End-to-End Optimal Algorithms for Integrated QoS, Traffic Engineering, and Failure Recovery

Bernardo A. Movsichoff, *Member, IEEE,* Constantino M. Lagoa, *Member, IEEE,* and Hao Che

Abstract

This paper addresses the problem of optimal Quality of Service (QoS), Traffic Engineering (TE) and Failure Recovery (FR) in Computer Networks by introducing novel algorithms that only use source inferrable information. More precisely, optimal data rate adaptation and load balancing laws are provided which are applicable to networks where multiple paths are available and multiple Classes of Service (CoS) are to be provided. Different types of multiple paths are supported, including point-to-point multiple paths, point-to-multipoint multiple paths, and multicasting. In particular, it is shown that the algorithms presented only need a minimal amount of information to achieve an optimal operating point. More precisely, they only require knowledge of whether a path is congested or not. Hence, the control laws provided in this paper allow source inferred congestion detection without the need for explicit congestion feedback from the network. The proposed approach is applicable to utility functions of a very general form and endows the network with the important property of robustness with respect to node/link failures; i.e., upon the occurrence of such a failure, the presented control laws reroute traffic away from the inoperative node/link and converge to the optimal allocation for the "reduced" network. The proposed control laws set the foundation for the development of feature-rich traffic control protocols at the IP, transport, or higher layers with provable global stability and convergence properties. Highly scalable QoS, TE, and FR features can be implemented based on these control laws, without the involvement of the routers in the network core.

Index Terms

Distributed Traffic control, QoS, Failure Recovery, Sliding Mode Control, Optimization

Bernardo A. Movsichoff and Constantino M. Lagoa are with the Department of Electrical Engineering, The Pennsylvania State University. Email: bernardo@gandalf.ee.psu.edu; lagoa@engr.psu.edu

Hao Che is with the Department of Computer Science and Engineering, University of Texas at Arlington. Email: hche@cse.uta.edu

I. Introduction

The Transport Control Protocol (TCP) window flow control algorithms use minimum information from the network as input to allow fully distributed traffic control. In other words, the only needed feedback information for the TCP window flow control is whether the forwarding path is congested or not. This allows the TCP source node to *infer* path congestion by counting the number of repetitive acknowledgments of the same packet or measuring end-to-end round-trip delay, making TCP a truly end-to-end protocol without the assistance of the underlying internetworking layer infrastructure. This has made the proliferation of the Internet applications at the global scale possible. An excellent example is the fast, ubiquitous adoption of World Wide Web due to its use of TCP as its underlying transport.

However, as the Internet has evolved into a global commercial infrastructure, there has been a great demand for new applications of global reach, for which today's Internet protocols cannot adequately support. For example, realtime applications, such as Voice over IP (VoIP) and video phone, have stringent delay and delay jitter requirements, which cannot be adequately supported by today's Internet protocols. As a result, in recent years, a large number of new Internet protocols were developed in an attempt to meet this demand. For example, Multiprotocol Label Switching (MPLS) has been envisioned as an ideal platform upon which guaranteed services could be developed. Service guarantee is achieved by setting up and managing a set of primary and backup Class-of-Service (CoS) aware label switched paths across an IP domain. In addition to MPLS, this approach requires a suite of protocols be implemented, e.g., DiffServ for Quality of Service (QoS), path protection/fast rerouting for link failure recovery (FR), and constraint-based routing for traffic engineering (TE). This, however, means that, to adequately support realtime applications, a whole suite of protocols with significant involvement of the IP core nodes need to be developed. This raises serious concerns about the scalability and complexity of using these protocols to support realtime applications at a global scale.

Hence a key question to be answered is whether it is possible to enable the above service quality features, including QoS, TE, and FR, with the involvement of communication end points only. In this paper, we put forward a much needed mathematical framework to make this possible. We show that a large family of

Distributed traffic Control Laws (DCLs) exists, which allows optimal, multiple CoSes, multipath[1] based rate adaptation and load balancing. The DCLs drive the network to an operation point where a user defined global utility function is maximized. The DCLs control the traffic independently at different traffic source nodes, e.g., edge nodes or end-hosts. A salient feature of this family of DCLs is that the needed information feedback from the network is minimum, i.e., whether a forwarding path is congested or not, which can be inferred at the source node itself, the same way as TCP congestion notification. This makes it possible to allow this family of DCLs to be operated end-to-end. A core node may be CoS and multipath agnostic and may employ any queue management/scheduling algorithms, e.g., simple FIFO queues, at its output ports. This family of DCLs allows fast timescale TE through multipath load balancing which is robust in the presence of link/node failures. In other words, the DCLs can automatically repartition the traffic in an optimal way among the rest of the multipath in react to any path failures. Hence, this family of DCLs by design has the capability to enable optimal, scalable QoS, TE, and FR, simultaneously. Moreover, since the mathematical formulation allows both point-to-point multipath and point-to-multipoint multipath, the family of DCLs can be applied to a connectionless IP network to enable sophisticated service quality features, solely based on a set of shortest paths from any given ingress node to a set of egress nodes.

The remainder of the paper is organized as follows: Section II reviews the related work. Section III presents notation and assumptions used throughout this paper, Section IV provides a precise statement of the problem to be solved, while Section V introduces the proposed optimal solution. Section VII on the other hand, discuss some implementation issues while Section VIII provides some simulation results. Finally, Section IX provides some conclusions and the Appendix presents the proof of the results in this paper.

II. Related Work

There is extensive literature on distributed traffic control. In particular, algorithms with a focus on TCP types of traffic were developed, including both empirical algorithms (e.g., see [7], [8]) and algorithms based

[1]Here a multipath is defined as a set of paths originated from a given source node to one (i.e.,point-to-point) or a set of (i.e., point-to-multipoint) sink nodes.

on control theory (e.g., see [2], [4]). However, these algorithms assume a single path and the approaches taken are not optimization based.

Since flows with different ingress-egress node pairs share the same network resources, the key challenge in the design of DCLs is the fact that there is a high degree of interaction between different flows due to the resource constraints. One approach to get around this is to incorporate a link congestion cost into the overall utility function, which replaces the link resource constraints. Then, the problem is solved using a gradient type algorithm, resulting in families of DCLs that support point-to-point multi-path load balancing for rate adaptive traffic, e.g., Golestani, et al. [10], Elwalid, et al. [5], and Guven, et al. [11].

Recently, significant research effort has been made in the design of DCLs with link capacity constraints explicitly taken into account. At the core of this endeavor is the development of DCLs which converge to an operation point where a given global utility function is maximized. This line of research has been proven to be fruitful. Large families of DCLs of this kind are obtained based on the nonlinear programming techniques, e.g., the work by Kelly, et al. [14], Low and Lapsley [18] [21], La and Anantharam [16], and Kar, et al. [13]. These families of DCLs generally require that a sum of link “prices" for all the links in the forwarding path to be periodically calculated and fed back to the source. Under the condition that the network is lightly loaded, the DCLs developed in [16] and [13] allow local control without feedback from the network. In particular, in [21], a family of rate adaptive control laws is design that requires only single bit binary feedback indicating whether the path is congested or not. Kelly, et. al. [14] found a TCP-like DCL that allows point-to-point multipath, under the condition that there is no feedback delay. Recently, Han, et al. [12] successfully extended the results in [14] to allow feedback delay. The results were applied to an overlay network of BGP peers with dedicated resources to allow point-to-point multi-path load balancing.

However, all the above results can only be applied to rate adaptive traffic. Recently, the authors of this paper developed a family of DCLs [17] based on nonlinear control theory [15]. This family of DCLs can be applied not only to usual rate adaptive traffic with point-to-point multipath, but also to rate adaptive traffic with minimum service requirements and/or maximum allowed sending rate and to services with targeted rate guarantee, all allowing for point-to-point multipath. The only needed feedback from the network is the number of congested links along the forwarding paths. Moreover, the technique applies to any utility function that can be expressed as a sum of concave terms.

