Maximizing Utility via Random Access Without Message Passing

Jiaping Liu†, Yung Yi∗, Alexandre Proutière‡, Mung Chiang†, and H. Vincent Poor† [1]

September 2008

Technical Report
MSR-TR-2008-128

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
http://www.research.microsoft.com

[1] †: Department of Electrical Engineering, Princeton University, USA. Email: {jiapingl,chiangm,poor}@princeton.edu. ∗: Department of Electrical Engineering and Computer Science, KAIST, South Korea. Email: yiyung@ee.kaist.ac.kr. ‡: Microsoft Research, Cambridge, UK. Email: alexandre.proutiere@microsoft.com.

abstract It has been an intensively sought-after goal to achieve high throughput and fairness in wireless scheduling through simple and distributed algorithms. Many recent papers on the topic have relied on various types of message passing among the nodes. The following question remains open: can scheduling without any message passing guarantee throughput-optimality and fairness? Over the last year, it has been suggested in three papers [1]–[3] that random access without message passing may be designed and proved to be optimal in terms of throughput and utility. In this paper, we first extend the algorithm in [2] and provide a rigorous proof of utility-optimality for random access without message passing for Poisson clock model. Then we turn to the more difficult discrete contention and backoff model with collisions, study its optimality properties, and control a tradeoff between long-term efficiency and short-term fairness that emerges in this model.

I. Introduction

There have been growing interests in the design of scheduling algorithms which efficiently and fairly exploit the radio resources in wireless networks in recent years. In their seminal work [4], Tassiulas and Ephremides developed a centralized scheduling algorithm, the Max-Weight scheduler, achieving throughput optimality. The traffic scenario considered in [4] is that of infinite buffers fed by exogenous random packet arrivals with fixed rates, and being throughput optimal means that the proposed algorithm achieves the maximum stability region. In this paper, we are interested in a different traffic scenario, more appropriate to represent the elasticity of traffic in data networks. We considered saturated users (i.e., with fully back-logged buffers), who perceive performance as a function of the average service rate, and the problem is to design a scheduling algorithm that achieves the desired trade-off between efficiency and fairness. Specifically, the proposed scheduling algorithm aims at maximize the sum of user utilities, where the utility is a non-decreasing and concave function of the user average service rate.

This optimization problem has received a lot of attention recently, for it appears not only as a MAC layer problem but also in joint rate control and scheduling through dual decomposition, e.g., in [5]–[8]. There has been a long series of work on distributed scheduling, indeed too long to list here, involving randomization, maximal weight matching, and random access with message passing. They usually require some information of the queues to be passed around among the nodes, e.g., in [9]–[14]. These signaling overhead reduces the effective throughput and makes the algorithms not fully distributed. This naturally leads to the following question that turns out to be very challenging: what about the performance of random access algorithms without any message passing?

In this paper, we focus on such algorithms. In recent papers, it has been demonstrated that such algorithms could also achieve strong throughput performance. For example, in [1], [15], [16], it has been shown that non-adaptive CSMA (Carrier Sense Multiple Access) algorithms, where each link accesses the channel with a fixed probability, are able to provide average throughputs close to throughput-optimality. Turning to random access with adaptive channel access rate, [17] first proposed a simulated-annealing based approach to distributed scheduling. Similar idea has been developed this year in two papers at similar time: in [3], Rajagopalan and Shah suggested that users can adapt their access channel rate depending on their buffer size, so that the system dynamics under the random CSMA algorithm solves the Max-Weight problem. As discussed in [18], one issue with this approach is that when the buffer of a given user becomes large, its channel access rate should also become large. Consequently, to ensure buffer stability and to control the system behavior for arbitrarily large buffers, one should be able to design a CSMA protocol with arbitrarily large access rates. This is made possible in [3] by implementing idealized CSMA algorithms, where Poisson clocks are used to control the channel accesses, and to ensure zero collisions. By simply limiting the virtual buffer sizes, the problem of large buffers and stability in the implementation of the simulated annealing technique may be avoided. In [2], Jiang and Walrand also use the idea of simulated annealing technique as in [3], [17], and they propose an adaptive CSMA algorithm (without message passing) to maximize utility. In this paper, we provide a further detailed analysis of such promising algorithms.

The contributions of this paper are three-fold:

- We first extend the algorithms presented in [2], and provide a generalized framework for random access without message passing with two styles of algorithms that combine the simulated-annealing approach in [17] and the loss network model in [19].
- We develop a rigorous proof of the convergence of these algorithms (such a proof is not presented in [2]). The proof of convergence is conducted by analyzing the behavior of stochastic sub-gradient algorithms modulated by a Markov chain.
- We then turn from the Poisson clock model used by the references above to the more challenging discrete-time contention and backoff model. There, the effect of collisions cannot be ignored and a tradeoff between long-

term efficiency and short-term fairness emerges. We quantify the performance of random access algorithms without message passing, and characterize the tradeoff: short-term fairness increases *exponentially* as efficiency loss decreases.

II. Network Model and Performance Metrics

Consider a wireless network composed by a set $\mathcal{L}$ of L interfering links. Interference is modeled by a boolean matrix $A \in \{0,1\}^{L\times L}$, where $A_{lk} = 1$ if and only if link l interferes link k. For simplicity we assume that A is a symmetric matrix[1]. Define by $\mathcal{N} \subset \{0,1\}^L$ the set of feasible link activation profiles or schedules as follows. A schedule $m \in \mathcal{N}$ is defined such that $m_l = 1$ if link l is active, or $m_l = 0$ otherwise; in addition, for all links k,l, $A_{kl} = 1$ implies $m_l \times m_k = 0$. The transmitter of link l is assumed to transmit at a fixed unit rate when active, and all links are saturated in the sense that the buffers of the corresponding transmitters never empty. A scheduling scheme determines which links are activated at each time. We restrict our analysis to ergodic scheduling schemes, and denote by $\gamma = (\gamma_l, l \in \mathcal{L})$ the long-term throughputs of the various links under the scheme considered.

Efficiency. Let Γ be the set of vectors $\gamma = (\gamma_l, l \in \mathcal{L})$ representing feasible throughputs of the various links. Further define $\Upsilon = \{\boldsymbol{\tau} = (\tau_m, m \in \mathcal{N}), \forall m, \tau_m \geq 0, \sum_{m\in\mathcal{N}} \tau_m = 1\}$. Then, we have:

$$\Gamma = \left\{ \gamma : \exists \boldsymbol{\tau} \in \Upsilon, \forall l \in \mathcal{L}, \gamma_l \leq \sum_{m\in\mathcal{N}:m_l=1} \tau_m \right\}.$$

Γ is referred to as the *maximal saturation throughput region*. It is a convex, coordinate convex set[2], whose boundary $\partial\Gamma$ can be represented using the set $\mathcal{M}$ of maximal schedules. A maximal schedule is a set of non-interfering links such that it is impossible to add a new link to this set without creating interference[3]. We have:

$$\partial\Gamma = \left\{ \gamma : \exists \boldsymbol{\tau} \in \Upsilon, \forall l, \gamma_l = \sum_{m\in\mathcal{M}:l\in m} \tau_m \right\}.$$

A scheduling scheme is said to be efficient if the resulting throughput vector γ is Pareto-efficient with respect to Γ, i.e., $\gamma \in \partial\Gamma$.

Long-term fairness. To quantify long-term fairness, we use the notion of utility explored by Kelly [20]. Let $U : \mathbb{R}^+ \to \mathbb{R}$ be a increasing and strictly concave function. We say that a scheduling scheme achieves U-fairness if it maximizes the global system utility $\mathcal{U} = \sum_{l\in\mathcal{L}} U(\gamma_l)$:

$$\max \sum_{l\in\mathcal{L}} U(\gamma_l), \text{ s.t. } \gamma \in \Gamma. \tag{1}$$

Of particular interest is Proportional Fairness, i.e., $U(\cdot) = \log(\cdot)$.

Short-term fairness. Short-term fairness is a notion that aims at quantifying the way interfering links share the channel in time and at a short time-scale. We define the short-term fairness index β of the network as

$$\beta = \frac{1}{\max\{E_l, l \in \mathcal{L}\}},$$

where E_l is the average duration during which link l do not transmit successfully.

III. Random back-off MAC protocol models

We consider random back-off algorithms, and first we assume that these algorithms are not adaptive. As a consequence, the transmitter of each link accesses the radio resource with fixed probability and keeps it for a random duration of fixed mean. We investigate two cases: (i) transmitters may try to access the channel at the instances of Poisson processes, independent across transmitters, with no collisions; (ii) time is slotted, where multiple transmitters may try to access the channel at the beginning of empty slots only, resulting in collisions.

[1]Interference is not necessarily symmetric. However, the need of acknowledgments at the MAC layer induces symmetry.

[2]A set $\mathcal{Y} \subset \mathbb{R}_+^L$ is coordinate-convex if $x \in \mathcal{Y}$ then for all $y \in \mathbb{R}_+^L$ with $y \leq x$, $y \in \mathcal{Y}$.

[3]Formally, a maximal schedule m is a set of links such that for all $l, l' \in m$, $A_{ll'} = 0$, and for all $l'' \notin m$, there exists $l \in m$ such that $A_{ll''} = 1$.

