Joint Resource Allocation and Admission Control in Wireless Mesh Networks

Tae-Suk Kim * Yong Yang † Jennifer C. Hou † and Srikanth V. Krishnamurthy *
* Department of Computer Science and Engineering
University of California, Riverside
{tskim, krish}@cs.ucr.edu
† Department of Computer Science
University of Illinois at Urbana-Champaign
{yang25, jhou}@cs.uiuc.edu

Abstract—**Many next generation applications (such as video flows) are likely to have associated *minimum* data rate requirements to ensure satisfactory quality as perceived by end-users. While there have been prior approaches on supporting quality-of-service (QoS) in mesh networks, they have largely ignored the issues that arise due to *self-interference*, the interference between different link layer transmissions of a single flow along a multi-hop path. In this paper, we develop a framework to address the problem of maximizing the *aggregate* utility of traffic flows in wireless mesh networks, with constraints imposed *both* due to self-interference and minimum rate requirements. The output of our framework is a schedule that dictates which links are to be activated simultaneously, and provides specifications of the resources associated with each of those links. Utilizing the proposed framework as a basis, we build an admission control module that intelligently manages the resources among the flows in the network and admits as many new flows as possible without violating the QoS of the existing flows. We provide numerical results to demonstrate the efficacy of our framework.**

I. Introduction

For many applications such as video, a minimum rate requirement has to be met in order to ensure satisfactory end-to-end quality [1]. In a shared wireless mesh network, ensuring that application demands are met requires the following inter-dependent functionalities: (a) **rate or congestion control:** control the rates at which the various traffic sources sharing the network inject traffic and (b) **resource allocation:** allocate resources to the different connections such that the minimum rate requirements of each connection are met and (c) **admission control:** ensure that newly admitted connections do not cause a violation of the minimum rate requirements of existing flows. Our

goal in this work is to design a framework towards jointly facilitating these functionalities.

The problem of resource allocation and congestion control in wired networks has received a lot of attention. In their seminal work, Kelly *et al.* [2] have modelled the problem of flow control as an optimization problem where the objective is to maximize the aggregate utility of elastic traffic sources subject to capacity constraints on the links that compose the network. Inspired by Kelly's work, there has been follow up work [3]–[5], where TCP congestion control is modelled a convex optimization problem, the objective being the maximization of an aggregate user utility; they propose distributed primal-dual solutions to the problem.

There have been more recent efforts on extending the above congestion cotnrol framework to wireless networks (discussed later in Section II); examples include the work in [6]–[12]. In contrast with wireline networks, the capacity of a wireless link is not fixed. It varies dynamically and depends on the other flows in the network (interference effects). The dependencies between flows is regulated by the protocols at both the link and transport layers. However, these prior efforts do not consider the provision of quality-of-service in terms of *minimum rates* to the flows that share the network. More importantly, *self-interference* where the packets of a flow interfere with other packets that belong to the same flow along a multi-hop path is rarely considered. Our framework addresses the above two important factors.

In more detail, we propose a framework for maximizing the aggregate utility of traffic sources while adhering to the capacity contraints of each link and the minimum rate requirements imposed by each of the sources. The framework takes into account the self-interference of flows and assigns (a) channels (b) transmission power levels and (c) time slots to each link such that the above objective is achieved. It dictates the rates at which each traffic source will send packets such that the minimum rate requirements of all coexisting flows are met. If the minimum rate

requirements of all the flows cannot be met, the framework rejects a subset of flows (based on fairness considerations) and recomputes the schedule and allocates resources to each of the remaining flows.

The major contributions of our work are as follows:

- We formulate a utility maximization problem to facilitate rate control with QoS support. Our formulation accounts for constraints due to realistic wireless transmission characteristics, i.e., the impact of *self-interference*. As a solution framework to the formulated problem, we propose the distributed primal-dual method which leverages the cross decomposition technique in [13].
- The scheduling problem (a sub-problem of the utility maximization problem) that assigns the optimal channel and transmit power to set of active links is known to be NP-hard [14]. Given the intractability of the problem, we propose an efficient channel and power allocation algorithm, and provide its theoretical performance bound as compared to the optimal solution.
- For cases where the QoS requirements cannot be met if all connections were to be admitted, we propose three different policies which drop a sub-set of flows and reallocate the resources among the existing flows. The proposed framework and the dropping policies are then integrated towards building an admission control module which makes our framework viable in dynamic settings where flows enter and exit the network. To the best of our knowledge, we are the first to propose an admission control policy that jointly addresses joint congestion control and resource allocation in a multi-hop wireless network.
- We perform exhaustive numerical simulation experiments to evaluate our proposed framework. We observe that our framework achieves better resource utilization than with random channel assignment or with fixed transmit powers. Our studies also demonstrate the efficacy of our admission control module.

The remainder of the paper is organized as follows. Related work is described in brief in Section II. In Section III, we describe the system model being considered. We formulate the problem for the rate control with QoS requirements for wireless mesh networks (WMNs) in Section IV. Our resource allocation framework and our admission control framework are described in Section V and Section VI, respectively. The performance evaluation of the proposed framework is detailed in Section VII. We conclude the paper in Section VIII.

II. RELATED WORK

In [2], Kelly *et al.*, model flow control in a wireline network as an optimization problem. Their objective is to maximize the aggregate utility of a multiplicity of elastic traffic sources. The work has been a basis for analyzing various transport-level (including TCP-based) congestion control algorithms. Follow up work appears in [3]–[5]. Recently, there has been a lot of research activity on extending the above congestion control framework to wireless networks. In contrast to a wireline link, the capacity of a link in wireless networks is not fixed. As discussed earlier, it depends on the interference due to other flows, which in turn is regulated by protocols at various layers. Thus, congestion control in wireless networks has cross layer dependencies.

Using mathematical decomposition techniques the cross-layer optimization problem of congestion control can be decomposed into two sub-problems: the rate control problem to be solved at transport layer and the scheduling problem at the lower link layer; the latter is tightly related to the underlying resources to be managed. There have been various approaches that have been proposed for the two layers independently. In particular, congestion control with power control has been studied in [15]. Link scheduling with contention control has been looked at in [6]–[9]. [16] considers the joint impact of link scheduling and routing. Soldati *et al.*, formulate link scheduling with power control as an optimization problem [17]. Channel assignment, routing, and link scheduling has been considered in [18] while link scheduling, routing and power control are considered in [19]. Resource management at the lower layers has been considered in [10]–[12]. Design of scheduling algorithms and their performance evaluations appear in [20]–[23].