Nevertheless, due to the needed use of the number of congested links in a forwarding path as the input to a DCL, the family of DCLs proposed in [17] requires explicit congestion feedback from the network. Hence, this family of DCLs can only be applied to a connection-oriented network, such as an MPLS enabled IP network. In this paper, a new family of DCLs is design, free of limitations suffered by the family of DCLs proposed in [17], while retaining all the nice features enjoyed by that family of DCLs. Moreover, the new family of DCLs allows both point-to-point multipath and point-to-multipoint multipath, making it applicable to a connectionless IP network using multiple source rooted shortest paths found by an underlying intradomain routing protocol.

Finally, note that in a related work in [20], the authors of this paper designed a family of DCLs which allows hop-by-hop rate adaptation and load balancing with minimum information exchange between neighboring nodes. It is particularly powerful to provide sophisticated service quality features at the internetworking layer in a connectionless IP network, while the family of DCLs developed in this paper is particularly useful to allow sophisticated service quality features to be developed at the transport or higher layers end-to-end.

III. Preliminaries

Throughout this paper, it is assumed that traffic flows can be described by a fluid flow model, where the only resource taken into account is link bandwidth. For simplicity, we first restrict ourselves to the point-to-point multipath only and address the point-to-multipoint and multicast cases later.

Consider a computer network where calls of different *types* are present. In this paper, *types* denote aggregate of calls with the same ingress and egress node, as well as service requirements; i.e., calls that share a given set of paths connecting the same ingress/egress node pair and whose service requirements are to be satisfied by the aggregate, not by individual calls. Note that when the edge nodes coincide with the end-hosts, the control laws developed in this paper become end-to-end control laws working at the transport layer servicing individual application flows.

More precisely, consider a computer network whose set of links is denoted by $\mathscr{L}$ and let c_l be the capacity of link $l \in \mathscr{L}$. Let n be the number of types of calls, n_i be the number of paths available for calls of type i

and $\mathscr{L}_{i,j}$ be the set of links used by calls of type i taking path j; i.e., if $B_{i,j} = \mathrm{card}(\mathscr{L}_{i,j})$, the cardinality of the set $\mathscr{L}_{i,j}$, then $B_{i,j}$ is the number of links in this path. Given calls of type i, let $x_{i,j}$ be the total data rate of calls of type i using path j. Also, let $\mathbf{x}_i \doteq [x_{i,1}, x_{i,2}, \ldots, x_{i,n_i}] \in \mathbf{R}^{n_i}$ denote the vector containing the data rates allocated to the different paths taken by calls of type i, and $\mathbf{x} \doteq \left[\mathbf{x}_1^T, \mathbf{x}_2^T, \ldots, \mathbf{x}_n^T\right]^T \in \mathbf{R}^N$ the vector containing all the data rates allocated to different call types and respective paths, where $N = n_1 + n_2 + \cdots + n_n$.

Now, a link $l \in \mathscr{L}$ is said to be congested if the aggregated data rate of the calls using the link reaches its capacity c_l. The congestion information $cg_{i,j}$ for calls $x_{i,j}$; i.e., calls of type i taking path j, is defined as $cg_{i,j} \doteq 1$ if any link $l \in \mathscr{L}_{i,j}$ is congested, and 0 otherwise. Moreover, $\overline{cg_{i,j}}$ denotes the logical not operation on $cg_{i,j}$; i.e., $\overline{cg_{i,j}} = 1 - cg_{i,j}$.

IV. Problem Statement

In this paper, we aim at solving the same problem as in [17]; i.e., developing data rate adaptation laws that maximize a given utility function subject to CoS requirements. Although addressing the same problem, the solution to the problem presented in this paper is not an incremental improvement on the solution provided in [17]. It sets the foundation for the development of a wide variety of traffic control protocols to enable QoS, TE, and FR features simultaneously, which only use source inferrable congestion information. We now define precisely the problem to be solved.

The results in this paper aim at maximizing utility functions of the form

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

subject to network constraints and CoS requirements, where $U_i(\cdot)$, $i = 1, 2, \ldots, n$, are differentiable concave functions, strictly increasing in each of their arguments, and α is a positive scaling constant. Given this, the problem of optimal resource allocation can be formulated (see [17]) as the following optimization problem:

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

subject to the network capacity constraints

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

the CoS requirements: the Assured Forwarding (AF) requirements

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

the Minimum Rate Guaranteed Service (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 Upper Bounded Rate Service (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 Minimum Service Guarantee and an Upper Bounded Rate (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 nonnegative

$$x_{i,j} \geq 0; \qquad i = 1, 2, \ldots, n; \; j = 1, 2, \ldots, n_i.$$

Obviously, the optimization problem above is a convex problem; i.e., maximizing a concave function over a convex set. If global information is available then algorithms like gradient descent could be used to solve it. However, generally, global information is not available. The objective of this paper is to provide decentralized adaptation laws that converge to the solution of the problem stated above[2].

A. Point-to-Multipoint Service

The problem formulation so far has only considered point-to-point multipath; i.e., multiple paths from an ingress node to a given egress node. This formulation, however, is too restrictive. It does not account for the possible need for point-to-multipoint multipath forwarding; i.e., forwarding from an ingress node to multiple egress nodes. This feature is particularly useful when traffic is to be balanced among multiple shortest paths

[2]The provisioning of the aggregated resource for AS, MRGS, and MRGUBS traffic running between any pair of nodes does need to ensure that at least one feasible distribution exists, which is beyond the scope of this paper. Some optimization algorithms with global information such as the one proposed by Mitra [19] can be employed to serve this purpose during the network resource planning phase.

to the destination network reachable via multiple egress nodes, as pointed out in [9]. Now, we show that point-to-multipoint multipath can be easily recast into the problem formulation provided above.

Assume that calls of type i use point-to-multipoint multipath and that it has M egress nodes. As before, we assume that there are several paths connecting the ingress node to each of the egress nodes and denote the data rate used by calls to receiver m that use path j by x_{i,j_m}. Moreover, let $\mathscr{L}_{i,j_m}$ be the set of links used by calls to receiver m taking path j_m.

In this case, one defines congestion of a path in the usual way; i.e., path j_m to receiver m is congested if at least one of the links in $\mathscr{L}_{i,j_m}$ is congested. Hence, as far as link constraints are concerned, no modification in the formulation is needed. The main difference between point-to-multipoint multipath and the point-to-point multipath discussed earlier, is the fact that CoS constraints are to be enforced on the total data rate; i.e., CoS constraints are defined in terms of

$$\sum_{m=1}^{M}\sum_{j=1}^{n_{i,m}} x_{i,j_m}$$

where $n_{i,m}$ is the number of paths available to calls whose receiver is m. Hence, in the case of point-to-multipoint multipath, the control laws will “look” at the overall aggregate data rate to all receivers. Apart from that small difference, the constraints involved are of the same form as the ones used for point-to-point multipath and, therefore, the data rate control laws are similar. Hence, to simplify the exposition, from this point on only point-to-point multipath is considered.

In summary, the problem formulation in this paper addresses a very general multipath forwarding problem including point-to-point, point-to-multipoint, and (using a similar formulation to the one described above) multicast multipaths. Fig. 1 gives an example to show different kinds of multipaths that may co-exist in the network. From ingress node 1 to egress node 3, there is a point-to-point multipath with two paths in it. This multipath, together with the path from ingress node 1 to egress node 5, can form a point-to-multipoint multipath with three paths in it. Also in Fig. 1, there is a multicast multipath from ingress node 2 to egress nodes 3, 4, and 5.

