Convergence Analysis of a Distributed CSMA Algorithm for Maximal Throughput in a General Class of Networks

Libin Jiang
Jean Walrand

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2008-185
http://www.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-185.html

December 29, 2008

Acknowledgement

This research was supported in part by MURI grant BAA 07-036.18

Convergence Analysis of a Distributed CSMA Algorithm for Maximal Throughput in a General Class of Networks

Libin Jiang and Jean Walrand
EECS Department, University of California at Berkeley
{ljiang,wlr}@eecs.berkeley.edu

***Abstract*—In [1], [2], we proposed a family of distributed scheduling and rate control algorithms to achieve the maximal throughput in a general class of networks with interference constraints, as well as approaching the optimal fairness among different data flows. These algorithms were inspired by CSMA (Carrier Sense Multiple Access). In this paper, we analyze and prove the convergence of such a scheduling algorithm and the queue stability it guarantees, with properly-chosen step sizes and adjustment periods. Convergence of other algorithms in [1], [2] can be proved similarly.**

I. Basic Model and Problem statement

For completeness, we first describe the basic model and problem setup as in [2].

A. Network Interference Model

Assume there are K *links* in the network, where each *link* is an (ordered) transmitter-receiver pair. The network is associated with a link contention graph (or "LCG") $G = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of vertexes (each of them represents a link) and $\mathcal{E}$ is the set of edges. Two links cannot transmit at the same time (i.e., "conflict") if and only if there is an edge between them. This is a very general interference model that can be applied to a general class of networks, including wireless networks and stochastic processing networks (SPN) [3].

Assume that G has N different Independent Sets ("IS", not confined to "Maximal Independent Sets"). Denote the i'th IS as $x^i \in \{0,1\}^K$, a 0-1 vector that indicates which links are transmitting in this IS. The k'th element of x^i, $x^i_k = 1$ if link k is transmitting, and $x^i_k = 0$ otherwise. We also refer to x^i as a "transmission state", and x^i_k as the "transmission state of link k". Later, we also use $\mathbf{x}$ to generally denote a transmission state, or simply a "state".

B. Throughput-optimality Objective

Assume i.i.d. traffic arrival at each link k with a *normalized* arrival rate λ_k. And denote the vector of arrival rates as $\lambda \in R^K_+$. Without loss of generality, assume that $\lambda_k > 0, \forall k$. (The link(s) with zero arrival rate can be removed from the problem.) We say that λ is *feasible* if and only if $\lambda = \sum_i \bar{p}_i \cdot x^i$ for some probability distribution $\bar{\mathbf{p}} \in \mathcal{R}^N_+$ satisfying $\bar{p}_i \geq 0$ and $\sum_i \bar{p}_i = 1$. That is, λ is a convex combination of the IS's, such that it is possible to serve the arriving traffic with some transmission schedule. We say that λ is *strictly feasible* iff it is in the interior of the capacity region, i.e., iff it can be written as $\lambda = \sum_i \bar{p}_i \cdot x^i$ where $\bar{p}_i > 0$ and $\sum_i \bar{p}_i = 1$. Denote the set of strictly feasible λ as $\mathcal{C}$.

Our objective is to design a distributed scheduling algorithm to support any strictly feasible λ, such that all queues are stabilized (i.e., no queue length goes to infinity) [2]. Such an algorithm is said to be "throughput-optimal".

II. A distributed CSMA Algorithm and its throughput-optimality

In [1], [2], a distributed algorithm based on CSMA (Carrier Sense Multiple Access) was proposed. The basic CSMA operation is described in the following. Before transmitting, link k waits for a random period of time that is exponentially distributed with rate R_k. If it does not sense another transmission of a conflicting link during that time, then the link starts transmitting; otherwise, it suspends its backoff and resumes it after the conflicting transmission is over. [1] The transmission time of link k is exponentially distributed with mean 1. Assuming that the sensing time is negligible, given the continuous distribution of the backoff times, the probability for two conflicting links to start transmission at the same time is zero, so collisions are ignored. (Although this is an approximation in wireless networks, it is not an issue in a general class of networks such as stochastic processing networks [3].) Define $r_k = \log(R_k)$ as the "transmission aggressiveness" (TA) of link k. And let $\mathbf{r}$ be the vector of r_k's.

The key idea of the proposed algorithm is for each link k to dynamically adjust r_k according to its empirical arrival rate and service rate. If the arrival rate is larger than the service rate (i.e., the queue length of link k increases), then r_k should be increased. And vice versa.

A. TA adjustment Algorithm

The following algorithm is updated from "Algorithm 3" in [2].

Acknowledgement: Thanks to Prof. Devavrat Shah (MIT) who suggested using increasing lengths of adjustment periods in the algorithm. Also thanks to Jinwoo Shin (MIT) for useful discussions. This research was supported in part by MURI grant BAA 07-036.18.

[1] If more than one backlogged links share the same transmitter, the transmitter maintains independent backoff timers for these links.

Algorithm 3: The vector $\mathbf{r}$ is updated at time t_i, $i = 1, 2, \ldots$. Let $t_0 = 0$ and $T_i := t_i - t_{i-1}, i = 1, 2, \ldots$. Define "period i" as the time between t_{i-1} and t_i, and $\mathbf{r}(i)$ be the value of $\mathbf{r}$ at the end of period i, i.e., at time t_i.

