Forty-Eighth Annual Allerton Conference
Allerton House, UIUC, Illinois, USA
September 29 - October 1, 2010

# Scheduling in Multihop Wireless Networks without Back-pressure

Shihuan Liu*, Eylem Ekici†, and Lei Ying*
*Department of Electrical and Computer Engineering, Iowa State University
Email: {liush08, leiying}@iastate.edu
†Department of Electrical and Computer Engineering, The Ohio State University
Email: ekici@ece.osu.edu

***Abstract*—This paper focuses on scheduling in multihop wireless networks. The well-known back-pressure scheduling algorithm is throughput optimal, but requires constant exchange of queue-length information among neighboring nodes for calculating the "back-pressure". In this paper, we propose a self-regulated MaxWeight scheduling, which does not require back-pressure calculation. We prove that the self-regulated MaxWeight scheduling is throughput optimal[1] when the traffic flows are associated with fixed routes and deterministic arrivals.**

## I. Introduction

This paper considers scheduling in multihop wireless networks. A widely-used algorithm to stabilize multihop flows in wireless networks is the back-pressure algorithm proposed in [1], which can stabilize any traffic flows that can be supported by any other routing/scheduling algorithm. We refer to [2], [3] for a comprehensive survey on the back-pressure algorithm and its variations. A key idea of the back-pressure algorithm is to use queue difference as link weight, and schedule the links with large weights. Therefore, the back-pressure algorithm requires constant exchange of queue-length information among neighboring nodes. Further, the sum of the queue-lengths along a route increases quadratically as the route length [4], which leads to poor delay performance. The delay performance of the back-pressure algorithm has received much attention recently. Different approaches have been developed to improve the delay performance of back-pressure [4], [5], [6]. These algorithms however still rely on *back-pressure* for scheduling, so still require a constant exchange of queue-length information.

In this paper, we consider the following question: *can the network be stabilized without using back-pressure?* We address this question in a multi-hop wireless network with fixed routing. We note that a multi-hop flow with a fixed route can be broken into multiple single-hop flows, one for each link on the route. A scheduling policy that stabilizes the collection of single-hop flows also provides sufficient service for supporting the set of multi-hop flows. Therefore, assuming each link knows the aggregated rate it needs to carry, an alternative scheduling approach is to let each link generate virtual packets according to the aggregated rate and then let the network schedule the links according to the virtual queues. When a virtual queue is scheduled, real packets are served according to the allocated link rate. This approach is throughput optimal under the fixed routing assumption, but again requires information exchange in network. A source needs to estimate the arrival rate and communicate the rate to all nodes along the route associated with the source. A directly following question is whether, and under what conditions, it is possible to stabilize a network without explicitly exchanging any information among nodes in the network. In the following, we present an algorithm that can achieve this goal for networks where *(i)* sources generate packet at constant rates, and *(ii)* routes are fixed.

We propose a self-regulated MaxWeight scheduling algorithm where each node estimates the aggregated link rate locally (i.e., by taking average over the past arrivals on that link). We would like to emphasize that the accuracy of link-rate estimates relies on the stability of the network because if one queue builds up, it blocks packets to down-stream nodes so that those nodes cannot accurately estimate the link rates. On the other hand, the stability of the network relies on the accuracy of the link-rate estimates, which is an interesting paradox and makes the stability of the self-regulated MaxWeight scheduling a non-trivial problem. In this paper, we prove that the self-regulated MaxWeight scheduling is throughput optimal when the traffic flows are associated with fixed routes and constant arrivals. The self-regulated MaxWeight scheduling combined with distributed scheduling algorithms such as the CSMA-based scheduling [7], [8], [9] provides a scheduling algorithm for multi-hop wireless networks, which does not require any information exchange in the network. Finally, we would like to comment that the proposed algorithm is motivated by the idea of regulators, which was first proposed for re-entrant lines in [10] and later used for scheduling in wireless networks [11]. Our algorithm however does not require any information exchange in network, while the algorithm in [11] requires the mean arrival rates to be communicated to regulators in the network.