None of the above efforts however, consider the problem of resource allocation with QoS support in terms of providing a minimum data rate to flows, in the presence of *self-interference* in mesh networks. In other words, they ignore the constraints that arise due to competition among the packets belonging to the same flow that spans multiple wireless hops. This effect is taken into account in our work.

III. SYSTEM MODEL

We consider a pre-planned WMN consisting of a set of stationary wireless nodes (routers) connected by a set L of unidirectional links. Some of the nodes are assumed to act as gateways to the Internet. Each node is equipped with a single network interface card (NIC) and is associated with one of C orthogonal (non-overlapping) channels for transmitting or receiving. A sender-receiver pair can communicate with each other only if both of them are tuned to the same channel. In this work dynamic channel switching is assumed to be possible with the NIC. Nodes operate in a half-duplex manner so that at any given time a node can either transmit or receive (but not both). The transmission power p_l on a link l is assumed to be

chosen in $[0, p_l^{max}]$. In order for the signal transmitted by a sender to be decoded properly at a receiver, the signal to interference and noise ratio (SINR) should be no less than a threshold β [1]. In addition, it is assumed that the network operates in a time-slotted mode; time is divided into slots of equal duration.

The network has S elastic traffic sources and each source s has an associated data rate r_s. We assume that each source s requires at the very least, a data rate r_s^{req} in order to satisfy its QoS requirement (among several possible QoS metrics, the minimum rate is the primary target of this work). Furthermore, the data rate that may be provided to s is assumed to be upper bounded by r_s^{max}; this corresponds to the peak sending rate of source s, r_s^{max}, and depends on the application requirements at s. For example, the maximum sending rate of a real-time application can be expected to be much lower than that of an elastic application; the latter can greedily consume any available bandwidth. Each source s has an associated utility function $U_s(r_s)$; the utility is assumed to directly reflect the QoS provided to source s when it is injecting packets into the network at a rate r_s. We assume the utility function to be positive, continuously differentiable, monotonically increasing and strictly concave over $[0, r_s^{max}]$. Our objective is then, to find the optimal resource allocation in terms of assigning channels, transmit powers, and time slots so as to maximize the sum of the sources' utilities; at the same time, their QoS requirements in terms of minimal rates have to be met. In the rest of the paepr, we interchangeably use L (links), C (channels), and S (sources) to denote both the corresponding set itself and the cardinality of the set.

IV. PROBLEM FORMULATION

In this section, we formulate the utility maximization problem with our desired objectives and constraints. The path that a source s uses in order to reach a gateway in the WMN is represented by a routing vector $\mathbf{V}$, the elements of which are given by:

$$v_{(l,s)} = \begin{cases} 1, & \text{If source } s \text{ uses link } l \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

for $l \in L$ and $s \in S$. We assume that the paths between the routers and gateways are pre-determined (in this paper, we focus on the impact of resource allocation).

We define a binary channel assignment vector $\mathbf{X}$ with elements $x_{(l,c)}$ defined by:

$$x_{(l,c)} = \begin{cases} 1, & \text{If link } l \text{ uses channel } c \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

for $l \in L$ and $c \in C$.

Since each node is equipped with a single NIC, the number of channels that can be assigned to a link is at most one. To this end, the following constraint should be satisfied for each link l:

$$\sum_{c=1}^{C} x_{(l,c)} \leq 1, \quad \forall l \in L. \quad (3)$$

Next, we impose constraints to account for the *self-interference* among links. In particular, each node can *either* send to or receive from other nodes at any time. Thus, two links that share a node are not permitted to be active simultaneously. To represent this condition formally, let $E(l)$ be the set of neighboring links which share either the sender or the receiver of link l. Then, in order for link l to be active in a time slot, the following constraint should be satisfied for link l:

$$\sum_{c=1}^{C} \left(x_{(l,c)} \sum_{e \in E(l)} \sum_{h=1}^{C} x_{(e,h)} \right) = 0, \quad \forall l \in L. \quad (4)$$

By forcing the product within the summation to be zero, we are essentially ensuring that no link that is adjacent to the considered link l is active at the same time as l.

The intersection of (1), (3) and (4) yields the set Π, of active links:

$$\Pi = \left\{ \mathbf{X} | x \in \{0, 1\} \bigcap \sum_c x_{(l,c)} \leq 1 \right. \\ \left. \bigcap \sum_c \left(x_{(l,c)} \sum_e \sum_h x_{(e,h)} \right) = 0, \forall l \in L \right\}. \quad (5)$$

Based on the assumptions on r_s and p_l (described in Section III), the following two sets are established for source rates and transmit powers, respectively:

$$\Psi = \{\mathbf{R} | r_s^{req} \leq r_s \leq r_s^{max}, \forall s \in S\}, \quad (6)$$

$$\Lambda = \{\mathbf{P} | 0 \leq p_l \leq p_l^{max}, \forall l \in L\}, \quad (7)$$

where $\mathbf{R}$ and $\mathbf{P}$ are the $S \times 1$ rate vector and $L \times 1$ power vector, respectively.

Unlike with links in a wired network, the capacity of a link in a wireless network is not fixed due to the shared nature of the wireless medium. We make the assumption that the interference experienced by a link can be modeled as a Gaussian random variable (when there a reasonably large number of interfering links, we can invoke the central limit theorem). Assuming that the channel is in addition, exposed to additive white gaussian noise (AWGN), the capacity of link l, h_l, can be expressed as

$$h_l = \frac{1}{T} \log(1 + K \ SINR_l), \quad (8)$$

where T is the symbol period, K is a constant depending on the modulation scheme used, and $SINR_l$ is the signal to interference and noise ratio on link l and is given by:

$$SINR_l = \frac{p_l g_{ll}}{\sum_{m \neq l \in L} \mathbf{X}_m \cdot \mathbf{X}_l^T p_m g_{lm} + \eta_l}, \quad (9)$$

where p_l is the transmit power of the sender on link l, g_{lm} is the link gain between the receiver on link l and the sender on link m, $\mathbf{X}_l$ is the lth row vector of $\mathbf{X}$, and η_l is the additive thermal white noise power. Note that the link capacity is a nonlinear function of the transmit powers $\mathbf{P}$ and the assigned channels $\mathbf{X}$; we denote the $L \times 1$ link capacity vector $(h_1, \ldots, h_L)^T$ by $\mathbf{H}(\mathbf{X}, \mathbf{P})$.

