Cross-Layer Rate Control in Wireless Networks with Lossy Links: Leaky-Pipe Flow, Effective Network Utility Maximization and Hop-by-Hop Algorithms

Qinghai Gao and Junshan Zhang
Dept. of Electrical Engineering
Arizona State University, Tempe, AZ 85287
Email: {qinghai.gao, junshan.zhang}@asu.edu

Stephen V. Hanly
ARC Special Research Centre on Ultra-Broadband Information Networks
Dept. of Electrical & Electronic Engineering
The University of Melbourne, Australia
Email: s.hanly@ee.unimelb.edu.au

***Abstract*— Due to multi-path fading and co-channel interference, wireless links are lossy in nature. As a result, the data rate of a given flow becomes "thinner and thinner" along its routing path, and the data rate received successfully at the destination node (*the effective rate*) is typically lower than the transmission rate at the source node (*the injection rate*). In light of this observation, each flow is treated as a "leaky-pipe" model in this study. Moreover, we introduce the notion of "effective utility" associated with the effective rate (not the injection rate) for each flow, and explore rate control mechanisms through effective network utility maximization (ENUM). We focus on two network models: 1) ENUM with link outage constraints with a maximum error rate at each link; 2) ENUM with path outage constraints where there exists an end-to-end outage requirement for each flow. For both problems, we explicitly take into account the "thinning" feature of data flows and devise distributed hop-by-hop rate control algorithms accordingly. Our numerical examples corroborate that higher effective network utility and better fairness among effective flow rates can be achieved by the ENUM algorithms than the standard NUM.**

I. Introduction

In the past decade, there has been tremendous research interest on decentralized rate control algorithms for wireline networks. In the seminal work [11], rate control schemes are obtained as distributed solutions to a network utility maximization (NUM) problem. Simply put, the NUM problem treats rate control as convex programming to maximize the total utility of the users, subject to resource constraints. The concavity of the objective function and the nice structure of the constraints yield distributed solutions amenable to implementation. Following [11], there has been a surge of interest in applying the NUM framework to study network congestion control mechanisms in different scenarios (see, e.g., [5], [7], [12], [15], [16], [18]).

The current congestion control schemes are primarily designed for wireline networks for which non-congestion related packet loss is very rare; and packet loss that does occur is mainly due to congestion. The standard NUM formulation does not explicitly take account of packet loss, although packet loss can be used as a signal of congestion to implement the rate control algorithms developed in the NUM framework [8]. Wireless links, on the other hand, are typically lossy and the packet loss rate is often order-higher than wireline links, due to channel impairments, fading, interference, etc.. In the present paper, we generalize the NUM approach to obtain new problem formulations, namely effective network utility maximization (ENUM), which explicitly take account of packet loss at wireless links. In particular, we develop a "leaky-pipe" flow model in which the data rate of a flow experiences "thinning" along each link of the connection.

Since congestion control in the Internet is implemented in an end-to-end manner (control is at the end points of the connections), it is typically assumed that a rate can be associated with each flow. In practice, however, bottlenecks can arise at links within wireless networks, and hence rates can vary across the links of a flow during the transient period, before equilibrium flow rates are achieved. It is clearly of critical importance to develop new models to account for this. This gives rise to the challenge of developing mathematical models that take account of this rate variation across the links of a flow. The concept of hop-by-hop congestion control in the Internet context is not new (see, e.g., [20], [23]), but one of the main contributions of this study is that we propose a hop-by-hop rate control model, and devise corresponding rate control algorithms which can explicitly take account of the thinning of data rates across the links.

One feature of the hop-by-hop algorithms we devise is that they require per flow state at the network nodes. Most transport layer protocols do not require this information so that they can be used in links with vast numbers of flows; and indeed scalability is very important for end-to-end protocols that operate across the Internet, potentially over very high bandwidth core networks. However, the focus in the present paper is on congestion control for wireless networks, where scalability is not such a concern. In fact, it is well recognized in the wireless literature that per flow state can be of great benefit in wireless channels, where adaptive rate control is often employed at the link layer, requiring per-flow channel state information [9], [22].

For a given flow, we call the transmission rate at the source node *the injection rate*, and the data rate correctly received at the destination node *the effective rate*. With the leaky-pipe flow model, it is natural to examine the utility corresponding to the effective rate (*the effective utility*) and explore effective network utility maximization. We shall study two different network models. In the first model, we consider ENUM with

This research is supported in part by Office of Naval Research through the grant N00014-05-1-0636, National Science Foundation through the grants ANI-0238550 and CNS-0721820 and the Australian Research Council through the grant DP0771385.

link outage constraints, i.e., the maximum error rate at each link should be no more than a threshold. In the second model, we consider ENUM with path outage constraints, where there exists an end-to-end outage requirement for each flow. For both problems, we explicitly take account of the thinning feature of data flows and develop hop-by-hop rate control algorithms with a back-pressure component [26]. In particular, one unique feature of these hop-by-hop algorithms is that the link price consists of both the link congestion price (which accounts for link congestion) and the link error price (which accounts for channel condition of the link), and the sum of the two prices is used for the rate update at the transmitter of each link.

The algorithms in this paper can be viewed as outcomes of cross-layer design. One can view the algorithms as jointly optimizing the link layer and the transport layer so that end-to-end objectives are met. It is well known that there is a tradeoff between link rate and outage probability in wireless channels [3]. As in [13], we use the NUM formulation (although extended to ENUM in the present case) to determine the optimal tradeoff for each flow, on each link, as determined dynamically by the higher-layer fairness objectives embodied in the ENUM problem. In contrast, link layer algorithms that are designed independently of higher layers are unable to make these adjustments, and designers are forced to make more conservative assumptions [17]. We show that considerable gains in effective network utility and fairness among effective flow rates are achieved by the cross-layer approach taken in the present paper.

The remainder of this paper is organized as follows. In Section II, we present related work on rate control. In Section III, the models for lossy links and leaky-pipe flows are described; and the standard NUM over lossy channels is presented for comparison. In Section IV, we present the ENUM problem with link outage constraints, and develop a hop-by-hop algorithm for rate control. We then address the problem of ENUM with path outage constraints in Section V. The numerical results are shown in Section VI. Finally, Section VII concludes this paper.

II. Related Work

There has been much work on end-to-end congestion control for wireless networks (see, e.g., [1], [14], [24]). Roughly speaking, these methods apply the rate control framework of wireline networks to wireless networks, assuming that congestion control is "shielded" from channel errors. A typical approach is to implement error recovery at the link layer, in an effort to make the link appear very reliable to the transport layer. It is not clear, however, how reliable the link layer should be and how the rate-reliability tradeoff should be made; and most practical systems cannot ensure the same level of reliability at the link layer as in wireline networks. A recent systematic approach to the rate-reliability tradeoff is provided in [13], and we take a similar approach in the present paper. An excellent survey on improving TCP performance over wireless networks can be found in [6].

