Resource Allocation for QoS Support in Wireless Mesh Networks

Tae-Suk Kim, Yong Yang, Jennifer C. Hou, Fellow, IEEE, and Srikanth V. Krishnamurthy, Fellow, IEEE

Abstract—Many next generation applications (such as video flows) are likely to have associated minimum data rate requirements in order to ensure satisfactory quality as perceived by end-users. In this paper, we develop a framework to address the problem of maximizing the aggregate utility of traffic flows in a multi-hop wireless network, with constraints imposed both due to self-interference and minimum rate requirements. The parameters that are tuned in order to maximize the utility are (i) transmission powers of individual nodes and (ii) the channels assigned to the different communication links. Our framework is based on using a cross-decomposition technique that takes both inter-flow interference and self-interference into account. The output of our framework is a schedule that dictates what links are to be activated in each slot and the parameters associated with each of those links. If the minimum rate constraint cannot be satisfied for all of the flows, the framework intelligently rejects a sub-set of the flows and recomputes a schedule for the remaining flows. We also design an admission control module that determines if new flows can be admitted without violating the rate requirements of the existing flows in the network. We provide numerical results to demonstrate the efficacy of our framework.

Index Terms—Wireless mesh network, QoS, congestion control, resource allocation, admission control.

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 modeled 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 modeled a convex optimization problem, the objective being the maximization of an aggregate user utility; in these efforts distributed primal-dual solutions to the problem are proposed.

There have been more recent efforts on extending the above congestion control framework to wireless networks (discussed later in Section II); examples include the work in [6]–[9] and [10]–[13]. In contrast with wireline networks, the capacity of a wireless link is not dependent on other flows in the network but on other flows that use links on the same channel (and that are close enough) and external interference. 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 supporting minimum rates to the flows that share the network. More importantly, the QoS needs to be provided under conditions of self-interference, where the packets of a flow interfere with other packets that belong to the same flow along a multi-hop path. 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 constraints 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 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 rate control with QoS requirements for wireless mesh networks (WMNs) in Section IV. Our resource allocation framework and our admission control module 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 [14]. Link scheduling with contention control has been looked at in [6]–[9]. [15] considers the joint impact of link scheduling and routing. Soldati et al., formulate link scheduling with power control as an optimization problem [16]. Channel assignment, routing, and link scheduling has been considered in [17] while link scheduling, routing and power control are considered in [18]. Resources management at the lower layers has been considered in [10]–[13]. Design of scheduling algorithms and their performance evaluations appear in [19]–[22]. 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. Finally, to the best of our knowledge, we are the first to propose an admission control policy for ensuring that existing flows get their minimum desired rates.

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 have the ability to perform functions of the gateway, and one of them is selected to act as the gateway 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$^1$. 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 ratio (SINR) should be no less than a threshold $\beta$ $^2$. 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. 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 $s$, 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 paper, 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 first formulate the resource allocation problem with our desired objectives and constraints. We then introduce the traditional approach developed for wired networks; this forms the basis for our resource allocation framework described later in Section V.

The path that a source $s$ uses in order to reach a gateway in the WMN is represented by a routing vector $V$, the elements of which are given by:

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

for $l \in L$ and $s \in S$.

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

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

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 $l$:

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

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 $E(l)$

$^1$This may lead to a performance degradation as we progress along the hops of a path. Thus, the network layer needs to explicitly account for the switching costs when choosing routes. In this paper, the route from each source is assumed to be computed after accounting for such switching costs.

$^2$\(\beta\) depends on characteristics of the physical layer of the underlying system.
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 $l$:

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

By forcing the product within the summation to be zero, we essentially ensure 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 $\Pi$, of active links:

$$
\Pi = \left\{ x \in \{0,1\} \mid \sum_{c} x_{(l,c)} \leq 1 \right\} \cap \left\{ \sum_{c} \left( x_{(l,c)} \sum_{e} \sum_{h} x_{(e,h)} \right) = 0, \forall l \in L \right\}
$$

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:

$$
\Psi = \left\{ R | r_s^{req} \leq r_s \leq r_s^{max}, \forall s \in S \right\},
\Lambda = \left\{ P | 0 \leq p_l \leq p_l^{max}, \forall l \in L \right\},
$$

where $R$ and $P$ 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. 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 $h_l = \log(1 + K \cdot SINR_l)/T$, 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:

$$
SINR_l = \frac{p_l g_l}{\sum_{m \neq l \in L} X_m \cdot X_l^T g_l + \eta_l},
$$

