

A Class of Cross-Layer Optimization Algorithms for Performance and Complexity Trade-offs in Wireless Networks

Xiaoying Zheng*, Feng Chen[†], Ye Xia* and Yuguang Fang[†]

* Computer and Information Science and Engineering Department, University of Florida

Email: {xiazheng, yx1}@cise.ufl.edu

[†] Electrical and Computer Engineering, University of Florida

Email: {chenf, fang}@ece.ufl.edu

Abstract—In this paper, we solve the problem of a joint optimal design of congestion control and wireless MAC-layer scheduling using a column generation approach with imperfect scheduling. We point out that the general subgradient algorithm has difficulty in recovering the time-share variables and experiences slower convergence. We first propose a two-timescale algorithm that can recover the optimal time-share values. Most existing algorithms have a component, called global scheduling, which is usually NP-hard. We apply imperfect scheduling and prove that if the imperfect scheduling achieves an approximation ratio ρ , then our algorithm converges to a sub-optimum of the overall problem with the same approximation ratio. By combining the idea of column generation and the two-timescale algorithm, we derive a family of algorithms that allow us to reduce the number of times the global scheduling is needed.

Index Terms—Cross-Layer design, Optimization, Column Generation, MAC-layer scheduling, Congestion control

I. INTRODUCTION

The joint congestion-control and scheduling problem in multi-hop wireless networks has become a very active research area in the last few years [3], [4], [6], [7], [10], [11], [13]–[15], [19], [21]–[23]. The problem can be formulated as the maximization of the aggregate source utility over the network capacity constraints. Unlike the similar problem in the wired network, the essential nature of the problem in the wireless setting is that the network capacity itself is a decision variable. Due to wireless interference, not all transmission configurations are allowed at each time instance. For instance, in the well-known model of the multiple access scheme for the 802.11 network, an allowed configuration is a subset of the links whose transmissions do not interfere with each other. Scheduling at the MAC layer is to decide which of the allowed configurations should be used and how they should be used (e.g., time shared). The result of scheduling implicitly determines the network capacity.

The standard subgradient algorithm is a good candidate in solving such a problem. By the subgradient technique, the rate control and the wireless resource allocation are decoupled: The sources adapt their source rates according to the path congestion costs, whereas the MAC-layer scheduling adjusts the time share of different allowed transmission configurations, thus varying the link capacities according to the link costs so as to support the flow rates. However, the standard subgradient technique has its own limitation, which will be discussed.

We propose a *two-timescale, column-generation* approach with *imperfect* global scheduling to solve the above problem.

Compared with the subgradient technique and others, our approach offers the following features.

- Our approach solves the difficult issue of recovering the time share using a two-timescale method. The issue arises when the Lagrangian function of the maximization problem is not strictly concave in all its primal variables. Specifically, in the subgradient algorithm, the dual problem converges to an optimal solution. However, the primal variables corresponding to the time share proportions oscillate. Our two-timescale algorithm ensures that both the primal and dual solutions converge to the optimum.
- The column generation method introduces one extreme point at a time and gradually expands the feasible set, where an extreme point corresponds to one allowed transmission configuration (also known as a *schedule*). Typically, introducing such an extreme point involves solving an NP-hard combinatorial optimization problem [14], [15]. In our approach, we allow the introduction of a sub-optimal extreme point, which is often far easier to obtain. This opens the door for the application of many heuristic algorithms in solving the hard combinatorial problem. Importantly, we show that, if the sub-optimal extreme point is a ρ -approximation solution to the combinatorial optimization problem, then the overall utility-maximization problem also achieves ρ approximation.
- By combining column generation and the aforementioned two-timescale algorithm, we in fact have a whole family of algorithms. On one side of the spectrum, we have a pure column generation algorithm; on the other side, we have a pure two-timescale algorithm. In between, we have a mixed algorithm that introduces new extreme points at varying degree of frequency, thus balancing various aspects of the algorithm, e.g., performance and complexity.

We subsequently call the sub-problem of finding a new extreme point in the column generation algorithm, *global scheduling*, since it involves finding an allowed transmission schedule from all possible ones. This sub-problem is a combinatorial optimization problem on an exponential number of possibilities. A *perfect schedule* refers to an optimal solution to the sub-problem; an *imperfect schedule* refers to a sub-optimal solution to the sub-problem. Other algorithms usually also contain this sub-problem. How to avoid global scheduling as much as possible and how to solve it fast when needed are

two key issues. This paper makes contributions in both.

We now give a brief summary of prior work on the joint design of congestion control, routing and scheduling in wireless networks. A survey of resource allocation and cross-layer control in wireless networks can be found in [10]. Varbrand et al. [3] propose the column generation method to solve the resource allocation problem in wireless ad hoc networks. Johansson and Xiao [11] extend the use of the column generation method to solve the same problem under more comprehensive wireless interference models. But both [3] and [11] give centralized solutions, where the restricted master problems are solved by some linear/nonlinear solvers (We are interested in distributed algorithms.); and they only consider the case where perfect scheduling is used. [13] also gives a centralized column generation solution. Bohacek and Wang [4] implicitly apply the column generation method and their approach is centralized. In [14], [15], the authors propose a way to solve this problem by a distributed subgradient algorithm with imperfect scheduling. Their approach and conclusion are different from ours and we will detail the differences in Section IV. In [6], [7], the authors formulate similar problems as ours and develop subgradient algorithms. [22] discusses the framework of cross-layer optimization in wireless networks. Another related paper is [19]. The two-timescale adaptive method is proposed in [9], and used in [5], [17] for the problem of multi-path routing. To our best knowledge, no prior work has combined the three elements together, two timescales, column generation and imperfect scheduling.

The paper is organized as follows. The network model and problem formulation are given in Section II. The two-timescale algorithm and its convergence proof are given in Section III. In Section IV, we present the column generation approach, combine it with the two-timescale method, and study the impact of imperfect scheduling. We show the performance with imperfect scheduling is bounded. In Section V, we give the experimental examples. The conclusion is drawn in Section VI.

II. PROBLEM DESCRIPTION

Let the network be represented by a directed graph $G = (V, E)$, where V is the set of nodes and E is the set of links. The presence of link $e \in E$ means that the network is able to send data from the start node of e to the end node of e . Unlike in a wired system where the capacity of a link is a fixed constant, in a wireless system, due to the shared nature of the wireless medium, the rate c_e of a link e depends not only on its own modulation/coding scheme, power assignment P_e and the ambient noise, but also on the interference from other transmitting links, which in turn depends on their power assignments. Let $P = (P_e)$ denote a vector of a global power assignment and let $c = (c_e)$ denote the vector of the corresponding link rates, where $0 \leq P_e \leq P_{e,\max}$ for all $e \in E$. We assume the data rates c are completely determined by the global power assignment P , which means there exists a rate-power function u such that $c = u(P)$ [14]. The rate-power function is determined by the interference model.

We describe the following model as an example. Let $G_{ee'}$ denote the effective power gain between the sender of link e and the receiver of link e' . Let σ_e denote the thermal noise power at e 's receiver. The signal to interference and noise ratio

(SINR) of link e is

$$\omega_e(P) = \frac{G_{ee}P_e}{\sigma_e + \sum_{e' \in E, e' \neq e} G_{e'e}P_{e'}}. \quad (1)$$

According to Shannon's capacity theorem, the maximum data rate of link e is $c_e = W \log(1 + \omega_e(P))$, where W is the system bandwidth. In practice, the link rate is usually lower than the Shannon capacity. Typical wireless systems allow a finite set of link rates, say c_e^1, \dots, c_e^k , which are associated with a set of thresholds for the SINR, $\beta_e^{(c_e^1)}, \dots, \beta_e^{(c_e^k)}$. This is usually due to the finite number of modulation/coding schemes built into the wireless transceiver. A link e can use the transmission rate c_e^j , if $\omega_e \geq \beta_e^{(c_e^j)}$.

To summarize, at any time instance, the number of possible rate vectors is finite. Each of these allowed rate vectors will be called a *schedule*. Let Q denote the total number of schedules. Let $c^{(i)} = (c_e^{(i)})$ denote the i^{th} schedule (rate vector) in the set of feasible schedules, for $i = 1, \dots, Q$, where the order is arbitrary. Though Q is finite, it might be exponential in the number of links. By time sharing of these feasible schedules, the achievable time-average link-rate region is the convex hull of $c^{(i)}$, $i = 1, \dots, Q$. Denote this convex hull by \mathcal{C} . Thus, \mathcal{C} is a convex polytope. With slight abuse of terminology, we call $c^{(i)}$, $i = 1, \dots, Q$, the extreme points of \mathcal{C} . In fact, some of them may not be extreme points of the polytope. For any $c \in \mathcal{C}$, it could be represented by the following convex combination of the extreme points of \mathcal{C} ,

$$c = \sum_{i=1}^Q \alpha_i c^{(i)}, \quad \sum_{i=1}^Q \alpha_i = 1, \alpha_i \geq 0, i = 1, \dots, Q, \quad (2)$$

where α_i denotes the time-share fraction of the schedule that uses the schedule $c^{(i)}$. One can find more discussion on wireless interference models in [11].