The target resource allocation in a WMN can then be formulated as the following utility maximization problem:

$$\max \quad \mathbf{1}^T \cdot \mathbf{U}(\mathbf{R}) \tag{10}$$
$$s.t \quad \mathbf{V} \cdot \mathbf{R} \preceq \mathbf{H}(\mathbf{X}, \mathbf{P})$$
$$\mathbf{X} \in \Pi$$
$$\mathbf{P} \in \Lambda$$
$$\mathbf{R} \in \Psi,$$

where $\mathbf{U}(\mathbf{R})$ is the $S \times 1$ utility function vector $(U_1(r_1), \ldots, U_S(r_S))^T$ and $\mathbf{1}$ is the $1 \times S$ unit vector. Note that the utility function can be varied depending on the fairness that we want to achieve. For example, $-(\log(r_i))^2/2$ for $U_i(r_i)$ $\forall i \in S$ with the additional constraints $r_i \leq r_j$, where the links j represent the one-hop neighbors of link i, achieves max-min fairness [24]. In this work we are interested in proportional fairness and therefore, we choose the function $\log(r_i)$ for $U_i(r_i)$ [25]; however, we point out that our framework is generic and can be applied with other utility functions.

V. OUR RESOURCE ALLOCATION FRAMEWORK

In this section, we present a framework to address the utility maximization problem, defined by (10). We start with introducing the application of traditional primal-dual method to the utility maximization problem.

A. Traditional Primal-Dual Approach to Our Utility Maximization Problem

At the first step, the primal-dual technique separates the problem defined in (10) into smaller sub-problems by introducing the Lagrange multipliers $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_L)$ with regard to the first constraint in (10) (the link capacity constraint); the λ-multiplied constraints are then shifted to the objective function to form the Lagrangian:

$$L(\lambda, \mathbf{R}, \mathbf{P}, \mathbf{X}) \tag{11}$$
$$= \mathbf{1}^T \cdot \mathbf{U}(\mathbf{R}) + \lambda \cdot (\mathbf{H}(\mathbf{X}, \mathbf{P}) - \mathbf{V} \cdot \mathbf{R})$$
$$= \left(\mathbf{1}^T \cdot \mathbf{U}(\mathbf{R}) - \lambda \cdot \mathbf{V} \cdot \mathbf{R}\right) + \lambda \cdot \mathbf{H}(\mathbf{X}, \mathbf{P}).$$

Note that λ_l can be interpreted to be the congestion price of link l. The original problem then becomes:

$$\max \quad L(\lambda, \mathbf{R}, \mathbf{P}, \mathbf{X}) \tag{12}$$
$$s.t \quad \mathbf{X} \in \Pi$$
$$\mathbf{P} \in \Lambda$$
$$\mathbf{R} \in \Psi.$$

Due to its separable structure, problem (12) can be decomposed into two sub-problems: the *congestion control problem* and the *scheduling problem*. The congestion control problem is defined by:

$$\max \quad \mathbf{1}^T \cdot \mathbf{U}(\mathbf{R}) - \lambda \cdot \mathbf{V} \cdot \mathbf{R} \tag{13}$$
$$s.t \quad \mathbf{R} \in \Psi.$$

The objective here is the maximization of the sum of each source's *utility gain* by choosing the optimal sending rate for each such source. This problem is typically solved by a congestion control mechanism at the transport layer (as with TCP).

The scheduling problem is given by

$$\max \quad \lambda \cdot \mathbf{H}(\mathbf{X}, \mathbf{P}) \tag{14}$$
$$s.t \quad \mathbf{X} \in \Pi$$
$$\mathbf{P} \in \Lambda.$$

Given λ, the problem is now to determine the *best* usage of the links that compose the network (i.e., transmission time schedule, transmit power and channel assignments). Note that a link will not be active if it is assigned zero power or has received no channel assignment. Both the PHY and link layers are involved when solving the scheduling problem. Let $q(\lambda)$ be the maximum value of problem (12). Then, the dual problem to the primal problem defined in (10), is

$$\min \quad q(\lambda) \tag{15}$$
$$\lambda \succeq \mathbf{0}.$$

Applying the traditioal distributed primal-dual approach to our problem involves determining source rates, resources (channel and transmit power), and link costs at each iteration k, as follows:

- The data rates of the sources are determined by

$$\mathbf{R}^{(k)} = \underset{\mathbf{R} \in \Psi}{\operatorname{argmax}} \left(\mathbf{1}^T \cdot \mathbf{U}(\mathbf{R}) - \lambda^{(k)} \cdot \mathbf{V} \cdot \mathbf{R}\right). \tag{16}$$

- The resources are determined by

$$\left(\mathbf{X}^{(k)}, \mathbf{P}^{(k)}\right) = \underset{\mathbf{X} \in \Pi, \mathbf{P} \in \Lambda}{\operatorname{argmax}} \lambda^{(k)} \cdot \mathbf{H}(\mathbf{X}, \mathbf{P}). \tag{17}$$

- The implicit link costs are updated by

$$\lambda^{(k+1)} = \left[\lambda^{(k)} + \gamma \left(\mathbf{V} \cdot \mathbf{R}^{(k)} - \mathbf{H}\left(\mathbf{X}^{(k)}, \mathbf{P}^{(k)}\right)\right)\right]^+, \tag{18}$$

where γ and $[\cdot]^+$ denote a positive step size and a projection onto the positive orthant, respectively. Note that the link cost update above is the result of applying the project gradient method [26] to (15). *However, a naïve application of the primal-dual method to (10) may not work properly for the following reasons.* **First,** the primal-dual approach implicitly assumes that all the links on the end-to-end path are simultaneously active while computing the optimal end-to-end rate for the paths. Incorporating

4

the *self-interference* constraints in (5), however, results in the activation of only a few links in each iteration, and consequently, the rates computed for most of the paths may simply be zero. **Second**, the primal-dual approach assumes that the scheduling problem (14) can be solved optimally. Given the characteristics of the set Π in (5), the problem is proven to be NP-Hard [14]. Thus, finding the optimum at every iteration will cost prohibitive levels of computational resources and time.

In order to address the first issue, we leverage the *cross decomposition* technique [13]. In a nutshell, using this technique we build a link schedule over multiple time slots, conforming to the constraints. This module is discussed in Section V-B. To address the second issue, we propose an efficient resource allocation algorithm in Section V-C.

