Cross-Layer Design in Multihop Wireless Networks

Lijun Chen, Steven H. Low, John C. Doyle

Engineering and Applied Science Division, California Institute of Technology, Pasadena, CA 91125, USA

Abstract

In this paper, we take a holistic approach to the protocol architecture design in multihop wireless networks. Our goal is to integrate various protocol layers into a rigorous framework, by regarding them as distributed computations over the network to solve some optimization problem. Different layers carry out distributed computation on different subsets of the decision variables using local information to achieve individual optimality. Taken together, these local algorithms (with respect to different layers) achieve a global optimality. Our current theory integrates three functions—congestion control, routing and scheduling—in transport, network and link layers into a coherent framework. These three functions interact through and are regulated by congestion price so as to achieve a global optimality, even in a time-varying environment. Within this context, this model allows us to systematically derive the layering structure of the various mechanisms of different protocol layers, their interfaces, and the control information that must cross these interfaces to achieve a certain performance and robustness.

Keywords: Theory-based approach, Cross-layer design, Dual decomposition, Multihop wireless networks

1. Introduction

The success of communication network has largely been a result of adopting a layered architecture. With this architecture, its design and implementation is divided into simpler modules that are separately designed and implemented and then interconnected. A protocol stack typically has five layers, application, transport (TCP), network (IP), data link (include MAC) and physical layer. Each layer controls a subset of the decision variables, hides the complexity of the layer below and provides well-defined services to the layer above. Together, they allocate networked resources to provide a reliable and usually best-effort communication service to a large pool of competing users.

However, the layered structure addresses only abstract and high-level aspects of the whole network protocol design. Various layers of the network are put together often in an ad hoc manner, and might not be optimal as a whole. In order to improve the performance and achieve efficient resource allocation, we need to understand interactions across layers and carry out cross-layer design. Moreover, in wireless networks there does not exist a good interface between the physical and network layers. Wireless links are unreliable and wireless nodes usually rely on random access mechanism to access wireless channel. Thus, the performance of link layer is not guaranteed, which will result in performance problems for the whole network such as degraded TCP performance. So, we need cross-layer design, i.e., to exchange information between physical/link layer with higher layers in order to achieve better performance.

Motivated by the duality model of TCP congestion control [16] [22] [18] [23], one approach to understand interactions across layers is to view the network as an optimization solver and various protocol layers as distributed algorithms solving an optimization problem. This approach and the associated utility maximization problem were originally proposed as an analytical tool for reverse engineering TCP congestion control where a network with fixed link capacities and prespecified routes is implicitly assumed. A natural framework for cross-layer design is then to extend the basic utility maximization problem to include decision variables of other layers, and seek a decomposition such that different layers carry out distributed computation on different subsets of decision variables using local information to achieve individual optimality, and taken together, these local algorithms achieve the global optimality. This approach has come to be known as layering-as-optimization-decomposition; see [6] for an extensive survey.

We apply this approach to design an overall framework for the protocol architecture in multihop wireless networks, with the goal of achieving efficient resource allocation through cross-layer design. We first consider the network with fixed channel or single-rate devices, and formulate network resource allocation as a utility maximization problem with rate constraints at the network layer and schedulability constraints at the link layer. We then apply duality theory to decompose the system problem vertically into congestion control, routing and scheduling subproblems that interact through congestion prices. Based on this decomposition, a distributed subgradient algorithm for joint congestion control, routing and scheduling is obtained, and proved to approach arbitrarily close to the optimum of the system problem. We next extend the dual subgradient algorithm to wireless multihop networks with time-varying channels and adaptive multi-rate devices. The stability of the resulting system is proved, and its performance is characterized with respect to an ideal reference system. We finally apply the general algorithm to the joint congestion control and medium access control design over the network with single-path routing and to the cross-layer congestion control, routing and scheduling design in the network without prespecified paths.

Our current theory integrates three functions— congestion control, routing and scheduling—in transport, network and link layers into a coherent framework. While the integration of all protocol components remain a big challenge, this framework is promising to be extended to provide a mathematical theory for network architecture, and allow us to systematically derive the layering structure of the various mechanisms of different protocol layers, their interfaces, and the control information that must cross these interfaces to achieve a certain performance and robustness. We also present a general technique and results regarding the stability and optimality of dual algorithm in face of time-varying parameters. As the flow contention graph that will be used to characterize feasible rate regions of the networks is a rather general construction and can be used to capture the interdependence or contention among parallel servers of any queueing networks, these results are applicable to any systems that can be modelled by a general model of queueing network that is served by a set of interdependent parallel servers with time-varying service capabilities.

The remainder of this paper is organized as follows. The next section briefly discusses related work. Section 3 presents details of the system model for the network with fixed channel or single-rate devices, and section 4 presents a distributed algorithm for joint congestion control, routing and scheduling via dual decomposition. Section 5 extends the dual algorithm to handle the network with time-varying channel and adaptive multi-rate devices. As specific cases of the general model and algorithm developed in sections 4 and 5, section 6 discusses joint congestion control and medium access control design in multihop wireless networks with single-path routing, and section 7 discusses cross-layer congestion control, routing and scheduling design in the network without prespecified paths. We conclude in section 8.

2. Related Work

The utility maximization framework [16] [22] [18] on TCP congestion control has been extensively applied and extended to study protocol design, especially congestion control (see, e.g., [48] [49]), fair channel access (see, e.g., [25] [39] [19] [9] [35] [8]), and cross-layer design (see, e.g., [44] [5] [20] [3] [4]), in wireless networks. Xue et al. [48] and Yi et al. [49] are among the first to formulate schedulability constraints at link layer for congestion control over multihop wireless networks. Xiao et al. [44] study joint routing and resource allocation, and are among the first to apply dual decomposition to cross-layer design in wireless networks. Chiang [5] is among the first to study joint congestion and power control. Lin et al. [20] and Chen et al. [3] are among the first to study joint congestion control and scheduling. Chen et al. [3] and Wang at al. [41] are among the first to study cross-layer design in the network with contention-based medium access.

The work presented in section 6 (see also [3]) is originally motivated to solve TCP unfairness problem over multihop wireless networks; see, e.g., [10], [37], [45], [46], [47]. The model used in section 7 (see also [4]) is motivated by Neely et al. [26] that studies dynamic power control and routing for time-varying wireless networks and by Hajeck et al. [11] and Kodialam at al. [17] that study joint routing and scheduling to determine the achievable rates in multi-hop wireless networks; similar decomposition for the network with deterministic wireless channel has also been revealed in the journal version of [26] and in [20].

The utility maximization in time-varying wireless networks is first studied in the context of fair scheduling. It has been shown that a family of primal scheduling algorithms maximize the sum of the utilities of the long-run average data rates provided to the users; see, e.g., [39] [19] [35]. In contrast, the result presented in section 5 (see also [4]) is for the dual algorithms. An earlier result for the dual scheduling algorithm is by Eryilmaz et al. [8] that studies fair resource allocation using queue-length based scheduling and congestion control. Another similar result is by Neely et al. [27] that studies fairness and optimal stochastic control for heterogeneous networks. All these three works use stochastic Lyapunov method to establish stability, but the technical details are somewhat different. Especially, the stability and optimality result presented in section 5 is based only on general properties of convexity and the definition of subgradients, and can be directly applied to a variety of time-varying systems that can be solved or modelled by the dual algorithms. Another comparable result is by Stolyar [36] that proposes greedy primal-dual algorithm to maximize network utility. It uses a very different technique to establish optimality.

Here, we focus on dual decomposition, but there are many different ways to decompose a given problem, each of which corresponds to a different layering scheme. See the survey article [6] and the references therein for various recent work on cross-layer design.

3. System Model

Consider a multihop wireless network with a set N of nodes and a set L of directed logical links. We assume a static topology and each link $l \in L$ has a fixed finite capacity c_l bits per second when active, i.e., we implicitly assume that the wireless channel is fixed or some underlying mechanism is used to mask the channel variation so that the wireless channel appears to have a fixed rate. This assumption will be relaxed in section 5. Wireless channel is interference limited, where links contend with each other for exclusive access to the channel. We will use the flow contention graph to capture the contention relations among links. The feasible rate region at link layer is then a convex hull of the corresponding rate vectors of independent sets of the flow contention graph. We will further describe rate constraints at the network layer by linear inequalities in terms of user service requirements and allocated link capacities. The resource allocation of the network is then formulated as a utility maximization problem with schedulability and rate constraints.

3.1. Flow Contention Graph and Schedulability Constraint

The interference among wireless links is usually specified by some interference model that describes physical constraints regarding wireless transmissions and successful receptions. For example, in a network with primary interference, links that share a common node cannot transmit or received simultaneously but links that do not share nodes can do so. It models a wireless network with multiple channels where simultaneous communications in a neighborhood are enabled by using orthogonal CDMA or FDMA channels. In a network with secondary interference, links mutually interfere with each other whenever either the sender or the receiver of one is within the interference range of the sender or receiver of the other. Given an interference model, we can construct a flow contention graph that captures the contention relations among the links; see, e.g., [25]. In the contention graph, each vertex represents a link, and an edge between two vertices denotes the contention between the corresponding links: two links interfere with each other and cannot transmit at the same time. Figure 1 shows an example of a simple multihop wireless network with primary interference and the corresponding flow contention graph.

