A Framework for Guaranteeing Statistical QoS (Extended Version)

Martin Reisslein, *Member, IEEE*, Keith W. Ross, *Senior Member, IEEE*, Srinivas Rajagopal

Abstract— **Continuous–media traffic (i.e., audio and video) can tolerate some loss but has rigid delay constraints. A natural QoS requirement for a continuous–media connection is a prescribed limit on the fraction of traffic that exceeds an end–to–end delay constraint. We propose and analyze a framework that provides such a statistical QoS guarantee to traffic in a packet–switched network. Providing statistical guarantees in a network is a notoriously difficult problem because traffic flows lose their original statistical characterizations at the outputs of queues. Our scheme uses *bufferless statistical multiplexing* combined with cascaded leaky–buckets for smoothing and traffic contracting. This scheme along with a novel method for bounding the loss probability gives a tractable framework for providing end–to–end statistical QoS. Using MPEG video traces, we present numerical results that compare the connection–carrying capacity of our scheme with that of guaranteed service schemes (i.e., no loss) using GPS and RCS. Our numerical work indicates that our scheme can support significantly more connections without introducing significant traffic loss.**

Keywords: Bufferless Multiplexing, Call Admission Control, End–to–End QoS, Multimedia Traffic, Regulated Traffic, Statistical Multiplexing, Statistical QoS, Traffic Smoothing.

I. Introduction

Continuous–media networking applications are increasingly popular in the Internet. These applications include Internet phone, real–time video conferencing, and streaming stored audio and video. But because the Internet provides only a best–effort service, the Quality of Service (QoS) perceived by a user is inconsistent and unpredictable. In particular, the QoS for a continuous–media session is often poor when the links between communicating entities are congested or subject to sudden and unpredictable traffic surges.

It is therefore desirable to introduce new services into the Internet that can guarantee QoS to continuous–media applications. The subject of providing QoS guarantees in packet–switched networks has been a major area of research over the past 10–20 years, both inside and outside of the Internet research community. One of the propositions that has resulted from this research is a specification for guaranteed QoS [47]. When an application uses this service, the application's packets have guaranteed bounds on delays with *no packet loss*. The guaranteed QoS service is a natural outgrowth of a body of research in the area of delay bound calculations for queueing networks with regulated traffic [7], [8], [34], [35], [58], [57], [18], [27], [4], [26].

It can be argued, however, that guaranteeing absolutely no packet loss is overly conservative for continuous–media applications, which can typically tolerate a small rate of loss. In fact, users may not perceive any quality degradation when there is infrequent packet loss, especially if the receiver employs error concealment techniques (e.g., see [53]). Furthermore, schemes that guarantee no loss typically have a low connection–carrying capacity for bursty continuous–media traffic (e.g., VBR video or speech with silence detection) [43], [20], [21], [19]. Alternatively stated, the no–loss schemes necessitate a high degree of bandwidth over provisioning.

This raises two important questions. First, is it possible to develop a comprehensive framework that provides *statistical QoS guarantees in a network*, that is, bounds on the fraction of traffic that exceeds an end–to–end delay constraint? Providing statistical guarantees in a network context is a notoriously difficult problem because traffic flows lose their original statistical characterizations at the outputs of queues. And if yes, can this statistical–QoS scheme have significantly better connection–carrying capacity than a guaranteed QoS scheme? In this article we first develop a framework that provides statistical QoS guarantees in a *network setting*. We also argue that our approach typically has significantly better connection–carrying capacity than a deterministic guaranteed QoS scheme.