B. The Cross Decomposition Approach

In order to leverage the cross decomposition technique, we reformulate (10) as

$$\max \quad \rho\left(\mathbf{H}\left(\mathbf{X}, \mathbf{P}\right)\right) \qquad (19)$$
$$s.t \quad \mathbf{X} \in \Pi$$
$$\mathbf{P} \in \Lambda,$$

where $\rho\left(\mathbf{H}\left(\mathbf{X}, \mathbf{P}\right)\right) = \{\max \ \mathbf{1}^T \cdot \mathbf{U}(\mathbf{R}) \ | \ \mathbf{V} \cdot \mathbf{R} \preceq \mathbf{H}\left(\mathbf{X}, \mathbf{P}\right) \ and \ \mathbf{R} \in \Psi\}$. For a fixed link capacity vector $\mathbf{H}\left(\mathbf{X}, \mathbf{P}\right)$ whose elements are all positive, $\rho\left(\mathbf{H}\left(\mathbf{X}, \mathbf{P}\right)\right)$ is solved by using the traditional primal-dual method introduced in Section V-A, and a corresponding link cost vector λ is obtained. Then, the schedule is updated by augmenting active links, which are found by solving the scheduling problem (14) with the obtained λ. Based on the augmented schedule, an average link capacity (using values up to the current time) is newly calculated and input into the problem for maximizing ρ. This procedure repeats until the rates have converged or problem has been classified as infeasible. The rationale behind the schedule update procedure is that λ is the subgradient to $\rho\left(\mathbf{H}\left(\mathbf{X}, \mathbf{P}\right)\right)$ at $\mathbf{H}\left(\mathbf{X}, \mathbf{P}\right)$. The convergence of the cross decomposition approach has been previously studied in [17]; it has been proven that the method converges faster than the mean value cross decomposition method in [27].

The primal-dual approach, revised with the proposed cross decomposition, is summarized as pseudo-code in Algorithm 1. Note here that during initialization, the link schedule in the first L slots is built using pure TDMA. Algorithm 1 terminates when either the sending rates of the sources converge to stable values or the problem is classified as infeasible. The sending rate for a source is deemed to have converged when the difference between the sending rates in two consecutive slots is less than a specified threshold ϵ, for σ consecutive time slots. A problem is classified as infeasible when a positive increase

in link cost is observed during χ consecutive slots for any link which carries data from sources with converged sending rates. The infeasible case will be discussed further in Section VI.

Algorithm 1 Proposed Primal-Dual Approach

1: **Initialization**: Schedule links using pure TDMA for the first L slots. $k \leftarrow L + 1$
2: **while** Rate has not converged or Problem has not been classified infeasible **do**
3:     Calculate $\rho\left(\overline{\mathbf{H}}\left(\mathbf{X}^{(k)}, \mathbf{P}^{(k)}\right)\right)$ by using the traditional primal-dual approach for $\overline{\mathbf{H}}\left(\mathbf{X}^{(k)}, \mathbf{P}^{(k)}\right)$, and let $\lambda^{(k)}$ be the obtained equilibrium link price;
4:     Calculate $\left(\mathbf{X}^{(k+1)}, \mathbf{P}^{(k+1)}\right)$ by solving the scheduling problem (14) for $\lambda^{(k)}$, and augment the schedule with the associated active links;
5:     Calculate the average link capacity $\overline{\mathbf{H}}\left(\mathbf{X}^{(k+1)}, \mathbf{P}^{(k+1)}\right) = \sum_{t=1}^{k+1} \mathbf{H}\left(\mathbf{X}^{(t)}, \mathbf{P}^{(t)}\right)/(k+1)$;
6:     $k \leftarrow k + 1$;
7: **end while**

C. Our Resource Allocation Approach

The proposed primal-dual approach requires the scheduling problem (14) to be solved at every iteration. For simplicity, we assume that (i) the $SINR$ on each link is much larger than 1, (ii) $T = 1$ and $K = 1$, and (iii) the thermal noise term is negligible (interference dominated setting). The scheduling problem (14) at iteration k is then expressed as:

$$\max_{\substack{\mathbf{X} \in \Pi \\ \mathbf{P} \in \Lambda}} \sum_l \lambda_l^{(k)} \log\left(\frac{p_l g_{ll}}{\sum_{m \neq l \in L} \mathbf{X}_m \cdot \mathbf{X}_l^T p_m g_{lm}}\right)$$
$$= \max_{\substack{\mathbf{X} \in \Pi \\ \mathbf{P} \in \Lambda}} \sum_l \lambda_l^{(k)} \left(\log\left(p_l g_{ll}\right) - \log\left(\sum_{m \neq l \in L} \mathbf{X}_m \cdot \mathbf{X}_l^T p_m g_{lm}\right)\right).$$
$$(20)$$

Unfortunately, solving (20) is not straightforward. The major difficulties arise from the fact that it requires a combinatorial decision in terms of channel and power assignments; this is known to be NP-Hard [14]. Thus, finding the optimum at every iteration will cost prohibitive levels of computational resources and time. Given this, we propose an efficient two phase approach towards finding an approximate solution to (20). In the first phase, channels are assigned to links as per a simple heuristic, and the optimal powers are calculated for the links in the second phase.

1) Channel Assignment: The proposed algorithm allocates channels in a way that (a) *self-interference* is avoided and (b) *co-channel interference* levels among links that use the same channel are kept as low as possible. With our algorithm, links with higher costs are assigned higher priorities in terms of channel assignment over the links with lower cost. This is because links with higher costs

suffer from higher levels of congestion and thus, scheduling these links is harder. The proposed channel assignment algorithm starts by sorting links in the descending order of their link costs. Then, channels are assigned to the links in that order. The proposed algorithm avoids *self-interference* by *not* assigning a channel to any link whose incident links have already been assigned channels. In other words, a link is eligible for activation only if it has no *active* neighbor links. In order to alleviate the effects of *co-channel interference*, the channel that is assigned to a link is selected based on the sum of link gains between all the interfering senders using the same channel and the receiver of the link. This sum is calculated for each of the channels and the channel with the least associated value is selected for the link. The proposed channel assignment is summarized in Algorithm 2, where we define $Q(c)$ to be the set of links that are assigned channel c. An active link is then assigned a transmit power based on our power assignment algorithm discussed next.

