A Sliding Mode Approach to Traffic Engineering in Computer Networks

Bernardo A. Movsichoff[1], Constantino M. Lagoa[1], and Hao Che[2]

[1] Department of Electrical Engineering. The Pennsylvania State University
bernardo@gandalf.ee.psu.edu, lagoa@engr.psu.edu
[2] Department of Computer Science and Engineering. University of Texas at Arlington
hche@cse.uta.edu

1 Introduction

In this chapter, we address the problem of optimal Traffic Engineering (TE) in computer networks; i.e., determining an optimal traffic allocation in the presence of both multiple paths and multiple Classes of Service (CoSs). More precisely, we aim at designing data rate adaptation laws that maximize the utilization of the network resources (as measured by a given utility function) subject to link capacity constraints and call service requirements.

This problem, as it is formulated here, is an "easy" problem; i.e., the problem of optimal traffic allocation can be stated as maximizing a concave function subject to linear constraints. Hence, if global information is available, one could use well-known algorithms, such as gradient descent, to determine the optimal traffic allocation.

However, in most computer networks, it is not possible to obtain an accurate measure of the network status. Even if this is possible, there would be a delay in the propagation of the information. Furthermore, obtaining such information would require the propagation of large amounts of data in the network, leading to a significant increase in the network traffic and, hence, a decrease on the resources available for user utilization. Given this, the objective of the work presented in this chapter is to develop decentralized adaptation laws that converge to the optimal traffic allocation using the least possible amount of feedback from the network. The algorithms presented here allow for the distribution of the traffic among several paths. Furthermore, they allow for multiple CoSs.

To accomplish these objectives, we use results from nonlinear control theory to design control laws that endow the network with the "right" dynamic behavior. More precisely, by treating calls as *flows*, we develop data rate adaptation algorithms that result in an asymptotically stable system whose equilibrium points are optimal traffic allocations. In other words, we endow the network with the necessary dynamics to "solve" the optimal traffic engineering problem mentioned above.

S. Tarbouriech et al. (Eds.): Advances in Communication Control Networks, LNCIS 308, pp. 111–136, 2004.

Central to the results presented here is Sliding Mode theory and its use in mathematical programming; e.g., see [14]. A first motivation for the use of the approach presented here is the fact that currently used Transport Control Protocol (TCP) congestion control algorithms can be thought of, in their essence, as sliding mode control laws. More precisely, in the TCP congestion avoidance phase, the data rate is increased linearly if no congestion is detected and decreased exponentially when congestion occurs. In other words, one has a congestion control law which is discontinuous along the surface that represents the "congestion barrier". Using tools other than sliding modes, it has been proven that the TCP congestion control law is optimal in the single path case ([2, 11]). However, this work does not extend to the multiple path case.

The realization that currently used adaptation laws can be studied using nonlinear control theory provides the motivation for the approach taken in this chapter. We show that one can use sliding mode theory to develop rate control laws for a class of service enabled network where several paths are available for each call. Moreover, we show that this can be done with minimal feedback from the network.

1.1 Literature Background

There is extensive literature on distributed traffic control. It includes both empirical algorithms (for example, see [4, 5]) and algorithms based on control theory; e.g., see [1, 2, 12]. These algorithms are designed for single path rate adaptation and the approach taken is not optimization-based.

Recently, several methodologies have been proposed which address the optimization-based rate adaptation problem using nonlinear optimization techniques. Their starting point is similar to the one in this chapter; i.e., maximization (minimization) of a utility (cost) function, subject to network resource constraints, where the constraints represent the interaction between different types of traffic; i.e., traffic with different ingress/egress nodes and/or different service requirements.

In the paper by Golestani, et al. [6], instead of using link resource constraints, a link congestion cost is incorporated into the overall utility function. The optimization problem was then solved using a gradient type algorithm. Iterative algorithms were proposed where individual sources periodically adjust their sending rates based on the congestion cost information periodically fed back from each of the links along the flow forwarding paths.

Kelly, et al. [7] use a Lagrangian multiplier technique to solve the optimization problem at hand. This results in a separation between the rate control executed at individual sources and the calculation of a "price" for each link in the network. The rate control at individual sources is based on the "prices" fed back from all the links in the data paths. The computation of the "price" is in turn based on the sending rate information fed back from all the sources using this link. Since only a relaxation of the original optimization problem is solved, it is not proven that the algorithm converges to the optimal traffic allocation. The same formulation is used in the work by La, et al. [8] where the algorithm provided is proven to converge to the optimum rate allocation in the single congested link case (low traffic networks).

Low [10] uses a technique which converts a constrained problem into a non-constrained dual problem. This reformulation results in a similar distributed control scheme to the one presented in [7]. Iterative algorithms were also proposed and their convergence is proven for the single-path case. A similar approach is taken by Sarkar, et al. [13] to address the optimization-based multi-rate multicast. A distributed control scheme is proposed and it degenerates to the one in [10] when there is only one path in the multicast tree.

2 Notation and Assumptions

In our model, the traffic flows are assumed to be described by a fluid flow model and the only resource considered is link bandwidth. In the remainder of this chapter we will use the terms call, session and flow interchangeably.

Consider a computer network where calls with different service requirements are present. We divide these calls into *types*. Types are aggregate of calls that, from the point a view of the traffic engineering algorithm, can be treated as a *unit*; i.e., these are calls with the same ingress and egress nodes and the service requirements are to be applied to the aggregate, not the individual calls. One can have call types with just one call. Moreover, one can have different call types with the same ingress/egress node pair. We assume that each call type can have several paths available. The objective is to find the allocation of the resources that leads to the maximization of a given utility function subject to the network resource constraints and CoS requirements.

More precisely, consider a computer network whose set of links is denoted by $\mathcal{L}$, with cardinality $\text{card}(\mathcal{L})$. Let c_l be the capacity of link $l \in \mathcal{L}$, n be the number of types of calls, n_i be the number of paths available for calls of type i and $\mathcal{L}_{i,j}$ be the set of links used by calls of type i taking path j. Given calls of type i, let $x_{i,j}$ be the total data rate of calls of type i using path j. Also, let

$$\mathbf{x}_i \doteq \left[x_{i,1}, x_{i,2}, \ldots, x_{i,n_i}\right]^T \in \mathbf{R}^{n_i}$$

denote the vector containing the data rates allocated to the different paths taken by calls of type i and

$$\mathbf{x} \doteq \left[\mathbf{x}_1^T, \mathbf{x}_2^T, \ldots, \mathbf{x}_n^T\right]^T \in \mathbf{R}^{n_1+n_2+\cdots+n_n}$$

the vector containing all the data rates allocated to different call types and respective paths. Now, a link $l \in \mathcal{L}$ is said to be congested if the aggregated data rate of the calls using the link reaches its capacity c_l. Given this, define $b_{i,j}(\mathbf{x})$ as the number of congested links along the j-th path of calls of type i. Finally, let $u(x)$ be the unit step function; i.e., $u(x) = 1$ if $x \geq 0$ and $u(x) = 0$ otherwise.

2.1 Classes of Service

Each call in the network is associated to some service requirement or Class of Service (CoS). We assume that five categories of CoSs share the network: Calls of type i,

$i = 1, 2, \ldots, s_1$, are assumed, without loss of generality, to be of Assured Service (AS) CoS category. By AS we mean that a target rate for the call should be guaranteed in average sense. More precisely, assuming that the target rate for $\mathbf{x}_i$ is Λ_i, the objective is to allocate the data rates in such a way that

$$\sum_{j=1}^{n_i} x_{i,j} = \Lambda_i,$$

for all $i = 1, 2, \ldots, s_1$. Calls of type i, $i = s_1 + 1, s_1 + 2, \ldots, s_2$, are assumed to be of the Minimum Rate Guaranteed Service category (MRGS). More precisely, this type of calls should satisfy the following requirement

$$\sum_{j=1}^{n_i} x_{i,j} \geq \theta_i,$$

for some $\theta_i > 0$ and all $i = s_1 + 1, s_1 + 2, \ldots, s_2$.

Calls of type i, $i = s_2 + 1, s_2 + 2, \ldots, s_3$, belong to the Upper Bounded Rate Service (UBRS) category; i.e., there is an upper bound on the maximum bandwidth that can be used. More precisely, traffic should be allocated in such a way that calls of type i satisfy

$$\sum_{j=1}^{n_i} x_{i,j} \leq \Theta_i,$$

for some $\Theta_i > 0$ and all $i = s_2 + 1, s_2 + 2, \ldots, s_3$.

Next, calls of type i, $i = s_3 + 1, s_3 + 2, \ldots, s_4$, are defined to be of the CoS category consisting traffic with both a Minimum Service Guarantee and an Upper Bounded Rate (MRGUBS); i.e., traffic should be allocated so that

$$\theta_i \leq \sum_{j=1}^{n_i} x_{i,j} \leq \Theta_i,$$

for some $\theta_i > 0$, $\Theta_i > 0$ and all $i = s_3 + 1, s_3 + 2, \ldots, s_4$.

Finally, calls of type i, $i = s_4 + 1, s_4 + 2, \ldots, n$ are assumed to be of the Best Effort (BE) class. Calls of this class do not have any service requirements.

3 The Network Optimization Problem

The results in this chapter aim at solving the problem of maximizing utility functions of the form

$$U(\mathbf{x}) \doteq \sum_{i=1}^{n} f_i(\mathbf{x}_i) \doteq \sum_{i=1}^{n} f_i\left(x_{i,1}, x_{i,2}, \ldots, x_{i,n_i}\right),$$

subject to the network constraints and CoS requirements, where $f_i(\cdot)$, $i = 1, 2, \ldots, n$, are differentiable increasing concave functions.