A. Poisson clock model

Here we consider the case where transmitter of link l runs a Poisson clock of rate λ_l. More precisely, the transmission attempts of link l form a modulated Poisson process of intensity λ_l if none of the interfering link is active, and 0 otherwise. With Poisson clocks, collisions do not occur. This model has been considered in [1], [15], [19], but assuming that λ_l/μ_l does not depend on the link l considered.

1) An equivalent loss network: We show that the system dynamics can be modeled as the evolution of the population in a loss network. The set of links of this loss network coincides with the set $\mathcal{L}$ of links of the actual network. Each of these links is of unit capacity. Now a transmission corresponds to an active client on a given route in the loss network. The route corresponding to a transmission on link l is the set $r_l = \{l\}$. Assume that the transmission durations are exponentially distributed, the numbers $\boldsymbol{N}(t) = (N_l(t), l \in \mathcal{L})$ of active clients on the various routes (corresponding to the activity of links in the initial network) is a Markov process whose transition kernel is:

$$\boldsymbol{N}(t) \to \boldsymbol{N}(t) + e_l \quad \text{with rate } \lambda_l \prod_{k:A_{kl}=1} 1_{N_k(t)=0},$$

$$\boldsymbol{N}(t) \to \boldsymbol{N}(t) - e_l \quad \text{with rate } \mu_l 1_{N_l(t)=1},$$

where e_l is L-dimensional vector with 1 in component l, and 0 elsewhere.

For fixed parameters $\boldsymbol{\lambda}, \boldsymbol{\mu}$, this transition kernel is that of a reversible process whose stationary distribution π is insensitive to the distributions of the transmission durations, and given by:

$$\forall n \in \mathcal{N}, \quad \pi(\boldsymbol{\rho}, n) = \frac{\prod_{l\in\mathcal{L}} \rho_l^{n_l}}{\sum_{m\in\mathcal{N}} \prod_{l\in\mathcal{L}} \rho_l^{m_l}}, \tag{2}$$

where $\rho_l = \lambda_l/\mu_l$. From π, we can compute the ergodic throughput achieved on each link. Hence the throughput of link l is a function of $\boldsymbol{\rho} = (\rho_1, \dots, \rho_L)$:

$$\gamma_l(\boldsymbol{\rho}) = \sum_{n\in\mathcal{N}:n_l=1} \pi(\boldsymbol{\rho}, n). \tag{3}$$

In the following, when $n = (n_1, \dots, n_L)$, we use the notation: $\boldsymbol{\rho}^n = \prod_{l\in\mathcal{L}} \rho_l^{n_l}$.

2) Saturation throughput region: We now characterize the throughput region obtained by random access (RA) protocols when we vary the parameter $\boldsymbol{\rho}$. Let Γ_{RA} be the set of vectors $\gamma(\boldsymbol{\rho})$ for all possible $\boldsymbol{\rho} \in \mathbb{R}_L^+$, then we see that Γ_{RA} almost coincides with the maximal saturation throughput region Γ:

Lemma 1:

$$\Gamma_{RA} = \Gamma \setminus \partial\Gamma. \tag{4}$$

Proof. The fact that $\Gamma_{RA} \subset \Gamma$ is straightforward. Γ_{RA} is coordinate-convex. It remains to show that any point of the boundary $\partial\Gamma$ can be seen as a limit of points in Γ_{RA}. Let $\boldsymbol{\gamma} \in \partial\Gamma$. We show that there exists a sequence $(\boldsymbol{\rho}_p, p \in \mathbb{N})$ such that $\boldsymbol{\gamma}(\boldsymbol{\rho}_p)$ converges to $\boldsymbol{\gamma}$. Note that there exists $\boldsymbol{\tau} \in \Upsilon$ such that for all $l \in \mathcal{L}$, $\gamma_l = \sum_{m\in\mathcal{M}:m_l=1} \tau_m$. It is then easy to construct a sequence satisfying the following properties: (i) for all l such that $\gamma_l = 0$, $\rho_{p,l} = 0$; (ii) for l such that $\gamma_l > 0$, $\lim_{p\to\infty} \rho_{p,l} = \infty$; (iii) for all $m, m' \in \mathcal{M}$ such that $\tau_m \tau_{m'} > 0$, $\boldsymbol{\rho}_p^m / \boldsymbol{\rho}_p^{m'} = \tau_m / \tau_{m'}$. Conditions (iii) form a system that admits a solution (in $\rho_{p,l}$). □

3) Maximizing utility: In view of Lemma 1, the following optimization problem,

$$\max \sum_{l\in\mathcal{L}} U(\gamma_l(\boldsymbol{\rho})), \text{ s.t. } 0 \le \rho_l \le \rho^{\max}, \forall l \in \mathcal{L} \tag{5}$$

approximates the convex optimization problem (1) as $\rho^{\max} \to \infty$. Since the vector $\boldsymbol{\gamma}^\star$ as the solution of (5) is Pareto-optimal, it can be approximated by $\boldsymbol{\gamma}(\boldsymbol{\rho}^\star)$, the solution of (5), only when some components of $\boldsymbol{\rho}$ tend to ∞.

Let us characterize the solution of (5) for Proportional Fairness. Define $\mathcal{M}_l = \{m \in \mathcal{M} : m_l = 1\}$.

Proposition 1: Denote by $\gamma^\star(\rho^{\max})$ the solution of (5), and define by $\mathcal{G}$ the set of $\boldsymbol{\rho}$ such that: $\forall l \in \mathcal{L}$,

$$1 + \sum_{l'\neq l} \frac{\sum_{m\in\mathcal{M}_l\cap\mathcal{M}_{l'}} \boldsymbol{\rho}^m}{\sum_{m\in\mathcal{M}_{l'}} \boldsymbol{\rho}^m} = L \times \frac{\sum_{m\in\mathcal{M}_l} \boldsymbol{\rho}^m}{\sum_{m\in\mathcal{M}} \boldsymbol{\rho}^m}. \tag{6}$$

We have:

$$\gamma^\star = \lim_{\rho^{\max}\to\infty} \gamma^\star(\rho^{\max}) = \lim_{|\boldsymbol{\rho}|\to\infty, \boldsymbol{\rho}\in\mathcal{G}} \gamma(\boldsymbol{\rho}).$$

The proof of the above proposition can be easily done using similar arguments as in the proof of Lemma 1, and identifying the Kuhn-Tucker conditions of (5).

Proof. For $\rho^{\max} > 0$, consider the problem

$$\max \sum_{l\in\mathcal{L}} \log(\gamma_l(\boldsymbol{\rho})), \text{ s.t. } 0 \le \rho_l \le \rho^{\max}\ \forall l \in \mathcal{L}, \tag{7}$$

where each $\gamma_l(\boldsymbol{\rho})$ is given in (3). If we denote $\gamma^*(\rho^{\max})$ by the optimal solution of (7), then $\gamma^*(\rho^{\max}) \to \gamma^*_R$ as $\rho^{\max} \to \infty$. Now we write the Lagrangian of (7),

$$L(\boldsymbol{\nu}, \boldsymbol{\rho}) = \left(\sum_{l\in\mathcal{L}} (\log \sum_{m:m_l=1} \boldsymbol{\rho}^m)\right) - L \log G(\boldsymbol{\rho}) + \sum_{l\in\mathcal{L}} \nu_l(\rho^{\max} - \rho_l),$$

where $G(\boldsymbol{\rho}) = \sum_{m\in\mathcal{N}} \boldsymbol{\rho}^m$. Thus by Kuhn-Tucker condition, for every $l' \in \mathcal{L}$,

$$\frac{\partial L}{\partial \rho_{l'}} = \frac{1}{\rho_{l'}} \sum_{l\in\mathcal{L}} \frac{\sum_{m:m_1=1=m_{l'}} \boldsymbol{\rho}^m}{\sum_{m:m_l=1} \boldsymbol{\rho}^m} - \frac{L}{\rho_{l'}} \times \frac{\sum_{m:m_{l'}=1} \boldsymbol{\rho}^m}{\sum_{m\in\mathcal{N}} \boldsymbol{\rho}^m} - \nu^*_{l'} = 0. \tag{8}$$

and

$$\nu^*_{l'}(\rho^{\max} - \rho_{l'}) = 0. \tag{9}$$

Hence when $\rho^{\max} \to \infty$, the equality (8) is equivalent to (6). Note that to obtain this equivalence, we just have to remark that when the ρ_l's should all tend to ∞ when $\rho^{\max} \to \infty$, and hence schedules that are not maximal (not in $\mathcal{M}$) have negligible probability to occur. □

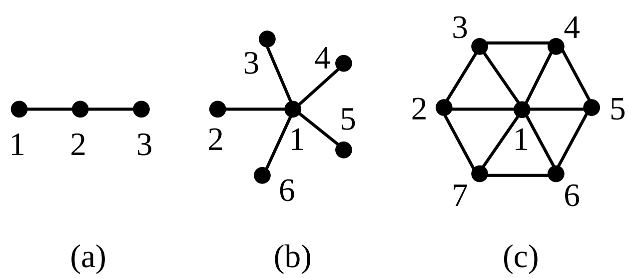


Fig. 1. Network examples showing interference between links: any two interfering links are connected.

