

This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

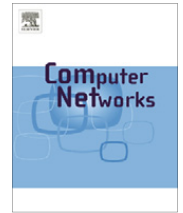
In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

<http://www.elsevier.com/copyright>



Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Congestion control and its stability in networks with delay sensitive traffic [☆]

Ying Li ^{a,*}, Antonis Papachristodoulou ^b, Mung Chiang ^a, A. Robert Calderbank ^a

^a Department of Electrical Engineering, Princeton University, Princeton, NJ 08544, United States

^b Department of Engineering Science, University of Oxford, Oxford OX1 3PJ, United Kingdom

ARTICLE INFO

Article history:

Received 6 February 2010

Received in revised form 16 May 2010

Accepted 2 July 2010

Available online 22 July 2010

Responsible Editor: A. Pitsillides

Keywords:

Congestion control

Stability

Quality of service

Optimization

ABSTRACT

We consider congestion control in a network with delay sensitive/insensitive traffic, modelled by adding explicit delay terms to the utility function measuring user's happiness on the Quality of Service (QoS). A new Network Utility Maximization (NUM) problem is formulated and solved in a decentralized way via appropriate algorithms implemented at the users (primal) and/or links (dual). For the dual algorithm, delay-independent and delay-dependent stability conditions are derived when propagation delays are taken into account. A system with voice and data traffic is considered as example and the properties of the congestion control algorithm are assessed.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

Internet congestion control [22] is a distributed algorithm to allocate available resources to competing sources so as not to exceed link capacities and hence avoid congestion collapse. Congestion signals are generated at the Active Queue Management (AQM) part of the algorithm implemented at the links; the congestion measure is usually based on either delay or packet loss. The source rates are then adapted at the Transmission Control Protocol (TCP) part of the algorithm according to the size of the aggregate price signal that the user sees on the links it uses. The challenge is to use these feedback signals in order to stabilize the system around a 'fair' resource allocation equilibrium for arbitrary networks in a robust way.

As extensively researched in the last few years, the bandwidth allocation problem can be cast as an optimization program [14,13], in the framework of Network Utility Maximization (NUM) [6]. Since the publication of the seminal paper [6] in 1998, the framework of NUM has found many applications in network resource allocation algorithms and the design of protocol stacks [4]. Consider a communication network with L logical links, wired or wireless, each with a fixed capacity of c_l bps, and S sources (i.e., end users), each transmitting at a source rate of x_s bps. Each source s emits one flow, using a fixed set $L(s)$ of links in its path, and has a utility function $U_s(x_s)$. Each link l is shared by a set $S(l)$ of sources. NUM, in its basic version, is the following problem of maximizing the aggregate utility, over the source rates \mathbf{x} , subject to linear flow constraints for all links:

$$\begin{aligned} & \text{maximize} && \sum_s U_s(x_s) \\ & \text{subject to} && \sum_s R_{ls} x_s \leq c_l, \quad \forall l, \\ & && \mathbf{x} \succeq 0, \end{aligned} \quad (1)$$

where R is the routing matrix,

[☆] This research was in part supported by NSF CCF-0448012, DARPA Grant HR0011-06-1-0008, and Oxford–Princeton collaboration scheme. Part of this paper was presented at Globecom 2007 [9] and American Control Conference 2008 [12].

* Corresponding author. Tel.: +1 469 417 9462.

E-mail addresses: yingli@princeton.edu, yli2@sta.samsung.com (Y. Li), antonis@eng.ox.ac.uk (A. Papachristodoulou), chiangm@princeton.edu (M. Chiang), calderbk@princeton.edu (A.R. Calderbank).

$$R_{ls} = \begin{cases} 1, & \text{if source } s \text{ uses link } l, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

The utility of each user s is a strictly concave, continuously differentiable non-decreasing function $U_s(x_s)$, measuring the user's 'happiness' when allocated rate x_s . Essentially, it is a measure of the Quality of Service (QoS). For best-effort data traffic, a function of the transmission rate alone is good enough to reflect QoS, but for delay sensitive traffic, a utility function that is only a function of the transmission rate may not reflect the user's perception of QoS. For example, for a Voice over IP (VoIP) application, the R -factor, which is a measure of user's satisfaction on the QoS, has a linear term on the end-to-end average delay [1].

When the utility functions are only functions of the transmission rate, the resource allocation problem can be decomposed into a primal and a dual problem by introducing duality-based price signals [14]. In this way, the congestion measures are the 'dual' variables, while the transmission rates are the 'primal' variables. The aim of the designed AQM and TCP algorithms is to drive the congestion signals and the source rates at or close to the optimum of the distributed resource allocation optimization problem. In the simplest case, the structure of the dynamics that are chosen for TCP and AQM are usually based on a sub-gradient descent algorithm on the dual decomposition [6,4,13,2], whose nonlinear dynamics can be shown to be asymptotically stable.

In this work, we study how to accommodate diverse applications in a network, where different types of traffic may have very different requirements for rate and delay. We extend the congestion control in [6] where the users are sensitive to the rate only, to the case where the users are sensitive to packet delay, as well as the rate. We focus on a new utility function which incorporates both the requirement on rate and delay. For user s , suppose the end-to-end average packet delay is $\sum_l R_{ls} d_l(y_l)$ where $y_l = \sum_s R_{ls} x_s$ is the link load on link l and $d_l(y_l)$ is the average delay experienced by a packet on link l . The new utility function of user s is

$$U_s = f_s(x_s) - b_s \sum_l R_{ls} d_l(y_l), \quad (3)$$

where f_s is a function of rate x_s and $b_s > 0$ is some constant incorporating the normalization and the relative importance of the delay versus rate of user s .

The new utility function (3) has an explicit term of delay and it couples the rate of the user and the flow on the links. Different from the objective function in problem (1) studied in [6] which is a sum of the utility of separable source rate, the objective function (3) is not a separable function of each user's rate any more: the users' rates are coupled in the delay function. In addition, the objective function (3) has a different motivation than the model in [6], since the objective function with explicit delay is suitable for applications such as VoIP. It explicitly reflects how the transmission rate and delay affect users' satisfaction of the service, and provides a richer framework on how to accomplish rate control for delay sensitive traffic.

We formulate a new NUM problem with this new utility function and solve it using a primal algorithm and a dual algorithm to design user and link dynamics. Compared to the basic NUM (1) we find that the dynamics for link price now depend on the delay term added in the utility function. A system with voice and data traffic is considered and a distributed resource allocation algorithm is proposed such that the utility of voice and data is jointly optimized. We find that the proposed algorithm can achieve higher quality of voice traffic and higher throughput of data traffic, sacrificing packet delays in data traffic.

Furthermore, we consider how the presence of communication delays in the network can affect the stability of the dual algorithm. Delays should not be ignored as in general their presence results in degradation of performance or even instabilities. In this paper we derive both delay-independent and delay-dependent stability conditions when propagation delays are taken into account.

The main contributions of this paper are as follows:

- (1) We analyze the dynamics for the NUM when explicit terms of delay in the utility functions, depending on the users' application, are included. The analysis gives us more insight on how to accomplish optimal congestion control for users running various applications with different requirements on rate and delay.
- (2) We apply the analysis to a system with voice and data traffic, and we propose distributed algorithms to allocate the resource such that the utility of voice and data is jointly optimized.
- (3) For the dual algorithm, delay-independent and delay-dependent stability conditions are derived for arbitrary network topologies, taking into account the communication delays in the dynamics.

The rest of the paper is organized as follows. Related work is introduced in Section 2. In Section 3, we develop algorithms that solve the NUM problem, in which user utilities are functions of the transmission rate and queueing and transmission delay. This results in a richer rate and queueing control algorithms whose dynamics are analyzed ignoring forward and backward communication delays. In Section 4, we illustrate the results on a system with voice and data traffic. In Section 5 we consider the stability of the dual algorithm in the presence of forward/backward communication delays in the dynamics, revealing a delay-independent condition, something unique for dual congestion control algorithms studied so far. Numerical examples are provided in Section 6. We conclude the paper in Section 7.

2. Related work

The seminal paper [6] focused on a utility function which was only a function of transmission rate and developed dynamics of the rate regulation at every source and the congestion price update at every link. The dynamics are shown to converge to the unique equilibrium which is the optimal solution of problem (1), as proved in [6].

Further studies on the stability of the dynamics have been undertaken in [14] (see also [22]). One thread of the extensions of the basic NUM framework in [6] is congestion control for delay sensitive traffic, e.g., [21,20,9–11] (see also the references therein), although this topic is under-explored.

In [21], the congestion control is studied for users with heterogeneous delay sensitivity. The utility function is $\hat{U}_s = U_s(x_s) - x_s h_s \sum_l R_{ls} d_l(y_l)$ where $y_l = \sum_s R_{ls} x_s$ is the total load on link l , $h_s \sum_l R_{ls} d_l(y_l)$ is the delay cost per unit flow, and h_s is a constant. But this utility \hat{U}_s does not always reflect users' requirements for rate and delay based on users' application. For example, for VoIP application, the R -factor which is a measure of user's satisfaction of the QoS, has a linear term of the average delay, not the product of the rate and the average delay [1].

In [20], the authors study joint congestion control and routing. By reverse engineering, when the link cost includes propagation delay, the optimization problem behind it is to maximize the total utility where the utility function has propagation delay in it. But the utility function does not include explicit average queueing delay, which can be a function of users' rates.

A more general utility function is considered in [10,11], which is a function of rate, reliability and delay, where the delay includes queueing and transmission delay. In [10,11], the optimal rate–reliability–delay tradeoff is studied for a network with composite links, where each link consists of several sub-links with different rate and reliability.

Unlike [10,11], in this paper, instead of examining the rate–reliability–delay tradeoff, we focus on the equilibrium and dynamic properties of the resulting dynamical system for utilities that are functions of both the transmission rate the end-to-end average delay, as in (3). Unlike [10,11] which studies the network with composite links, we focus on the single sub-link case, i.e., problem (1) with objective (3), which is the case that each link consists of only one sub-link with one pair of rate and reliability. When we study the single sub-link case, the special structure allows us to design a more simplified algorithm compared with [10,11], while [10,11] did not reveal the problem structure that enables a simplified algorithm. Another important difference is that [10,11] did not study the stability when the communication delays are taken into account in the dynamics, while this paper studies it and derives the stability conditions.