Given this, the problem of optimal resource allocation can be formulated as the following optimization problem:

$$\max_{\mathbf{x}} U(\mathbf{x})$$

subject to the network capacity constraints

$$\sum_{i,j:\ l \in \mathcal{L}_{i,j}} x_{i,j} - c_l \leq 0; \qquad l \in \mathcal{L},$$

the AS requirements

$$\sum_{j=1}^{n_i} x_{i,j} = \Lambda_i; \qquad i = 1, 2, \ldots, s_1,$$

the MRGS requirements

$$\sum_{j=1}^{n_i} x_{i,j} \geq \theta_i; \qquad i = s_1 + 1, s_1 + 2, \ldots, s_2,$$

the UBRS requirements

$$\sum_{j=1}^{n_i} x_{i,j} \leq \Theta_i; \qquad i = s_2 + 1, s_2 + 2, \ldots, s_3,$$

the MRGUBS requirements

$$\theta_i \leq \sum_{j=1}^{n_i} x_{i,j} \leq \Theta_i; \qquad i = s_3 + 1, s_3 + 2, \ldots, s_4$$

and all data rates are non-negative $x_{i,j} \geq 0$, for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n_i$.

Obviously, the optimization problem above is a convex problem; i.e., maximizing a concave function over a convex set. Algorithms such as gradient descent could be used to solve it provided global information is available. However, this is not generally the case. The objective of this work is then to provide decentralized adaptation laws that converge to the solution of the problem stated above.

4 Sliding Mode Control Laws

In this section the proposed solution to the optimization problem above is presented. This solution consists of a family of Sliding Mode control laws that achieve optimal utilization of network resources.

4.1 A Family of Optimal Control Laws

Define the following family of control laws: For $i = 1, 2, \ldots, s_1$ (AS calls), let

$$\dot{x}_{i,j} = z_{i,j}\big(t, \mathbf{x}_i(t), b_{i,j}(t)\big) \left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha b_{i,j}(\mathbf{x}) - \beta_i r_i(\mathbf{x}_i) + \xi_{i,j} u(-x_{i,j}) \right],$$

where

$$r_i(\mathbf{x}_i) = \begin{cases} 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} > \Lambda_i \\ -1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} < \Lambda_i \end{cases}.$$

For $i = s_1+1, s_1+2, \ldots, s_2$ (MRGS calls), let

$$\dot{x}_{i,j} = z_{i,j}\big(t, \mathbf{x}_i(t), b_{i,j}(t)\big)\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha b_{i,j}(\mathbf{x}) + \beta_i^m r_i^m(\mathbf{x}_i) + \xi_{i,j} u(-x_{i,j})\right],$$

where

$$r_i^m(\mathbf{x}_i) = \begin{cases} 0 & \text{if } \sum_{j=1}^{n_i} x_{i,j} > \theta_i \\ 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} < \theta_i \end{cases}.$$

For $i = s_2+1, s_2+2, \ldots, s_3$ (UBRS calls), let

$$\dot{x}_{i,j} = z_{i,j}\big(t, \mathbf{x}_i(t), b_{i,j}(t)\big)\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha b_{i,j}(\mathbf{x}) - \beta_i^M r_i^M(\mathbf{x}_i) + \xi_{i,j} u(-x_{i,j})\right],$$

where

$$r_i^M(\mathbf{x}_i) = \begin{cases} 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} > \Theta_i \\ 0 & \text{if } \sum_{j=1}^{n_i} x_{i,j} < \Theta_i \end{cases}.$$

For $i = s_3+1, s_3+2, \ldots, s_4$ (MRGUBS calls), let

$$\dot{x}_{i,j} = z_{i,j}\big(t, \mathbf{x}_i(t), b_{i,j}(t)\big)\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha b_{i,j}(\mathbf{x}) + \beta_i^m r_i^m(\mathbf{x}_i) - \beta_i^M r_i^M(\mathbf{x}_i) + \xi_{i,j} u(-x_{i,j})\right],$$

where

$$r_i^m(\mathbf{x}_i) = \begin{cases} 0 & \text{if } \sum_{j=1}^{n_i} x_{i,j} > \theta_i \\ 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} < \theta_i \end{cases} \quad \text{and} \quad r_i^M(\mathbf{x}_i) = \begin{cases} 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} > \Theta_i \\ 0 & \text{if } \sum_{j=1}^{n_i} x_{i,j} < \Theta_i \end{cases}.$$

For $i = s_4+1, s_4+2, \ldots, n$ (BE calls), let

$$\dot{x}_{i,j} = z_{i,j}\big(t, \mathbf{x}_i(t), b_{i,j}(t)\big)\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha b_{i,j}(\mathbf{x}) + \xi_{i,j} u(-x_{i,j})\right].$$

In the equations above $z_{i,j}(\cdot)$, α, β_i, β_i^m, β_i^M and $\xi_{i,j}$ are design parameters and $b_{i,j}$ is the number of congested links encountered by calls of type i taking path j.

We now formally establish the optimality of these control laws; i.e., the control laws presented above converge to the optimal traffic allocation for the problem at hand. The proof of this result is presented in Section 10.1.

Theorem 1 (Optimal Control Laws). *Assume that all data rates are bounded; i.e., there exists an* $\rho \in \mathbf{R}$ *such that the data rate vector* $\mathbf{x}$ *always belongs to the set*

$$\mathcal{X} \doteq \left\{ \mathbf{x} \in \mathbf{R}^{n_1+n_2+\cdots+n_n} : x_{i,j} \leq \rho;\ l \in \mathcal{L}_{i,j};\ j = 1,2,\ldots,n_i;\ i = 1,2,\ldots,n \right\}.$$

Also, assume that at the optimal traffic allocation, each congested link has at least one non-binding class of service or a BE call with a nonzero data rate. Furthermore, assume that the components of the gradient of the utility function; i.e., $\nabla U(\mathbf{x})$*, are bounded in* $\mathcal{X}$*.*

Let $\zeta > 0$ *be a given (arbitrarily small) constant and let* $z_{i,j}\big(t, \mathbf{x}_i(t), b_{i,j}(t)\big)$ *be scalar functions continuous in t for all choices of* $\mathbf{x}_i(t) \in \mathcal{X}$ *and* $b_{i,j}(t) \in \{0,1\}$*, satisfying* $z_{i,j}\big(t, \mathbf{x}_i(t), b_{i,j}(t)\big) > \zeta$*, for all* $t > 0$*. Moreover, let* $\alpha > \alpha_{min}$*,* $\beta_i > \beta_{min}$*,* $\beta_i^m > \beta_{min}$*,* $\beta_i^M > \beta_{min}$ *and* $\xi_{i,j} > \xi_{min}$ *be positive constants, with*

$$\alpha_{min} \doteq \max_{i,j,\mathbf{x}\in\mathcal{X}} \frac{\partial U(\mathbf{x})}{\partial x_{i,j}}, \qquad \beta_{min} \doteq \alpha_{min} \max_{i,j} B_{i,j}, \qquad \xi_{i,j,min} \doteq \alpha B_{i,j} + \beta_i,$$

where $\beta \in \{\beta_i, \beta_i^m, \beta_i^M\}$ *and* $B_{i,j}$ *is the number of links used by calls of type i taking path j.*

Then, the control laws presented in Section 4.1 converge to the maximum of the utility function

$$U(\mathbf{x}) \doteq \sum_{i=1}^{n} f_i(\mathbf{x}_i) \doteq \sum_{i=1}^{n} f_i\left(x_{i,1}, x_{i,2}, \ldots, x_{i,n_i}\right), \tag{1}$$

subject to the network link capacity constraints and AS, MRGS, UBRS, MRGUBS and non-negativity requirements.

4.2 A Family of Quasi-Optimal Control Laws

It should be noted that the proposed algorithm might converge to a non-optimal equilibrium if the true number of congested links is not known. Nevertheless, experiments suggest that if a large value of α is used then the algorithm becomes more robust with respect to this loss of information. However, there is a trade-off in doing so, since larger values of α will result in larger oscillations.

To be more precise, let us consider the case of a single path per call type and only BE service being provided. Furthermore, assume that all data rates are bounded away from zero and that the utility function that one is trying to maximize is

$$U(\mathbf{x}) = \sum_{i=1}^{n} f_i(x_i).$$

In the case of a single path, the control law proposed in this work becomes

$$\dot{x}_i = z_i(t,\mathbf{x})\left[d_i(x_i) - \alpha b_i(\mathbf{x})\right],$$

where $d_i(x_i) = df_i/dx_i$. Assume that the only information available is whether the path is congested or not in which case b_i is either zero or one. Then, according to [11], the control law converges to the maximum of the utility function

$$\tilde{U}(\mathbf{x}) = \sum_{i=1}^{n} \int_0^{x_i} \log\left(\frac{\alpha}{\alpha - d_i(u_i)}\right) du_i.$$

This result shows that we do not converge to the desired point. However, if α is large, then the two utility functions are approximately the same.

Quasi-Optimal Laws

Given the results above, a new family of control laws is proposed that only uses less feedback from the network. The only information required is whether a path is congested or not, as opposed to the number of congested links in it. These laws are only quasi-optimal. However, they require much less information and therefore, their implementation is much simpler.