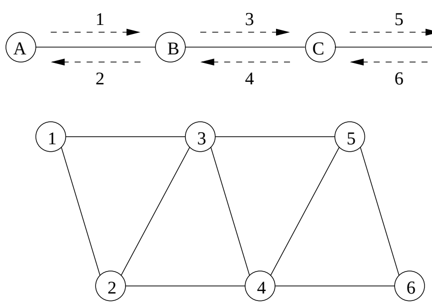

Figure 1: Example of a multihop wireless network with 4 nodes and 6 logical links and the corresponding flow contention graph.

Given a flow contention graph, we can identify all its independent sets of vertices. An independent set is a set of vertices that have no edges between each other [7]. The links in an independent set can transmit simultaneously. Let E denote the set of all independent sets with each independent set indexed by e. We represent an independent set e as a $|L|$-dimensional rate vector r^e, where the lth entry is

$$r_l^e := \begin{cases} c_l & \text{if } l \in e, \\ 0 & \text{otherwise.} \end{cases}$$

The feasible rate region Π at the link layer is then defined as the convex hull of these rate vectors

$$\Pi := \left\{ r : r = \sum_e a_e r^e, \ a_e \geq 0, \ \sum_e a_e = 1 \right\}. \tag{1}$$

Thus, a link flow vector y satisfies schedulability constraint if $y \in \Pi$.

The contention graph is a general construct, and can capture the interdependence or contention among parallel servers of any queueing networks. For example, it can characterize the contention relations in the network where wireless nodes are equipped with multiple radios or communicate through multiple channels.

3.2. Rate Constraint

Let $f_l \geq 0$ denote the amount of capacity allocated to link l. From the schedulability constraint, f should satisfy

$$f \in \Pi. \tag{2}$$

Assume that the network is shared by a set S of sources, with each source $s \in S$ transmitting at rate x_s bits per second. In the following, we will formulate rate constraints for networks with different kinds of routing.

The Network with Single-Path Routing

Each source s uses a path consisting of a set $L_s \subset L$ of links. The sets L_s define an $|L| \times |S|$ routing matrix

$$R_{ls} = \begin{cases} 1 & \text{if } l \in L_s, \\ 0 & \text{otherwise.} \end{cases}$$

Thus, the aggregate rate over link l is $\sum_{s \in S} R_{ls} x_s$. The rate constraint is written as

$$Rx \leq f, \tag{3}$$

i.e., the aggregate link rate should not exceed the link capacity.

The Network with Multipath Routing

Each source s can send traffic along a set T_s of given paths. Each path $r \in T_s$ contains a set of $L_r^s \subset L$ of links, which defines a $|L| \times |T_s|$ routing matrix H^s whose (l, r)th entry is given by

$$H_{lr}^s = \begin{cases} 1 & \text{if } l \in L_r^s, \\ 0 & \text{otherwise.} \end{cases}$$

Denote by x_s^r the rate at which source s sends along path r. Thus, the source rate $x_s = \sum_{r \in T_s} x_s^r$. The rate constraint is written as

$$\sum_{s, \ r \in T_s} H_{lr}^s x_s^r \leq f_l, \ l \in L. \tag{4}$$

The Network without Prespecified Paths

Since no end-to-end path is given, we will use multicommodity flow model for routing. Let D denote the set of destination nodes of network layer flows. Let $f_{i,j}^k \geq 0$ denote the amount of capacity of link (i, j) allocated to the flows to destination k. Then the aggregate capacity on link (i, j) is $f_{i,j} := \sum_{k \in D} f_{i,j}^k$. Let $x_i^k \geq 0$ denote the flow generated at node i towards destination k. Then the aggregate capacity for its incoming flows and generated flow to the destination k should not exceed the summation of the capacities for its outgoing flows to k

$$x_i^k \leq \sum_{j:(i,j) \in L} f_{i,j}^k - \sum_{j:(j,i) \in L} f_{j,i}^k, \ i \in N, \ k \in D, \ i \neq k. \tag{5}$$

Equation (5) is the rate constraint for resource allocation. For simplicity of presentation, we assume that there is at most one flow between any node and destination pair $[i, k] \in S \times D$. Thus, $x_i^k = x_s$ if i is the source node of flow $s = [i, k]$, and $x_i^k = 0$ otherwise.

3.3. Problem Formulation

We see from the last subsection that all three kinds of rate constraints are expressed as linear inequalities. If we represent the "routing" of the user (source) service requirement by a linear function $H(x)$ of the source rates x, and represent the "allocation" of the service capacity by a linear function $A(f)$ of the link capacities f, since the service requirement should not exceed the allocated service capacity, we have the following inequality constraint

$$H(x) \leq A(f). \tag{6}$$

The linear constraint (6) is a very general relation. The rate constraints (3), (4) and (5) are just its different concrete representations.

Following [16] [22] [18], assume each source s attains a utility $U_s(x_s)$ when it transmits at a rate x_s. We assume $U_s(\cdot)$ is continuously differentiable, increasing, and strictly concave. Our objective is to choose source rates x and allocated capacities f so as to solve the following global problem

$$\max_{x,f} \quad \sum_s U_s(x_s) \tag{7}$$

$$\text{subject to} \quad H(x) \leq A(f), \tag{8}$$

$$f \in \Pi. \tag{9}$$

The system problem (7)–(9) is a convex optimization problem, but it is impractical to solve it centrally in real networks. Distributed algorithm can be derived by solving its Lagrange dual problem, as we will show in the next section.

4. Distributed Algorithm via Dual Decomposition

4.1. Distributed Algorithm

Consider the Lagrangian of the problem (7)–(9) with respect to the rate constraint

$$L(p, x, f) = \sum_s U_s(x_s) - p^T(H(x) - A(f)).$$

Given p, the above Lagrangian has a nice decomposition structure: it is the summation of two independent terms, of source rates and link capacities respectively. Interpreting p as the "congestion price" and maximizing the Lagrangian over x and f for fixed p, we obtain the following joint congestion control and scheduling algorithm:

Congestion control: At time t, given congestion price $p(t)$, the sources adjust flow rates x according to the congestion price

$$x(t) = x(p(t)) = \arg\max_x \sum_s U_s(x_s) - p^T(t)H(x). \tag{10}$$

Scheduling: Over link l, send an amount of data for each flow according to the rates f such that

$$f(t) = f(p(t)) \in \arg\max_{f\in\Pi} p^T(t)A(f). \tag{11}$$

Note that there does not exist an explicit routing component in the dual decomposition. Instead, the routing is implicitly solved in (10) if the set of paths from which a source can choose is given, and solved in (11) if no path is prespecified for the source. We see that, by dual decomposition, the flow optimization problem decomposes into separate "local" optimization problems of transport, network and link layers respectively, and they interact through congestion prices.

Defining dual function $D(p) = \max_{x,f\in\Pi} L(p, x, f)$, by duality we have (see, e.g., Chapter 5 in [2])

$$\max_{x,f} \sum_s U_s(x_s) = \min_{p\geq 0} D(p) = \min_{p\geq 0} \max_{x,f\in\Pi} L(p, x, f).$$

The dual problem $\min_p D(p)$ can be solved by using the subgradient method [33] [2], where the Lagrangian multipliers are adjusted in the opposite direction to the subgradient of the dual function

$$g(p) = A(f(p)) - H(x(p)). \tag{12}$$

Congestion price update: The network (links or nodes) updates the congestion price, according to

$$p(t + 1) = [p(t) + \gamma_t(H(x(p(t))) - A(f(p(t))))]^+, \tag{13}$$

where γ_t is a positive scalar stepsize, and "+" denotes the projection onto the set $\Re^+$ of nonnegative real numbers. The algorithm has a nice interpretation in terms of law of supply and demand and their regulation through pricing. Equation (13) says that, if the demand $H(x(p(t)))$ for service capacity exceeds the supply $A(f(p(t)))$, the price p will rise, which will in turn decrease the demand (see equation (10)) and increase the supply (see equation (11)).

Before proceeding, we explain the notation used in this paper. We denote a link either by a single index l or by the directed pair (i, j) of nodes it connects. We use s or alternatively node pair $[i, k]$ to denote a network layer flow. We overload the use of the source rate x, the link capacity f and the congestion price p throughout this paper, depending on different kinds of routing involved. For example, x refers to the source rate x_s or the source rate x_i^k at node i towards destination k, depending on the specific context. Similarly, f refers to both the link capacity $\{f_{i,j}\}$ and the capacity $\{f_{i,j}^k\}$ over link (i, j) that is allocated to destination k.

4.2. Convergence Analysis

Using results on the convergence of the subgradient method [33] [2], we show that, for constant stepsize, the algorithm is guaranteed to converge to within a neighborhood of the optimal value. For diminishing stepsize, the algorithm is guaranteed to converge to the optimal value. We would like a distributed implementation of the subgradient algorithm, and thus a constant stepsize $\gamma_t = \gamma$ is more convenient. Note that the dual cost usually will not monotonically approach the optimal value, but wander around it under the subgradient algorithm. The usual criterion for stability and convergence is not applicable. Here we define convergence in a statistical sense [3]. Let $\overline{p}(t) := \frac{1}{t} \sum_{\tau=1}^{t} p(\tau)$ be the average price by time t.

Definition 1. *Let p^* denote an optimal value of the dual variable. Algorithm (10)–(13) with constant stepsize is said to converge* statistically *to p^*, if for any $\delta > 0$ there exists a stepsize γ such that* $\limsup_{t\to\infty} D(\overline{p}(t)) - D(p^*) \leq \delta$.

Clearly, an optimal value p^* exists. The following theorem guarantees the statistical convergence of the subgradient method.