In 1998, the seminal paper [11] set the ground for the network utility maximization approach. Recently, this approach has been applied to congestion control in wireless networks with channel errors (see, e.g., [5], [13]). In [5], the authors propose an end-to-end algorithm that adjusts the number of connections at the application layer, treating packet loss as a factor for inaccurate feedback. The NUM approach has been used in [13] to explore the rate-reliability tradeoff mentioned above. One major difference between [13] and our work is that, in [13], the data rate of a given flow is treated unchanged from hop to hop along its route. In contrast, we explicitly study the leaky-pipe flows which accurately model the rate change per hop over lossy links, and the focus here is on devising distributed hop-by-hop algorithms.

Hop-by-hop congestion control has been studied for the Internet in the 90's (see, e.g., [20], [23]). Recently, hop-by-hop mechanisms for wireless networks are studied in [25], [27], [28]. In [28], a distributed hop-by-hop algorithm is proposed where each node adds its current congestion price to that it received from a downstream node and passes this information toward the upstream node. The basic idea is that every node in the path of the session operates a congestion control algorithm. The source node ultimately receives the sum of all price information from the corresponding downstream nodes and uses the information for adapting rates. In [25], an implicit hop-by-hop congestion control scheme is proposed which enforces that the input rate for a given flow does not exceed the output rate at any intermediate node. This is accomplished by preventing the transfer of a second packet to a node until this node forwards the previous one. In [27], a hop-by-hop congestion control is proposed for sensor networks, where local packet inter-arrival time, service time, and buffer occupancy ratio are exploited to detect congestion at every intermediate sensor node.

III. System Model

A. Rate Control in Standard NUM Framework

Consider a multi-hop wireless network that consists of a set $\mathcal{L} = \{1, \cdots, L\}$ of links. The network is shared by a set $\mathcal{S} = \{1, \cdots, S\}$ of sources. The source node of flow s transmits at a data rate of $x_s \in [m_s, M_s]$, where m_s and M_s are the minimum and maximum rates, respectively. We assume that each source s emits one flow using a fixed set of links $L(s)$ for its route. For each link l, let $S(l) = \{s | l \in L(s)\}$ be the set of sources that use link l. Note that $l \in L(s)$ if and only if $s \in S(l)$.

The standard NUM framework was originally proposed for wireline networks, where it is assumed that each link has a fixed link capacity c_l, $l \in \mathcal{L}$, and the data rate of flow s does not change along its route. Each flow has a utility function $U_s(x_s)$ associated with the flow rate x_s. The utility function is often assumed to be twice-differentiable, increasing and strictly concave in its argument. A utility can be interpreted as the level of satisfaction attained by a user as a function of resource allocation, and different shapes of utility functions lead to different types of fairness. For example, a family of utility functions is defined as follows [21]:

$$U^{\alpha}(x) = \begin{cases} \log x, & \text{if } \alpha = 1, \\ (1-\alpha)^{-1} x^{1-\alpha}, & \alpha \geq 0, \alpha \neq 1. \end{cases} \tag{1}$$

Denote $I_s = [m_s, M_s]$. The standard NUM problem is formulated as follows [11]:

$$\begin{array}{ll} \max\limits_{\{x_s \in I_s\}} & \sum_s U_s\left(x_s\right) \\ s.t. & \sum_{s \in S(l)} x_s \leq c_l, \quad \forall l. \end{array} \tag{2}$$

Appealing to the Lagrangian dual method, a dual algorithm for rate control is given by [18], [19]:

- Source rate update:

$$x_s(n+1) = \left[U_s'^{-1}\left(\sum_{l \in L(s)} \lambda_l(n) \right) \right]_{I_s} \tag{3}$$

  where $U_s'^{-1}(\cdot)$ denotes the inverse of the derivative of U_s and $[\cdot]_{I_s}$ stands for the projection onto the set I_s.
- Link price update:

$$\lambda_l(n+1) = \left[\lambda_l(n) - \varepsilon(n) \left(c_l - \sum_{s \in S(l)} x_s(n) \right) \right]^+ \tag{4}$$

  where $\varepsilon(n)$ is the step size and $[z]^+ = \max\{z, 0\}$.

One underlying assumption in the NUM framework is that *the same flow rate is present at all the links along the route* (we call this type of flow "standard flow" in this paper). In wireless networks, however, *the data rate of a flow becomes "thinner and thinner" along its route*, due to the lossy nature of wireless links. Therefore, it is more sensible to examine a utility function associated with the rate correctly received at the destination node. To study this, we first present the models for lossy links and leaky-pipe flows in the following.

B. Models for Lossy Links and Leaky-Pipe Flows

1) Lossy link model: We assume that there is receiver channel state information (CSI) and no transmitter CSI at each link, since transmitter CSI incurs extra overhead and requires considerable additional complexity. We define the link outage probability to be the probability that the transmission rate exceeds the instantaneous channel capacity. Specifically, we consider a block fading channel where the channel condition remains constant over one coherent interval and then changes to an independent realization. With average received SNR $\bar{\gamma}$, the instantaneous channel capacity, given by $C = \log_2(1+\bar{\gamma}g)$, is a function of the random channel gain g, and is therefore random. Suppose that the transmission data rate is R bits/s/Hz. The data is received correctly if the instantaneous channel realization is such that $\log_2(1+\bar{\gamma}g)$ is greater than or equal to R. Otherwise the data received over that transmission burst cannot be decoded correctly and the receiver declares an outage. The outage probability is thus given by [10]

$$p_{out}(R) = P\{\log_2(1+\bar{\gamma}g) < R\}.$$

For a Rayleigh fading channel, the corresponding outage probability is given by

$$p_{out}(R) = 1 - \exp\left(-\frac{2^R - 1}{\bar{\gamma}} \right). \tag{5}$$

Accordingly, the δ-outage capacity is defined as the highest rate R such that the outage probability $p_{out}(R)$ is less than δ. Note that there is a one-one mapping between the outage probability and the corresponding outage capacity. For instance, the δ-outage capacity over Rayleigh fading is given by [10]

$$c_{out}(\delta) = \log_2(1 - \bar{\gamma} \ln(1-\delta)). \tag{6}$$

2) Leaky-Pipe Flow Model: The outage probability characterizes the probability of data loss over single link. At a link l with transmission rate R_l, since data is only received correctly on $1 - p_{out}(R_l)$ transmissions, the correctly received data rate is given by [10]

$$R_{eff} = R_l \cdot (1 - p_{out}(R_l)). \tag{7}$$

For a flow s traversing multiple hops, the data rate correctly received at the destination node is give by

$$y_s = x_s \prod_{l \in L(s)} \left[1 - p_l\left(R_l\right) \right], \tag{8}$$

where x_s is the transmission rate at the source, $L(s)$ is the set of links along the route, and $p_l(R_l)$ is the outage probability at link l with link data rate R_l. It can be seen that the data rate of a flow decreases every hop along its route (this is particularly pronounced over wireless links), and it is in this sense that we call it a leaky-pipe flow.