These control laws are similar to the optimal ones and are obtained by replacing the quantity $b_{i,j}$ in the expressions of Section 4.1 by $bin_{i,j}$ defined as

$$bin_{i,j} \doteq \begin{cases} 0 & \text{if no links are congested in the path} \\ 1 & \text{if at least one link in the path is congested} \end{cases}.$$

We now show that this relaxed version of the control laws converges to a small neighborhood of the optimal solution to the problem at hand. This is formally stated through the following theorem whose proof is presented in Section 10.2.

Theorem 2 (Quasi-Optimal Laws). *Assume that the hypothesis in Theorem 1 are satisfied. Furthermore, assume that a similar set of conditions are also satisfied when replacing $b_{i,j}$ by $bin_{i,j}$.*

Then, given any $\varepsilon > 0$ there exists $\alpha^* > \alpha_{min}$ such that for all $\alpha > \alpha^*$ satisfying the conditions above, the control laws in Section 4.2 converge to an ε-neighborhood of $\mathbf{x}^*$, where $\mathbf{x}^*$ achieves the maximum of the utility function (1) subject to the network link capacity constraints, assured service requirements and non-negativity of all the data rates.

5 Reducing Oscillation

The families of control laws presented in Section 4 can lead to excessive oscillation when data optimal data rates are close to zero. The structure of the non-negativity constraints can be exploited to address this issue. More precisely, a simple truncation procedure can be used to reduce oscillation and still preserve optimality.

The following Lemma introduces this improved control laws and establishes their optimality properties. For a proof see the Appendix.

Lemma 1. *Define the following family of control laws: For $i = 1,2,\ldots,s$; i.e., AS calls, let*

$$p_{i,j} = z_{i,j}\big(t,\mathbf{x}_i(t),cg_{i,j}(t)\big)\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha cg_{i,j}(t) - \beta_i(t)r_i\right],$$

where $r_i(t)$ is defined as before. For $i = s+1, s+2,\ldots,n$; i.e., BE calls, let

$$p_{i,j} = z_{i,j}\big(t,\mathbf{x}_i(t),cg_{i,j}(t)\big)\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha cg_{i,j}(t)\right].$$

The family of control laws is then given by

$$\dot{x}_{i,j} = \begin{cases} p_{i,j} & \text{if } x_{i,j} > 0 \\ \max\{0,p_{i,j}\} & \text{if } x_{i,j} = 0 \end{cases}.$$

Then under the same assumptions of Theorems 1 and 2, the control laws above converge to the optimal (respectively the given ε-neighborhood of the optimal) traffic allocation $\mathbf{x}^*$.

Proof. See Section 1.

6 Discrete Time Control Laws and Oscillation Reduction

The implementation of the above control laws in this chapter has to be performed in discrete time in a "real" network. Therefore, we now describe a discrete-time approximation of the continuous-time control laws.

As is the case with any discrete time controller design, there are different ways of obtaining difference state equations from the differential state equations. The approach used here is the forward rule approximation. This method avoids computational delays inherent to other discretization techniques such as trapezoidal or backward rule approximation.

Let $\dot{x}_{i,j} = g_{i,j}(\mathbf{x},t)$ denote the continuous time control law described in Section 4. The discrete approximation that we propose is

$$x^d_{i,j}\big[(k+1)t_d\big] = x^d[kt_d] + t_d g_{i,j}\big(\mathbf{x}(kt_d),kt_d\big);\ k = 0,1,\ldots$$

where t_d is the sampling period. Obviously, since this is not a continuous time law, Sliding Mode theory does not apply. However, one has the following result.

Proposition 1. *Let* $\mathbf{x}(t)$ *be the trajectory obtained using the control laws in Section 4 and let* $\mathbf{x}^d(t)$ *be the corresponding discrete time trajectory obtained using the discretization algorithm above. Define the set* $\mathcal{X}$ *as in Theorem 1.*

Given any time interval $[t_0, t_1]$ *and constant* $\varepsilon > 0$*, there exists a* $\delta > 0$ *such that if* $t_d z_{i,j}(t,\mathbf{x}) < \delta$*, for all* $t > 0$ *and* $\mathbf{x} \in \mathcal{X}$*, then* $\left\|\mathbf{x}(t) - \mathbf{x}^d(t)\right\| < \varepsilon$ *for all* $t \in [t_0, t_1]$.

Proof. Direct application of result 2, page 95 of [3]. □

When implementing the control laws developed in this paper, one is faced with several issues: First, one has to implement a discrete time version of the control algorithms, such as the one described above. Second, usually one uses finite word length which leads to a quantization of the possible data rate values. Finally, there is delay in the propagation of the congestion information. All of these lead to a well known phenomenon: Oscillation. Even in this case, the discretization of the control laws presented in this paper is approximately optimal. We now state the precise result.

Proposition 2. *Let* $\mathbf{x}(t)$ *be the trajectory obtained using the control laws in Section 4 and let* $\mathbf{x}^r(t)$ *be the corresponding discrete time trajectory obtained using the discretization algorithm above and in the presence of delays in the propagation of the congestion information. Let* t_r *be an upper bound on the largest delay. Again, define* $\mathcal{X}$ *as in Theorem 1.*

Given any time interval $[t_0, t_1]$ *and constant* $\varepsilon > 0$*, there exists a* $\delta > 0$ *such that if* $\max\{t_d, t_r\} z_{i,j}(t,\mathbf{x}) < \delta$*, for all* $t > 0$ *and* $\mathbf{x} \in \mathcal{X}$*, then* $\|\mathbf{x}(t) - \mathbf{x}^r(t)\| < \varepsilon$ *for all* $t \in [t_0, t_1]$.

Proof. Direct application of result 2, page 95 of [3]. □

6.1 Adaptive Oscillation Reduction

Although we do have approximate optimality, the performance might degrade if the delays are too large and/or the value of $z_{i,j}(\cdot)$ is too low. Therefore, we now provide a method for choosing the functions $z_{i,j}(\cdot)$ which reduces the amplitude of oscillation of the data rates. The main idea is the following: Oscillation occurs when a link is congested. Hence, when there is no congestion, one would like the rates to increase at a reasonably fast rate. Once congestion is about to occur, one would like to decrease the rate of change in order to reduce the amplitude of the oscillations.

Hence, we propose the following scheme for choosing the value of $z_{i,j}(\cdot)$: Let $T > 0$ be given. Initialize $z_{i,j}(0,\mathbf{x}) = k$, where $k > 0$ is a constant. If congestion is detected at time t_0 for calls of type i taking path j let

$$z_{i,j}(t,\mathbf{x}) = \omega(t - t_0)\,; \quad t_0 \le t < t_0 + T,$$

where $\omega : [0,T] \to [\zeta, k]$ is a decreasing function and ζ is some positive constant. Now, at $t_0 + T$ repeat the same reasoning. If there is no congestion, let $z_{i,j}(t,\mathbf{x}) = k$ until congestion is detected. Once congestion is detected let $z_{i,j}(t,\mathbf{x})$ be equal to a shifted version of $\omega(t)$. Examples of such functions are provided in the next section

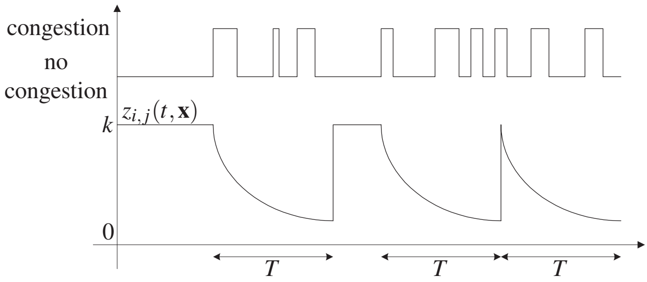


Fig. 1. Example of a scaling function $z_{i,j}(\cdot)$

and the desired behavior is depicted in Fig. 1. Notice that the value $z_{i,j}(\cdot)$ is continuously reset every T seconds. This ensures that the network is able to adapt to changes in the traffic demand. Also, although we reduce the rate of change of $x_{i,j}$ when congestion is detected, the algorithm still converges to the optimum. This is a consequence of the fact that the functions $z_{i,j}(\cdot)$ mentioned above satisfy the conditions of Theorem 1.

Now, this choice for the functions $z_{i,j}(\cdot)$ leaves one with three free parameters that have to be selected. The period T of $z_{i,j}(\cdot)$ should be chosen taking into account the behavior of the demand on the network. More precisely, it should be equal to the interval of time in between substantial changes in the demand. Resetting $z_{i,j}(\cdot)$ in this form will enable the network to more rapidly adapt to the demand. Now, the parameters k and ζ of $z_{i,j}(\cdot)$ should be inversely proportional to the round trip time of the path j of calls of type i, which can be easily estimated. The reasoning behind this particular choice has to do with the fact that the largest delay for the propagation of the congestion information is the round trip time. Now, the exact value of these parameters depends on how one would like the network to behave. If they are high, then the network will quickly react to changes in the demand, but the oscillations will be large and the network might be operating far away from the optimal behavior. If the values k and ζ are low, then one will have small oscillations and a better steady state operating point. However, there will be a much larger transient behavior.

7 Robust Load Sharing

In practice, in order to distribute traffic among the multiple paths available for a given call, the utility function should reflect the fact be it is not important how traffic is distributed among these paths, but only how much traffic of a given type the network can handle. That is, the utility function should be of the form

$$U(\mathbf{x}) = \sum_{i=1}^{n} f_i(x_{i,1} + x_{i,2} + \cdots + x_{i,n_i}).$$

For this kind of utility functions, the control laws that are obtained in Section 4 provide a crucial added benefit: Robustness with respect to failures. In the case of a link failure, the algorithm tries to reroute the traffic to the other available paths and provides an optimal traffic allocation for the new network configuration. In other words, in the case of a link failure, the algorithm provides a systematic (optimal) way of redistributing traffic.