To be specific, we analyze the stability of these dynamics, and propose a simple algorithm for the single sub-link case. The case we study is problem (1) with objective (3), and this case deserves particular attention as networks with links each being single sub-link are common; their special structure makes the algorithm simpler and the algorithm we propose can ensure delay sensitive traffic over such networks. The comparison of this work and [10,11] is listed in Table 1.

In another thread of extending the dynamics analysis of the basic NUM in [6], some researchers have included communication delays in the dynamics and developed stability analysis conditions, such as in [16,23]. The problem of analyzing system behavior at the linear and nonlinear level with heterogeneous delays is difficult [19] but several procedures have been developed for that purpose: for example the methodology developed in [16,23] analyzes the linearizations of the nonlinear equations and the methods developed in [18,24] deal with nonlinear system descriptions.

In this paper we incorporate communication delays in the dynamics developed for the network with delay sensitive traffic, use the linearization of the scheme around the optimum, and derive two conditions, one based on the delay and one that is delay-independent. The basic NUM (1) [6], is thus extended in two directions: on one hand, the users have utility functions that incorporate the delay, which provides a richer profile for rate adaptation; but also, communication delays in the feedback dynamics are included and stability conditions are derived. Using similar tools as in [16,23] which only derived delay-dependent stability conditions, we find that for this problem, delay-dependent and delay-independent conditions arise, something unique in the case of dual congestion schemes studied thus far.

Table 2 summarizes the key differences between this paper with related work.

3. Congestion control for delay sensitive traffic

3.1. Basic system review

Consider a network of L communication links shared by S sources, with routing matrix R . Associated with each source s is a transmission rate x_s . All sources whose flow passes through resource l contribute to the aggregate rate y_l , the rates being added with some forward time delay $\tau_{s,l}^f$:

Table 1
Comparison of this work and [10,11].

Refs. [10,11]	This paper
Focuses on rate-reliability-delay tradeoff	Focuses on equilibrium and dynamic properties
Network with composite links, where each link consists of several sub-links with different rate and reliability	Single sub-link case, i.e., problem (1) with objective (3), which is the case that each link consists of only one sub-link. Special problem structures now enable algorithms with much less overhead
Designs algorithms for the general case of composite links, and the algorithm is complicated for the single sub-link case. It did not reveal the simplified structure and algorithm	Studies the single sub-link case and the special structure allows us to design simplified algorithms compared with [10,11]
Did not study the stability when the communication delays are taken into account in the dynamics	Studies the stability when the communication delays are taken into account in the dynamics and derives the stability conditions
Composite link case studied in [10,11] is not a widely used case	The single sub-link case studied in this paper deserves particular attention as networks with links each being single sub-link are common

Table 2

Comparison of this work and related work.

Paper	Utility function has rate term	Utility function has explicit delay term based on users' application requirements	Studies the stability when the communication delays are taken into account in the dynamics
[6]	Yes	No	n/a
[16,23]	Yes	No	Yes (only delay-dependent stability condition derived)
[10,11]	Yes	Yes	No
This paper	Yes	Yes	Yes (both delay-dependent and delay-independent stability conditions derived)

$$y_l(t) = \sum_i R_{ls} x_s(t - \tau_{s,l}^f). \quad (4)$$

The resources l react to the aggregate rate y_l by setting a price p_l . This is the AQM part of the algorithm. The prices of all the links that source s uses are added to form q_s , the aggregate price for source s , again through a delay $\tau_{s,l}^b$:

$$q_s(t) = \sum_l R_{ls} p_l(t - \tau_{s,l}^b). \quad (5)$$

The prices q_s can then be used to set the rate x_s of source s . This is the TCP part of the algorithm, which completes the picture shown in Fig. 1. The capacity of link l is denoted by c_l . The forward and backward delays can be combined to yield the Round Trip Time (RTT) for source s , τ_s :

$$\tau_s = \tau_{s,l}^f + \tau_{s,l}^b. \quad (6)$$

What needs to be specified now are two control laws that describe how the s th source reacts to the price signal q_s that it sees

$$\dot{x}_s = F_s(x_s, q_s, \tau_s), \quad (7)$$

and how the l th router reacts to the signal y_l it observes

$$\dot{p}_l = G_l(y_l, p_l, c_l). \quad (8)$$

Here F_s models TCP algorithms (e.g., Reno, Vegas) and G_l models AQM algorithms (e.g., RED, REM).

Note that $\sum_{s \in S(l)} x_s$ is equivalent to $\sum_s R_{ls} x_s$, and similarly $\sum_{l \in L(s)} p_l$ is equivalent to $\sum_l R_{ls} p_l$. Throughout the paper, we use both notations interchangeably.

It is well known that the resource allocation algorithm can be reverse-engineered as the solution of an optimization problem [14,22]. We associate with each user s a strictly concave, continuously differentiable non-decreasing utility function $U_s(x_s)$ when being allowed to have a transmission rate x_s . Then the optimization of the whole system with delays $\tau_{s,l}$'s assumed zero can be cast as problem (1). The uniqueness of solution to the above problem is guaranteed since U_s are strictly concave functions.

The optimization problem (1) can be decomposed into a primal problem that the sources are trying to solve and a dual that the links are trying to solve, regarding the sources x_s as primal variables and the prices set by the links p_l as dual variables. The optimal point of the two sub-problems coincides with the optimal point of the original problem, which is unique. More details can be found in [6,14]. The dynamical system defined by (7) and (8) with delays ignored aims to drive the system close to or exactly at the

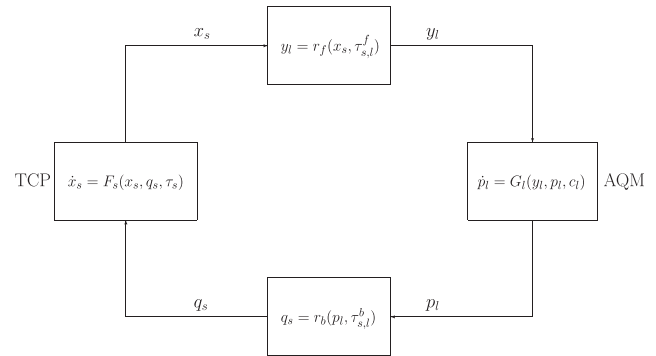


Fig. 1. The Internet as an interconnection of sources and links through delays.

optimal point $(\mathbf{x}^*, \mathbf{p}^*)$, using well-known sub-gradient algorithms.

The heterogeneous delays $\tau_{s,l}^f$'s and $\tau_{s,l}^b$'s are introduced to the system in order to describe the transmission and propagation time needed for the packets to reach the destination and acknowledgements to be received by the source. The presence of such delays is most of the times destabilizing and may affect greatly the performance of the system.

In this section we study the system dynamics ignoring the communication delays $\tau_{s,l}$'s, for delay sensitive traffic where the utility function depends on both rate and queueing delay, then in the next section stability analysis is performed for this system when communication delays are introduced.

3.2. Delay sensitive utility maximization

Extending problem (1) to the system with new utility function (3), the optimization problem becomes

$$\begin{aligned} & \text{maximize} && \sum_s \left[f_s(x_s) - b_s \sum_l R_{ls} d_l(y_l) \right] \\ & \text{subject to} && \sum_s R_{ls} x_s = y_l, \quad \forall l, \\ & \text{variables} && \mathbf{x}, \mathbf{y}, \end{aligned} \quad (9)$$

and with feasible condition that $\mathbf{x} \succeq 0$ and $y_l < c_l, \forall l$. Note that the inequality sign is equivalent to the equality sign in the constraint $\sum_{s=1}^S R_{ls} x_s \leq y_l$ because the objective function is monotonically decreasing in y_l .

Assuming differentiability of the functions involved, the following conditions are necessary for (\mathbf{x}, \mathbf{y}) to be optimal for problem (9) [2],

$$x_s = \arg \max_{x_s} \left[f_s(x_s) - \left(\sum_l R_{ls} p_l \right) x_s \right], \quad (10)$$

$$p_l = \left(\sum_s R_{ls} b_s \right) d'_l(y_l), \quad (11)$$

$$y_l = \arg \min_{y_l} \left[\left(\sum_s R_{ls} b_s \right) d_l(y_l) - p_l y_l \right]. \quad (12)$$

3.3. Analysis of the dynamics

Here we discuss the dynamics ignoring the communication delays τ_s 's. We follow a similar dynamics analysis as in [6]. We will add τ_s 's back to the dynamics in Section 5.

We make the following assumptions,

Assumption 1. The function $f_s(x_s)$ is increasing and strictly concave in $x_s \geq 0$, $\forall s$.

Assumption 2. The function $d_l(y_l)$ is positive, increasing and strictly convex in $y_l \geq 0$, and $d_l(c_l) = \infty$, $\forall l$.

Note that Assumption 2 implies that $y_l \leq c_l$. The function $d_l(y_l)$ is a delay function. For instance, if d_l is the average queueing delay of an $M/M/1$ queue [7], then $d_l(y_l) = \frac{a}{c_l - y_l}$ where $a > 0$ is some constant. With this utility function, problem (9) still decentralizes and has similar properties to the one with $b_s = 0$.

3.3.1. A primal algorithm

Based on conditions (10)–(12), the dynamics can be written as

$$\frac{d}{dt} x_s(t) = \kappa \left(f'_s(x_s(t)) - \sum_l R_{ls} p_l(t) \right), \quad (13)$$

where

$$p_l(t) = \left(\sum_{s \in S(l)} b_s \right) d'_l(y_l(t)), \quad (14)$$

$$y_l(t) = \sum_{s \in S(l)} x_s(t). \quad (15)$$

Theorem 1. Let

$$\mathcal{U}(\mathbf{x}) = \sum_s f_s(x_s) - \sum_s b_s \sum_{l \in L(s)} d_l \left(\sum_{s \in S(l)} x_s \right). \quad (16)$$

The function $\mathcal{U}(\mathbf{x})$ is strictly concave in \mathbf{x} . The unique value \mathbf{x}^* maximizing $\mathcal{U}(\mathbf{x})$ is a stable point of the dynamic system (13)–(15), to which all trajectories converge.