C. Application of Standard NUM over Lossy Links

To apply the standard NUM in wireless networks with lossy links, one needs to specify the link capacity constraints first as in Equation (2). The δ_l-outage capacity is a natural choice, for a link l, allowing the specification of the maximum outage probability, δ_l. Then the standard NUM operates over lossy links as follows: at each link, the link congestion price is calculated as in Equation (4), but using the total data rate observed at the link, not the sum of the injection rates; and at the source node of each flow, the flow rate is updated based on the sum of the congestion prices along its route, as in Equation (3). This straightforward application of the NUM framework takes into account the lossy nature of the links, but only in the way link prices are computed. The lossy nature of the links is not accounted for in the objective function of the NUM problem itself. In the following sections, we develop the ENUM framework, which does take account of the lossy nature of the links in the objective function. We use the standard NUM approach outlined here to provide a benchmark against which we can compare the algorithms we develop.

Notice that the standard NUM approach here does not require cooperation between the transport layer and the link layer, and hence does not require cross-layer design. The quantity δ_l is fixed, and does not adapt to the application-layer requirements embodied in the NUM objective function.

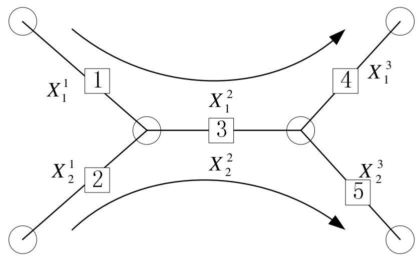

Fig. 1. An example network with leaky-pipe flows

IV. ENUM WITH LINK OUTAGE CONSTRAINTS

In this section, we study the rate control problem with link outage constraints. An example network with leaky-pipe flows is depicted in Figure 1. For each flow s, the number of hops is $H_s = |L(s)|$. The source node of flow s transmits at a rate of x_s over its first link. At the ith link of flow s, the data rate is denoted by $x_s^{l(s,i)}$, where $l(s,i)$ denotes the unique link number of the ith link of flow s. For example, in Figure 1, the third hop of flow 1 has the link number 4, i.e., $l(1,3)=4$. For the sake of brevity, we also denote $x_s^{l(s,i)}$ as x_s^i when there is no confusion in the context. At the destination node of flow s, the correctly received data rate can be represented as $x_s^{H_s+1}$. Each flow s has a utility function associated with the effective rate $x_s^{H_s+1}$.

A principal objective in this problem formulation is to maximize the overall effective network utility across all flows, subject to two constraints: 1) at each link l, there exists an upper bound on the outage probability, which defines the link outage capacity accordingly; and 2) for each flow s, there are flow conservation constraints for connecting links, with the lossy link property taken into account. Then we have the following optimization problem:

$$\begin{aligned} \text{ENUM I:}\quad & \max_{\{x\}} && \textstyle\sum_s U_s(x_s^{H_s+1}) \\ & s.t. && P\{C_l < \textstyle\sum_{s\in S(l)} x_s^l\} \le \delta_l, \quad \forall l \\ &&& x_s^{i+1} = x_s^i[1-p_{out}(l(s,i))], \\ &&& \qquad i = 1\cdots H_s, \quad \forall s, \end{aligned} \tag{9}$$

where $p_{out}(l(s,i))$ is the outage probability of the ith link of flow s. Note that the outage parameter δ_l is intimately related to the tradeoff between rate and outage. If δ_l increases, higher data rate can be transmitted, but higher data loss may take place.

For wireless links experiencing Rayleigh fading, the ENUM problem (9) becomes:

$$\begin{aligned} & \max_{\{x\}} && \textstyle\sum_s U_s(x_s^{H_s+1}) \\ & s.t. && \textstyle\sum_{s\in S(l)} x_s^l \le \hat{c}_l, \quad \forall l \\ &&& x_s^{i+1} = x_s^i \exp\Big(-\frac{2^{\sum_{t\in S(l(s,i))} x_t^{l(s,i)}}-1}{\bar{\gamma}_{l(s,i)}}\Big), \\ &&& \qquad i = 1\cdots H_s, \quad \forall s, \end{aligned} \tag{10}$$

where $\hat{c}_l$ is the corresponding δ_l-outage capacity of link l.

Compared with the standard NUM problem in (2), there are two unique features in the ENUM formulation: 1) The objective function is the overall effective network utility, which yields fairness among the rates achieved at the flow destinations, rather than amongst the rates injected at the sources; 2) A flow conservation constraint is imposed using the leaky-pipe flow model, so that the link data rate is modeled more accurately than in the standard flow model.

The ENUM problem (10) is in general a non-convex problem. However, with some auxiliary variables and appropriate transformation, the problem can be converted into a convex one under some regularity conditions. Define $\tilde{x}_s^l = \log(x_s^l)$. Then problem (10) boils down to the following:

$$\begin{aligned} & \max_{\{x\}} && \textstyle\sum_s U_s(\exp(\tilde{x}_s^{H_s+1})) \\ & s.t. && \textstyle\sum_{s\in S(l)} \exp(\tilde{x}_s^l) \le \hat{c}_l, \quad \forall l \\ &&& \tilde{x}_s^{i+1} \le \tilde{x}_s^i - \frac{2^{\sum_{t\in S(l(s,i))}\exp(\tilde{x}_t^{l(s,i)})}-1}{\bar{\gamma}_{l(s,i)}}, \\ &&& \qquad i = 1\cdots H_s, \quad \forall s, \end{aligned} \tag{11}$$

where we change the equality into inequality, since the objective function is an increasing function of the effective rate.

Under mild conditions [12], the objective function in Equation (11) is a concave function of $\tilde{x}_s^{H_s+1}$. For example, for the family of utility functions defined in Equation (1), the objective function is concave when $\alpha \ge 1$. In this paper, we assume that $U_s(\exp(\cdot))$ is a strictly concave function in its argument. So problem (11) is a convex problem with a unique global optimal point. It is not difficult to see that the Slater condition is satisfied, and strong duality holds [4]. In the following, we use Lagrangian dual method and develop a hop-by-hop algorithm. First, we form the Lagrangian as follows:

$$\begin{aligned} L(\tilde{x},\lambda,\nu) = & \textstyle\sum_s U_s(\exp(\tilde{x}_s^{H_s+1})) \\ & + \textstyle\sum_l \lambda_l \Big(\hat{c}_l - \sum_{s\in S(l)} \exp(\tilde{x}_s^l)\Big) \\ & + \textstyle\sum_s \sum_{i=1}^{H_s} \nu_s^i \Big(\tilde{x}_s^i - \frac{2^{\sum_{t\in S(l(s,i))}\exp(\tilde{x}_t^{l(s,i)})}-1}{\bar{\gamma}_{l(s,i)}} - \tilde{x}_s^{i+1}\Big). \end{aligned} \tag{12}$$

Then the Lagrangian dual function is given by

$$D(\lambda,\nu) = \max_{\{\tilde{x}\}} L(\tilde{x},\lambda,\nu). \tag{13}$$

Accordingly, the dual problem is given by

$$\min_{\{\lambda,\nu\}} \quad D(\lambda,\nu). \tag{14}$$

Following [7], [30], we now devise a primal-dual algorithm to find the optimal rates and prices. More specifically, the rates and prices are updated in such a way that the Lagrangian is maximized with respect to x for the primal, and minimized with respect to λ and ν for the dual.