Initially, set $\mathbf{r}(0) = \mathbf{0}$.[2] Then at time t_i ($i = 1, 2, \ldots$), update

$$r_k(i) = [r_k(i-1)+\alpha(i)(\lambda'_k(i)-s'_k(i)+\min\{c/r_k(i-1), \bar{w}\})]_+$$

where $\lambda'_k(i)$ and $s'_k(i)$ are the empirical average arrival rate and service rate of link k in period i (note that $E(\lambda'_k(i)) = \lambda_k$), $c > 0, \bar{w} > 0$ are small constants, and the step size satisfies $\alpha(i) \to 0$ as $i \to \infty$ and $\sum_i \alpha(i) = \infty$, $\sum_i \alpha(i)^2 < \infty$. (Note: This is an enhancement of Algorithm 1 [1] where the update is $r_k(i) = [r_k(i-1) + \alpha(i)(\lambda'_k(i) - s'_k(\mathbf{r}(i)))]_+$. It ensures that after convergence of the algorithm, the service rate is strictly larger than the arrival rate for each link, which in turn ensures that all queues are stable and the queue lengths tend to be small.)

Also, the $\alpha(i)$ and T_i are chosen such that

$$\sum_{m=0}^{\infty}[\alpha(m+1)\sum_{i=1}^{m}\alpha(i)]^2 < \infty \tag{1}$$

and

$$\sum_{m=0}^{\infty}[\alpha(m+1)\cdot\sum_{i=1}^{m}\alpha(i)\cdot f(m)/T_{m+1}] < \infty \tag{2}$$

where

$$f(m) = \exp\{(\frac{5}{2}K+1)\cdot[\lambda_{max}\cdot\sum_{i=1}^{m}\alpha(i)+\log(2)]\} \tag{3}$$

where K is the number of links, and $\lambda_{max} = \bar{\lambda} + \bar{w}$, where $\bar{\lambda}$ is the maximal average arrival rate in any time slot (It can be taken as the maximal instantaneous arrival rate).

Example: The setting $\alpha(i) = 1/[(i+1)\log(i+1)]$ and $T_i = i$ satisfies conditions (1) and (2). Note that this setting does not depend on, or require the knowledge of K and λ_{max}, and thus can generally apply to any network.

Proof: For $m \geq 1$, $0 \leq \sum_{i=1}^{m}\alpha(i) \leq \alpha(1) + \int_1^m 1/[(x+1)\log(x+1)]dx \leq c_1 + \log\log(m+1)$ where $c_1 = \alpha(1) - \log\log 2 > 0$. So

$$\begin{aligned} f(m) \quad \leq \quad & \exp\{(\frac{5}{2}K+1)[\lambda_{max}\log\log(m+1) + \\ & \lambda_{max}c_1 + \log(2)]\} \end{aligned}$$

for $m \geq 1$. When $m = 0$, $\alpha(m+1)\cdot\sum_{i=1}^{m}\alpha(i)\cdot f(m)/T_{m+1} = 0$, so the L.H.S. of (2) is

$$\begin{aligned} & \sum_{m=1}^{\infty}[\alpha(m+1)\cdot\sum_{i=1}^{m}\alpha(i)\cdot f(m)/T_{m+1}] \\ & \leq \exp\{(\frac{5}{2}K+1)\cdot[\lambda_{max}c_1+\log(2)]\}\cdot \\ & \quad \sum_{m=1}^{\infty}\frac{[\log(m+1)]^{(\frac{5}{2}K+1)\cdot\lambda_{max}}[\log\log(m+1)+c_1]}{(m+1)(m+2)\log(m+2)}. \end{aligned}$$

When $m \geq M$ for a large enough M, $[\log(m+1)]^{(\frac{5}{2}K+1)\cdot\lambda_{max}}[\log\log(m+1)+c_1] \leq m^{1/2}$. Thus

$$\begin{aligned} & \sum_{m=M}^{\infty}\frac{[\log(m+1)]^{(\frac{5}{2}K+1)\cdot\lambda_{max}}[\log\log(m+1)+c_1]}{(m+1)(m+2)\log(m+2)} \\ & \leq \sum_{m=M}^{\infty}\frac{m^{1/2}}{m^2\log(M+2)} \\ & = \frac{1}{\log(M+2)}\sum_{m=M}^{\infty}m^{-3/2} < \infty. \end{aligned}$$

So (2) holds. It is easy to check that (1) also holds. ∎

Similarly, it can be verified that $\alpha(i) = c_0/[(a\cdot i+b+1)\log(a\cdot i+b+1)]$ and $T_i = a\cdot i+b$ (with constants $a > 0, b > 0, c_0 > 0$) also satisfy conditions (1) and (2).

B. Review of the ideas behind Algorithm 3

Before stating and proving the main convergence theorem in this paper, we need some definitions and a review of the ideas behind Algorithm 3.

Let $x_0(m-1)$ be the "initial" state of the CSMA Markov chain at the beginning of period m (i.e., at time t_{m-1}). Define the random vector $U(m-1) := (\mathbf{s}'(m-1), \lambda'(m-1), \mathbf{r}(m-1), x_0(m-1))$ for $m > 1$ and $U(0) = (\mathbf{r}(0) = \mathbf{0}, x_0(0))$. For $m \geq 1$, let $\mathcal{F}_{m-1}$ be the σ-field generated by $U(0), U(1), \ldots, U(m-1)$. It is easy to see that $U(0), U(1), \ldots$ is also a Markov process.