Proof. The assumptions on f_s and d_l for all s and l ensure that $\mathcal{U}(\mathbf{x})$ is strictly concave in $\mathbf{x} \geq 0$ with an interior maximum and the maximal \mathbf{x}^* is thus unique. Setting the derivatives $\frac{\partial}{\partial x_s} \mathcal{U}(x_s) = f'_s(x_s) - \left(\sum_{l \in L(s)} \left(\sum_{s \in S(l)} b_s \right) d'_l \left(\sum_{s \in S(l)} x_s \right) \right)$ to zero identifies the maximum. Further

$$\begin{aligned} \frac{d}{dt} \mathcal{U}(\mathbf{x}(t)) &= \sum_s \frac{\partial \mathcal{U}}{\partial x_s} \cdot \frac{d}{dt} x_s(t) \\ &= \kappa \left[f'_s(x_s(t)) - \sum_l R_{ls} \left(\sum_s R_{ls} b_s \right) d'_l \left(\sum_s R_{ls} x_s(t) \right) \right]^2, \end{aligned}$$

establishing that $\mathcal{U}(\mathbf{x}(t))$ is strictly increasing with t , unless $\mathbf{x}(t) = \mathbf{x}^*$, the unique \mathbf{x} maximizing $\mathcal{U}(\mathbf{x})$. The function $\mathcal{U}(\mathbf{x})$ is thus a Lyapunov function for the dynamic system (13)–(15), and the theorem follows. \square

3.3.2. A dual algorithm

Based on conditions (10)–(12), the dynamics can be written as

$$\dot{p}_l(t) = \kappa_l \left[\sum_s R_{ls} x_s - d_l^{-1} \left(\frac{p_l}{B_l} \right) \right]_{p_l}^+, \quad (17)$$

$$x_s(t) = f_s^{-1}(q_s), \quad (18)$$

where $q_s = \sum_l R_{ls} p_l$, $B_l = \sum_s R_{ls} b_s > 0$, $\kappa_l > 0$, and $[f(x)]_x^+$ means

$$[f(x)]_x^+ = \begin{cases} f(x), & x > 0, \\ \max\{f(x), 0\}, & x = 0. \end{cases}$$

Theorem 2. Let

$$\mathcal{V}(\mathbf{p}) = \sum_s \int_0^{\sum_{l \in L(s)} p_l} f_s^{-1}(\eta_s) d\eta_s - \sum_l \int_0^{p_l} d_l^{-1} \left(\frac{\eta_l}{\sum_s R_{ls} b_s} \right) d\eta_l. \quad (19)$$

Function $\mathcal{V}(\mathbf{p})$ is strictly concave in \mathbf{p} . The unique value λ^* maximizing $\mathcal{V}(\mathbf{p})$ is a stable point of the dynamic system (17) and (18), to which all trajectories converge.

Proof. The assumptions on f_s and d_l for all s and l ensure that $\mathcal{V}(\mathbf{p})$ is strictly concave in $\mathbf{p} \geq 0$ with an interior maximum and the maximal \mathbf{p}^* is thus unique. Setting the derivatives $\frac{\partial}{\partial p_l} \mathcal{V}(p_l) = (\sum_s R_{ls} f_s^{-1}(\sum_l R_{ls} p_l)) - d_l^{-1} \left(\frac{p_l}{\sum_s R_{ls} b_s} \right)$ to zero identifies the maximum. Further

$$\begin{aligned} \frac{d}{dt} \mathcal{V}(\mathbf{p}(t)) &= \sum_s \frac{\partial \mathcal{V}}{\partial p_l} \cdot \frac{d}{dt} p_l(t) \\ &= \kappa \left[\sum_s R_{ls} f_s^{-1} \left(\sum_l R_{ls} p_l(t) \right) - d_l^{-1} \left(\frac{p_l(t)}{\sum_s R_{ls} b_s} \right) \right]^2, \end{aligned}$$

establishing that $\mathcal{V}(\mathbf{p}(t))$ is strictly increasing with t , unless $\mathbf{p}(t) = \mathbf{p}^*$, the unique \mathbf{p} maximizing $\mathcal{V}(\mathbf{p})$. The function $\mathcal{V}(\mathbf{p})$ is thus a Lyapunov function for the dynamic system (17) and (18), and the theorem follows. \square

3.4. Comparison with the dynamics for basic NUM

We compare the dynamics above to the dynamics in [6].

For the primal algorithm, the source dynamics are the same as in [6], but the congestion price here is

$p_l(t) = (\sum_s R_{ls} b_s) d'_l(y_l(t))$ where $y_l(t) = \sum_s R_{ls} x_s(t)$ while the price in [6] is $p_l(t) = p_l(\sum_s R_{ls} x_s(t))$. Hence we find that if the price in [6] were $p_l(y_l) = (\sum_s R_{ls} b_s) d'_l(y_l(t))$ then the algorithm would actually solve problem (9).

For the dual algorithm, the source dynamics are the same as in [6], but the price dynamics here are $\frac{d}{dt} p_l(t) = \kappa \left(\sum_s R_{ls} x_s(t) - d_l^{-1} \left(\frac{p_l(t)}{\sum_s R_{ls} b_s} \right) \right)$, while the price dynamics in [6] is $\frac{d}{dt} p_l(t) = \kappa (\sum_s R_{ls} x_s(t) - h_l(p_l(t)))$. Hence we find that if in [6] the price dynamics were $\frac{d}{dt} \lambda_l(t) = \kappa (\sum_s R_{ls} x_s(t) - h_l(p_l(t)))$ where $h_l(p_l(t)) = d_l^{-1} \left(\frac{p_l(t)}{\sum_s R_{ls} b_s} \right)$ then the algorithm actually would be solving problem (9).

Note that the analysis may also be applicable to the case where the function d_l is in terms of the packet loss due to limited buffer size of the queues.

3.5. The impact

The dynamics developed here can be used for congestion control in a network with users who may have different types of traffic, e.g., traffic with fixed/variable rate, delay sensitive/insensitive, etc. For traffic which is of fixed rate but delay sensitive, like VoIP, the basic NUM does not take that into account. The new framework in this work can accomplish congestion control for mixed traffic in one framework.

To do this, we add a coefficient a_s to f_s , so that we can incorporate the fixed rate traffic, which corresponds to $a_s = 0$. Hence the problem becomes the following,

$$\begin{aligned} & \text{maximize} \quad \sum_s a_s f_s(x_s) - \sum_s b_s \sum_l R_{ls} d_l(y_l) \\ & \text{subject to} \quad \sum_s R_{ls} x_s = y_l \quad \forall l. \end{aligned} \quad (20)$$

We can have the following types of traffic, Traffic A: fixed rate, with delay requirement ($a_s = 0$, $b_s > 0$), like VoIP; Traffic B: variable rate, with delay requirement ($a_s > 0$, $b_s > 0$), like real-time data; Traffic C: variable rate, no delay requirement ($a_s > 0$, $b_s = 0$), like file-downloading.

Note that for the source s with fixed rate traffic only, where $a_s = 0$, we do not need to update x_s . For the source s with delay insensitive traffic only, where $b_s = 0$, we may approximately let b_s be a small positive number, to ensure that $\sum_s R_{ls} b_s$ is positive for all l , so that (14) and (17) are well-defined update equations.

As discussed above, compared with the basic NUM, to get the optima, we only need to change the price dynamics.

4. Applications

In this section, we apply the dynamics to a wireless network with voice and data traffic. Superscripts \sim and $\tilde{\sim}$ are used to indicate voice and data, respectively.

Suppose that priority queueing is used, where voice packets have high priority, and data packets have low priority. Assume that at the source nodes, the arrival processes of voice and data are independent, Poisson, and independent of the service times. Two separate queues

are maintained for voice packets and data packets, respectively.

We consider the nonpreemptive priority rule whereby a packet undergoing service is allowed to complete service without interruption even if a packet of higher priority arrives in the meantime. The case of preemptive priority is similar, and will not be presented here.

Based on the assumption above, we now further specify the delay function in the objective function. By the Kleinrock independence approximation [7], the average delay of a voice packet for source \tilde{s} is [3]

$$\delta_{\tilde{s}}(\tilde{\mathbf{y}}) = \sum_{l \in L(\tilde{s})} \left(\frac{K}{c_l} + \frac{K}{2c_l} \frac{\tilde{y}_l + \tilde{y}_l}{c_l - \tilde{y}_l} \right) \quad (21)$$

and the average delay of a data packet for source \tilde{s} is [3]

$$\delta_{\tilde{s}}(\tilde{\mathbf{y}}) = \sum_{l \in L(\tilde{s})} \left(\frac{K}{c_l} + \frac{K}{2(c_l - \tilde{y}_l)} \frac{\tilde{y}_l + \tilde{y}_l}{c_l - \tilde{y}_l - \tilde{y}_l} \right), \quad (22)$$

where c_l is the capacity of link l , $\tilde{y}_l = \sum_{\tilde{s} \in \tilde{S}(l)} x_{\tilde{s}}$ and $\tilde{y}_l = \sum_{\tilde{s} \in \tilde{S}(l)} x_{\tilde{s}}$ are the total voice and data traffic on link l , respectively, and K is the packet size. Note that for VoIP traffic, $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ are fixed.

Suppose link l has packet error rate v_l . The end-to-end average packet error rate of source s (either \tilde{s} or \tilde{s}) is

$$\eta_s = 1 - \prod_{l \in L(s)} (1 - v_l). \quad (23)$$

For VoIP traffic, we consider the utility function as a function of R -factor. The “ R -factor” formula measures voice quality and it links delay and loss to Mean Opinion Score (MOS) for speech. The relationship between R -factor and voice quality is that as the R -factor increases by 10 from 50 to 100, the voice quality is poor, low, medium, high, and best, respectively [1]. For source \tilde{s} , its R -factor is [1]

$$R_s^{fac} = R_a - \alpha_1 \delta_{\tilde{s}} - \alpha_2 (\delta_{\tilde{s}} - \alpha_3) H - \beta_1 - \beta_2 \log(1 + 100\beta_3 \psi_{\tilde{s}}), \quad (24)$$

where $\delta_{\tilde{s}}$ is the one-way end-to-end delay in milliseconds, $\psi_{\tilde{s}}$ is the packet loss, and the remaining parameters are constants defined as: $R_a = 94.2$, $\alpha_1 = 0.024$, $\alpha_2 = 0.11$, $\alpha_3 = 177.3$, $H = 0$ if delay $\delta < \alpha_3$, $H = 1$ otherwise, and β_1 , β_2 and β_3 are codec dependent parameters, for example, for the G.729 codec: $\beta_1 = 12$, $\beta_2 = 15$, $\beta_3 = 0.6$. Suppose we only consider packet loss due to unreliable wireless link, then $\psi_{\tilde{s}} = \eta_{\tilde{s}}$, where $p_{\tilde{s}}$ is the end-to-end packet loss of source \tilde{s} , then R_s^{fac} is a function of delay $\delta_{\tilde{s}}$.

