation Results

In this subsection, we perform simulations to validate the optimality of the proposed FCSMA policy with CSI for scheduling elastic traffic over wireless fading channels. In the simulation, there are $N = 10$ links. The number of arrivals in each slot follows a common Bernoulli distribution with mean λ. All links suffer from the ON-OFF channel fading independently with probability $p = 0.8$ that the channel is available at each time slot. Under this setup, we can get the capacity region [3]: $\Lambda_1(\mathbf{C}) = \{\lambda : \lambda < \frac{1-(1-p)^L}{L}\}$. Through numerical calculation, we can get $\lambda < 0.1$. In the simulation, we also consider the FCSMA algorithm with zero hitting time, which assumes it can reach the favorable state instantaneously. We compare our proposed FCSMA policy with $f(x) = e^x$ with QCSMA algorithm [10] with the weight $\log\log(X_l[t]C_l[t] + e)$. To that end, we divide each time slot into M mini-slots. In FCSMA policy, if the link contends for the channel successfully, it will occupy that channel in the rest of time slot; while in QCSMA policy, each link contends for the channel and transmits the data in 1 mini-slot. Here, we do not consider the overhead that the QCSMA policy needs to contend for the channel, which will greatly degrade its performance.

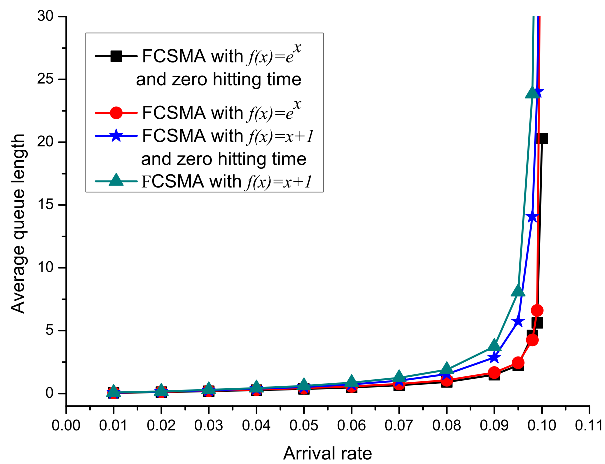

Fig. 2: Average queue length for FCSMA algorithm with CSI

From Figure 2, we can observe that FCSMA algorithm with both $f(x) = e^x$ and $f(x) = x+1$ can achieve full capacity. However, the average queue length under exponential function is smaller than that under linear function, which implies that the FCSMA algorithm with steeper function yields better delay performance. Recall that FCSMA policy waits for random duration before accessing the channel, this random duration can be arbitrarily small when the number of links increases and the queue length is high. We can see from simulations that FCSMA policy has almost the same delay performance as that with zero hitting time.

From Figure 3, we can observe that the average queue length grows very fast under the QCSMA policy with $M = 1$ while the average queue length of FCSMA always stays at a low level. The reason for the poor performance of QCSMA scheme over wireless fading channels is that the underlying Markov chain is controlled by the channel state processes. If the running time of QCSMA policy has the same time scale with the fading block, this Markov chain cannot converge to the steady-state. However, FCMSA policy can quickly lock into one state and exhibits good performance, which is shown in Theorem 2 to be optimal if we carefully choose the parameters. In addition, as M increases, the performance of QCSMA improves. The reason is that the underlying Markov chain has enough time to converge to the steady-state and thus yields better performance.

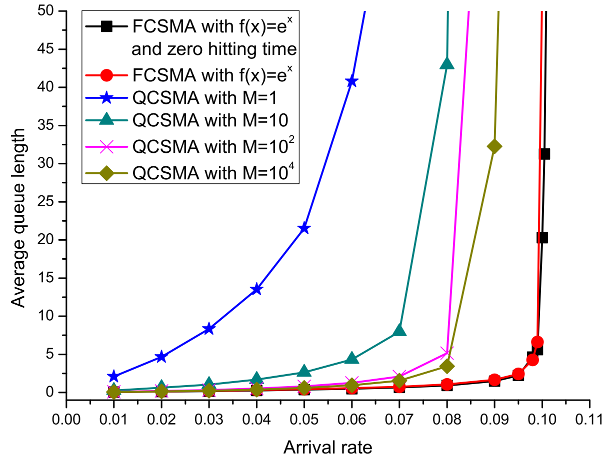

Fig. 3: Performance comparison between FCSMA and QCSMA

IV. Scenario 2: Scheduling Elastic Traffic without CSI over wireless fading channels

A. FCSMA algorithm implementation

We study the case when each link does not know the CSI at the beginning of each time slot. The capacity region $\mathbf{\Lambda}_2(\mathbf{C})$ for unknown CSI over wireless fading channels can be characterized by using the same argument as in [15] and is shown as follows:

Lemma 3: The capacity region $\mathbf{\Lambda}_2(\mathbf{C})$ is a set of arrival rate vectors $\boldsymbol{\lambda}$ such that there exist non-negative numbers $\alpha(\mathbf{s})$ satisfying

$$\sum_{\mathbf{s}\in\mathcal{S}} \alpha(\mathbf{s}) = 1, \forall \mathbf{c}, \tag{11}$$

$$\lambda_l < \sum_{\mathbf{s}\in\mathcal{S}} \alpha(\mathbf{s})\mathbb{E}[C_l]s_l, \forall l, \tag{12}$$

where $\mathbf{s} = (s_l)_{l=1}^{L}$.

Next, we propose an optimal FCSMA algorithm without CSI that can stabilize the system for any arrival rate within the capacity region $\mathbf{\Lambda}_2(\mathbf{C})$.

FCSMA Algorithm without CSI for Scheduling Elastic Traffic over wireless fading channels:

At each time slot t, choose the rates

$$R_l[t] = f(Q_l[t]), \forall l, \tag{13}$$

where $f \in \mathcal{F}$.

Theorem 3: If $\log f \in \mathcal{B}$ and $f(0) \geq 1$, FCSMA algorithm without CSI for scheduling elastic traffic is $(\log f)$-optimal over wireless fading channels, i.e., for any arrival rate $\lambda \in \mathbf{\Lambda}_2(\mathbf{C})$, it makes the system $(\log f)$-stable.

The proof is similar to that in Theorem 2 which considers the elastic traffic with known CSI. We skip it for brevity. Note that CSMA algorithms (e.g., [11] and [12]) without CSI can also achieve optimal throughput over wireless fading channels if the weight has the form $\log\log(q)$ or $\log(q)/g(q)$, where q is the queue length. However, due to the fast hitting time, FCSMA algorithm yields better delay performance than QCSMA algorithm, as we will see in the following simulations.

B. Simulation Results

In this subsection, we perform simulations to validate the throughput optimality and investigate the delay performance of the proposed FCSMA policy without CSI for scheduling elastic traffic over wireless fading channels. The simulation setup is the same as that in Section III-C. It is easy to see that the capacity region: $\Lambda_2(\mathbf{C}) = \{\lambda : \lambda < p\}$. Thus, we can get $\lambda < 0.8$. We compare our proposed FCSMA policy with QCSMA algorithm [10] with the weight $\log\log(Q_l[t] + e)$.

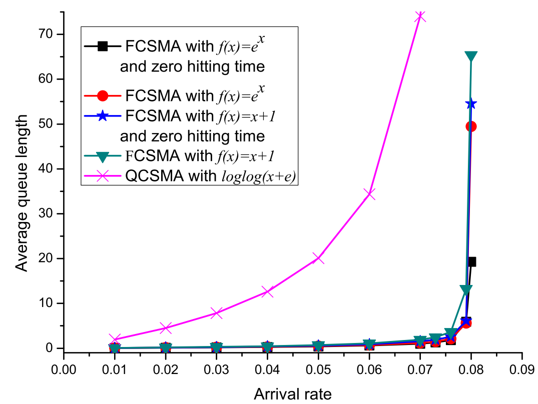

Fig. 4: Average queue length for FCSMA algorithm without CSI

From Figure 4, we can clearly see that FCSMA algorithm with both $f(x) = e^x$ and $f(x) = x + 1$ can achieve full capacity. However, the average queue length of FCSMA algorithm with both exponential and linear functions is smaller than that of QCSMA (see [10], [11]) with $\log\log$ function. The reason for the poor delay performance of QCSMA scheme is that the convergence of underlying Markov Chain is governed by the mixing time, which is normally large. As in Section III-C, we can also observe that FCSMA policy has almost the same delay performance as that with zero hitting time.