As in [2], Algorithm 3 above is a subgradient dual algorithm to solve the following optimization problem[3], but with inaccurate subgradients:

$$\begin{array}{ll} \max_{\mathbf{u},\mathbf{w}} & -\sum_i u_i\log(u_i) + c\sum_k\log(w_k) \\ \text{s.t.} & \sum_i(u_i\cdot x_k^i) \geq \lambda_k + w_k, \forall k \\ & u_i \geq 0, \sum_i u_i = 1 \\ & 0 \leq w_k \leq \bar{w}, \forall k \end{array} \tag{4}$$

where λ is strictly feasible. Let $\mathbf{r}^* \in \mathcal{R}_+^K$ be the optimal dual variables (to be further explained below) of problem (4). It was shown in [2] that $s_k(\mathbf{r}^*) > \lambda_k, \forall k$, where $s_k(\mathbf{r}^*)$ is the average service rate of link k with the TA vector $\mathbf{r}^*$. So if our algorithm makes $\mathbf{r}$ converge to $\mathbf{r}^*$, then eventually the service rate is strictly larger than the arrival rate for each link, which in turn ensures that all queues are stable.

To see that Algorithm 3 is a subgradient dual algorithm with inaccurate subgradients, note that a partial Lagrangian of problem (4) is

$$\begin{aligned} L(\mathbf{u},\mathbf{w};\mathbf{r}) \quad = \quad & -\sum_i u_i\log(u_i) + c\sum_k\log(w_k) + \\ & \sum_k[r_k(\sum_i u_i\cdot x_k^i - \lambda_k - w_k)] \\ = \quad & [-\sum_i u_i\log(u_i) + \sum_k(r_k\sum_i u_i\cdot x_k^i)] + \\ & \sum_k[c\cdot\log(w_k) - r_k w_k] - \sum_k(r_k\lambda_k) \end{aligned} \tag{5}$$

where r_k's are dual variables.

Let $L(\mathbf{r}) := \max_{\mathbf{u},\mathbf{w}} L(\mathbf{u},\mathbf{w};\mathbf{r})$ subject to the constraints that $u_i \geq 0, \sum_i u_i = 1$ and $0 \leq w_k \leq \bar{w}, \forall k$. Also, denote by $\mathbf{u}(\mathbf{r})$ and $\mathbf{w}(\mathbf{r})$ the maximizers. It follows from the optimization theory [8] that the vector $\mathbf{g} \in \mathcal{R}^K$ whose k'th

[2]In fact, $\mathbf{r}(0)$ can be any finite value. We assume $\mathbf{r}(0) = \mathbf{0}$ for the ease of exposition.

[3]A slight difference of problem (4) from that in [2] is that here we impose an upper bound $\bar{w}$ to $w_k, \forall k$.

element $g_k := \sum_i u_i(\mathbf{r}) \cdot x_k^i - \lambda_k - w_k(\mathbf{r})$ is a subgradient of $L(\cdot)$ at $\mathbf{r}$. It was shown in [2] that $\sum_i u_i(\mathbf{r}) \cdot x_k^i = s_k(\mathbf{r})$, and it's easy to find that $w_k(\mathbf{r}) = (c/r_k) \wedge \bar{w}$ (since $\partial L(\mathbf{u}, \mathbf{w}; \mathbf{r})/\partial w_k = c/w_k - r_k$).

So given a vector of dual variables $\mathbf{r}(m-1)$ at the beginning of the m'th period of Algorithm 3, the vector $\mathbf{g}(m)$ whose k'th element $g_k(m) := s_k(\mathbf{r}(m-1)) - \lambda_k - (c/r_k(m-1)) \wedge \bar{w}$ is a subgradient. To find $\mathbf{r}^*$ that minimizes $L(\mathbf{r})$, Algorithm 3 follows the opposite direction of $\mathbf{g}(m)$. However, we only have an estimation of $g_k(m)$, denoted by

$$g'_k(m) = s'_k(m) - \lambda'_k(m) - (c/r_k(m-1)) \wedge \bar{w}. \quad (6)$$

The "error bias" in the k'th element of the estimated subgradient is defined as

$$\begin{aligned} B_k(m): &= E[g'_k(m)|\mathcal{F}_{m-1}] - g_k(m) \\ &= E[s'_k(m)|\mathcal{F}_{m-1}] - s_k(\mathbf{r}(m-1)). \end{aligned} \quad (7)$$

Define also the zero-mean "noise"

$$\eta_k(m) := (s'_k(m) - E[s'_k(m)|\mathcal{F}_{m-1}]) - (\lambda'_k(m) - \lambda_k).$$

Since both $s'_k(m)$ and $\lambda'_k(m)$ are bounded, the noise is also bounded: $|\eta_k(m)| \le c_2$ for some $c_2 > 0$.

Then, we have

$$g'_k(m) = g_k(m) + B_k(m) + \eta_k(m). \quad (8)$$

C. Main convergence theorem and the stability of queues

Theorem 1: Assume that λ is strictly feasible. Then with Algorithm 3, $\mathbf{r}$ converges to $\mathbf{r}^*$ with probability 1. Moreover, all queues are stable (i.e., positive recurrent).

Proof:

The proof is composed of two parts. In the first part, we show that with Algorithm 3 and condition (2), the error bias (7) decreases "fast enough" with time. In the second part (Lemma 1), we use the result of part 1 and condition (1) to prove the convergence of $\mathbf{r}$ to $\mathbf{r}^*$.