4) Toy examples:

Linear networks. Consider a network of L links whose interference graph is depicted in Figure 1(a). The above analysis implies that we realize a proportional fair sharing of the channel letting the ρ_l's tending to ∞, and such that:

$$\begin{array}{ll} \text{if } L = 2k+1, & k\prod_{i=0}^{k} \rho_{2i+1} = (k+1)\prod_{i=1}^{k} \rho_{2i} \\ \text{if } L = 2k, & \prod_{i=1}^{k} \rho_{2i-1} = \prod_{i=1}^{k} \rho_{2i}. \end{array}$$

In the first case ($L = 2k+1$), the throughput are: $\gamma_1 = \ldots = \ldots \gamma_{2k+1} = (k+1)/(2k+1)$ and $\gamma_2 = \ldots = \gamma_{2k} = k/(2k+1)$. In the second case ($L = 2k$), all throughputs are equal to 1/2.

Star networks. Consider now a $(k+1)$-link star network as shown in Figure 1(b). A proportional fair sharing is realized when $k\rho_1 = \rho_l^k$ for all $l \ge 2$. The corresponding throughputs are: $\gamma_1 = 1/(k+1)$ and $\gamma_l = k/(k+1)$ for all $l \ge 2$. For the $(2k+1)$-link star network of Figure 1(c), Proportional Fairness is achieved when $k\rho_1 = \rho_l^k$ for all $l \ge 2$, and the corresponding throughputs are: $\gamma_1 = 1/(2k+1)$ and $\gamma_l = k/(2k+1)$ for all $l \ge 2$.

B. *Discrete-time contention and back-off model*

In real systems, transmitters run discrete back-off algorithms. Time is slotted and the back-off window must be greater than one slot. Collisions are now inevitable. At the beginning of a slot, the source of link l transmits with probability p_l when the channel is sensed idle, and 0 otherwise. The average packet transmission duration (also, referred to as holding time) is $1/\mu_l$ slots. We assume that the duration of a collision is the same as that of a successful transmission (the model can be readily generalized to account for the scenarios where collisions may be shorter, e.g., when using RTS/CTS).

1) The equivalent loss network: As before, for each link in the original system, we add a unit-capacity link in the loss network. The loss network has a set $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2$ of routes.

- Routes in $\mathcal{R}_1$ go through a single link, and there is one of such routes per link. These routes represent successful transmissions, i.e., when a client is active on route $r = \{l\} \in \mathcal{R}_1$, it means that in the original system, the source of link l is successfully transmitting a packet. When a client starts being active on a route of $\mathcal{R}_1$, it remains active for $1/\mu$ slots.
- Routes in $\mathcal{R}_2$ are any set of neighboring links, and represent collisions. To define these routes formally, we introduce the following notation: let $\mathcal{L}' \subset \mathcal{L}$, for $l_1, l_2 \in \mathcal{L}'$, we write $l_1 \overset{\mathcal{L}'}{\bowtie} l_2$ if there exists a sequence $j_0, \dots, j_k$ of links in $\mathcal{L}'$, such that $j_0 = l_1$, $j_k = l_2$ and for all $i = 1, \dots, k$, $A_{j_{i-1} j_i} = 1$. Then,

$$\mathcal{R}_2 = \left\{ r = \mathcal{L}' \subset \mathcal{L} : \forall l_1, l_2 \in \mathcal{L}', l_1 \overset{\mathcal{L}'}{\bowtie} l_2 \right\}.$$

  When there is a client in the loss network on a route $r \in \mathcal{R}_2$ of length k, it means that k links started transmissions simultaneously. When a client starts being active on a route of $\mathcal{R}_2$, it remains active for $1/\mu$ slots.

For the linear network of Figure 1, the corresponding loss network is presented in Figure 2. It has 3 unit-capacity links 1, 2, and 3; $\mathcal{R}_1 = \{\{1\}, \{2\}, \{3\}\}$, $\mathcal{R}_2 = \{\{1, 2\}, \{2, 3\}, \{1, 2, 3\}\}$.

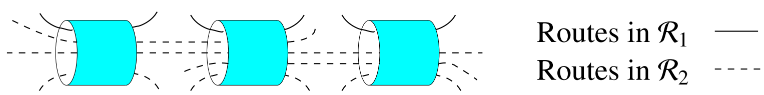


Fig. 2. The loss network corresponding to the linear network of Figure 1(a).

The state of the loss network n belongs to $\{0, 1\}^{|\mathcal{R}|}$. We write $n_r = 0$ when there is no client on route r, and $n_r = 1$ otherwise. In state n, we denote by $\mathcal{L}(n)$ the set of active links in the initial network. Also introduce the function $f(l, n) = \mathbf{1}_{\{A_{kl}=0, \forall k \in \mathcal{L}(n)\}}$. Assume that the packet transmission and collision durations are exponentially distributed, then the loss network state $N(t)$ is a Markov chain whose transition kernel is defined as follows. Consider two feasible states n, n' with:

$$n' = n + \sum_{r \in G(n,n')} e_r - \sum_{r \in D(n,n')} e_r.$$

Note that in the above expression, $G(n, n')$ (resp.) denotes the set of routes that become active (resp. inactive) in transition $n \to n'$ (e.g. $G(n, n') = \{r \in \mathcal{R} : n_r = 0, n'_r = 1\}$). The transition $N(t) = n$ to $N(t + 1) = n'$ occurs with probability $P(n, n')$ with:

$$P(n, n') = P_G(n, n') P_D(n, n') P_E(n, n'),$$

where

$$\begin{aligned}
P_G(n, n') &= \prod_{l \in \mathcal{L}(n') \cap \overline{\mathcal{L}(n)}} p_l f(l, n), \\
P_E(n, n') &= \prod_{l \notin \mathcal{L}(n') \cup \mathcal{L}(n)} ((1 - p_l) f(l, n) + 1 - f(l, n)), \\
P_D(n, n') &= \mu^{|D(n,n')|} (1 - \mu)^{|\mathcal{L}(n) \setminus D(n,n')|}.
\end{aligned}$$

Remark that the expression of the transition kernel is just complicated by the fact that the loss network evolves in discrete time. Note also that the arrival rates in the loss network depend on the network state, so that in general, the network looses its reversibility [21]. Hence, its stationary distribution π can not be explicitely written, but it can easily computed numerically. From the stationary distribution π, we can deduce the throughputs of the various links in the initial system:

$$\gamma_l(p, C, \mu) = C_l \sum_n \pi(n) \left[p_l f(l,n) \prod_{k:A_{kl}=1} ((1-p_k) f(k,n) + 1 - f(k,n)) \right].$$

The above expression is classically obtained using the cycle formula.

2) Approximating the Poisson Clock Model: We see that modelling the network dynamics under discrete back-off algorithms is more complicated than understanding its dynamics under random access algorithms based on Poisson clocks. Fortunately, Poisson clocks can be approximated by discrete back-off algorithms. In fact, we can build a sequence of systems with discrete back-off algorithms converging to any system running under Poisson clocks. Consider a network operating under random MAC algorithms with Poisson clocks of parameters $\rho_l = \lambda_l/\mu_l$ for all links l. Let us design a sequence of systems running discrete back-off algorithms indexed by ϵ. The transmission probability and the average holding time in the system ϵ are:

$$p_l = \epsilon\lambda_l, \quad \mu_l = \epsilon\mu_l.$$

When ϵ tends to 0, we can show that the system behaves as if the transmission attempts for link l were controlled by a Poisson clock of rate λ_l and transmission duration were of mean μ_l. This result will be formalized in Section V, and the penalty in throughput when using discrete back-off algorithms will be quantified.

IV. Poisson Clock Model: Utility-Maximization

In this section, we present two algorithms based on CSMA random protocols that aim at approximately solving (1), both without message passing between nodes, and we formally prove their convergence. These algorithms follow the same principles as those proposed in [2], and are based on a classical dual decomposition of (1) into a source rate control problem and the Max-Weight (MW) scheduling problem. The MW scheduling problem is then solved using on the simulated-annealing technique mentioned in the introduction.

Both algorithms developed here have two interacting components: *(i)* A first component operates in *continuous* time and defines at each instant which links are transmitting. These scheduling decisions are made according to CSMA protocols with Poisson clocks to avoid collisions. If the CSMA transmission parameters are fixed, the set of active links then follow the dynamics of a stochastic loss network as described in Section III-A; *(ii)* A second component operates in *discrete* time, and periodically updates the CSMA transmission parameters used in the first component.

The difference between the two algorithms is as follows:

• *Algorithm 1.* In the first algorithm, we try to approximately solve the MW scheduling using CSMA random access by "freezing" the CSMA transmission parameters for periods referred to as "frames" of duration F slots. The frame size is chosen large enough to let the dynamics of the stochastic loss network capture the behavior of CSMA protocols with fixed parameters to convergence to its stationary regime. The latter regime is made so as to approximately solve the MW scheduling problem. The CSMA parameters are updated at the beginning of each *frame.*

• *Algorithm 2.* The second algorithm aims at, without freezing, adapting the CSMA transmission parameters so as to approximately solve (5). Here the underlying stochastic loss network and the CSMA transmission parameters evolve simultaneously: the algorithm can be seen as a sub-gradient algorithm modulated by a Markov process.