In order to guarantee deterministic or statistical QoS, connections need to make contracts with the network in order to limit, in some sense, the amount of traffic the connections send into the network over intervals of time. Only by making and enforcing contracts can a network expect to be able to provide guarantees. Leaky buckets, being relatively easy to implement, are convenient mechanisms for defining and enforcing traffic contracts. Sources that conform to leaky–bucket characterizations are said to be *regulated sources*. In recent years, several research teams have carefully studied the problem of providing statistical QoS guarantees to regulated sources that are multiplexed in a *single* shared buffer [15][33][37]. With shared buffer multiplexers, however, it is difficult (if not impossible) to tightly characterize a connection's traffic once the traffic

Manuscript received May 1999; revised September 2001; approved by IEEE/ACM Transactions on Networking Editor E. W. Knightly.

Supported partially by NSF grant NCR96–12781.

Parts of this work have been presented at the *IEEE Conference on Decision and Control*, Tampa, FL, December 1998 and *IEEE Infocom*, New York, NY, March 1999.

Martin Reisslein is with the Department of Electrical Engineering, Telecommunications Research Center, Arizona State University, Tempe, AZ 85287–7206, USA (e–mail reisslein@asu.edu).

Keith W. Ross is with Institute Eurecom, 2229 Route des Cretes, 06904 Sophia–Antipolis, France (e–mail: ross@eurecom.fr).

Srinivas Rajagopal is with Manugistics Group Inc., 585 E. Swedesford Rd., Wayne, PA 19087, USA (e–mail: srajagop@manu.com).

passes through the shared buffer. Therefore, the existing solutions do not extend to the network environment in a satisfactory manner.

Although our approach also uses leaky bucket regulators, it provides meaningful statistical guarantees in a network context. The QoS guarantees provided by our scheme can be roughly stated as follows: the fraction of traffic that exceeds a specific end–to–end delay constraint is below a prescribed bound. The scheme allows each connection *to have its own end–to–end delay constraint and its own bound on the fraction of traffic that exceeds this delay limit.* Such a statistical QoS guarantee is particularly appropriate for continuous–media traffic, whereby timestamping and a playout buffer can ensure the continuous playout of video or audio without jitter [38]. Our traffic management scheme has the following components: (*i*) each connection's traffic is smoothed at the connection's input as much as allowed by the connection's delay constraint; (*ii*) all nodes within the network employ bufferless statistical multiplexing; (*iii*) admission control is based on the worst–case assumption that sources are adversarial to the extent permitted by the connection's regulator, while concurrently assuming the connections generate traffic independently. A critical device in our is scheme is a novel bound for a connection's traffic loss at a single node.

Our scheme has the following features:

- Admission control is solely based on the connections' regulator parameters, which are policable. It is not based on more complex, difficult–to–police statistical characterizations.
- It allows for statistical multiplexing in the network while meeting the QoS requirements. The smoothing at the input increases the statistical multiplexing gain.
- It allows for per–connection QoS requirements: the connections can have vastly different delay and loss requirements.
- Because the multiplexing is bufferless, the switches require only small input buffers (when traffic is packetized), thereby reducing switch cost.
- A connection's traffic characterization does not change as the traffic passes through a bufferless multiplexer, that is, the traffic leaving the network node conforms to the same regulator constraints as the traffic entering the node. This feature is particularly useful when analyzing multihop networks.

The statistical multiplexing within the network increases the connection carrying capacity of the network significantly at the expense of miniscule losses in the network. We provide numerical examples that demonstrate that by allowing for very small losses of the order of 10^{-7} (which can be effectively hidden by error concealment techniques [53]) our scheme can typically support two to three times the number of connections that deterministic service disciplines (GPS, RCS, etc.) can support.