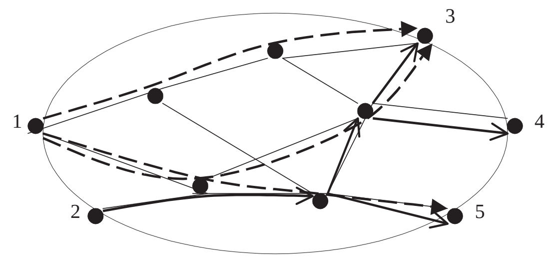

Fig. 1. Examples of point-to-point multipath, point-to-multipoint multipath and a multicast path.

V. A Novel Family of Distributed Rate Adaptation Control Laws

Before presenting the main results in this paper, this section introduces the proposed solution to the optimization problem above, a family of control laws that achieve optimal rate allocation.

Let $f_{i,j}$ be defined as

$$f_{i,j}(\mathbf{x}) \doteq \left(1 - e^{-\partial U/\partial x_{i,j}}\right),$$

and let

$$(y)^+_{x=0} = \begin{cases} \max\{y,0\} \text{ if } x = 0; \\ y \text{ if } x \neq 0. \end{cases}$$

Also, let $z_{i,j}(t,\mathbf{x})$ be positive scalar functions for all i and all j. Now, define the following family of control laws: For $i = 1,2,\ldots,s_1$; i.e., AF calls, let

$$\dot{x}_{i,j} = \left(z_{i,j}(t,\mathbf{x})\left[f_{i,j}(\mathbf{x}) - \left(1 - \overline{\alpha g_{i,j}} r_i\right)\right]\right)^+_{x_{i,j}=0}, \qquad \text{where} \qquad r_i(\mathbf{x}_i) = \begin{cases} r_{min} < 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} > \Lambda_i \\ r_{max} > 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} < \Lambda_i, \end{cases}.$$

For $i = s_1+1, s_1+2, \ldots, s_2$; i.e., MRGS calls, let

$$\dot{x}_{i,j} = \left(z_{i,j}(t,\mathbf{x})\left[f_{i,j}(\mathbf{x}) - \left(1 - \overline{\alpha g_{i,j}} r_i^m\right)\right]\right)^+_{x_{i,j}=0}, \qquad \text{where} \qquad r_i^m(\mathbf{x}_i) = \begin{cases} 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} > \theta_i \\ r_{max}^m > 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} < \theta_i, \end{cases}$$

For $i = s_2+1, s_2+2, \ldots, s_3$; i.e., UBRS calls, let

$$\dot{x}_{i,j} = \left(z_{i,j}(t,\mathbf{x})\left[f_{i,j}(\mathbf{x}) - \left(1-\overline{\alpha g_{i,j}} r_i^M\right)\right]\right)^+_{x_{i,j}=0}, \qquad \text{where} \qquad r_i^M(\mathbf{x}_i) = \begin{cases} r_{min}^M < 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} > \Theta_i \\ 1 & \text{if } \sum_{j=1}^{n_i} x_{i,j} < \Theta_i, \end{cases}.$$

For $i = s_3+1, s_3+2, \ldots, s_4$; i.e., MRGUBS calls, let

$$\dot{x}_{i,j} = \left(z_{i,j}(t,\mathbf{x})\left[f_{i,j}(\mathbf{x}) - \left(1-\overline{\alpha g_{i,j}} r_i^m r_i^M\right)\right]\right)^+_{x_{i,j}=0},$$

where

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

The quantities r_{min}, r_{max}, r_{max}^m, and r_{min}^M are predetermined positive constants chosen to satisfy convergence of the algorithm, as shown in Theorem 1.

Finally, for $i = s_4+1, s_4+2, \ldots, n$; i.e., BE calls, let

$$\dot{x}_{i,j} = \left(z_{i,j}(t,\mathbf{x})\left[f_{i,j}(\mathbf{x}) - \left(1-\overline{\alpha g_{i,j}}\right)\right]\right)^+_{x_{i,j}=0}.$$

A. Main Result

The main result of this paper establishes that the control laws presented above, converge to the solution of the optimization problem posed. This is formally stated in the following theorem.

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

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

Also, assume that at the optimal traffic allocation, each congested link has at least one BE call with non-zero data rate and that the elements of the gradient of the utility function are bounded in $\mathscr{X}$. Let $\zeta > 0$ be a given (arbitrarily small) constant and let $z_{i,j}(t,\mathbf{x})$ be scalar continuous functions satisfying $z_{i,j}(t,\mathbf{x}) > \zeta$, for all $t > 0$ and all $\mathbf{x} \in \mathscr{X}$. Furthermore, let

$$0 < r_{min}, r_{min}^M < r_{lower} < 1 < r_{upper} < r_{max}, r_{max}^m,$$

where $r_{lower} = e^{-v_{k,\max}}$, $r_{upper} = e^{v_{k,\max}}$, and

$$v_{k,\max} = \max_{i,j} B_{i,j} \max_{i,j,\mathbf{x}\in\mathscr{X}} \frac{\partial U}{\partial x_{i,j}}.$$

The quantity $B_{i,j}$, as defined in Section III, is the number of links in path j taken by calls of type i. Then, the control laws presented above converge to a traffic allocation that maximizes the utility function $U(\mathbf{x})$ subject to the network's capacity constraints, CoS requirements and non-negativity of all the data rates.

VI. A TCP-like Control Law For Multipath BE Traffic

It turns out that the linear increase/exponential decrease behavior of the TCP algorithm in its congestion avoidance phase is a particular case of the control laws provided in the previous section. Moreover, these control laws indicate how one can generalize the TCP algorithm to the multipath case. To see this, consider calls of type i belonging to the BE CoS and assume that the aggregate data rate is bounded away from zero. Moreover, assume that the aggregate rate is "large." Now, assume that the associated factor in the utility function is

$$U_i\left(x_{i,1}, x_{i,2}, \ldots, x_{i,n_i}\right) = \log\left(\sum_{j=1}^{n_i} x_{i,n_i}\right).$$

Moreover, take

$$z_{i,j}(t,\mathbf{x}) = \frac{\zeta}{1 - e^{-\alpha \partial U_i / \partial x_{i,j}}}, \quad \text{for some } \zeta > 0.$$

It turns out that, with these parameters and if $\sum_{j=1}^{n_i} x_{i,j}$ is large, the control laws exhibit a TCP-like behavior; i.e., if there is no congestion, the data rate increases linearly. If congestion is detected, the data rates decrease exponentially. More precisely, if no congestion is detected, one has $\dot{x}_{i,j} = \zeta$. If congestion is detected, since it is assumed that the data rate is large

$$e^{\alpha \partial U_i / \partial x_{i,j}} \approx 1 + \frac{\alpha}{\sum_{j=1}^{n_i} x_{i,j}}$$

and, hence,

$$\dot{x}_{i,j} \approx -\frac{\zeta}{\alpha} \sum_{j=1}^{n_i} x_{i,j}.$$

In other words, in multipath case, a TCP-like congestion control law should decrease the sending window by an amount proportional to the aggregate data rate. Obviously, this reduces to the usual TCP algorithm if one just has one path.

VII. Implementation Issues

It is important to note that the new family of DCLs provides the much needed mathematical foundation which allows the use of source inferred congestion detection and notification to maintain layer abstraction. Also important is to realize that the new family of DCLs allows the rate control to be decoupled from the congestion detection mechanisms in use. This means that any queue management algorithm and queue scheduling discipline used in the core nodes, can coexist with the family of DCLs running at the edge nodes or end-hosts. In other words, the implementation of any DCL in this family only needs to consider the two end nodes, provided that a source inferred congestion detection and notification is available. However, having said that, one must realize that different queue management algorithms and queue scheduling disciplines do have an impact on the overall performance for any end-to-end traffic control mechanism (see [3]).