A. Algorithm 1

First consider the dual problem of (5) with Lagrange multipliers $\boldsymbol{\nu} = (\nu_l : l \in \mathcal{L})$:

$$\min_{\boldsymbol{\nu} \geq 0} D(\boldsymbol{\nu}), \tag{10}$$

where

$$D(\boldsymbol{\nu}) = \sum_{l\in\mathcal{L}} B_l(\boldsymbol{\nu}) + H(\boldsymbol{\nu}), \quad (11)$$

$$B_l(\boldsymbol{\nu}) = \max_{\gamma_l}[U_l(\gamma_l) - \nu_l\gamma_l], \quad (12)$$

$$H(\boldsymbol{\nu}) = \max_{m\in\mathcal{N}} \sum_{l\in\mathcal{L}} \nu_l m_l. \quad (13)$$

(5) can be solved by jointly solving the source rate control problem (12) and the MW scheduling problem (13). Let q_l be an intermediate variable representing the dual variable ν_l: $\nu_l = W(q_l)$, where $W(\cdot)$ is a strictly increasing continuous and continuously differentiable function from $\mathbb{R}_+$ to $\mathbb{R}_+$. The role of $W(\cdot)$ defines a different "weight" in the MW scheduling problem. The standard subgradient algorithm to solve the dual problem (10) is to update $q_l[t]$ every slot as:

$$q_l[t+1]=\left[q_l[t]+\frac{b[t]}{W'(q_l[t])}\Big((U')^{-1}(W(q_l[t]))-m_l^\star[t]\Big)\right]_{q^{\min}}^{q^{\max}}, \quad (14)$$

where $\boldsymbol{m}^\star[t]$ is the solution of (13). It was shown in [5]–[7] that the update (14) converges the solution of (5), when for all x, $W(x)=x$. Note that other functions $W(\cdot)$ might have been used, e.g. $\log(1+\cdot)$. Algorithm 1 adds a simulated-annealing technique to solve the MW scheduling problem. More discussions on choices of W function can be found in Section VI.

The variables used by Algorithm 1 are defined in discrete time (i.e., at the beginning of each slot), and are for each link l:

- $\rho_l[t]$, which represents for this link the channel access intensity during slot t in the CSMA protocol, and can be split into the attempt intensity $\lambda_l[t]$ and the channel holding time $1/\mu_l[t]$ as follows: $\lambda_l[t] = g(\rho_l[t])$ and $1/\mu_l[t]=\rho_l[t]/\lambda_l[t]$, for a given function $g(\cdot)$.
- An intermediate variable $q_l[t]$, that will be directly related to $\rho_l[t]$.
- $S_l[t]$, a random variable representing the amount of service handled on link l during slot t. Algorithm 1 further uses the following parameters: $q^{\max}>q^{\min}>0$, $V>0$, $W(\cdot)$, step size $b>0$, and frame size $F\in\mathbb{N}$.

Algorithm 1.

1) During slot t, the network operates under a continuous-time random CSMA protocol as described in III-A, with parameters $\lambda_l[t]$ and $\mu_l[t]$ for all l; each link l then updates its received service $S_l[t]$ during this slot:

$$S_l[t]=\int_t^{t+1} N_l(u)du, \quad (15)$$

where $N_l(u)$ is equal to 1 if link l is active at time u, and 0 otherwise, under the random CSMA protocol of parameters $\lambda_l[t]$ and $\mu_l[t]$ for all l;

2) At the beginning of slot t, if $t=iF$ for $i\in\mathbb{N}$, each link l updates its values of $q_l[t]$ and $\rho_l[t]$ (and the corresponding $\lambda_l[t]$ and $\mu_l[t]$) as:

$$q_l[t]=\left[q_l[t-F]+\frac{b}{W'(q_l[t-F])}\Big(FU'^{-1}(W(q_l[t-F]))-\sum_{j=0}^{F-1}S_l[t+j]\Big)\right]_{q^{\min}}^{q^{\max}}, \quad (16)$$

$$\rho_l[t]=\exp\{VW(q_l[t])\}, \quad (17)$$

where $[\cdot]_x^y=\min(y,\max(\cdot,x))$. If $t\neq iF$ for all $i\in\mathbb{N}$, $q_l[t]$ and $\rho_l[t]$ remain unchanged.

Note that by dividing F at both hand sides in (16), and redefining $q_l[t]$, $q^{\min}$, and $q^{\max}$ as the F-scaled quantify as well as viewing t as a frame index, the dynamics in (16) can be simplified as:

$$q_l[t+1]=\left[q_l[t]+\frac{b}{W'(q_l[t])}\Big((U')^{-1}(W(q_l[t]))-D_l[t]\Big)\right]_{q^{\min}}^{q^{\max}}. \quad (18)$$

where $D_l[t]$ is the *average* service rate of link l during the t-th frame. Now the idea behind Algorithm 1 is that freezing the CSMA parameters for a long period of time (F slots), the average service rate $D_l[t]$ becomes close to the service rate obtained in the MW scheduler, when the parameter V is well chosen. We clarify this in the proof of Theorem 1. We denote by $\bar{\gamma}_l(\tau){=}\frac{1}{\tau}\sum_{t=0}^{\tau-1}S_l[t]$ the average service rate on link l by time τ. We also introduce:

$$G{=}\frac{U'^{-1}(W(q^{\max}))}{2\sup_{q^{\min}\le q\le q^{\max}}W'(q)}.$$

Theorem 1: For any $\delta{>}0$, sufficiently large frame size F and parameter V can be chosen such that:

$$\liminf_{\tau\to\infty}\sum_l U_l(\bar{\gamma}_l(\tau))\ge U^\star_\delta - bG^2,$$

where $U^\star_\delta$ denotes the maximum network utility obtained solving:

$$\max\sum_{l\in\mathcal{L}}U(\gamma_l),\quad \text{s.t. } \gamma\in\Gamma_\delta{=}(1{-}\delta)\Gamma. \tag{19}$$

Proof of Theorem 1. The proof consists in two steps. In the first step, we show that we can choose F and V such that the average service rates, $D_l[t]$, $l{\in}\mathcal{L}$, approximately solve the MW scheduling problem. In the second step, we show that (18) approximately solves (5).
Step 1. The idea of the proof of this step can be found in [3], and here we skip the details. Consider an arbitrary frame t where $q_l[t]=q_l$ for all l. Let $\pi(\boldsymbol{\rho})$ be the stationary distribution of schedules under CSMA protocols of parameters $\boldsymbol{\rho}{=}\exp(VW(q_l))$. Fix $\epsilon{>}0$, and define:

$$K^\epsilon_{\boldsymbol{q}}{=}\left\{m\in\mathcal{N}{:}\sum_{l\in\mathcal{L}}W(q_l)m_l\le(1{-}\epsilon)\sum_{l\in\mathcal{L}}W(q_l)m^\star_l\right\},$$

where $\boldsymbol{m}^\star$ denotes the MW schedule given $\boldsymbol{q}$. The following result, whose proof can be adapted from [3] and presented in Appendix just for completeness, states that we can reduce the stationary probability of $K^\epsilon_{\boldsymbol{q}}$ arbitrarily, by increasing V. For any $\boldsymbol{q}$,

$$\pi(\boldsymbol{\rho},K^\epsilon_{\boldsymbol{q}})\quad\le\quad\frac{L}{\epsilon VW(q^{\min})}. \tag{20}$$

Now, consider the following random quantity: $\sum_{l\in\mathcal{L}}D_l[t]q_l$, where recall that $D_l[t]$ depends on F. Then, we can easily prove that:

$$Pr\left[\sum_{l\in\mathcal{L}}D_l(t)q_l\ge(1{-}\epsilon)\sum_{l\in\mathcal{L}}m^\star_l(t)q_l\right]\ge1-\frac{L}{\epsilon VW(q^{\min})}-\theta(F),$$

where $\theta(F)\to0$, as $F\to\infty$. This is because when the CSMA parameters are fixed, the corresponding loss network is ergodic, and we can find a function $\theta(\cdot)$ uniform in the CSMA parameters (the latter belongs to compact sets due to the limitation of $q^{\min}$ and $q^{\max}$). Let

$$1{-}\delta{=}(1{-}\epsilon)\Big(1-\frac{L}{\epsilon VW(q^{\min})}\Big){-}\theta(F).$$

Then, we deduce that

$$E\left[\sum_{l\in\mathcal{L}}D_l[t]q_l\right]\ge\left((1{-}\epsilon)\Big(1-\frac{L}{\epsilon VW(q^{\min})}\Big){-}\theta(F)\right)\sum_{l\in\mathcal{L}}m^\star_l q_l,$$

which means that we approximately solve (13) on average: $\sum_l W(q_l)D_l\ge(1{-}\delta)\max_{m\in\mathcal{N}}\sum_l W(q_l)m_l$.
Step 2. Step 1 guarantees that at each frame, the algorithm becomes a sub-gradient algorithm of the dual problem of (19). We can then apply the Lyapunov techniques in [7], [22], and conclude the proof of the theorem. Note that the term bG^2 is the penalty we have to pay for having a constant step-size b. □

B. Algorithm 2

Algorithm 2 follows the same principles as Algorithm 1, but here we do not freeze the CSMA transmission parameters, so that the loss network dynamics and the evolution of the CSMA parameters have to be jointly studied. The convergence analysis is complicated. The proof of convergence of Algorithm 2 constitutes one of the main contributions of the paper. We first present a version of Algorithm 2 where the step-size is varying and decreasing, and then we explain how to extend the results when the step-size is fixed but small.

The variables used by Algorithm 2 are the same as those used in Algorithm 1. The algorithm further uses the following parameters: $q^{\max}>q^{\min}>0$, $V>0$, $W(\cdot)$, and step sizes $b:\mathbb{N}\to\mathbb{R}_+$ such that:

$$\sum_{t=0}^{\infty} b[t]=\infty, \quad \sum_{t=0}^{\infty} b[t]^2<\infty. \tag{21}$$

For example, $b[t]=1/(t+1)$, for $t\in\mathbb{N}$.