8 Simulation Examples

In this Section, some simulation examples are presented that exemplify the behavior of the control laws presented in this chapter. A brief discussion of the observed behavior and the tradeoffs incurred in the design of the parameters is also presented. These simulations address the case of only AS and BE sharing the network, since the other CoSs do not include any additional complexity from a mathematical standpoint.

The model of the network used here is taken from [8] and is shown in Fig. 2, where also all the links' bandwidths and delays, as well as source and destination nodes are shown. Here, however, several paths and CoSs are available. Overall, $n = 8$ types of calls are considered and the paths available are shown in Table 1. Recall that n_i indicates the number of paths available to calls of type i. Furthermore, calls of types 3 and 5 are assumed to be of AS type with target rates $\Lambda_3 = \Lambda_5 = 1\,\text{Mbps}$.

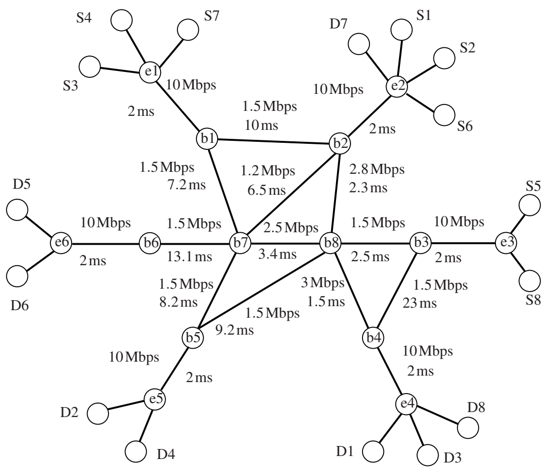


Fig. 2. Topology of the network

The utility function proposed belongs to the family presented in Section 4; i.e.,

$$U(\mathbf{x}) \doteq \sum_{i=1}^{8} \log\left(0.5 + \sum_{j=1}^{n_i} x_{i,j}\right); \qquad \begin{array}{l} i = 1,\ldots,8 \\ j = 1,\ldots,n_i \end{array}.$$

Given the utility function and service requirement above, Theorems 1 and 2 are applied to obtain the following families of control laws: For $i = 1,3$ and $j = 1,\ldots,n_i$;

Table 1. Paths available for each type of calls

type 1	$x_{1,1}: e_2b_2b_8b_4e_4$	type 5	$x_{5,1}: e_3b_3b_8b_7b_6e_6$
$n_1 = 4$	$x_{1,2}: e_2b_2b_8b_3b_4e_4$	$n_5 = 2$	$x_{5,2}: e_3b_3b_4b_8b_5b_7b_6e_6$
	$x_{1,3}: e_2b_2b_7b_8b_3b_4e_4$		
	$x_{1,4}: e_2b_2b_7b_8b_4e_4$		
type 2	$x_{2,1}: e_2b_2b_8b_5e_5$	type 6	$x_{6,1}: e_2b_2b_1b_7b_6e_6$
$n_2 = 3$	$x_{2,2}: e_2b_2b_7b_5e_5$	$n_6 = 3$	$x_{6,2}: e_2b_2b_8b_7b_6e_6$
	$x_{2,3}: e_2b_2b_1b_7b_5e_5$		$x_{6,3}: e_2b_2b_7b_6e_6$
type 3	$x_{3,1}: e_1b_1b_7b_8b_4e_4$	type 7	$x_{7,1}: e_1b_1b_2e_2$
$n_3 = 2$	$x_{3,2}: e_1b_1b_2b_8b_4e_4$	$n_7 = 3$	$x_{7,2}: e_1b_1b_7b_2e_2$
			$x_{7,3}: e_1b_1b_7b_8b_2e_2$
type 4	$x_{4,1}: e_1b_1b_7b_5e_5$	type 8	$x_{8,1}: e_3b_3b_4e_4$
$n_4 = 4$	$x_{4,2}: e_1b_1b_7b_8b_5e_5$	$n_8 = 2$	$x_{8,2}: e_3b_3b_8b_4e_4$
	$x_{4,3}: e_1b_1b_2b_7b_5e_5$		
	$x_{4,4}: e_1b_1b_2b_8b_5e_5$		

i.e., AS calls

$$\dot{x}_{i,j} = z_{i,j}(t,\mathbf{x}) \left[\frac{1}{0.5 + \sum_{j=1}^{n_i} x_{i,j}} - \alpha cg_{i,j}(\mathbf{x}) - \beta_i r_i(\mathbf{x}_i) + \xi_{i,j} u(-x_{i,j}) \right]$$

and for $i = 1,2,4,6,7,8$ and $j = 1,\ldots,n_i$; i.e., BE calls

$$\dot{x}_{i,j} = z_{i,j}(t,\mathbf{x}) \left[\frac{1}{0.5 + \sum_{j=1}^{n_i} x_{i,j}} - \alpha cg_{i,j}(\mathbf{x}) - \xi_{i,j} u(-x_{i,j}) \right],$$

where $cg_{i,j}$ denotes the congestion information used; i.e., $b_{i,j}$ for the optimal control laws and $bin_{i,j}$ for the quasi-optimal ones. The design parameters α, β_i and $\xi_{i,j}$ and the functions $z_{i,j}(t,\mathbf{x})$ are chosen to satisfy the convergence conditions set forth by Theorem 1. It is clear from the last equation, that the term 0.5 in the denominator imposes an upper bound on the derivatives, that otherwise will tend to infinity as all the data rates go to zero.

8.1 Ideal Conditions

As a first step and in order to have a starting point for comparison, the optimal control laws are simulated for an almost ideal case where one has very small delays and sampling intervals. The discrete time version of the control laws are obtained using a sampling interval of $t_d = 0.1$ ms, while all delays were chosen equal to 0.1 ms. The remaining parameters were taken as $\alpha = 4$, $\beta_3 = \beta_5 = 22$ and $\xi_{i,j} = B_{i,j}\alpha + \beta_i + 0.0001$, while the oscillation reducing function was kept constant at 0.375.

Fig. 3 shows that, under these ideal conditions, the utility function converges to the optimal value. The plots of the data rates show a clear sliding motion and demonstrate that the AS requirements are being met.

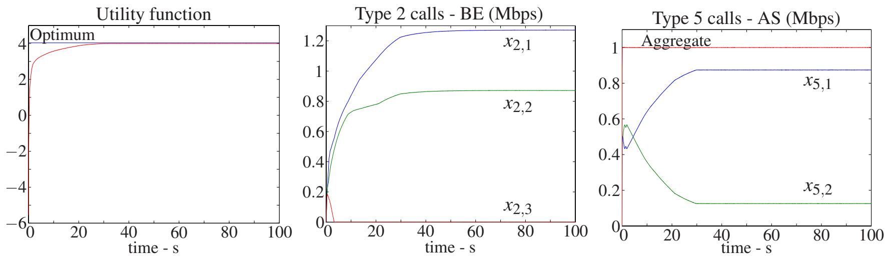
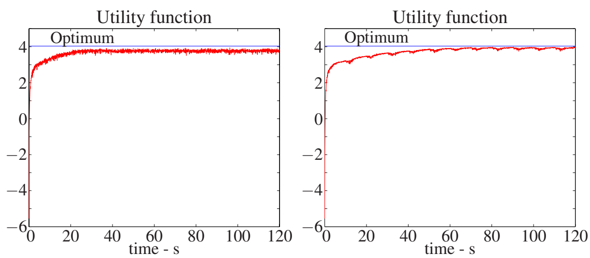


Fig. 3. Ideal Conditions: small delays and t_d

8.2 Non-ideal Conditions

The existence of delays in the propagation of information as well as the discretization of the continuous time laws introduce undesired oscillations in the trajectories of the data rates. This phenomenon can be explained using Sliding Modes theory. Indeed, Sliding Modes can be derived as a limiting procedure of the motion in a boundary layer around the discontinuity surfaces. These boundary layers can occur due to non-ideal switching caused by delays, hysteresis, etc. It is then only natural to observe an oscillatory behavior in this case. Moreover, the larger the magnitude of the derivatives, the larger the boundary layer will be. Hence, the effects of α and the parameters of the oscillation reduction function k, ζ and T can all be explained in a similar way and the simulations all show the same type of behavior. For more thorough simulation examples than the ones presented in this chapter, the reader is referred to [9].

Fig. 4 shows the effect of larger delays and sampling interval on the utility function for both the cases of constant and time-varying $z_{i,j}$. It can be seen that under



(a) Constant $z_{i,j}(t,\mathbf{x})$ (b) Time varying $z_{i,j}(t,\mathbf{x})$

Fig. 4. Oscillation reduction on the utility functions for larger delays and t_d

these conditions the control laws converge to a value close to the optimal one and not to the optimal itself. Moreover, with a constant $z_{i,j}$ (Fig. 4(a)) there is excessive oscillation as opposed to the case of varying $z_{i,j}$ (Fig. 4(b)) where oscillations are milder.

However, the use of this oscillation reduction scheme introduces further undesired effects. Namely, the speed of convergence may be reduced and, as a consequence, a slower reaction to changing conditions can result. On the other hand, by having milder oscillations the network spends less time working above link capacity with the effect that less packets are dropped and fewer retransmissions are needed. Finally, Fig. 5 shows the trajectories obtained for the case of varying $z_{i,j}$, where the AS requirements are met.