As a result, there are two key components in the implementation of the family of DCLs; i.e., the implementation of the DCL in the edge nodes or end-hosts and the design of source inferred congestion detection and notification mechanisms. The implementation of the DCL control plane and data plane functions in the edge nodes or end hosts are similar to the one described in [20]. In this paper, we focus on the issues related to the design of source inferred congestion detection and notification mechanisms.

Note that due to the wide applicability of the new family of DCLs with respect to rate adaptation, multi-path load balancing, and multiple CoSs, for both connectionless and connection-oriented networks, it is difficult to address detailed implementation issues, unless the network architecture to which the DCL applies is defined. In what follows, we only discuss the general aspects of the implementation issues.

A. *Discretization, Delays and Quantization*

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. 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 1: Let $\mathbf{x}(t)$ be the trajectory obtained using the control laws in Section V 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 and t_d be the discretization period. Again, define $\mathscr{X}$ 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 $\mathbf{x} \in \mathscr{X}$, 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 [6]. ■

Remark: One can sharpen the result above. More precisely, one can prove that the control laws proposed in this paper are asymptotically stable in the presence of both delays and discretization if the gains $z_{i,j}$ converge "slowly" to zero as $t \to \infty$.[3] However, in this case, the network would react very slowly to changes in operating conditions (such as change in traffic demand and/or link/node failure). Hence, this case is not studied further in this paper.

B. Congestion Detection and Notification

To mantain the transport or higher layers abstraction, a source inferred congestion detection and notification mechanism is desirable for the implementation of this family of DCLs in a connectionless IP network. However, unless the transport or higher layer protocol that implements this family of DCLs is defined, the exact source inferred congestion detection and notification mechanism cannot be decided. For example, if a DCL in this family is used in association with a TCP-like reliable transport protocol, a source inferred congestion detection and notification mechanism based on, for example, ACK counts can then be adopted. On the other hand, if the DCL is used in association with an UDP-like unreliable transport protocol, the forwarding path congestion

[3] Stability of networks under delays has been addressed by several authors; e.g., see [1], [21], [24]. However, these results, as opposed to the ones presented in this paper, require a "tight cooperation" between the sending nodes and the network routers.

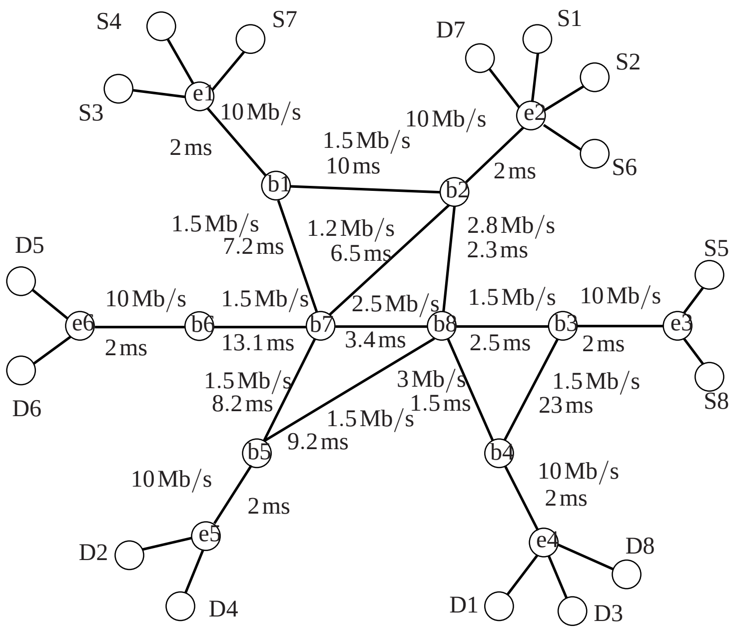

Fig. 2. Topology of the network

may be detected and notified by periodically sending an echo packet to the destination node and measuring the round-trip time of the echoed packet.

The above source inferred congestion detection and notification approaches can also be used in the context of a connection-oriented network, such as an MPLS one. In addition, other mechanisms can be employed; e.g., mechanisms using a signaling protocol for congestion detection and notification or the one described in [17].

C. Failure Detection and Notification

The node/link failure detection and notification may or may not be integrated with the congestion detection and notification mechanism. Again, they are dependent on the actual protocol that implements an DCL in this familiy. For example, a source inferred congestion detection and notification using echo packets to infer path congestion may also be used to infer possible node/link failures. On the other hand, in an MPLS network, the path protection mechanism under development [22] can be leveraged to allow failure detection and notification, separate from the congestion detection and notification mechanisms.

VIII. Simulation Examples

In this section simulation examples are presented, that help in the understanding of the behavior of the proposed control laws. In particular, it is shown that the control laws converge to the optimal traffic allocation while satisfying service requirements and that they provide an optimal way of reacting to link failures. These examples use a discrete-time version of the control laws and a flow approximation for the calls. Furthermore, for simplicity, only calls of AF and BE CoS categories are taken into account. Given the structure of the algorithm, the behavior with other CoSs will be similar.

A. Simulation Setup

The model of the network used for these examples is the same as in [17] which was originally used by La, et al. [16]. The topology is shown in Fig. 2 along with all link capacities and delays. There are overall $n = 8$ types of calls corresponding to the source/destination pairs indicated in the figure. The paths available for each one of these calls are indicated in Table I, where n_i is the number of paths available for calls of type i.

Utilization will be measured by the function

$$U(\mathbf{x}) = \sum_{i=1}^{8} 0.1 \log\left(0.5 + \sum_{j=1}^{n_i} x_{i,j}\right);$$

i.e., $\alpha = 0.1$, where n_i is again indicated in the table. The term 0.5 is included to avoid an infinite derivative at 0 data rate. As for the AF service requirements, calls of types $i = 3$ and $i = 5$ are assumed to have target rates $\Lambda_3 = \Lambda_5 = 1\,\mathrm{Mb/s}$.

Given this, the control laws presented in Section V are of the following form: For $i = 1,3$ and $j = 1,2$; i.e., AF calls

$$\dot{x}_{i,j} = z_{i,j}(t,\mathbf{x})\left[\left(1 - e^{-0.1\left(\sum_{j=1}^{n_i} x_{i,j} + 0.5\right)^{-1}}\right) - \left(1 - \overline{\alpha g_{i,j}} r_i\right)\right]$$

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

$$\dot{x}_{i,j} = z_{i,j}(t,\mathbf{x})\left[\left(1 - e^{-0.1\left(\sum_{j=1}^{n_i} x_{i,j} + 0.5\right)^{-1}}\right) - \left(1 - \overline{\alpha g_{i,j}}\right)\right],$$

where r_i was chosen with a margin of ± 0.001 with respect to the bounds set forth in Theorem 1. The same oscillation reduction scheme as in [17] was used with $z_{i,j}$ taken as $z_{i,j}(t) = \omega(t - t_0)$, where $\omega(t) = 1.8(0.25 +$

TABLE I

PATHS AVAILABLE FOR EACH TYPE OF CALLS

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

0.65^t). Finally, discretization of the continuous was done using a backward rule approximation: Let $\dot{x}_{i,j} = g_{i,j}(\mathbf{x},t)$ denote the continuous time laws derived in Section V. Then the discrete-time counterpart is

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

The discretization step was chosen as $t_d = 5\,\text{ms}$ and the resetting interval as $T = 10\,\text{s}$.

The simulation results are shown in Fig. 3. It can be seen that the utility function converges to a value close to the optimal one, while satisfying the AF requirements imposed on calls of types $n = 3$ and $n = 4$. Calls of BE category of type $n = 2$ are also included as an example of the obtained behavior.

It can be seen that the trajectory of the data rates exhibits an oscillatory behavior. This phenomenon is due to non-ideal implementation factors such as delays and discretization (that were not considered in Section V). Furthermore, these factors prevent the algorithm from reaching the true optimum. Instead, convergence to a small neighborhood of the optimum is achieved. Section VIII-C provides some examples that show the sensitivity of these results to the choice of the parameters of the adaptation laws.