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

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

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

where $U(R)$ is the $S \times 1$ utility function vector $(U_1(r_1), \ldots, U_S(r_S))^T$ and $1$ 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 [25]. In this work we are interested in proportional fairness and therefore, we choose the function $\log(r_i)$ for $U_i(r_i)$ [26]; 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 (8). As a solution method, the traditional primal-dual method [2] to the utility maximization problem can be conceived. The primal-dual technique separates the problem into smaller sub-problems by introducing the Lagrange multipliers $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_L)$ with regard to the first constraint in (8) (the link capacity constraint), and the original problem then becomes:

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

where $L(\lambda, R, P, X) = (1^T \cdot U(R) - \lambda \cdot V \cdot R) + \lambda \cdot H(X, P)$. Due to its separable structure, problem (9) can be decomposed into two sub-problems: the congestion control problem and the scheduling problem. The congestion control problem is defined by:

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

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

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

Given $\lambda$, 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.

\footnote{We wish to clarify here that the interference is modeled using a Gaussian random variable and does not necessarily have to be AWGN. We assume that the interference is not just received from those that are able to decode packets from a node (referred to as neighbors) but other nodes that are further away. If one accounts for the additive interference from such nodes, we expect the total number of signals to be fairly large (more than 10). Furthermore, with multi-path, additional copies of each individual signal are also received.}
The basic operation of the primal-dual method involves determining source rates and resources (channel and transmit power) distributively by solving (10) and (11) with given \( \lambda \), and updating \( \lambda \) based on the value of source rates and resources. However, a naive application of the primal-dual method to (8) 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 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 (11) can be solved optimally. Given the characteristics of the set \( \Pi \) in (5), the problem is proven to be NP-Hard [27]. Thus, finding the optimum at every iteration will cost prohibitive levels of computational resources and time.

To address the first issue, we leverage the cross decomposition technique [28]. 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-A. To address the second issue, we propose an efficient resource allocation algorithm in Section V-B.

### A. The Cross Decomposition Approach

In order to leverage the cross decomposition technique, we reformulate (8) as

\[
\max_{\lambda \in \Lambda} \rho(H(X, P)) \tag{12}
\]

\[
s.t. \quad X \in \Pi, \quad P \in \Lambda,
\]

where \( \rho(H(X, P)) = \{\max 1^T \cdot U(R) \mid V \cdot R \leq H(X, P) \text{ and } R \in \Psi\} \). For a fixed link capacity vector \( H(X, P) \) whose elements are all positive, \( \rho(H(X, P)) \) is solved by using the traditional primal-dual method, and a corresponding link cost vector \( \lambda \) is obtained. Then, the schedule is updated by augmenting active links, which are found by solving the scheduling problem (11), with the obtained \( \lambda \). 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 \( \rho \). This procedure repeats until the rates have converged or the problem has been classified as infeasible. The rationale behind the schedule update procedure is that \( \lambda \) is the sub-gradient to \( \rho(H(X, P)) \) at \( H(X, P) \). The convergence of the cross decomposition approach has been previously studied in [16]; it has been proven that the method converges faster than the mean value cross decomposition method in [29]. 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.

### B. Our Resource Allocation Approach

The proposed primal-dual approach requires the scheduling problem (11) 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 (11) at iteration \( k \) is then expressed as:

\[
\max_{X \in \Pi, P \in \Lambda} \sum_{l=1}^{K} \alpha_k \log \left( \frac{p(lg_l) + \sum_{m \neq l \in L} x_m \cdot x_l^T \cdot p_m g_{lm}}{\sum_{m \neq l \in L} x_m \cdot x_l^T \cdot p_m g_{lm}} \right) \tag{13}
\]

Unfortunately, solving (13) is not straightforward. The major difficulties arise from the fact that it requires a combinational decision in terms of channel and power assignments; this is known to be NP-Hard [27]. 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 (13). 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.

---

**Algorithm 1 Proposed Primal-Dual Approach**

1. **Initialization**: Schedule links using pure TDMA for the first \( L \) slots. \( k \leftarrow L + 1 \)
2. **loop**
   3. Calculate \( \rho(H(X(k), P(k))) \) by using the traditional primal-dual approach for \( H(X(k), P(k)) \), and let \( \lambda(k) \) be the obtained equilibrium link price;
   4. Calculate \( X(k+1), P(k+1) \) by solving the scheduling problem (11) for \( \lambda(k) \), and augment the schedule with the associated active links;
   5. Calculate the average link capacity \( H(X(k+1), P(k+1)) \) = \( \sum_{i=1}^{k+1} H(X(i), P(i))/k+1 \);
   6. \( k \leftarrow k + 1 \);
3. **end loop**
discussed next.