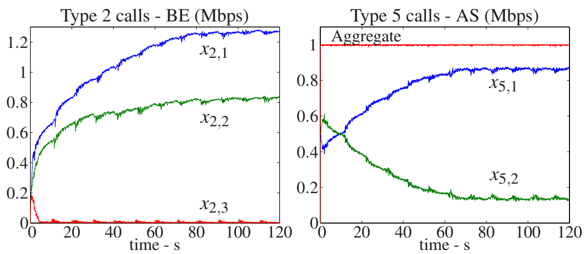


Fig. 5. Data rates for time-varying $z_{i,j}(t,\mathbf{x})$ with larger delays and t_d

8.3 Robustness

One of the most important features of the control laws presented in this chapter is the following: They endow the network with robustness against link and/or node failures. We claim that since the data rates are being updated adaptively, as soon as link failure is detected, the control law will reroute all the traffic away from the paths using this link. The control laws running at the sending nodes can handle this situation by treating link failure as congestion; i.e., the adaptation laws are oblivious to the failure. Instead, they simply reduce the sending rates because congestion is detected along the paths that include the broken link.

In order to test this feature, the link connecting nodes b_7-b_8 was opened at time $t = 120$ s. This link is particularly problematic because both type 3 and 5 (the AS calls) lose one of the two paths available to them. Furthermore, the path that remains for type 5 shares links with almost every other type, making it necessary for the other call types to also reroute some of their traffic and reduce their aggregated data rates. Finally, all the variables involved were set to their nominal values; i.e., $\alpha = 4$, $\beta_3 = \beta_5 = 22$, $T = 10$ s and delays from Fig. 2.

The simulations in Fig. 6 show that the control laws excel at this task: The network reacts almost immediately to the failure. Furthermore, the utility function converges to its new optimal value, while satisfying the AS requirements. The link failure forces the network to reroute the AS traffic to the available paths. This means transferring the AS traffic to alternative paths that were almost unused by these types of traffic. This, in turn, forces the network to substantially reduce the resources allocated to BE traffic. For example, in Fig. 6 we see that data rates of BE traffic of type 2 had to be greatly reduced to make sure that AS traffic requirements were met. In

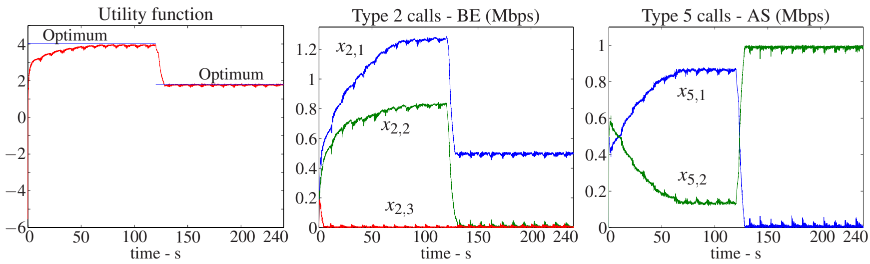


Fig. 6. Response to a link failure

other words, the control laws presented here endow the network with the capacity to quickly react to failures, always enforcing the AS requirements and distributing both the AS traffic and BE traffic among the available paths in such a way as to maximize the utilization of the network resources.

8.4 Quasi-optimal Control Laws

In this Section, simulations of the quasi-optimal control laws are presented. These laws become more relevant in light of the fact that under non-ideal conditions, the optimal laws converge only to a neighborhood of the optimal; i.e., the optimality obtained by using more information is lost but the complexity is not.

These quasi-optimal laws are also used to show an improved version that exploits the form of the non-negativity constraints on the data rates to further reduce oscillations around zero. Indeed, by removing the term $\xi_{i,j}$ in the adaptation laws and truncating the data rates, these exhibit much less oscillation. It can be seen in Fig. 7, that both control laws (with and without ξ) converge to a value close to the optimal one. However, the ones without ξ do so with a less oscillatory behavior.

If one compares the behavior of these control laws against the optimal ones in the previous sections, one sees that they provide comparable performance. The quasi-optimal ones however, require much less information that allows for a much simpler implementation.

9 Conclusions

In this chapter, a unified framework for traffic engineering is presented. The approach presented enables one to address the problem of rate adaptation and load balancing in computer networks. Moreover, the algorithms presented can be applied to the case where several CoS are to be provided to network users and several paths are available between each pair of source/destination nodes. Furthermore, the algorithms endow the network with the ability of optimally adjust to changing conditions such as link or node failures. The issue of oscillation mitigation is also addressed. Namely, an adaptive scheme is provided that reduces oscillation while keeping the ability to react

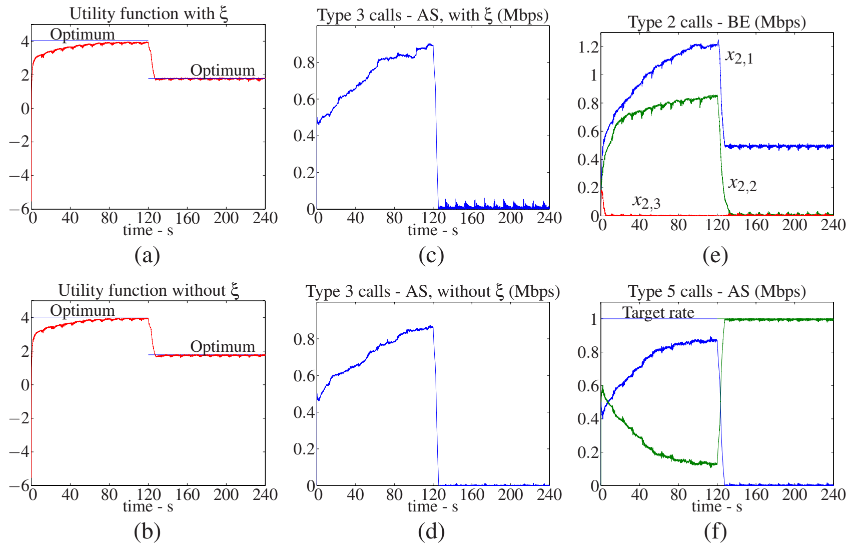


Fig. 7. (a,b) Utility function, (c,d) Oscillation of data rates close to zero, (e) Example of Best Effort calls without ξ, (f) Assured service calls without ξ.

to changes in demand and/or network configuration. To the best of our knowledge, this is the first comprehensive solution to this problem.

Effort is now being put on the implementation and further development of rate control laws. The proposed laws have several free parameters and different choices can lead to very different transient behaviors. Hence, we are now developing rules for determining these parameters when one is to deploy the proposed algorithms in a "real" network. Furthermore, we aim at modifying the rate adaptation laws so that one achieves optimal traffic allocation with less feedback from the network and, hence, with less traffic overhead.

10 Proof of the Main Results

In this Section, the proof of the main results in this chapter are presented but first some preliminaries, including additional notation is presented, that will enable us to present them in a more intuitive way.

The problem at hand can be represented in the following standard form

$$\max_{\mathbf{x}} U(\mathbf{x})$$

subject to the inequality and equality constraints of the form

$$h_k(\mathbf{x}) \le 0; \;\; k = 1, 2, \ldots, m \qquad \text{and} \qquad h_k(\mathbf{x}) = 0; \;\; k = m+1, \ldots, L,$$

where $U(\mathbf{x})$ is a concave differentiable increasing function and $h_k(\mathbf{x})$ are affine functions for all k. The inequality constraints correspond to the link capacity constraints, the restriction that the data rates have to be nonnegative and MRGS, UBRS and MRGUBS CoS constraints. On the other hand, the equality constraints correspond to the AS requirements. Now, define the matrix

$$Z(t,\mathbf{x}) \doteq \mathrm{diag}\big(z_{1,1}(t,\mathbf{x}),\ldots,z_{1,n_1}(t,\mathbf{x}),\ldots,z_{n,1}(t,\mathbf{x}),\ldots,z_{n,n_n}(t,\mathbf{x})\big).$$

Note that, given the special form of the constraints it is easy to see that the adaptation laws presented in Section 4.1 can be represented in the following form

$$\dot{\mathbf{x}} = Z(t,\mathbf{x})\left[\nabla U(\mathbf{x}) - H(\mathbf{x})\,\mathbf{v}(\mathbf{x})\right],$$

where $\nabla U(\cdot)$ denotes the gradient of the function $U(\cdot)$, $H(\cdot)$ is the following matrix

$$H(\cdot) \doteq \left[\nabla h_1(\cdot)\ \nabla h_2(\cdot)\ \cdots\ \nabla h_L(\cdot)\right]$$

and $\mathbf{v}(\cdot) = \left[v_1(\cdot), v_2(\cdot),\ldots,v_L(\cdot)\right]^T$ is an L-dimensional vector whose entries are of the form

$$v_k(\mathbf{x}) = \begin{cases} \gamma_k & \text{if } h_k(\mathbf{x}) > 0 \\ 0 & \text{if } h_k(\mathbf{x}) < 0, \end{cases}$$

for $k = 1,2,\ldots,m$, where $\gamma_k = \alpha$ for the constraints associated with link capacity; i.e., $k = 1,2,\ldots,\mathrm{card}(\mathcal{L})$, $\gamma_k = \xi_{i,j}$ for the constraints associated with non-negativity of $x_{i,j}$, $\gamma_k = \beta_i^M$ for the maximum allowed rate constraints on calls of type i and $\gamma_i = \beta_i^m$ for the minimum service guarantee constraints on calls of type i. Also,

$$v_k(\mathbf{x}) = \begin{cases} \xi_k & \text{if } h_k(\mathbf{x}) > 0 \\ -\xi_k & \text{if } h_k(\mathbf{x}) < 0 \end{cases}$$

for $k = m+1, m+2,\ldots,L$, where $\xi_k = \beta_i$ for the AS constraint k associated with calls of type i. Also, let the admissible domain be the set

$$\mathcal{D} \doteq \left\{\mathbf{x} \in \mathbf{R}^{\sum_{i=1}^n n_i} : h_k(\mathbf{x}) \le 0 \text{ for } k = 1,2,\ldots,m; h_k(\mathbf{x}) = 0 \text{ for } k = m+1,\ldots,L\right\}.$$

Essentially, the proof requires 4 steps. First, we prove that the adaptation law converges to the maximum of the function

$$\widehat{U}(\mathbf{x}) \doteq U(\mathbf{x}) - \Xi(\mathbf{x}),$$

where $\Xi(\mathbf{x}) \doteq \left[h_1(\mathbf{x}), h_2(\mathbf{x}),\ldots,h_L(\mathbf{x})\right]\mathbf{v}(\mathbf{x})$. Second, we provide necessary and sufficient conditions for the maximum of $\widehat{U}(\mathbf{x})$ to coincide with the maximum of $U(\mathbf{x})$. Third, we show that under this conditions this procedure converges to the solution of the optimization problem at hand. The final step is the realization that, under the conditions of Theorem 1, the necessary and sufficient conditions mentioned above are satisfied.