V. Scenario 3: Scheduling Inelastic Traffic with CSI over wireless fading channels

A. Basic Setup

We consider the same channel model as in Section III. We assume that each packet has a delay bound of T time slots, which means that if a packet cannot be served during T slots after it arrives, it will be dropped. For convenience, we call a set of T consecutive time slots a *frame*. In the context of fully-connected networks, we associate each real-time flow with a link, and hence use these two terms interchangeably. We assume that all data arrives at each link at the beginning of each frame. Let $A_l[kT]$ denote the amount of data arriving at link l in frame k that are independently distributed over links and identically distributed over time with mean λ_l, and $A_l[kT] \leq A_{\max}$ for some $A_{\max} < \infty$. All the remaining data is dropped at the end of a frame. Each link has a maximum allowable drop rate $\rho_l\lambda_l$, where $\rho_l \in (0, 1)$ is the maximum fraction of data that can be dropped at link l. For example, $\rho_l = 0.1$ means that at most 10% of data can be dropped at link l on average.

Our goal is to find the schedule $\{\mathbf{S}[t]\}_{t\geq 1}$ under the scheduling constraint that at most one link can be scheduled at each time slot and dropping rate constraint that the average drop rate of each link should not be greater than its maximum allowable drop rate. To solve this optimal control problem, we use the intelligent technique in [20] to introduce a virtual queue $X_l[kT]$ for each link l to track the amount of dropped data in frame k. Specifically, the amount of data arriving at virtual queue l at the end of frame k is denoted as $D_l[kT]$, which is equal to $A_l[kT] - \min\{\sum_{t=kT}^{(k+1)T-1} C_l[t]S_l[t], A_l[kT]\}$. We use $I_l[kT]$ to denote the service for virtual queue l at the end of the frame k with mean $\rho_l\lambda_l$, and $I_l[kT] \leq I_{\max}$ for some $I_{\max} < \infty$. Further, we let $U_l[kT]$ denote the unused service for queue l at the end of frame k, which is upper-bounded by $I_{\max}$. Then, the evolution of virtual queue l is described as follows:

$$X_l[(k+1)T] = X_l[kT] + D_l[kT] - I_l[kT] + U_l[kT], \forall l. \tag{14}$$

In this and next section, we consider two main scenarios: known channel state and unknown channel state. For the known channel state case, we assume that the channel state is constant for the duration of a frame and each link knows CSI at the beginning of each frame. For the unknown channel state case, we allow that the channel state changes from time slot to time slot and each link does not know CSI before each transmission, but can determine how much data has been transmitted at each slot after we get feedback from the receiver.

We consider the class of stationary policies $\mathcal{P}_2$ that select $\mathbf{S}[t]$ as a function of $(\mathbf{X}[kT], \mathbf{A}[kT], \mathbf{C}[kT])$ for the known channel state scenario and a function of $(\mathbf{X}[kT], \mathbf{A}[kT])$ for the unknown channel state scenario in frame k, where $\mathbf{X}[kT] = (X_l[kT])_{l=1}^L$ and $\mathbf{A}[kT] = (A_l[kT])_{l=1}^L$. If all virtual queues are mean rate stable, then the average drop rate will meet the required dropping rate constraint automatically (see [20]). We define the maximal satisfiable region as a maximum set of arrival processes for which all virtual queues are mean rate stable under any policy. We call an algorithm *optimal* if it can make all virtual queues mean rate stable for any arrival process within the maximal satisfiable region.

B. FCSMA algorithm implementation

In this subsection, we first characterize the maximal satisfiable region and then propose an optimal FCSMA algorithm with CSI for scheduling inelastic traffic over wireless fading channels.

Consider the class $\mathcal{P}_2$ of stationary policies that base their scheduling decision on the observed vector $(\mathbf{X}[kT], \mathbf{A}[kT], \mathbf{C}[kT])$ in frame k. The next lemma establishes a condition that is necessary for stabilizing the system.

Lemma 4: If there is a policy $\mathbf{P}_0 \in \mathcal{P}$ that can stabilize the virtual queue $\mathbf{X}$, then there exist non-negative numbers $\alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1})$ such that

$$\sum_{\mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1} \in \mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1}) = 1, \forall \mathbf{a}, \mathbf{c}, \tag{15}$$

$$\lambda_l(1-\rho_l) < \sum_{\mathbf{a}} P_{\mathbf{A}}(\mathbf{a}) \sum_{\mathbf{c}} P_{\mathbf{C}}(\mathbf{c}) \sum_{\mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1} \in \mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1}) \min\left\{\sum_{i=0}^{T-1} c_l s_l^i, a_l\right\}, \forall l, \tag{16}$$

where $\mathbf{s}^i = (s_l^i)_{l=1}^L$, $P_{\mathbf{A}}(\mathbf{a}) = P(\mathbf{A}[t] = \mathbf{a})$ and $P_{\mathbf{C}}(\mathbf{c}) = P(\mathbf{C}[t] = \mathbf{c})$.

The proof is almost the same as in [15]. The main difference lies in that our proof deals with the necessary conditions for stabilizing virtual queues instead of data queues as in [15]. We omit it for conciseness. Note that the right hand side of inequality (16) is the total average service provided for each link during one frame; while $\boldsymbol{\lambda} \circ (\mathbf{1} - \boldsymbol{\rho})$ is the total average amount of data at each link that need to be served. Thus, to the meet the maximum drop rate requirement, (16) should be satisfied. We define the *maximal satisfiable* *region* $\Lambda_3(\boldsymbol{\rho}, \mathbf{C})$ as follows:

$$\Lambda_3(\boldsymbol{\rho}, \mathbf{C}) \triangleq \{\mathbf{A} : \exists \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^i) \geq 0, \forall i = 0, 1, ..., T-1, \text{such that both } (15) \text{ and } (16) \text{ satisfy}\}.$$

We are now ready to develop an optimal centralized algorithm with CSI for scheduling inelastic traffic over wireless fading channels.

Centralized Algorithm with CSI for Scheduling Inelastic Traffic over Wireless Fading Channels:

In each frame k, given ($\mathbf{X}[kT]$,$\mathbf{C}[kT]$), perform

$$\{\mathbf{S}^*[t]\}_{t=kT}^{(k+1)T-1} = \text{RAND}\left\{\arg \max_{\{\mathbf{S}[t]\}_{t=kT}^{(k+1)T-1}} \sum_l f(X_l[kT]) \min\left\{C_l[kT] \sum_{t=kT}^{(k+1)T-1} S_l[t], A_l[kT]\right\}\right\}, \quad (17)$$

where $f \in \mathcal{F}$ and RAND$\{\cdot\}$ denotes ties are broken uniformly at random.

Remark: In [18], the authors proposed a centralized algorithm with $f(x) = x$. Our proposed centralized algorithm is more general, which allows more flexibilities in distributed implementations.

Next, we establish the optimality of the centralized algorithm with CSI under certain conditions for function f.

Theorem 4: If $f \in \mathcal{B}$, the centralized algorithm with CSI for scheduling inelastic traffic is optimal over wireless fading channels, i.e., for any arrival process $\mathbf{A} \in \Lambda_3(\boldsymbol{\rho}, \mathbf{C})$, it make the system mean rate stable.

The proof is a generalization of that in [18] and is a special case of that in Theorem 6, where we use the fact that FCSMA techniques to mimic the centralized algorithm when the virtual queue lengths are large enough. Thus, we omit it for brevity. Even though the above centralized algorithm is optimal, it cannot directly be applied in practice due to the need of centralized coordination. Next, we propose a greedy algorithm that is well suited for distributed implementation. To that end, we first give the key identity that facilitates the development of greedy solutions.

Lemma 5: Let $a \geq 0$ and $c[t] \geq 0, \forall t = 0, 1, ..., T-1$. If $s[t] \in \{0, 1\}, \forall t$, then

$$\min\left\{\sum_{t=0}^{T-1} c[t]s[t], a\right\} = \sum_{t=0}^{T-1} \min\left\{c[t], \left(a - \sum_{j=0}^{t-1} c[j]s[j]\right)^+\right\} s[t], \quad (18)$$

where $(x)^+ = \max\{x, 0\}$.