Theorem 2. *Let p^* be an optimal price. If the norm of the subgradients is uniformly bounded, i.e., there exists G such that $\|g(p)\|_2 \leq G$ for all p, then*

$$D(p^*) \leq \limsup_{t\to\infty} D(\overline{p}(t)) \leq D(p^*) + \frac{\gamma G^2}{2}, \tag{14}$$

i.e., the algorithm (10)–(13) converges statistically to p^*.

Proof. The first inequality $D(p^*) \leq \limsup_{t\to\infty} D(\overline{p}(t))$ always holds, since $D(p^*)$ is the minimum of the dual function $D(p)$. Now we prove the second inequality. By equation (13), we have

$$\begin{aligned}
& \|p(t+1) - p^*\|_2^2 \\
= \; & \|[p(t) - \gamma g(p(t))]^+ - p^*\|_2^2 \\
\leq \; & \|p(t) - \gamma g(p(t)) - p^*\|_2^2 \\
= \; & \|p(t) - p^*\|_2^2 - 2\gamma g(p(t))^T(p(t) - p^*) + \gamma^2\|g(p(t))\|_2^2 \\
\leq \; & \|p(t) - p^*\|_2^2 - 2\gamma(D(p(t)) - D(p^*)) + \gamma^2\|g(p(t))\|_2^2,
\end{aligned}$$

where the last inequality follows from the definition of subgradient. Applying the inequalities recursively, we obtain

$$\begin{aligned}
\|p(t+1) - p^*\|_2^2 \quad \leq \quad & \|p(1) - p^*\|_2^2 - 2\gamma \sum_{\tau=1}^{t}(D(p(\tau)) - D(p^*)) \\
& + \gamma^2 \sum_{\tau=1}^{t} \|g(p(\tau))\|_2^2.
\end{aligned}$$

Since $\|p(t+1) - p^*\|_2^2 \geq 0$, we have

$$\begin{aligned}
2\gamma \sum_{\tau=1}^{t}(D(p(\tau)) - D(p^*)) \quad & \leq \quad \|p(1) - p^*\|_2^2 + \gamma^2 \sum_{\tau=1}^{t} \|g(p(\tau))\|_2^2 \\
& \leq \quad \|p(1) - p^*\|_2^2 + t\gamma^2 G^2.
\end{aligned}$$

From this inequality we obtain

$$\frac{1}{t}\sum_{\tau=1}^{t} D(p(\tau)) - D(p^*) \quad \leq \quad \frac{\|p(1) - p^*\|_2^2}{2t\gamma} + \frac{\gamma G^2}{2}.$$

Since D is a convex function, by Jensen's inequality,

$$D(\overline{p}(t)) - D(p^*) \quad \leq \quad \frac{\|p(1) - p^*\|_2^2}{2t\gamma} + \frac{\gamma G^2}{2}.$$

Thus, $\limsup_{t\to\infty} D(\overline{p}(t)) \leq D(p^*) + \frac{\gamma G^2}{2}$, i.e., the algorithm converges converges statistically to p^*. □

The assumption of bounded norm for subgradient $g(p)$ is reasonable, since f is finite and we always have an upper bound on x in practice. Theorem 2 implies that the congestion price p approaches p^* statistically when the stepsize γ is small enough.

Let the primal function be $P(x) := \sum_s U_s(x_s)$ and achieve its optimum at x^*. Define $\overline{x}(t) := \frac{1}{t}\sum_{\tau=1}^{t} x(\tau)$, the average data rate up to time t. As time goes to infinity, $\overline{x}(t)$ must be in the feasible rate region (determined by equations (8)–(9)), otherwise $\overline{p}(t)$ will go unbounded as time goes to infinity, which contradicts Theorem 2.

Theorem 3. *Let x^* be the optimal source rates. Under the same assumption of Theorem 2, the algorithm (10)–(13) converges statistically to within a small neighborhood of the optimal values $P(x^*)$, i.e.,*

$$P(x^*) \geq \liminf_{t\to\infty} P(\overline{x}(t)) \geq P(x^*) - \frac{\gamma G^2}{2}. \tag{15}$$

Proof. The first inequality $P(x^*) \geq \liminf_{t\to\infty} P(\overline{x}(t))$ holds, since $\overline{x}(t)$ is in the feasible rate region as t goes to infinity. Now we prove the second inequality. By equation (13), we have

$$\begin{aligned}
& \|p(t+1)\|_2^2 \\
\leq \; & \|p(t) - \gamma g(p(t))\|_2^2 \\
= \; & \|p(t)\|_2^2 - 2\gamma g(p(t))^T p(t) + \gamma^2\|g(p(t))\|_2^2 \\
= \; & \|p(t)\|_2^2 + 2\gamma \sum_s U_s(x_s(t)) - 2\gamma(\sum_s U_s(x_s(t)) \\
& -p^T(t)H(x(t))) - 2\gamma p^T(t)A(f(t)) + \gamma^2\|g(p(t))\|_2^2 \\
\leq \; & \|p(t)\|_2^2 + 2\gamma \sum_s U_s(x_s(t)) - 2\gamma(\sum_s U_s(x_s^*) \\
& -p^T(t)H(x^*)) - 2\gamma p^T(t)A(f(t)) + \gamma^2\|g(p(t))\|_2^2 \\
= \; & \|p(t)\|_2^2 + 2\gamma P(x(t)) - 2\gamma P(x^*) + \gamma^2\|g(p(t))\|_2^2 \\
& -2\gamma p^T(t)(A(f(t)) - H(x^*)) \\
\leq \; & \|p(t)\|_2^2 + 2\gamma P(x(t)) - 2\gamma P(x^*) + \gamma^2\|g(p(t))\|_2^2,
\end{aligned}$$

where the second inequality follows from the fact that $x(t)$ is the maximizer in the problem (10), and the third inequality follows from the fact that $f(t)$ is the maximizer in problem (11). Applying the inequalities recursively, we obtain

$$\|p(t+1)\|_2^2 \leq \|p(1)\|_2^2 + 2\gamma \sum_{\tau=1}^{t}(P(x(\tau)) - P(x^*)) + \gamma^2 \sum_{\tau=1}^{t} \|g(p(\tau))\|_2^2.$$

Since $\|p(t+1)\|_2^2 \geq 0$, we have

$$\begin{aligned}
2\gamma \sum_{\tau=1}^{t}(P(x(\tau)) - P(x^*)) \quad & \geq \quad -\|p(1)\|_2^2 - \gamma^2 \sum_{\tau=1}^{t} \|g(p(\tau))\|_2^2 \\
& \geq \quad -\|p(1)\|_2^2 - t\gamma^2 G^2.
\end{aligned}$$

From this inequality we obtain

$$\frac{1}{t}\sum_{\tau=1}^{t} P(x(\tau)) - P(x^*) \geq \frac{-\|p(1)\|_2^2 - t\gamma^2 G^2}{2t\gamma}.$$

Since P is a concave function, by Jensen's inequality,

$$P(\overline{x}(t)) - P(x^*) \geq \frac{-\|p(1)\|_2^2 - t\gamma^2 G^2}{2t\gamma}.$$

Thus, $\liminf_{t\to\infty} P(\overline{x}(t)) \geq P(x^*) - \frac{\gamma G^2}{2}$, i.e., the algorithm (10)–(13) converges statistically to within a small neighborhood of the optimal values $P(x^*)$. □

Since $P(x)$ is continuous, Theorem 3 implies that the average source rate approaches the optimal x^* when γ is small enough.

5. Extension to Networks with Time-Varying Channels

In the last section, we consider wireless multihop networks with fixed channels or single-rate devices, i.e., the capacity c_l is a constant when link l is active. However, recent years have seen the growing popularity and demand of multi-rate wireless network devices (e.g., 802.11a cards) that can adjust transmission rate according to the time-varying channel state and improve overall network utilization. Here, we consider the networks with time-varying channels and adaptive multi-rate devices.

5.1. Distributed Algorithm

We assume that time is slotted, and the channel is fixed within a time slot but independently changes between different slots.[1] Let $h(t)$ denote the channel state in time slot t. Corresponding to the channel state h, the capacity of link l is $c_l(h)$ when active and the feasible rate region at the link layer is $\Pi(h)$, which is defined in a similar way as in (1). We further assume that the channel state is a finite state process with identical distribution $q(h)$ in each time slot,[2] and define the mean feasible rate region as

$$\overline{\Pi} := \{\bar{r} : \bar{r} = \sum_h q(h) r(h), r(h) \in \Pi(h)\}. \tag{16}$$

Ideally, we would like to have a distributed algorithm that solves the following utility maximization problem

$$\max_{x,f} \quad \sum_s U_s(x_s) \tag{17}$$

$$\text{subject to} \quad H(x) \leq A(f), \tag{18}$$

$$f \in \overline{\Pi}. \tag{19}$$

However, if we solve the above problem via dual decomposition, we may get a link rate assignment which is infeasible for the channel state at a given time slot. Instead we directly extend the algorithm (10)–(13) to handle time-varying channel.

Congestion control: At time t, given congestion price $p(t)$, the sources adjust flow rates x according to the congestion price

$$x(t) = x(p(t)) = \arg\max_x \sum_s U_s(x_s) - p^T(t) H(x). \tag{20}$$

Scheduling: In the beginning of period t, each node monitors the channel state $h(t)$, and over link l send an amount of data for each flow according to the rates f such that

$$f(t) = f(p(t)) \in \arg\max_{f \in \Pi(h(t))} p^T(t) A(f). \tag{21}$$

Congestion price update: The network (links or nodes) updates the congestion price, according to

$$p(t+1) = \lfloor [p(t) + \gamma(H(x(p(t))) - A(f(p(t))))]^+ \rfloor. \tag{22}$$

[1]It is straightforward to extend our results to a network where the channel state process is modulated by a hidden Markov chain.