For the utility function for data source \tilde{s} , we can use the following weighted sum of utility on throughput and utility on delay, with the weight $w_s \in [0, 1]$ indicating the relative importance of throughput and delay [10,11],

$$\begin{aligned} U_{\tilde{s}}(x_s \rho_s, \delta_s) = & w_s \frac{x_s \rho_s - (x_s \rho_s)^{\min}}{(x_s \rho_s)^{\max} - (x_s \rho_s)^{\min}} \\ & - (1 - w_s) \frac{\delta_s - \delta_s^{\min}}{\delta_s^{\max} - \delta_s^{\min}}, \end{aligned} \quad (25)$$

where $\rho_s = 1 - \eta_s$. Different users can have different weights for delay and throughput, and different maximum delay constraints based on their traffic types.

To characterize the tradeoff of voice and data, we optimize the weighted sum of voice utility and data utility [10,11]. The generalized NUM problem is formulated as

$$\begin{aligned}
 & \text{maximize} && \frac{\nu}{|\tilde{S}|} \sum_{\tilde{s}} U_{\tilde{s}}(R_{\tilde{s}}^{fac}) + \frac{1-\nu}{|\tilde{S}|} \sum_{\tilde{s}} U_{\tilde{s}}(x_{\tilde{s}} \rho_{\tilde{s}}, \delta_{\tilde{s}}) \\
 & \text{subject to} && \tilde{y}_l + \tilde{y}_l \leq c_l, \quad \forall l, \\
 & && R_{\tilde{s}}^{fac} \leq R_{\tilde{s}}^{fac}(\delta_{\tilde{s}}), \quad \forall \tilde{s}, \\
 & && \delta_{\tilde{s}}(\tilde{\mathbf{y}}) \leq \delta_{\tilde{s}}, \quad \forall \tilde{s}, \\
 & && \delta_{\tilde{s}}(\tilde{\mathbf{y}}) \leq \delta_{\tilde{s}}, \quad \forall \tilde{s}, \\
 & \text{variables} && \mathbf{R}^{fac}, \tilde{\mathbf{x}}, \tilde{\delta}, \tilde{\delta},
 \end{aligned} \tag{26}$$

and the box constraints on \mathbf{R}^{fac} , $\tilde{\delta}$, $\tilde{\mathbf{x}}$ and $\tilde{\delta}$, where $\nu \in [0, 1]$ is a constant, $R_{\tilde{s}}^{fac}(\delta_{\tilde{s}})$ is as in (24), $\delta_{\tilde{s}}(\tilde{\mathbf{y}})$ and $\delta_{\tilde{s}}(\tilde{\mathbf{y}})$ are as above. The utility function $U_{\tilde{s}}$ is given by (25) and for $U_{\tilde{s}}$, we use $U_{\tilde{s}}(R_{\tilde{s}}^{fac}) = R_{\tilde{s}}^{fac} / (R_{\tilde{s}}^{fac, max} - R_{\tilde{s}}^{fac, min})$.

In problem (26), the first constraint is the rate constraint. The second states that the network must provide an R -factor to each VoIP source that is no less than the requested R -factor. The third and fourth ensure the delay requirements.

Note that we use δ to denote delay as a function and as the value of this function (and similarly for R^{fac}).

By adapting Algorithm 2 in [10,11] for the link composed by several sub-links, we can have a corresponding distributed algorithm for the case of every link having one sub-link (the case in this work). Due to space limitation, we omit the details. The algorithm needs each voice source, data source, and link to solve its optimization problem, and to update either the willingness to pay or the congestion price.

For priority queueing, if the voice traffic is low compared with the link capacity, the average delay is small. When the average delay is less than 177.3 ms, by (24), in R -factor only the term of linear delay $\alpha_1 \delta$ affects the optimal solution. Adapting the dynamics in Section 3, we can explicitly find how to update the congestion price, and how to adjust the rate of the data source. Voice source does not need to solve any optimization problem, and the variables representing the willingness to pay are not needed for voice or data source. The dynamics become simpler than the ones in Algorithm 2 adapted to the single sub-link case in [10,11].

The problem can be simplified to

$$\begin{aligned}
 & \text{maximize} && - \sum_{\tilde{s}} \nu_{0\tilde{s}} \sum_{l \in L(\tilde{s})} \tilde{y}_l + \sum_{\tilde{s}} \nu_{1\tilde{s}} x_{\tilde{s}} \\
 & && - \sum_{\tilde{s}} \nu_{2\tilde{s}} \sum_{l \in L(\tilde{s})} \frac{1}{c_l - \tilde{y}_l - \tilde{y}_l} \\
 & \text{subject to} && \sum_{\tilde{s} \in \tilde{S}(l)} x_{\tilde{s}} = \tilde{y}_l, \quad \forall l, \\
 & \text{variables} && \tilde{\mathbf{x}}, \tilde{\mathbf{y}},
 \end{aligned} \tag{27}$$

where $\nu_{0\tilde{s}}$, $\nu_{1\tilde{s}}$ and $\nu_{2\tilde{s}}$ are corresponding coefficients.

To guarantee the convergence, we add $-\sum_{\tilde{s}} \epsilon x_{\tilde{s}}^2$ in the objective function, to make it strictly concave in $x_{\tilde{s}}$, where ϵ is a positive number, sufficiently small to ensure the

objective function is increasing in $x_{\tilde{s}} \in [x_{\tilde{s}}^{min}, x_{\tilde{s}}^{max}]$, guaranteeing a small approximation error.

We adapt the dynamics of the dual algorithm developed in Section 3, and we propose the following distributed algorithm to find the optimal solution. Note that Eq. (27) is actually a special case of (9), with $f(x_{\tilde{s}}) = \nu_{1\tilde{s}} x_{\tilde{s}} - \sum_{l \in L(\tilde{s})} \sum_{\tilde{s} \in \tilde{S}(l)} \nu_{0\tilde{s}} x_{\tilde{s}}$, $b_{\tilde{s}} = \nu_{2\tilde{s}}$, and $d_l(\tilde{y}_l) = \frac{1}{c_l - \tilde{y}_l - \tilde{y}_l}$. Based on our theory, the dynamics converge. The algorithm we propose can be regarded as the discrete-time version of the dynamics of our theory in previous section. The algorithm can also be derived from optimization theory and tools developed in [2].

Distributed Algorithm 1.

At data source \tilde{s} :

At each iteration t , at data source \tilde{s} , its rate is computed,

$$x_{\tilde{s}}(t) = \min \left\{ x_{\tilde{s}}^{max}, \max \left\{ x_{\tilde{s}}^{min}, \frac{\nu_{1\tilde{s}} - p^{\tilde{s}}(t)}{2\epsilon} \right\} \right\}, \tag{28}$$

where $p^{\tilde{s}}(t) = \sum_{l \in L(\tilde{s})} p_l(t)$ is the end-to-end congestion price at iteration t .

At link l :

At each iteration t , at link l , its congestion price is updated by

$$p_l(t+1) = [p_l(t) - \alpha(t)(\tilde{y}_l(t) - \tilde{x}^l(t))]^+, \quad \forall l, \tag{29}$$

$$\begin{aligned}
 \tilde{y}_l(t) = \min \left\{ \tilde{y}_l^{max}, \max \left\{ \tilde{y}_l^{min}, c_l - \tilde{y}_l - \sqrt{\frac{\sum_{\tilde{s} \in \tilde{S}(l)} \tilde{\nu}_{2\tilde{s}}}{p_l(t) - \sum_{\tilde{s} \in \tilde{S}(l)} \nu_{0\tilde{s}}}} \right\} \right\} \text{ if} \\
 p_l(t) > \sum_{\tilde{s} \in \tilde{S}(l)} \nu_{0\tilde{s}}, \text{ otherwise } \tilde{y}_l(t) = \sum_{\tilde{s} \in \tilde{S}(l)} x_{\tilde{s}}^{min}, \text{ where}
 \end{aligned}$$

$\tilde{x}^l(t) = \sum_{\tilde{s} \in \tilde{S}(l)} x_{\tilde{s}}(t)$ is the aggregate rate of data traffic on link l at iteration t , $\alpha(t)$ is the step size, and $[a]^+ = \max\{a, 0\}$.

The message passing is needed for source to get its end-to-end congestion price $q^{\tilde{s}}(t) = \sum_{l \in L(\tilde{s})} p_l(t)$, and for link to get its aggregate rate $\tilde{x}^l(t) = \sum_{\tilde{s} \in \tilde{S}(l)} x_{\tilde{s}}(t)$, the aggregate coefficients of voice source and data source on it, i.e., $\sum_{\tilde{s} \in \tilde{S}(l)} \nu_{0\tilde{s}}$ and $\sum_{\tilde{s} \in \tilde{S}(l)} \nu_{2\tilde{s}}$.

For the sub-gradient method [2], the step size can be the one satisfying $\lim_{t \rightarrow \infty} \alpha(t) = 0$ and $\lim_{t \rightarrow \infty} \sum_{i=1}^t \alpha(i) = \infty$, e.g., $\alpha(t) = \beta/t$, $\beta > 0$. The algorithm can converge to the optimal solution of problem (27).

5. Models with communication delays

In this section, we will concentrate on the dual congestion control scheme with dynamics at the links but a static source law. The stability properties of the undelayed system (assuming $\tau_{s,l}$'s to be zero) have been obtained in Theorem 2.