Algorithm 2 Channel Assignment

1: **Initialization**: $x_{(l,c)} \leftarrow 0$, and $Q(c) \leftarrow \emptyset$, $\forall\ l \in L$ and $\forall$ $c \in C$;
2: Sort links by descending order of λ, and label i-th link in the sorted list as l_i;
3: **for** $j = 1$ to L **do**
4:    **if** $\sum_e \sum_c x_{(e,c)} = 0$, for $e \in E(l_j)$ **then**
5:       Calculate $d_c = \sum_{q \in Q(c)} g_{ql_j}$, $\forall\ c \in C$;
6:       Allocate channel $c_{l_j} = argmin_c\{d_1, d_2, \ldots, d_C\}$ to link l_j;
7:       Assign l_j to $Q(c_{l_j})$;
8:    **end if**
9: **end for**

---

2) Power Control: With channel assignment as described in the previous subsection, we have $\mathbf{X}^{(k)}$ specified at the beginning of slot k. Let m' be a member of the set of links satisfying $X_{m'}^{(k)} \cdot \left(X_l^{(k)}\right)^T = 1$ for $m' \neq l$. The scheduling problem (20) is then reduced to

$$\max_{\mathbf{P} \in \Lambda} \sum_l \lambda_l^{(k)} \left(\log\left(p_l g_{ll}\right) - \log\left(\sum_{m'} p_{m'} g_{lm'}\right) \right). \quad (21)$$

The problem (21) is non-convex, and thus, we apply geometric programming [26] towards solving it. Geometric programming transforms the *seemingly* non-convex problem into a convex problem through a logarithmic change of variables. Let $\hat{p}_l = \log p_l$ for $\forall l \in L$. Then, (21) can be written as:

$$\max_{\hat{\mathbf{P}} \in \hat{\Lambda}} \sum_l \lambda_l^{(k)} \left(\log\left(e^{\hat{p}_l} g_{ll}\right) - \log\left(\sum_{m'} e^{\hat{p}_{m'}} g_{lm'}\right) \right)$$
$$= \max_{\hat{\mathbf{P}} \in \hat{\Lambda}} \sum_l \lambda_l^{(k)} \left(\hat{p}_l + \log\left(g_{ll}\right) - \log\left(\sum_{m'} e^{\hat{p}_{m'}} g_{lm'}\right) \right),$$
$$(22)$$

where $\hat{\Lambda} = \left\{ \hat{\mathbf{P}} | -\infty \leq \hat{p}_l \leq \log p_l^{max}, \forall\ l \in L \right\}$.

Note that the objective function in (22), for each link l, is a concave function; it consists of linear and concave terms (log-exp-sum is known to be convex [26]) and the sum of the concave functions is also a concave. The transformed problem (22) is thus a concave optimization problem for which solutions can be found with efficient techniques such as the interior point method [26]. After solving this optimization problem, the solution can be mapped back to the original space (using the relation $p_l = e^{\hat{p}_l}$).

3) Performance of Proposed Resource Allocation: Next, we analyze the proposed resource allocation strategy in terms of its convergence and efficiency. In particular, we compare the performance of our approach with that of an optimal schedule (produced by exhaustive search) and discuss known properties with regards to its convergence. We first begin with some preliminaries.

Preliminaries: The minimum SINR requirement for successful decoding (discussed earlier in Section III) dictates whether two distinct links using same channel can be active simultaneously. If the SINR values measured at the two receivers are simultaneously higher than β, the two links can be active together; if not, the two links are classified as *interfering* links. $\mathbf{T}_l$ denotes the set of interfering links for link l; if link l is not active, some of the links in $\mathbf{T}_l$ can be active simultaneously on the same channel; if l is active, none of the links in $\mathbf{T}_l$ can be active. We define the *opportunity cost* O_l for link l to be the maximum number of links in $\mathbf{T}_l$ that can be scheduled to be active simultaneously. This implies that a transmission on link l deprives O_l other links in terms of being able to transmit.

To compute O_l, we first find $\mathbf{T}_l$ with the maximum cardinality. To do this, we assume that the transmitter on link l is transmitting with the maximum power p_l^{max}. Then, any link that is unable to meet the SINR threshold β even with the maximum transmit power, is a member of $\mathbf{T}_l$. Given $\mathbf{T}_l$, the opportunity cost for link l is said to be ω if there are ω links [2] in $\mathbf{T}_l$ (we call this set $\mathbf{T}_l^\omega = \{m_1, m_2, \cdots, m_\omega\}$) such that every link $m_j \in \mathbf{T}_l^\omega$ satisfies the SINR requirement:

$$SINR_{m_j} = \frac{p_{m_j} g_{m_j m_j}}{\sum_{m_i \neq m_j \in \mathbf{T}_l} p_{m_i} g_{m_j m_i} + \eta_{m_j}} \geq \beta. \quad (23)$$

In order to facilitate the computation of O_l, we use a simple but powerful concept from matrix theory. In the matrix form, the SINR requirements (23) can be expressed as:

$$(\mathbf{I} - \mathbf{F})\mathbf{P} \succeq \mathbf{u} \quad \text{and} \quad \mathbf{P} \succ \mathbf{0}, \quad (24)$$

---

[2]For the ease of presentation, we re-label the links as per the sequence in which they were made members of the set $\mathbf{T}_l^\omega$, instead of their original link numbers.

where, $\mathbf{I}$ is the unit vector of size $\omega \times \omega$, $\mathbf{P} = (p_{m_1}, p_{m_2}, \cdots, p_{m_\omega})$ is the column vector of transmit powers, $\mathbf{u} = (\frac{\beta \eta_{m_1}}{g_{m_1 m_1}}, \frac{\beta \eta_{m_2}}{g_{m_2 m_2}}, \cdots, \frac{\beta \eta_{m_\omega}}{g_{m_\omega m_\omega}})$ is the column vector of thermal noise powers scaled by the SINR threshold and link gains, and $\mathbf{F}$ is the $\omega \times \omega$ matrix with entries are:

$$f_{(j,i)} = \begin{cases} 0, & \text{If } j = i \\ \frac{\beta g_{m_j m_i}}{g_{m_i m_i}}, & \text{otherwise.} \end{cases} \quad (25)$$

According to the Perron-Frobenius theorem [28], if the Perron-Frobenius eigenvalue of $\mathbf{F}$, $\rho_{\mathbf{F}}$, is less than 1, i.e., $\rho_{\mathbf{F}} < 1$, then, there exists a vector $\mathbf{P} \succ \mathbf{0}$ (i.e., $p_{m_j} > 0$ for all j) such that $(\mathbf{I} - \mathbf{F})\mathbf{P} \succeq \mathbf{u}$. Applying the Perron-Frobenius theorem to all the combinations of links that belong to $\mathbf{T}_l$, the opportunity cost of link l, O_l, can be readily found. From among the opportunity costs of all the links in the network, we denote the maximum opportunity cost by Ω.

Performance relative to an optimal scheduler: The following proposition provides a performance bound on our proposed scheduling algorithm.

Theorem 1: The proposed scheduling algorithm achieves a performance ratio[3] of $\Delta/(\Omega + 2)\Theta$.