A. Network Model

Suppose there is a set of source-destination pairs. Let S be the set of sources and x_s be the source rate of source $s \in S$. Assume the flow between each source-destination pair is routed along the fixed single path, and denote this path by p_s for each source s . Define $U_s(x_s)$, $x_s \geq 0$, the utility function for each source $s \in S$. Assumptions on the utility functions are, for every $s \in S$,

- A1: U_s is increasing, strictly concave and twice continuously differentiable for all $x_s \geq 0$.
- A2: $U_s(x_s) \geq 0$ for all $x_s \geq 0$.
- A3: $U_s'(x_s)$ is well-defined and bounded at $x_s = 0$.

The optimal resource allocation and scheduling problem is formulated as

$$\max \sum_{s \in S} U_s(x_s) \quad (3)$$

$$\begin{aligned} \text{s.t.} \quad & \sum_{s: e \in p_s} x_s \leq c_e, \quad \forall e \in E \\ & c \in \mathcal{C} \\ & x_s \geq 0, \quad \forall s \in S. \end{aligned} \quad (4)$$

By replacing (4) with the equivalent expression in (2), we rewrite the above problem as follows.

$$\begin{aligned} \text{MP:} \quad & \max \sum_{s \in S} U_s(x_s) & (5) \\ \text{s.t.} \quad & \sum_{s: e \in p_s} x_s \leq \sum_{i=1}^Q \alpha_i c_e^{(i)}, \quad \forall e \in E & (6) \\ & \sum_{i=1}^Q \alpha_i = 1 \\ & x_s \geq 0, \quad \forall s \in S \\ & \alpha_i \geq 0, \quad \forall i = 1, \dots, Q. \end{aligned}$$

Note that $c_e^{(i)}$ is a constant instead of a decision variable, and the only decision variables are x and α . We call the above problem the master problem (MP).

B. Dual of Master Problem

Let λ_e be the Lagrangian multiplier associated with the constraint (6). The Lagrangian function of MP is

$$\begin{aligned} L(x, \alpha, \lambda) &= \sum_{s \in S} U_s(x_s) + \sum_{e \in E} \lambda_e \left(\sum_{i=1}^Q \alpha_i c_e^{(i)} - \sum_{s: e \in p_s} x_s \right). \\ &= \sum_{s \in S} (U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e) + \sum_{i=1}^Q \alpha_i \left(\sum_{e \in E} \lambda_e c_e^{(i)} \right). \end{aligned}$$

The dual function is

$$\begin{aligned} \theta(\lambda) &= \max_{\alpha} L(x, \alpha, \lambda) & (7) \\ \text{s.t.} \quad & \sum_{i=1}^Q \alpha_i = 1 \\ & x_s \geq 0, \quad \forall s \in S \\ & \alpha_i \geq 0, \quad \forall i = 1, \dots, Q. \end{aligned}$$

Now the dual problem of MP is

$$\begin{aligned} \text{Dual-MP:} \quad & \min \theta(\lambda) & (8) \\ \text{s.t.} \quad & \lambda \geq 0. \end{aligned}$$

III. A TWO-TIMESCALE ALGORITHM

In this section, we will illustrate how the MP can be solved by a two-timescale algorithm. In Section IV, we will combine this two-timescale algorithm with a column generation algorithm and derive a family of algorithms.

We first consider the rate control problem with *fixed* time fraction vector α .

$$\begin{aligned} \text{MP-A:} \quad & \Phi(\alpha) := \max_x \sum_{s \in S} U_s(x_s) & (9) \\ \text{s.t.} \quad & \sum_{s: e \in p_s} x_s \leq \sum_{i=1}^Q \alpha_i c_e^{(i)}, \quad \forall e \in E & (10) \\ & x_s \geq 0, \quad \forall s \in S. \end{aligned}$$

The above problem MP-A has a strictly concave objective function and has a unique solution with respect to the only variable, vector x . $\Phi(\alpha)$ denotes the optimal objective function value of MP-A under each α . The original problem MP can be re-written as

$$\begin{aligned} \text{MP-B:} \quad & \max_{\alpha} \Phi(\alpha) & (11) \\ \text{s.t.} \quad & \sum_{i=1}^Q \alpha_i = 1 \\ & \alpha_i \geq 0, \quad \forall i = 1, \dots, Q. \end{aligned}$$

A. Solve Problem MP-A with the Subgradient Method

The problem MP-A could be solved by the subgradient algorithm¹. Let λ_e be the Lagrange multiplier associated with the constraint (10). The Lagrangian function of MP-A is

$$\begin{aligned} L_A(\alpha, x, \lambda) &= \sum_{s \in S} U_s(x_s) + \sum_{e \in E} \lambda_e \left(\sum_{i=1}^Q \alpha_i c_e^{(i)} - \sum_{s: e \in p_s} x_s \right) \\ &= \sum_{s \in S} (U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e) + \sum_{i=1}^Q \alpha_i \left(\sum_{e \in E} \lambda_e c_e^{(i)} \right). \end{aligned}$$

The dual function is

$$\theta_A(\alpha, \lambda) = \sum_{s \in S} \max_{x_s \geq 0} \{U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e\} + \sum_{i=1}^Q \alpha_i \left(\sum_{e \in E} \lambda_e c_e^{(i)} \right)$$

Since MP-A is maximizing a strictly concave function with linear constraints, the strong duality holds for MP-A [2]. Since there is no duality gap at the optimum of MP-A under a fixed α , we can re-write $\Phi(\alpha)$ as the optimal objective function value of the dual problem of MP-A,

$$\text{Dual-MP-A:} \quad \Phi(\alpha) = \min_{\lambda \geq 0} \theta_A(\alpha, \lambda). \quad (12)$$

The dual problem (12) can be solved by the subgradient method as in Algorithm 1 ([1], [2]), where $\delta(t)$ is a positive scalar stepsize, and $[\cdot]_+$ denote the projection onto the non-negative domain.

Algorithm 1 Fast Timescale: Subgradient Algorithm for Solving MP-A

$$\lambda_e(t+1) = [\lambda_e(t) - \delta(t) \left(\sum_{i=1}^Q \alpha_i c_e^{(i)} - \sum_{s: e \in p_s} x_s(t) \right)]_+, \quad \forall e \in E \quad (13)$$

$$x_s(t+1) = [(U'_s)^{-1} \left(\sum_{e \in p_s} \lambda_e(t+1) \right)]_+, \quad \forall s \in S \quad (14)$$

Let $(x^*(\alpha), \lambda^*(\alpha))$ denote the optimal primal-dual solutions of MP-A under a fixed α .

Theorem 1: (Convergence of the Subgradient Algorithm for MP-A): With the diminishing step size rule, i.e., $\lim_{t \rightarrow \infty} \delta(t) = 0$ and $\sum_{t=1}^{\infty} \delta(t) = \infty$, let $\{\lambda(t)\}$ be a sequence generated by Algorithm (13)-(14). Then $\theta_A(\alpha, \lambda(t)) \rightarrow \theta_A(\alpha, \lambda^*(\alpha))$ and there exists a subsequence T such that $\{\lambda(t)\}_T \rightarrow \lambda^*(\alpha)$ [1].

Theorem 2: (Convergence of the Primal Sequence for MP-A): Assume A1. In Algorithm (13)-(14), if $\lambda(t) \rightarrow \lambda^*(\alpha)$, then $x(t) \rightarrow x^*(\alpha)$.

Proof: The proof is standard and is omitted. ■

B. Update Time Fraction on a Slower Timescale

The above rate control algorithm (13)-(14) works under the assumption that the time fraction vector α remains constant. Now we discuss how to adjust $\alpha_i, i = 1, \dots, Q$, to solve the

¹Since the time-share variable α is a constant vector, there is no difficulty with the subgradient algorithm here.

problem MP-B. We assume the update of α is much slower so that the minimization of $\theta_A(\alpha, \lambda)$ over λ can be regarded as being instantaneous. Here, we follow the approaches in [5], [9], [17].