For convenience, we define $\nu_s^0 = 0, \forall s$. Take derivative of $L(\tilde{x},\lambda,\nu)$ with respect to $\tilde{x}_s^i$. We have the gradient as follows:

for $i = 1, \cdots, H_s$,

$$\begin{aligned} L_{\tilde{x}_s^i}(\tilde{x},\lambda,\nu) = & -\lambda_{l(s,i)}\exp(\tilde{x}_s^i) + \nu_s^i - \nu_s^{i-1} \\ & - \frac{\sum_{t\in S(l(s,i))} \nu_t^{l(s,i)} 2^{\sum_{t\in S(l(s,i))}\exp(\tilde{x}_t^{l(s,i)})}\exp(\tilde{x}_s^i)\log 2}{\bar{\gamma}_{l(s,i)}}, \end{aligned} \tag{15}$$

and for $i = H_s + 1$,

$$L_{\tilde{x}_s^i}(\tilde{x},\lambda,\nu) = U_s'(\exp(\tilde{x}_s^{H_s+1}))\exp(\tilde{x}_s^{H_s+1}) - \nu_s^{H_s}. \tag{16}$$

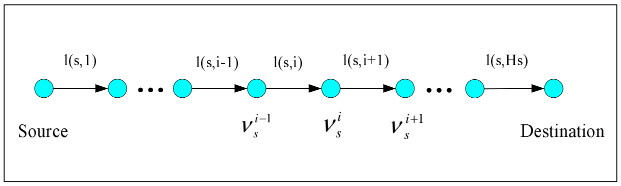

Fig. 2. The hop-by-hop rate control model

The gradient of $L(\tilde{x},\lambda,\nu)$ with respect to ν_s^i is given by

$$L_{\nu_s^i}(\tilde{x},\lambda,\nu)=\tilde{x}_s^i-\frac{2^{\sum_{t\in S(l(s,i))}\exp(\tilde{x}_t^{l(s,i)})}-1}{\bar{\gamma}_{l(s,i)}}-\tilde{x}_s^{i+1}, \quad (17)$$

and the gradient of $L(\tilde{x},\lambda,\nu)$ with respect to λ_l is given by

$$L_{\lambda_l}(\tilde{x},\lambda,\nu)=\hat{c}_l-\sum_{s\in S(l)}\exp(\tilde{x}_s^l). \quad (18)$$

Combining Equations (16) and (17), we use the complementary slackness condition to conclude that

$$\nu_s^{H_s}=U_s'\left(\exp(\tilde{x}_s^{H_s+1})\right)\exp\left(\tilde{x}_s^{H_s+1}\right), \quad (19)$$

$$\tilde{x}_s^{H_s+1}=\tilde{x}_s^{H_s}-\frac{2^{\sum_{t\in S(l(s,H_s))}\exp(\tilde{x}_t^{l(s,H_s)})}-1}{\bar{\gamma}_{l(s,H_s)}}. \quad (20)$$

Next, we use the gradient projection method to find the solution for x, ν and λ [2]. As in [11], the Lagrangian multiplier λ is interpreted as the *link congestion price*. Moreover, from Equation (15), we define the *link error price* at link l as follows:

$$\mu_l=\frac{(\sum_{t\in S(l)}\nu_t^l)2^{\sum_{t\in S(l)}\exp(\tilde{x}_t^l)}\log 2}{\bar{\gamma}_l}. \quad (21)$$

At the ith link of flow s, the flow rate x_s^i can be updated as follows:

$$\begin{aligned}\tilde{x}_s^i(n+1)=&\left[\tilde{x}_s^i(n)+\varepsilon(n)\left(\nu_s^i(n)-\nu_s^{i-1}(n)\right.\right.\\&\left.\left.-\ \exp(\tilde{x}_s^i(n))(\lambda_{l(s,i)}(n)+\mu_{l(s,i)}(n))\right)\right]_{\tilde{I}_s}, \quad (22)\\&\qquad i=1,\cdots,H_s,\end{aligned}$$

where $\varepsilon(n)$ is the step size.

At the receiver of the ith link of flow s, ν_s^i can be updated as follows:

$$\begin{aligned}\nu_s^i(n+1)=&\left[\nu_s^i(n)-\varepsilon(n)\left(\tilde{x}_s^i(n)-\tilde{x}_s^{i+1}(n)\right.\right.\\&\left.\left.-\ \frac{2^{\sum_{t\in S(l(s,i))}\exp(\tilde{x}_t^{l(s,i)}(n))}-1}{\bar{\gamma}_{l(s,i)}}\right)\right]^+, i=1,\cdots,H_s-1.\end{aligned} \quad (23)$$

At link l, the link congestion price λ_l can be updated as follows:

$$\lambda_l(n+1)=\left[\lambda_l(n)-\varepsilon(n)\left(\hat{c}_l-\sum_{s\in S(l)}\exp(\tilde{x}_s^l(n))\right)\right]^+, \quad l=1,\cdots,L. \quad (24)$$

The hop-by-hop algorithm for leaky-pipe flows can be summarized in Algorithm 1 (see Figure 2 for the model of the hop-by-hop algorithm).

Algorithm 1 ENUM with link outage constraints

Initialization: Set $n=0$, $x_s=x_s(0)$ and $\nu_s=\nu_s(0)$ for each flow, and $\lambda_l=\lambda_l(0)$ for each link, where $x_s(0)$, $\nu_s(0)$ and $\lambda_l(0)$ are some nonnegative vectors (with appropriate dimensions).

Iterations:

1) At the ith link of flow s, the link transmits with data rate $x_s^i(n)$.

2) At the ith link of flow s, the receiving node measures the correctly received incoming rate and the outgoing rate, and then updates the ν_s^i based on Equation (23).

3) At each link l, the link congestion price λ_l and link error prices μ_l are updated based on Equation (24) and (21), respectively.

4) At the ith link of flow s, the transmission rate x_s^i is updated based on Equation (22).

5) Set n = n+1. Go to step 1 (until satisfying termination criterion).

Remarks:

1. The link price consists of the link congestion price (which accounts for link congestion) and the link error price (which accounts for link channel condition), and the sum of the two prices is used for the rate update at the transmitter of each link, as we can see from Equation (22).