Proof: Let $\mathbf{X_g}(\in \Pi)$ (or respectively, $\mathbf{X}^*(\in \Pi)$) and $\mathbf{P_g}$ (or respectively, $\mathbf{P}^*$) denote the channel assignment and the power allocation resulting from the proposed algorithm (or respectively, the optimal scheduler algorithm). Then, in order to prove the proposition, we need to show that:

$$\lambda \cdot \mathbf{H}(\mathbf{X_g}, \mathbf{P_g}) \succeq \frac{\Delta}{(\Omega + 2)\Theta} \cdot \lambda \cdot \mathbf{H}(\mathbf{X}^*, \mathbf{P}^*). \quad (26)$$

Consider the link l with the largest λ, which is chosen first by the proposed channel assignment algorithm. If this link is scheduled on channel c, then no link in $\mathbf{T}_l$ can be scheduled simultaneously, with link l on channel c. On the other hand, the optimal scheduler may select at most Ω links in its schedule in lieu of link l. Due to *self-interference* constraints, if link l is scheduled, the links incident on link l cannot be scheduled. Thus, the optimal scheduler can include at most 2 other links (i.e., one incident on the sender and one incident on the receiver) in addition to the previously chosen Ω links.

Let Δ and Θ denote the minimum link capacity achieved, from among the links scheduled by the proposed algorithm and the potentially possible maximum link capacity, respectively. Θ is achievable when only the single link whose link gain is the largest as compared to all the other links, transmits using the maximum power. The maximum capacity gain with the optimal schedule is no greater than $\Theta/\Delta \cdot \mathbf{H}(\mathbf{X_g}, \mathbf{P_g})$. The value of $\lambda \cdot \mathbf{H}(\mathbf{X}^*, \mathbf{P}^*)$ (with the optimal schedule), which consists of at most $(\Omega + 2)$

[3]This is the maximum ratio by which the results of an approximation algorithm may differ from the optimal solution.

links, is therefore at most $((\Omega + 2)\Theta/\Delta) \cdot \lambda \cdot \mathbf{H}(\mathbf{X_g}, \mathbf{P_g})$. We can then remove all the interfering links and the incident links found above from the list of links to be scheduled with both the proposed and optimal schedulers. Then, applying the same argument inductively on the links that are selected by the proposed scheduling algorithm, the proof can be established. ∎

Convergence: Lin *et al.*, [20] study the convergence of approximation algorithms with a performance ratio $\alpha \in (0, 1]$. They demonstrate that the user rate vector $\mathbf{R}$ eventually converges to a small neighborhood of the optimal rate. However, in many cases, the convergence may lead to inadequate QoS (in terms of rate) for some of the flows. In such a case, some of the flows will have to be *dropped* in order to satisfy the QoS for the other flows.

D. Implementation

The proposed cross decomposition approach can be implemented either in a centralized fashion or in a distributed manner. For centralized implementation, a coordinator (in all likelihood one of gateway nodes) is responsible for running Algorithm 1 and notifying the nodes of the transmission schedule and the channel and transmit power assignments. The original primal-dual approach can be implemented in distributed manner with guaranteed convergence [3], [17]. This is attributable to the decomposable structure of the utility maximization problem. The proposed resource allocation mechanism benefits from this structure as well and thus, can be implemented in a decentralized way. In this case, each source is responsible for making a decision on its sending data rate as per Equation (16). Each transmitter n is also responsible for using the correct channel and power on its outbound links $L_n \subset L$ as per (17) and updating the cost of the links as per (18). The transmitter on each link l needs to periodically exchange information on its individual resource usage $(x_{(l,c)}, p_l)$ for $l \in L_n$ and $c \in C$, and its link cost λ_l, $l \in L_n$, with all the other nodes. Elaborating on the technical details of implementing such an approach is beyond the scope of this paper; however, a reliable flooding protocol (as with any link state routing approach) could be used.

VI. ADMISSION CONTROL

In this section we extend our primal-dual framework to support admission control handling dynamic settings where flows enter and exit network.

A. Handling Infeasible QoS requests

The proposed resource allocation framework attempts to achieve both fairness and the QoS requirements as specified by the utility maximization problem (10). However, in the first constraint of (10), if sum of QoS requirements of the various sources over a link exceeds the link capacity,

the link cost in (18) will not converge; it will increase continuously as we progressively go through time (in terms of slots) and this leads to an infeasible solution. In such a scenario, the only solution would be to gradually *drop* a sub-set of the sources until the rate requirements of the rest of the sources are met. The objective could be to drop as few sources as possible.

For any link, if the link cost increases by γ per slot during χ consecutive slots, a schedule is considered to be infeasible. In order to handle this infeasible scenario, we first solve (10) with r_s^{req} relaxed to 0 for every source $s \in S$. Each source s whose assigned rate meets its QoS requirement (i.e., $r_s \geq r_s^{req}$) is put into a set G; the other nodes are put into a set $\overline{G}$. Members in $\overline{G}$ are the sources that are candidates for being dropped.

We consider three dropping policies or rules. As per our *first policy*, we choose the source for which, the difference or gap between the required rate and the assigned rate is the maximum. The rule is referred to as MG (for maximum gap). After removing the above source from $\overline{G}$, we solve the *relaxed* form of (10) again with the sources in $G \bigcup \overline{G}$. The process is repeated until no sources are left in $\overline{G}$, i.e., until there is no active source for which the QoS requirements are not met. The proposed *resource adjustment* method is summarized in Algorithm 3. We consider two additional policies: MR and MRG. Policy MR (for maximum rate) selects the source $k \in \overline{G}$ for which, the QoS requirement is the maximum, i.e., $k = argmax_{s \in \overline{G}} r_s^{req}$. If there are multiple sources with the same maximum rate, one of these sources is randomly selected and dropped. With MRG, MR is applied first and subsequently, in the case of a tie MG is applied.

Algorithm 3 Adaptive Resource Allocation with MG

1: **Initialization**: $G \leftarrow \emptyset, \overline{G} \leftarrow \emptyset$
2: Perform Algorithm 1 on the utility maximization problem (10) with $r_s^{req} = 0, \forall s \in S$;
3: Put s into G such that $r_s \geq r_s^{req}$; Otherwise, put into $\overline{G}$;
4: **while** $\overline{G} \neq \emptyset$ **do**
5:     Remove k from $\overline{G}$ such that $k = argmax_{s \in \overline{G}}(r_s^{req} - r_s)$;
6:     Solve (10) with $r_s^{req} = 0, \forall s \in G \bigcup \overline{G}$;
7:     $G \leftarrow \emptyset, \overline{G} \leftarrow \emptyset$
8:     Put s into G such that $r_s \geq r_s^{req}$; Otherwise, put into $\overline{G}$;
9: **end while**

B. Admission Control

An admission control strategy is essential to provide protection to the sources that are currently being serviced. In other words, the QoS of existing flows in terms of a minimum rate (being currently provided) cannot be compromised in order to accommodate *new incoming* flows. Our resource allocation framework can be easily adapted to support admission control.