B. Robustness Against Link Failures

The control laws presented in this paper excel at re-routing traffic upon a failure in a node or link. In order to show this feature, the link connecting nodes b_7 and b_8 was opened at time $t_{fail} = 120\,\text{s}$. The behavior of the control laws is shown in Fig. 4 from time $t = 120\,\text{s}$ on. Note from Table I that both AF calls lose one of the

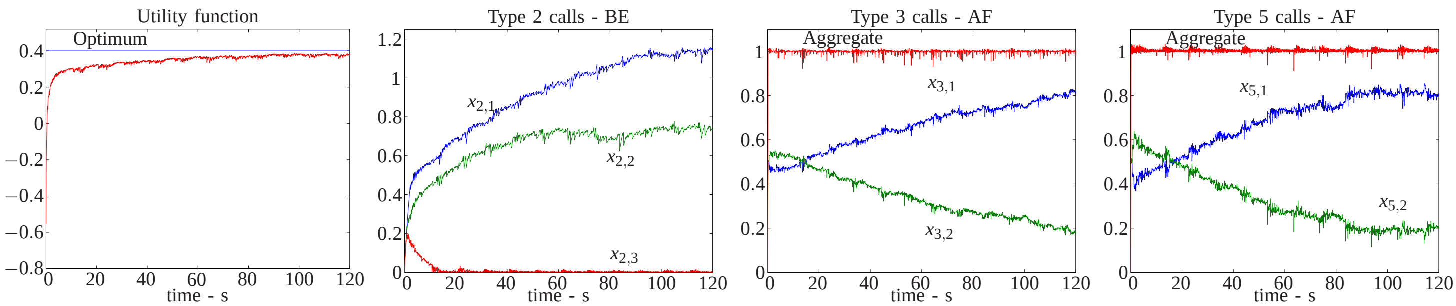

Fig. 3. Example of Simulation Results.

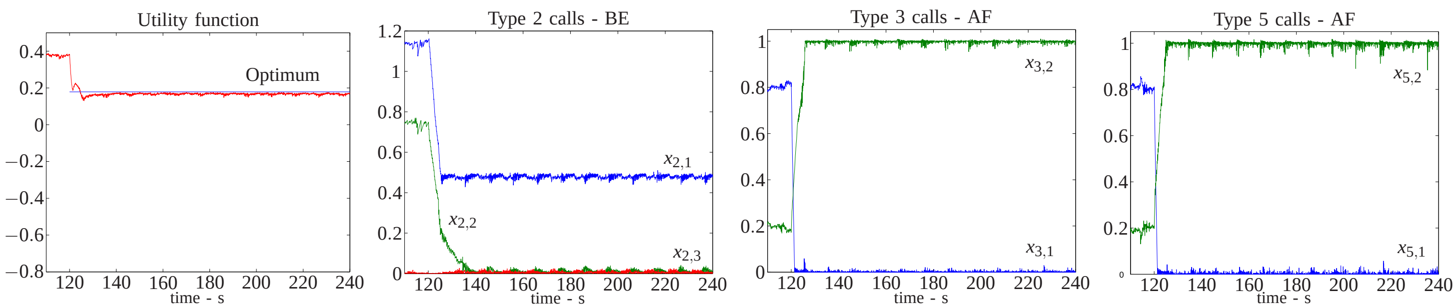

Fig. 4. Robustness Against Link/Node Failures.

two paths they have available so this can be considered to be an extreme situation. As an example, calls of type $i=2$ have to "kill" all traffic on one of the available paths and greatly reduce another. Also, note that in the case when a source inferred congestion detection and notification is used for both congestion *and* failure detection and notification, the control laws implemented at the edge nodes are oblivious to the failure. They simply react to what they perceive as being congestion. In fact, these simulations do not attempt to detect link failure.

C. Sensitivity to the Design Parameters

In this section some relevant simulation are presented, showing the behavior of the algorithm under different choices of the design parameters.

1) Oscillation Reduction Functions: Perhaps one of the most important features of the adaptation laws presented in this paper is the adaptive oscillation reduction, since it has a big impact on performance. Fig. 5 show the behavior for a constant $z_{i,j}=\omega(0)=2.25$; i.e., the maximum value allowed for the time-varying $z_{i,j}$. In comparison with Fig. 3 the observed oscillation is clearly larger in magnitude. Moreover, due to the larger oscillations, convergence to a larger neighborhood of the optimal is obtained and departures from the average

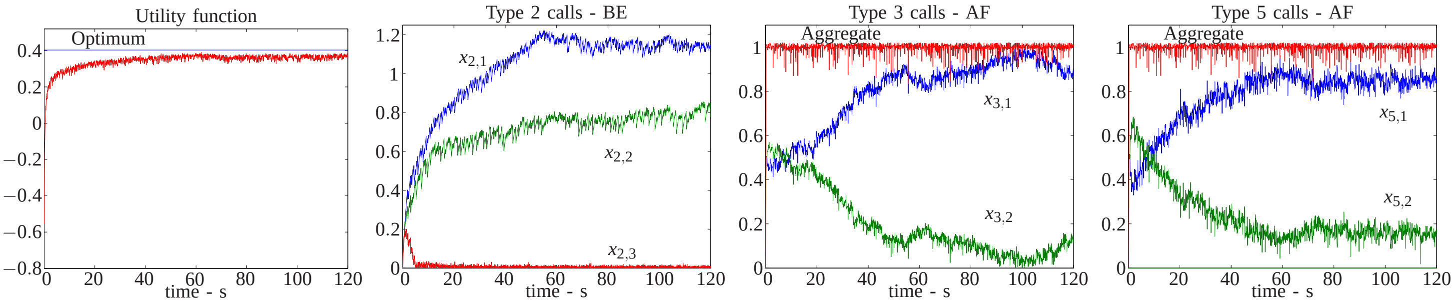

Fig. 5. Example of Constant $z_{i,j}$.

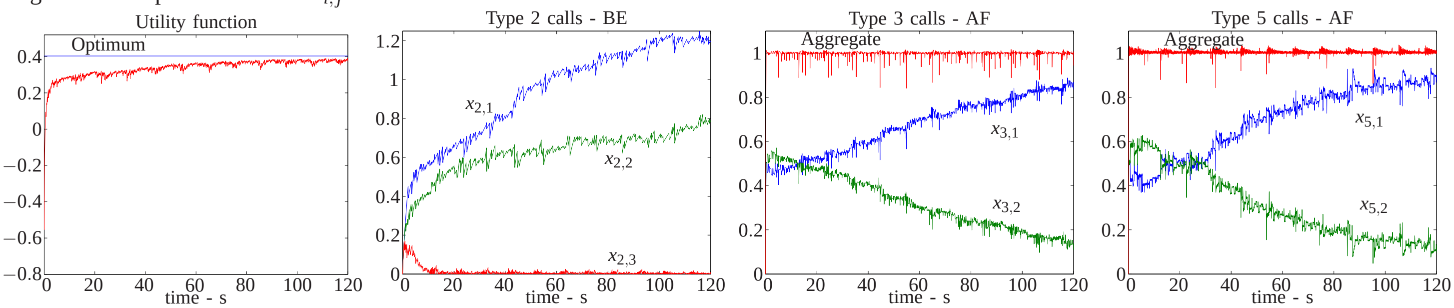

Fig. 6. Example of Larger t_d.

target rates for AF are also larger (providing a worse service to these users). On the other hand the transient response is faster due to larger data rate derivatives.

2) Discretization Step t_d: Another parameter that has a bearing in the performance of the algorithm is the discretization step. In order to show its influence, it was chosen as $t_d = 10\,\text{ms}$. Fig. 6 shows this scenario. Clearly, oscillations are also larger in this case. However, the response is still acceptable and a smaller $z_{i,j}$ could be used to limit the magnitude of the spikes.