## II. Basic Model

In this section, we describe the basic model. We consider a network represented by a graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{L}$ is the set of directed links. Denote by $(m, n)$ the link from node $m$ to node $n$, which implies

[1]An algorithm is said to be throughput-optimal if it can stabilize any traffic that can be stabilized by any other algorithm.

that node $m$ can communicate with node $n$. Furthermore, let $\boldsymbol{\mu} = \{\mu_{(m,n)}\}$ denote a link-rate vector such that $\mu_{(m,n)}$ is the transmission rate over link $(m,n)$. A link-rate vector $\boldsymbol{\mu}$ is said to be *admissible* if the link-rates specified by $\boldsymbol{\mu}$ can be achieved *simultaneously*. Define $\Gamma$ to be the set of all admissible link-rate vectors. It is easy to see that $\Gamma$ depends on the choice of interference model and might not be a convex set. Furthermore, $\Gamma$ is time-varying if channels are time-varying. Furthermore, we assume that there exists $\mu_{\max}$ such that $\mu_{(m,n)} \leq \mu_{\max}$ for all $(m,n) \in \mathcal{L}$ and all admissible $\boldsymbol{\mu}$.

We consider multihop traffic flows with fixed routing in this paper. Let $f$ denote a flow, and $\mathbf{R}_f$ denote the route associated with flow $f$. We use $\mathcal{F}$ to denote the set of all flows in the network. Assume that time is discretized, and let $X_f$ $(f \in \mathcal{F})$ denote the number of packets injected by flow $f$ at time $t$. In this paper, we assume $X_f$ is a constant. Denote by $S_f$ the source node of flow $f$, and $D_f$ the destination node of flow $f$. Further, let $\mu^f_{(m,n)}$ denote the rate at which packets of flow $f$ are served over link $(m,n)$.

## III. NECESSARY CONDITIONS FOR STABILITY

In this section, we study the necessary conditions for network stability. We say a traffic configuration $\{X_f\}_{f\in\mathcal{F}}$ is supportable if there exists $\{\bar{\mu}^f_{(m,n)}\}_{(m,n)\in\mathcal{L}}$ such that the following two conditions hold:

(i) For any flow $f$ and node $n \neq D_f$,

$$X_f \mathbb{I}_{\{S_f=n\}} + \sum_{m:(m,n)\in\mathcal{L}} \bar{\mu}^f_{(m,n)} \leq \sum_{b:(n,b)\in\mathcal{L}} \bar{\mu}^f_{(n,b)}, \quad (1)$$

where $\mathbb{I}_{\{S_f=n\}}$ equals to one if $S_f = n$ occurs and equals to zero otherwise.

(ii)

$$\left\{ \sum_{f\in\mathcal{F}} \bar{\mu}^f_{(m,n)} \right\} \in \mathcal{CH}(\Gamma), \quad (2)$$

where $\mathcal{CH}(\Gamma)$ is the convex hull of $\Gamma$.

Recall that each flow is associated with a fixed route. It is easy to see the necessary condition (1) is equivalent to the following statement: For any flow $f$ and $(m,n) \in \mathbf{R}_f$, we have

$$\bar{\mu}^f_{(m,n)} \geq X_f. \quad (3)$$

## IV. SELF-REGULATED MAXWEIGHT SCHEDULING FOR MULTIHOP WIRELESS NETWORKS

In this section, we introduce the self-regulated MaxWeight scheduling for multhop wireless networks.

**Two-stage queue architecture:** Each node maintains two types of queues: per-flow queues and per-link queues as shown in Figure 1. An incoming packet is at first buffered at the corresponding per-flow queue and then moved to the per-link queue where the link is on the route of the flow (the details will be described later). In this paper, we denote by $Q^n_f$ the length of the queue maintained at node $n$ for flow $f$, and $Q^n_{(n,b)}$ the length of the queue maintained at node $n$ for link $(n,b)$. Clearly, queue for link $(n,b)$ is maintained only at node $n$, so we simplify $Q^n_{(n,b)}$ to be $Q_{(n,b)}$ without causing any confusion.