Proof: Please see Appendix B for the proof. ■

Based on Lemma 5, the objective function in (17) can be rewritten as

$$\begin{aligned}
&\sum_l f(X_l[kT]) \min\left\{C_l[kT] \sum_{t=kT}^{(k+1)T-1} S_l[t], A_l[kT]\right\} \\
=& \sum_{t=kT}^{(k+1)T-1} \sum_l f(X_l[kT]) \min\left\{C_l[kT], \left(A_l[kT] - C_l[kT] \sum_{j=kT}^{t-1} S_l[j]\right)^+\right\} S_l[t]. \quad (19)
\end{aligned}$$

We can observe that equation (19) decouples the scheduling decisions over a frame and help develop the greedy solutions that are easy to be implemented distributively.

Greedy Algorithm with CSI for Scheduling Inelastic Traffic over Wireless Fading Channels:

At each time slot $t \in \{kT, kT+1, ..., (k+1)T-1\}$ in frame k, select link $l^*[t]$ such that

$$l^G[t] = \text{RAND}\left\{\arg\max_l f(X_l[kT]) \min\left\{C_l[kT], \left(A_l[kT] - C_l[kT]\sum_{j=kT}^{t-1} S_l[j]\right)^+\right\}\right\}, \quad (20)$$

where $f \in \mathcal{F}$.

Theorem 5: The Greedy Algorithm with CSI is an optimal solution to the problem (17) and thus is optimal for scheduling inelastic traffic over wireless fading channels if $f \in \mathcal{B}$.

The proof is a special case of that in Theorem 8. The channel state is constant over a frame in the proof for Theorem 5, while the channel state changes from slot to slot in a frame in that for Theorem 8 which makes it more challenging to deal with. Next, we expand on the distributed implementation of the greedy solution by using the FCSMA technique.

FCSMA Algorithm with CSI for Scheduling Inelastic Traffic over Wireless Fading Channels:

At each time slot $t \in \{kT, kT+1, ..., (k+1)T-1\}$ in frame k, choose the rates

$$R_l[t] = g(X_l[kT])^{\min\left\{C_l[kT], \left(A_l[kT] - C_l[kT]\sum_{j=kT}^{t-1} S_l[j]\right)^+\right\}}, \forall l, \quad (21)$$

where $f \in \mathcal{F}$.

Theorem 6: If $f(x) = \log g(x) \in \mathcal{B}$ and $g(0) \geq 1$, FCSMA algorithm with CSI for scheduling inelastic traffic is optimal over wireless fading channels, i.e., for any arrival process $\mathbf{A} \in \Lambda_3(\boldsymbol{\rho}, \mathbf{C})$, it make the system mean rate stable.

Proof: See the Appendix C for the proof. ■

C. Simulation Results

In this subsection, we perform simulations to validate the optimality of the proposed FCSMA policy with CSI for scheduling inelastic traffic with deadline constraint T slots over wireless fading channels. In the simulation, there are $L = 10$ links and each frame has $T = 5$ slots. All links require the maximum fraction of dropped packets to not exceed $\rho = 0.3$. The number of arrivals in each frame follows common Bernoulli distribution that the number of arrivals equal to T with probability λ. All links suffer from the ON-OFF channel fading independently with probability $p = 0.8$ that the channel is available in each frame. The service for virtual queue also follows Bernoulli distribution that the maximum available service equals to T with probability $\rho\lambda$. Under this setup, we can use the same technique in paper [3] to get the maximal satisfiable region: $\Lambda_3(\boldsymbol{\rho}, \mathbf{C}) = \{\lambda : L(1-\rho)\lambda < 1 - (1-p\lambda)^L\}$. Through numerical calculation, we can get $\lambda < 0.038$.

From Figure 5, we can observe that FCSMA algorithm with both $g(x) = e^x$ and $g(x) = x+1$ can achieve maximal satisfiable region. In addition, we see that the average virtual queue length of FCSMA algorithm with exponential function is smaller than that with linear function. However, the meaning of smaller virtual queue length is unclear in this setup. We will explore it in our future research. Furthermore, we can observe that FCSMA policy has almost the same performance as that with zero hitting time as in Section III-C.

VI. Scenario 4: Scheduling Inelastic Traffic without CSI over Wireless Fading Channels

In this section, we consider the inelastic traffic scheduling without CSI over wireless fading channels. We assume that each link knows how much data has been transmitted at the end of each slot by using per-slot

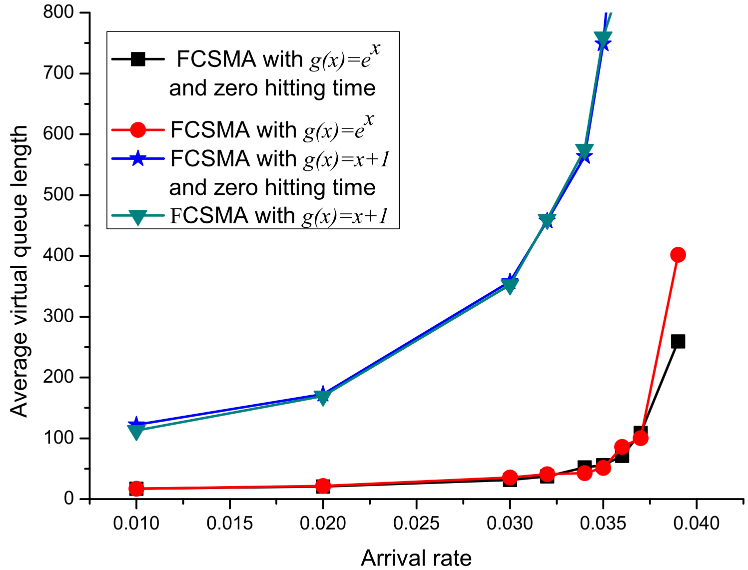

Fig. 5: Average queue length for FCSMA algorithm with CSI for inelastic traffic

feedback information. The per-slot feedback complicates the design of distributed scheduling algorithm. But, we still can find a similar greedy solution as in Section V and design its distributed algorithm by using FCSMA techniques.

Consider the class $\mathcal{P}_2$ of stationary policies that base their scheduling decision on the observed vector $(\mathbf{X}[kT], \mathbf{A}[kT])$ in frame k. The next lemma establishes a condition that is necessary for stabilizing the system.

Lemma 6: If there is a policy $\mathbf{P}_0 \in \mathcal{P}$ that can stabilize the virtual queue $\mathbf{X}$, then there exist non-negative numbers $\alpha_0(\mathbf{a}; \mathbf{s}^0), \alpha_1(\mathbf{a}, \mathbf{s}^0; \mathbf{s}^1), ..., \alpha_{T-1}(\mathbf{a}, \mathbf{s}^0, ..., \mathbf{s}^{T-2}; \mathbf{s}^{T-1})$, such that

$$\sum_{\mathbf{s}^0 \in \mathcal{S}} \alpha_0(\mathbf{a}; \mathbf{s}^0) = 1, \text{ and } \sum_{\mathbf{s}^i \in \mathcal{S}} \alpha_i(\mathbf{a}, \mathbf{s}^0, ..., \mathbf{s}^{i-1}; \mathbf{s}^i) = 1, \forall \mathbf{a}, i = 1, 2, ..., T-1, \tag{22}$$

$$\lambda_l(1-\rho_l) < \sum_{\mathbf{a}} P_{\mathbf{A}}(\mathbf{a}) \sum_{\mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1} \in \mathcal{S}} \alpha_0(\mathbf{a}; \mathbf{s}^0)\alpha_1(\mathbf{a}, \mathbf{s}^0; \mathbf{s}^1)...\alpha_{T-1}(\mathbf{a}, \mathbf{s}^0, ..., \mathbf{s}^{T-2}; \mathbf{s}^{T-1})\mathbb{E}\left[\min\left\{\sum_{i=0}^{T-1} c_l s_l^i, a_l\right\}\right], \forall l. \tag{23}$$

The proof is almost the same as [15] and hence is omitted here. We define maximal satisfiable region $\mathbf{\Lambda}_4(\boldsymbol{\rho}, \mathbf{C})$ as follows:

$$\mathbf{\Lambda}_4(\boldsymbol{\rho}, \mathbf{C}) \triangleq \{\mathbf{A} : \exists \alpha_0(\mathbf{a}; \mathbf{s}^0), \alpha_1(\mathbf{a}, \mathbf{s}^0; \mathbf{s}^1), ..., \alpha_{T-1}(\mathbf{a}, \mathbf{s}^0, ..., \mathbf{s}^{T-2}; \mathbf{s}^{T-1}) \geq 0, \text{such that both (22) and (23) satisfy}\}.$$

Next, we develop centralized algorithm without CSI for scheduling inelastic traffic over wireless fading channels.