In the following consider period $m+1$ (i.e., from t_m to t_{m+1}). At time t_m with the TA vector $\mathbf{r}(m)$, denote the corresponding CSMA Markov chain by $X(t)$ (for convenience we drop the index $m+1$). $X(t)$ is a continuous time Markov chain (CTMC). The probability of state $\mathbf{x}$ in the stationary distribution of $X(t)$ is [1]

$$\pi_x(\mathbf{r}(m)) = \frac{1}{C(\mathbf{r}(m))} \exp(\sum_k x_k r_k(m))$$

where $C(\mathbf{r}(m)) = \sum_{\mathbf{x}'} \exp(\sum_k x'_k r_k(m))$.

Since $\mathbf{r}(m) \ge \mathbf{0}$,

$$C(\mathbf{r}(m)) \le \sum_{\mathbf{x}'} \exp(\mathbf{1}^T \mathbf{r}(m)) \le 2^K \exp(\mathbf{1}^T \mathbf{r}(m))$$

since there are at most 2^K states. Also, $\exp(\sum_k x_k r_k(m)) \ge 1$. So, the minimal probability in the stationary distribution

$$\pi_{min}(\mathbf{r}(m)) \ge 1/\exp(\mathbf{1}^T \mathbf{r}(m) + K \cdot \log(2)).$$

Since $\lambda'_k(i) + \min\{c/r_k(i), \bar{w}\} \le \lambda_{max}$ and $s'_k(\mathbf{r}(i)) \ge 0$, we have $r_k(i+1) \le r_k(i) + \alpha(i)\lambda_{max}, \forall i, k$. Recall that $r_k(1) = 0, \forall k$. So $r_k(m) \le \lambda_{max} \sum_{i=1}^m \alpha(i), \forall k$. Thus,

$$\pi_{min}(\mathbf{r}(m)) \quad \ge \quad \exp\{-K \cdot [\lambda_{max} \sum_{i=1}^m \alpha(i) + \log(2)]\}. (9)$$

To proceed with the proof, we first "uniformize" the CSMA Markov chain $X(t)$. Through "uniformization", one can use certain results known for a DTMC (discrete time Markov chain) to analyze a CTMC. Define a constant $A_{m+1} = K \cdot \exp(\lambda_{max} \sum_{i=1}^m \alpha(i))$, and let $N(t)$ be a Poisson process with rate A_{m+1}. Denote Q as the transition rate matrix of $X(t)$. Define a DTMC $Z(n)$, independent of $N(t)$, with transition probability matrix P, where $P(x, x') = Q(x, x')/A_{m+1}, \forall x \neq x'$, and $P(x, x) = 1 - \sum_{x' \neq x} Q(x, x')/A_{m+1}$. (To see that P is a valid transition probability matrix, first note that $\sum_{x'} P(x, x') = 1$. Second, $P(x, x') \ge 0, \forall x' \neq x$. Also, since $r_k(m) \le \lambda_{max} \sum_{i=1}^m \alpha(i), \forall k$, we have $Q(x, x') \le \exp(\lambda_{max} \sum_{i=1}^m \alpha(i))$. Notice that $Q(x, x') > 0$ for at most K different x', i.e., state x can at most transit to K other states by changing the state of any one of the K links, so $\sum_{x' \neq x} Q(x, x') \le A_{m+1}$. Therefore, $P(x, x) \ge 0$.) Note that the stationary distribution of Z_n is the same as that of $X(t)$ (since the detailed balance equations still hold).

An important observation is that the CTMC defined as $Y(t) := Z(N(t))$ is equivalent to $X(t)$. To see this, assume that $Y(t_0) = x_0$, i.e., $Z(N(t_0)) = x_0$. Then the Markov chain transits to state $x' \neq x_0$ before $t_0 + h$ (h is very small) with probability $A_{m+1}h \cdot P(x, x') + o(h) = Q(x, x')h + o(h)$ (since this happens iff there is an arrival in $N(t)$ before $t_0 + h$ and the DTMC transits to x' upon this arrival).

Now we are ready to estimate how far $E[s'_k(m+1)|\mathcal{F}_m]$ is from the desired value $s_k(\mathbf{r}(m))$ (although it will take a number of steps). Let the vector $\mu_{\mathbf{x}_0}(\mathbf{r}(m); t)$ be the probabilities of all states at time $t_m + t$ (where $0 \le t \le T_{m+1}$), given that the initial state at time t_m is $\mathbf{x}_0(m)$ (where the subscript $\mathbf{x}_0$ is a shorthand for $x_0(m)$). And let $\mu_{\mathbf{x}_0}(\mathbf{r}(m); t, \mathbf{x}')$ be the probability of $\mathbf{x}'$ at time $t_m + t$. Let $\mathbf{x}(t_m + t)$ be the state at time $t_m + t$, and $x_k(t_m + t)$ be link k's state at that time. Note that in the time interval $[t_m, t_m + T_{m+1}]$, the TA is fixed at $\mathbf{r}(m)$. We first compute $E[s'_k(m+1)|\mathcal{F}_m]$ as follows.