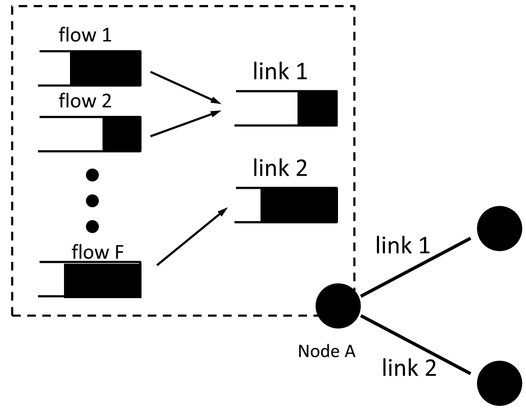


Fig. 1. The two-stage queue architecture

**Self-Regulated MaxWeight Scheduling**

- MaxWeight scheduling: Compute the admissible link-rate vector $\mu^*(t)$ such that

$$\mu^*(t) = \arg\max_{\boldsymbol{\mu}\in\Gamma} \sum_{(n,b)\in\mathcal{L}} \mu_{(n,b)}(t) Q_{(n,b)}(t). \quad (4)$$

- Per-link queue transmission: Node $n$ transmits

$$s_{(n,b)}(t) \triangleq \min\{\mu^*_{(n,b)}(t), Q_{(n,b)}(t)\}$$

packets to node $b$ over link $(n,b)$. The packets are deposited into per-flow queues at node $b$ according to their flows. We let $s^f_{(n,b)}(t)$ denote the number of packets of flow $f$ that are transmitted over link $(n,b)$ at time $t$. Note that

$$s_{(n,b)}(t) = \sum_f s^f_{(n,b)}(t)$$

always holds.

- Per-flow queue transmission: Denote by $a^n_f(t)$ the number of packets deposited into queue $Q^n_f$ at time slot $t$, i.e.,

$$a^n_f(t) = \sum_{(b,n)\in\mathcal{L}} s^f_{(b,n)}(t).$$

For each flow $f$, node $n$ maintains a rate estimate

$$\tilde{X}^n_f(t) = (1+\gamma)\frac{\sum_{\tau=1}^t a^n_f(\tau)}{t}, \quad (5)$$

where $\gamma$ is a positive constant which can be set as small as desired. At time slot $t$, node $n$ moves

$$s^n_f(t) \triangleq \min\{\tilde{X}^n_f(t), Q^n_f(t)\}$$

packets from queue $Q^n_f$ to queue $Q_{(n,b)}$ where $b$ is the next hop to reach destination $D_f$ from node $n$ (in other words, $b$ is on the route to destination $D_f$). We further define

$$a_{(n,b)}(t) \triangleq \sum_{f:(n,b)\in\mathbf{R}_f} s^n_f(t).$$

□

The notations are illustrated in Figure 2. From the definition of the self-regulated MaxWeight scheduling algorithm, we can see that the dynamics of queue $Q_f^n$ and $Q_{(n,b)}$ are:

$$\begin{aligned} Q_f^n(t+1) &= Q_f^n(t) - s_f^n(t) + a_f^n(t) \\ Q_{(n,b)}(t+1) &= Q_{(n,b)}(t) - s_{(n,b)}(t) + a_{(n,b)}(t), \end{aligned}$$

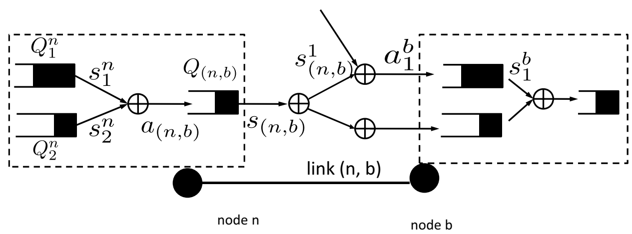


Fig. 2. The notations and the flows

In the following, we will prove that the self-regulated MaxWeight scheduling algorithm stabilizes any traffic that satisfies the necessary conditions. The analysis consists of two steps: we first show that the per-link queues are bounded (Lemma 1), and then using induction to prove that the per-flow queues are bounded as well (Lemma 2).

*Lemma 1:* Given a set of traffic flows $\{(1+\epsilon)X_f\}$ is supportable for some $\epsilon > \gamma$, there exists a positive constant $M > 0$ such that the lengths of the per-link queues are no more than $M$ for all $t \geq 0$.