3) Scaling of the Utility Function: The scaling of the utility function does not alter the solution of the optimization problem at hand. It does, however, change the bounds on the quantities r_i. Due to the exponential dependence on the gradient, it is advisable to choose a small value of α such that the resulting value of r_{max} is in the order of 1. Simulations have shown that the algorithm is very sensitive to α with the amplitude of the oscillations increasing substantially when one increases this parameter. However, convergence to a neighborhood of the optimal is still achieved as one can expect. Also, the AF constraints are satisfied in the average but large departures from the imposed average rate can happen for high values of α.

4) Propagation Delays: As mentioned before, delays in propagation of information result in an oscillating behavior. More precisely, an increase in the delays will result in a change of behavior similar to the one studied on [17]. Depiction of this behavior is not presented here due to space constraints. The reader is referred to [17] for a more complete study of the influence of propagation delays.

IX. Conclusion

In this paper, a new family of distributed traffic control laws is obtained, which enables scalable quality of service, traffic engineering, and failure recovery features simultaneously, using only source inferrable information. More specifically, these features are enabled through fast timescale CoS-based, dynamic multipath load balancing and rate adaptation, performed by a set of control laws running at the edge nodes locally, independent of each other. Moreover, these control laws drive the network to a operation point where a global design objective is achieved; e.g., maximizing the network revenue. A salient feature of this family of control laws is that the input to each control law is whether a forwarding path in a multipath is congested or not. This feature allows a source node to infer the network congestion, without explicit feedback from the network core. This makes it possible to design a wide variety of highly scalable distributed traffic control protocols with proven optimality and stability.

Effort is now being put in the implementation of the control laws presented in this paper. In particular, these laws have several parameters for which only bounds are provided. Hence, criteria is now being developed for the determination of these "free parameters".

Appendix: Proof of Main Results

In this appendix, the proof of Theorem 1 is presented. We set the stage by introducing some additional notation. Due to space constraints, only the main steps of the proof are presented.

To simplify the exposition to follow, let the problem at hand be recast in the following form

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

subject to inequality constraints

$$h_k(\mathbf{x}) \leq 0; \quad k = 1, 2, \ldots, m$$

and the equality constraints

$$h_k(\mathbf{x}) = 0 \quad k = m+1, m+2, \ldots, L.$$

Now, let the admissible domain be defined as the set

$$\mathscr{C} = \left\{ \mathbf{x} \in \mathbf{R}^N \colon h_k(\mathbf{x}) \leq 0 \text{ for } k \in \{1,2,\ldots,m\} \text{ and } h_k(\mathbf{x}) \text{ not a CoS constraint} \right\};$$

i.e., the set of data rates that can be admitted by the network without any further constraints. Also, let the feasible set be defined as

$$\mathscr{D} = \left\{ \mathbf{x} \in \mathbf{R}^N \colon h_k(\mathbf{x}) \leq 0 \text{ for } k = 1,2,\ldots,m, \text{ and } h_k(\mathbf{x}) = 0 \text{ for } k = m+1, m+2, \ldots, L \right\};$$

i.e., the set of data rates $\mathbf{x} \in \mathscr{C}$ satisfying all the CoS constraints of the optimization problem.

The proof follows by first observing that the control laws in Section V converge to the admissible set in finite time. Then, once inside $\mathscr{C}$ the adaptation laws can be shown to be equivalent to the modified laws

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

where $\mathbf{Z}(\mathbf{x},t)$ is a positive definite matrix and $\mathbf{H}(\mathbf{x}) = \left[\nabla h_1(\mathbf{x}), \nabla h_2(\mathbf{x}), \ldots, \nabla h_L(\mathbf{x})\right]$. In other words, they can be recast in the same form as that of the control laws developed in [17].

Lemma 1: Let r_i satisfy the conditions set forth in Theorem 1. Then vector $\mathbf{x}$ converges to the admissible domain $\mathscr{C}$ in finite time.

Proof: Let $x_{i,j} \geq 0$, for any given i and j, such that $\mathbf{x} \notin \mathscr{C}$ and let $\varepsilon > 0$ be an arbitrarily small constant. By construction of the control laws it holds that $\dot{x}_{i,j} \leq -\varepsilon < 0$. Hence, since the derivative is strictly negative outside $\mathscr{C}$, $x_{i,j}$ reaches the admissible region $\mathscr{C}$ in finite time. ∎

The following Lemma, central to the proof of the results in this paper, provides an alternative representation of the proposed control laws.

Lemma 2: For all $\mathbf{x} \in \mathscr{C}$, the control laws above can be expressed as

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

where $\mathbf{Z}(\mathbf{x},t)$ is a positive definite matrix and

$$\mathbf{H}(\mathbf{x}) = \left[\nabla h_1(\mathbf{x}), \nabla h_2(\mathbf{x}), \ldots, \nabla h_L(\mathbf{x})\right].$$

Proof: If at a given time $x_{i,j}$ is not sliding along the surface $x_{i,j} = 0$, the laws presented in Section V can be formulated as follows: Let $\mathscr{I}_{i,j}$ be the set of indices $k \in \{1,2,\ldots,m\}$ such that the capacity constraints

$h_k(\mathbf{x})$ involve the data rate $x_{i,j}$. Also, let $\mathscr{I}_{i,\text{CoS}}$ be the set of indices $k \in \{1,2,\ldots,L\}$ such that the constraints $h_k(\mathbf{x})$, $k \in \mathscr{I}_{i,\text{CoS}}$ impose CoS requirements on the data rate $x_{i,j}$. Note that this set is empty if calls of type i are of the BE class. Then,

$$\dot{x}_{i,j} = z_{i,j}\left[f_{i,j}(\mathbf{x}) - \left(1 - \prod_{k\in\mathscr{I}_{i,j}\cup\mathscr{I}_{i,\text{CoS}}} u_k\right)\right],$$

where the quantities u_k are defined as follows: For $k \in \mathscr{I}_i^{\text{CoS}}$, $i = 1,2,\ldots,s_1$ (AF constraints) let $u_k \doteq r_i$. For $k \in \mathscr{I}_i^{\text{CoS}}$, $i = s_1+1, s_1+2,\ldots,s_2$ (MGRS constraints) let $u_k \doteq r_i^m$. For $k \in \mathscr{I}_i^{\text{CoS}}$, $i = s_2+1, s_2+2,\ldots,s_3$ (UBRS constraints) let $u_k \doteq r_i^M$. For $k \in \mathscr{I}_i^{\text{CoS}}$, $i = s_3+1, s_3+2,\ldots,s_4$ (MRGUBS constraints) let $u_k \doteq r_i^m r_i^M$. Finally, for $k \in \mathscr{I}_{i,j}$, $i = s_4+1, s_4+2,\ldots,n$ (capacity constraints) let $u_k \doteq \overline{\alpha g_{i,j}}$.

Given the formulation above and to prove that the control laws can be put in the form mentioned, two cases are considered: i) At the given time instant $x_{i,j}$ is sliding along the surface $x_{i,j} = 0$ and ii) At the given time instant, $x_{i,j}$ is not sliding along the surface $x_{i,j} = 0$.