Algorithm 2.

1) During slot t, the network operates under a continuous-time random CSMA protocol with parameters $\lambda_l[t]$ and $\mu_l[t]$ for all l as in Algorithm 1; each link l then updates its received service $S_l[t]$ during this slot;
2) at the end of slot t, each link l updates its value of $q_l[t]$ as:

$$q_l[t+1]=\left[q_l[t]+\frac{b[t]}{W'(q_l[t])}(U'^{-1}(\frac{W(q_l[t])}{V})-S_l[t])\right]_{q^{\min}}^{q^{\max}}, \tag{22}$$

it also updates $\rho_l[t+1]=\exp\{W(q_l[t+1])\}$, and the corresponding $\lambda_l[t+1]$ and $\mu_l[t+1]$.

The following result states that the above algorithm converges almost surely when t tends to ∞, and furthermore that for appropriate choices of the parameters $q^{\max}$, $q^{\min}$ and V, it approximately maximizes the network utility. In the following, we use the notation $\pi^{\boldsymbol{q}}$ to denote the distribution on $\mathcal{N}$ defined by:

$$\forall m\in\mathcal{N}, \quad \pi^{q}(m)=\frac{\prod_{l:m_l=1}\rho_l}{\sum_{m'\in\mathcal{N}}\prod_{l:m_l=1'}\rho_l}, \tag{23}$$

where $\rho_l=\exp\{W(q_l)\}$ for all l. We also define $\boldsymbol{\rho}(\boldsymbol{q})=(\exp\{W(q_1)\},\cdots,\exp\{W(q_L)\})$.

Theorem 2: For the discrete-time process $\{\boldsymbol{q}[t],t\in\mathbb{N}\}$ with $\boldsymbol{q}[0]=\boldsymbol{q}_0$ for any $\boldsymbol{q}_0\in\mathbb{R}_+^L$, we have

$$\lim_{t\to\infty}\boldsymbol{q}[t]=\boldsymbol{q}^\star, \quad \text{a.s..}$$

Furthermore, the corresponding limiting distribution $\pi^{\boldsymbol{q}^\star}$ and the average throughput $\boldsymbol{\gamma}(\boldsymbol{\rho}(\boldsymbol{q}^\star))$ solve:

$$\begin{aligned} \max \quad & V\sum_{l\in\mathcal{L}}U(\gamma_l)-\sum_{m\in\mathcal{M}}\pi_m\log\pi_m \\ \text{s.t.} \quad & \gamma_l\leq\sum_{m\in\mathcal{N}:m_l=1}\pi_m, \\ & \sum_{m\in\mathcal{N}}\pi_m=1, \end{aligned} \tag{24}$$

(i.e., if $\pi^\star$ and $\boldsymbol{\gamma}^\star$ denote the optimum of (24), then $\pi^{q\star}(m)=\pi_m^\star$, $\gamma_l^\star=\gamma_l(\boldsymbol{\rho}(\boldsymbol{q}^\star))$).

As we will demonstrate in the proof of Theorem 2, the algorithm can be seen as a stochastic sub-gradient algorithm with two interacting time-scales, a first fast time-scale corresponding to the stochastic dynamics of the network under the continuous-time random CSMA protocol, and a second slow time-scale where the parameters $\boldsymbol{\rho}$ of the CSMA protocol are updated. The separation of time-scales is ensured by the property of the decreasing step-size $b[t]$: for t large enough, the updates of the $q_l[t]$'s become smoothed and slow. Then, it will let the network under the CSMA protocol converge, thus the dynamics at the fast time-scale is averaged. Finally, the algorithm, averaged over the fast time-scale, will prove to be the sub-gradient algorithm of the dual problem of (24), which is also noticed in [2]. Compared to [2], we mathematically justify the separation of time-scales, which does not necessarily occur for such multi time-scale algorithms, and provide the proof of convergence.

Proof. The proof of the theorem is in two steps. First we show that the dynamics of the network under the continuous-time random CSMA protocol can be averaged. Then we prove that the resulting averaged algorithm converges to the solution of (24).

Step 1. From the discrete-time sequence $\{\boldsymbol{q}[t]\}_{t=0}^{\infty}$, we define a continuous function $\bar{\boldsymbol{q}}(t)$, $t>0$: define $t_n=\sum_{i=1}^{n}b[i]$, and for all for all $t_n<t\leq t_{n+1}$,

$$\bar{q}_l(t)=q_l[n]+(q_l[n+1]-q_l[n])\times(\frac{t-t_n}{t_{n+1}-t_n}). \tag{25}$$

We have:

Lemma 2: $\{\bar{\boldsymbol{q}}(\tau+\cdot),\tau\geq 0\}$ converges almost surely when $\tau\to\infty$ to the solution of the following system of o.d.e.'s: for all l,

$$\frac{d\tilde{q}_l}{dt}=\left(U'^{-1}(W_l(\tilde{q}_l)/V)-\sum_{m\in\mathcal{N}:m_l=1}\pi^{\tilde{q}}(m)\right)\cdot\frac{\mathbf{1}_{\{q^{\min}\leq\tilde{q}_l\leq q^{\max}\}}}{W'(\tilde{q}_l)}, \tag{26}$$

with $\tilde{\boldsymbol{q}}(\tau)=\bar{\boldsymbol{q}}(\tau)$.

Proof of Lemma 2. Let us attach to each link l a variable $a_l[t]$ such that $a_l[t]=1$ if the link is active at time t (at the end of slot t), and $a_l[t]=0$ otherwise. Now with these additional variables, we may build a Markov chain based on $\boldsymbol{S}[t]$: It can be easily seen that $\boldsymbol{Y}[t]=(\boldsymbol{S}[t],\boldsymbol{a}[t])$ is a non-homogeneous Markov chain whose transition kernel between times t and $t+1$ depends on $\boldsymbol{\rho}[t]$ only. i.e., on $\boldsymbol{q}[t]$ only.

Moreover, if we fix $\boldsymbol{q}[t]=\boldsymbol{q}^0$ for all t and recall that $\{\boldsymbol{N}(t),t>0\}$ is ergodic with stationary distribution π, then by definition of $\boldsymbol{S}[t]$ in (15), $\boldsymbol{Y}[t]$ is ergodic with stationary distribution such that the stationary average of $S_l[t]$ is equal to $\sum_{m\in\mathcal{N}:m_l=1}\pi^{\boldsymbol{q}^0}(m)$. Finally, we observe that the transition kernel of $\boldsymbol{Y}[t]$ is a continuous in $\boldsymbol{q}^0$. Now we rewrite the updates of the q_l's as:

$$q_l[t+1]=q_l[t]+b[t]h(\boldsymbol{q}[t],\boldsymbol{Y}(t)),\ \forall l\in\mathcal{L}. \tag{27}$$

where $h:\mathbb{R}_+^L\times([0,1]^L\times\{0,1\})\to\mathbb{R}_+$ is defined by:

$$h(\boldsymbol{q},\boldsymbol{Y})=\frac{1}{W'(q_l)}(U'^{-1}(W_l(\tilde{q}_l)/V)-S_l))\cdot\mathbf{1}_{\{q^{\min}\leq q_l\leq q^{\max}\}},$$

when $\boldsymbol{Y}=(\boldsymbol{S},\boldsymbol{a})$. h is continuous and Lipschitz in its first argument for $\boldsymbol{q}\in[q^{\min},q^{\max}]^L$, and uniformly Lipschitz in its second argument. Moreover we have $\sup_{t>0}\|\boldsymbol{q}(t)\|<\infty$ a.s.. Hence we can apply the convergence theorem of stochastic approximations on multiple timescales in [23] (Corollary 8, pp. 74), which concludes the proof of the lemma. □

Lemma 2 shows that the interpolation of the discrete-time sequence $\{\boldsymbol{q}[t],t>0\}$ asymptotically approaches the trajectory of a continuous-time o.d.e. limit $\tilde{\boldsymbol{q}}(t)$. Thus, if there exists an equilibrium $\boldsymbol{q}^\star$ such that $\lim_{t\to\infty}\tilde{\boldsymbol{q}}(t)=\boldsymbol{q}^\star$, then we would also have: $\lim_{t\to\infty}\boldsymbol{q}[t]=\boldsymbol{q}^\star$ a.s.

Step 2. To complete the proof of the theorem, we show that (26) may be interpreted as the sub-gradient algorithm (projected on a bounded interval) corresponding to the Lagrange dual problem of (24). Step 2 has been derived in [2], but we present it for completeness. Let $D(\boldsymbol{\nu},\eta)$ denote the dual function of (24), then (26) is the sub-gradient algorithm of the following:

$$\min\ D(\boldsymbol{\nu},\eta),\ \text{s.t.}\ \nu^{\min}\leq\nu_l\leq\nu^{\max},\ \forall l\in\mathcal{L}. \tag{28}$$

Here we include the upper-bound $\nu^{\max}$ that corresponds to the limitation of the q_l's: $\nu^{\max}=W(q^{\max})$ and $\nu^{\min}=W(q^{\min})$. We assume that we are able to choose $q^{\min},q^{\max}$ (and hence $\nu^{\min},nu^{\max}$) appropriately (which we will address at the end of this subsection), so that the solution of the un-constrainted dual problem of (24) is strictly within the interval $[\nu^{\min},\nu^{\max}]$ component-wise. Then (28) has the same solution as the dual of (24).