*Proof:* First, note that $\sum_{\tau=1}^t a_f^n(\tau) \leq tX_f$ according to causality, which implies that

$$\tilde{X}_f^n(t) \leq (1+\gamma)X_f, \forall t.$$

Since $\{(1+\epsilon)X_f\}$ is supportable, according to the necessary conditions, there exist $\bar{\mu}$ such that

$$\bar{\mu}_{(m,n)}^f \geq (1+\epsilon)X_f$$

for all flow $f$ and link $(m,n)$ that is on the route of $f$. Let $\bar{\mu}_{(m,n)} = \sum_{f:(m,n)\in\mathbf{R}_f} \bar{\mu}_{(m,n)}^f$, then for any link $(m,n) \in \mathcal{L}$, we can conclude that

$$\begin{aligned} a_{(m,n)}(t) &= \sum_{f:(m,n)\in\mathbf{R}_f} \tilde{X}_f^m(t) \\ &\leq \sum_{f:(m,n)\in\mathbf{R}_f} (1+\gamma)X_f \\ &= \sum_{f:(m,n)\in\mathbf{R}_f} (1+\epsilon)X_f - \sum_{f:(m,n)\in\mathbf{R}_f} (\epsilon-\gamma)X_f \\ &\leq \sum_{f:(m,n)\in\mathbf{R}_f} \bar{\mu}_{(m,n)}^f - (\epsilon-\gamma) \sum_{f:(m,n)\in\mathbf{R}_f} X_f \\ &= \bar{\mu}_{(m,n)} - (\epsilon-\gamma) \sum_{f:(m,n)\in\mathbf{R}_f} X_f. \qquad (6) \end{aligned}$$

Note that $\epsilon > \gamma$.

We construct Lyapunov function as follows:

$$V(t) = \sum_{(m,n)\in\mathcal{L}} Q_{(m,n)}^2(t).$$

Then

$$\begin{aligned} & V(t+1) - V(t) \\ = & \sum_{(m,n)\in\mathcal{L}} Q_{(m,n)}^2(t+1) - \sum_{(m,n)\in\mathcal{L}} Q_{(m,n)}^2(t) \\ = & \sum_{(m,n)\in\mathcal{L}} \left(Q_{(m,n)}(t+1) + Q_{(m,n)}(t)\right) \times \\ & \left(Q_{(m,n)}(t+1) - Q_{(m,n)}(t)\right) \\ = & \sum_{(m,n)\in\mathcal{L}} \left(2Q_{(m,n)}(t) + a_{(m,n)}(t) - s_{(m,n)}(t)\right) \times \\ & \left(a_{(m,n)}(t) - s_{(m,n)}(t)\right) \\ \leq & \delta + \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \left(a_{(m,n)}(t) - s_{(m,n)}(t)\right) \\ = & \delta \\ + & \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \left(a_{(m,n)}(t) - \bar{\mu}_{(m,n)}\right) \qquad (7) \\ + & \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \left(\bar{\mu}_{(m,n)} - s_{(m,n)}(t)\right), \qquad (8) \end{aligned}$$

where $\delta = L(\mu_{\max})^2$.

From inequality (6), we can get

$$\begin{aligned} & \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \times \left(a_{(m,n)}(t) - \bar{\mu}_{(m,n)}\right) \\ \leq & -(\epsilon-\gamma) \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \left(\sum_{f:(m,n)\in\mathbf{R}_f} X_f\right). \end{aligned}$$

Furthermore, from the MaxWeight scheduler (4) and necessary condition (2),

$$\begin{aligned} & \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \times \left(\bar{\mu}_{(m,n)} - s_{(m,n)}(t)\right) \\ \leq & \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \times \left(\bar{\mu}_{(m,n)} - \mu_{(m,n)}^*(t)\right) + 2\delta \\ \leq & 2\delta. \end{aligned}$$

Therefore, we conclude that

$$\begin{aligned} & V(t+1) - V(t) \\ \leq & 3\delta - (\epsilon-\gamma) \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \left(\sum_{f:(m,n)\in\mathbf{R}_f} X_f\right). \end{aligned}$$