We consider communication delays, which capture the transmission and propagation time needed for the packets to reach the destination and acknowledgements to be received by the source. When the communication delays during packet/acknowledgement transfer are introduced, i.e., when $\tau_{s,l}$'s are in the dynamics, the dynamic analysis becomes more complicated, and a scalable analysis methodology is difficult. Here we will obtain two results that

hold for arbitrary network topologies: one delay-independent and the other delay-dependent. The tools we use are based on [16,23].

Note that the delay $d_l(y_l)$ in the utility function for the delay sensitive traffic considered here is mainly the queueing delay, which provides a richer profile for rate control; the delay $\tau_{s,l}$ in our dynamic analysis is the communication delay which includes the queueing delay, propagation delay, processing delay, and so on. For the simplicity of the analysis, $\tau_{s,l}$'s are assumed to be constants, the same assumption as in [16,23]. In future work, we will consider the case where $\tau_{s,l}$'s are time varying.

The routing matrix R is assumed to be fixed and full row rank. This means that there are no algebraic constraints between link flows, i.e., they can vary independently by choice of source flows x_s . As a consequence, equilibrium prices are uniquely determined.

5.1. Dynamics for the model with communication delays

The presence of delays is most of the times destabilizing and may affect greatly the system performance. Stability analysis of linear systems with time delay has been investigated greatly in the past years [15]. Just as in the stability analysis of systems described by linear Ordinary differential equations, there are in general two methodologies for investigating stability: using time-domain (Lyapunov) arguments and using frequency domain arguments. Using a frequency domain methodology more accurate descriptions of the stability boundaries can be obtained and this method is scalable for the special case of Internet congestion control [23]. On the other hand, Lyapunov-based arguments are more conservative; they are however useful for the exact investigation of the stability of nonlinear systems [8]. Here we will analyze the system described in Section 3 using a generalized Nyquist criterion.

The nonlinear delayed model when heterogeneous time-delays are taken into account becomes

$$\begin{aligned}\dot{p}_l(t) &= \kappa_l \left[\sum_{s=1}^S R_{ls} x_s(t - \tau_{s,l}^f) - d_l'^{-1} \left(\frac{p_l(t)}{B_l} \right) \right]_{p_l}^+, \\ x_s(t) &= f_s'^{-1}(q_s(t)), \\ q_s(t) &= \sum_{l=1}^L R_{ls} p_l(t - \tau_{s,l}^b).\end{aligned}$$

We define the following matrices to simplify the notation:

$$F = \text{diag}\{f_s''(x_s^*)\} < 0, \quad D = \text{diag}\{d_l''(y_l^*)\} > 0, \\ K = \text{diag}\{\kappa_l\} > 0, \quad B = \text{diag}\{B_l\} > 0,$$

$$[R_f(\theta)]_{ls} = \begin{cases} e^{-\theta \tau_{s,l}^f}, & \text{if user } s \text{ uses link } l, \\ 0, & \text{otherwise,} \end{cases} \\ [R_b(\theta)]_{ls} = \begin{cases} e^{-\theta \tau_{s,l}^b}, & \text{if user } s \text{ uses link } l, \\ 0, & \text{otherwise,} \end{cases}$$

where θ is the Laplace variable. Note that

$$R_b(\theta) = R_f(-\theta) \text{diag}\{e^{-\theta \tau_s}\}.$$

Linearizing the nonlinear time-delayed system about the equilibrium

$$y_l^* = d_l'^{-1} \left(\frac{p_l^*}{B_l} \right), \quad x_s^* = f_s'^{-1}(q_s^*),$$

given the relations $y_l^* = \sum_s R_{ls} x_s^*$ and $q_s^* = \sum_{l=1}^L R_{ls} p_l^*$, we get:

$$\dot{p}_l(t) = \kappa_l \left[\sum_s R_{ls} \delta x_s(t - \tau_{s,l}^f) - \frac{\delta p_l(t)}{B_l d_l''(y_l^*)} \right], \quad (30)$$

$$\delta x_s(t) = \frac{1}{f_s''(x_s^*)} \delta q_s(t), \quad (31)$$

$$\delta q_s(t) = \sum_{l=1}^L R_{ls} \delta p_l(t - \tau_{s,l}^b). \quad (32)$$

In the following two subsections, we will present two stability results for the linearization, one that is delay-independent and one that is delay-dependent.

5.2. Delay-dependent sufficient condition for stability

Theorem 3. Given Eqs. (30)–(32), for $\kappa_l = \frac{1}{y_l^*}$ and $f_s''(x_s^*) = -\frac{M_s \tau_s}{x_s^* \alpha_s}$ where $\alpha_s < \pi/2$ and $M_s = \sum_{l=1}^L R_{ls}$ the equilibrium is asymptotically stable.

Proof. Taking Laplace transforms and dropping all δ 's:

$$\begin{aligned}\theta p(\theta) - p(0) &= K \left(R_f(\theta) x(\theta) - B^{-1} D^{-1} p(\theta) \right), \\ x(\theta) &= F^{-1} q(\theta), \quad q(\theta) = R_b^T(\theta) p(\theta),\end{aligned}$$

where θ is the variable used in Laplace transforms.

Combining we get:

$$\theta p(\theta) - p(0) = K \left(R_f(\theta) F^{-1} R_b^T(\theta) - B^{-1} D^{-1} \right) p(\theta).$$

Therefore,

$$p(0) = \left(\theta I + K B^{-1} D^{-1} \right) p(\theta) + K R_f(\theta) (-F^{-1}) T(\theta) R_f^T(-\theta) p(\theta).$$

This system is stable if its poles lie in the left half plane, i.e., if the solution to

$$\det \left(\left(\theta + \frac{\kappa_l}{B_l d_l} \right) I + \left(\theta + \frac{\kappa_l}{B_l d_l} \right) G \right) = 0$$

has only negative real parts, where the return ratio is

$$G = \text{diag} \left\{ \frac{\kappa_l}{\theta + \frac{\kappa_l}{B_l d_l}} \right\} R_f(\theta) \text{diag} \left\{ \frac{\tau_s}{-\hat{f}_s} \right\} \text{diag} \left\{ \frac{e^{-\theta \tau_s}}{\tau_s} \right\} R_f^T(-\theta),$$

where $\hat{f}_s = f_s''(x_s^*)$ and $\hat{d}_l = d_l''(y_l^*)$. We now show that if $\hat{f}_s = -\frac{M_s \tau_s}{x_s^* \alpha_s}$ and $\kappa_l = \frac{1}{y_l^*}$, then for $\alpha_s < \pi/2$ the equilibrium is asymptotically stable for arbitrary topologies.

Since the open-loop system is stable, we need to ensure that the eigenvalues of the above function G , for $\theta = j\omega$ do not encircle the -1 point. These eigenvalues are the same as the eigenvalues of

$$\text{diag} \left(\frac{j\omega}{j\omega + \frac{\kappa_l}{B_l d_l}} \right) \hat{R}(j\omega) \text{diag} \left(\frac{\alpha_s e^{-j\omega \tau_s}}{j\omega \tau_s} \right) \hat{R}^T(-j\omega),$$

where

$$\hat{R}(j\omega) = \text{diag} \left\{ \sqrt{\frac{1}{y_l^*}} \right\} R_f(j\omega) \text{diag} \left\{ \sqrt{\frac{x_s^*}{M_i}} \right\}.$$

Note $\bar{\sigma}^2(\hat{R}(j\omega)) \leq 1$ where $\bar{\sigma}^2(\hat{R}(j\omega)) = \rho(\hat{R}(j\omega)\hat{R}^T(-j\omega))$, where $\rho(Z)$ denotes the spectral radius of a matrix Z . The argument is similar as the one in [23].

Now if λ is an eigenvalue of the above, then there exists a v for which $\|v\| = 1$ such that

$$\text{diag} \left(\frac{j\omega}{j\omega + \frac{\kappa_l}{B_l d_l}} \right) \hat{R}(j\omega) \text{diag} \left(\frac{\alpha_s e^{-j\omega\tau_s}}{j\omega\tau_s} \right) \hat{R}^T(-j\omega) v = \lambda v$$

and hence

$$\lambda = \frac{v^* \hat{R}(j\omega) \text{diag} \left(\frac{\alpha_s e^{-j\omega\tau_s}}{j\omega\tau_s} \right) \hat{R}^T(-j\omega) v}{v^* \text{diag} \left(1 - \frac{\kappa_l}{B_l d_l j\omega} \right) v}.$$

Since $\|\hat{R}^T(-j\omega) v\| \leq 1$ this implies $\lambda \in \frac{\text{Co} \left(0 \cup \left\{ \frac{\alpha_s e^{-j\omega\tau_s}}{j\omega\tau_s} \right\} \right)}{\text{Co} \left\{ \left(1 - \frac{\kappa_l}{B_l d_l j\omega} \right) \right\}}$ and

since the eigen-loci cannot cross the real axis at or to the left of the point -1 for $\alpha_s < \pi/2$, the closed loop system is stable by the generalized Nyquist stability criterion [23]. \square

We note that the condition is similar to the one that is given in [16] – however, the dynamics that we are considering are far more interesting, as they take into account delay sensitive traffic.

Remark 1. In general, it may be difficult for the links to be able to estimate y_l^* and for the sources to know M_s . However it is possible to relax the condition that $\kappa_l = \frac{1}{y_l^*}$ to the condition that $\kappa_l = \frac{1}{c_l}$, which can be implemented in a decentralized way. Note that due to the nature of the algorithm, $y_l^* < c_l$, i.e., the links may not be fully utilized.

5.3. Delay-independent sufficient condition for stability

Here we present a delay-independent stability condition.

Theorem 4. Given Eqs. (30)–(32), if $\kappa_l = \frac{1}{y_l^*}$, $f_s''(x_s^*) = -\frac{M_s}{\alpha_s x_s^*}$, $\kappa_l > B_l d_l''(y_l^*)$ and $\alpha_s < 1$ then the equilibrium is asymptotically stable.

Proof. Again, we write $\hat{f}_s = f_s''(x_s^*)$ and $\hat{d}_l = d_l''(y_l^*)$. The return ratio G in this case is the same as in the proof of Theorem 3.