Algorithm 2 Channel Assignment

1: Initialization: \( x_{(e,c)} \leftarrow 0 \), and \( Q(c) \leftarrow \emptyset \), \( \forall \, l \in L \) and \( \forall \, e \in C \);
2: Sort links by descending order of \( \Lambda \), and label \( i \)-th link in the sorted list as \( l_i \);
3: for \( j = 1 \) to \( L \) do
4: if \( \sum_{e} x_{(e,c)} = 0 \), for \( e \in E(l_j) \) then
5: Calculate \( d_e = \sum_{q \in Q(c)} g_{l_j q}, \forall \, e \in C \);
6: Allocate channel \( c_l_j = \arg \min_{c} \{ d_1, d_2, \ldots, d_C \} \) 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 \( X{(k)} \) specified at the beginning of slot \( k \). Let \( m_l \) be a member of the set of links satisfying \( X{(k)}_m \), \( X{(k)}_{l}^T = 1 \) for \( m \neq l \). The scheduling problem (13) is then reduced to

\[
\max_{P \in \Lambda} \sum_{e} x_{(e,c)} \left( \log (p_{l_j} g_{l_j}) - \log \left( \sum_{m} p_{m_l} g_{m_l} \right) \right). \tag{14}
\]

The problem (14) is non-convex, and thus, we apply geometric programming [30] towards solving it. Geometric programming transforms the seemingly non-convex problem into a convex problem through a logarithmic change of variables. Let \( p_l = \log p_{l_l} \) for \( \forall \, l \in L \). Then, (14) can be written as:

\[
\max_{P \in \Lambda} \sum_{e} x_{(e,c)} \left( \log (e^{p_l} g_{l_j}) - \log \left( \sum_{m} e^{p_{m_l}} g_{m_l} \right) \right) \tag{15}
\]

= \max_{P \in \Lambda} \sum_{e} x_{(e,c)} \left( \hat{p}_l + \log (g_{l_j}) - \log \left( \sum_{m} e^{p_{m_l}} g_{m_l} \right) \right),

where \( \Lambda = \left\{ \hat{P} \mid -\infty < \hat{p}_l \leq \log p_{l_l}^{\text{max}}, \forall \, l \in L \right\} \).

Note that the objective function in (15), for each link \( l \), is a concave function; it consists of linear and concave terms (log-exp-sum is known to be convex [30]) and the sum of the concave functions is also a concave. The transformed problem (15) is thus a concave optimization problem for which solutions can be found with efficient techniques such as the interior point method [30]. 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 efficiency and convergence. In particular, we provide the complexity order of proposed resource allocation scheme, and compare the performance of our approach with that of an optimal schedule (produced by exhaustive search).

Complexity order: With the proposed channel assignment algorithm (see Pseudo-code for Algorithm 2), for every link the channel assignment is performed for its neighboring links; the number of such links is at most \( L \). The complexity order is thus \( O(L^2) \). The complexity of the proposed power control algorithm is of the same order of the solver to the Problem (15). The complexity of the applied interior point method is known to be \( O(L^{1/2}) \) [30]. As a consequence, the complexity of the proposed allocation method is \( O(L^{3/2}) \); thus, in essence, it is a polynomial time algorithm.

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\(^4\) of \( \Delta/(\Omega + 2) \Theta \), where \( \Delta \) and \( \Theta \) denote the minimum link capacity achieved, from among the links scheduled by the proposed algorithm and the potentially possible maximum link capacity, respectively. \( \Omega \) denotes the maximum opportunity cost over all the links in the network. The opportunity cost of a link is defined as the maximum number of links that can be scheduled to be active simultaneously if the link is not scheduled. Due to constraints on space, we refer to [31] for the proof and a discussion on properties with regards to convergence.