Now note that $\max_{(m,n)\in\mathcal{L}} Q_{(m,n)}(t) \geq \sqrt{\frac{V(t)}{L}}$. So $V(t) > L\left(\frac{2\delta}{(\epsilon-\gamma)\min_f X_f}\right)^2$ implies that

$$\max_{(m,n)\in\mathcal{L}} Q_{(m,n)}(t) > \frac{2\delta}{(\epsilon-\gamma)\min_f X_f},$$

and

$$\begin{aligned} & V(t+1) - V(t) \\ \leq\ & 3\delta - (\epsilon - \gamma) \sum_{(m,n)\in\mathcal{L}} 2Q_{(m,n)}(t) \left( \sum_{f:(m,n)\in\mathbf{R}_f} X_f \right) \\ \leq\ & 3\delta - 2(\epsilon-\gamma) \max_{(m,n)\in\mathcal{L}} Q_{(m,n)}(t) \left( \min_f X_f \right) \\ <\ & -\delta. \end{aligned}$$

From the analysis above, it is easy to show that there exists a constant $M_V$ such that $V(t) \leq M_V$ for all $t$. Since $Q_{(n,b)}(t) \leq \sqrt{V(t)}$ holds for all $t$, we can conclude the lemma by choosing $M = \sqrt{M_V}$. ■

Next, based on Lemma 1, we will prove that all per-flow queues are bounded as well.

*Lemma 2:* Given a set of traffic flows $\{X_f\}$ such that $\{(1+\epsilon)X_f\}$ is supportable for some $\epsilon > \gamma$, under the self-regulated MaxWeight scheduling, there exists a constant $\tilde{M}$ such that, the lengths of the per-flow queues are no more than $\tilde{M}$ for all $t$.

*Proof:* For the node $S_f$, the number of packets arriving at each time slot to queue $Q^f_{s_f}$ is a constant, i.e.,

$$a_f^{S_f}(t) = X_f, \forall t.$$

Since $a_f^{S_f}(t)$ is a constant, the estimate $\tilde{X}_f^{S_f}(t)$ is a constant as well. From (5), we have

$$\tilde{X}_f^{S_f}(t) = (1+\gamma)X_f, \forall t.$$

Therefore $Q^f_{s_f}(t) \leq X_f$ for all $t$.

Next we use induction to prove all per-flow queues are bounded by a constant. Denote by $n_f^i$ the $i^{\text{th}}$ node on the route of flow $f$. Now assume that

$$Q_f^{n_f^j}(t) \leq \tilde{M}_i \text{ for all } t, f \text{ and } j \leq i. \text{ (induction assumption)}$$

From Lemma 1 we know, for any traffic that $\{(1+\epsilon)X_f\}$ is supportable and $\epsilon > \gamma$, all per-link queues are bounded by a constant $M$. In other words, $Q_{(n_f^i, n_f^{i+1})}(t) \leq M$ for all $t$. We next define

$$M_\delta = i(M_i + M).$$

Now consider the departures of the per-flow queue $Q_f^{n_f^{i+1}}$ at node $n_f^{i+1}$. It is easy to see that

$$tX_f - M_\delta \leq \sum_{\tau=1}^{t} a_f^{n_f^{i+1}}(\tau) \leq tX_f$$

because $M_\delta$ is an upper bound on the number of packets belonging to flow $f$ and queued at nodes $n_f^1$ to node $n_f^i$ (the up-streaming nodes of node $i+1$).

We then can obtain that

$$\tilde{X}_f^{n_f^{i+1}}(t) = (1+\gamma)\frac{\sum_{\tau=1}^{t} a_f^{n_f^{i+1}}(\tau)}{t},$$

which converges to $(1+\gamma)X_f$ as $t \to \infty$. In other words, there exists $t_\delta$, such that when $t > t_\delta$,

$$(1+\gamma)X_f \geq \tilde{X}_f^{n_f^{i+1}}(t) > (1+\frac{\gamma}{2})X_f$$

holds for all $t$. Now we define a "super time slot", each super time slot consists of $M$ time slots, where $M$ is a large number such that $M > \frac{2M_\delta}{\gamma X_f}$. We index the super time slots using $T$. During super time slot $T$ such that $TM \geq t_\delta$, the number of arrivals to $Q_f^{n_f^{i+1}}$, denoted by $a_f^{n_f^{i+1}}(T)$, satisfies the following inequality

$$X_f M - M_\delta \leq a_f^{n_f^{i+1}}(T) \leq X_f M + M_\delta$$

because otherwise, the number of packets that belong to flow $f$ and are queued at the up-streaming nodes of node $i+1$ is more than $M_\delta$ at time $M(T+1)$ (when the first inequality fails) or at time $MT$ (when the second inequality fails). According to the condition that $M > \frac{2M_\delta}{\gamma X_f}$, we further have that

$$a_f^{n_f^{i+1}}(T) \leq (1+\gamma/2)X_f M$$

for all $T$ such that $MT \geq t_\delta$.