The eigenvalues of G are the same as the ones of

$$\text{diag} \left\{ \frac{1}{\theta + \frac{\kappa_l}{B_l d_l}} \right\} R_f(\theta) \text{diag} \left\{ \frac{1}{-\hat{f}_s} \right\} \text{diag} \{ e^{-\theta\tau_s} \} R_f^T(-\theta) \text{diag} \{ \kappa_l \}.$$

In the same way as before, and imposing that $\hat{f}_s = -\frac{M_s}{x_s^* \alpha_s}$ and that $\kappa_l = \frac{1}{y_l^*}$ we get:

$$\lambda = \frac{v^* \hat{R}(j\omega) \text{diag}(\alpha_s e^{-j\omega\tau_s}) \hat{R}^T(-j\omega) v}{v^* \text{diag} \left(\frac{\kappa_l}{B_l d_l} + j\omega \right) v}.$$

In this case, for asymptotic stability we require that $\kappa_l > B_l \hat{d}_l$ and $\alpha_s < 1$. This condition is delay-independent. \square

Remark 1 also holds in this case. Note that the delay-independent condition allows us to have a fixed gain at the sources which is irrespective of the size of the delay, i.e., the sources do not have to compensate their gain for long delays. The price, of course, is performance degradation.

The delay-independent condition above ensures that the Nyquist plot not only does not encircle the -1 point (which is what we ensured in the delay-dependent condition), but rather that the whole Nyquist plot never leaves the unit disc.

5.4. A simple network

In order to put the above results in perspective, we will consider the stability conditions we get by looking at a simple, single-link single-source network with the ones given by Theorems 3 and 4 when they are reduced to this simple case.

The model for such a simple network reads:

$$\delta \hat{p} = \kappa \left(\frac{1}{\hat{f}} \delta \hat{p}(t - \tau) - \frac{\delta p(t)}{B \hat{d}} \right), \quad (33)$$

where $\hat{f} = f''(x^*)$ and $\hat{d} = d''(y^*)$. This system is delay-

dependent stable [5] if $-\frac{1}{\hat{f}} > \frac{1}{B \hat{d}}$ and $\tau < \frac{\arccos\left(\frac{\hat{f}}{B \hat{d}}\right)}{\kappa \sqrt{\frac{1}{\hat{f}^2} - \frac{1}{B^2 \hat{d}^2}}}$. Theorem 3

requires that $\kappa = \frac{1}{x^*}$, $\hat{f} = -\frac{\tau}{x^* \alpha}$, $\alpha < \pi/2$ and imposes

$\tau < \frac{\arccos\left(-\frac{\tau}{\alpha B \hat{d} x^*}\right)}{\frac{1}{x^*} \sqrt{\frac{2^2 x^{*2}}{\tau^2} - \frac{1}{B^2 \hat{d}^2}}}$. This condition is valid for $\alpha B \hat{d} x^* > \tau$. Let

us introduce a variable $\mu = \frac{\tau}{x^* \alpha B \hat{d}}$, $\mu \in [0, 1]$. Then the above condition reads $\alpha \sqrt{1 - \mu^2} < \arccos(-\mu)$. Indeed this condition is only valid for $\alpha < \frac{\pi}{2}$. Hence the two conditions, the one given in Theorem 3 for arbitrary network sizes and the one specific to (33) are equivalent; therefore the delay-dependent condition is necessary and sufficient for the single-source single-link case.

System is delay-independent stable if $\frac{1}{B \hat{d}} > \frac{1}{-\hat{f}}$. The delay-independent condition that we have in Theorem 4 reduces in the single-source single-link case to $\kappa = \frac{1}{x^*}$, $\hat{f} = -\frac{1}{\alpha x^*}$, $B \hat{d} < \kappa$, $\alpha < 1$. Combining these we get $B \hat{d} < -\alpha \hat{f}$, and since we required $\alpha < 1$ in the conditions of Theorem 4, the two conditions are again equivalent in the simplest network case.

6. Numerical results

6.1. Models without communication delays

In this section, we present numerical examples by analyzing a simple network, shown in Fig. 2. On each stream, there are both VoIP and data traffic. Let the maximum supportable rate on each link be $c_l = 2$ Mbps. Suppose the packet length is $K = 400$ bits. Suppose that every link is

with average packet error rate $\nu = 0.1063$. Note that we assume a fixed packet error rate.

We use Distributed Algorithm 1 to compute the optimal rates for the sources with data traffic by solving problem (26), where in the objective function the weight for the VoIP is $\nu = 0.5$, and the weight for the data throughput is $w = 0.5$. The total data rate of all the users with respect to the iteration is shown in Fig. 3. The convergence can be seen from the figure. The solution of the rates is readily verified to be optimal by comparing with the results from the centralized algorithm.

To see how the data and voice traffic tradeoff when they share the links, and to see how much advantage priority queueing can provide, we plot different curves for the tradeoffs of the throughput of the data traffic and R -factor of the voice traffic in Fig. 4, and the corresponding average packet delay in Fig. 5. In the figures, ν and $(1 - \nu)$ are the weights of the voice and data traffic in the objective function of problem (26), respectively, and w and $(1 - w)$ are the weights of the throughput and the delay of the data traffic, respectively. In Fig. 4, there is only one curve which is for the case where we include the packet loss due to the packet delay exceeding some deadline, as well as due to the unreliable wireless link.

From the figures, it can be seen that the priority queueing can achieve higher R -factor of voice traffic and higher throughput of data traffic, with the sacrifice of the packet delay of data traffic. The packet loss of voice traffic due to exceeding the deadline can affect the R -factor a lot. Priority queueing not only helps the average packet delay of voice traffic, but also it reduces the packet loss due to exceeding the deadline.

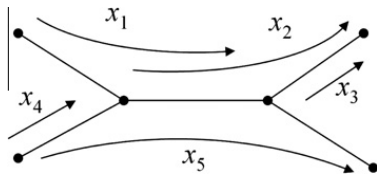


Fig. 2. Network topology and flow routes.

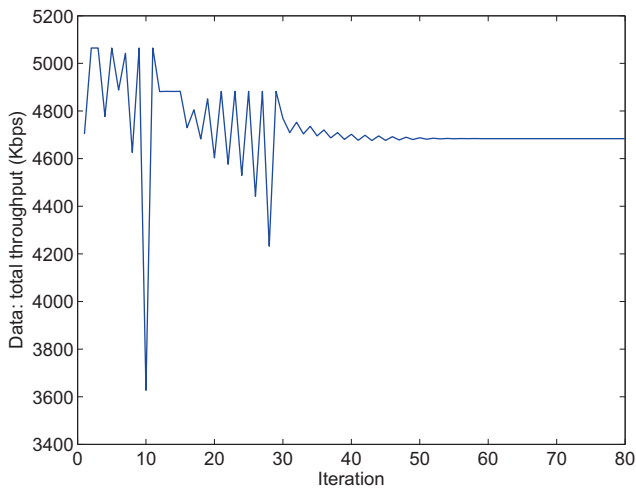


Fig. 3. Convergence of Algorithm 1: Total data rate w.r.t. iterations.

In Fig. 4, it can be seen that no matter how we change the weight of the voice ν , the achieved R -factor by priority queueing is almost the same. So the R -factor achieved is not adaptive. But if we allow some of the voice packets to have high priority and the remaining voice packets to have low priority, then the system can achieve all the possible R -factor from the one achieved by priority queueing to the one achieved by no priority, by adjusting the fraction of the packets with high priority.

In Fig. 5, it can be seen that for the cases of no priority, as the voice weight ν increases, the packet delay decreases, which is because that the higher weight ν indicates the system favors higher R -factor, and to get higher R -factor, the packet delay should be reduced. But for the cases of priority, if it is delay insensitive data traffic, for $w = 1$, the average delay of the data packet is about 1500 ms, which is not shown in the figure; if it is delay sensitive data traffic, for $w = 0.5$, the figure shows that the average packet delay of data or voice almost keeps the same when ν grows. This is because for the priority case, due to higher priority, the

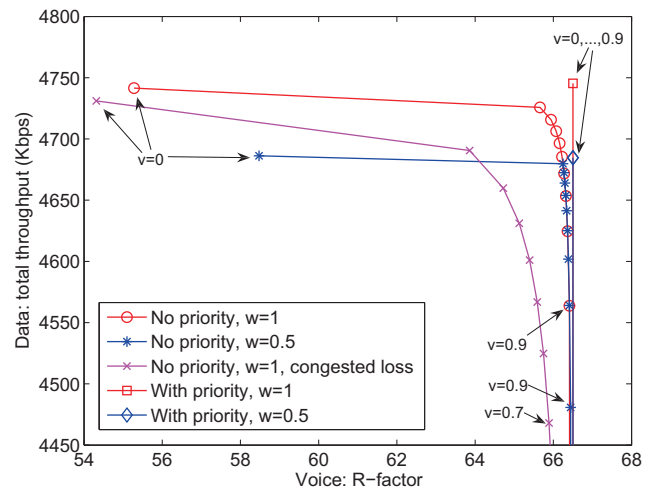


Fig. 4. Tradeoff between voice and data traffic: R -factor and throughput.

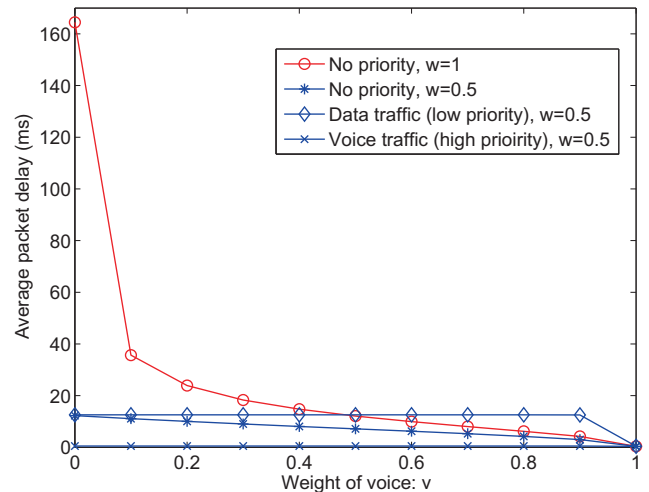


Fig. 5. Average packet delay w.r.t. different weight ν .

voice packet can get very small average packet delay in this example, so it is not sensitive to ν , except when $\nu = 1$.

6.2. Models with communication delays

In order to illustrate our results in the previous section, we will use the same network of five links and five users as shown in Fig. 2 as an example. Assume each flow is with delay sensitive data traffic.