Centralized Algorithm without CSI for Scheduling Inelastic Traffic over Wireless Fading Channels:

In each frame k, given $(\mathbf{X}[kT])$, perform

$$\{\mathbf{S}^*[t]\}_{t=kT+1}^{(k+1)T} = \text{RAND}\left\{\arg\max_{\{\mathbf{S}[t]\}_{t=kT}^{(k+1)T-1}} \mathbb{E}\left[\sum_l f(X_l[kT]) \min\left\{\sum_{t=kT}^{(k+1)T-1} C_l[t]S_l[t], A_l[kT]\right\}\right]\right\}, \quad (24)$$

where $f \in \mathcal{F}$.

Remark: In [18], the authors designed a centralized algorithm with $f(x) = x$. Our proposed centralized algorithm generalizes this to a large space of functions f, and allows for more flexibilities in distributed implementations.

Next, we establish the optimality of the centralized algorithm without CSI under certain conditions for function f.

Theorem 7: If $f \in \mathcal{B}$, centralized algorithm without CSI for scheduling inelastic traffic is optimal over wireless fading channels, i.e., for any arrival process $\mathbf{A} \in \mathbf{\Lambda}_4(\boldsymbol{\rho}, \mathbf{C})$, it makes the system mean rate stable.

The proof is a generalization of that in [18] and can be followed the same argument as that in Theorem 6. Thus, we omit it here for brevity. The centralized algorithm without CSI is quite complicated, since it couples the scheduling decisions in each frame. Under the per-slot feedback assumption, the optimization problem (24) can be solved by using dynamic programming. Based on Lemma 5, we have the following identity:

$$\begin{aligned}&\sum_l f(X_l[kT]) \min\left\{\sum_{t=kT}^{(k+1)T-1} C_l[t]S_l[t], A_l[kT]\right\}\\ =&\sum_{t=kT}^{(k+1)T-1} \sum_l f(X_l[kT]) \min\left\{C_l[t], \left(A_l[kT] - \sum_{j=kT}^{t-1} C_l[j]S_l[j]\right)^+\right\} S_l[t]. \end{aligned}\quad (25)$$

By using (25), we can get the following backward equation (see [26]) for the optimization problem (24).

Backward Equation for (24):
At each slot $t \in \{kT, kT+1, ..., (k+1)T-1\}$ in frame k, given $\mathbf{X}[kT]$ and $\{(\mathbf{C}[j], \mathbf{S}[j])\}_{j=kT}^{t-1}$, select link $l^*[t]$ such that

$$\begin{aligned}l^*[t] = \text{RAND}\Bigg\{&\arg\max_l \Bigg(f(X_l[kT])\mathbb{E}\left[\min\left\{C_l[t], \left(A_l[kT] - \sum_{j=kT}^{t-1} S_l[j]C_l[j]\right)^+\right\}\right]\\ &+ \max_{\{\mathbf{S}[r]\}_{r=t+1}^{(k+1)T-1}} \sum_{i=t+1}^{(k+1)T-1} \sum_{l'=1}^{L} f(X_{l'}[kT])\mathbb{E}\left[\min\left\{C_{l'}[i], \left(A_{l'}[kT] - \sum_{j=kT}^{i-1} S_{l'}[j]C_{l'}[j]\right)^+\right\}\right] S_{l'}[i]\Bigg)\Bigg\}, \end{aligned}\quad (26)$$

where $f \in \mathcal{F}$.

At first glance, the optimal solution to problem (26) at each time slot depends on the future slots and thus is difficult to be implemented distributively. However, it may still be possible to decouple the scheduling decisions over a frame, since the channel states are i.i.d. across over time slots. Next, we will show that this is the case in our setup.

Greedy Algorithm without CSI for Scheduling Inelastic Traffic over Wireless Fading Channels:
At each time slot $t \in \{kT, kT+1, ..., (k+1)T-1\}$ in frame k, given $\mathbf{X}[kT]$ and $\{(\mathbf{C}[j], \mathbf{S}[j])\}_{j=kT}^{t-1}$, select

link $l^*[t]$ such that

$$l^G[t] = \text{RAND}\left\{ \arg\max_l f(X_l[kT])\mathbb{E}\left[\min\left\{C_l[t], \left(A_l[kT] - \sum_{j=kT}^{t-1} C_l[j]S_l[j]\right)^+\right\}\right]\right\}, \quad (27)$$

where $f \in \mathcal{F}$.

Theorem 8: The Greedy algorithm without CSI is optimal for problem (26) and thus is optimal for scheduling inelastic traffic over wireless fading channels if $f \in \mathcal{B}$.

Proof: Without loss of generality, we consider the frame $k = 0$ and $t \in \{0, 1, ..., T-1\}$. We will show that if

$$l^G \in \arg\max_l f(X_l[0])\mathbb{E}\left[\min\left\{C_l[t], A_l[0]\right\}\right], \quad (28)$$

then

$$\begin{aligned} & f(X_{l^G}[0])\mathbb{E}\left[\min\left\{C_{l^G}[t], A_{l^G}[0]\right\}\right] + \max_{\{\mathbf{S}[r]\}_{r=t+1}^{T-1}} \sum_{i=t+1}^{T-1} \sum_l f(X_l[0])\mathbb{E}\left[\min\left\{C_l[i], \left(A_l[0] - \sum_{j=0}^{i-1} S_l[j]C_l[j]\right)^+\right\}\right] \\ \geq & f(X_m[0])\mathbb{E}\left[\min\left\{C_m[t], A_m[0]\right\}\right] + \max_{\{\mathbf{S}[r]\}_{r=t+1}^{T-1}} \sum_{i=t+1}^{T-1} \sum_l f(X_l[0])\mathbb{E}\left[\min\left\{C_l[i], \left(A_l[0] - \sum_{j=0}^{i-1} S_l[j]C_l[j]\right)^+\right\}\right], \forall m \neq l^G. \end{aligned} \quad (29)$$

Recall that at most one link can be scheduled at each slot. Let $\mathbf{d} = (d[t], d[t+1], ...d[T-1])$ be a sequence of links chosen in the remaining time of a frame, where the element $d[i]$ denotes the link that is scheduled at slot i. Let $\mathcal{D}$ be the collection of the sequence of selected links. Let $W_{\mathbf{d}}$ be the total weight when we choose $\mathbf{d}$ in a frame, that is,

$$W_{\mathbf{d}} = f(X_{d[t]}[0])\mathbb{E}\left[\min\left\{C_{d[t]}[t], A_{d[t]}[0]\right\}\right] + \sum_{i=t+1}^{T-1} \sum_l f(X_l[0])\mathbb{E}\left[\min\left\{C_l[i], \left(A_l[0] - \sum_{j=0}^{i-1} S_l[j]C_l[j]\right)^+\right\}\right], \quad (30)$$

where $S_{d[r]}[r] = 1$ and $S_l[r] = 0, \forall l \neq d[r]$.

Let $\mathcal{F}_l = \{\mathbf{d} \in \mathcal{D}, d[t] = l\}$. Then, (29) can be rewritten as

$$\max_{\mathbf{d} \in \mathcal{F}_{l^*}} W_{\mathbf{d}} \geq \max_{\mathbf{d} \in \mathcal{F}_m} W_{\mathbf{d}}, \forall m \neq l^G. \quad (31)$$

Given any $m \neq l^G$, we have the following two cases:

(1) If $\mathbf{d} \in \mathcal{F}_m$ includes the element l^G, then a permutation of $\mathbf{d}$ with the first element being l^G should be in $\mathcal{F}_{l^G}$. Since the channel states are i.i.d. over time slots, any permutation of $\mathbf{d}$ does not change its weight and thus $W_{\mathbf{d}} \leq \max_{\mathbf{e} \in \mathcal{F}_{l^G}} W_{\mathbf{e}}$.

(2) If $\mathbf{d} \in \mathcal{F}_m$ does not include the element l^G, then it is easy to see that $W_{\mathbf{d}} \leq W_{\mathbf{e}}$, where $\mathbf{e} = (l^G, d[t+1], d[t+2], ..., d[T-1])$. Since $\mathbf{e} \in \mathcal{F}_{l^G}$, we have $W_{\mathbf{d}} \leq \max_{\mathbf{e} \in \mathcal{F}_{l^G}} W_{\mathbf{e}}$.