[2]Even if the channel state is a continuous process, we only have finite choices of modulation schemes. The corresponding capacities take discrete values.

Here "$\lfloor\ \rfloor$" denotes the function floor with respect to γ^2. We choose such a discrete congestion price to facilitate the stability analysis in the next subsection.

The above algorithm cannot be derived from the dual decomposition of the problem (17)–(19). However, we will use the problem (17)–(19) as a reference system, and characterize the performance of the above algorithm with respect to it.

5.2. Stochastic Stability

Note that congestion price $p(t)$ takes discrete values. Thus, congestion price $p(t)$ evolves according to a discrete-time, discrete-space Markov chain. We need to show that this markov chain is stable, i.e., the congestion price process reaches a steady state and does not become unbounded. It is easy to check that the Markov chain has a countable state space, but is not necessarily irreducible. In such a general case, the state space is partitioned in transient set T and different recurrent classes R_i. We define the system to be *stable* if all recurrent states are positive recurrent and the Markov process hits the recurrent states with probability one [38]. This will guarantee that the Markov chain will be absorbed/reduced into some recurrent class, and the positive recurrence ensures the ergodicity of the Markov chain over this class. We have the following result.

Theorem 4. *The Markov chain described by equation (22) is stable.*

Proof. Denote the dual function of the problem (17)–(19) by $\overline{D}(p)$ with an optimal price p^* and subgradient $\bar{g}(p)$, i.e., $\bar{g}(p) = A(\bar{f}(p)) - H(x(p))$ with $\bar{f}(p) \in \arg\max_{f \in \overline{\Pi}} p^T A(f)$. Consider the the Lyapunov function $V(p) = \|p - p^*\|_2^2$, we have

$$\begin{aligned}
& E[\Delta V_t(p)|p] \\
= \quad & E[V(p(t+1)) - V(p(t)) \mid p(t) = p] \\
= \quad & E[V(\lfloor [p(t) - \gamma g(p(t))]^+ \rfloor) - V(p(t)) \mid p(t) = p] \\
= \quad & E[V([p(t) - \gamma g(p(t))]^+ - \epsilon) - V(p(t)) \mid p(t) = p] \\
\leq \quad & E[V(p(t) - \gamma g(p(t))) - V(p(t)) \mid p(t) = p] \\
& + E[-2\epsilon \cdot ([p(t) - \gamma g(p(t))]^+ - p^*) + \|\epsilon\|_2^2 \mid p(t) = p] \\
\leq \quad & E[V(p(t) - \gamma g(p(t))) - V(p(t)) \mid p(t) = p] \\
& + 2\epsilon \cdot p^* + \|\epsilon\|_2^2,
\end{aligned}$$

where $\epsilon = [p(t) - \gamma g(p(t))]^+ - \lfloor [p(t) - \gamma g(p(t))]^+ \rfloor$. Note that $\mathbf{0} \leq \epsilon < \gamma^2 \mathbf{1}$, with $\mathbf{0}$ denoting the zero vector and $\mathbf{1}$ the vector with every component being 1. Thus, $2\epsilon \cdot p^* + \|\epsilon\|_2^2 < 2\gamma^2 \|p^*\|_1 + \gamma^4 \|\mathbf{1}\|_2^2$. Let $\Delta = 2\|p^*\|_1 + \gamma^2 \|\mathbf{1}\|_2^2$, we have

$$\begin{aligned}
& E[\Delta V_t(p)|p] \\
\leq \quad & E[V(p(t) - \gamma g(p(t))) - V(p(t)) \mid p(t) = p] + \gamma^2 \Delta \\
= \quad & E[-\gamma g(p(t))^T (2(p(t) - p^*) - \gamma g(p(t))) \mid p(t) = p] + \gamma^2 \Delta \\
= \quad & 2\gamma \bar{g}(p)^T (p^* - p) + \gamma^2 E[\|g(p(t))\|_2^2 \mid p(t) = p] + \gamma^2 \Delta \\
\leq \quad & 2\gamma \bar{g}(p)^T (p^* - p) + \gamma^2 (G^2 + \Delta),
\end{aligned}$$

where we again use the assumption that the norm of $g(p(t))$ is bounded above by G. By the definition of subgradient, we further get

$$E[\Delta V_t(p)|p] \quad \leq \quad 2\gamma(\overline{D}(p^*) - \overline{D}(p)) + \gamma^2 (G^2 + \Delta).$$

Let

$$\delta = \max_{\overline{D}(p)-\overline{D}(p^*)\le\gamma(G^2+\Delta)} \|p-p^*\|_2$$

and define $\mathcal{A} = \{p : \|p-p^*\|_2 \le \delta\}$. We obtain

$$E[\Delta V_t(p)|p] \le -\gamma^2(G^2+\Delta)\mathcal{I}_{p\in\mathcal{A}^c} + \gamma^2(G^2+\Delta)\mathcal{I}_{p\in\mathcal{A}},$$

where $\mathcal{I}$ is the index function. Thus, by Theorem 3.1 in [38], which is an extension of Foster's criterion [1], the Markov chain $p(t)$ is stable. □

The above proof shows that the distance to the optimal p^* has negative conditional mean drift for all prices that have sufficiently large distance to p^*, and implies that the congestion price will stay near p^* when γ is small enough.

5.3. Performance Evaluation

We now characterize the performance of the algorithm (20)–(22) in terms of the dual and primal objective functions of the reference system problem (17)–(19).

Theorem 5. *The algorithm (20)–(22) converges statistically to within a small neighborhood of the optimal value $\overline{D}(p^*)$, i.e.,*

$$\overline{D}(p^*) \le \overline{D}(E[p(\infty)]) \le \overline{D}(p^*) + \frac{\gamma(G^2+\Delta)}{2}, \quad (23)$$

where $p(\infty)$ denotes the state of the Markov chain $p(t)$ in the steady state, and $\Delta = 2\|p^*\|_1 + \gamma^2\|\mathbf{1}\|_2^2$.

Proof. The first inequality $\overline{D}(p^*) \le \overline{D}(E[p(\infty)])$ always holds, since $\overline{D}(p^*)$ is the minimum of the dual function $\overline{D}(p)$. Now we prove the second inequality. From the proof of Theorem 4, we have

$$\begin{aligned} E[\Delta V_t(p)|p] &= E[V(p(t+1)) - V(p(t)) \mid p(t) = p] \\ &\le 2\gamma(\overline{D}(p^*) - \overline{D}(p)) + \gamma^2(G^2+\Delta). \end{aligned}$$

Taking expectation over p, we get

$$\begin{aligned} E[\Delta V_t(p)] &= E[V(p(t+1)) - V(p(t))] \\ &\le 2\gamma(\overline{D}(p^*) - E[\overline{D}(p)]) + \gamma^2(G^2+\Delta). \end{aligned}$$

Taking summation from $\tau = 0$ to $\tau = t-1$, we obtain

$$\begin{aligned} E[V(p(t))] \le\ & E[V(p(0))] - 2\gamma\sum_{\tau=0}^{t-1} E[\overline{D}(p(\tau))] \\ & + 2\gamma t\overline{D}(p^*) + t\gamma^2(G^2+\Delta). \end{aligned}$$

Since $E[V(p(t))] \ge 0$, we have

$$2\gamma\sum_{\tau=0}^{t-1} E[\overline{D}(p(\tau))] - 2\gamma t\overline{D}(p^*) \le E[V(p(0))] + t\gamma^2(G^2+\Delta).$$

From this inequality we obtain

$$\frac{1}{t}\sum_{\tau=0}^{t-1} E[\overline{D}(p(\tau))] - \overline{D}(p^*) \le \frac{E[V(p(0))] + t\gamma^2(G^2+\Delta)}{2t\gamma}.$$

Note that $p(t)$ is stationary and ergodic in some steady state by Theorem 4, and so is $\overline{D}(p(t))$. Thus,

$$\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1} E[\overline{D}(p(\tau))] = E[\overline{D}(p(\infty))].$$

So,

$$E[\overline{D}(p(\infty))] - \overline{D}(p^*) \le \frac{\gamma(G^2+\Delta)}{2}.$$

Since $\overline{D}(p)$ is a convex function, by Jensen's inequality,

$$\overline{D}(E[p(\infty)]) - \overline{D}(p^*) \le \frac{\gamma(G^2+\Delta)}{2},$$

i.e., the algorithm converges statistically to within $\gamma(G^2+\Delta)/2$ of the optimal value $\overline{D}(p^*)$. □

Since $\overline{D}(p)$ is a continuous function, Theorem 5 implies that the congestion price p approaches p^* statistically when γ is small enough.