C. Implementation

The proposed approach requires both the transport (in terms of end-to-end rate allocation) and the physical layer (in terms of channel and power schedule) to be aligned. Coordination between the two layers can be implemented on different timescales [16]; end-to-end rate allocation (through TCP/AQM) on the fast time-scale and incremental channel and power updates on the slow time-scale. As demonstrated in [32], most of the common TCP/AQM variants can be interpreted as distributed methods for solving the optimization network flow problem (determines the end-to-end rates under fixed link capacity). Based on an initial schedule (a simple TDMA link schedule for the first \( L \) slots), we run the TCP/AQM scheme until convergence (this may require the schedule to be applied repeatedly). After rate convergence, each node reports the link prices associated with its incoming and outgoing links to gateway where the proposed resource allocation scheme is adopted. On receiving the link prices from the entire set of node, the gateway finds the channels and transmit powers by applying the resource allocation scheme proposed in Section V-B; it then augments the schedule. The procedure is then repeated with this revised schedule.

Implementing the proposed algorithm is viable following IEEE 802.16 standard [33]. A mesh frame consists of control and data subframes, and therefore two schedules are required for centralized operations, one for the control subframe and one for the data subframe. The control subframe is used for exchanging centralized scheduling messages. Assuming all that routers in the network are time synchronized, a router calculates its control schedule by extracting a breadth-first topology-based tree included in a mesh centralized schedule configuration message (MSH-CSCF) transmitted by a wireless mesh network (WMN) gateway. Given the control schedule, each router transmits its link price information using a mesh centralized schedule message (MSH-CSCH) request to the gateway. On receiving all MSH-CSCH messages, the gateway propagates MSH-CSCH grants, which include channel and power allocation information for the data subframe schedule augmented.

\(^4\)This is the maximum ratio by which the results of an approximation algorithm may differ from the optimal solution.
VI. ADMISSION CONTROL

In this section we extend our primal-dual framework to support admission control to handle dynamic settings where flows enter and exit the 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 (8). However, in the first constraint of (8), if sum of QoS requirements of the various sources on a link exceeds the link capacity, the link cost, represented by $\lambda$, 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 subset 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 $\gamma$ per slot during $\chi$ consecutive slots, a schedule is considered to be infeasible. In order to handle this infeasible scenario, we first solve (8) 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 $\text{MRG}$ (for maximum gap). After removing the above source from $\overline{G}$, we solve the relaxed form of (8) again with the sources in $G \cup \overline{G}$. 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: $\text{MR}$ and $\text{MRG}$. Policy $\text{MR}$ (for maximum rate) selects the source $k \in \overline{G}$ for which, the QoS requirement is the maximum, i.e., $k = \arg\max_{s \in \overline{G}} r_{s}^{req}$. If there are multiple sources with the same maximum rate, one of these sources is randomly selected and dropped. With $\text{MRG}$, $\text{MR}$ is applied first and subsequently, in the case of a tie $\text{MRG}$ is applied.

Algorithm 3 Adaptive Resource Allocation with $\text{MRG}$

1: Initialization: $G \leftarrow \emptyset$, $\overline{G} \leftarrow \emptyset$
2: Perform Algorithm 1 on the utility maximization problem (8) 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 $k = \arg\max_{s \in \overline{G}} (r_{s}^{req} - r_{s})$;
6: Solve (8) with $r_{s}^{req} = 0, \forall s \in G \cup \overline{G}$;
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 (8), 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 $E \cup N$, 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 (8) for the sources in $E \cup N$;
4: if (8) is infeasible then
5: Run Algorithm 3 and Get $G$;
6: while $E \cap G \neq E$ 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 $E \cap G = E$ 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; the wireless routers can serve as both access points (APs) for client nodes and relaying nodes for forwarding data received from neighboring nodes. The size of network for evaluation is 800m × 400m. Eight different topologies are considered; half of them (topology numbers 1 to 4) are generated by design to balance the network load and the other half (topology numbers 5 to 8) are randomly generated. Each topology consists of 30 wireless routers and 1 gateway. Two topologies (topology 1 and 5) under consideration are shown in Figure 1. In order to focus on the effects of power control and link scheduling, the routing issues are excluded by fixing the route to the gateway from each router as shown in the topologies during evaluations. For traffic generation, six different routers are randomly selected as traffic sources in each topology. We adopt the log-distance path loss model with the path loss exponent set to 3 [34]. Links use one of 3 non-overlapping orthogonal channels (as
with 802.11g [35]). The maximum transmit power for each link is set to 1000 mW, and the maximum sending rate for each of the sources is set to 54 Mbps.