In more detail, the admission control process works as follows. Let us assume that new sources (possibly multiple), N, request services, each source with its own minimum rate specifications (as before). The set of existing sources is called E. First, we solve the utility maximization problem (10), with both the new and existing sources, by using Algorithm 1. If the rates requested are feasible, then all the new connections are allowed to join the network. If the requested rates are infeasible, then the Algorithm 3 is invoked. However, in lieu of dropping the flow with the largest QoS requirement in $E \bigcup N$, we drop the source with the largest QoS requirement in N. The process is repeated until all the sources in N are either *admitted* or *dropped*. We summarize our approach in Algorithm 4 below.

Algorithm 4 Admission Control

1: **Initialization**: $E \leftarrow \emptyset, N \leftarrow \emptyset$
2: Put the existing sources into E and the new one(s) into N;
3: Perform Algorithm 1 on (10) for the sources in $E \bigcup N$;
4: **if** (10) is *infeasible* **then**
5:     Run Algorithm 3 and Get G;
6:     **while** $E \bigcap G \neq E$ and $N \neq \emptyset$ **do**
7:         Reject a new source with the maximum QoS requirement in N;
8:         Run Algorithm 3 and Get G;
9:     **end while**
10:     **if** $E \bigcap G = E$ **then**
11:         Admit all new source(s) in N;
12:     **end if**
13: **else**
14:     Admit all new source(s) in N;
15: **end if**

VII. NUMERICAL EVALUATIONS

In this section, we evaluate the performance of our proposed framework via extensive numerical simulations.

A. Simulation Setup

For the purposes of evaluation, we consider a typical mesh network with stationary wireless routers, deployed in an area of size of $800m \times 400m$; the wireless routers can serve as both access points (APs) for client nodes and relaying nodes for forwarding data received from neighboring nodes. The topology of WMN under consideration is shown in Figure 1; it consists of 10 wireless routers and one gateway. A solid line between any two routers indicates the link over which the data between the routers is transmitted. Six routers, W1, W6, W7, W8, W9, and W10, are considered as the traffic sources; their routes to the gateway (G) are given by:

- Source #1: W1 → G
- Source #2: W6 → W2 → G
- Source #3: W7 → W3 → W1 → G

8

TABLE I
PERFORMANCE TRACES OF VARIOUS ADMISSION SCHEMES

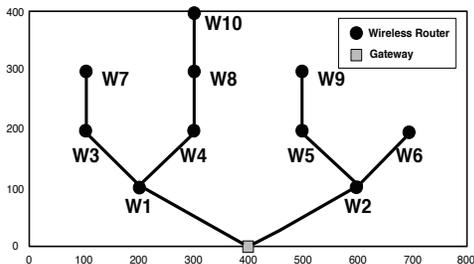| new request / schemes	No Admission Control	RC-FP with MR	Proposed with MR	Proposed with OPT
W7(0.29) & W1(0.40)	(N)W7: .510 (**S**) (N)W1: 1.0 (**S**)	(N)W7: Rejected (N)W1: Rejected	(N)W7: 0.510 (**S**) (N)W1: 1.0 (**S**)	(N)W7: 0.510 (**S**) (N)W1: 1.0 (**S**)
W8(0.30) & W10(0.29)	(N)W8: .255 (**NS**) (N)W10: .255 (**NS**) (E)W7: .255 (**NS**) (E)W1: .510 (**S**)	(N)W8: Rejected (N)W10: Rejected	(N)W8: Rejected (N)W10: .339 (**S**) (E)W7: .339 (**S**) (E)W1: .677 (**S**)	(N)W8: .339 (**S**) (N)W10: Rejected (E)W7: .339 (**S**) (E)W1: .677 (**S**)
W9(0.40) & W6(0.39)	(N)W9: .381 (**NS**) (N)W6: .381 (**NS**) (E)W7: .277 (**NS**) (E)W1: .382 (**NS**) (E)W8: .278 (**NS**) (E)W10: .278 (**NS**)	(N)W9: Rejected (N)W6: Rejected	(N)W9: .400 (**S**) (N)W6: .392 (**S**) (E)W7: .392 (**S**) (E)W1: .400 (**S**) (E)W10: .392 (**S**)	(N)W9: .400 (**S**) (N)W6: .392 (**S**) (E)W7: .392 (**S**) (E)W1: .400 (**S**) (E)W8: .392 (**S**)

Fig. 1. Network topology

- Source #4: W8 → W4 → W1 → G
- Source #5: W9 → W5 → W2 → G
- Source #6: W10 → W8 → W4 → W1 → G

We adopt the simple but widely used signal propagation model where, the gain on a link spanning a distance d is inversely proportional to d^4 [29]. Links use one of 3 non-overlapping orthogonal channels (as with 802.11g [30]). The maximum transmit power for each link is set to 1, and the maximum sending rate for each of the sources is set to 1.

B. Evaluating our admission control strategy

To evaluate the proposed admission control strategy, we consider a scenario where service requests with minimum rate requirements from the traffic sources are generated dynamically. The requests from W7 and W1 arrive first. Later, the requests for W8 and W10 arrive together followed by those of W9 and W6. An admission control policy based on the proposed resource allocation framework and the dropping rule MR is compared with policies based on three other different combinations of resource allocation strategies and dropping rules: (a) the proposed resource allocation, no dropping, and no admission control (every new source is accepted), (b) a random channel assignment strategy with multiple channels and fixed power (RC-FP) and MR and (c) the proposed resource allocation and the optimal dropping (OPT). We find the

best dropping policy (OPT) using exhaustive search; in this small topology, this can be done in a reasonable time. Table I shows the results with each admission control policy. As before, N and E represent the new and the existing sources. We use the notation **S** and **NS** to indicate whether a source is **S**atisfied or **N**ot **S**atisfied in terms of its QoS requirement.