Corollary 6. *$x(t)$ is a stable Markov chain. Moreover, the average arrival rates $E[x(\infty)] \in \overline{\Pi}$, where $x(\infty)$ denotes the state of the process $x(t)$ in the steady state.*

Proof. $x(t)$ is a deterministic, finite-value function of $p(t)$. $x(t)$ is a stable Markov chain, since $p(t)$ is. $E[x(\infty)] \in \overline{\Pi}$, otherwise the average congetsion price $E[p(\infty)]$ will go unbounded, which contradicts Theorem 4. □

Theorem 7. *Let $\overline{P}(x)$ be the primal function and x^* be the optimal source rates of the reference system problem (17)–(19). The algorithm (20)–(22) converges statistically to within a small neighborhood of the optimal value $\overline{P}(x^*)$, i.e.,*

$$\overline{P}(x^*) \ge \overline{P}(E[x(\infty)]) \ge \overline{P}(x^*) - \frac{\gamma(G^2+\Delta)}{2}. \quad (24)$$

Proof. The proof for the theorem is a straightforward extension of the proof of Theorem 3, following similar procedure as in the proof of Theorem 5. We skip the detail here. □

Since $\overline{P}(x)$ is a continuous function and has a unique optimal, Theorem 7 implies that the average source rate approaches the optimal of the ideal reference system (17)–(19) when stepsize γ is small enough. Theorems 5 and 7 show that, surprisingly, the algorithm (20)–(22) can be seen as a distributed algorithm to approximately solve the ideal reference system problem that is not readily solvable due to stochastic channel variations.

Our proofs for stability and performance bounds are rather general. They only use general properties of convexity and Markovity and the definition of subgradients. We thus have presented a general technique and results regarding the stability and optimality of dual algorithm for convex optimization in face of time-varying parameters. As the flow contention graph is a rather general construct and can be used to capture the interdependence or contention among parallel servers of any queueing networks, the aforementioned results are applicable to any systems that can be modelled by a general model of queueing network that is served by a set of interdependent parallel servers with time-varying service capabilities. In the next two sections, we will discuss two such applications. Other examples include fair scheduling in a generalized switch [34], and TCP [22] with time-varying capacity as in last-hop wireless networks. It can include power control as well [5], as power does not change convexity of the feasible rate region.

As specific cases of the general model and algorithm presented in sections 4 and 5, we will discuss joint congestion control and medium access control design in multihop wireless networks with single-path routing and cross-layer congestion control, routing and scheduling design in the network without prespecified paths in the next two sections, respectively.

6. Joint Congestion Control and Media Access Control Design

TCP was originally designed for wireline networks, where links are assumed to have fixed capacities. However, as wireless channel is a shared medium and interference-limited, wireless links are "elastic" and the capacities they obtain depend on the bandwidth sharing mechanism used at the link layer. This may result in various TCP performance problems in wireless networks.

One such problem is TCP unfairness over multihop wireless networks. Many existing wireless MAC protocols, such as DCF specified in IEEE 802.11 standard [13], are traffic independent and do not consider the actual requirements of the flows competing for the channel. These MAC protocols suffer from the unfairness problem, caused by the location dependency of the contentions, and exacerbated by the contention resolution mechanisms such as the binary exponential backoff algorithm adopted in DCF. When they interact with TCP, TCP will further penalize these flows with more contention. This will result in significant TCP unfairness in multihop wireless networks [10] [37] [45] [46] [47]. To illustrate this, consider the example in Figure 2, and assume there are four network-layer flows $A \to B$, $C \to D$, $E \to F$ and $G \to H$. The flow $C \to D$ experiences more contention and will build up a queue faster than the other three flows. TCP will further penalize it by reducing the congestion window more aggressively, and the resulting throughput of flow $C \to D$ will be much less than that of the other flows.

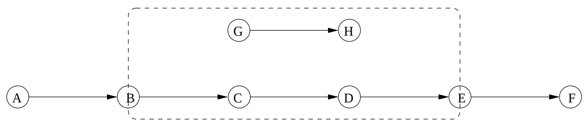

Figure 2: An example of a simple multihop wireless network.

In addition to the location dependency of contentions, correlation among links is also the key to understand the interaction between transport and MAC layers. In wireline networks, link bandwidth is well defined and links are disjoint resources. But in wireless networks, as we mentioned above, links are correlated due to interference, and network-layer flows that do not transverse a common link may still compete with each other. Thus, congestion is located at some spatial contention region [47]. Consider again the example in Figure 2, and assume there are two network-layer flows $A \to F$ and $G \to H$. Link-layer flows BC, CD, DE and GH contend with each other, and congestion is located in the spatial contention region denoted by the rectangle. So, unlike wireline networks where link capacities provide constraints for resource allocation, in multihop wireless networks the contention relations between link-layer flows provide fundamental constraints for resource allocation. We need to exploit the interaction between transport and link (MAC) layers, in order to improve the performance.

The equations (2)–(3) capture the constraints that arise from channel contention among wireless links. We model the resource allocation for multihop wireless networks as a utility maximization problem with these constraints,

$$\max_{x,f} \quad \sum_s U_s(x_s) \tag{25}$$

$$\text{subject to} \quad Rx \le f, \tag{26}$$

$$f \in \Pi, \tag{27}$$

which is a special case of the system problem (7)–(9). With this formulation, we can explicitly exploit the interaction between transport and MAC layers, and systematically carry out joint design of congestion and media access control. In the next subsection, a dual algorithm solving the system problem (25)–(27) is derived by applying the algorithm (10)–(13). The algorithm motivates a scheme for media access control in which link-layer flows are scheduled according to congestion prices.

6.1. Distributed Algorithm

Consider the Lagrangian of the problem (25)–(27) with respect to the rate constraint

$$L(p, x, f) = \sum_s U_s(x_s) - p^T (Rx - f). \tag{28}$$

Interpret p_l as the congestion price at link l, we can use the algorithm (10)–(13) to solve the problem (25)–(27) and its dual.

Rate control: At time t, given congestion price $p(t)$, source s adjusts its sending rate x_s according to the aggregate congestion price $\sum_l R_{ls} p_l$ along its path

$$x_s(t) = U_s'^{-1}\Big(\sum_l R_{ls} p_l(t)\Big). \tag{29}$$

Scheduling: Over link l, send an amount of data for each flow according to the rate f such that

$$f(t) = f(p(t)) \in \arg\max_{f \in \Pi} p^T f. \tag{30}$$

If the network with time-varying channel is considered, each node monitors channel state $h(t)$ and over link l sends an amount of data for each flow according to the rate f such that

$$f(t) = f(p(t)) \in \arg\max_{f \in \Pi(h(t))} p^T f. \tag{31}$$

Congestion price update: Each link l updates its price, according to

$$p_l(t+1) = \Big[p_l(t) + \gamma_t\Big(\sum_s Rls x_s(p(t)) - f_l(p(t))\Big)\Big]^+. \tag{32}$$

The above algorithm motivates a joint design scheme where the link layer flows are scheduled according to congestion prices of the links. Also, note that equations (29) and (32) are completely distributed and can be implemented at individual sources and links using only local information. We will discuss the distributed solution to scheduling problem (30) in the next subsection.

6.2. Scheduling over multihop Networks

We now come to the scheduling problem (30), which will also show out in the next section. Scheduling over multihop networks is a difficult problem and in general NP-hard. To see this, note that problem (30) is equivalent to a maximum weight independent set problem over the flow contention graph, which is NP-hard for general graphs. It is easy to design some heuristic algorithm but is hard to bound its performance.

With the primary interference model, the scheduling problem (30) is equivalent to the maximum weighted matching problem[3] over the connectivity graph $\{N, L\}$ of the network. Maximum weighted matching problem can be computed in polynomial time (see, e.g., [29]), but this requires centralized implementation. If implemented over a multihop network, each node needs to notify the central node of its weight and local connectivity information such that the central node can reconstruct the network topology as a weighted graph. This will lead to an immense communication overhead which is expensive in time and resources. There also exist simpler greedy sequential algorithms to compute a weighted matching at most a factor of 2 away from the maximum; see, e.g., [30]. But they also require centralized implementation. We seek a distributed algorithm where each node participates in the computation itself using only local information.

A few distributed approximation algorithms exist for maximum weighted matching problem; see, e.g., [40] [42] [12]. In [12], the author presents a simple distributed algorithm to compute a weighted matching at most a factor of 2 away from the maximum in linear running time $O(|L|)$. This algorithm is a distributed variant of the sequential greedy algorithm presented in [30]. We have utilized this algorithm to solve the scheduling problem (30) distributedly, see [4] for details. The resulting scheduling algorithm for multihop wireless networks is one of the best distributed algorithms in terms of computational complexity and performance bound. It has a linear complexity $O(|L|)$. Such a low complexity is important for the scalability and efficiency of multihop wireless networks. It achieves a performance of $1/2$ of the maximum weight in the worst case, and in practice, numerical simulations show it typically achieves a performance within about $4/5$ of the maximum weight. There also exist few other distributed approximation algorithms; see, e.g., [21] [43] [24] [14] [32] [28]. Especially, in [14], [32] and [28] the authors present distributed random access algorithms that achieve nearly 100% throughput.

[3] A matching in a graph is a subset of links, no two of which share a common node. The weight of a matching is the total weight of all its links. A maximum weighted matching in a graph is a matching whose weight is maximized over all matchings of the graph.

As for the overall performance of our cross-layer design with approximate scheduling, we can extend the result in [21] to show that the performance is no worse than that achieved by an exact design with a feasible rate region $\frac{1}{2}\Pi$ at the link layer. Moreover, in [4] we also see that this distributed scheduling algorithm only results in a very small degradation in the performance of the cross-layer design for the network with time-varying channel, since in this situation the exact solution of the scheduling is not as important and reasonable approximations work well.

6.3. Numerical Examples

In this subsection, we provide numerical examples to complement the analysis in previous subsections. We consider a simple network with secondary interference as shown in Figure 2, and assume that there are three network layer flows $G \rightarrow H$, $A \rightarrow F$ and $D \rightarrow F$ with the same utility function $U_s(x_s) = \log x_s$. We have chosen a simple topology to facilitate discussion.