The problem of providing end–to–end statistical QoS guarantees in a network has received a great deal of attention in recent years. The early works [23], [3] in this area derive probabilistic bounds on the delay of flows in a network, while [44] discusses a conceptual framework for QoS assurances in a network. A scheme which is able to provide end–to–end statistical QoS in a network of Generalized Processor Sharing (GPS) schedulers is developed in [14]. End–to–end statistical QoS guarantees are also provided by the scheme proposed in [25], which employs Traffic–Controlled Rate–Monotonic Priority Scheduling [24]. Our approach was developed independently of [14], [25] and was first presented in [41], [42]. In this article we extend our approach and present it in a comprehensive manner. The GPS based scheme [14] is further refined in [22]. Schemes for providing end–to–end statistical QoS in a network of Earliest Deadline First (EDF) schedulers are developed in [1], [49]. A comparison of the EDF based schemes and the GPS based schemes is conducted in [50]. An approach that statistically bounds the burstiness of flows in a network is presented in [51]. A framework for achieving end–to–end statistical QoS through coordinated network scheduling is devised in [29]. In [16] aggregation of flows in core routers of the Internet is exploited to decompose the network and analyze the end–to–end queuing behavior using tools developed for the analysis of a single queue. Finally, there have been several efforts to extend the deterministic network calculus [7], [8], [9], [4], [26], which relies to a large extend on arrival envelopes and service curves, to probabilistic network services. Different definitions of probabilistic service curves have been studied in [10], [36]. A probabilistic network calculus for a class of so–called "dynamic F–servers" is developed in [4]. A calculus for providing end–to–end statistical QoS is developed and evaluated in [2], [30]. This calculus employs effective service curves and applies in rather general settings.

This article is organized as follows. In Section II we formally define the cascaded leaky–bucket regulators and the statistical QoS requirement. We also discuss the smoothers at the network ingresses and describe our network model. In Section III we focus on a single node. We determine the worst–case traffic and outline our smoothing and admission control procedure. We also consider general smoothers and show that the optimal smoother is a single–buffer smoother which smoothes traffic as much as the delay limit permits. In Section III-B we evaluate our smoothing/bufferless multiplexing scheme in the context of a single node numerically using traces of MPEG encoded video. In Section III-C we compare our scheme to designs based on buffered statistical multiplexing. In Section IV we analyze multihop networks. In Section IV-A we compare the performance of our smoothing/bufferless multiplexing scheme with that of deterministic service disciplines in multihop networks. In Section V we discuss how the responsibilities of smoothing, call admission control and traffic policing can be shared by the application and the network when our smoothing/bufferless multiplexing scheme is employed. We conclude in Section VI.

II. Regulated Traffic and the Statistical QoS Requirement

In this article we study networks consisting of interconnected bufferless nodes. We assume a virtual circuit, connection–oriented network and view traffic as fluid, that is, packets are infinitesimal. The fluid model, which closely approximates a packetized model with small packets, permits us to focus on the central issues and significantly simplifies notation.

Each connection j entering the network has an associated *regulator function* (also often referred to as arrival envelope in the literature), denoted by $\mathcal{E}_j(t)$, $t \geq 0$. The regulator function constrains the amount of traffic that connection j can send into the network over all time intervals. Specifically, if $A_j(t)$ is the amount of traffic that connection j sends into the network over the interval $[0, t]$, then $A_j(t)$ is required to satisfy

$$A_j(t+\tau) - A_j(\tau) \leq \mathcal{E}_j(t) \quad \forall \tau \geq 0, \quad t \geq 0. \tag{1}$$

A popular regulator is the simple regulator, which consists of a peak rate controller in series with a leaky bucket; for the simple regulator, the regulator function takes the following form:

$$\mathcal{E}_j(t) = \min\{\rho_j^1 t,\ \sigma_j^2 + \rho_j^2 t\}.$$

For a given source type, the bound on the traffic provided by the simple regulator may be loose and lead to overly conservative admission control decisions. For many source types (e.g., for VBR video), it is possible to get a tighter bound on the traffic and dramatically increase the admission region. In particular, regulator functions of the form

$$\mathcal{E}_j(t) = \min\{\rho_j^1 t,\ \sigma_j^2 + \rho_j^2 t, \ldots,\ \sigma_j^{L_j} + \rho_j^{L_j} t\} \tag{2}$$

are easily implemented with cascaded leaky buckets; it is shown in [54] that the additional leaky buckets can lead to substantially larger admission regions for multiplexing with deterministic QoS. We shall show that this is also true to some extend for multiplexing with a statistical QoS requirement. Specifically, we shall demonstrate that with three properly selected leaky buckets, we can achieve the maximum admission region. With two carefully selected leaky buckets we can achieve most of this admission region; however, in most cases these two leaky buckets differ from the simple regulator in that both leaky buckets have a non–zero bucket depth σ (see Appendix B for details).