$$\begin{aligned} & E[s'_k(m+1)|\mathcal{F}_m] \\ & \quad = E[\int_0^{T_{m+1}} I(x_k(t_m + t) = 1) dt / T_{m+1}] \\ & \quad = \int_0^{T_{m+1}} P(x_k(t_m + t) = 1) dt / T_{m+1} \\ & \quad = \sum_{\mathbf{x}': x'_k = 1} [\int_0^{T_{m+1}} \mu_{\mathbf{x}_0}(\mathbf{r}(m); t, \mathbf{x}') dt / T_{m+1}] \\ & \quad = \sum_{\mathbf{x}': x'_k = 1} \bar{\mu}_{\mathbf{x}_0}(\mathbf{r}(m); T_{m+1}, \mathbf{x}') \end{aligned}$$

where

$$\bar{\mu}_{\mathbf{x}_0}(\mathbf{r}(m); T_{m+1}, \mathbf{x}') = \int_0^{T_{m+1}} \mu_{\mathbf{x}_0}(\mathbf{r}(m); t, \mathbf{x}') dt / T_{m+1}$$

is the time-averaged probabilities of state $\mathbf{x}'$ in the interval. We use $\bar{\mu}_{\mathbf{x}_0}(\mathbf{r}(m); T_m)$ to denote the vector of probabilities of all states.

Let $\pi_{\mathbf{x}_0}(\mathbf{r}(m))$ be the probability of $\mathbf{x}_0$ in the stationary distribution of $X(t)$. Use $||\cdot||_{var}$ to denote the variation distance between two distributions. Let β_1 be the second largest eigenvalue of P. The following inequality [4] uses the fact that $Y(t)$ is equivalent to $X(t)$,

$$\begin{aligned}
&||\mu_{\mathbf{x}_0}(\mathbf{r}(m);t) - \pi(\mathbf{r}(m))||_{var} \\
&\quad \le \frac{1}{2}\sqrt{\frac{1-\pi_{\mathbf{x}_0}(\mathbf{r}(m))}{\pi_{\mathbf{x}_0}(\mathbf{r}(m))}}\exp(-A_{m+1}(1-\beta_1)t) \\
&\quad \le \frac{1}{2}\sqrt{\frac{1}{\pi_{min}(\mathbf{r}(m))}}\exp(-A_{m+1}(1-\beta_1)t).
\end{aligned}$$

So,

$$\begin{aligned}
&||\bar{\mu}_{\mathbf{x}_0}(\mathbf{r}(m);T_{m+1}) - \pi(\mathbf{r}(m))||_{var} \\
&= ||\int_0^{T_{m+1}}[\mu_{\mathbf{x}}(\mathbf{r}(m);t) - \pi(\mathbf{r}(m))]dt/T_{m+1}||_{var} \\
&\le \int_0^{T_{m+1}}||\mu_{\mathbf{x}}(\mathbf{r}(m);t) - \pi(\mathbf{r}(m))||_{var}dt/T_{m+1} \\
&\le \frac{1}{2}\sqrt{\frac{1}{\pi_{min}(\mathbf{r}(m))}}\frac{1-\exp(-A_{m+1}(1-\beta_1)T_{m+1})}{A_{m+1}(1-\beta_1)T_{m+1}} \\
&\le \frac{1}{2}\sqrt{\frac{1}{\pi_{min}(\mathbf{r}(m))}}\frac{1}{A_{m+1}(1-\beta_1)T_{m+1}} \qquad (10)
\end{aligned}$$

where the first inequality has used the fact that $||\cdot||_{var}$ is a convex function.

Also, β_1 can be bounded by [5]

$$\beta_1 \le 1-\phi^2/2 \qquad (11)$$

where ϕ is the "conductance" of P [5], defined as

$$\phi := \min_{S\subset\Omega,\pi(S)\in(0,1/2]}\frac{F(S,S^c)}{\pi_S(\mathbf{r}(m))}$$

where $\pi_S(\mathbf{r}(m)) = \sum_{x\in S}\pi_x(\mathbf{r}(m))$, and $F(S,S^c)$ is the "ergodic flow" from S to S^c: $F(S,S^c) = \sum_{x\in S,x'\in S^c}F(x,x') = \sum_{x\in S,x'\in S^c}\pi_x(\mathbf{r}(m))P(x,x')$. Then similar to [6], we have

$$\begin{aligned}
\phi &\ge \min_{S\subset\Omega,\pi(S)\in(0,1/2]}F(S,S^c) \\
&\ge \min_{x\ne x',P(x,x')>0}F(x,x') \\
&= \min_{x\ne x',P(x,x')>0}\{\pi_x(\mathbf{r}(m))\cdot P(x,x')\}.
\end{aligned}$$

For any $x \ne x'$ such that $P(x,x') > 0$, it must be that $Q(x,x') > 0$. Note that $Q(x,x') = 1$ or $Q(x,x') = \exp(r_k(m))$ for some k. Since $r_k(m) \ge 0$ in our algorithm, $Q(x,x') \ge 1$. So, $P(x,x') = Q(x,x')/A_{m+1} \ge 1/A_{m+1}$. Plugging this into the last inequality, we find

$$\phi \ge \min_x \pi_x(\mathbf{r}(m))/A_{m+1} = \pi_{min}(\mathbf{r}(m))/A_{m+1}.$$