The Network with Fixed Channel and Single-Rate Devices

We first consider a network with fixed link capacities. For simplicity, we assume that all the links have one unit of capacity when active. Figure 3 shows the evolution of source rates and their averages with the joint algorithm (29), (30) and (32) with stepsize $\gamma = 0.2$. We see that the source rates converge quickly to a neighborhood of the optimal and oscillate around the optimal. This oscillating behavior mathematically results from the non-differentiability of the dual function and physically can be interpreted as due to the scheduling process. However, the average source rates are smooth and approach the optimum monotonically. Figure 4 shows the evolution of the corresponding end-to-end congestion prices and the averages of the three flows. Similarly, the congestion prices approach the optimum quickly. We also note that the performance of the algorithm is much better than the bound of $\gamma G^2/2$ specified in Theorems 2 and 3.

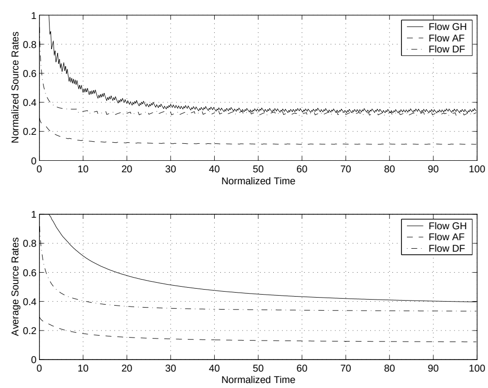

Figure 3: The evolution of source rates in the network with fixed link capacities.

The choice of the stepsize γ is important. It characterizes the "optimality" of the algorithm, as shown in Theorems 2 and 3 (and also in Theorems 5 and 7). It also affects the convergence speed. In oder to study the impact of different choices of the

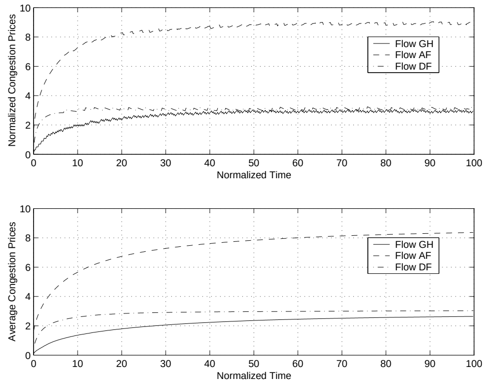

Figure 4: The evolution of congestion prices in the network with fixed link capacities.

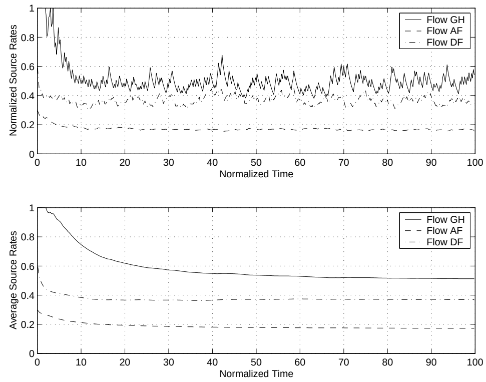

Figure 5: The evolution of source rates in the network with time-varying link capacities.

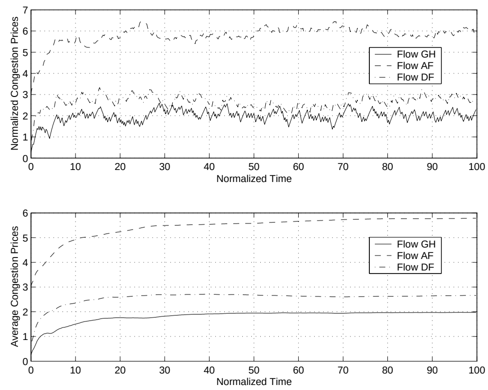

Figure 6: The evolution of congestion prices in the network with time-varying link capacities.

stepsize on the performance of the algorithm, we have run simulations with different stepsizes. We found that the smaller the stepsize, the slower the convergence and the closer to the optimal, which is a general characteristic of any gradient based algorithm. So, there is a tradeoff between convergence speed and optimality. In practice, the end user can first choose large stepsizes to ensure fast convergence, and subsequently, the stepsizes can be reduced once the source rate starts oscillating around some mean value.

The Network with Time-Varying Channel and Multirate Devices

We now consider a network with time-varying link capacities. For simplicity, we assume that the capacities of all links are identically, uniformly distributed over 0.5, 1 and 1.5 units. Thus, the average capacity for each link when active is the same as that in the example with fixed link capacities.

Figures 5 and 6 show the evolution of source rates, congestion prices and their averages with the same stepsize $\gamma = 0.2$. The source rates and congestion prices have much larger variations than those with fixed channel, due to the channel variations. But the average source rates and congestion prices are still smooth, and converge quickly and monotonically to optimal values. Our simulation results have confirmed the conclusions from Theorems 5 and 7, which say that the average source rates and congestion prices approach the optimum of an ideal system with the best feasible rate region at the link layer, and that algorithm (29), (31) and (32) can been seen as a distributed algorithm to solve this ideal system problem. Also note that, although the average link capacities when active are the same as those in fixed channel, each flow achieves larger sending rate. This is due to multi-user diversity: our "optimal" scheduling (31) has implicitly considered multi-user diversity.

6.4. Summary

We have presented a model for the joint design of congestion control and media access control for multihop wireless networks, where the resulting dual algorithm is to solve a utility maximization problem with constraints that arise from contention for the wireless channel. This algorithm motivates a joint design where link-layer flows are scheduled according to the congestion prices of the links.

There exist other ways to solve the resource allocation problem (25)–(27). In [3], we also derive a primal algorithm by solving the relaxation of the system problem (25)–(27). Based on the algorithm, we propose a traffic-dependent scheme for contention-based medium access control and generate congestion price directly from the MAC layer. As scheduling in multihop wireless networks is an intrinsically hard problem, contention-based medium access seems a must. To further integrate congestion control and contention-based medium access in utility maximization framework will be a future research step.

7. Joint Congestion Control, Routing and Scheduling Design

In the last section, we have discussed the resource allocation in multihop wireless networks where the path for each network layer flow is given. However, as wireless spectrum is a scarce resource, it may be costly to maintain end-to-end paths, and congestion control based on end-to-end feedback may consume too much bandwidth in signalling. Moreover, most routing schemes for multihop networks select paths that minimize hop count; see, e.g., [15] [31]. This implicitly predefines a path for any source-destination pair, independent of the pattern of traffic demand and interference/contention among links. This may result in congestion at some region while other regions are underutilized. In order to achieve high end-to-end throughput and efficient resource allocation, the paths should not be decided exogenously but jointly optimized with congestion control and scheduling.

Since the actual paths that will be used are not specified *a priori*, we will use multicommodity flow model for routing and model the resource allocation as a utility maximization problem with the constraints (2) and (5),

$$\max_{x,f} \quad \sum_s U_s(x_s) \tag{33}$$

$$\text{subject to} \quad x_i^k \leq \sum_{j:(i,j)\in L} f_{i,j}^k - \sum_{j:(j,i)\in L} f_{j,i}^k, \tag{34}$$

$$f \in \Pi, \tag{35}$$

where $i \in N,\ k \in D, i \neq k$, and $x_i^k = 0$ if $[i,k] \notin S \times D$. Again, this problem is a special case of the system problem (7)–(9). In the next subsection, we apply the algorithm (10)–(13) to obtain a distributed subgradient algorithm for joint congestion control, routing and scheduling. This algorithm motivates a joint design where the source adjusts its sending rate according to the congestion price generated locally at the source node, and back-pressure from the differential price of neighboring nodes is used for optimal scheduling and routing.

7.1. Distributed Algorithm

Consider the Lagrangian of the problem (33)–(35) with respect to the rate constraint

$$L(p,x,f) = \sum_s U_s(x_s) - \sum_{i\in N, k\in D, i\neq k} p_i^k (x_i^k - \sum_{j:(i,j)\in L} f_{i,j}^k + \sum_{j:(j,i)\in L} f_{j,i}^k). \tag{36}$$

Interpret p_l^k as the congestion price at node i for the flows to destination k, we can use the algorithm (10)–(13) to solve the problem (33)–(35) and its dual.

Rate control: At time t, given congestion price $p(t)$, the source s adjusts its sending rate x_s according to the local congestion price at the source node

$$x_s(p) = U_s'^{-1}(p_s), \tag{37}$$

where $p_s = p_k^i$ for $s = [i,k] \in S \times D$. In contrast to traditional TCP congestion control where the source adjusts its sending rate according to the aggregate price along its path, in this algorithm the congestion price is generated locally at the source node.

Note that, since

$$\sum_{i,k} p_i^k \left(\sum_j f_{i,j}^k - \sum_j f_{j,i}^k \right) = \sum_{i,j,k} f_{i,j}^k \left(p_i^k - p_j^k \right),$$

the scheduling problem is equivalent to the following problem

$$\max_{f\in\Pi} \sum_{i,j} f_{i,j} \max_k \left(p_i^k - p_j^k \right). \tag{38}$$

This motivates the following joint scheduling and routing algorithm:

Scheduling: Each node i collects congestion price information from its neighbor j, finds destination $k(t)$ such that $k(t) \in \arg\max_k (p_i^k(t) - p_j^k(t))$, and calculates differential price $w_{i,j}(t) = p_i^{k(t)}(t) - p_j^{k(t)}(t)$ and passes this information to its neighbors. Allocate capacities $\widetilde{f_{i,j}}(t)$ over links (i,j) such that

$$\widetilde{f}(t) \in \arg\max_{f\in\Pi} \sum_{(i,j)\in L} w_{i,j}(t) f_{i,j}. \tag{39}$$

If the network with time-varying channel is considered, each node monitors the channel state $h(t)$ and allocates capacities $\widetilde{f_{i,j}}(t)$ over links (i,j) such that

$$\widetilde{f}(t) \in \arg\max_{f\in\Pi(h(t))} \sum_{(i,j)\in L} w_{i,j}(t) f_{i,j}. \tag{40}$$

Routing: Over link (i,j), send a number of bits for destination $k(t)$ according to the rate determined by the scheduling.