2. Equation (23) reveals that ν_s^i is intimately related to the available processing capacity (e.g., available buffer space) at the receiver of the link. For instance, ν_s^i decreases if the incoming rate $\tilde{x}_s^i(n)-\frac{2^{\sum_{t\in S(l(s,i))}\exp(\tilde{x}_t^{l(s,i)}(n))}-1}{\bar{\gamma}_{l(s,i)}}$ is greater than the outgoing rate $\tilde{x}_s^{i+1}(n)$. Accordingly, the rate update depends on the difference of available processing capacities between adjacent nodes $(\nu_s^i-\nu_s^{i-1})$, as we can see from Equation (22). This is reminiscent of the back-pressure algorithm where the flow with greater queue length difference is granted higher priority. In fact, if we treat ν_s^i as the available buffer and assume each node has the same buffer size, then $(\nu_s^i-\nu_s^{i-1})$ can be viewed as the queue length difference between the transmitter and receiver at the ith link of flow s.

3. Based on Equation (19), $\nu_s^{H_s}$ can be viewed as the effective flow weight, which guarantees the fairness among the destination nodes. This can be explained by the fact that $\nu_s^{H_s}$ is a non-increasing function of the effective rates $x_s^{H_s+1}$. When the effective rate $x_s^{H_s+1}$ increases, its weight $\nu_s^{H_s}$ decreases consequently.

The convergence of Algorithm 1 can be proved using standard techniques in distributed gradient algorithm's convergence analysis [2], [29]. For the sake of limited space, we omit the proof here. For convenience, we define $x\triangleq\{x_1^1,\cdots,x_1^{H_1+1},\cdots,x_S^1,\cdots,x_S^{H_S+1}\}$ and $p\triangleq\{\lambda_1,\cdots,\lambda_L,\nu_1^1,\cdots,\nu_1^{H_1},\cdots,\nu_S^1,\cdots,\nu_S^{H_S}\}$. Suppose that the step size satisfies the following condition: $\varepsilon(n)>0,\varepsilon(n)\to 0,\sum_n\varepsilon(n)\to\infty$, and $\sum_n\varepsilon(n)^2<\infty$. Define $m=\{m_s,s\in S\}$ and $M=\{M_s,s\in S\}$. We have the flowing result.

Theorem 4.1: For any given initial rates $m < x(0) < M$ and prices $p(0) \geq 0$, the sequence $\{x(n), p(n)\}$, generated by Algorithm 1, converges to the optimal point.

V. ENUM WITH PATH OUTAGE CONSTRAINTS

In the layered approach with standard NUM, the link layer does not know the specifics about the flows, e.g., how many hops a flow may traverse, or the application-layer requirements of the flow; thus, a conservative approach to setting the link outage probability target needs to be made. For example, since some flows require highly reliable services, and may traverse many hops, it is necessary to make the links very reliable to accommodate this case. However, in the hop-by-hop ENUM algorithm described above, there is an interaction between the layers (in the way price signals are computed) and per flow rate allocation takes place on each link, so there is no intrinsic need to provide a per-link outage constraint in this case. An alternative, as we shall investigate in this section, is to provide a path outage constraint for each flow, and let the ENUM algorithm find the appropriate rate-outage tradeoff for each flow on each link in a dynamic manner. Specifically, for flow s, it is required that the end-to-end outage is no greater than $1 - \pi_s$. Then the problem is formulated as follows:

$$\begin{array}{lll} \text{ENUM II:} & \max\limits_{\{x\}} & \sum_s U(x_s^{H_s+1}) \\ & s.t. & \frac{x_s^{H_s+1}}{x_s^1} \geq \pi_s, \quad \forall s \\ & & x_s^{i+1} = x_s^i[1 - p_{out}(l(s,i))], \\ & & \qquad i = 1 \cdots H_s, \quad \forall s. \end{array} \tag{25}$$

Define $\tilde{x}_s^l = \log(x_s^l)$. For wireless link experiencing Rayleigh fading, problem (25) reduces to the following:

$$\begin{array}{ll} \max\limits_{\{x\}} & \sum_s U_s(\exp(\tilde{x}_s^{H_s+1})) \\ s.t. & \tilde{x}_s^1 + \log(\pi_s) \leq \tilde{x}_s^{H_s+1}, \quad \forall s \\ & \tilde{x}_s^{i+1} \leq \tilde{x}_s^i - \frac{2^{\sum_{t \in S(l(s,i))} \exp(\tilde{x}_t^{l(s,i)})} - 1}{\bar{\gamma}_{l(s,i)}}, \\ & \qquad i = 1 \cdots H_s, \quad \forall s, \end{array} \tag{26}$$

where we change the equality into inequality, since the objective function is an increasing function of the effective rate. Write down the Lagrangian as

$$\begin{aligned} &L(\tilde{x}, \theta, \nu) = \textstyle\sum_s U_s(\exp(\tilde{x}_s^{H_s+1})) \\ &+ \textstyle\sum_s \theta_s \left(\tilde{x}_s^{H_s+1} - \tilde{x}_s^1 - \log(\pi_s)\right) \\ &+ \textstyle\sum_s \sum_{i=1}^{H_s} \nu_s^i \left(\tilde{x}_s^i - \frac{2^{\sum_{t \in S(l(s,i))} \exp(\tilde{x}_t^{l(s,i)})} - 1}{\bar{\gamma}_{l(s,i)}} - \tilde{x}_s^{i+1} \right). \end{aligned} \tag{27}$$

Using the same approach as in Section IV, we derive the primal-dual algorithm as follows. The link error price μ_l at link l is defined the same as in Equation (21) in Section IV. At the ith link of flow s, the flow rate x_s^i can be updated as follows:
for $i = 1$,

$$\begin{aligned} \tilde{x}_s^1(n+1) = \big[\tilde{x}_s^1(n) + \varepsilon(n) \big(-\theta_s(n) + \nu_s^1(n) \\ - \exp(\tilde{x}_s^1(n))\mu_{l(s,1)}(n)\big)\big]_{\tilde{I}_s}, \end{aligned} \tag{28}$$

for $i = 2, \cdots, H_s$,

$$\begin{aligned} \tilde{x}_s^i(n+1) = \big[\tilde{x}_s^i(n) + \varepsilon(n) \big(\nu_s^i(n) - \nu_s^{i-1}(n) \\ - \exp(\tilde{x}_s^i(n))\mu_{l(s,i)}(n)\big)\big]_{\tilde{I}_s}, \end{aligned} \tag{29}$$

for $i = H_s + 1$,

$$\tilde{x}_s^{H_s+1}(n+1) = \tilde{x}_s^{H_s}(n) - \frac{2^{\sum_{t \in S(l(s,H_s))} \exp(\tilde{x}_t^{l(s,H_s)}(n))} - 1}{\bar{\gamma}_{l(s,H_s)}}. \tag{30}$$