6.2.1. Delay-dependent condition for stability

To satisfy the delay-dependent sufficient condition as in Theorem 3, we let $f_s(x) = \frac{M_s \tau_s}{\alpha_s} x_s (1 - \log(x_s/x_s^{\max}))$, where x_s^{\max} is the maximum allowable rate of source s x_s . Then we have $f_s^{-1}(x_s) = x_s^{\max} \exp(-x_s \alpha_s / (M_s \tau_s))$. Suppose $d_l(y_l) = \frac{a}{c_l - y_l}$, where a is a constant, c_l is the capacity link of link l . We have $d_l^{-1}(z) = c_l - \sqrt{\frac{a}{z}}$.

Set the parameters as $a = 1$, $b_s = 2$, $c_l = 200$, and $x_s^{\max} = c_l$, for each user and each link. Let $\kappa_l = 1/c_l$, $\forall l$. Note that we relax the condition $\kappa_l = 1/y_l^*$ in Theorem 3 for the sake of implementation, as discussed in Remark 1.

Figs. 6 and 7 are for fixed $\alpha = \pi/2 - 0.1$ (i.e., $\alpha < \pi/2$), but different delays τ_s . For Fig. 6, the delays are $\tau_{11}^f = 5$, $\tau_{11}^b = 5$, $\tau_{12}^f = 8$, $\tau_{12}^b = 2$, $\tau_{22}^f = 10$, $\tau_{22}^b = 10$, $\tau_{23}^f = 14$, $\tau_{23}^b = 6$, $\tau_{33}^f = 5$, $\tau_{33}^b = 5$, $\tau_{44}^f = 6$, $\tau_{44}^b = 4$, $\tau_{54}^f = 8$, $\tau_{54}^b = 12$, $\tau_{52}^f = 10$, $\tau_{52}^b = 10$, $\tau_{55}^f = 12$, $\tau_{55}^b = 8$, and for Fig. 7, the delays are doubled. It can be easily checked that the delay-dependent sufficient condition is satisfied. The figures show that the system is stable, and the system arrives at the equilibrium, which has been verified to be the optimum resource allocation. Fig. 7 is with larger delay than Fig. 6. Comparing these two figures, it can be seen that as the delay τ becomes larger, the system gets to the equilibrium much more slowly as expected.

When α becomes larger, the system equilibrium becomes unstable. This can be seen if we compare Figs. 6 and 8, which have the same delays τ , but different α .

6.2.2. Delay-independent condition for stability

Now we turn our attention to the delay-independent conditions. We let $f_s(x) = \frac{M_s}{\alpha_s} x_s (1 - \log(x_s/x_s^{\max})) \forall s$ and $d_l(y_l) = a/(c_l - y_l)$, $\forall l$. We set the parameters as $a = 1$, $b_s = 60,000$, $\alpha_s = 0.99$, $c_l = 80$, and $x_s^{\max} = c_l \forall s, \forall l$. Let $\kappa_l = 1/y_l^* \forall l$. It can be checked that the delay-independent sufficient condition of Theorem 4 holds.

Figs. 9 and 10 show the rates of the five users $x_s(t)$. Fig. 9 is for smaller delays τ_s , and Fig. 10 is for larger delays τ_s . It can be seen that no matter how large the delays are, the system is stable, and it converges to the optimal resource allocation. But the performance of the system becomes worse as the delay is increased and is not as good as the one for the delay-dependent conditions.

6.2.3. Discussion

The conditions given in Theorems 3 and 4, look similar, but produce very different results.

Consider a scenario in which the condition $\kappa_l > B_l d_l''(y_l^*) \forall l$ holds, which is part of the delay-independent conditions as in Theorem 4. In this case, the delay-depen-

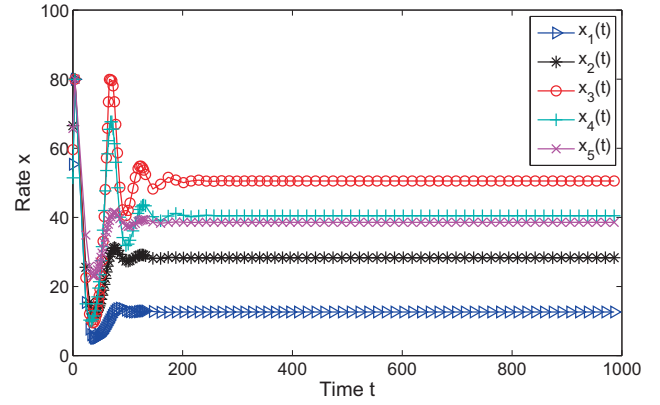


Fig. 6. Stable (under delay-dependent condition): $(\tau_1, \tau_2, \tau_3, \tau_4, \tau_5) = (10, 20, 10, 10, 20)$, $\alpha_s = \pi/2 - 0.1$.

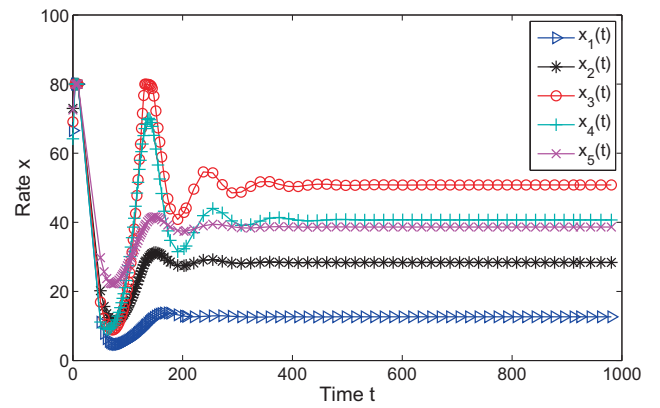


Fig. 7. Stable (under delay-dependent condition): $(\tau_1, \tau_2, \tau_3, \tau_4, \tau_5) = (20, 40, 20, 20, 40)$, $\alpha_s = \pi/2 - 0.1$.

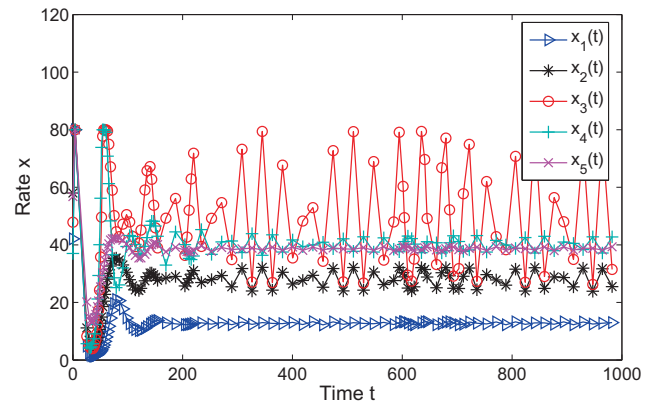


Fig. 8. Unstable (violating delay-dependent condition): $(\tau_1, \tau_2, \tau_3, \tau_4, \tau_5) = (10, 20, 10, 10, 20)$, $\alpha = \pi/2 + 1$.

dent conditions in Theorem 3 can be improved to $\alpha_s/\tau_s < 1$ (no limitation on α_s) instead of the original one $\alpha_s < \pi/2$, which means that stability is retained for even bigger gains α_s . If, however, the condition $\kappa_l > B_l d_l''(y_l^*)$ breaks and $\alpha_s > \pi/2$, then the system will become unstable. Hence caution needs to be exercised when designing algorithms based on delay independent criteria.

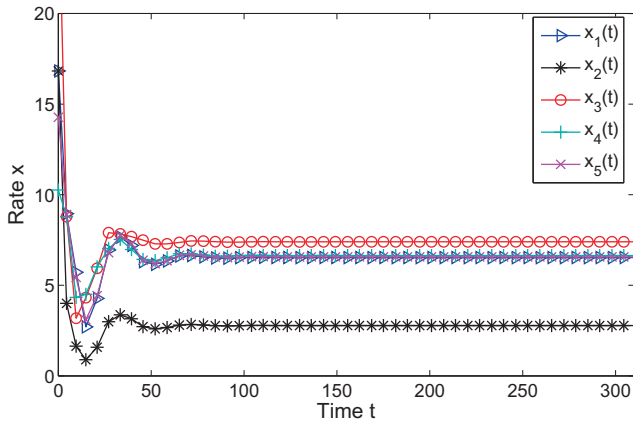


Fig. 9. Stable (under delay-independent condition): $(\tau_1, \tau_2, \tau_3, \tau_4, \tau_5) = (10, 20, 10, 10, 20)$, $\alpha_s = 0.99$.

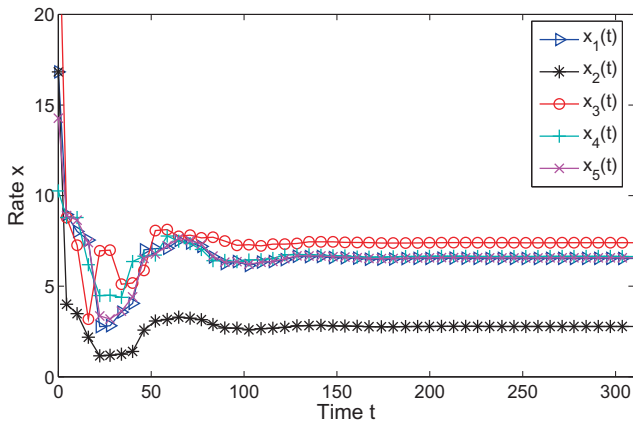


Fig. 10. Stable (under delay-independent condition): $(\tau_1, \tau_2, \tau_3, \tau_4, \tau_5) = (20, 40, 20, 20, 40)$, $\alpha_s = 0.99$.

7. Conclusions

Different applications (delay sensitive/insensitive) have different QoS requirements and hence users have utilities that are functions of both rate and delay. In this paper we considered a new network utility maximization problem with such utility functions, and we solved it in a decentralized way. The resulting link price dynamics are now functions of the delay, reflecting how rate and delay affect user's satisfaction on the QoS.

The results were applied to the system with voice (delay sensitive) and data traffic (delay sensitive/insensitive), and a distributed algorithm was proposed, that allocates the resource such that the utility of voice and data is jointly optimized. In particular, in a network with data and voice traffic with priority queueing, the algorithm can result in higher R -factor of voice traffic and higher throughput of data traffic, sacrificing packet delay of data traffic.