Throughout this article we assume that each regulator has the form (2). Without loss of generality we may assume that $\rho_j^1 > \rho_j^2 > \cdots > \rho_j^{L_j}$ and $\sigma_j^2 < \sigma_j^3 < \cdots < \sigma_j^{L_j}$. For ease of notation, we set $\rho_j = \rho_j^{L_j}$. Note that for connection–j traffic, the long–run average rate is no greater than ρ_j and the peak rate is never greater than ρ_j^1.

Each connection also has a QoS requirement. We consider a QoS requirement that is particularly appropriate for multimedia traffic that has stringent end–to–end delay requirements but can tolerate some loss. Specifically, each connection has a connection–specific delay limit and a connection–specific loss bound. Let d_j and ϵ_j denote the delay limit and loss bound for connection j. Any traffic that overflows at one of the bufferless links in the network is considered to have infinite delay, and therefore violates the delay limit. The QoS requirement is as follows: the long–run fraction of connection–j traffic that is delayed by more than d_j seconds must be less than ϵ_j.

This QoS requirement can assure continuous, uninterrupted playback for a multimedia connection as follows. Each packet (which we assume to be infinitesimally small in our fluid analysis) is time–stamped at the source. If a packet from connection j is time–stamped with value x, the packet (if not lost in the node) arrives at the receiver no later than $x + d_j$. The receiver delays playout of the packet until time $x + d_j$. Thus, by including a buffer at each receiver, the receiver can playback a multimedia stream without jitter with a fixed delay of d_j and with a loss probability of at most ϵ_j.

The first aspect of our strategy is to pass each connection's traffic through a buffered smoother at the connection's input to the network. We design the smoother for connection j so that the connection–j traffic is *never* delayed by more than d_j in the smoother. After having smoothed a connection's traffic, we pass the smoothed traffic to the network, and the traffic follows its route through the network. At each link along its route, the connection's traffic is statistically multiplexed with traffic from other connections. The second aspect of our strategy is to remove all of the buffers inside the network; that is, we use bufferless statistical multiplexing rather than buffered multiplexing before each link in the network. In our fluid model, a connection's traffic that arrives to a bufferless link either flows through the link without any delay or overflows at the link, and therefore has infinite delay. The QoS requirement of a connection j is met if the fraction of connection–j traffic that overflows any of the links along the route of connection j is less than ϵ_j. Also, note that provided the loss at each link is small, we can reasonably approximate a connection's traffic at the output of the multiplexer as being identical to its traffic at the input of the multiplexer. In other words, a connection that satisfies a certain regulator constraint at the input of a node satisfies the same regulator constraint at the output of the node.

For the smoother at the input of connection j to the network we initially use a buffer which serves traffic at rate c_j^*. When the smoother buffer is nonempty, traffic is drained from the buffer at rate c_j^*. When the smoother buffer is empty and connection–j's traffic is arriving at a rate less than c_j^*, traffic leaves the buffer exactly at the rate at which it enters the buffer. For the fluid model and QoS criterion of this article we shall show that more complex smoothers consisting of cascaded leaky buckets do not improve performance.

Using the theory developed in [7], it can be shown that the maximum delay in the smoother is

$$\max_{t \geq 0} \left\{ \frac{\mathcal{E}_j(t)}{c_j^*} - t \right\}.$$

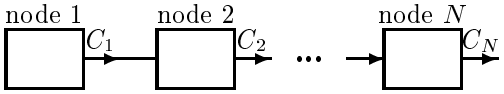


Fig. 1. Multihop network with N nodes.