Thus, we have $\max_{\mathbf{e} \in \mathcal{F}_{l^G}} W_{\mathbf{e}} \geq W_{\mathbf{d}}, \forall d \in \mathcal{F}_m$ and hence we have the desired result (31). ■

Next, we illustrate the distributed implementation of greedy solutions by using FCSMA techniques.

FCSMA Algorithm without CSI for Scheduling Inelastic Traffic over Wireless Fading Channels:
At each time slot $t \in \{kT, kT+1, ..., (k+1)T-1\}$ in frame k, given $\mathbf{X}[kT]$ and $\{(\mathbf{C}[j], \mathbf{S}[j])\}_{j=kT}^{t-1}$, choose the rates

$$R_l[t] = g(X_l[kT])^{\mathbb{E}\left[\min\left\{C_l[t], \left(A_l[kT] - \sum_{j=kT}^{t-1} C_l[j]S_l[j]\right)^+\right\}\right]}, \forall l, \quad (32)$$

where $f \in \mathcal{F}$.

Theorem 9: If $f(x) = \log g(x) \in \mathcal{B}$ and $g(0) \geq 1$, FCSMA algorithm without CSI for scheduling inelastic traffic is optimal over wireless fading channels, i.e., for any arrival process $\mathbf{A} \in \mathbf{\Lambda}_4(\boldsymbol{\rho}, \mathbf{C})$, it makes the system mean rate stable.

The proof is similar to that in Theorem 6 which considers the inelastic traffic with CSI over wireless fading channels. We skip it for conciseness.

A. Simulation Results

In this subsection, we perform simulations to validate the optimality of the proposed FCSMA policy without CSI for scheduling inelastic traffic with deadline constraint T slots over wireless fading channels. The simulation setup is the same as that in Section V-C. The main difference is that the fading channels change from slot to slot. The maximal satisfiable region under this setup is $\Lambda_4(\boldsymbol{\rho}, \mathbf{C}) = \{\lambda : L(1-\rho)\lambda < p\left(1-(1-\lambda)^L\right)\}$. Through numerical calculation, we can get $\lambda < 0.031$.

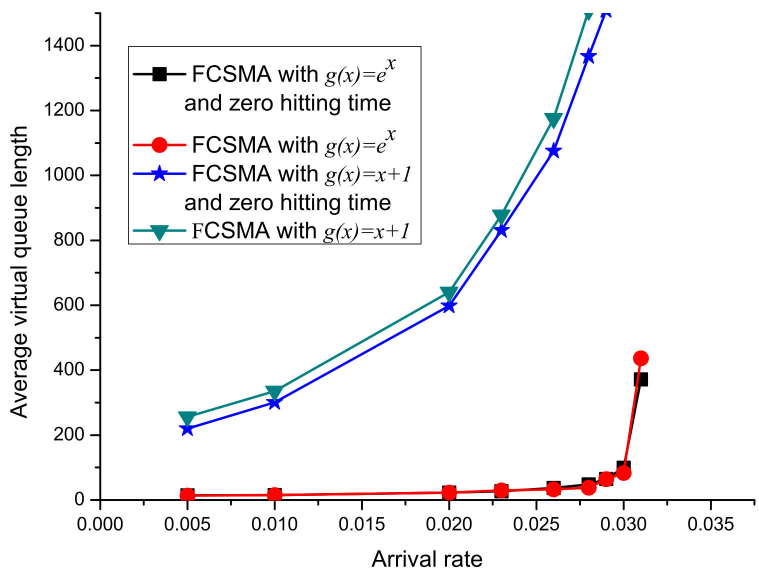

Fig. 6: Average queue length for FCSMA algorithm without CSI for inelastic traffic

From Figure 6, we can observe that FCSMA algorithm with both $g(x) = e^x$ and $g(x) = x + 1$ can achieve full capacity. In addition, we see that the average virtual queue length of FCSMA algorithm with exponential function is smaller than that with linear function. However, the meaning of smaller virtual queue length is less understood in this setup. We will explore it in our future research. Furthermore, we can observe that FCSMA policy can achieve almost the same performance as that with zero hitting time as in Section III-C.

VII. Conclusions

In this paper, we first proposed an FCSMA algorithm that quickly reach the favorable state in fully connected network topologies. Due to the fast hitting time, the FCSMA policy exhibits significant advantages over existing CSMA algorithms for time-varying applications, which are important and popular in wireless networks. Then, we apply FCSMA algorithm to design optimal policies for scheduling elastic/inelastic traffic with/without CSI over wireless fading channels. In the future, we will try to explore distributed scheduling algorithms for time-varying environments in multi-hop network topologies.

Appendix A
Proof for Theorem 1

(1) By taking the expectation of (6), we have

$$\mathbb{E}[Y(\mathbf{Q}[t+1])] - \mathbb{E}[Y(\mathbf{Q}[t])] \leq -\epsilon \mathbb{E}\left[\sum_{l=1}^{L} f(Q_l[t])\right] + B. \tag{33}$$

By summing the above over $t \in \{0, 1, ..., T-1\}$ for some $T > 0$, we have

$$\mathbb{E}[Y(\mathbf{Q}[T])] - \mathbb{E}[Y(\mathbf{Q}[0])] \leq -\epsilon \sum_{t=0}^{T-1} \sum_{l=1}^{L} \mathbb{E}\left[f(Q_l[t])\right] + BT. \tag{34}$$

Thus, by using the fact $\mathbb{E}[Y(\mathbf{Q}[t])] \geq 0$, we have

$$\frac{1}{T} \sum_{t=0}^{T-1} \sum_{l=1}^{L} \mathbb{E}\left[f(Q_l[t])\right] \leq \frac{B}{\epsilon} + \frac{\mathbb{E}[Y(\mathbf{Q}[0])]}{T\epsilon} \tag{35}$$

By taking a limit as $T \to \infty$, we have the desired result.
(2) From inequality (34), we have for all time slots $T \geq 0$,

$$\mathbb{E}[Y(\mathbf{Q}[T])] - \mathbb{E}[Y(\mathbf{Q}[0])] \leq BT. \tag{36}$$

Thus, we have

$$\mathbb{E}[h(Q_l[T])] \leq \mathbb{E}\left[\sum_{l=1}^{L} h(Q_l[T])\right] \leq BT + \mathbb{E}[Y(\mathbf{Q}[0])]. \tag{37}$$

Since $h(x) = \int_0^x f(y)dy + f(0)$, we have

$$\mathbb{E}\left[\int_0^{Q_l[T]} f(y)dy\right] \leq BT + \mathbb{E}[Y(\mathbf{Q}[0])] - f(0), \forall T \geq 1. \tag{38}$$

Note that $\mathbb{E}[Y(\mathbf{Q}[0])] = \sum_{l=1}^{L} h(Q_l[0]) \geq f(0)$. Let $B_1 := B + \mathbb{E}[Y(\mathbf{Q}[0])] - f(0)$, then, we have

$$\mathbb{E}\left[\int_0^{Q_l[T]} f(y)dy\right] \leq B_1 T, \forall T \geq 1. \tag{39}$$

Note that $w(x) := \int_0^x f(y)dy$ is a convex function, since $f(x)$ is an non-decreasing function and $w''(x) = f'(x) \geq 0$. Thus, by Jensen's inequality, we have

$$\mathbb{E}\left[\int_0^{Q_l[T]} f(y)dy\right] \geq \int_0^{\mathbb{E}[Q_l[T]]} f(y)dy. \tag{40}$$

In addition, we have

$$\int_0^{\mathbb{E}[Q_l[T]]} f(y)dy \geq \int_{\frac{\mathbb{E}[Q_l[T]]}{2}}^{\mathbb{E}[Q_l[T]]} f(y)dy \geq \frac{\mathbb{E}[Q_l[T]]}{2} f\left(\frac{\mathbb{E}[Q_l[T]]}{2}\right). \tag{41}$$

Thus, we have

$$\frac{\mathbb{E}[Q_l[T]]}{2} f\left(\frac{\mathbb{E}[Q_l[T]]}{2}\right) \leq B_1 T, \forall T \geq 1, \tag{42}$$

which implies that

$$\frac{\mathbb{E}[Q_l[T]]}{T} \leq \frac{2B_1}{f\left(\frac{\mathbb{E}[Q_l[T]]}{2}\right)}. \tag{43}$$

Let $\epsilon > 0$, since $f \in \mathcal{F}$, we can find $Q_0 > 0$ such that $x > Q_0$ implies $\frac{2B_1}{f(\frac{x}{2})} < \epsilon$. In addition, there exists $T_0 > 0$ such that $T > T_0$ implies $\frac{Q_0}{T} < \epsilon$. Thus,
(i) If $\mathbb{E}[Q_l[T]] > Q_0$, then

$$\frac{\mathbb{E}[Q_l[T]]}{T} \leq \frac{2B_1}{f\left(\frac{\mathbb{E}[Q_l[T]]}{2}\right)} < \epsilon; \tag{44}$$

(ii) If $\mathbb{E}[Q_l[T]] \leq Q_0$, then

$$\frac{\mathbb{E}[Q_l[T]]}{T} \leq \frac{Q_0}{T} < \epsilon. \tag{45}$$

Thus, we have $\limsup_{T\to\infty} \frac{\mathbb{E}[Q_l[T]]}{T} = 0$. Since $\liminf_{T\to\infty} \frac{\mathbb{E}[Q_l[T]]}{T} \geq 0$, we have $\lim_{T\to\infty} \frac{\mathbb{E}[Q_l[T]]}{T} = 0$.