Let k index the time slots (called stages) of the slow timescale. At stage k , given the time fraction vector $\alpha(k)$, suppose $\lambda(k) \in \operatorname{argmin}_{\lambda \geq 0} \theta_A(\alpha(k), \lambda)$ is an optimal dual solution to MP-A. Let's call $\lambda_e(k)$ the price or cost of link e . Therefore, $\lambda_e(k)c_e^{(i)}$ is the cost of link e under the i^{th} schedule (i.e., the i^{th} extreme point of \mathcal{C}); and $\sum_{e \in E} \lambda_e(k)c_e^{(i)}$ is the cost of the network under the i^{th} schedule, which will be called the *cost of the schedule*. Let $i(k)$ be the index of a schedule achieving the maximum schedule cost under the link costs $\lambda(k)$, i.e.,

$$i(k) = \operatorname{argmax}_{i=1}^Q \left\{ \sum_{e \in E} \lambda_e(k)c_e^{(i)} \right\}. \quad (15)$$

If there is a tie, an arbitrary maximizing index is chosen. (15) may be called a scheduling problem [14], since it aims at finding a schedule. Because (15) is an optimization problem over all allowed schedules, $1, \dots, Q$, we call (15) a *global scheduling problem*, and the achieved maximum cost the *global maximum cost of the schedule*. We denote this global maximum cost under a fixed λ by

$$\gamma(\lambda) = \max_{1 \leq i \leq Q} \left\{ \sum_{e \in E} \lambda_e c_e^{(i)} \right\}. \quad (16)$$

The time fraction update is shown in Algorithm 2, which is similar to the one in [5], [9], [17].

Algorithm 2 Slow Timescale: Time Fraction Update for Solving MP-B

$$\alpha_i(k+1) = \alpha_i(k) + \Delta_i(k) \quad (17)$$

$$\Delta_i(k) = \begin{cases} -\min\{\xi(k)(\sum_{e \in E} \lambda_e(k)c_e^{(i(k))} - \sum_{e \in E} \lambda_e(k)c_e^{(i)}), \alpha_i(k)\}, & \text{if } i \neq i(k) \\ -\sum_{i \neq i(k)} \Delta_i(k), & \text{if } i = i(k). \end{cases} \quad (18)$$

Here, $\xi(k)$ is a positive step size. Note that $\Delta_i(k) \leq 0$ for $i \neq i(k)$ and $\Delta_i(k) \geq 0$ for $i = i(k)$. Hence, the algorithm increases the time fraction of the most costly schedule while decreases the time fractions of other active schedules, i.e., those schedules with positive time fractions $\alpha_i(k)$. Furthermore, if $\sum_{i=1}^Q \alpha_i(k) = 1$, then $\sum_{i=1}^Q \alpha_i(k+1) = 1$. Hence, $\alpha(k)$ will always be valid time fraction vectors for all k if $\sum_{i=1}^Q \alpha_i(0) = 1$.

It can be verified that

$$\sum_{i=1}^Q \Delta_i(k) = 0 \quad (19)$$

$$\sum_{i=1}^Q \Delta_i(k) \sum_{e \in E} \lambda_e(k)c_e^{(i)} \geq 0. \quad (20)$$

Equality in (20) occurs if and only if $\Delta_i(k) = 0$ for all i , which is equivalent to

$$\alpha_i(k) \left(\sum_{e \in E} \lambda_e(k)c_e^{(i(k))} - \sum_{e \in E} \lambda_e(k)c_e^{(i)} \right) = 0, \forall i. \quad (21)$$

Conditions in (19) - (21), and those described in the preceding paragraph guarantee that the time fraction update algorithm converges to the correct optimal value. As in [5], we consider a continuous-time, differentiable version of the algorithm (17)-(18). First, define the set

$$\Lambda(\alpha) = \{ \lambda \geq 0 : \theta_A(\alpha, \lambda) = \min_{\lambda' \geq 0} \theta_A(\alpha, \lambda') \}. \quad (22)$$

The differentiable version of the algorithm (17)-(18) satisfies the following conditions, for any $\lambda(\alpha) \in \Lambda(\alpha)$.

$$\sum_{i=1}^Q \dot{\alpha}_i = 0, \quad (23)$$

$$\sum_{i=1}^Q \dot{\alpha}_i \sum_{e \in E} \lambda_e(\alpha)c_e^{(i)} \geq 0, \quad (24)$$

$$\sum_{i=1}^Q \dot{\alpha}_i \sum_{e \in E} \lambda_e(\alpha)c_e^{(i)} = 0 \text{ if and only if } \dot{\alpha}_i = 0, \forall i. \quad (25)$$

The condition in (25) is equivalent to

$$\alpha_i \left(\sum_{e \in E} \lambda_e(\alpha)c_e^{(i(k))} - \sum_{e \in E} \lambda_e(\alpha)c_e^{(i)} \right) = 0, \forall i. \quad (26)$$

Theorem 3: The time fraction update algorithm (17)-(18) converges to an optimal solution of the problem MP-B.

Proof:

$$\begin{aligned} \Phi(\alpha) &= \min_{\lambda \geq 0} \theta_A(\alpha, \lambda) \\ &= \min_{\lambda \geq 0} \left\{ \sum_{s \in S} (U_s(x_s(\lambda)) - x_s(\lambda) \sum_{e \in p_s} \lambda_e) \right. \\ &\quad \left. + \sum_{i=1}^Q \alpha_i \left(\sum_{e \in E} \lambda_e c_e^{(i)} \right) \right\}. \end{aligned}$$

Note that $\theta_A(\alpha, \lambda)$ is a continuous function. For each $\alpha \geq 0$, $\theta_A(\alpha, \cdot)$ is bounded from below (say, by $\sum_{s \in S} U_s(0)$). Hence, $\Phi(\alpha)$ is well defined on $\alpha \geq 0$. Furthermore, $\theta_A(\cdot, \lambda)$ is concave (actually linear), for each fixed λ . Hence, $\Phi(\alpha)$ is a concave function in α , which means it has directional derivatives. We will apply Danskin's theorem ([2], page 717). The theorem requires λ to be in a compact set. In other words, it requires that there exists a compact set Λ independent of α such that $\Phi(\alpha) = \min_{\lambda \geq 0} \theta_A(\alpha, \lambda) = \min_{\lambda \in \Lambda} \theta_A(\alpha, \lambda)$. We will next construct one such compact set. Since $U_s(\cdot)$ is concave, we have $U'_s(0) \geq U'_s(x_s)$ for all $x_s \geq 0$. Under assumption A3, take some $K \geq \max_{s \in S} U'_s(0) > 0$. Let $\Lambda = \{ \lambda : 0 \leq \lambda_e \leq K, \forall e \in E \}$. For any $\lambda \notin \Lambda$, there exists a non-empty subset $E_1 \subseteq E$, where $\lambda_e > K$ for any $e \in E_1$ and $\lambda_e \leq K$ for any $e \notin E_1$. Let denote a subset of sources by $S_1 \subseteq S$, where for any source $s \in S_1$, its routing path p_s contains some links in the set E_1 . We construct a vector $\lambda' \in \Lambda$ where $\lambda'_e = K$ for any $e \in E_1$, and $\lambda'_e = \lambda_e$ for any link $e \in E \setminus E_1$. For any $s \in S$, if its accumulated path cost is no less than K , then the maximum of $U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e$

in the definition of $\theta_A(\alpha, \lambda)$ is achieved at $x_s = 0$, which means for any $s \in S_1$,

$$U_s(0) = \max_{x_s \geq 0} \{U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e\} = \max_{x_s \geq 0} \{U_s(x_s) - x_s \sum_{e \in p_s} \lambda'_e\}.$$

Then

$$\begin{aligned} \theta_A(\alpha, \lambda) &= \sum_{i=1}^Q \alpha_i \left(\sum_{e \in E} \lambda_e c_e^{(i)} \right) + \sum_{s \in S_1} \max_{x_s \geq 0} \{U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e\} \\ &+ \sum_{s \in S \setminus S_1} \max_{x_s \geq 0} \{U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e\} \\ &\geq \sum_{i=1}^Q \alpha_i \left(\sum_{e \in E} \lambda'_e c_e^{(i)} \right) + \sum_{s \in S_1} \max_{x_s \geq 0} \{U_s(x_s) - x_s \sum_{e \in p_s} \lambda'_e\} \\ &+ \sum_{s \in S \setminus S_1} \max_{x_s \geq 0} \{U_s(x_s) - x_s \sum_{e \in p_s} \lambda'_e\} \\ &= \theta_A(\alpha, \lambda'). \end{aligned} \quad (27)$$

Thus for any α , the minimum of $\theta_A(\alpha, \lambda)$ over $\lambda \geq 0$ occurs in Λ .

The conditions required by Danskin's theorem are met. Let $\Phi'(\alpha; \dot{\alpha})$ denote the directional derivative of $\Phi(\alpha)$ in the direction of $\dot{\alpha}$. Let $\theta'_A(\alpha, \lambda; \dot{\alpha})$ be the directional derivative of $\theta_A(\cdot, \lambda)$ at α in the direction of $\dot{\alpha}$. Then, by Danskin's theorem,

$$\begin{aligned} \Phi'(\alpha; \dot{\alpha}) &= \min_{\lambda \in \Lambda(\alpha)} \theta'_A(\alpha, \lambda; \dot{\alpha}) \\ &= \min_{\lambda \in \Lambda(\alpha)} \sum_{i=1}^Q \left(\sum_{e \in E} \lambda_e c_e^{(i)} \right) \dot{\alpha}_i \\ &= \sum_{i=1}^Q \left(\sum_{e \in E} \bar{\lambda}_e(\alpha) c_e^{(i)} \right) \dot{\alpha}_i. \end{aligned} \quad (28)$$

where $\bar{\lambda} \in \Lambda(\alpha)$ achieves the minimum.