Let us first consider case i). In this case, one has $\dot{x}_{i,j} = 0$ and the motion can be put in the form

$$\dot{x}_{i,j} = z_{i,j}(\mathbf{x},t)\left[\frac{\partial U}{\partial x_{i,j}} - \sum_{k\in\mathscr{I}_{i,j}\cup\mathscr{I}_{i,j}^{\text{CoS}}} \log\frac{1}{u_{k,\text{eq}}} + \xi_{i,j,\text{eq}}\right]$$

where $\xi_{i,j,\text{eq}} \geq 0$ is such that

$$\xi_{i,j,\text{eq}} = \sum_{k\in\mathscr{I}_{i,j}\cup\mathscr{I}_{i,j}^{\text{CoS}}} \log\frac{1}{u_{k,\text{eq}}} - \frac{\partial U}{\partial x_{i,j}}$$

and $u_{k,\text{eq}}$ are the equivalent controls (see [23]) corresponding to the constraints involving $x_{i,j}$. Note that, in this case, such $\xi_{i,j,\text{eq}} \geq 0$ exists because one only has a sliding motion along $x_{i,j} = 0$ if the control laws presented would result in a non-positive derivative of $x_{i,j}$ and, hence,

$$\frac{\partial U}{\partial x_{i,j}} - \sum_{k\in\mathscr{I}_{i,j}\cup\mathscr{I}_{i,j}^{\text{CoS}}} \log\frac{1}{u_{k,\text{eq}}} = -\log\left(1 - f_{i,j}(\mathbf{y})\right) + \log\left(\prod_{k\in\mathscr{I}_{i,j}\cup\mathscr{I}_{i}^{\text{CoS}}} u_{k,\text{eq}}\right) \leq 0$$

Now, let us consider case ii) where $x_{i,j}$ is not sliding along the surface $x_{i,j} = 0$. Now, since $f_{i,j}(\mathbf{x}_i) > 0$, when $\mathbf{x} \in \mathscr{C}$ either $\mathbf{x}$ is an inner point of $\mathscr{C}$ where by definition $u_k = 1$ for all $k \in \mathscr{I}_{i,j}$ or a sliding mode occurs on some surface $s(\mathbf{x}) = 0$, where $\mathbf{x} \in \partial\mathscr{C}$ (the boundary of $\mathscr{C}$). In the latter case, using the equivalent control method (see [23]) there exists $u_{k,\text{eq}}$, such that

$$\dot{x}_{i,j}(t) = z_{i,j}(\mathbf{x},t)\left[-\left(1 - f_{i,j}(\mathbf{x})\right) + \prod_{k\in\mathscr{I}_{i,j}\cup\mathscr{I}_{i,j}^{CoS}} u_{k,\text{eq}}\right].$$

Moreover, since $\max_{\mathbf{x}\in\mathscr{C}} f_{i,j}(\mathbf{x}_i) = \mu < 1$, then there exists a constant $\chi > 0$ such that $\chi < u_{k,\text{eq}}$, for all $\mathbf{x} \in \mathscr{C}$. For $k \in \mathscr{I}_{i,j}^{CoS}$ this is immediate since the lower bound on $u_{k,\text{eq}}$ is $r_{\min}^{M}$. If $k \in \mathscr{I}_{i,j}$, this is a consequence of the fact that $u_{k,\text{eq}} = 1$ if the constraint in not active. If the constraint is active, then one has to have

$$\prod_{k\in \mathscr{I}_{i,j}\cup\mathscr{I}_{i,j}^{CoS}} u_{k,\text{eq}} \geq 1-\mu$$

for all i,j such that $k \in \mathscr{I}_{i,j}$ since, if this is not satisfied, $\dot{x}_{i,j} < 0$ for all i,j such that $k \in \mathscr{I}_{i,j}$ and one could not have a sliding mode along the boundary of constraint k.

Hence, given that the log function has a bounded derivative in the interval $\left[1-\mu,1\right]$, the evolution of $x_{i,j}$ can be represented as

$$\begin{aligned}
\dot{x}_{i,j} &= \hat{z}_{i,j}(\mathbf{x},t)\left[-\log\left(1-f_{i,j}(\mathbf{y})\right)+\log\left(\prod_{k\in \mathscr{I}_{i,j}\cup\mathscr{I}_{i}^{\text{CoS}}} u_{k,\text{eq}}\right)\right] \\
&= \hat{z}_{i,j}(\mathbf{x},t)\left[\log\frac{1}{1-f_{i,j}(\mathbf{y})} - \sum_{k\in \mathscr{I}_{i,j}\cup\mathscr{I}_{i}^{\text{CoS}}} \log\frac{1}{u_{k,\text{eq}}}\right] = \hat{z}_{i,j}(\mathbf{x},t)\left[\frac{\partial U}{\partial x_{i,j}} - \sum_{k\in \mathscr{I}_{i,j}\cup\mathscr{I}_{i,j}^{\text{CoS}}} \log\frac{1}{u_{k,\text{eq}}}\right],
\end{aligned}$$

where $\hat{z}_{i,j}(\mathbf{x},t) = \gamma z_{i,j}(\mathbf{x},t) \geq \hat{\mu}$ and $\gamma \in \left[1-\mu,1\right]$. This is a consequence of the fact that the Mean Value Theorem implies that $\log(a)-\log(b) = (a-b)/c$ for some $c \in [\min\{a,b\},\max\{a,b\}]$.

Now, given the two cases addressed above, we note that $\xi_{i,j,\text{eq}}$ corresponds to a single value of k such that $h_k(\mathbf{x})$ imposes a non-negativity constraint on $x_{i,j}$ and, hence,

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

where $\mathbf{v}(\mathbf{x})$ is a column vector containing the quantities $\log\left(1/u_{2,\text{eq}}\right)$ and $\xi_{i,j,\text{eq}}$ ordered by k, and $\mathbf{Z}(\mathbf{x},t)$ is a positive definite diagonal matrix with elements $\hat{z}_{i,j}$. ■

We are now ready to prove convergence of the rate adaptation control laws. The line of reasoning is the same as in [17]. Hence, we refer the reader to [17] and [23] for proofs of the intermediate results presented below. Define the auxiliary function

$$\widehat{U}(\mathbf{x}) = U(\mathbf{x}) - \Xi(\mathbf{x}), \qquad \text{where} \qquad \Xi(\mathbf{x}) = \left[h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_L(\mathbf{x})\right]\mathbf{v}(\mathbf{x}).$$

Theorem 2: Let $\mathbf{v}^0$ be a vector whose entries are of the form

$$0 \leq v_k \leq \gamma_k \,; \quad k = 1, 2, \ldots, m$$

$$-\psi_k \leq v_k \leq \psi_k \,; \quad k = m+1, m+2, \ldots, L \,,$$

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 $\nabla U(\mathbf{x}^*) = \mathbf{H}(\mathbf{x}^*)\mathbf{v}^0$.

Lemma 3: The function $\widehat{U}(\mathbf{x})$, for $\mathbf{x} \in \mathscr{C}$ does not decrease along the trajectories.

Lemma 4: The time derivative of $\widehat{U}(\mathbf{x})$, for $\mathbf{x} \in \mathscr{C}$, is zero only when $\dot{\mathbf{x}} = 0$.

Lemma 5: The stationary points of $\widehat{U}$ are the maximum points of $\widehat{U}$.

The results above imply the following.

Theorem 3: 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 2 is satisfied and vector $\mathbf{v}^0$ is an inner point of the set defined in Theorem 2, except for the non-binding constraints.

We are now finally ready to address the proof of the main result in this paper.