Appendix B
Proof for Lemma 5

If $T = 1$, we have LHS $= \min\{c[0]s[0], a\}$ and RHS $= \min\{c[0], a\}s[0]$. Since $s[0] = 0$ or 1, LHS $=$ RHS.

Assume that $T = k$, (18) is true, that is,

$$\min\left\{\sum_{t=0}^{k-1} c[t]s[t], a\right\} = \sum_{t=0}^{k-1} \min\left\{c[t], \left(a - \sum_{j=0}^{t-1} c[j]s[j]\right)^+\right\} s[t]. \tag{46}$$

Then, for $T = k+1$, we have

$$\begin{aligned}
&\sum_{t=0}^{k} \min\left\{c[t], \left(a - \sum_{j=0}^{t-1} c[j]s[j]\right)^+\right\} s[t] \\
&= \sum_{t=0}^{k-1} \min\left\{c[t], \left(a - \sum_{j=0}^{t-1} c[j]s[j]\right)^+\right\} s[t] + \min\left\{c[k], \left(a - \sum_{j=0}^{k-1} c[j]s[j]\right)^+\right\} s[k] \\
&\stackrel{(a)}{=} \min\left\{\sum_{t=0}^{k-1} c[t]s[t], a\right\} + \min\left\{c[k]s[k], \left(a - \sum_{j=0}^{k-1} c[j]s[j]\right)^+\right\} \\
&\stackrel{(b)}{=} \min\left\{\sum_{t=0}^{k} c[t]s[t], a + c[k]s[k], \max\left\{a, \sum_{t=0}^{k-1} c[t]s[t]\right\}, a + \left(a - \sum_{j=0}^{k-1} c[j]s[j]\right)^+\right\}, \qquad (47)
\end{aligned}$$

where step (a) follows the induction assumption and uses the fact that $s[t]$ only takes 0 or 1; step (b) uses the following identity: $\min\{a,b\} + \min\{c,d\} = \min\{a+c, a+d, b+c, b+d\}$.

(1) If $a \geq \sum_{t=0}^{k-1} c[t]s[t]$, then

$$\begin{aligned}
(47) &= \min\left\{\sum_{t=0}^{k} c[t]s[t], a + c[k]s[k], a, a + \left(a - \sum_{j=0}^{k-1} c[j]s[j]\right)^+\right\} \\
&= \min\left\{\sum_{t=0}^{k} c[t]s[t], a\right\}, \qquad (48)
\end{aligned}$$

where the last step follows the fact that $c[k]s[k] \geq 0$ and $\left(a - \sum_{j=0}^{k-1} c[j]s[j]\right)^+ \geq 0$.

(2) If $a < \sum_{t=0}^{k-1} c[t]s[t]$, then

$$\begin{aligned}
(47) &= \min\left\{\sum_{t=0}^{k} c[t]s[t], a + c[k]s[k], \sum_{t=0}^{k-1} c[t]s[t], a\right\} \\
&= \min\left\{\sum_{t=0}^{k} c[t]s[t], a\right\}. \qquad (49)
\end{aligned}$$

Thus, by induction, we have the desired result.

Appendix C
Proof for Theorem 6

Consider the Lyapunov function $V(\mathbf{X}) \triangleq \sum_{l=1}^{L} h(X_l)$, where $h'(x) = f(x)$. Then, by using a similar argument to the proof of Lemma 1 in [27] (also see [28]), it is not hard to show that if for any process $\mathbf{A} \in \Lambda_1(\boldsymbol{\rho}, \mathbf{C})$, there exists $\gamma > 0$ and $H \geq 0$ such that

$$\Delta V(\mathbf{X}) \triangleq \sum_{l=1}^{L} \mathbb{E}\left[f(X_l)(D_l[kT] - I_l[kT]) | \mathbf{X}[kT] = \mathbf{X}\right] \leq -\gamma \sum_{l=1}^{L} f(X_l) + H, \qquad (50)$$

then, the system is f-stable, which implies that the system is mean rate stable by Theorem 1. Next, we will show inequality (50) to complete the proof. By substituting the expression of $D_l[kT]$ (see the discussion before (??)) into $\Delta V(\mathbf{X})$, we have

$$\Delta V(\mathbf{X}) = \underbrace{\sum_{l=1}^{L} \mathbb{E}\left[f(X_l)(A_l[kT] - I_l[kT])|\mathbf{X}[kT] = \mathbf{X}\right]}_{\triangleq \Delta V_1(\mathbf{X})} - \underbrace{\mathbb{E}\left[\sum_{l=1}^{L} f(X_l) \min\left\{\sum_{t=kT}^{(k+1)T-1} C_l[kT] S_l^F[t], A_l[kT]\right\}\middle|\mathbf{X}[kT] = \mathbf{X}\right]}_{\triangleq \Delta V_2(\mathbf{X})},$$

where $\mathbf{S}^F[t] = (S_l^F[t])_{l=1}^{L}$ denotes the schedule chosen by FCSMA algorithm at time t. Let

$$W_l[t] = f(X_l[kT]) \min\left\{C_l[kT], \left(A_l[kT] - C_l[kT]\sum_{j=kT}^{t-1} S_l[j]\right)^+\right\}, \forall t = kT, kT+1, ..., (k+1)T-1,$$

where $\mathbf{S}[j] = (S_l[j])_{l=1}^{L}$ is a feasible schedule. Let $W^G[t]$ be the weight of link picked by the Greedy Algorithm with CSI at time slot t. Recall that $W^G[t] = \max_l W_l[t]$. Next, we will derive an upper bound for $\Delta V_1(\mathbf{X})$ by using Lemma 4 and give a lower bound for $\Delta V_2(\mathbf{X})$.

First, let's focus on ΔV_1. By Lemma 4, there exist non-negative numbers $\alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1})$ satisfying (15) and for a $\delta > 0$ small enough, we have

$$\lambda_l(1-\rho_l) \leq \sum_{\mathbf{a}} P_{\mathbf{A}}(\mathbf{a}) \sum_{\mathbf{c}} P_{\mathbf{C}}(\mathbf{c}) \sum_{\mathbf{s}^0,\mathbf{s}^1,...,\mathbf{s}^{T-1}\in\mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1}) \min\left\{\sum_{j=0}^{T-1} c_l s_l^j, a_l\right\} - \delta. \tag{51}$$

By using (51), we have

$$\begin{aligned}
\Delta V_1 &= \sum_{l=1}^{L} f(X_l)\lambda_l(1-\rho_l) \\
&\leq \sum_{\mathbf{a}} P_{\mathbf{A}}(\mathbf{a}) \sum_{\mathbf{c}} P_{\mathbf{C}}(\mathbf{c}) \sum_{\mathbf{s}^0,\mathbf{s}^1,...,\mathbf{s}^{T-1}\in\mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1}) \sum_{l=1}^{L} f(X_l) \min\left\{\sum_{j=0}^{T-1} c_l s_l^j, a_l\right\} - \delta \sum_{l=1}^{L} f(X_l) \\
&\overset{(a)}{\leq} \sum_{\mathbf{a}} P_{\mathbf{A}}(\mathbf{a}) \sum_{\mathbf{c}} P_{\mathbf{C}}(\mathbf{c}) \sum_{\mathbf{s}^0,\mathbf{s}^1,...,\mathbf{s}^{T-1}\in\mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1}) \sum_{t=kT}^{(k+1)T-1} W^G[t] - \delta \sum_{l=1}^{L} f(X_l), \\
&\overset{(b)}{\leq} \sum_{\mathbf{a}} P_{\mathbf{A}}(\mathbf{a}) \sum_{\mathbf{c}} P_{\mathbf{C}}(\mathbf{c}) \sum_{t=kT}^{(k+1)T-1} W^G[t] - \delta \sum_{l=1}^{L} f(X_l),
\end{aligned} \tag{52}$$

where (a) follows from Theorem 5 that the Greedy Algorithm with CSI maximizes $\sum_{l=1}^{L} f(X_l) \min\left\{\sum_{j=0}^{T-1} c_l s_l^j, a_l\right\}$ for any feasible schedules $\mathbf{s}^0$, $\mathbf{s}^1$,...,$\mathbf{s}^{T-1}$, given virtual queue lengths, channel state information and arrivals; (b) switches the order of $\sum_{t=kT}^{(k+1)T-1} W^G[t]$ and $\sum_{\mathbf{s}^0,\mathbf{s}^1,...,\mathbf{s}^{T-1}\in\mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1})$, and uses the fact that $\sum_{\mathbf{s}^0,\mathbf{s}^1,...,\mathbf{s}^{T-1}\in\mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}^0, \mathbf{s}^1, ..., \mathbf{s}^{T-1}) = 1$.