Lemma 2. *The function $\widehat{U}(\mathbf{x})$ does not decrease along the trajectories.*

Proof. If a sliding mode does not occur then

$$\frac{d\widehat{U}}{dt} = \left[\nabla U - H(\mathbf{x})\mathbf{v}(\mathbf{x})\right]^T \dot{x} = \left[\nabla U - H(\mathbf{x})\mathbf{v}(\mathbf{x})\right]^T Z(t,\mathbf{x})\left[\nabla U - H(\mathbf{x})\mathbf{v}(\mathbf{x})\right] \geq 0$$

since the matrix $Z(t,\mathbf{x})$ is positive definite.

Now, assume that a sliding mode occurs in the intersection of the surfaces $h_k(\mathbf{x}) = 0$, $k \in \mathcal{I}$. Let $H_1(\mathbf{x})$ be the matrix whose columns are $\nabla h_k(\mathbf{x})$ for $k \in \mathcal{I}$ (and in the same order as in $H(\mathbf{x})$). Also, let $H_2(\mathbf{x})$ be the matrix with columns $\nabla h_k(\mathbf{x})$ for $k \notin \mathcal{I}$ (again in the same order as in $H(\mathbf{x})$). Then, given that a sliding mode occurs in the intersection of the surfaces $h_k(\mathbf{x}) = 0$, $k \in \mathcal{I}$, we have

$$H_1(\mathbf{x})^T Z(t,\mathbf{x})\left[\nabla U - H_1(\mathbf{x})\mathbf{v}_1(\mathbf{x}) - H_2(\mathbf{x})\mathbf{v}_2(\mathbf{x})\right] = 0$$

where $\mathbf{v}_1(\mathbf{x})$ is the vector containing $v_k(\mathbf{x})$, for $k \in \mathcal{I}$ and $\mathbf{v}_2(\mathbf{x})$ is the vector containing $v_k(\mathbf{x})$, for $k \notin \mathcal{I}$. Now, assume that $\det\left[H_1(\mathbf{x})^T Z(t,\mathbf{x}) H_1(\mathbf{x})\right] \neq 0$ (a reasoning similar to the one in [14] can be done to address the case where this does not happen). From now on, to simplify the exposition, we drop the dependency on $\mathbf{x}$. Then, the equivalent control is

$$\mathbf{v}_{1,\mathrm{eq}} = \left(H_1^T Z H_1\right)^{-1}\left(H_1^T Z \nabla U - H_1^T Z H_2 \mathbf{v}_2\right)$$

and replacing in the expression for $\dot{\mathbf{x}}$, the resulting sliding motion is

$$\dot{\mathbf{x}} = \sqrt{Z} P \sqrt{Z}\left(\nabla U - H_2 \mathbf{v}_2\right),$$

where $\sqrt{Z}$ is well defined since $Z > 0$ and P is given by

$$P \doteq I - \sqrt{Z} H_1 \left(H_1^T \sqrt{Z}\sqrt{Z} H_1\right)^{-1} H_1^T \sqrt{Z}.$$

Now, let Ξ_1 be the elements h_k of Ξ with $k \in \mathcal{I}$ (in the same order as in Ξ). Also, let Ξ_2 be the elements h_k of Ξ with $k \notin \mathcal{I}$ (again in the same order as in Ξ). Since a sliding mode occurs, during this motion we have

$$\widehat{U} = U - \Xi_2 \mathbf{v}_2.$$

Now, since U and Ξ_2 are continuously differentiable and along this sliding motion $\mathbf{v}_2$ is constant, we have

$$\frac{d\widehat{U}}{dt} = \left(\nabla U - H_2 \mathbf{v}_2\right)^T \dot{\mathbf{x}}.$$

Now, notice that $P = P^T = P^2$. Hence,

$$\frac{d\widehat{U}}{dt} = \left(\nabla U - H_2 \mathbf{v}_2\right)^T \sqrt{Z} P \sqrt{Z}\left(\nabla U - H_2 \mathbf{v}_2\right) = \left\|P\sqrt{Z}\left(\nabla U - H_2 \mathbf{v}_2\right)\right\|^2 \geq 0.$$

□

Lemma 3. *The time derivative of* $\widehat{U}$ *is zero only when* $\dot{\mathbf{x}} = 0$.

Proof. If a sliding mode does not occur we have $d\widehat{U}/dt = \left[\nabla U - H\mathbf{v}\right]^T Z\left[\nabla U - H\mathbf{v}\right]$ and since Z is positive definite

$$\frac{d\widehat{U}}{dt} = 0 \Rightarrow \nabla U - H\mathbf{v} = 0 \Rightarrow Z\left[\nabla U - H\mathbf{v}\right] = 0 \Rightarrow \dot{\mathbf{x}} = 0.$$

Now assume that a sliding mode occurs in the intersection of the surfaces $h_k(\mathbf{x}) = 0$, $k \in \mathcal{I}$. In this case,

$$\frac{d\widehat{U}}{dt} = \left\|P\sqrt{Z}\left(\nabla U - H_2\mathbf{v}_2\right)\right\|^2.$$

Hence,

$$\frac{d\widehat{U}}{dt} = 0 \Rightarrow P\sqrt{Z}\left(\nabla U - H_2\mathbf{v}_2\right) = 0 \Rightarrow \sqrt{Z}P\sqrt{Z}\left(\nabla U - H_2\mathbf{v}_2\right) = 0 \Rightarrow \dot{\mathbf{x}} = 0.$$

□

Lemma 4. *The stationary points of $\widehat{U}$ are the maximum points of $\widehat{U}$.*

Proof. Let $\mathbf{x}_0$ be a stationary point on the intersection of surfaces given by $H_1 = 0$. In this case, we have

$$\nabla U(\mathbf{x}_0) - H_1(\mathbf{x}_0)\mathbf{v}_{1,\text{eq}}(\mathbf{x}_0) - H_2(\mathbf{x}_0)\mathbf{v}_2(\mathbf{x}_0) = 0.$$

Now, consider the function $\widehat{U}^*(\mathbf{x}) = U(\mathbf{x}) - H_1(\mathbf{x})\mathbf{v}_{1,\text{eq}}(\mathbf{x}_0) - H_2(\mathbf{x})\mathbf{v}_2(\mathbf{x})$. Given that the components of the equivalent control satisfy

$$\begin{aligned} 0 \le v_{1,\text{eq},k}(\mathbf{x}_0) \le \gamma_k \qquad & \text{for } 1 \le k \le m \text{ and} \\ -\xi_k \le v_{1,\text{eq},k}(\mathbf{x}_0) \le \xi_k \qquad & \text{for } m < k \le L, \end{aligned}$$

it holds that $H_1(\mathbf{x})\mathbf{v}_{1,\text{eq}}(\mathbf{x}_0) \le H_1(\mathbf{x})\mathbf{v}_1(\mathbf{x})$ and, as a consequence $\widehat{U}^*(\mathbf{x}) \ge \widehat{U}(\mathbf{x})$. Now, since U is a concave function, h_k are convex functions for $1 \le k \le m$ and h_k are linear functions for $m+1 \le k \le L$ then $\widehat{U}^*$ is a concave function and hence it has a unique maximum. Furthermore, $\widehat{U}^*$ is continuously differentiable and

$$\nabla\widehat{U}^*(\mathbf{x}_0) = 0.$$

Therefore, $\max_{\mathbf{x}} \widehat{U}^*(\mathbf{x}) = \widehat{U}^*(\mathbf{x}_0)$. Now, since $\widehat{U}^*(\mathbf{x}) \ge \widehat{U}(\mathbf{x})$ and $\widehat{U}^*(\mathbf{x}_0) = \widehat{U}(\mathbf{x}_0)$, we conclude that $\widehat{U}(\mathbf{x})$ reaches its maximum at $\mathbf{x}_0$. Therefore, any stationary point of the optimization procedure is a maximum of $\widehat{U}(\mathbf{x})$. Now, assume that a maximum point $\mathbf{x}^*$ of $\widehat{U}(\mathbf{x})$ is not a stationary point. Then, we have $d\widehat{U}(\mathbf{x}^*)/dt > 0$ and so $\widehat{U}$ will increase along the trajectory which contradicts the fact that $\mathbf{x}^*$ is a maximum point of $\widehat{U}(\mathbf{x})$. □

Lemma 5. *If the set of all maximum points is bounded (which is our case) then* $\mathbf{x}$ *will converge to this set from any initial condition.*

Proof. The proof is very similar to the one in [14], Chapter 15, Section 3. It makes use of the results from Lemmas 2, 3 and 4. Therefore, we refer the reader to it. □

Theorem 3. *Let* $\mathbf{v}^0$ *be a vector whose entries are of the form*

$$\begin{aligned} 0 \le v_k \le \gamma_k\,; &\quad k = 1,2,\ldots,m \\ -\xi_k \le v_k \le \xi_k\,; &\quad k = m+1, m+2, \ldots, L, \end{aligned}$$

where $v_k = 0$ *for non-binding constraints. Then, the maximum of* $\widehat{U}(\mathbf{x})$ *coincides with the optimal* $U(\mathbf{x}^*)$ *if and only if there exists* $\mathbf{x}^*$ *such that* $\nabla U(\mathbf{x}^*) = H(\mathbf{x}^*)\mathbf{v}^0$.