The Lagrangian of (24) is given as

$$L(\boldsymbol{\gamma},\boldsymbol{\pi};\boldsymbol{\nu},\eta)=(\sum_{l\in\mathcal{L}}V\log\gamma_l-\nu_l\gamma_l)+(\sum_{l\in\mathcal{L}}\nu_l\sum_{m\in\mathcal{N}:m_l=1}\pi_m$$
$$-\sum_{m\in\mathcal{N}}\pi_m\log\pi_m)+\eta(\sum_{m\in\mathcal{N}}\pi_m-1).$$

The Kuhn-Tucker conditions are given by:

$$VU'(\gamma_l)=\nu_l,\ \forall l\in\mathcal{L} \tag{29}$$

$$-1-\log\pi_m+\sum_{l:m_l=1}\nu_l+\eta=0,\ \forall m\in\mathcal{N}. \tag{30}$$

$$\nu_l(\gamma_l-\sum_{m\in\mathcal{N}:m_l=1}\pi_m)=0. \tag{31}$$

Recall that the algorithm, averaged over the fast time-scale, is defined by (26), and by letting the transmission parameters in the CSMA protocol be $\rho_l=\exp\{W(\tilde{q}_l)\}$. Now if $\nu_l=W(\tilde{q}_l)$ for all l, (30) is solved for $\pi^{\tilde{\boldsymbol{q}}}$ (see (23)). Now the sub-gradient of (31) (when accounting for (29)) is:

$$\frac{d\nu_l}{dt}=\left(U'^{-1}(\frac{\nu_l}{V})-\sum_{m\in\mathcal{M}:m_l=1}\pi_m^{\tilde{\boldsymbol{q}}}\right)\cdot\mathbf{1}_{\{0\le\nu_l\le\nu^{\max}\}},$$

which is equivalent to (26). Since (28) is a convex problem, we see that (26) converges to its unique equilibrium $\boldsymbol{\nu}^\star$. Finally by our assumption that $\nu^{\max}$ has been chosen large enough so that $\nu_l^\star$ is strictly within the interval $[\nu^{\min},\nu^{\max}]$ for every $l\in\mathcal{L}$. Then $\boldsymbol{\nu}^\star$ also provides the primal solution $\gamma(\boldsymbol{\rho}(\boldsymbol{q}^\star))$ to the strictly convex problem (24). The proof of Theorem 2 is completed. □

Theorem 2 shows that Algorithm 2 asymptotically solves the optimization problem (24). Recall that our original objective is to provide a solution to (1). If we let $\boldsymbol{\gamma}^\star$ denote the optimum of (1), and $\boldsymbol{\gamma}(\boldsymbol{\rho}(\boldsymbol{q}^\star))$ denote that of (24) as corresponding to the dual optimum $\boldsymbol{q}^\star$, then the gap between the attained utilities is given by [2] (simply bounding the term $\sum_m\pi_m\log\pi_m$):

$$\|\sum_{l\in\mathcal{L}}U(\gamma_l(\boldsymbol{\rho}(\boldsymbol{q}^\star)))-U(\gamma_l^\star)\|\le\log|\mathcal{N}|/V.$$

Algorithm 2 as described in (22) adopts a sequence of time-varying step-sizes $b[t]$, which requires transmitters synchronization. From a practical perspective, we may then need to consider constant, but small, step-size b. Denote by $\boldsymbol{q}^b[t]$ the variable obtained with Algorithm 2 with constant step-size b. We are also able to prove that the modified Algorithm approaches optimality, but in a weaker sense as stated in the following corollary.

Corollary 1 (constant step-size): As $b\to 0$,

$$\limsup_{t\to\infty}\|\bar{\boldsymbol{q}}^b(t)-\boldsymbol{q}^\star\|\to 0 \text{ in probability},$$

where $\boldsymbol{q}^\star$ is defined in Theorem 2 .

Proof. For $b\to 0$, the interpolated trajectory $\bar{\boldsymbol{q}}^b(t)$ (defined similarly as in (25)) converges in distribution to the o.d.e. limit $\tilde{\boldsymbol{q}}(t)$ in (26) uniformly on compact sets (see [23], Theorem 7 of pp. 114) , i.e. for any $T>0$, as $b\to 0$:

$$\sup_{t\in[M,M+T]}\|\bar{\boldsymbol{q}}^b(t)-\tilde{\boldsymbol{q}}(t)\|\to 0 \tag{32}$$

in distribution, uniformly w.r.t. M. Now fix $\epsilon>0$. (32) implies that for all $\delta>0$, there exists b_0 such that for all $b<b_0$, and for all M:

$$P\left[\sup_{t\in[M,M+T]}\|\bar{\boldsymbol{q}}^{b_0}(t)-\tilde{\boldsymbol{q}}(t)\|>\epsilon/2\right]<\delta.$$

From Theorem 2, there exists M_ϵ such that for all $t>M_\epsilon$, $\|\tilde{\boldsymbol{q}}(t)-\boldsymbol{q}^*\|<\epsilon/2$, a.s.. Now we have: for all $M>M_\epsilon$,

$$P\left[\sup_{t\in[M,M+T]}\|\bar{\boldsymbol{q}}^b(t)-\boldsymbol{q}^\star\|>\epsilon\right]<\delta.$$

We obtained the convergence stated in the corollary. □

To conclude this section, we discuss the relation between parameters V and ρ^{min}, $\rho^{\max}$. We have assumed in the proof of Theorem 2 that $\rho^{\min}$, $\rho^{\max}$ are carefully chosen such that the equilibrium of (26) does not stay at either the upper bound or the lower bound. In fact, we also observe that in (29), $\nu^{\max}=\log\rho^{\max}$ where $\rho^{\max}$ is the same as defined in (5), γ_l is upper bounded by a positive constant $\gamma^{\max}$, and also $U'(\cdot)$ is a decreasing mapping by the strict concavity of $U(\cdot)$. Then by (29), we have

$$V \leq \log(\rho^{\max})/U'(\gamma^{\max}). \tag{33}$$

Hence in Algorithm I, we can always choose $\rho^{\min}$ as a sufficiently small constant and V, $\rho^{\max}$ satisfying (33) to guarantee our assumption.

V. Discrete-time Contention and Back-off Model: Impact and Tradeoff

In the previous section, we have presented two algorithms based on CSMA algorithms to approximately maximize utility. These algorithms rely on the use of Poisson clocks in the CSMA algorithms to provide the mathematical convenience. In practice, we have to use discrete-time contention and back-off algorithms and must take into account the impact of collisions. In this section, we investigate its impact on the performance of our algorithms with discrete back-off protocols, and also characterize their trade-off between efficiency and short-term fairness.

A. Impact of collisions on efficiency

Consider discrete back-off algorithms as discussed in Section III-B. The transmitter of link l accesses the channel with probabilities $p_l = \epsilon\lambda_l$ and keeps the channel for a fixed duration $1/\epsilon\mu_l$. Let $\rho_l = \lambda_l/\mu_l$, and to simplify the analysis, we assume $\mu_l = \mu$ for all $l \in \mathcal{L}$, since the design objective is to set the desired ρ_l which can be realized by varying λ_l. For the collisions, we consider two cases: 1) the duration of collisions is equal to $1/\epsilon\mu$. 2) RTS/CTS signaling procedure is used where the collision duration is a small constant, taken equal to one slot here. Let $\pi(\epsilon,\boldsymbol{\rho})$ and $\boldsymbol{\gamma}(\epsilon,\boldsymbol{\rho})$ denote the stationary distribution and average throughput vector, and then using standard approximation techniques of Markov processes, we get the following by Taylor expansion:

Proposition 2: For all $n \in \{0,1\}^L$, for all $l \in \mathcal{L}$:

$$\pi(\epsilon,\boldsymbol{\rho},n) = \pi(\boldsymbol{\rho},n)1_{n\in\mathcal{N}} + O(\epsilon), \tag{34}$$

$$\gamma_l(\epsilon,\boldsymbol{\rho}) = \gamma_l(\boldsymbol{\rho}) - \epsilon F_l(\boldsymbol{\lambda},\boldsymbol{\rho}). \tag{35}$$

Thus (35) can give an estimation of the difference in the optimal average throughput between the Poisson clock model in Section (IV) and the discrete-time contention model. The function $F_l(\boldsymbol{\lambda},\boldsymbol{\rho}) = -\frac{\partial}{\partial\epsilon}\gamma_l|_{\epsilon\to 0}$ depends on the interference properties of the network, and also on whether RTS/CTS is used or not. However, it is in general difficult to derive. For illustrative purposes, we give its explicit expression through following examples.