Combined with (11), $\beta_1 \le 1-[\pi_{min}(\mathbf{r}(m))/A_{m+1}]^2/2$. Thus $1/(1-\beta_1) \le 2\cdot A_{m+1}^2[\pi_{min}(\mathbf{r}(m))]^{-2}$. Plugging this into (10) and use (9), we have

$$\begin{aligned}
&||\bar{\mu}_{\mathbf{x}_0}(\mathbf{r}(m);T_{m+1}) - \pi(\mathbf{r}(m))||_{var} \\
&\quad \le A_{m+1}[\pi_{min}(\mathbf{r}(m))]^{-5/2}/T_{m+1}. \\
&\quad \le K\cdot f(m)/T_{m+1}
\end{aligned}$$

where $f(m)$ is defined in (3). So,

$$\begin{aligned}
&|E[s'_k(m+1)|\mathcal{F}_m] - s_k(\mathbf{r}(m))| \\
&\quad = |\sum_{\mathbf{x}':x'_k=1}\bar{\mu}_{\mathbf{x}_0}(\mathbf{r}(m);T_{m+1},\mathbf{x}') - s_k(\mathbf{r}(m))| \\
&\quad \le 2||\bar{\mu}_{\mathbf{x}_0}(\mathbf{r}(m);T_{m+1}) - \pi(\mathbf{r}(m))||_{var} \\
&\quad \le 2\cdot K\cdot f(m)/T_{m+1}, \forall k.
\end{aligned}$$

Also, $E[\lambda'_k(m+1)] = \lambda_k$. Therefore, the error bias $B_k(m+1) = E[s'_k(m+1)|\mathcal{F}_m] - s_k(\mathbf{r}(m))$ satisfies $|B_k(m+1)| \le 2K\cdot f(m)/T_{m+1}$. Denote by $\mathbf{B}(m)$ the vector of $B_k(m+1)$'s. Since $|r_k(m) - r^*_k| \le \bar{r} + \lambda_{max}\sum_{i=1}^m \alpha(i)$, where $\bar{r} = \max_k r^*_k$, we show that the term $(\mathbf{r}(m)-\mathbf{r}^*)^T\cdot\mathbf{B}(m+1)$ is diminishing:

$$\begin{aligned}
&\sum_{m=0}^{\infty}\alpha(m+1)|(\mathbf{r}(m)-\mathbf{r}^*)^T\cdot\mathbf{B}(m+1)| \\
&\quad \le 2K^2\sum_{m=0}^{\infty}[\alpha(m+1)(\bar{r}+\lambda_{max}\sum_{i=1}^m\alpha(i))\cdot f(m)/T_{m+1}] \\
&\quad = 2K^2\cdot\lambda_{max}\sum_{m=0}^{\infty}[\alpha(m+1)\cdot\sum_{i=1}^m\alpha(i)\cdot f(m)/T_{m+1}] \\
&\qquad + 2K^2\cdot\bar{r}\sum_{m=0}^{\infty}[\alpha(m+1)\cdot f(m)/T_{m+1}] < \infty \qquad (12)
\end{aligned}$$

where the last step has used condition (2).

Lemma 1: If (12) and (1) hold, then with Algorithm 3, $\mathbf{r}$ converges to $\mathbf{r}^*$ with probability 1.

The line of proof is similar to that of Theorem 3.1 in [7], but with more intricacies. The complete proof is given in the Appendix.

With Algorithm 3, $\mathbf{r}$ converges to the optimal value $\mathbf{r}^*$. Since $s_k(\mathbf{r}^*) - \lambda_k \ge \delta(\lambda), \forall k$ for some $\delta(\lambda) > 0$, one can then show that the queues are stable as follows.

Choose a large-enough time t' when $\mathbf{r}$ has converged to $\mathbf{r}^*$. Choose T to be large enough such that in the time interval $[t, t+T]$ where $t \ge t'$, $E(\lambda'_k - s'_k)$ is very close to $\lambda_k - s_k(\mathbf{r}^*)$ such that $E(\lambda'_k - s'_k) \le -\delta(\lambda)/2$ (T can be found since roughly speaking, the mixing time does not increase after t' due to the convergence of $\mathbf{r}$).

If $Q_k(t) > T$, then the queue length of link k is always positive before time $t+T$, because Q_k at most decreases with a rate of 1. Therefore $Q_k(t+T) = Q_k(t) + T\cdot(\lambda'_k - s'_k)$. So, $E(Q_k(t+T)|Q_k(t)) - Q_k(t) = T\cdot E(\lambda'_k - s'_k) \le -T\cdot\delta(\lambda)/2 < 0$. In other words, the queue has negative drift if it's large enough after time t'. Therefore queue k is stable for any k.

D. A variation of Algorithm 3 with bounded $\mathbf{r}$

If it is known that the optimal $\mathbf{r}^*$ satisfies that $r^*_k \le r_{max}, \forall k$ for some $r_{max} > 0$, then the updated in Algorithm 3 can be modified into

$$\begin{aligned}
r_k(i) \;=\; &\min\{r_{max}, [r_k(i-1) + \alpha(i)(\lambda'_k(i) - s'_k(i) \\
&+ \min\{c/r_k(i-1), \bar{w}\})]_+\}
\end{aligned}$$

where $\alpha(i) \to 0$ as $i \to \infty$ and $\sum_i \alpha(i) = \infty$, $\sum_i \alpha(i)^2 < \infty$.

Condition (2) can be relaxed to

$$\sum_{m=0}^{\infty}[\alpha(m+1)/T_{m+1}] < \infty$$

and condition (1) is not needed.

For example, $\alpha(i) = 1/i$ and $T_i = i$ are sufficient to guarantee convergence of $\mathbf{r}$ and the stability of the queues. The proof is similar and is thus omitted.