Then, by (24),

$$\Phi'(\alpha; \dot{\alpha}) \geq 0. \quad (29)$$

By the Lasalle invariance principle [12], $\alpha(t)$ converges to an invariant set inside $\{\alpha : \Phi'(\alpha; \dot{\alpha}) = 0\}$. Take a trajectory in this invariant set, which satisfies $\Phi'(\alpha; \dot{\alpha}) \equiv 0$. By (28), $\sum_{i=1}^Q \left(\sum_{e \in E} \bar{\lambda}_e(\alpha) c_e^{(i)} \right) \dot{\alpha}_i \equiv 0$. Then, by (25), $\dot{\alpha}_i \equiv 0$ for all i . Hence, the invariant set has only one point, which will be denoted by α^* . Hence, $\alpha(t)$ converges to α^* .

Next, we will show that α^* solves problem MP-B. Let $x^*(\alpha^*)$ and $\lambda^*(\alpha^*)$ be the optimal solution of MP-A under α^* . MP-A maximizes a strictly concave function with linear constraints, and hence, the KKT conditions are both necessary and sufficient optimality conditions for MP-A [2]. Thus, at the optimum $(x^*(\alpha^*), \lambda^*(\alpha^*))$, we have that $x^*(\alpha^*)$ is primal feasible and $\lambda^*(\alpha^*)$ is dual feasible for MP-A, and

$$x^*(\alpha^*) = \operatorname{argmax}_{x \geq 0} \{U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e^*(\alpha^*)\} \quad (30)$$

$$\lambda_e^*(\alpha^*) \left(\sum_{i=1}^Q \alpha_i^* c_e^{(i)} - \sum_{s: e \in p_s} x_s^*(\alpha^*) \right) = 0, \forall e \in E. \quad (31)$$

At α^* , we have $\dot{\alpha} = 0$. Hence, according to (26), we have

$$\alpha_i^* > 0 \text{ only if } \lambda_e^*(\alpha^*) c_e^{(i)} = \max_{j=1}^Q \{\lambda_e^*(\alpha^*) c_e^{(j)}\}. \quad (32)$$

Also, by (23), if we initialize the update of α at some $\alpha(0)$ satisfying $\sum_{i=1}^Q \alpha_i(0) = 1$, we will have

$$\sum_{i=1}^Q \alpha_i^* = 1, \quad (33)$$

which implies that

$$\alpha^* = \operatorname{argmax}_{\alpha \geq 0: \sum_i \alpha_i = 1} \left\{ \sum_{i=1}^Q \alpha_i \left(\sum_{e \in E} \lambda_e^*(\alpha) c_e^{(i)} \right) \right\}. \quad (34)$$

Obviously, $x^*(\alpha^*)$, $\lambda^*(\alpha^*)$ and α^* are all non-negative. These non-negativity conditions, the fact that $x^*(\alpha^*)$ is primal feasible for MP-A, and the conditions in (30) - (31) and (33) - (34) are the optimality conditions of the MP. Hence, $(x^*(\alpha^*), \alpha^*, \lambda^*(\alpha^*))$ is an optimal primal-dual solution to the MP (also to MP-B). ■

C. Summary of the Two-Timescale Algorithm

To summarize, the two-timescale algorithm consists of

- a fast timescale distributed algorithm for rate control, which adapts the source rates and link prices according to (13)-(14),
- a slow timescale algorithm for updating the time fraction according to (17)-(18).

However, in most wireless interference models, problem (15) does not even have a centralized polynomial-time solution. This has been the main obstacle in developing practical rate control/scheduling algorithms. In next section, we will try to overcome this difficulty.

IV. COLUMN GENERATION METHOD WITH IMPERFECT GLOBAL SCHEDULING

The global scheduling problem (15) is usually an NP-hard combinatorial problem [11], [14], [15]. One fundamental reason is that the convex polytope, \mathcal{C} , usually has an exponential number of extreme points in terms of the number of links. The column generation method with imperfect global scheduling can be introduced to overcome this difficulty. The column generation part reduces the number of times when the global scheduling problem is invoked. Imperfect scheduling uses fast approximation or heuristic algorithms for speedup.

A. Column Generation Method

The main idea of column generation is to start with a subset of the extreme points of \mathcal{C} and bring in new extreme points only when needed. Consider a subset of \mathcal{C} formed by convex combination of q extreme points, i.e., $\mathcal{C}^{(q)} = \{c : c = \sum_{i=1}^q \alpha_i c^{(i)}, \sum_{i=1}^q \alpha_i = 1, \alpha_i \geq 0, \forall i = 1, \dots, q\}$. We can formulate the following restricted master problem (RMP) for $c \in \mathcal{C}^{(q)}$.

$$q^{th}\text{-RMP:} \quad \max \sum_{s \in S} U_s(x_s) \quad (35)$$

$$\begin{aligned} \text{s.t.} \quad & \sum_{s: e \in p_s} x_s \leq \sum_{i=1}^q \alpha_i c_e^{(i)}, \quad \forall e \in E \quad (36) \\ & \sum_{i=1}^q \alpha_i = 1 \\ & x_s \geq 0, \quad \forall s \in S \\ & \alpha_i \geq 0, \quad \forall i = 1, \dots, q. \end{aligned}$$

The value of q is usually small and the extreme points of $\mathcal{C}^{(q)}$ in the q^{th} -RMP are enumerable.

Let λ_e be the Lagrange multiplier associated with the constraint (36). The Lagrangian function of the q^{th} -RMP is

$$\begin{aligned} L^{(q)}(x, \alpha, \lambda) &= \sum_{s \in S} U_s(x_s) + \sum_{e \in E} \lambda_e \left(\sum_{i=1}^q \alpha_i c_e^{(i)} - \sum_{s: e \in p_s} x_s \right) \\ &= \sum_{s \in S} (U_s(x_s) - x_s \sum_{e \in p_s} \lambda_e) + \sum_{i=1}^q \alpha_i \left(\sum_{e \in E} \lambda_e c_e^{(i)} \right). \end{aligned}$$

The dual function is

$$\begin{aligned} \theta^{(q)}(\lambda) &= \max_{\alpha} L^{(q)}(x, \alpha, \lambda) \\ \text{s.t.} \quad & \sum_{i=1}^q \alpha_i = 1 \\ & x_s \geq 0, \quad \forall s \in S \\ & \alpha_i \geq 0, \quad \forall i = 1, \dots, q. \end{aligned} \quad (37)$$

The dual problem of the q^{th} -RMP can be formulated similarly as in (8).

The q^{th} -RMP is more restricted than the MP. Thus, any optimal solution to the q^{th} -RMP is feasible to the MP and serves as a lower bound of the optimal value of the MP. By gradually introducing more extreme points (columns) into $\mathcal{C}^{(q)}$ and expanding the subset $\mathcal{C}^{(q)}$, we will improve the lower bound of the MP [3], [11], [13].

B. Apply the Two-Timescale Algorithm to the RMP

The two-timescale algorithm can be used to solve the q^{th} -RMP. Here, we define the following problem under the link cost vector $\lambda(k)$.

$$i^{(q)}(k) = \operatorname{argmax}_{i=1}^q \left\{ \sum_{e \in E} \lambda_e(k) c_e^{(i)} \right\}. \quad (38)$$

The optimization is taken over the q currently known schedules (extreme-point link-rate vectors). The problem in (38) is called the *local scheduling problem*, and the achieved maximum cost is called the *local maximum cost of the schedule*. We denote this local maximum cost under $\lambda \geq 0$ by

$$\gamma^{(q)}(\lambda) = \max_{1 \leq i \leq q} \left\{ \sum_{e \in E} \lambda_e c_e^{(i)} \right\}. \quad (39)$$

If there is more than one link-rate vector achieving the local maximum cost, the tie is broken arbitrarily.

C. Bounding the Gap between the MP and the q^{th} -RMP

Now the question is how to check whether the optimum of the q^{th} -RMP is optimal for the MP, and if not, how to introduce a new column (schedule or extreme point). It turns out there is an easy way to do both.