The $w_{i,j}$ values represent the maximum differential congestion price of destination k flows between nodes i and j. The above algorithm uses backpressure to do optimal scheduling and find optimal routing. Also note that the scheduling problem is solved by the following assignment,

$$f_{i,j}^k(t) = \begin{cases} \widetilde{f_{i,j}}(t) & \text{if } k = k(t), \\ 0 & \text{if } k \neq k(t). \end{cases}$$

Congestion price update: Each node i updates its price with respect to destination k, according to

$$p_i^k(t+1) = [p_i^k(t) + \gamma_t(\, x_i^k(p(t)) - (\sum_{j:(i,j)\in L} f_{i,j}^k(p(t)) - \sum_{j:(j,i)\in L} f_{j,i}^k(p(t))\,)\,)]^+, \tag{41}$$

and passes the price p_i^k to its neighbors. Note that $p_i^k(t)$ is interpreted as congestion price at the beginning of times lot t.

The above dual algorithm motivates a joint congestion control, routing and scheduling design where at the transport layer sources s individually adjust their rates according to the local congestion price at the source nodes, and nodes i individually update their prices according to (41), and at the network/link layer nodes i solve the scheduling (39) and route data flows accordingly. Also, note that the congestion control is not an end-to-end scheme. There is no need to maintain end-to-end paths and no communication overhead for congestion control.

7.2. Numerical Examples

In this subsection, we provide numerical examples to complement the analysis in the previous subsections. We consider a simple multihop network with primary interference as shown in Figure 7, and assume that there are two network layer flows $A \rightarrow F$ and $B \rightarrow E$ with the same utility $U_s(x_s) = \log x_s$.

The Network with Fixed Channel and Single-Rate Devices

We consider first the network with fixed link capacities. For simplicity, we assume that links CE, EC, BF and FB have one unit of capacity and all other links have 2 units of capacity when active. Figure 8 shows the evolution of source rate and congestion price of each flow with the joint algorithm (37), (39) and

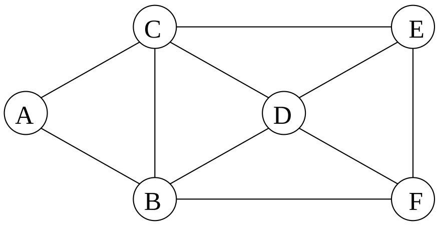

Figure 7: A simple network with two network layer flows. All links are bidirectional.

(41) with stepsize $\gamma = 0.2$. We see that they converge quickly to a neighborhood of the optimal and oscillate around the optimal. However, Figure 9 shows that the average source rates and congestion prices are smooth and approach the optimum monotonically. We again note that the performance of the algorithm is much better than the bound of $\gamma G^2/2$ specified in Theorems 2 and 3.

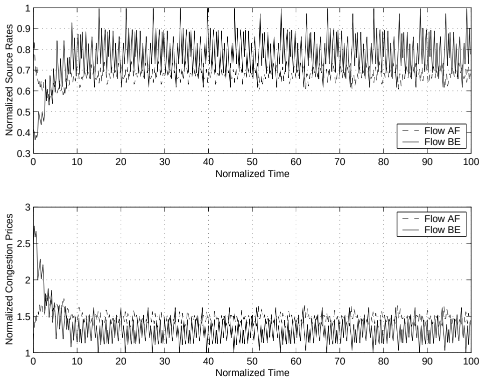

Figure 8: Source rates and congestion prices in the network with fixed link capacities.

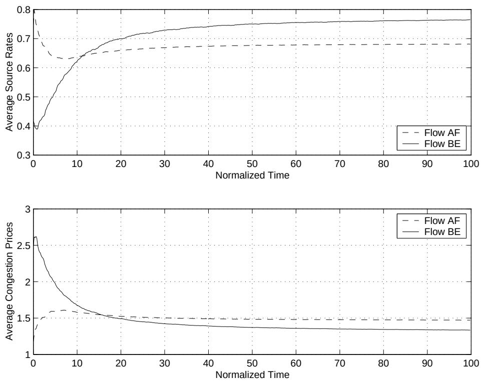

Figure 9: The average source rates and congestion prices in the network with fixed link capacities.

The Network with Time-Varying Channel and Multirate Devices

We now consider the network with time-varying link capacities. For simplicity, we assume that links CE, EC, BF and FB's capacities are identically, uniformly distributed over 0.5, 1 and 1.5 units, while other links' capacities are identically, uniformly distributed over 1, 2 and 3 units. Thus, the average capacity for each link when active is the same as that in the example with fixed link capacities.

Figures 10 and 11 show the evolution of source rates, congestion prices and their averages with the same stepsize $\gamma = 0.2$. The source rates and congestion prices have much larger variations than those with fixed channel, due to the channel variations. But the average source rates and congestion prices are still smooth, and converge quickly and monotonically to optimal values. Note that, although the average link capacity when active is the same as that in fixed channel, each flow achieves larger sending rates. This is again due to multi-user diversity that we exploit when doing scheduling. Also note that the increase in sending rate of flow BE is much more notable. This is because node B has more neighbors and thus a much larger multi-user diversity.

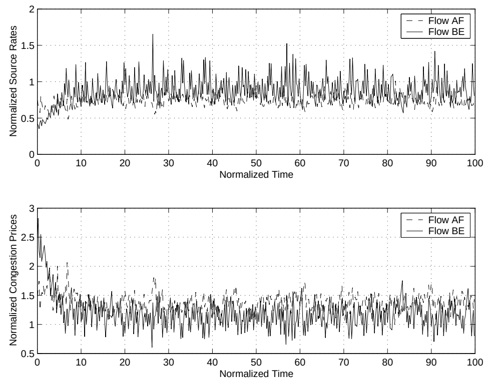

Figure 10: Source rates and congestion prices in the network with time-varying link capacities.

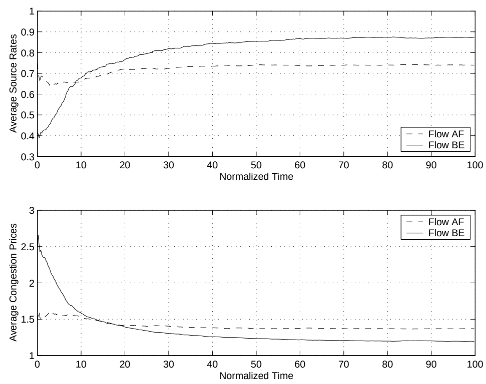

Figure 11: The average source rates and congestion prices in the network with time-varying link capacities.

7.3. Summary

We have presented a model for the joint design of congestion control, routing and scheduling for multihop wireless networks. The resulting dual algorithm motivates a joint design where at the transport layer, sources s adjust their rates according to the local congestion price at the source nodes, and at the network/link layer nodes solve the scheduling and route data flows according to backpressure in congestion between neighboring nodes. As our design only requires nodes exchanging local information with their neighbors and does not need to maintain end-to-end paths, it has a very low communication overhead and can adapt to changing topologies such as those in mobile multihop networks.

8. Conclusions

We have seen in this paper that, by formulating a general utility maximization problem for the network design, duality theory leads to a natural "vertical" decomposition into functional modules of various layers of the protocol stack and "horizontal" decomposition into distributed computation across various network nodes or links. As shown in Figure 12, our current theory integrates three functions—congestion control, routing and scheduling—in transport, network and link layers into a coherent framework. With this layering scheme, the dual variables of the utility maximization problem capture the network state information and are the information that is passed across the interfaces among different layers. These layers are interacting through and coordinated by the dual variables, i.e., congestion prices, so as to achieve global optimality. Even though this framework does not provide all the design and implementation details (such as the implementation of congestion prices and signalling mechanism), it helps us understand issues, clarify ideas, and suggests directions, leading to better and more robust designs for multihop wireless networks.

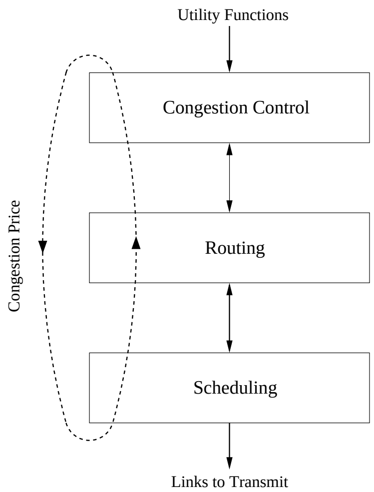

Figure 12: Layering as dual decomposition.

This framework—layering as dual decomposition in particular and layering as optimization decomposition in general—is promising to be extended to provide a mathematical theory for network architecture, and to allow us to systematically derive the layering structure of the various mechanisms of different protocol layers, their interfaces, and the control information that must cross these interfaces to achieve a certain performance and robustness. In this general framework, application needs (possibly, plus other performance metrics such as network cost) form the objective function (i.e., network utility to be maximized) and the restrictions in resource provisioning are translated into the constraints of the generalized network utility maximization problem. By choosing different objective functions and having different sets of decision variables involved, we can explicitly characterize and trade off different design objectives such as performance, scalability and robustness.