III. Diminishing Step size with equal adjustment period

In the above algorithm, T_i increases with i. In this section we consider updating $\mathbf{r}$ once every time slot with step sizes $\beta(t)$, $t = 1, 2, 3, \ldots$. Specifically, for $t_{i-1} \le t < t_i = t_{i-1} + T_i$, let $\beta(t) = \alpha(i)/T_i$. (In other words, we even out the updates throughout period i instead of only update $\mathbf{r}$ at the end of period i. Intuitively, this is very similar to the above algorithms and should have similar results.

Assume that $\alpha(i) = 1/[(i+1)\log(i+1)]$ and $T_i = i$. Since the total time of the first n period is $1+2+\cdots+n = n(n+1)/2 \approx n^2/2$ when n is large, then time t is approximately in period $\sqrt{2t}$. Thus,

$$\begin{aligned}\beta(t) &\approx \alpha(\sqrt{2t})/\sqrt{2t} \\ &\approx 1/[\sqrt{2t}\log(\sqrt{2t})]/\sqrt{2t} \\ &= 1/[t\log(2t)].\end{aligned}$$

Coincidentally, this is similar to the form of $\alpha(i)$. Although not formally proved, setting the step size $\beta'(t) = 1/[t\log(2t)]$ should also lead to throughput-optimal performance, as well as $\beta'(t) = c_0/[(a\cdot t+b)\log(2(a\cdot t+b))]$ where $a, b, c_0 > 0$.

Appendix: Proof of Lemma 1

Let $\mathbf{r}^*$ be the optimal dual variables of problem (4). Use $||\cdot||$ to denote the L2 norm. Since $r_k(m) = [r_k(m-1) - \alpha(m)\cdot g'_k(m)]_+$ by Algorithm 3, we have

$$\begin{aligned}&||\mathbf{r}(m) - \mathbf{r}^*||^2 \\ &\le ||\mathbf{r}(m-1) - \alpha(m)\cdot g'_k(m) - \mathbf{r}^*||^2 \\ &= ||\mathbf{r}(m-1) - \mathbf{r}^*||^2 - \alpha(m)\cdot[\mathbf{r}(m-1)-\mathbf{r}^*]^T\mathbf{g}'(m) \\ &\quad + \alpha^2(m)||\mathbf{g}'(m)||^2\end{aligned}$$

where the first inequality follows from the fact that the projection $[\cdot]_+$ is non-expansive [8]. Denote $d(m) = ||\mathbf{r}(m) - \mathbf{r}^*||^2$. Since $||\mathbf{g}'(m)||^2$ is bounded (cf. (6)), write $||\mathbf{g}'(m)||^2 \le C$. Using this and (8),

$$\begin{aligned}d(m) \le\ & d(m-1) + \alpha(m)\cdot[\mathbf{r}^* - \mathbf{r}(m-1)]^T\mathbf{g}(m) \\ &+\alpha(m)\cdot[\mathbf{r}^* - \mathbf{r}(m-1)]^T[\mathbf{B}(m)+\eta(m)] \\ &+\alpha^2(m)\cdot C. \qquad (13)\end{aligned}$$

Assume that $\mathbf{r}(m-1) \notin H_\mu := \{\mathbf{r}|L(\mathbf{r}) \le \mu + L(\mathbf{r}^*)\}$. Since $\mathbf{g}(m)$ is a subgradient of $L(.)$ at $\mathbf{r}(m-1)$, we have $[\mathbf{r}^* - \mathbf{r}(m-1)]^T\mathbf{g}(m) \le L(\mathbf{r}^*) - L(\mathbf{r}) \le -\mu$. So

$$\begin{aligned}&E(d(m)|\mathcal{F}_{m-1}) \\ &\quad\le d(m-1) - \alpha(m)\mu \\ &\qquad + \alpha(m)\cdot[\mathbf{r}^* - \mathbf{r}(m-1)]^T\mathbf{B}(m) \\ &\qquad + \alpha^2(m)\cdot C. \qquad (14)\end{aligned}$$

As shown before, $|B_k(m)| \le 2\cdot f(m)/T_m$ in any realization. So $|\sum_m\{\alpha(m)\cdot[\mathbf{r}^* - \mathbf{r}(m-1)]^T\mathbf{B}(m)\}| < \infty$ by inequality (12) and $\sum_m \alpha^2(m)\cdot C < \infty$. Then we use the same super-martingale lemma (Lemma A.1) in [7] to conclude that the set H_μ is recurrent for $\{\mathbf{r}(m)\}$.

Next, by (13) we have for $n \ge m$,

$$\begin{aligned}d(n) \le\ & d(m-1) \\ &+ \sum_{i=m}^{n}\{\alpha(i)\cdot[\mathbf{r}^* - \mathbf{r}(i-1)]^T\mathbf{g}(i)\} \qquad (15) \\ &+ \sum_{i=m}^{n}\{\alpha(i)\cdot[\mathbf{r}^* - \mathbf{r}(i-1)]^T[\mathbf{B}(i)+\eta(i)]\} \qquad (16) \\ &+ C\sum_{i=m}^{n}\alpha^2(i). \qquad (17)\end{aligned}$$

Since $C\sum_{i=1}^{\infty}\alpha^2(i) < \infty$, we have

$$\lim_{m\to\infty} C\sum_{i=m}^{\infty}\alpha^2(i) = 0. \qquad (18)$$

Also,

$$\sum_{i=1}^{\infty}|\alpha(i)\cdot[\mathbf{r}^* - \mathbf{r}(i-1)]^T\mathbf{B}(i)| < \infty$$

by (12). So

$$\lim_{m\to\infty}\sum_{i=m}^{\infty}|\alpha(i)\cdot[\mathbf{r}^* - \mathbf{r}(i-1)]^T\mathbf{B}(i)| = 0. \qquad (19)$$