Let $(x^*, \alpha^*, \lambda^*)$ denote one of the optimal primal-dual solutions of the MP, and $(\bar{x}^{(q)}, \bar{\alpha}^{(q)}, \bar{\lambda}^{(q)})$ denote one of the optimal primal-dual solutions of the q^{th} -RMP. Since the strong duality holds for both problems, we have

$$\sum_{s \in S} U_s(x_s^*) = \theta(\lambda^*), \quad \sum_{s \in S} U_s(\bar{x}_s^{(q)}) = \theta^{(q)}(\bar{\lambda}^{(q)}). \quad (40)$$

Since the q^{th} -RMP is more restricted than the MP, we have

$$\sum_{s \in S} U_s(x_s^*) \geq \sum_{s \in S} U_s(\bar{x}_s^{(q)}). \quad (41)$$

Combining (40) and (41), we get the following lower bound for the optimal objective value of the MP.

$$\sum_{s \in S} U_s(x_s^*) \geq \sum_{s \in S} U_s(\bar{x}_s^{(q)}) = \theta^{(q)}(\bar{\lambda}^{(q)}). \quad (42)$$

By the weak duality [2], for any λ feasible to the dual problem of the MP, $\theta(\lambda)$ is an upper bound for the optimal objective value of the MP. In particular, consider $\bar{\lambda}^{(q)}$, which is optimal to the dual of the q^{th} -RMP and feasible to the dual of the MP. $\theta(\bar{\lambda}^{(q)})$ is an upper bound of $\sum_{s \in S} U_s(x_s^*)$, i.e.,

$$\theta(\bar{\lambda}^{(q)}) \geq \sum_{s \in S} U_s(x_s^*). \quad (43)$$

By inspecting the dual functions (37) and (7) of the q^{th} -RMP and the MP, respectively, we note that $\bar{x}^{(q)}$ is the unique Lagrangian maximizer at $\bar{\lambda}^{(q)}$ for both (37) and (7). By the definitions of the dual functions,

$$\begin{aligned} & \theta(\bar{\lambda}^{(q)}) - \theta^{(q)}(\bar{\lambda}^{(q)}) \\ &= \max_{\alpha \geq 0, \sum_{i=1}^q \alpha_i = 1} \left\{ \sum_{i=1}^q \alpha_i \left(\sum_{e \in E} \bar{\lambda}_e^{(q)} c_e^{(i)} \right) \right\} \\ & \quad - \max_{\alpha \geq 0, \sum_{i=1}^q \alpha_i = 1} \left\{ \sum_{i=1}^q \alpha_i \left(\sum_{e \in E} \bar{\lambda}_e^{(q)} c_e^{(i)} \right) \right\} \\ &= \gamma(\bar{\lambda}^{(q)}) - \gamma^{(q)}(\bar{\lambda}^{(q)}). \end{aligned}$$

In the last equality, we have used (15) and (38). Hence, the gap between the upper and lower bounds for the optimal objective value of the MP is $\gamma(\bar{\lambda}^{(q)}) - \gamma^{(q)}(\bar{\lambda}^{(q)})$, which is exactly the difference between the global maximum cost and the local maximum cost of the schedule under $\bar{\lambda}^{(q)}$. Therefore, we conclude the following fact.

Lemma 4: Let $(\bar{x}^{(q)}, \bar{\alpha}^{(q)}, \bar{\lambda}^{(q)})$ denote one of the optimal primal-dual solutions of the q^{th} -RMP. $(\bar{x}^{(q)}, \bar{\alpha}^{(q)}, \bar{\lambda}^{(q)})$ is optimal to the MP if and only if $\gamma(\bar{\lambda}^{(q)}) = \gamma^{(q)}(\bar{\lambda}^{(q)})$.

D. Introduce One More Extreme Point (Column or Schedule)

If the gap between the upper and lower bound, $\gamma(\bar{\lambda}^{(q)}) - \gamma^{(q)}(\bar{\lambda}^{(q)})$, is not narrow enough, then \mathcal{C} is not sufficiently well characterized by $\mathcal{C}^{(q)}$ and a new extreme point should be added to the RMP. We state the rule of introducing a new column in the following.

Fact 5: Any schedule achieving a cost greater than the local maximum cost of the schedule could enter the subset $\mathcal{C}^{(q)}$ in the RMP. The schedule achieving the global maximum cost of the schedule is one possible candidate and is often preferred.

Lemma 4 says, at the current link cost $\bar{\lambda}^{(q)}$, if none of the schedules that achieve the global maximum cost of the schedule are in the subset $\mathcal{C}^{(q)}$, then the current optimal solution of the q^{th} -RMP is not optimal for the MP. In this case, there are reasons to prefer the introduction of the globally optimal schedule specified by (15) as the new extreme point to the restricted master problem. This strategy is a local greedy approach to improve the lower bound of the optimal value of the MP. In fact, it can be viewed as a conditional gradient method for optimizing the lower bound, when the lower bound is viewed as a function of c [11].

E. Column Generation by Imperfect Global Scheduling

The global scheduling problem (15) is usually NP-hard, which makes the step of column generation very difficult. However, according to Fact 5, we do not have to solve it precisely. Instead, we may solve it approximately, and this is referred to as *imperfect global scheduling* [14].²

Suppose we are able to solve (15) with an approximation ratio $\rho \geq 1$, i.e.,

$$\gamma(\lambda) \leq \rho \gamma_\rho(\lambda), \quad (44)$$

where $\gamma_\rho(\lambda)$ is the cost of the schedule given by the approximate solution. Note that both $\gamma(\lambda)$ and $\gamma_\rho(\lambda)$ are non-negative for all vectors $\lambda \geq 0$.

1) *A ρ -approximation Approach:* We develop a column generation method with imperfect global scheduling as follows.

Algorithm 3 Column Generation with Imperfect Global Scheduling

- Initialize: Start with a collection of q schedules
 - Step 1: Run the slow timescale update (17)-(18) (which will call the fast timescale algorithm) for several (a finite number) times on the q^{th} -RMP.
 - Step 2: Solve the global scheduling problem (15) with approximation ratio ρ under the current dual cost λ .
 - If the schedule corresponding to the *approximate solution* of (15) is already in the current collection of schedules, go to Step 1;
 - otherwise, introduce this schedule into the current collection of schedules, increase q by 1, and go to Step 1.
-

We make several comments regarding Algorithm 3.

- If the approximate schedule derived in step 2 has a lower schedule cost than that of an existing schedule already selected, we define the existing schedule with the highest cost as the solution to the approximation algorithm. Hence, the cost of the imperfect (approximate) schedule cannot be lower than any of the existing schedules.
- In the worst case, the column generation method may bring in all the extreme points. However, it often happens that, within a relatively small number of column-generation steps, the optimal solution to the MP is already in $\mathcal{C}^{(q)}$. Thus, the original problem may be solved without generating all the extreme points [11].
- Our focus here is on approximation algorithms because we will be able to show guaranteed performance bound on the MP problem later. Other types of imperfect scheduling can also be used, including many heuristics algorithms and random search algorithms. Examples of the latter include genetic algorithms and simulated annealing [20].
- Note that since the number of extreme points of $\mathcal{C}^{(q)}$ is usually small and enumerable, it is possible for the nodes in the network to store the current collection of schedules. In order to compute the cost of each known schedule in each slow timescale update, each link e

can independently compute its corresponding term for each known schedule based on the local link dual cost. Then, those components of the schedule cost can be collected by some controller elected by the nodes in the network. The controller can compute the cost of each known schedule, the locally most costly schedule, update the time fractions by (18), and broadcast the results. Other than that, the two-timescale algorithm (13)-(14) and (17)-(18) on the q^{th} -RMP is completely decentralized. Furthermore, if the global scheduling problem (15) can be solved approximately in a decentralized fashion, then Algorithm 3 is completely decentralized except the part of the controller. In Section V, we will introduce one interference model, under which (15) can be solved in a decentralized fashion approximately [14], [15].

- Algorithm 3 in fact describes a whole class of algorithms. To see this, consider the special case where $\rho = 1$, i.e., the case of perfect global scheduling. In one end of the spectrum, if the slow-timescale algorithm in step 1 runs only once on the RMP, the algorithm becomes a pure two-timescale algorithm as in Section III. In the other end of the spectrum, if the slow-timescale algorithm runs on the RMP until convergence, the algorithm becomes a pure column generation method with the two-timescale algorithm as a building block for solving the restricted problems between consecutive column generation steps. By choosing different numbers of times to run the slow-timescale algorithm in step 1, we have many algorithms, representing different performance, convergence speed and complex tradeoffs.

2) Convergence with Imperfect Global Scheduling:

Theorem 6: Under the condition that the fast timescale optimization in the two-timescale algorithm can be regarded as being instantaneous, there exists a q , $1 \leq q \leq Q$, such that Algorithm 3 converges to one optimal primal-dual solution of this particular q^{th} -RMP, i.e., $(\bar{x}^{(q)}, \bar{\alpha}^{(q)}, \bar{\lambda}^{(q)})$. Furthermore, after Algorithm 3 converges to $(\bar{x}^{(q)}, \bar{\alpha}^{(q)}, \bar{\lambda}^{(q)})$, $\gamma_\rho(\bar{\lambda}^{(q)}) = \gamma^{(q)}(\bar{\lambda}^{(q)})$.