Proof. See [14], Chapter 15, Section 4. □

Theorem 4. *The control laws presented above converge to the set of maximum points of the utility function* $U(\mathbf{x})$ *if this set is bounded, the condition of Theorem 3 is satisfied and vector* $\mathbf{v}^0$ *is an inner point of the set defined in Theorem 3, except for the non-binding constraints.*

Proof. See [14], Chapter 15, Section 4. The proof follows by using Theorem 3 and Lemma 5. □

Remark 1. The components of vector $\mathbf{v}^0$ are the Lagrange multipliers of the optimization problem at hand.

10.1 Proof of Theorem 1

The conditions on the parameters α, β_i, β_I^m, β_i^M and $\xi_{i,j}$ imposed in Theorem 1 imply that the necessary and sufficient conditions of Theorem 3 are satisfied for the optimization problem at hand. Indeed, if each congested link is used by a non-binding CoS or a BE call, then in $\nabla U(\mathbf{x}^*) = H(\mathbf{x}^*)\mathbf{v}^0$ the components of $\mathbf{v}^0$ associated with capacity constraints; i.e., v_k^0 for $k = 1,2,\ldots,\mathrm{card}(\mathcal{L})$, appear in a set of equations decoupled from the remaining components of $\mathbf{v}^0$. Then, the worst case (larger) value of v_k^0, $k = 1,2,\ldots,\mathrm{card}(\mathcal{L})$ is

$$v_{k,max}^0 = \max_{i,j,\mathbf{x}\in\mathcal{X}} \frac{\partial U(\mathbf{x})}{\partial x_{i,j}}$$

Now, using this information in the remaining equations, it is possible to solve for $v_{k,max}^0$, $k = m+1,\ldots .L$. Since, $U(\mathbf{x})$ is an increasing function in all its variables $x_{i,j}$, the worst case for v_k^0 associated with CoS constraints is

$$v_{k,max}^0 = \sum_{\kappa\in\mathcal{K}\subseteq\{1,2,\ldots,\mathrm{card}(\mathcal{L})\}} v_{\kappa,max}^0 = \max_{i,j} B_{i,j} \max_{i,j,\mathbf{x}\in\mathcal{X}} \frac{\partial U(\mathbf{x})}{\partial x_{i,j}} \doteq \alpha_{min} \max_{i,j} B_{i,j}.$$

Once these are determined, all that remains is to pick the worst case value $v_{k,max}^0$ associated with non-negativity constraints. Since each one of these appears in a single equation, where all the other multipliers are already determined, the worst case value is given by

$$v_{k,max}^0 = \alpha_{min} \max_{i,j} B_{i,j} + \beta, \qquad \beta \in \{\beta_i, \beta_i^m, \beta_i^m\}.$$

Hence, in order to satisfy the conditions in Theorem 3

$$v_k \leq v^0_{k,max} < \gamma_k\,, \quad k = 1,2,\ldots,m$$
$$|v_k| \leq \left|v^0_{k,max}\right| < \xi_k\,, \quad k = m+1,\ldots,L.$$

Therefore, the family of adaptation laws proposed in this paper converge to the maximum of the utility function $U(\mathbf{x})$ subject to $\mathbf{x} \in \mathcal{D}$. In other words, they converge to the optimum of our optimization problem. □

10.2 Proof of Theorem 2

In this section we present the proof for Theorem 2, but first we set the stage by introducing some notation that is needed.

The control laws presented in Section 4.2 can be written in terms of a *modified control* as follows: Let $\mathcal{I}_{i,j}$ and $\mathcal{I}_i$ be the set of indices $k \in \{1,2,\ldots,L\}$ such that the inequality, respectively equality, constraints $h_k(\mathbf{x})$ involve the data rate $x_{i,j}$. Furthermore, let $\mathcal{I}_{i,j}$ be classified into $\mathcal{I}^{\alpha}_{i,j}$ and $\mathcal{I}^{\xi}_{i,j}$ for the capacity and non-negativity constraints respectively. Note that each of the sets $\mathcal{I}^{\xi}_{i,j}$ and $\mathcal{I}_i$ consist of a single point. Now, define the *modified control* u_k as

$$u_k = \begin{cases} 0 & \text{if } h_k > 0 \\ u_{k,\mathrm{eq}} & \text{if } h_k = 0 \\ 1 & \text{if } h_k < 0 \end{cases} \qquad k \in \{1,2,\ldots,m\}$$

$$u_k = \begin{cases} (1-r_i) & \text{if } h_k \neq 0 \\ u_{k,\mathrm{eq}} & \text{if } h_k = 0 \end{cases} \qquad k \in \{m+1,\ldots,L\} \cap \mathcal{I}_i,$$

where $u_{k,\mathrm{eq}}$ applies when a sliding mode occurs in the surface $h_k(\mathbf{x}) = 0$ and is defined as the convex combination of the maximum ($\overline{u}_k$) and minimum ($\underline{u}_k$) values of u_k; i.e., $u_{k,\mathrm{eq}} \doteq \lambda \underline{u}_k + (1-\lambda)\overline{u}_k$, with $\lambda \in [0,1]$. Finally, to simplify the notation let $z_{i,j}$ denote

$$z_{i,j}\left(t, \mathbf{x}_i(t), \{u_k\}_{k \in \mathcal{I}^{\alpha}_{i,j}}\right).$$

Then, the following *modified control* laws are equivalent to the ones in Section 4.2: For $i = 1,2,\ldots,s$; i.e, AS calls, let

$$\dot{x}_{i,j} = z_{i,j}\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha\left(1 - \prod_{k \in \mathcal{I}^{\alpha}_{i,j}} u_k\right) - \beta_i(1-u_l) + \xi_{i,j}(1-u_p)\right],$$

where $l \in \mathcal{I}_i$ and $p \in \mathcal{I}^{\xi}_{i,j}$. For $i = s+1, s+2, \ldots, n$; i.e., BE calls, let

$$\dot{x}_{i,j} = z_{i,j}\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha\left(1 - \prod_{k \in \mathcal{I}^{\alpha}_{i,j}} u_k\right) + \xi_{i,j}(1-u_p)\right].$$

Remark 2. There are as many parameters u_k as there are constraints in the optimization problem at hand, as opposed to one parameter r_i per type of calls and one parameter $bin_{i,j}$ and $\xi_{i,j}$ per path.

In a similar manner, the *convergent* control laws presented in Section 4.2 can also be recast as a *modified control* form: For $i = 1,2,\ldots,s$; i.e, AS calls, let

$$\dot{x}_{i,j}^{int} = z_{i,j}\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha\sum_{k\in\mathcal{I}_{i,j}^{\alpha}}(1-u_k^{int}) - \beta_i(1-u_l) + \xi_{i,j}(1-u_p)\right],$$

where $l \in \mathcal{I}_i$ and $p \in \mathcal{I}_{i,j}^{\xi}$. For $i = s+1, s+2,\ldots,n$; i.e., BE calls, let

$$\dot{x}_{i,j}^{int} = z_{i,j}\left[\left.\frac{\partial f_i}{\partial x_{i,j}}\right|_{\mathbf{x}_i} - \alpha\sum_{k\in\mathcal{I}_{i,j}^{\alpha}}(1-u_k^{int}) + \xi_{i,j}(1-u_p)\right].$$

Finally, let $\mathrm{ord}(\alpha^{-1})$ and $\mathrm{Ord}(\alpha^{-1})$ denote terms of the order of α^{-1} in the sense

$$\lim_{\alpha\to\infty}\alpha\,\mathrm{ord}(\alpha^{-1}) = 0 \qquad \text{and} \qquad \lim_{\alpha\to\infty}\alpha\,\mathrm{Ord}(\alpha^{-1}) = \mathrm{cte}.$$

Proof. The following proof stems from the fact that the control laws in Section 4.2 provide convergence in finite time to the admissible region $\mathcal{C}$, where

$$\mathcal{C} = \left\{\mathbf{x}\in\mathbf{R}^{n_1+\cdots+n_n} : h_k(\mathbf{x}) \le 0,\ k = 1,2,\ldots,m\right\}.$$

Given the assumptions on the utility function $U(\mathbf{x})$, the conditions on α, β_i and $\xi_{i,j}$ can be satisfied. Therefore, the derivatives $x_{i,j}$ are bounded away from 0 for all $\mathbf{x} \notin \mathcal{C}$ and convergence to $\mathcal{C}$ in finite time is guaranteed.

For simplicity, the case where $z_{i,j}(\cdot) = 1$ is shown here. However, a straightforward modification can be done to address the case of a general $z_{i,j}(\cdot)$.