At the receiver of the ith link of flow s, ν_s^i can be updated as follows:
for $i = 1, \cdots, H_s - 1$,

$$\begin{aligned} \nu_s^i(n+1) = \Big[\nu_s^i(n) - \varepsilon(n) \Big(\tilde{x}_s^i(n) - \tilde{x}_s^{i+1}(n) \\ - \tfrac{2^{\sum_{t \in S(l(s,i))} \exp(\tilde{x}_t^{l(s,i)}(n))} - 1}{\bar{\gamma}_{l(s,i)}}\Big)\Big]^+, \end{aligned} \tag{31}$$

for $i = H_s$,

$$\nu_s^{H_s}(n+1) = \theta_s(n) + U_s'\left(\exp(\tilde{x}_s^{H_s+1}(n))\right) \exp\left(\tilde{x}_s^{H_s+1}(n)\right). \tag{32}$$

At the destination node of flow s, the flow reliability price θ_s can be updated as follows:

$$\begin{aligned} \theta_s(n+1) = \big[\theta_s(n) - \varepsilon(n) \big(\tilde{x}_s^{H_s+1}(n) - \tilde{x}_s^1(n) \\ - \log(\pi_s)\big)\big]^+, s = 1, \cdots, S. \end{aligned} \tag{33}$$

The hop-by-hop algorithm for ENUM with path outage constraints is summarized as follows:

Algorithm 2 ENUM with path outage constraints

Initialization: Set $n = 0$, $x_s = x_s(0)$ and $\nu_s = \nu_s(0)$ for each flow, and $\theta_s = \theta_s(0)$ for each link, where $x_s(0)$, $\nu_s(0)$ and $\theta_s(0)$ are some nonnegative vectors (with appropriate dimensions).

Iterations:

1) At the ith link of flow s, the link transmits with data rate $x_s^i(n)$.

2) At the ith link of flow s, the receiving node measures the correctly received incoming rate and the outgoing rate, and then updates the ν_s^i based on Equation (31) for $i = 1, \cdots, H_s - 1$ and Equation (32) for $i = H_s$.

3) At the destination node of flow s, the flow reliability price θ_s is updated based on Equation (33). It is then fed back to the source node of flow s.

4) At the ith link of flow s, the transmission rate x_s^i is updated based on Equation (28) for $i = 1$, Equation (29) for $i = 2, \cdots, H_s$, and Equation (30) for $i = H_s + 1$.

5) Set n = n+1. Go to step 1 (until satisfying termination criterion).

Remarks:

1. The flow rate and the link outage are jointly optimized by Algorithm 2. With an end-to-end reliability requirement, the optimal link outage is derived dynamically from the algorithm. In contrast, link layer algorithms that are designed independently of higher layers are unable to make these adjustments, and designers are forced to make conservative assumptions.

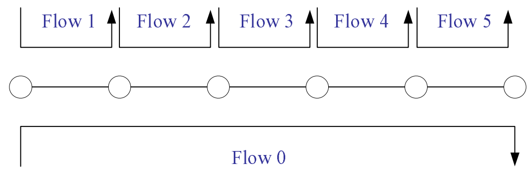

Fig. 3. Network model I

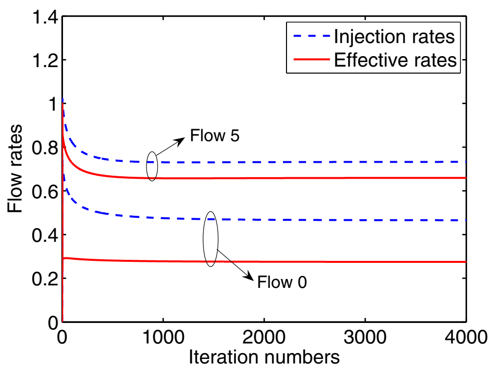

Fig. 4. NUM: injection rates vs effective rates

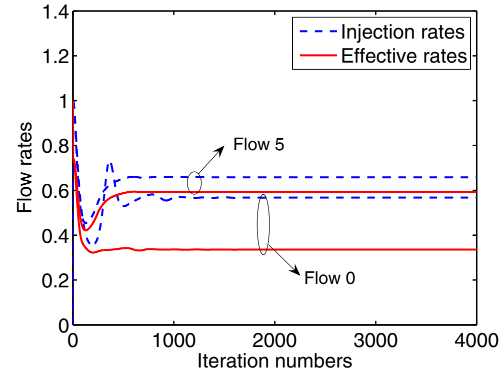

Fig. 5. ENUM: injection rates vs effective rates

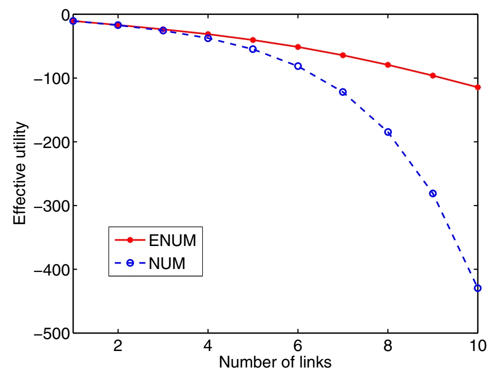

Fig. 6. Effective utility: NUM vs ENUM

2. Corresponding to the end-to-end outage constraint, the flow reliability price θ_s is introduced. When the end-to-end reliability $x_s^{H_s+1}/x_s^1$ is less than the threshold π_s, the reliability price θ_s increases, as we can see from Equation (33).

3. At the intermediate links of each flow, the link error price reflects the supply-demand relationship. When the total data rate over the link increases, the link error price increases accordingly; and vice versa.

The convergence of Algorithm 2 can also be proved using standard techniques. For convenience, we define $x \triangleq \{x_1^1, \cdots, x_1^{H_1+1}, \cdots, x_S^1, \cdots, x_S^{H_S+1}\}$ and $q \triangleq \{\theta_1, \cdots, \theta_S, \nu_1^1, \cdots, \nu_1^{H_1}, \cdots, \nu_S^1, \cdots, \nu_S^{H_S}\}$. We have the following theorem:

Theorem 5.2: For any given initial rates $m < x(0) < M$ and prices $q(0) \geq 0$, the sequence $\{x(n), q(n)\}$, generated by Algorithm 2, converges to the optimal point.

VI. Numerical Examples

A. ENUM with Link Outage Constraints

In this subsection, we use numerical examples to illustrate the advantage of the ENUM over the standard NUM, with same link outage constraints. Consider a wireless network with n links, where the links are numbered from 1 to n, with link 1 at the leftmost. The average SNR of each link is 10dB and the link outage constraint is set to be $\delta_l = 0.1, l \in 1 \cdots n$. The n links are shared by $n+1$ flows. Specifically, flow 0 traverses all the links, and flow 1 to flow n are one-hop flows over link 1 to link n, respectively. An example network with 5 links are shown in Figure 3.