Finally,

$$W(n) := \sum_{i=1}^{n}\{\alpha(i)\cdot[\mathbf{r}^* - \mathbf{r}(i-1)]^T\eta(i)\}$$

is a martingale. To see this, note that
(a) $W(n) \in \mathcal{F}_n$;
(b) $E|W(n)| < \infty, \forall n$; and
(c) $E(W(n)|\mathcal{F}_{n-1}) - W(n-1) = \alpha(n)\cdot[\mathbf{r}^* - \mathbf{r}(n-1)]^T E[\eta(n)|\mathcal{F}_{n-1}] = 0$.
Also, since

$$|(\mathbf{r}^* - \mathbf{r}(m-1))^T\eta(m)| \le K\cdot c_2[\bar{r} + \lambda_{max}\sum_{i=1}^{m-1}\alpha(i)]$$

(recall that $|\eta_k(m)| \le c_2$), we have

$$\begin{aligned}&E\{[\alpha(m)\cdot(\mathbf{r}^* - \mathbf{r}(m-1))^T\eta(m)]^2\} \\ &\quad= \alpha(m)^2 E[|(\mathbf{r}^* - \mathbf{r}(m-1))^T\eta(m)|^2] \\ &\quad\le \alpha(m)^2K^2c_2^2[\bar{r} + \lambda_{max}\sum_{i=1}^{m-1}\alpha(i)]^2.\end{aligned}$$

Therefore

$$\begin{aligned}
&\sup_n E(W(n)^2) \\
&\quad = \sup_n \sum_{m=1}^{n} E\{[\alpha(m)\cdot(\mathbf{r}^* - \mathbf{r}(m-1))^T \eta(m)]^2\} \\
&\quad \le \sum_{m=1}^{\infty} E\{[\alpha(m)\cdot(\mathbf{r}^* - \mathbf{r}(m-1))^T \eta(m)]^2\} \\
&\quad \le \sum_{m=1}^{\infty} \{\alpha(m)^2 K^2 c_2^2 [\bar{r} + \lambda_{max} \sum_{i=1}^{m-1} \alpha(i)]^2\} \\
&\quad < \infty
\end{aligned}$$

where the last step follows from condition (1). By the L2 Martingale Convergence Theorem [9], $W(n)$ converges with probability 1. So

$$\begin{aligned}
&\sup_{n \ge m \ge N_0} |\sum_{i=m}^{n} \{\alpha(i)\cdot[\mathbf{r}^* - \mathbf{r}(i-1)]^T \eta(i)\}| \\
&\qquad = \sup_{n \ge m \ge N_0} |W(n) - W(m-1)| \to 0 \qquad (20)
\end{aligned}$$

as $N_0 \to \infty$ with probability 1.

Combining (18), (19) and (20), we know that with probability 1, for any $\epsilon > 0$, after $\mathbf{r}(m-1)$ returns to H_μ for some large enough m (due to recurrence of H_μ),

$$\begin{aligned}
&\sum_{i=m}^{n} \{\alpha(i)\cdot[\mathbf{r}^* - \mathbf{r}(i-1)]^T [\mathbf{B}(i) + \eta(i)]\} \\
&\qquad + C \sum_{i=m}^{n} \alpha^2(i) \le \epsilon
\end{aligned}$$

for any $n \ge m$. Since the term (15) is always non-positive, we have $d(n) \le d(m-1) + \epsilon, \forall n \ge m$. In other words, $\mathbf{r}$ cannot move far away from H_μ after iteration $m-1$. Since the above argument hold for H_μ with arbitrarily small μ and any $\epsilon > 0$, $\mathbf{r}$ converge to $\mathbf{r}^*$ with probability 1.

References

[1] L. Jiang and J. Walrand, "A Distributed CSMA Algorithm for Throughput and Utility Maximization in Wireless Networks," the Forty-Sixth Annual Allerton Conference, Sep. 23-26, 2008.

[2] L. Jiang and J. Walrand, "A Distributed Algorithm for Maximal Throughput and Optimal Fairness in Wireless Networks with a General Interference Model", Technical Report, UC Berkeley, Apr. 2008. URL: http://www.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-38.html

[3] J. M. Harrison and R. J. Williams, "Workload interpretation for Brownian models of stochastic processing networks," Mathematics of Operations Research, 32 (2007), 808–820.

[4] P. Diaconis and D. Strook. Geometric bounds for eigenvalues of Markov chains. Annals of Applied Probability, 1 (1991), pp. 36-61.

[5] V. Guruswami, "Rapidly Mixing Markov Chains: A Comparison of Techniques."

[6] S. Rajagopalan and D. Shah, "Distributed Algorithm and Reversible Network", CISS'08.

[7] J. Zhang, D. Zheng, and M. Chiang, "The impact of stochastic noisy feedback on distributed network utility maximization," IEEE Transactions on Information Theory, vol. 54, no. 2, pp. 645-665, February 2008.

[8] S. Boyd and L. Vandenberghe, "Convex Optimization", Cambridge University Press, 2004.

[9] R. Durrett, "Probability: Theory and Examples", Duxbury Press, 3rd edition, March 16, 2004.