Example 1. Networks with full interference: Consider a network of L links with full interference, i.e., $A_{kl}=1$ for all k,l. We have $\gamma_l(\boldsymbol{\rho}) = \rho_l/(1+\sum_{k\in\mathcal{L}}\rho_k)$. In absence of RTS/CTS we obtain:

$$F_l(\boldsymbol{\lambda},\boldsymbol{\rho}) = \frac{\rho_l(\sum_{k\neq l}\lambda_k)(\sum_{k\in\mathcal{L}}\rho_k)}{(1+\sum_{k\in\mathcal{L}}\rho_k)^2},$$

which reduces to $F_l(\boldsymbol{\lambda},\boldsymbol{\rho}) = \frac{\lambda L(L-1)\rho^2}{(1+L\rho)^2}$ if for all l, $\rho_l=\rho$ and $\lambda_l=\lambda$. With RTS/CTS we obtain:

$$F_l(\boldsymbol{\lambda},\boldsymbol{\rho}) = \frac{\rho_l\lambda_l + \rho_l\sum_{k\in\mathcal{L}}\rho_k(L-1)(\lambda_l-\lambda_k)}{(1+\sum_{k\in\mathcal{L}}\rho_k)^2},$$

which reduces to $F_l(\boldsymbol{\lambda},\boldsymbol{\rho}) = \frac{\lambda\rho}{(1+L\rho)^2}$ if for all l, $\rho_l=\rho$ and $\lambda_l=\lambda$.

Example 2. A linear network with 3 links: We consider the 3-link linear network as shown in Figure 1(a), where link 2 interferes with link 1 and 3, but link 1 and 3 are able to transmit simultaneously. By symmetry, at the equilibrium we have $\rho_1=\rho_3$, and thus

$$F_1(\boldsymbol{\lambda},\boldsymbol{\rho}) = -\lambda_1\cdot\frac{\rho_1^3(2+\rho_1)}{1+2\rho_1+\rho_1^2+\rho_2},\ F_2(\boldsymbol{\lambda},\boldsymbol{\rho}) = -\lambda_1\cdot\frac{\rho_2(2+\rho_1+\rho_1^2)}{1+2\rho_1+\rho_1^2+\rho_2}.$$

Note that for proportional fairness, we have $2\rho_1=\rho_2^2$ at the equilibrium.

Now to evaluate the global loss in terms of efficiency of of Algorithms 1 and 2 when discrete back-off algorithms are used, we have to consider both the loss due to collisions and the loss due to the sub-optimality of the algorithms with Poisson clock. For Algorithm 2, the total loss of throughput on link l is then upper bounded by:

$$\frac{C_1}{\log\rho^{\max}} + \epsilon|F_l(\boldsymbol{\lambda}^\star,\boldsymbol{\rho}^\star)|.$$

B. Short-term fairness vs. efficiency trade-off

Let us first compute the short-term fairness index. Using cycle formula, at the equilibrium (after the convergence of the algorithm), the average of periods where during which link l do not transmit successfully is given by:

$$E_l = \frac{1}{\epsilon\mu} \times \frac{1-\gamma_l(\boldsymbol{\rho}^\star)}{\gamma_l(\boldsymbol{\rho}^\star)}.$$

Then the short-term fairness index is $\beta = 1/\max_l E_l = C_2 \epsilon\mu$, where C_2 is a constant. We now investigate the relationship between loss of efficiency and short-term fairness: for example, what short-term fairness can we obtain given that we wish to guarantee at least a certain efficiency loss? To simplify again the analysis, we consider the examples we discussed. We believe that similar trade-offs can also be obtained in general networks. For fully-interfered networks, denote by $\boldsymbol{\rho}$ the optimal link access intensity (which turns out to be equal to $\rho^{\max}$ in this trivial example). From simple calculations, we deduce that the guarantee on the efficiency loss for Algorithm 2 with discrete back-off protocol is:

$$\text{without RTS/CTS:} \quad \frac{C_1}{\log\rho} + C_3\rho\beta,$$
$$\text{with RTS/CTS:} \quad \frac{C_1}{\log\rho} + C_4\beta,$$

where C_3 and C_4 are positive constants. To guarantee a small loss of efficiency δ, we need to choose ρ greater than $exp(C_1/\delta)$ in both cases, and a short-term fairness index such that:

$$\text{without RTS/CTS:} \quad \beta \le \frac{\delta}{C_3 \exp(C_1/\delta)},$$
$$\text{with RTS/CTS:} \quad \beta \le \frac{\delta}{C_4}.$$

Similar results can also be applied to the 3-link example provided. As a consequence, *efficiency* has a significant price in terms of short-term fairness for both cases as discussed above. However, when RTS/CTS is used, the impact is less significant in these particular examples. Note also that the efficiency of the algorithms increases very slowly with the access intensity $\boldsymbol{\rho}$, and it might be difficult to approach the utility-optimal regime. As an example, if we allow only a very small loss in efficiency, say, δ=0.001, then we need extremely large holding time that will serve only a limited set of links and starve other links. However, when we have 10% efficiency loss, i.e., δ=0.1, is adopted, a reasonable quantity of holding time would work. We illustrate and verify these findings via simulations in the next section.

VI. Numerical Examples

In this section, we present numerical experiments to illustrate the analytical results derived in the previous sections. We use only Algorithm 2, since Algorithms 1 and 2 have similar dynamics. More importantly, Algorithm 2 is more practical due to the notion of frame (whose size is typically long for high efficiency) in Algorithm 1.

We consider a linear network with 3 links and a star network with 7 links, as shown in Figure 1(a) and Figure 1(c). We focus on the Proportional-Fairness throughout simulations, i.e., $U(\cdot) = \log(\cdot)$. In the linear network, at the Proportional-Fairness, we have that $\gamma_2^\star = 1/3 \approx 0.33$, $\gamma_1^\star = \gamma_3^\star = 2/3 \approx 0.66$ through numerical computation of (1). Similarly, we have $\gamma_1^\star = 1/7 \approx 0.142$, $\gamma_l^\star = 3/7 \approx 0.428$, $2 \le l \le 7$. We run discrete back-off CSMA random access protocol described in Section III-B without RTS/CTS (With RTS/CTS, throughput and thus long-term fairness will even increase due to constant collision duration). We set $p \le p^{\max} = 0.1$ and $1/\mu \le 500$, such that $\rho^{\max} = 50$. The holding time $1/\mu$ is equivalent for all links in the network. We choose V=1 for linear network and V=0.5, guided by the rule in (33). We need a smaller V in the star network, since $\gamma_1^{\max}$ is smaller for the link 1 than that in the linear network.

We vary other parameters such as weight functions and step size to investigate and compare the performance. Our choice of parameters in the above does not have special meaning, and we could observe the similar results for other parameters. We omit their results due to space limitations.

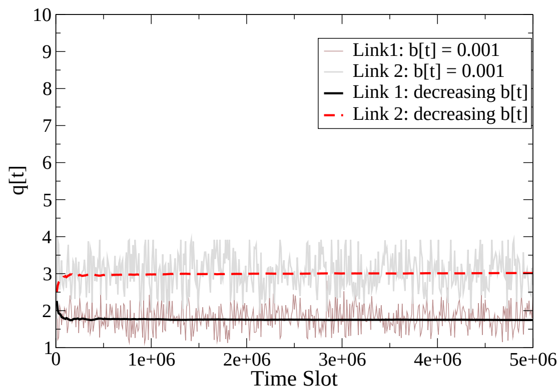
(1) Long-term throughput and fairness. Tables I and II (averaged over 10 experiments) show the long-term throughputs (thus fairness by comparing them to the Proportional-Fair points) for links 1 and 2 for two networks as well as different weight functions. The results for other links are similar due to symmetry. We observe that the results have good match with the Proportional-Fairness with maximum gap of about 10% from optimality. As explained in Section IV, this gap is unavoidable, which depends on $\rho^{\max}$, V, as well as network topologies. We also

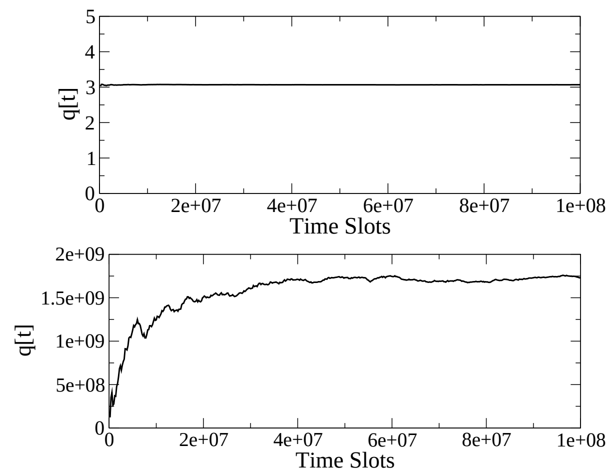
TABLE I
LONG-TERM THROUGHPUT IN LINEAR NETWORK

	$W(x)=x$		$W(x)=\log(x+1)$		$W(x)=\log\log(x+e)$	
	const.	dec.	const.	dec.	const.	dec.
link 1	0.580	0.570	0.582	0.569	0.580	0.567
link 2	0.312	0.328	0.311	0.329	0.311	0.335

TABLE II
LONG-TERM THROUGHPUT IN STAR NETWORK

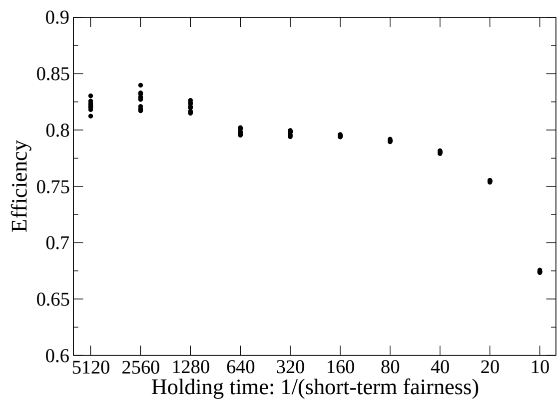
	$W(x)=x$		$W(x)=\log(x+1)$		$W(x)=\log\log(x+e)$	
	const.	dec.	const.	dec.	const.	dec.
link 1	0.097	0.085	0.096	0.091	0.096	0.084
link 2	0.353	0.369	0.352	0.355	0.351	0.352





(a) Traces of q of link 1 in the linear network: for constant step size, $b = 0.001$, and for decreasing step size, $b[t] = 1/(t+1)$.