For the dual algorithm, in the presence of heterogeneous propagation delays, delay-independent and delay-dependent stability conditions were derived. This is the first dual algorithm that allows for delay-independent stability. The conditions in Theorems 3 and 4 involve knowl-

edge of y_i^* at the links. This may not be possible, but as remarked in Remark 1, the two conditions can be relaxed to $\kappa_l = \frac{1}{c_l}$, where c_l is the capacity of the link. This relaxation would make the two conditions conservative, but is more easily implementable.

Future work will concentrate on developing a distributed algorithm for the case in which packet loss due to the delay exceeding a fixed deadline is included. Also, the choice of the delay-free utility function f_i at the users is restricted in Theorems 3 and 4, as $f_i''(x_i^*) = \frac{\beta}{x_i^*}$ for some $\beta < 0$ is required, which implies a particular shape of utility function. Future research will concentrate on primal-dual algorithms that would allow the sources to adapt to a utility function of their choice at a much slower time-scale, like the one in [17].

References

- [1] ITU-T Recommendation G.107: The Emodel, A Computational Model for Use in Transmission Planning, 1998.
- [2] D. Bertsekas, Nonlinear Programming, Athena Scientific, 1999.
- [3] D. Bertsekas, R. Gallager, Data Networks, Prentice Hall, 1992.
- [4] M. Chiang, S.H. Low, A.R. Calderbank, J.C. Doyle, Layering as optimization decomposition: a mathematical theory of network architectures, *Proceedings of IEEE* 95 (1) (2007) 255–312.
- [5] J.K. Hale, S.M.V. Lunel, Introduction to Functional Differential Equations, Applied Mathematical Sciences, vol. 99, Springer-Verlag, 1993.
- [6] F. Kelly, A. Maulloo, D. Tan, Rate control in communication networks: shadow prices, proportional fairness and stability, *Journal of the Operational Research Society* 49 (1998) 237–252.
- [7] L. Kleinrock, Queueing Systems, Theory, vol. 1, Wiley, 1976.
- [8] Y. Kuang, Delay Differential Equations with Applications in Population Dynamics, Mathematics in Science and Engineering, vol. 191, Academic Press, 1993.
- [9] Y. Li, M. Chiang, R.A. Calderbank, Congestion control in networks with delay sensitive traffic, in: *Proceedings of IEEE Globecom*, November 07.
- [10] Y. Li, M. Chiang, R.A. Calderbank, S.N. Diggavi, Optimal rate-reliability-delay tradeoff in networks with composite links, in: *Proceedings of IEEE INFOCOM*, May 2007.
- [11] Y. Li, M. Chiang, R.A. Calderbank, S.N. Diggavi, Optimal rate-reliability-delay tradeoff in networks with composite links, *IEEE Transactions on Communications* 57 (5) (2009) 1390–1401.
- [12] Y. Li, A. Papachristodoulou, M. Chiang, Stability of congestion control schemes with delay sensitive traffic, in: *Proceedings of the American Control Conference*, June 2008.
- [13] S.H. Low, A duality model of TCP and queue management algorithms, *IEEE/ACM Transactions on Networking* 11 (4) (2003) 525–536.
- [14] S.H. Low, D.E. Lapsley, Optimization flow control, I: Basic algorithm and convergence, *IEEE/ACM Transactions on Networking* (1999).
- [15] S.-I. Niculescu, Delay Effects on Stability: A Robust Control Approach, Lecture Notes in Control and Information Sciences, vol. 269, Springer, Verlag, 2001.
- [16] F. Paganini, J. Doyle, S. Low, Scalable laws for stable network congestion control, in: *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, FL, 2001.
- [17] F. Paganini, Z. Wang, J.C. Doyle, S.H. Low, Congestion control for high performance, stability and fairness in general networks, *IEEE/ACM Transactions on Networking* 13 (2005) 43–56.
- [18] A. Papachristodoulou, Analysis of nonlinear time delay systems using the sum of squares decomposition, in: *Proceedings of the American Control Conference*, 2004.
- [19] A. Papachristodoulou, L. Li, J.C. Doyle, Methodological frameworks for large-scale network analysis and design, *ACM SIGCOMM Computer Communications Review* (3) (2004) 7–20.
- [20] J. Pongsajapan, S.H. Low, Reverse engineering TCP/IP-like networks using delay-sensitive utility functions, in: *Proceedings of IEEE INFOCOM*, May 2007.
- [21] J.S. Stidham, Pricing and congestion management in a network with heterogeneous users, *IEEE Transactions on Automatic Control* 49 (6) (2004) 976–981.

- [22] R. Srikant, The Mathematics of Internet Congestion Control, Birkhäuser, 2004.
- [23] G. Vinnicombe, On the Stability of End-to-End Congestion Control for the Internet. Technical Report, Cambridge University, CUED/F-INFENG/TR.398, December 2000.
- [24] L. Ying, G. Dullerud, R. Srikant, Global stability of Internet congestion controllers with heterogeneous delays, IEEE/ACM Transactions on Networking 14 (3) (2006) 579–591.



Ying Li received Ph.D. degree in Electrical Engineering from Princeton University, Princeton, NJ, USA, in October 2008 and just joined Samsung Telecommunications America, Dallas, TX, USA. She received the B.E. degree (with honor) and the M.E. degree in Electrical Engineering from Xi'an Jiaotong University, Xi'an, China, in 1997 and 2000, respectively, and the M.A. degree in Electrical Engineering at Princeton University, Princeton, NJ, USA, in 2005. She was a visiting Ph.D. student in Swiss Federal Institute of Technology (EPFL), Switzerland, in summer 2007, and in Motorola Multimedia Research Lab, Schaumburg, IL, USA, in fall 2007, respectively. She worked as a faculty member of research and teaching assistant in Dept. of Information and Communication Engineering at Xi'an Jiaotong University, China, from 2000 to 2003, and as a visiting scholar in Fuji Xerox Co. Ltd., Japan, from 2000 to 2001. Her research interests include communications, networking, optimization, information theory, and signal processing.



Antonis Papachristodoulou received a B.A.(M.A.)/M.Eng. degree in electrical and information sciences from the University of Cambridge, Cambridge, UK, in 2000, as a member of Robinson College. In 2005 he received a Ph.D. in Control and Dynamical Systems, with a minor in Aeronautics, from the California Institute of Technology. In 2005 he held a David Crighton Fellowship at the University of Cambridge and a postdoctoral research associate position at the California Institute of Technology before joining the

Department of Engineering Science at the University of Oxford, Oxford, UK in January 2006, where he is now a University Lecturer in Control Engineering. His research interests include scalable analysis of nonlinear systems using convex optimization based on Sum of Squares programming, analysis and design of large-scale networked control systems with communication constraints and Systems and Synthetic Biology.



Mung Chiang is an Associate Professor of Electrical Engineering, and an Affiliated Faculty of Applied and Computational Mathematics and of Computer Science, at Princeton University. He received the B.S. (Honors) in Electrical Engineering and Mathematics, M.S. and Ph.D. degrees in Electrical Engineering from Stanford University in 1999, 2000, and 2003, respectively, and was an Assistant Professor at Princeton University 2003–2008. His research areas include optimization, distributed control, and stochastic analysis of communication networks, with applications to the internet, wireless networks, broadband access networks, content distribution, and network

economics.

His awards include Presidential Early Career Award for Scientists and Engineers 2008 from the White House, TR35 Young Innovator Award 2007 from Technology Review, Young Investigator Award 2007 from ONR, Young Researcher Award Runner-up in Continuous Optimization 2004–2007 from Mathematical Programming Society, CAREER Award 2005 from NSF, as well as Frontiers of Engineering Symposium participant 2008 from NAE and Engineering Teaching Commendation 2007 from Princeton University. He was a Princeton University Howard B. Wentz Junior Faculty and a Hertz Foundation Fellow. His paper awards include ISI citation Fast Breaking Paper in Computer Science, IEEE GLOBECOM Best Paper three times, and IEEE INFOCOM Best Paper finalist. His guest and associate editorial services include IEEE/ACM Trans. Netw., IEEE Trans. Inform. Theory, IEEE J. Sel. Area Comm., IEEE Trans. Comm., IEEE Trans. Wireless Comm., and J. Optimization and Engineering. He has 2 US patents issued and numerous filed, and co-chaired 38th Conference on Information Sciences and Systems and 9th IEEE WiOpt Conference.



A. Robert Calderbank received the B.Sc. degree in 1975 from Warwick University, England, the M.Sc. degree in 1976 from Oxford University, England, and the Ph.D. degree in 1980 from the California Institute of Technology, all in mathematics.

Dr. Calderbank is Professor of Electrical Engineering and Mathematics at Princeton University where he directs the Program in Applied and Computational Mathematics. Prior to joining Princeton in 2004, he was Vice President for Research at AT&T, responsible

for the first industrial research lab in the world where the primary focus is data at scale. At the start of his career at Bell Labs, innovations by Dr. Calderbank were incorporated in a progression of voiceband modem standards that moved communications practice close to the Shannon limit. Together with Peter Shor and colleagues at AT&T Labs he showed that good quantum error correcting codes exist and developed the group theoretic framework for quantum error correction. He is a co-inventor of space-time codes for wireless communication, where correlation of signals across different transmit antennas is the key to reliable transmission. Dr. Calderbank served as Editor in Chief of the IEEE TRANSACTIONS ON INFORMATION THEORY from 1995 to 1998, and as Associate Editor for Coding Techniques from 1986 to 1989. He was a member of the Board of Governors of the IEEE Information Theory Society from 1991 to 1996 and from 2006 to 2008. Dr. Calderbank was honored by the IEEE Information Theory Prize Paper Award in 1995 for his work on the Z_4 linearity of Kerdock and Preparata Codes (joint with A.R. Hammons Jr., P.V. Kumar, N.J.A. Sloane, and P. Sole), and again in 1999 for the invention of space-time codes (joint with V. Tarokh and N. Seshadri). He received the 2006 IEEE Donald G. Fink Prize Paper Award and the IEEE Millennium Medal, and was elected to the US National Academy of Engineering in 2005.