B. Results

The evaluations of the proposed resource allocation framework are presented in subsection VII-B1. The admission control strategy is evaluated in subsection VII-B2.

1) Performance Comparison of Resource Allocation Schemes: In this subsection, we evaluate the performance of the proposed channel allocation and power control scheme. To this end, we consider a scenario where all the traffic sources in each topology have no QoS requirements. The performance of the proposed scheme is compared with that of (a) a resource allocation strategy with a random channel assignment and a fixed power with the maximum transmit power (RC-FP), (b) a random channel assignment strategy and the proposed power control (RC-PC), (c) the congestion-aware channel assignment with a fixed power (CA-FP) [36], which assigns channels such that the product of link cost and link capacity (transmit power is fixed to the maximum one) is maximized (the scheduling problem (11) with given $P$). Figure 2 shows the average goodput of traffic sources achieved with the different schemes with the considered topologies. In summary, the proposed channel and power allocation method achieves a higher goodput than RC-FP, RC-PC, and CA-FP by 125%, 20%, and 74%, respectively. The goodput gain of the proposed scheme comes from the combination of power control and channel assignment. In particular, we make the observation from Figure 2 that the contribution of power control and channel assignment to the goodput gain are varied depending on topology type. The contribution of the proposed power control is evaluated through the ratio of power control gain (achieved by comparing the goodput of RC-FP and that of RC-PC) to the total gain (achieved by comparing the goodput of RC-FP and that of the proposed). Naturally, the contribution of proposed channel assignment is obtained by ‘1 - the contribution from the power control’. In the planned topologies, the contribution from power control (66% on average) is slightly larger than that from channel assignment (34% on average); the power control contributes 75% to the goodput gain, thus, dominating over the channel assignment gain, with the random topologies. Compared with CA-FP, the proposed scheme is observed to achieve the gains with every topology: 63% and 85% gain on average, with the planned and random topologies, respectively. This indicates that the gains from the proposed power control overwhelms the gains from the channel assignment of CA-FP; in particular, the gains become more significant in ad hoc deployments, i.e., with the random topologies.

Figure 3 demonstrates the trends in the average source rates, when the cross decomposition approach for topology 1 is used with various resource allocation schemes. The speed of convergence is closed to that with Frank-Wolfe methods [37], which require $O(1/\epsilon)$ iterations to reach an accuracy of $\epsilon$. We utilize the termination condition by that in an iteration rate increment should be less than 0.01; the iterations for all schemes are terminated by around 430 slots as shown in the figure. Similar trends (less than or equal to 430 slots for termination) are also observed with the other topologies.

The proposed power control assumes that the power levels can be assigned from a continuous sample space, which is unlikely with commercial off-the-shelf hardware. With such system, power levels are typically assigned from a discrete set. To investigate the impact of using discrete power levels on throughput performance, three levels of adjustment granularity are introduced: 100 mW, 1 mW, and 10 uW. Figure 4 shows the average performance with this setting, when RC-PC scheme is applied over all the topologies. When a 100 mW unit is used (therefore, total of 10 available power levels) for adjusting transmit power, the throughput of RC-PC amounts to 85% of that with continuous power levels. Throughputs of 92% and 99% are as compared to the continuous case attained with adjusting knobs of 10 mW (1000 available power levels) and 10 uW (100000 available power levels), respectively. As shown in the figure however, the marginal gains via increasing the number of power levels diminishes linearly from 40% (compared over RC-FP) to 10% as more towards finer granularity.

2) Evaluating our admission control strategy: To evaluate the proposed admission control strategy, we consider a scenario where service requests from the traffic sources are generated dynamically for topology 1. The requests from N23 and N25 arrive first. Later, the requests for N14 and N30 arrive together followed by those of N13 and N16. An admission control policy based on the proposed resource allocation scheme and the dropping rule $MR$ is compared with policies based on four 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) RC-FP and MR, (c) CA-FP and MR and (d) the proposed resource allocation and OPT. 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 Satisfied or Not Satisfied 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 except N13 and N23, which are classified as S). 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, which is also applicable to the result of CA-FP. The admission control with the proposed resource allocation framework and the dropping rule MR, supports five sources with their QoS requirements satisfied (all except N30). It rejects N30 since accommodating both N14 and N30 causes a violation of QoS for N25. 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 resource allocation framework for wireless mesh networks. The framework maximizes the aggregate utility of flows taking into account constraints that arise due to self-interference (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

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.