A. *Proof of Theorem 1*

The definition of $f_{i,j}$ together with the conditions on r_{lower} and r_{upper} imply that the Lagrange multipliers at the KKT point of the optimization problem at hand lie in the convex hull generated by the set of all possible $\mathbf{v}$. Indeed, if each congested link is traversed by a BE call, then in the KKT conditions at the optimum $\mathbf{x}^*$

$$\nabla U(\mathbf{x}^*) = \mathbf{H}(\mathbf{x}^*)\mathbf{v}^0$$

the components of $\mathbf{v}^0$ associated with capacity constraints; i.e., v_k^0 for $k = 1, 2, \ldots, \mathrm{card}(\mathscr{L})$, 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, $k = 1, 2, \ldots, \mathrm{card}(\mathscr{L})$ is

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

Now, using this information in the remaining equations, it is possible to solve for $v_{k,max}^0$, $k = \mathrm{card}(\mathscr{L}) + 1, \ldots, L$. Since $U(\mathbf{x})$ is an increasing function in all its arguments $x_{i,j}$, the largest absolute value of v_k^0 associated with

CoS constraints is given by

$$v^0_{k,max} = \sum_{\kappa \in \mathscr{K}} v^0_{\kappa,max} = \max_{i,j} B_{i,j} \max_{i,j,\mathbf{x}\in\mathscr{X}} \frac{\partial U(\mathbf{x})}{\partial x_{i,j}},$$

where

$$\mathscr{K} \doteq \left\{ \kappa\colon\ \kappa \in \mathscr{I}_{i,j^*};\ j^* = \arg \max_{j=1,2,\ldots,n_i} \operatorname{card}(\mathscr{I}_{i,j});\ i\colon k \in \mathscr{I}_i^{\mathrm{CoS}} \right\}.$$

Finally, note that the multipliers for the non-negativity constraints appear in a single equation where all the others are already determined. Therefore, the worst case value is given by

$$v^0_{k,max} = 2 \max_{i,j} B_{i,j} \max_{i,j,\mathbf{x}\in\mathscr{X}} \frac{\partial U(\mathbf{x})}{\partial x_{i,j}}$$

Hence, it should hold that

$$\begin{aligned} v_k &\le v^0_{k,max} < \gamma_k; && k = 1,2,\ldots,m \\ |v_k| &\le \left| v^0_{k,max} \right| < \psi_k; && k = m+1,\ldots,L. \end{aligned}$$

That is: For capacity constraints $u_k < e^{-v_{k,max}}$, and for CoS constraints $u_k < e^{-v_{k,max}}$ and $u_k > e^{v_{k,max}}$. For the capacity constraints the condition is trivially satisfied with $u_k = 0$, while for COS constraints, these are the conditions imposed on r_{lower} and r_{upper}. Finally, for the positivity constraints, the condition is also satisfied since the equivalent control associated with these constraints $\xi_{i,j,\mathrm{eq}}$ can have arbitrarily high values (see proof of Lemma 2).

Therefore, Theorems 2 and 3 hold. Hence, the family of adaptation laws proposed in this paper converge to the maximum of the utility function $U(\mathbf{x})$ subject to $\mathbf{x} \in \mathscr{D}$. In other words, they converge to the optimum of the optimization problem at hand. ■

References

[1] T. Alpcan and T. Basar, "Global stability analysis of an end-to-end congestion control scheme for general topology networks with delay," in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 1, December 2003, pp. 1092–1097.

[2] F. Bonomi, D. Mitra, and J. B. Seery, "Adaptive algorithms for feedback-based flow control in high-speed, wide-area networks," *IEEE J. Select. Areas Commun.*, vol. 13, no. 7, pp. 1267–1283, Sept. 1995.

[3] L. Breslau, E. W. Knightly, S. Shenker, I. Stoica, and H. Zhang, “Endpoint admission control: Architectural issues and performance,” in *Proc. ACM SIGCOMM 2000*, Stockholm, Sweden, Aug./Sept. 2000, pp. 57–69.

[4] D. Chiu and R. Jain, “Analysis of the Increase/Decrease algorithms for congestion avoidance in computer networks,” *J. Comput. Networks and ISDN Syst.*, vol. 17, no. 1, pp. 1–14, June 1989.

[5] A. Elwalid, C. Jin, S. Low, and I. Widjaja, “Mate: Mpls adaptive traffic engineering,” in *Proceedings of IEEE INFOCOM’01*, vol. 3, Anchorage, AK, April 2001, pp. 1300–1309.

[6] A. F. Filippov, *Differential Equations with Discontinuous Righthand Sides*, ser. Mathematics and Its Applications. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1988.

[7] S. Floyd and T. Henderson, “The new Reno modification to TCP’s fast recovery algorithm,” IETF RFC 2582, Apr. 1999.

[8] S. Floyd and V. Jacobson, “Random early detection gateways for congestion avoidance,” *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

[9] B. Fortz, J. Rexford, and M. Thorup, “Traffic engineering with traditional IP routing protocols,” *IEEE Communications Magazine*, vol. 40, no. 10, pp. 118–124, Oct. 2002.

[10] S. J. Golestani and S. Bhattacharyya, “A class of end-to-end congestion control algorithms for the Internet,” in *Proc. Int. Conf. Network Protocols, ICNP*, Austin, TX, USA, Oct. 1998, pp. 137–150.

[11] T. Guven, C. Kommareddy, R. La, M. Shayman, and B. Bhattacharjee, “Measurement based optimal multi-path routing,” in *Proceedings of INFOCOM’04*, vol. 1, March 2004, pp. 187 – 196.

[12] H. Han, S. Shakkottai, C. V. Hollot, R. Srikant, and D. Towsley, “Overlay TCP for multi-path routing and congestion control,” in *ENS-INRIA ARC-TCP Workshop*, Paris, France, Nov. 2003. [Online]. Available: http://comm.csl.uiuc.edu/~srikant/Papers/multipath_v3.pdf

[13] K. Kar, S. Sarkar, and L. Tassiulas, “A simple rate control algorithm for max total user utility,” in *Proc. IEEE INFOCOM 2001*, vol. 1, Anchorage, AK, USA, Apr. 2001, pp. 133–141.

[14] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, “Rate control in communication networks: Shadow prices, proportional fairness and stability,” *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237–252, Mar. 1998.

[15] S. K. Korovin and V. I. Utkin, “Using sliding modes in static optimization and nonlinear programming,” *Automatica*, vol. 10, no. 5, pp.

525–532, Sept. 1974.

[16] R. J. La and V. Anantharam, “Charge-sensitive TCP and rate control in the Internet,” in *Proc. IEEE INFOCOM 2000*, vol. 3, Tel Aviv, Israel, Mar. 2000, pp. 1166–1175.

[17] C. M. Lagoa, H. Che, and B. A. Movsichoff, “Adaptive control algorithms for decentralized optimal traffic engineering in the Internet,” *IEEE/ACM Trans. Networking*, vol. 12, no. 3, pp. 415–428, June 2004.

[18] S. H. Low and D. E. Lapsley, “Optimization flow control, I: Basic algorithm and convergence,” *IEEE/ACM Trans. Networking*, vol. 7, no. 6, pp. 861–874, Dec. 1999.

[19] D. Mitra and K. Ramakrishnan, “A case study of multiservice, multipriority traffic engineering design for data networks,” in *Proc. GLOBECOM ’99*, vol. 1b, Rio de Janeiro, Brazil, Dec. 1999, pp. 1077–1083.

[20] B. A. Movsichoff, C. M. Lagoa, and H. Che, “Decentralized optimal traffic engineering in connectionless networks,” *IEEE J. Select. Areas Commun.*, vol. 23, no. 2, pp. 293–303, Feb. 2005.

[21] F. Paganini, Z. Wang, J. C. Doyle, and S. H. Low, “Congestion control for high performance, stability, and fairness in general networks,” *IEEE/ACM Trans. Netw.*, vol. 13, no. 1, pp. 43–56, 2005.

[22] V. Sharma and F. Hellstrand, “Framework for Multi-Protocol Label Switching (MPLS)-based recovery,” IETF RFC 3469, Feb. 2003, editors.

[23] V. I. Utkin, *Sliding Modes in Control and Optimization*, ser. Communications and Control Engineering Series. Berlin, Heidelberg: Spriger-Verlag, 1992, vol. 66, no. 1.

[24] L. Ying, G. E. Dullerud, and R. Srikant, “Global stability of internet congestion controllers with heterogeneous delays,” in *Proceedings of the 2004 American Control Conference*, vol. 4, 2004, pp. 2948–2953.