As seen in Table I, without admission control, the QoS requirements of the existing flows are violated (they are now classified as **NS**). RC-FP results in the rejection of all the new requests even when it works in conjunction with the proposed dropping rule and the admission control strategy. This is directly attributable to the poor resource allocation with RC-FP. This demonstrates that without smart resource allocation, admission control may not be very useful. The admission control with the proposed resource allocation framework and the dropping rule MR, supports five sources with their QoS requirements satisfied (all except W8). It rejects W8 since accommodating both W8 and W10 causes a violation of QoS for W7. We observe that the scheme achieves a performance that is comparable to that with the optimal policy, OPT.

VIII. CONCLUSIONS

In this paper, we develop a framework for supporting QoS in wireless mesh networks. The framework maximizes the aggregate utility of flows taking into account constraints that arise due to self-intereference (wireless channel imposed constraints) and minimum rate requirements of sources (QoS requirements). If a solution is not feasible, the framework selectively drops a few of the sources and redistributes the resources among the others in a way that their QoS requirements are met. The proposed framework readily leads to a simple and effective admission control mechanism. We demonstrate the efficacy of our approach with numerical results. We also theoretically compute performance bounds with our network, as compared with an optimal strategy.

REFERENCES

[1] S. Ahmed and A. Ramani. Exploring the Requirements for QoS in Mobile Ad hoc Networks. *Journal of Information & Communication Technology*, 1(2):1–9, 2007.

[2] F. Kelly. Charging and Rate Control for Elastic Traffic. *European Transactions on Telecommunication*, 8:33–37, 1997.

[3] S. Low and D. Lapsley. Optimization Flow Control.I: Basic Algorithm and Convergence. *IEEE/ACM Transactions on Networking*, 7(6):861–874, 1999.

[4] H. Yaiche, R. Mazumdar, and C. Rosenberg. A Game Theoretic Framework for Bandwidth Allocation and Pricing in Broadband Networks. *IEEE/ACM Transactions on Networking*, 8(5):667–678, 2000.

[5] S. Kunniyur and R. Srikant. End-to-End Congestion Control Schemes: Utility Functions, Random Losses and ECN Marks. In *IEEE INFOCOM*, 2000.

[6] X. Wang and K. Kar. Cross-Layer Rate Control for End-to-End Proportional Fairness in Wireless Networks with Random Access. In *ACM MobiHoc*, 2000.

[7] J. Lee, M. Chiang, and R. Calderbank. Jointly Optimal Congestion and Contention Control in Wireless Ad Hoc Networks. *IEEE Communications Letters*, 10(3):216–218, 2006.

[8] J. Zhang and D. Zheng. A Stochastic Primal-Dual Algorithm for Joint Flow Control and MAC Design in Multi-hop Wireless Networks. In *CISS*, 2006.

[9] L. Bui, A. Eryilmaz, R. Srikant, and X. Wu. Joint congestion control and distributed scheduling in multihop wireless networks with a nodeexclusive interference model. In *IEEE INFOCOM*, 2006.

[10] M. Neely, E. Modiano, and C. Rohrs. Dynamic Power Allocation and Routing for Time Varying Wireless Networks. In *IEEE INFOCOM*, 2003.

[11] R. Cruz and A. Santhanam. Optimal Routing, Link Scheduling and Power Control in Multi-hop Wireless Networks. In *IEEE INFOCOM*, 2003.

[12] T. ElBatt and A. Ephremides. Joint Scheduling and Power Control for Wireless Ad Hoc Networks. *IEEE Transactions on Wireless Communications*, 3(1):74–85, 2004.

[13] K. Holmberg. *Primal and dual decomposition as organizational design: Price and/or resource directive decomposition*. Technical Report LiTH-MAT-R-94-03, Dept. of Mathematics, Linköping Institute of Technology, Linköping, Sweden, 1994.

[14] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. In *ACM MobiCom*, 2003.

[15] M. Chiang. Balancing Transport and Physical Layer in Multihope Wireless Networks: Jointly Optimal Congestion and Power Control. *IEEE Journal on Selected Areas in Communications*, 23(1):104–116, 2005.

[16] A. Eryilmaz and R. Srikant. Joint Congestion Control, Routing and MAC for Stability and Fairness in Wireless Networks. *IEEE Journal on Selected Areas in Communications*, 24(8):1514–1524, 2006.

[17] P. Soldati, B. Johansson, and M. Johansson. Proportionally fair allocation of end-to-end bandwidth in STDMA wireless networks. In *ACM MobiHoc*, 2006.

[18] X. Lin and S. Rasool. A Distributed Joint Channel-Assignment, Scheduling and Routing Algorithm for Multi-Channel Ad-hoc Wireless Networks. In *IEEE INFOCOM*, 2007.

[19] M. Johansson and L. Xiao. Scheduling, Routing and Power Allocation for Fairness in Wireless Networks. In *IEEE VTC*, 2004.

[20] X. Lin and N. B. Shroff. The Impact of Imperfect Scheduling on Cross-Layer Congestion Control in Wireless Networks. *IEEE/ACM Transactions on Networking*, 14(2):302–315, 2006.

[21] G. Sharma, R. R. Mazumdar, and N. B. Shroff. Maximum Weighted Matching with Interference Constraints. In *IEEE International Workshop on Foundations and Algorithms For Wireless Networking*, 2006.

[22] X. Wu and R. Srikant. Regulated maximal matching: A distributed scheduling algorithm for multi-hop wireless networks with node-exclusive spectrum sharing. In *IEEE CDC*, 2005.

[23] P. Chaporkar, K. Kar, and S. Sarkar. Throughput Guarantees in Maximal Scheduling in Wireless Networks. In *Allerton Conference on Communication, Control and Computing*, 2005.

[24] X. Wang and K. Kar. Distributed Approaches for Proportional and Max-Min Fairness in Random Access Adhoc Networks. In *CISS*, 2006.

[25] F. Kelly, A. Maulloo, and D. Tan. Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability. *Journal of Operations Research Society*, 49(3):237–252, 1998.

[26] S. Boyd. *Convex Optimization*. Cambridge University Press, 2004.

[27] K. Holmberg and K. Kiwiel. Mean value cross decomposition for nonlinear convex programs. *Optimization Methods and Software*, 21(3):401–417, 2006.

[28] E. Seneta. *Nonnegative Matrices and Markov Chains*. Springer, 1981.

[29] J. Kivinen, X. Zhao, and P. Vainikainen. Empirical characterization of wideband indoor radio channel a 5.3 ghz. *IEEE Transactions on Antenna and Propagation*, 49(8):1192–1203, 2001.

[30] IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specications. ISO/IEC 8802-11:2005/Amd.4:2006(E).