For the sake of illustration, we use the α-fair utility function with $\alpha = 5$, as defined in Equation (1). In the standard NUM approach, the algorithm described in subsection III-C is used. For the network with 5 links, the injection rates and the effective rates of flow 0 and flow 5 from NUM are depicted in Figure 4. It can be seen that the effective rate of flow 0 (0.2748) is much lower than that of flow 5. For ENUM, the injection rates and the effective rates are shown in Figure 5. Although flow 0 still has the lowest rate among the flows, it can be seen that ENUM yields a higher effective rate (0.3356) for flow 0 than under NUM. In fact, the effective rates of the flows in ENUM are closer to each other than those in the NUM approach, indicating that better fairness among effective rates is achieved by ENUM. Our intuition for this is that ENUM explicitly takes into account the data loss in its objection function, whereas NUM is concerned with the injection rates only.

Next, we investigate the performance while increasing the number of links in the network from 1 to 10. The effective utilities are shown in Figure 6 as a function of the number of links in the network. Clearly, the effective utility of ENUM can be significantly higher than that of NUM; and the gain of ENUM over NUM increases as the number of hops increases.

We also take a closer look at flow 0. The equilibrium injection rates and the effective rates of flow 0, from both NUM and ENUM, are plotted in Figure 7 as a function of the number of links. As the number of hops increases, the injection

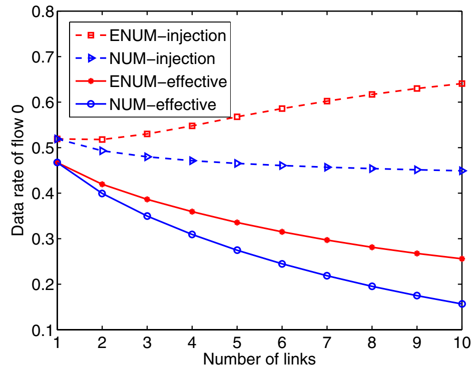

Fig. 7. Date rates of flow 0

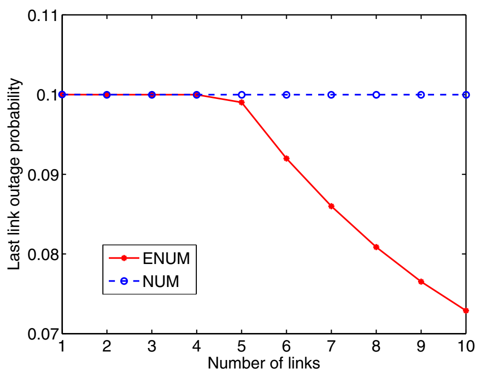

Fig. 8. Outage probability of last hop

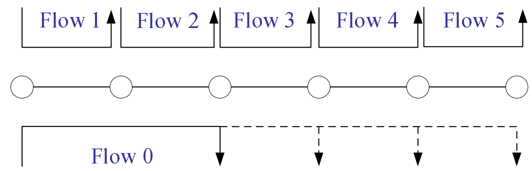

Fig. 9. Network model II

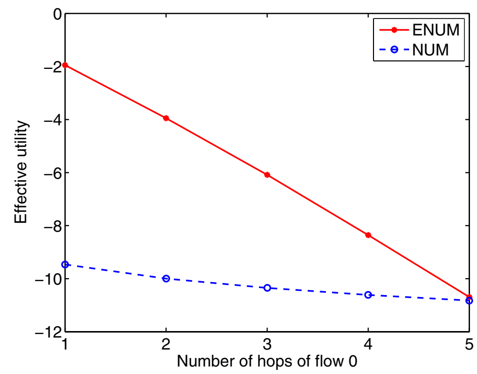

Fig. 10. Effective utility

rate of flow 0 in ENUM increases, whereas it decreases in NUM. Accordingly, ENUM achieves a higher effective rate for flow 0 than does NUM. This is also due to the fairness embodied in the objective function: the injection rate of flow 0 in ENUM increases to compensate for the higher end-to-end loss rate as the number of hops increases.

It is also of interest to examine the actual link outage probabilities of the links. In Figure 8, we show the outage probabilities of the last link with increasing network size. It can be seen that when the number of links is small, the links are more or less fully occupied in both NUM and ENUM. However, as the number of links increases, the one-hop flows in ENUM decrease their injection rates, incurring a smaller outage probability. Consequently, the n-hop flow suffers less data loss along its route and better fairness among the effective rates is achieved. This is the main reason why the overall gain in effective utility for ENUM increases with the number of hops in the network. The rate-outage tradeoff on each link adapts to improve the overall effective utility of the network.

B. *ENUM with Path Outage Constraints*

In the following, we examine ENUM with path outage constraints. As shown in Figure 9, we consider a network model with 5 links, each with an average SNR 10dB. The network is shared by 6 flows. Specifically, flow 1 to flow 5 are one-hop flows over link 1 to link 5, respectively, while flow 0 may travel over multiple hops, starting from link 1. For example, in Figure 9, flow 0 traverses two hops over link 1 and link 2.

We use logarithm utility function to study the performance of NUM and ENUM. For ENUM, we set the path outage constraint to be $1 - \pi_s = 0.1$. For NUM, since the link constraint needs to be pre-determined, a conservative link constraint needs to be pre-set to accommodate the case where a flow traverses all the links. For that case, the link outage constraints of NUM are set to be $\delta = 0.02$ for all links.

We first show in Figure 10 the effective utilities obtained by ENUM and NUM, as the number of hops of flow 0 increase from 1 to 5. We can see ENUM achieves higher effective utility than NUM. When the number of hops increases, the improvement of ENUM over NUM decreases. This can be explained as follows. When the actual number of hops is small, the end-to-end reliability yields a link outage probability higher than 0.02, and accordingly higher data rates are allowed over the link and higher effective rates can be achieved by ENUM. As the number of hops increases, the link outage probabilities obtained by ENUM approach the conservative choices made by NUM, and lower effective utilities are consequently achieved.

To examine the joint optimization of flow rate and link outage, we take a closer look at the data rate of flow 0, as depicted in Figure 11. It can be clearly seen that as the number of hops increases, the data rate of flow 0 decreases. This is due to the fact that, for a given end-to-end flow reliability requirement, the optimal link outage probability depends on the flow specifications, especially on the number of hops of the flows. As the number of hops increases, a lower link outage

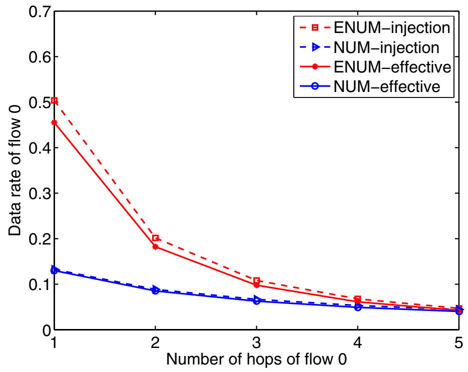

Fig. 11. Date rates of flow 0

probability is achieved allowing the flow to meet the end-to-end reliability requirement; and lower effective rates are obtained as a result.