Let $\mathbf{x}^*$ denote the optimal solution to the problem posed in Section 3 and let $\mathbf{x}^{eq}$ denote any equilibrium point of the laws above that might exist.

Without loss of generality, let a sliding mode occur in the admissible region along the intersection of a set of surfaces $h_k(\mathbf{x}) = 0$, $k \in \mathcal{I}$. Since a sliding mode occurs along these surfaces it follows that $\dot{h}_k(\mathbf{x}) = 0$, for all $k \in \mathcal{I}$ and all α. Hence, given the specific linear dependence of the previous expression on $x_{i,j}$, the term

$$\alpha\left(1 - \prod_{k\in\mathcal{I}\cap\mathcal{I}_{i,j}^{\alpha}} u_{k,\mathrm{eq}}\right)$$

is bounded for all i, j and α. Moreover, as $\alpha \to \infty$, it holds that $u_{k,\mathrm{eq}} \to 1$, $\forall k \in \mathcal{I}$. Therefore, for all i, j, the following Taylor expansion holds around $u_k = 1$, $\forall k \in \mathcal{I}$

$$\alpha\left(1 - \prod_{k\in\mathcal{I}\cap\mathcal{I}_{i,j}^{\alpha}} u_{k,\mathrm{eq}}\right) = \alpha\sum_{k\in\mathcal{I}\cap\mathcal{I}_{i,j}^{\alpha}}\Big((1-u_{k,\mathrm{eq}}) + \mathrm{Ord}\big((1-u_{k,\mathrm{eq}})^2\big)\Big).$$

Since the left hand side is bounded for every α it holds that

$$\lim_{\alpha\to\infty} \alpha \sum_{k\in\mathcal{I}\cap\mathcal{I}^{\alpha}_{i,j}} \mathrm{Ord}\big((1-u_{k,\mathrm{eq}})^2\big) = 0, \quad \text{and} \quad \alpha \sum_{k\in\mathcal{I}\cap\mathcal{I}^{\alpha}_{i,j}} \big(1-u_{k,\mathrm{eq}}\big) < M \quad M\in\mathbf{R}$$

Equivalently we can write,

$$1-\prod_{k\in\mathcal{I}\cap\mathcal{I}^{\alpha}_{i,j}} u_{k,\mathrm{eq}} = \sum_{k\in\mathcal{I}\cap\mathcal{I}^{\alpha}_{i,j}} (1-u_{k,\mathrm{eq}}) + \mathrm{ord}(\alpha^{-1}).$$

Now, using the expression for the equivalent control (10)

$$\big(H_1^T H_1\big)(\mathbf{v}_{1,\mathrm{eq}}) = \big(H_1^T \nabla U + H_1^T H_2 \mathbf{v}_2\big)$$
$$\big(H_1^T H_1\big)\big(\mathbf{v}^{bin}_{1,\mathrm{eq}} + \overline{\mathrm{ord}}(\alpha^{-1})\big) = \big(H_1^T \nabla U + H_1^T H_2 \mathbf{v}^{bin}_2\big),$$

where $\dot{\mathbf{v}}_2 = \dot{\mathbf{v}}^{bin}_2 = 0$ and $\mathbf{v}_2 = \mathbf{v}^{bin}_2$. Subtracting these, $\big(\mathbf{v}_{1,\mathrm{eq}} - \mathbf{v}^{bin}_{1,\mathrm{eq}}\big) = \overline{\mathrm{ord}}(\alpha^{-1})$, where $\overline{\mathrm{ord}}(\alpha^{-1})$ is a vector whose components are of the form $\mathrm{ord}_k(\alpha^{-1})$. Hence,

$$\dot{x}_{i,j} = \dot{x}^{int}_{i,j} + \alpha\,\mathrm{ord}(\alpha^{-1}).$$

Furthermore, since $\dot{U}(\mathbf{x}) = \nabla U(\mathbf{x})^T \dot{\mathbf{x}}$,

$$\dot{U}(\mathbf{x}) = \nabla U(\mathbf{x})^T \dot{\mathbf{x}}^{int} + \nabla U(\mathbf{x})\alpha\,\mathrm{ord}(\alpha^{-1}) = \dot{U}^{int}(\mathbf{x}) + \mathrm{Ord}(\alpha^{-1}).$$

Now, given $\varepsilon > 0$ define $\mathcal{B}_\varepsilon(\mathbf{x}^*)$ to be an open neighborhood of radius ε around the optimal $\mathbf{x}^*$.

Therefore, since $dU^{int}(\mathbf{x})/dt > 0$ for all $\mathbf{x} \neq \mathbf{x}^*$, and the set $\mathbf{R}$ is dense; there exists α^* such that for all $\alpha > \alpha^*$, and all $\mathbf{x} \notin \mathcal{B}_\varepsilon(\mathbf{x}^*)$, it holds that $\dot{U}(\mathbf{x}) > 0$. Hence, there are no stationary points of the control law outside $\mathcal{B}_\varepsilon(\mathbf{x}^*)$ and since for every trajectory $U(\cdot)$ is strictly increasing outside this neighborhood, then $\mathbf{x} \to \mathcal{B}_\varepsilon(\mathbf{x}^*)$.

Finally, since the choice of ε is arbitrary, it holds that for all $\varepsilon > 0$ there exists α^*, such that the above control laws converge to $\mathcal{B}_\varepsilon(\mathbf{x}^*)$, for all $\alpha > \alpha^*$. □

10.3 Sketch of the Proof of Lemma 1

The proof essentially requires two steps. First, we show that a sliding motion on the surfaces $x_{i,j} = 0$ is obtained. Second, we show that the utility function does not decrease along these trajectories, in essence the same reasoning used in Section 10.1.

Since the proposed laws reduce to the ones in Theorem 1 for the case $x_{i,j} > 0$, we concentrate without loss of generality, on the case $x_{i,j} = 0$.

Let $x_{i,j} = 0$ at time $t = t_0$, for some i, j. If at t_0, $p_{i,j} > 0$, then the adaptation laws reduce to the ones in Theorem 1 and convergence is guaranteed.

On the other hand, if $p_{i,j} < 0$, a sliding mode will occur on the surface $x_{i,j} = 0$, with $\dot{x}_{i,j} = 0$. Indeed, since the constraints of the optimization problem at hand are affine, by forcing $\dot{x}_{i,j} = 0$ a sliding mode on the aforementioned surface occurs, as is the case with the laws in Theorem 1. Furthermore, the equivalent control is also the same. This is the case because the equivalent control has the effect of keeping the trajectories tangent to the sliding surface with $\dot{x}_{i,j} = 0$. Moreover, all other derivatives $\dot{x}_{k,l}$ such that $x_{k,l} \neq 0$ at time t_0 are the same.

Therefore using the same reasoning as in Section 10.1, the utility function does not decrease along the trajectories generated by the above control laws and a near-optimum is achieved. □

References

[1] Flavio Bonomi, Debasis Mitra, and Judith B. Seery. Adaptive algorithms for feedback-based flow control in high-speed, wide-area networks. *IEEE J. Select. Areas Commun.*, 13(7):1267–1283, September 1995.

[2] D. Chiu and R. Jain. Analysis of the Increase/Decrease algorithms for congestion avoidance in computer networks. *J. Comp. Networks and ISDN systems*, 17(1):1–14, June 1989.

[3] Aleksei Fedorovich Filippov. *Differential Equations with Discontinuous Right-hand Sides*. Mathematics and Its Applications. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1988.

[4] Sally Floyd and Tom Henderson. The new Reno modification to TCP's fast recovery algorithm. IETF, RFC 2582, April 1999.

[5] Sally Floyd and Van Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Trans. Networking*, 1(4):397–413, August 1993.

[6] S. Jamaloddin Golestani and Supratik Bhattacharyya. A class of end-to-end congestion control algorithms for the Internet. In *Proc. Int. Conf. Network Protocols, ICNP*, pages 137–150, October 1998.

[7] Frank P. Kelly, Aman K. Maulloo, and Dacid K. H. Tan. Rate control in communication networks: Shadow prices, proportional fairness and stability. *J. Oper. Res. Soc.*, 49(3):237–252, March 1998.

[8] Richard J. La and Venkat Anantharam. Charge-sensitive TCP and rate control in the Internet. In *Proc. IEEE INFOCOM'2000*, pages 1166–1175, March 2000.

[9] Constantino M. Lagoa, Hao Che, and Bernardo Adrián Movsichoff. Adaptive control algorithms for decentralized optimal traffic engineering in the Internet. To appear in IEEE/ACM Trans. Networking, 2004.

[10] Steven H. Low. Optimization flow control with on-line measurement or multiple paths. In *Proc. 16th Int. Teletraffic Cong.*, Edinburgh, U.K., June 1999.

[11] Laurent Massoulié and James Roberts. Bandwidth sharing: Objectives and algorithms. In *Proc. IEEE INFOCOM '99*, pages 1395–1403, March 1999.

[12] Gopalakrishnan Ramamurthy and Aleksandar Kolarov. Application of control theory for the design of closed loop rate control for ABR service. In *Proc. Int. Test Conf.,ITC*, pages 751–760, Washington, USA, 1997.

[13] Saswati Sarkar and Leandros Tassiulas. Distributed algorithms for computation of fair rates in multirate multicast trees. In *Proc. IEEE INFOCOM'2000*, volume 1, pages 52–61, Tel Aviv, Israel, March 2000.

[14] Vadim I. Utkin. *Sliding Modes in Control and Optimization*, volume 66 of *Communications and Control Engineering Series*. Spriger-Verlag, Berlin, Heidelberg, 1992.