Proof: Since the fast timescale algorithm is assumed to converge instantaneously, we only need to consider the slow timescale algorithm and the column generation steps. Since the number of extreme points of \mathcal{C} is finite, eventually Algorithm 3 will stop introducing new extreme points. Hence, there exists a q , $1 \leq q \leq Q$, such that, after Algorithm 3 stops introducing new extreme points, the number of extreme points that have been introduced is q . Let the convex hull formed by these q points be denoted by $\mathcal{C}^{(q)}$. After Algorithm 3 no longer introduces new extreme points, it behaves just like the two-timescale algorithm but on the restricted set $\mathcal{C}^{(q)}$. According to the theorems in Section III, the two-timescale algorithm converges. Thus, Algorithm 3 converges to $(\bar{x}^{(q)}, \bar{\alpha}^{(q)}, \bar{\lambda}^{(q)})$ on this particular q^{th} -RMP.

We next show that, after Algorithm 3 converges to $(\bar{x}^{(q)}, \bar{\alpha}^{(q)}, \bar{\lambda}^{(q)})$, we have $\gamma_\rho(\bar{\lambda}^{(q)}) = \gamma^{(q)}(\bar{\lambda}^{(q)})$. First, note that $\gamma_\rho(\bar{\lambda}^{(q)}) \geq \gamma^{(q)}(\bar{\lambda}^{(q)})$ by the comment after Algorithm 3. Next, it must be true that $\gamma_\rho(\bar{\lambda}^{(q)}) \leq \gamma^{(q)}(\bar{\lambda}^{(q)})$. Otherwise, the schedule whose cost is $\gamma_\rho(\bar{\lambda}^{(q)})$ must not have already been in $\mathcal{C}^{(q)}$ and will be selected to enter. This violates the assumption that the algorithm never selects more than q schedules. ■

3) *Performance Bound under Imperfect Scheduling:* Theorem 6 says that the column generation method with imperfect

²Note that the local scheduling problem (38) can be easily solved precisely since the number of extreme points of $\mathcal{C}^{(q)}$ is usually small, and hence, enumerable.

global scheduling converges to a sub-optimum of the MP. Next, we will prove that the performance of this sub-optimum is bounded.

Theorem 7 (Bound of Imperfect Global Scheduling):

Under assumption A2, if the column generation method with imperfect global scheduling converges to $(\bar{x}^{(q)}, \bar{\alpha}^{(q)}, \bar{\lambda}^{(q)})$ on the q^{th} -RMP, we have

$$\theta^{(q)}(\bar{\lambda}^{(q)}) \leq \sum_{s \in S} U_s(x_s^*) \leq \theta(\bar{\lambda}^{(q)}) \leq \rho \theta^{(q)}(\bar{\lambda}^{(q)}). \quad (45)$$

Proof: Since the q^{th} -RMP is more restricted than the MP, we have $\theta^{(q)}(\bar{\lambda}^{(q)}) \leq \sum_{s \in S} U_s(x_s^*)$. By the weak duality, we have $\sum_{s \in S} U_s(x_s^*) \leq \theta(\bar{\lambda}^{(q)})$.

By the definition of the dual function for the MP in (7), we have

$$\begin{aligned} & \theta(\bar{\lambda}^{(q)}) \\ &= \max_{x \geq 0} \left\{ \sum_{s \in S} (U_s(x_s) - x_s \sum_{e \in p_s} \bar{\lambda}_e^{(q)}) \right\} + \gamma(\bar{\lambda}^{(q)}) \\ &\leq \rho \max_{x \geq 0} \left\{ \sum_{s \in S} (U_s(x_s) - x_s \sum_{e \in p_s} \bar{\lambda}_e^{(q)}) \right\} + \rho \gamma(\bar{\lambda}^{(q)}) \\ &= \rho \max_{x \geq 0} \left\{ \sum_{s \in S} (U_s(x_s) - x_s \sum_{e \in p_s} \bar{\lambda}_e^{(q)}) \right\} + \rho \gamma^{(q)}(\bar{\lambda}^{(q)}) \\ &= \rho \theta^{(q)}(\bar{\lambda}^{(q)}). \end{aligned}$$

The first inequality holds because, under assumption A2, $\max_{x \geq 0} \left\{ \sum_{s \in S} (U_s(x_s) - x_s \sum_{e \in p_s} \bar{\lambda}_e^{(q)}) \right\} \geq 0$ for any λ (which can be checked by plugging in $x_s = 0$ for all s), $\rho \geq 1$, and (44) is assumed. The second equality holds because $\gamma_\rho(\bar{\lambda}^{(q)}) = \gamma^{(q)}(\bar{\lambda}^{(q)})$ by Theorem 6. ■

Since the strong duality holds on the q^{th} -RMP, $\sum_{s \in S} U_s(\bar{x}_s^{(q)}) = \theta^{(q)}(\bar{\lambda}^{(q)})$, we have the following.

Corollary 8 (ρ -Approximation Solution to the MP): Under the assumption A2, we have

$$\sum_{s \in S} U_s(\bar{x}_s^{(q)}) \leq \sum_{s \in S} U_s(x_s^*) \leq \rho \sum_{s \in S} U_s(\bar{x}_s^{(q)}). \quad (46)$$

If $\rho = 1.0$, (46) holds with equality, then Algorithm 3 is the column generation method with perfect global scheduling, and this algorithm converges to one optimum of MP.

Corollary 8 says that the column generation method with imperfect global scheduling converges to a sub-optimum of the MP and achieves the same approximation ratio as the approximate solution to the global scheduling problem. Finally,

Corollary 9 (Convergence under Perfect Scheduling):

Assume A2. Let $\rho = 1$ in Algorithm 3, which corresponds to perfect global scheduling. Then, Algorithm 3 converges to an optimum of the MP.

In [14], [15], the authors propose a way to solve this problem by a distributed subgradient algorithm with imperfect scheduling. With perfect scheduling, their approach guarantees the convergence of the link dual costs and the primal source rates; but it does not recover the time share fraction of the schedules, which oscillates due to the limitation of subgradient algorithm. However, with imperfect scheduling, their approach does not guarantee the convergence. Their performance bounds are not of the constant approximation ratio type, and they are dependent of the utility function. In contrast, our Algorithm 3 guarantees the convergence of the link dual costs, the source rates and the time share proportions; and it converges to a

sub-optimal solution whose function value is no less than a constant fraction of the true optimum value. The constant is independent of the utility function.

V. NUMERICAL EXAMPLES

In this section we will show the performance of our algorithm by simulation. We will use the following node exclusive interference model. The model requires that, first, the data rate of each link is fixed at c_e ; and second, at any time instance, each node can only send to or receive from one other node. Under this model, the scheduling problem (15) becomes the *maximum weighted matching (MWM)* problem [14], [15], [18]. There is a polynomial-time algorithm to solve MWM precisely [8] and a greedy algorithm to solve it approximately with an approximation ratio $\rho = 2$. The greedy algorithm is more useful to our problem because it is distributed [14]. Under this model, our column generation algorithm with imperfect scheduling will converge to an approximate solution for the MP with an approximation ratio $\rho = 2$, and it is completely decentralized.

We remark that the node exclusive interference model is a simple instance of the conflict-graph-based models that capture the contention relations among the links [4], [6]. In a conflict graph, each vertex represents one wireless link in the network, and an edge represents contention between the two corresponding links, which are not allowed to transmit at the same time. A set of links in the wireless network that can transmit data simultaneously, i.e., a schedule, is an independent set in the corresponding conflict graph. The scheduling problem (15) becomes the *maximum weighted independent set (MWIS)* problem, where the edge weight is $\lambda_e c_e$. Generally, MWIS has no approximate solution. Some heuristics or random search algorithms seem necessary to carry out the imperfect scheduling.

The possible choices of utility function $U_s(x_s)$ could be

$$U_s(x_s) = w_s \ln(x_s + e) \quad (47)$$

or

$$U_s(x_s) = w_s \frac{(x_s + a_s)^{1-\beta}}{1-\beta}, \quad 0 < \beta < 1, \quad (48)$$

where w_s are the weights for $s \in S$, e is the base of the natural logarithm and $a_s > 0$ is a small constant, which make the utility functions (47) and (48) satisfy the assumptions A2 and A3. These utility functions have been discussed in [16]. In this paper, we will use the utility function in (47) with $w_s = 1.0$ for all $s \in S$.