There exist, however, some challenging issues with this framework. First, utility design, i.e., how to model the user or application needs, is not an easy task, especially for real-time applications. Second, the general utility maximization problems may be nonlinear, nonconvex optimization with integer constraints. Third, this framework only involves the functionalities of the data plane of the network, but leaves out the issues related to the control plane such as implementation and management complexity. To address these issues will be a future research step.

Acknowledgments

The authors would like to thank Mung Chiang for collaborations on the topic. This work is partially supported by Army Institute for Collaborative Biotechnologies, AFOSR award FA9550-05-1-0032, ARO grant DAAD19-02-1-0283, NSF grant CNS-0435520, NSF NetSE grant CNS-0911041, ARO MURI grant W911NF-08-1-0233, ONR MURI grant N000140810747, and the Caltech Lee Center for Advanced Networking.

References

[1] S. Asmussen, *Applied Probability and Queues*, second edition, Springer, 2003.

[2] D. Bertsekas, *Nonlinear Programming*, 2nd edition, Athena Scientific, 1999.

[3] L. Chen, S. Low and J. Doyle, Joint congestion control and media access control design for ad hoc wireless networks, *Proceedings of IEEE Conference on Computer Communications (Infocom)*, March 2005.

[4] L. Chen, S. Low, M. Chiang and J. Doyle, Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks, *Proceedings of IEEE Conference on Computer Communications (Infocom)*, April 2006.

[5] M. Chiang, Balancing transport and physical layers in wirless multihop networks: Jointly optimal congestion control and power control, *IEEE Journal on Selected Areas in Communications*, 23(1):104-116, January 2005.

[6] M. Chiang, S. H. Low, R. A. Calderbank, and J. C. Doyle, Layering as optimization decomposition, *Proceedings of the IEEE*, 2007.

[7] R. Diestel, *Graph Theory*, Springer-Verlag, 1997.

[8] A. Eryilmaz and R. Srikant, Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control, *Proceedings of IEEE Conference on Computer Communications (Infocom)*, March 2005.

[9] Z. Fang and B. Bensaou, Fair bandwidth sharing algorithms based on game theory frameworks for wireless ad hoc networks, *Proceedings of IEEE Conference on Computer Communications (Infocom)*, March 2004.

[10] M. Gerla, R. Bagrodia, L. Zhang, K. Tang and L. Wang, TCP over wireless multihop protocols: Simulation and experiments, *Proceedings of IEEE International Conference on Communications (ICC)*, June 1999.

[11] B. Hajeck and G. Sasaki, Link scheduling in polynomial time, *IEEE Transactions on Information Theory*, 34(5):901-917, September 1988.

[12] J. Hoepman, Simple distribute weighted matchings, preprint, October 2004. Available at http://arxiv.org /abs/cs/0410047.

[13] IEEE, Wireless LAN media access control (MAC) and physical layer (PHY) specifications, IEEE Standard 802.11, June 1999.

[14] L. Jiang and J. Walrand, A distributed CSMA algorithm for throughput and utility maximization in wireless networks, *Allerton Conference on Communication, Control, and Computing*, 2008.

[15] D. B. Johnson and D. A. Maltz, Dynamic source routing in ad-hoc wireless networks, *Mobile Computing*, T. Imielinski and H. Korth (eds.), Kluwer Academic Publishers, 1996.

[16] F. P. Kelly, A. K. Maulloo and D. K. H. Tan, Rate control for communication networks: Shadow prices, proportional fairness and stability, *Journal of Operations Research Society*, 49(3):237-252, March 1998.

[17] M. Kodialam and T. Nandagopal, Charaterizing achievable rates in multihop wireless networks: The joint routing and scheduling problem, *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)*, September 2003.

[18] S. Kunniyur and R. Srikant, End-to-end congestion control schemes: Utility functions, random losses and ECN marks, *IEEE/ACM Transactions on Networking*, 11(5):689-702, October 2003.

[19] H. J. Kushner and P. A. Whiting, Convergence of proportional-fair sharing algorithms under general conditions, *IEEE/ACM Transactions on Wireless Communications*, 3(4):1250-1259, July 2004.

[20] X. Lin and N. Shroff, Joint rate control and scheduling in multihop wireless networks, *Proceedings of IEEE Conference on Decision and Control (CDC)*, December 2004.

[21] X. Lin and N. Shroff, The impact of imperfect scheduling on cross-layer rate control in multihop wireless networks, *Proceedings of IEEE Conference on Computer Communications (Infocom)*, March 2005.

[22] S. H. Low and D. E. Lapsley, Optimal flow control, I: Basic algorithm and convergence, *IEEE/ACM Transactions on Networking*, 7(6):861-874, December 1999.

[23] S. H. Low, A duality model of TCP and queue management algorithms, *IEEE/ACM Transactions on Networking*, 11(4):525-536, August 2003.

[24] E. Modiano, D. Shah and G. Zussman, Maximizing throughput in wireless networks via gossiping, *Proceedings of ACM International Conference on Measurement and Modeling of Computer Systems (Sigmetrics)*, June 2006.

[25] T. Nandagopal, T. E. Kim, X. Gao and V. Bharghhavan, Achieving MAC layer fairness in wireless packet networks, *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)*, August 2000.

[26] M. Neely, E. Modiano and C. Rohrs, Dynamic power allocation and routing for time varying wireless networks, *Proceedings of IEEE Conference on Computer Communications (Infocom)*, April 2003. Journal version, *IEEE Journal on Selected Areas in Communications*, 23(1):89-103, January 2005.

[27] M. Neely, E. Modiano and C. P. Li, Fairness and optimal stochastic control for heterogeneous netwroks, *Proceedings of IEEE Conference on Computer Communications (Infocom)*, March 2005.

[28] J. Ni, B. Tan and R. Srikant, Q-CSMA: Queue length-based CSMA/CA algorithms for achieving maximum throughput and low delay in wireless networks, *Proc. IEEE INFOCOM Mini-Conference*, 2010.

[29] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithm and Complexity*, Dover Publications, 1998.

[30] R. Preis, Linear time 1/2-approximation algorithm for maximum weighted matching in general graphs, *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, March 1999.

[31] C. E. Perkins and E. M. Royer, Ad-hoc on-demand distance vector routing, *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, Febuary 1999.

[32] J. Shin, D. Shah and S. Rajagopalan, Network adiabatic theorem: an efficient randomized protocol for contention resolution, *Proceedings of ACM SIGMETRICS*, 2009.

[33] N. Z. Shor, *Minimization Methods for Non-Differentiable Functions*, Springer-Verlag, 1985.

[34] A. L. Stolyar, MaxWeight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic, *Annuals of Applied Probability*, 14(1):1-53, August 2004.

[35] A. L. Stolyar, On the asymptotic optimality of the gradient scheduling algorithm for multiuser throughput allocation, *Operation Research*, 53(1):12-25, January-Febuary 2005.

[36] A. L. Stolyar, Maximizing queueing network utility subject to statbility: Greedy primal-dual algorithm, *Queueing Systems*, 50(4):401-457, August 2005.

[37] K. Tang and M. Gerla, Fair sharing of MAC under TCP in wireless ad hoc networks, *Proceedings of IEEE Workshop on Multiaccess, Mobility and Teletraffic for Wireless Communications (MMT)*, October 1999.

[38] L. Tassiulas and A. Ephremides, Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks, *IEEE Transactions on Automatic Control*, 37(12):1936-1948, December 1992.

[39] D. N. Tse, Multiuser Diversity in Wireless Networks, http://www.eecs.berkly.edu/dtse/stanford416.ps, April 2001.

[40] R. Uehara and Z. Chen, Parallel approximation algorithms for maximum weighted matching in general graphs, *Information Processing Letters*, 76(1-2):13-17, November 2000.

[41] X. Wang and K. Kar, Cross-layer rate control for end-to-end proportional fairness in wireless networks with random access, *Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, May 2005.

[42] M. Wattenhofer and R. Wattenhofer, Distributed weighted matching, *Proceedings of the 18th Annual Conference on Distributed Computing (DISC)*, October 2004.

[43] X. Wu and R. Srikant, Regulated maximal matching: A distributed scheduling algorithm for multi-hop wireless networks with node-exclusive spectrum sharing, *Proceedings of IEEE Conference on Decision and Control (CDC)*, December 2005.

[44] L. Xiao, M. Johnasson and S. Boyd, Simultaneous routing and resource allocation for wireless networks, *Proceedings of IEEE Conference on Decision and Control (CDC)*, December 2001.

[45] S. Xu and T. Saadawi, Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?, *IEEE Communications Magazine*, 39(6):130-137, June 2001.

[46] K. Xu, S. Bae, S. Lee and M. Gerla, TCP behavior across multihop wireless networks and the wired internet, *Proceedings of ACM International Workshop on Wireless Mobile Multimedia (WoWMoM)*, September 2002.

[47] K. Xu, M. Gerla, L. Qi and Y. Shu, Enhance TCP fairness in ad hoc wireless networks using neighborhood RED, *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)*, September 2003.

[48] Y. Xue, B. Li and K. Nahrstedt, Price-based resource allocation in wireless ad hoc networks, *Proceedings of IEEE/ACM International Workshop on Quality of Service (IWQoS)*, June 2003.

[49] Y. Yi and S. Shakkottai, Hop-by-hop congestion control over a wireless multi-hop network, *Proceedings of IEEE Conference on Computer Communications (Infocom)*, March 2004.