(b) Convergence rates for weight functions: loglog(x+e) converges faster than x



(c) Tradeoff between efficiency and short-term fairness in the line network

Fig. 3. Simulation Results

see that the weight functions do not have significant impact on long-term behaviors of the algorithm. Figure 3(a) shows example trajectories of $q_1[t]$ and their convergence behaviors in the linear network. With a constant step size $b[t]{=}0.001$, the trajectories are oscillating within some neighborhood of the converged point, whereas with a decreasing step size $b[t]{=}1/(t{+}1)$ the convergence is realized.

(2) Convergence rate vs. weight functions. However, weight functions may have impact on transients, as shown in Figure 3(b), where we have $W(x) = x$ and $W(x) = \log\log(x+e)$. We observe that the trajectories with $W(x) = x$ converge much faster than those with $W(x)=\log\log(x+e)$ for the same decreasing step-size rule, $b[t]=1/(t+1)$. An intuitive explanation is that for the given $\boldsymbol{\rho}^\star$ (which is same for both weight functions), the $q_l^\star$ with $\log\log(x+e)$ is significantly larger than just x based on the equation $\rho_l = \exp(W(q_l))$ (compare the y-axis in Figure 3(b): 3 vs. 10^9!). Thus, it takes more time to track the system and enjoy the averaging effect for convergence.

(3) Efficiency and short-term fairness tradeoff. Figure 3(c) shows tradeoff between efficiency and short-term fairness, where 10 experiments have been made with different random seeds for each value in x-axis. Efficiency and short-term fairness are measured by the sum of long-term throughputs in links 1 and 2, and by holding time, respectively. Note that real short-term fairness index β is inversely proportional to holding time. Holding time in the x-axis is *log-scaled* and rearranged in the decreasing order to see the tradeoff clearly. For the same given $\rho^{\max}=50$, we vary the holding times (thus, $p^{\max}$ should decrease accordingly, such that $\rho^{\max}$ is the same). As analyzed in Section V, we clearly see the tradeoff, because smaller holding time typically leads to larger access probability, which in turn decreases throughput due to collisions in discrete systems. We observe that for about 20% efficiency loss, it is enough to have holding time of 40. However, efficiency gain after holding time of 320 is very minor, which illustrates our analysis that for a small efficiency loss, we need pay a lot of cost of short-term fairness.

VII. Conclusion

Achieving optimality in terms of throughput and fairness has been known to require intelligent, yet complex scheduling mechanisms with heavy message passing. In this paper, we proposed two simple algorithms based on the CSMA random access without message passing, which provably solve a long-standing problem of realizing optimality with "zero complexity." The algorithm development ideas and convergence proof techniques are based on a combination of powerful technique of loss network modeling and simulated annealing for distributed scheduling, both developed about 20 years ago. Algorithms were analyzed in two models: continuous-time Poisson clock model without collisions, and discrete-time contention and backoff model with collisions. In the more practical and less explored discrete setting, we characterized the tradeoff between fairness and efficiency, where we need to pay a significant price of short-term fairness for long-term optimality. As future work, extensions to joint congestion control, routing, and scheduling over multihop wireless networks can be readily carried out.

Appendix

A. Proof of (20)

First, since we work on a fixed $\boldsymbol{p}$ and W, we omit dependency of all notions on $\boldsymbol{p}$ and W in this proof.

From the results in time-reversible loss network, we know that the stationary distribution π_V satisfies the following: for any schedule $m \in \mathcal{N}$,

$$\pi_V(m) \propto \exp(VW(m)),$$

where $W(m)$ is the weight of the schedule m, i.e., $W(m)=\sum_l W(q_l)m_l$. Let us denote by $m*$ the max-weight schedule, and further denote by $\pi^\star$ the distribution concentrating on $m^\star$, i.e., $\pi^\star(m^\star)=1$, and $\pi^\star(\mathcal{N}\backslash\{m^\star\})=0$.

We can easily prove that

$$\pi_V = \arg\sup_{\mu \in \mathcal{D}_N} (E_\mu[VW(m)] + H(\mu)), \tag{36}$$

where $\mathcal{D}_N$ is the set of all distributions on $\mathcal{N}$, and $H(\mu)$ is the entropy of the distribution μ, i.e., $H(\mu) = -\sum_{m\in\mathcal{N}} \mu(m) \log \mu(m)$.

Now, since π_V maximizes (36), we obviously have:

$$H(\pi^\star) + VW(m^\star) \leq H(\pi_V) + VE_{\pi_V}[W(m)], \tag{37}$$

where the LHS is obtained by assigning a particular distribution $\pi^\star$.

Note that:

$$\begin{aligned} E_{\pi_V}[W(m)] &\leq W(m^\star)\pi_V(\mathcal{N}\backslash K^\epsilon) + (1-\epsilon)W(m^\star)\pi_V(K^\epsilon) \\ &= W(m^\star) - \epsilon W(m^\star)\pi_V(K^\epsilon). \end{aligned} \tag{38}$$

From (37) and (38), and also from $H(\pi^\star)=0$,

$$\pi_V(K^\epsilon)\leq\frac{H(\pi_V)}{\epsilon V W(m^\star)}.$$

Finally, the result follows from the fact that $W(m^\star)\geq W(q^{\min})$, and $H(\mu)\leq\log|\mathcal{N}|\leq L$, for any μ.

References

[1] A. Proutiere, Y. Yi, and M. Chiang, "Throughput of random access without message passing," in *Proceedings of CISS*, 2008.

[2] L. Jiang and J. Walrand, "A CSMA distributed algorithm for throughput and utility maximization in wireless networks," UCB, Tech. Rep., 2008.

[3] S. Rajagopalan and D. Shah, "Distributed algorithm and reversible network," in *Proceedings of CISS*, 2008.

[4] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1949, December 1992.

[5] X. Lin and N. B. Shroff, "The impact of imperfect scheduling on cross-layer rate control in wireless networks," in *Proceedings of IEEE Infocom*, 2005.

[6] M. J. Neely, E. Modiano, and C. Li, "Fairness and optimal stochastic control for heterogeneous networks," in *Proceedings of IEEE Infocom*, 2005.

[7] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Joint optimal congestion control, routing, and scheduling in wireless ad hoc networks," in *Proceeding of IEEE Infocom*, 2006.

[8] A. Eryilmaz and R. Srikant, "Joint congestion control, routing, and mac for stability and fairness in wireless networks," *IEEE Journal on Selected Areas of Communication (JSAC), Special Issue on Nonlinear Optimization of Communication Systems*, vol. 24, no. 8, 2006.

[9] L. Tassiulas, "Linear complexity algorithms for maximum throughput in radionetworks and input queued switches," in *Proceedings of Infocom*, 1998.

[10] E. Modiano, D. Shah, and G. Zussman, "Maximizing throughput in wireless networks via gossiping," in *Proceedings of ACM Sigmetrics*, 2006.

[11] A. Eryilmaz, A. Ozdaglar, and E. Modiano, "Polynomial complexity algorithms for full utilization of multi-hop wireless networks," in *Proceedings of Infocom*, 2007.

[12] S. Sanghavi, L. Bui, and R. Srikant, "Distributed link scheduling with constant overhead," in *Proceedings of ACM Sigmetrics*, 2007.

[13] S. Ray and S. Sarkar, "Arbitrary throughput versus complexity tradeoffs in wireless networks using graph partitioning," in *Proceedings of Information Theory and Applications Second Workshop*, 2007.

[14] C. Joo and N. B. Shroff, "Performance of random access scheduling schemes in multi-hop wireless networks," in *Proceedings of Infocom*, 2007.

[15] M. Durvy and P. Thiran, "Packing approach to compare slotted and non-slotted medium access control," in *Proceedings of IEEE Infocom*, 2006.

[16] C. Bordenave, D. McDonald, and A. Proutiere, "Performance of random multi-access algorithms, an asymptotic approach," *Proceedings of ACM Sigmetrics*, 2008.

[17] B. Hajek, "Cooling schedules for optimal annealing," *Mathematics of Operations Research*, vol. 13, no. 2, pp. 311–329, 1988.

[18] Y. Yi, A. Proutiere, and M. Chiang, "Complexity in wireless scheduling: Impact and tradeoffs," in *Proceedings of ACM Mobihoc*, 2008.

[19] F. Kelly, "One-dimensional circuit-switched networks," *Annals of Probability*, vol. 15, pp. 1166–1179, 1987.

[20] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," in *Journal of the Operational Research Society*, vol. 49, 1998.

[21] F. Kelly, "Loss networks," *Annals of Applied Probability*, vol. 1, pp. 319–378, 1991.

[22] M. J. Neely, E. Modiano, and C. Li, "Fairness and optimal stochastic control for heterogeneous networks," in *Proceedings of IEEE Infocom*, 2005.

[23] V. Borkar, *Stochastic Approximation: A Dynamical Systems Viewpoint*, 2008, draft copy.