As discussed in Section IV, we can introduce new extreme points at varying degree of frequency. In the experiments, we will use three frequencies: fast, medium and slow. With the fast frequency, we try to introduce extreme points by solving the global scheduling problem (15) at each slow-timescale update (17)-(18), in which case, Algorithm 3 degenerates into the pure two-timescale algorithm. With the slow frequency, we try to introduce a new extreme point after every 20 slow-timescale updates of (17)-(18). Our experiences have shown that the restricted master problem with our experiment sizes is often optimized within 20 slow-timescale updates. If so, Algorithm 3 becomes the pure column generation method. With the medium frequency, we introduce a new extreme point every 5 slow-timescale updates.

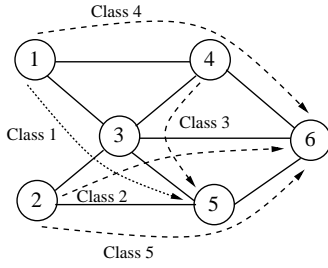


Fig. 1. Small Network Topology

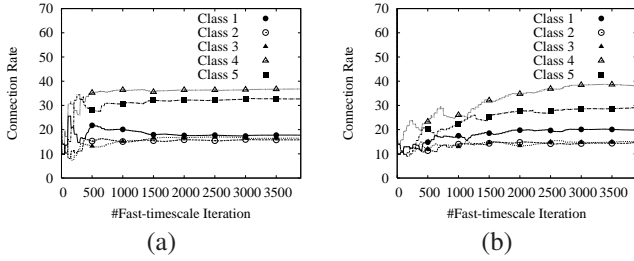


Fig. 2. Small Network: (a) Fast Frequency, with Perfect Global Scheduling; (b) Fast Frequency, with Imperfect Global Scheduling.

The network in Fig. 1 has been studied in [14], [15]. There are 5 classes of connections as shown in Fig. 1. The capacity of each link is fixed at 100 units. We initialize the experiments with a set of schedules, where each contains exactly one single transmitting link. This corresponds to the traditional TDMA scheduling [13]. Fig. 2 shows the convergence of the connection rates with perfect scheduling and imperfect scheduling, respectively, where both are introducing new columns at the fast frequency. In Fig. 2 (a), we have two groups of connections. Class 4 and Class 5 achieve higher rates because they involve less wireless interference compared with others. Fig. 2 (b) gives the same order of the connections in terms of their rates. But, the connections are not separated into obvious rate groups. Though the two scheduling schemes do not give the exactly same connection rates, their final objective function values are very close: 16.0989 for the imperfect scheduling and 16.1351 for the perfect one. We note that with our specific objective function in (48), a minor change in the connection rates will not change the objective too much. Fig. 3 shows the two schemes get the correct time fraction and the long time average link capacities are able to support the source flow rates. It means our two-timescale algorithm solves both the primal and dual problems at the same time.

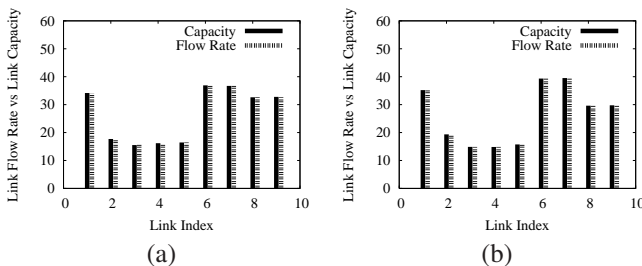


Fig. 3. Small Network: (a) Fast Frequency, with Perfect Global Scheduling; (b) Fast Frequency, with Imperfect Global Scheduling.

We next experiment with a larger network with 15 nodes. The network is randomly generated and 20 end-to-end connections are placed on this network randomly. For each connection, the routing is the fixed shortest path routing. In the experiment, it turns out these 20 connections use 28 directed links. The capacity of each link is fixed at 100 units. Fig. 4 shows the 5 connections with the highest rates. Again, the perfect scheduling is more likely to group connections.

Next, we evaluate the algorithm with different frequencies of introducing columns on the large network. In Fig. 5, with both perfect and imperfect scheduling, the fast scheme always improves the objective function value more quickly at the beginning, while the slow scheme improves it much more slowly than the other two schemes. The reason is that, with the fast scheme, plenty of schedules are introduced quickly. The slow scheme always tries to take full advantage of the current collection of schedules. But later, the slow scheme catches up the fast scheme, judging from the trend of the curves. This motivates the use of the medium scheme. In Fig. 5, we see that the medium scheme increases the objective function value nearly as quickly as the fast scheme at the beginning and it surpasses the fast scheme soon after. The curves show some oscillations at the initial phase for the medium and slow schemes. This is because those two schemes spend more effort to obtain better performance from the current collection of schedules. At the initial phase, with fewer schedules but more optimized time sharing, introducing one more schedule abruptly will decrease the function value by a little bit.

In Table I, we compare the three schemes for their computation costs. Since the most expensive computation is for solving the global scheduling problem (15), the total computation time is mainly characterized by the number of times the global scheduling problem is solved. One expects that lowering the frequency of introducing new schedules is correlated with fewer computations for the global scheduling problem. But, we know no theoretical reasons why this must be true. The result confirms the expectation: The number of such computations is 300, 60, and 15, for the fast, medium and slow schemes. The reduction is dramatic.

We also find, with a lower frequency, the algorithm usually produces a solution with fewer active schedules.³ Fewer active schedules may be desirable since it is easier to manage and control them, which may reduce the system complexity and control overhead. With the perfect scheduling, the slow scheme (i.e., the pure column generation approach) only uses (i.e., time-share) 15 active schedules in the end, which are all those that were ever computed and entered. In other words, there are no redundant schedules; nor are there redundant computations for the schedules. The fast and medium schemes use 49 active schedules. In the fast scheme, 7 schedules have been introduced into the collection but are not used in the final optimal solution. In the medium scheme, the number of redundant schedules is 3.

For the imperfect scheduling, we find that both the fast and the medium schemes generate much fewer schedules than in the perfect scheduling, although the number of computations for the schedules remain the same⁴. The fast scheme even has fewer redundant schedules than the medium scheme,

³In these 6 experiments, the initial TDMA-style schedules are all inactive in the optimal solutions, and we didn't count them in the table.

⁴However, each computation is less expensive than in the perfect scheduling case, since it is approximate.

which a little counter-intuitive. The reason might be that the approximation algorithm is not as sensitive to the change of link prices as the precise algorithm. Significant changes in link prices are needed to trigger the discovery of a new schedule.

Based on the study of Fig. 4 and Table I, we conclude that the pure two-timescale (fast) or the pure column generation (slow) algorithms have both pros and cons. An intermediate algorithm (medium) may achieve a more desirable balance among factors such as optimization performance, the computational cost, and system complexity and overhead.

Next, we show that, in the pure column generation method, the gap between the lower and upper bounds for the optimal object value decreases as the restricted master problem expands. With the imperfect scheduling, we can compute the upper bound by $\theta^{(q)}(\bar{\lambda}^{(q)}) - \gamma^{(q)}(\bar{\lambda}^{(q)}) + \gamma(\bar{\lambda}^{(q)}) \leq \theta^{(q)}(\bar{\lambda}^{(q)}) - \gamma^{(q)}(\bar{\lambda}^{(q)}) + \rho\gamma_\rho(\bar{\lambda}^{(q)})$, where $\rho = 2$ in our case. The lower bound is obtained from the current best solution. Fig. 6 shows that the gap is quickly narrowed after 10 columns have entered. It also shows that the objective values of both the perfect scheduling and imperfect scheduling are inside the two bounds. Also, our imperfect scheduling almost achieves the global optimum of the original problem.

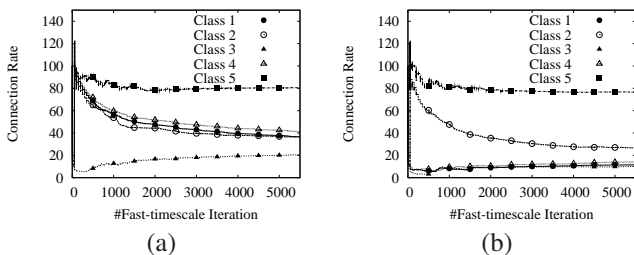


Fig. 4. Large Network: (a) Fast Frequency, with Perfect Global Scheduling; (b) Fast Frequency, with Imperfect Global Scheduling.

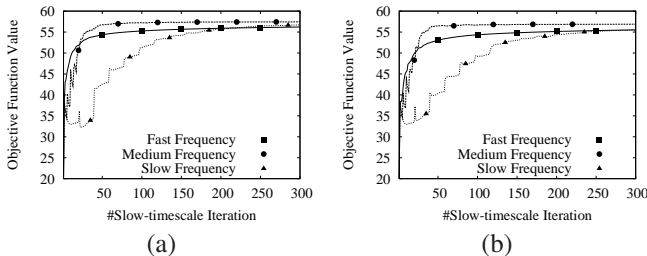


Fig. 5. Large Network: (a) Perfect Global Scheduling; (b) Imperfect Global Scheduling.