Thus, we have

$$\begin{aligned}\Delta V_1 \leq & \mathbb{E}\left[\sum_{t=kT}^{(k+1)T-1} W^G[t] \middle| \mathbf{X}[kT] = \mathbf{X}\right] - \delta \sum_{l=1}^{L} g(X_l) \\ = & \mathbb{E}\left[\mathbb{E}\left[\sum_{t=kT}^{(k+1)T-1} W^G[t] \middle| \mathbf{X}[kT], \mathbf{A}[kT], \mathbf{C}[kT]\right] \middle| \mathbf{X}[kT]\right] - \delta \sum_{l=1}^{L} g(X_l). \end{aligned} \tag{53}$$

Note that $W^G[t]$ is non-increasing in t within each frame, since the number of remaining packets cannot increase as t increases. Pick any $\overline{W} > 0$ and let

$$\begin{aligned} &\mathcal{F}_0 \triangleq \{W^G[kT] \leq \overline{W}, W^G[kT+1] \leq \overline{W}, ..., W^G[(k+1)T-1] \leq \overline{W}\}; \\ &\mathcal{F}_j \triangleq \{W^G[kT+j-1] > \overline{W}, W^G[kT+j] \leq \overline{W}\}, \forall j = 1, 2, ..., T-1; \\ &\mathcal{F}_T \triangleq \{W^G[(k+1)T-1] > \overline{W}\}, \end{aligned} \tag{54}$$

where $\mathcal{F}_j$ corresponds to the event where the weight chosen by Greedy Algorithm is greater than $\overline{W}$ in the first j slots in frame k. Thus, $(\mathcal{F}_j)_{j=0}^{T}$ forms a partition of a set $\{W^G[kT], W^G[kT+1], ..., W^G[(k+1)T-1]\}$. Then, we have

$$\begin{aligned} &\mathbb{E}\left[\sum_{t=kT}^{(k+1)T-1} W^G[t] \middle| \mathbf{X}[kT], \mathbf{A}[kT], \mathbf{C}[kT]\right] = \mathbb{E}\left[\sum_{j=0}^{T} \sum_{t=kT}^{(k+1)T-1} W^G[t] \mathbb{1}_{\mathcal{F}_j} \middle| \mathbf{X}[kT], \mathbf{A}[kT], \mathbf{C}[kT]\right] \\ \leq & \mathbb{E}\left[\sum_{j=0}^{T} \left(\sum_{t=kT}^{kT+j-1} W^G[t] \mathbb{1}_{\mathcal{F}_j} + (T-j)\overline{W}\right) \middle| \mathbf{X}[kT], \mathbf{A}[kT], \mathbf{C}[kT]\right] \\ = & \mathbb{E}\left[\sum_{j=1}^{T} \sum_{t=kT}^{kT+j-1} W^G[t] \mathbb{1}_{\mathcal{F}_j} \middle| \mathbf{X}[kT], \mathbf{A}[kT], \mathbf{C}[kT]\right] + \frac{T(T+1)\overline{W}}{2}. \end{aligned} \tag{55}$$

Thus, ΔV_1 becomes

$$\Delta V_1 \leq \mathbb{E}\left[\sum_{j=1}^{T} \sum_{t=kT}^{kT+j-1} W^G[t] \mathbb{1}_{\mathcal{F}_j} \middle| \mathbf{X}[kT] = \mathbf{X}\right] + \frac{T(T+1)\overline{W}}{2} - \delta \sum_{l=1}^{L} g(X_l). \tag{56}$$

Second, let's consider ΔV_2. Let

$$W_l^F[t] = f(X_l[kT]) \min \left\{ C_l[kT], \left(A_l[kT] - C_l[kT] \sum_{j=kT}^{t-1} S_l^F[j] \right)^+ \right\}. \tag{57}$$

Then, by using key identity (18) and switching the summations over l and t, we have

$$\Delta V_2 = \mathbb{E}\left[\sum_{t=kT}^{(k+1)T-1} \sum_{l=1}^{L} W_l^F[t] S_l^F[t] \middle| \mathbf{X}[kT] = \mathbf{X}\right]. \tag{58}$$

Let $\epsilon > 0$ and $\zeta > 0$. For each event $\mathcal{F}_j, \forall j = 1, 2, ..., T$, we have $W^G[kT] > \overline{W}, ..., W^G[kT+j-1] > \overline{W}$.

By using Lemma 1, we obtain that for any $\zeta' > 0$, choose $\overline{W}$ such that

$$\Pr\left\{\sum_{l=1}^{L} W_l^F[t]S_l^F[t] \geq (1-\epsilon)W^G[t]\middle|\mathcal{F}_j\right\} \geq 1-\zeta', \forall t = kT, kT+1, ..., kT+j-1. \tag{59}$$

Hence, we have

$$\begin{aligned}
&\Pr\left\{\sum_{t=kT}^{kT+j-1}\sum_{l=1}^{L} W_l^F[t]S_l^F[t] \geq (1-\epsilon)\sum_{t=kT}^{kT+j-1} W^G[t]\middle|\mathcal{F}_j\right\} \\
&\geq \Pr\left\{\sum_{l=1}^{L} W_l^F[t]S_l^F[t] \geq (1-\epsilon)W^G[t], \forall t = kT, kT+1, ..., (k+1)T+j-1\middle|\mathcal{F}_j\right\} \\
&\geq 1-j\zeta' \geq 1-T\zeta',
\end{aligned} \tag{60}$$

where we use the fact that given any two events $\mathcal{E}_1$ and $\mathcal{E}_2$ such that $\Pr\{\mathcal{E}_1\} \geq 1-\epsilon_1$ and $\Pr\{\mathcal{E}_2\} \geq 1-\epsilon_2$, we have $\Pr\{\mathcal{E}_1 \bigcap \mathcal{E}_2\} \geq 1-\epsilon_1-\epsilon_2$. By picking ζ' small enough such that $1-T\zeta' > 1-\zeta$, we have

$$\Pr\left\{\sum_{t=kT}^{kT+j-1}\sum_{l=1}^{L} W_l^F[t]S_l^F[t] \geq (1-\epsilon)\sum_{t=kT}^{kT+j-1} W^G[t]\middle|\mathcal{F}_j\right\} \geq 1-\zeta, \forall j = 1, ..., T, \tag{61}$$

which implies that

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=kT}^{kT+j-1}\sum_{l=1}^{L} W_l^F[t]S_l^F[t]\mathbb{1}_{\mathcal{F}_j}\middle|\mathbf{X}[kT]=\mathbf{X}\right] = \Pr\{\mathcal{F}_j\}\mathbb{E}\left[\sum_{t=kT}^{kT+j-1}\sum_{l=1}^{L} W_l^F[t]S_l^F[t]\middle|\mathbf{X}[kT]=\mathbf{X},\mathcal{F}_j\right] \\
&\geq \Pr\{\mathcal{F}_j\}(1-\epsilon)(1-\zeta)\mathbb{E}\left[\sum_{t=kT}^{kT+j-1} W^G[t]\middle|\mathbf{X}[kT]=\mathbf{X},\mathcal{F}_j\right] \\
&= (1-\epsilon)(1-\zeta)\mathbb{E}\left[\sum_{t=kT}^{kT+j-1} W^G[t]\mathbb{1}_{\mathcal{F}_j}\middle|\mathbf{X}[kT]=\mathbf{X}\right], \forall j = 1, ..., T.
\end{aligned} \tag{62}$$