VII. Conclusions

We take a cross-layer optimization approach to study rate control in multi-hop wireless networks with lossy links. Due to link error, the data rate of a given flow becomes "thinner and thinner" along its path. For this leaky-pipe flow model, we define the injection rate as the transmission rate at the source node, and the effective rate as the correctly received data rate at the destination node. Associated with the effective rate, each flow has an effective utility function. We explore rate control via effective network utility maximization for two network models. First, we study ENUM with link outage constraints. Compared to standard NUM with the same link outage constraints, ENUM yields higher effective network utility and better fairness among the effective rates. Second, we consider ENUM with path outage constraints. With an end-to-end outage requirement, the optimal link outage probabilities are dynamically determined by the rate control algorithm. In contrast, the standard NUM may need to make a conservative choice about link outage probability to accommodate the worst case scenario. For both problems, the thinning feature of data flows is explicitly taken into account and back-pressure hop-by-hop rate control algorithms are developed. We show that considerable gains in effective network utility and fairness among effective flow rates are achieved by the ENUM approach.

References

[1] H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Trans. Networking*, vol. 5, no. 6, pp. 756–769, 1997.
[2] D. Bertsekas and J. Tsitsiklis, *Parallel and distributed computation*. Athena scientific, 1997.
[3] E. Biglieri, J. Proakis, and S. Shamai, "Fading channels: Information-theoretic and communications aspects," *IEEE Trans. Inform. Theory*, vol. 44, 1998.
[4] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
[5] M. Chen and A. Zakhor, "Flow control over wireless network and application layer implementation," in *Proc. IEEE Infocom '06*, Barcelona, Spain, Apr. 2006.
[6] X. Chen, H. Zhai, J. Wang, and Y. Fang, "A survey on improving TCP performance over wireless networks," in: Resource Management in Wireless Networking, edited by M. Cardei, I. Cardei and D.-Z. Du, pp. 657-695, Kluwer Academic Publishers/Springer, 2005 .
[7] E. K. P. Chong and B. E. Brewington, "Decentralized rate control for tracking and surveillance networks," *Ad Hoc Networks*, vol. 5, no. 6, pp. 910–928, 2007.
[8] R. Gibbens and F. Kelly, "On packet marking at priority queues," *IEEE Trans. Automat. Contr.*, vol. 47, pp. 1016–1020, 2002.
[9] A. Goldsmith and P. Varaiya, "Capacity of fading channels with channel side information," *IEEE Trans. Inform. Theory*, vol. 43, no. 6, pp. 1986–1992, Nov. 1997.
[10] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.
[11] F. P. Kelly, A. Maulloo, and D. K. H. Tan, "Rate control for communication networks: shadow price, rairness and stability," *Journal of Operational Research Society*, vol. 49, no. 3, pp. 237–252, Mar. 1998.
[12] J. Lee, M. Chiang, and A. Calderbank, "Joint optimal congestion and medium access control in ad hoc wireless networks," *IEEE Commun. Letters*, vol. 10, no. 3, pp. 216–218, Mar. 2005.
[13] J. Lee, M. Chiang, and A. R. Calderbank, "Price-based distributed algorithms for rate-reliabiity tradeoff in network utility maximization," *IEEE J. Select. Areas Commun.*, vol. 24, no. 5, pp. 962–976, May 2006.
[14] P. Lefevre and G. Vivier, "Understanding TCP's behavior over wireless links," in *Proc. Commun. and Vehic. Tech.*, 2000, pp. 123–130.
[15] X. Lin, N. Shroff, and R. Srikant, "On the connection-level stability of congestion-controlled communication networks," *submitted to IEEE Trans. Inform. Theory*, 2005.
[16] ——, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. Select. Areas Commun.*, vol. 24, no. 8, pp. 1452– 1463, Aug. 2006.
[17] X. Lin and N. B. Shroff, "The impact of imperfect scheduling on cross-layer rate control in multihop wireless networks," in *Proc. IEEE INFOCOM '05*, Miami, Florida, 2005.
[18] S. Low and D. Lapsley, "Optimization flow control I: Basic algorithm and convergence," *IEEE/ACM Trans. Networking*, vol. 7, no. 6, pp. 861–874, Dec. 1999.
[19] S. Low and R. Srikant, "A mathematical framework for designing a low-loss low-delay Internet," *Network and Spatial Economics*, vol. 4, no. 1, pp. 75–101, 2004.
[20] P. Mishra, H. Kanakia, and S. K. Tripathi, "On hop-by-hop rate-based congestion control," *IEEE/ACM Trans. Networking*, vol. 4, no. 2, pp. 224–239, 1996.
[21] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Networking*, vol. 8, no. 5, pp. 556–567, Oct. 2000.
[22] S. Nanda, K. Balachandran, and S. Kumar, "Adaptation techniques in wireless packet data services," *IEEE Commun. Mag.*, vol. 38, no. 1, pp. 54–64, Jan. 2000.
[23] C. Ozveren, R. Simcoe, and G. Varghese, "Reliable and efficient hop-by-hop flow control," *IEEE J. Select. Areas Commun.*, vol. 13, no. 4, pp. 642–650, 1995.
[24] N. Samaraweera, "Non-congestion packet loss detection for TCP error recovery using wireless links," *IEE Proceedings of Communications*, vol. 146, no. 4, Aug. 1999.
[25] B. Scheuermann, C. Lochert, and M. Mauve, "Implicit hop-by-hop congestion control in wireless multihop networks," *Ad Hoc Networks*, 2007.
[26] L. Tassiulas, "Adaptive back-pressure congestion control based on local information," *IEEE Trans. Automat. Contr.*, vol. 40, no. 2, pp. 236–250, Feb. 1995.
[27] C. Wang, K. Sohraby, and B. Li, "SenTCP: a hop-by-hop congestion control protocol for wireless sensor networks," in *Proc. IEEE Infocom '05*, Mar. 2005.
[28] Y. Yi and S. Shakkottai, "Hop-by-hop congestion control over a wireless multi-hop network," in *Proc. IEEE Infocom '04*, Hong Kong, China, Mar. 2004, pp. 2548– 2558.
[29] J. Zhang and D. Zheng, "A stochastic primal-dual algorithm for joint flow control and mac design in multi-hop wireless networks," in *Proc. CISS '06*, Princeton, Mar. 2006.
[30] J. Zhang, D. Zheng, and M. Chiang, "Impact of stochastic noisy feedback on distributed network utility maximization," in *Proc. IEEE Infocom '07*, Anchorage, Alaska, USA, May 2007.