TABLE I
PERFORMANCE COMPARISON OF THE FAMILY OF ALGORITHMS

	Fast Per.	Medium Per.	Slow Per.
#Schedules Computed	300	60	15
#Active Schedules	49	49	15
#Schedules Introduced	56	52	15
	Fast Imper.	Medium Imper.	Slow Imper.
#Schedules Computed	300	60	15
#Active Schedules	19	19	15
#Schedules Introduced	22	30	15

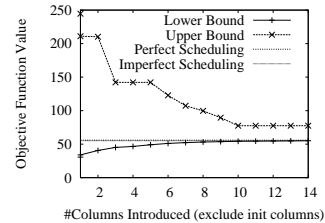


Fig. 6. Bounds for the Optimal Objective Value of the MP. Pure Column Generation Method with Imperfect Global Scheduling.

Finally, we have also applied the subgradient algorithms for these experiments, and found that it is very difficult to tune the algorithm parameters to reach convergence.

VI. CONCLUSIONS

This paper studies the problem of how to allocate wireless resources to maximize the aggregate source utility. This optimization problem has two difficulties: First, the Lagrangian function is not strictly concave with respect to the time-share variables, which makes the subgradient algorithm unable to recover the optimal values for those variables; second, its constraint set is a convex polytope usually containing an exponential number of extreme points. In order to recover the correct time-share variables, we develop a two-timescale algorithm. To overcome the difficulty of the global scheduling problem, we adopt a column generation approach with imperfect global scheduling. If the imperfect scheduling has bounded performance, then our overall utility optimization algorithm converges to a sub-optimum with bounded performance. The combination of the two-timescale algorithm and column generation leads to a family of algorithms with interesting tradeoffs.

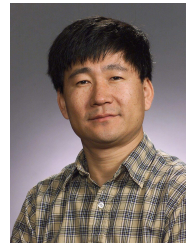
REFERENCES

- [1] Mokhtar S. Bazaraa, Hanif D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. Wiley-Interscience, 3rd edition, 2006.
- [2] Dimitri Bertsekas. *Nonlinear Programming*. Athena Scientific, 2nd edition, 1999.
- [3] Patrik Bjorklund, Peter Varbrand, and Di Yuan. Resource optimization of spatial TDMA in ad hoc radio networks: a column generation approach. In *Proceedings of INFOCOM*, 2003.
- [4] Stephan Bohacek and Peng Wang. Toward tractable computation of the capacity of multihop wireless networks. In *Proceedings of INFOCOM*, May 2007.
- [5] L. Chen, T. Ho, S. H. Low, M. Chiang, and J. C. Doyle. Optimization based rate control for multicast with network coding. In *Proceedings of IEEE INFOCOM*, 2007.
- [6] Lijun Chen, Steven H. Low, Mung Chiang, and John C. Doyle. Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks. In *Proceedings of the IEEE Infocom 2006*, Barcelona, Spain, April 2006.
- [7] Lijun Chen, Steven H. Low, and John C. Doyle. Joint congestion control and media access control design for ad hoc wireless networks. In *Proceedings of the IEEE Infocom 2005*, Miami, USA, March 2005.
- [8] H. N. Gabow. Data structures for weighted matching and nearest common ancestors with linking. In *Symposium on Discrete Algorithms*, 1990.
- [9] Robert G. Gallager. A minimum delay routing algorithm using distributed computation. *IEEE Transactions on Communications*, pages 73–85, January 1977.
- [10] Leonidas Georgiadis, Michael J. Neely, and Leandros Tassiulas. Resource allocation and cross-layer control in wireless networks. *Found. Trends New.*, 1(1):1–144, 2006.
- [11] Mikael Johansson and Lin Xiao. Cross-layer optimization of wireless networks using nonlinear column generation. *IEEE Transaction on Wireless Communications*, 5(2):435–445, Feb. 2006.

- [12] H. K. Khalil. *Nonlinear Systems*. Prentice-Hall, 1996.
- [13] Sastry Kompella, Jeffrey E. Wieselthier, and Anthony Ephremides. A cross-layer approach to optimal wireless link scheduling with SINR constraints. In *MilCom 2007*, 2007.
- [14] Xiaojun Lin, Ness B. Shroff, and R. Srikant. The impact of imperfect scheduling on cross-layer rate control in wireless networks. *IEEE/ACM Transaction on Networking*, 14(2):302–315, April 2006.
- [15] Xiaojun Lin, Ness B. Shroff, and R. Srikant. A tutorial on cross-layer optimization in wireless networks. *IEEE Journal on Selected Areas in Communications*, 24(8):1452–1463, Aug. 2006.
- [16] J. Mo and J. Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking*, 8(5), October 2000.
- [17] Fernando Paganini. Congestion control with adaptive multipath routing based on optimization. In *The 40th Annual Conference on Information Sciences and Systems*, 2006.
- [18] Gaurav Sharma, Ness B. Shroff, and Ravi R. Mazumdar. Maximum weighted matching with interference constraints. In *PERCOMW '06: Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops*, Washington, DC, 2006.
- [19] Gaurav Sharma, Ness B. Shroff, and Ravi R. Mazumdar. Joint congestion control and distributed scheduling for throughput guarantees in wireless networks. In *Proceedings of INFOCOM*, May 2007.
- [20] Leandros Tassiulas. Linear complexity algorithms for maximum throughput in radio networks and input queued switches. In *Proceedings of IEEE INFOCOM*, 1998.
- [21] Jiantao Wang, Lun Li, Steven H. Low, and John C. Doyle. Cross-layer optimization in TCP/IP networks. *IEEE/ACM Transactions on Networking*, 13(3):582 – 595, June 2005.
- [22] Jun Yuan, Zongpeng Li, Wei Yu, and Baochun Li. A cross-layer optimization framework for multicast in multi-hop wireless networks. In *Wireless Internet, 2005. Proceedings. First International Conference on*, pages 47–54, July 2005.
- [23] H. Zhai and Y. Fang. Impact of routing metrics on path capacity in multi-rate and multi-hop wireless ad hoc networks. In *Proceedings of The 14th IEEE International Conference on Network Protocols (ICNP'06)*, 2006.

PLACE
PHOTO
HERE

area, including performance evaluation of network protocols and algorithms, congestion control, resource allocation, and load balancing on peer-to-peer networks. He is also interested in probability theory, stochastic processes and queuing theory.



Yuguang "Michael" Fang (IEEE, S'92-M'94-S'96-M'97-SM'99-F'08) received a Ph.D. degree in Systems Engineering from Case Western Reserve University in January 1994 and a Ph.D. degree in Electrical Engineering from Boston University in May 1997. He was an assistant professor in the Department of Electrical and Computer Engineering at New Jersey Institute of Technology from July 1998 to May 2000. He then joined the Department of Electrical and Computer Engineering at University of Florida in May 2000 as an assistant professor, got an early promotion to an associate professor with tenure in August 2003 and to a full professor in August 2005. He holds a University of Florida Research Foundation (UFRF) Professorship from 2006 to 2009. He has published over 200 papers in refereed professional journals and conferences. Dr. Fang received the National Science Foundation Faculty Early Career Award in 2001 and the Office of Naval Research Young Investigator Award in 2002, and is the recipient of the Best Paper Award in IEEE International Conference on Network Protocols (ICNP) in 2006 and the recipient of the IEEE TCGN Best Paper Award in the IEEE High-Speed Networks Symposium, IEEE Globecom in 2002. Dr. Fang is also active in professional activities. He is a Fellow of IEEE and a member of ACM. He has served on several editorial boards of technical journals including IEEE Transactions on Communications, IEEE Transactions on Wireless Communications, IEEE Wireless Communications Magazine and ACM Wireless Networks. He was an editor for IEEE Transactions on Mobile Computing and currently serves on its Steering Committee. He has been actively participating in professional conference organizations such as serving as the Steering Committee Co-Chair for QShine, the Technical Program Vice-Chair for IEEE INFOCOM'2005, Technical Program Symposium Co-Chair for IEEE Globecom'2004, and a member of Technical Program Committee for IEEE INFOCOM (1998, 2000, 2003-2009).

PLACE
PHOTO
HERE

Xiaoying Zheng is a PhD candidate in the Department of Computer and Information Science and Engineering at the University of Florida. She received her bachelor's and master's degrees in computer science and engineering from Zhejiang University, P.R. China, in 2000 and 2003, respectively. Her research interests include applications of optimization theory in networks, peer-to-peer overlay networks, content distribution and congestion control.

PLACE
PHOTO
HERE

Feng Chen received BE degree from Electrical Engineering Department in Huazhong University of Science and Technology, Wuhan, China in July, 2005. Since August, 2005, she has been pursuing the PhD Degree in the Department of Electrical and Computer Engineering, University of Florida. She is a student member of the IEEE. She is also a member in Eta Kappa Nu.