Now consider super time slot $T+1$ (recall that $MT \geq t_\delta$). The estimates $\tilde{X}_f^{n_f^{i+1}}(t)$ for $t$ such that $M(T+1) \leq t < M(T+2)$ satisfies

$$\tilde{X}_f^{n_f^{i+1}}(t) > \left(1+\frac{\gamma}{2}\right)X_f,$$

which implies that the available service rate during super time slot $T+1$ satisfies

$$\tilde{X}_f^{n_f^{i+1}}(T) > \left(1+\frac{\gamma}{2}\right)X_f M \geq a_f^{n_f^{i+1}}(T),$$

which implies that the available service rate for the per-flow queue $Q_f^{n_f^{i+1}}$ during super time slot $T+1$ is always greater than the arrivals at super time slot $T$. In other words, under the proposed scheme, the arrival packets at super time slot $T$ can always be completely served at the end of super time slot $T+1$. Therefore there exists an $M_{i+1}$ value such that $Q_f^{n_f^{i+1}}(t) \leq M_{i+1}$ for all $f$, $i$ and $t$. The lemma follows from the induction principle. ■

Based on Lemma 1 and Lemma 2, we have the following theorem.

*Theorem 3:* Given a set of flows with constant arrival rates $\{X_f\}$ such that $\{(1+\epsilon)X_f\}$ is supportable for some $\epsilon > \gamma$, all queues under the self-regulated MaxWeight algorithm are bounded.

## V. Conclusion

In this paper, we considered scheduling in multihop wireless networks,and proposed the self-regulated MaxWeight scheduling that does not require the exchange of queue-length information among neighboring nodes, hence completely eliminates the communication overhead when combined with recent CSMA-based scheduling algorithms.

## Acknowledgement

Research supported by NSF CNS-0831756, CNS-0953165, and CCF-0914912 and the DTRA grants HDTRA1-08-1-0016 and HDTRA1-09-1-0055.

## References


[1] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Automat. Contr.*, vol. 4, pp. 1936–1948, December 1992.

[2] X. Lin, N. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1452–1463, 2006.

[3] L. Georgiadis, M. J. Neely, and L. Tassiulas, *Resource Allocation and Cross-Layer Control in Wireless Networks*. NOW, 2006, foundations and Trends in Networking.

[4] L. Bui, R. Srikant, and A. L. Stolyar, "Novel architectures and algorithms for delay reduction in back-pressure scheduling and routing," *Proc. IEEE Infocom. Mini Conf.*, 2009.

[5] L. Ying, S. Shakkottai, and A. Reddy, "On combining shortest-path and back-pressure routing over multihop wireless networks," in *Proc. IEEE Infocom.*, Rio de Janeiro, Brazil, 2009.

[6] L. Huang and M. Neely, "Delay reduction via lagrange multipliers in stochastic network optimization," in *Proc. Intl. Symposium on Modelling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 2009.

[7] L. Jiang and J. Walrand, "Distributed CSMA algorithm for throughput and utility maximization in wireless networks," in *Proc. Ann. Allerton Conf. Communication, Control and Computing*, 2008.

[8] J. Ni and R. Srikant, "Distributed CSMA/CA algorithms for achieving maximum throughput in wireless networks," *Technical Report*, 2009.

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

[10] C. Humes, "A regulator stabilization technique: Kumar-seidman revisited," *IEEE Trans. Automat. Contr.*, vol. 39, no. 1, pp. 191–196, Janunary 1994.

[11] X. Wu and R. Srikant, "Regulated maximal matching: A distributed scheduling algorithm for multi-hop wireless networks with node-exclusive spectrum sharing," in *Proc. Conf. on Decision and Control*, 2005, pp. 5342–5347.