Thus, we have

$$\begin{aligned}
\Delta V_2 &= \mathbb{E}\left[\sum_{j=0}^{T}\sum_{t=kT}^{(k+1)T-1}\sum_{l=1}^{L} W_l^F[t]S_l^F[t]\mathbb{1}_{\mathcal{F}_j}\middle|\mathbf{X}[kT]=\mathbf{X}\right] \\
&\geq \mathbb{E}\left[\sum_{j=1}^{T}\sum_{t=kT}^{kT+j-1}\sum_{l=1}^{L} W_l^F[t]S_l^F[t]\mathbb{1}_{\mathcal{F}_j}\middle|\mathbf{X}[kT]=\mathbf{X}\right] \\
&\geq (1-\epsilon)(1-\zeta)\mathbb{E}\left[\sum_{j=1}^{T}\sum_{t=kT}^{kT+j-1} W^G[t]\mathbb{1}_{\mathcal{F}_j}\middle|\mathbf{X}[kT]=\mathbf{X}\right]
\end{aligned} \tag{63}$$

Thus, by using (56) and (63), ΔV becomes

$$\Delta V \leq (\epsilon+\zeta-\epsilon\zeta)\mathbb{E}\left[\sum_{j=1}^{T}\sum_{t=kT}^{kT+j-1} W^G[t]\mathbb{1}_{\mathcal{F}_j}\middle|\mathbf{X}[kT]\right] - \delta\sum_{l=1}^{L} g(X_l) + \frac{T(T+1)\overline{W}}{2}. \tag{64}$$

Since

$$\mathbb{E}\left[\sum_{j=1}^{T}\sum_{t=kT}^{kT+j-1}W^G[t]\mathbb{1}_{\mathcal{F}_j}\middle|\mathbf{X}[kT]\right]\leq\mathbb{E}\left[\sum_{t=kT}^{(k+1)T-1}W^G[t]\middle|\mathbf{X}[kT]\right]\leq A_{\max}\sum_{l=1}^{L}g(X_l),\tag{65}$$

we have

$$\Delta V\leq(\epsilon+\zeta-\epsilon\zeta)A_{\max}\sum_{l=1}^{L}g(X_l)-\delta\sum_{l=1}^{L}g(X_l)+\frac{T(T+1)\overline{W}}{2}=-\gamma\sum_{l=1}^{L}g(X_l)+H,\tag{66}$$

where $H=\frac{T(T+1)\overline{W}}{2}$ and $\gamma=\delta-A_{\max}(\epsilon+\zeta-\epsilon\zeta)$. We can choose β,ϵ,ζ small enough such that $\gamma>0$.

References

[1] B. Li and A. Eryilmaz. A fast-csma algorithm for deadline-constrained scheduling over wireless fading channels. In *Proc. International Workshop on Resource Allocation and Cooperation in Wireless Networks (RAWNET/WNC3)*, Princeton, New Jersey, May 2011.

[2] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policiesfor maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 36(12):1936–1948, 1992.

[3] L. Tassiulas and A. Ephremides. Dynamic server allocation to parallel queueswith randomly varying connectivity. *IEEE Transactions on Information Theory*, 39(2):466–478, 1993.

[4] A. Eryilmaz, R. Srikant, and J. Perkins. Stable scheduling policies for fading wireless channels. *IEEE/ACM Transactions on Networking*, 13(2):411–424, 2005.

[5] M. J. Neely, E. Modiano, and C. E. Rohrs. Dynamic power allocation and routing for time varying wireless networks. In *Proc. IEEE International Conference on Computer Communications. (INFOCOM)*, San Francisco, California, April 2003.

[6] S. Shakkottai and A. Stolyar. Scheduling for multiple flows sharing atime-varying channel: The exponential rule. *Translations of the AMS, Series 2,* Avolume in memory of F. Karpelevich, 207:185–202, 2002.

[7] A. Eryilmaz, A. Ozdaglar, D. Shah, and E. Modiano. Distributed cross-layer algorithms for the optimal control of multihop wireless networks. *IEEE/ACM Transactions on Networking*, 18(2):638–651, 2010.

[8] B. Li and A. Eryilmaz. On the limitation of randomization for queue-length-based scheduling in wireless networks. In *Proc. IEEE International Conference on Computer Communications. (INFOCOM)*, Shanghai, China, April 2011.

[9] L. Jiang and J. Walrand. A distributed csma algorithm for throughput and utility maximization in wireless networks. *IEEE Transactions on Networking*, 18(3):960–972, June 2010.

[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 International Conference on Computer Communications. (INFOCOM)*, mini-Conference, San Diego, CA, March 2010.

[11] J. Ghaderi and R. Srikant. On the design of efficient csma algorithms for wireless networks. In *Proc. IEEE International Conference on Decision and Control. (CDC)*, Atlanta, GA, December 2010.

[12] S. Rajagopalan, D. Shah, and J. Shin. Network adiabatic theorem: an efficient randomized protocol for contention resolution. In *Proc. IEEE International Joint Conference on Measurement and Modeling of Computer Systems. (SIGMETRICS)*, Seattle, WA, June 2009.

[13] D. Shah, D. Tse, and J. Tsitsiklis. Hardness of low delay network scheduling. *IEEE Transactions on Information Theory*, 57(12):7810–7817, December 2011.

[14] M. Lotfinezhad and P. Marbach. Throughput-optimal random access with order-optimal delay. In *Proc. IEEE International Conference on Computer Communications. (INFOCOM)*, Shanghai, China, April 2011.

[15] L. Tassiulas. Scheduling and performance limits of networks with constantly varyingtopology. *IEEE Transactions on Information Theory*, 43:1067–1073, May 1997.

[16] I. Hou, V. Borkar, and P. R. Kumar. A theory of qos for wireless. In *Proc. IEEE International Conference on Computer Communications. (INFOCOM)*, Rio de Janeiro, Brazil, April 2009.

[17] I. Hou and P. R. Kumar. Scheduling heterogeneous real-time traffic over fading wireless channels. In *Proc. IEEE International Conference on Computer Communications. (INFOCOM)*, San Diego, CA, March 2010.

[18] J. Jaramillo and R. Srikant. Optimal scheduling for fair resource allocation in ad hoc networks with elastic and inelastic traffic. In *Proc. IEEE International Conference on Computer Communications. (INFOCOM)*, San Diego, CA, March 2010.

[19] H. Gangammanavar and A. Eryilmaz. Dynamic coding and rate-control for serving deadline-constrained traffic over fading channels. In *Proc. IEEE International Symposium on Information Theory. (ISIT)*, Austin, TX, June 2010.

[20] M. Neely. *Stochastic Network Optimization with Aapplication to Communication and Queueing Systems*. Morgan & Claypool, 2010.

[21] X. Lin and N. Shroff. The impact of imperfect scheduling on cross-layer rate control in multihop wireless networks. In *Proc. IEEE International Conference on Computer Communications. (INFOCOM)*, Miami, Florida, March 2005.

[22] P. Marbach, A. Eryilmaz, and A. Ozdaglar. Achievable rate region of csma schedulers in wirelessnetworks with primary interference constraints. In *Proc. IEEE Conference on Decision and Control (CDC)*, New Orleans, Louisiana, December 2007.

[23] S. Rajagopalan and D. Shah. Distributed algorithm and reversible network. In *Proc. IEEE Conference on Information Sciences and Systems (CISS)*, Princeton, New Jersey, March 2008.

[24] L. Tassiulas. Linear complexity algorithms for maximum throughput in radio networksand input queued switches. In *Proc. IEEE International Conference on Computer Communications. (INFOCOM)*, San Francisco, California, April 1998.

[25] J. Ni and R. Srikant. Distributed csma/ca algorithms for achieving maximum throughput inwireless networks. In *Proc. IEEE International Workshop on Information Theory and Applications (ITA)*, San Diego, California, February 2009.

[26] D. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific; 3rd edition, 2007.

[27] B. Li and A. Eryilmaz. On the boundaries of randomization for throughput-optimal scheduling in wireless networks. In *Proc. Allerton Conference on Communication, Control, and Computing (Allerton)*, Monticello, Illinois, September 2010.

[28] B. Li and A. Eryilmaz. Exploring the throughput boundaries of randomized schedulers in wireless networks. *to appear in IEEE/ACM Transactions on Networking*, 2012.