We set the smoother rate to

$$c_j^* = \min\left\{c_j \geq 0 : \max_{t\geq 0}\left\{\frac{\mathcal{E}_j(t)}{c_j} - t\right\} \leq d_j\right\}, \qquad (3)$$

where d_j is the delay requirement for connection j. Since the bufferless nodes inside the network introduce no additional delay, traffic from connection j that flows through the network without loss has an end–to–end delay of no more than d_j. It is straightforward to show from (3) that the smoother rate can be expressed as

$$c_j^* = \max_{t\geq 0}\frac{\mathcal{E}_j(t)}{d_j + t}. \qquad (4)$$

Intuitively, c_j^* is the smallest smoother rate that guarantees (deterministically) that the traffic is delayed by no more than d_j in the smoother. When considering a plot of the regulator function $\mathcal{E}_j(t)$ and the straight line $c_j^* t$ as a function of time t, c_j^* is the smallest slope c_j^* such that the maximum horizontal distance between $\mathcal{E}_j(t)$ and $c_j^* t$ is less than or equal to d_j.

A. Network Model

An important characteristic of our framework is that it provides statistical QoS guarantees in a *network*. We shall illustrate this characteristic in the context of a multihop network with intervening local traffic flows. Consider a multihop network with N nodes, as shown in Figure 1. Each node is a bufferless multiplexer, that is, buffering is not permitted at each of the N nodes. Let C_n denote the transmission rate for the link between the nth and the $(n+1)$st node.

One connection, which we label connection 0, passes through all N nodes. All of the other connections pass through exactly one node. We denote $I(n)$ for the set of connections that pass through node n. We assume throughout that the traffic generated by the streams is mutually independent. In this paper we shall show how an end–to–end statistical guarantee can be provided to connection–0. To this end, we first solve the single–node case in the following section.

We note that in the considered network, the multiplexed streams are independent at each node. This independence is exploited in our calculation of the bound on the loss probability, which in turn is the basis for our call admission rule. In a more general network, where several streams (that are independent at the network ingress) traverse several nodes together, correlations may be introduced among the streams. However, the bufferless multiplexers introduce correlations among the streams only in case there is loss, i.e., when the aggregate arrival rate of the streams exceeds the link capacity. Otherwise, i.e., when there is no loss, the streams are not "aware" of each other, and the independence is preserved. We expect that in the typical network operating regime the probability of loss is kept quite small, say on the order of 10^{-7} to 10^{-5}, by employing the call admission rule developed in this article. Thus, there are typically only miniscule correlations introduced when several flows traverse a number of common nodes. We expect that these miniscule correlations have a negligible impact on the calculation of the bound on the loss probability.

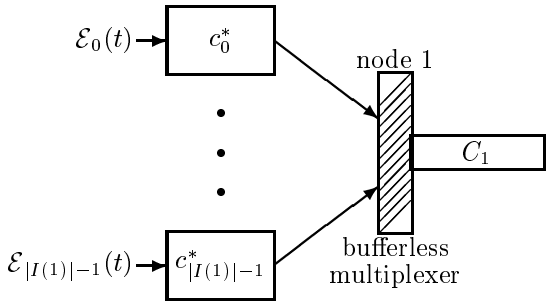


Fig. 2. Node 1 is a bufferless multiplexer. The independent smoothed streams in $I(1)$ are multiplexed onto the output link of capacity C_1.

III. Guaranteeing Statistical QoS: Single Node Analysis

In this section we determine the worst–case traffic and derive the optimal smoothing strategy. For this purpose we initially focus on a particular node n, $1 \leq n \leq N$. $|I(n)|$ smoothed streams are multiplexed onto the output link of capacity C_n. Each of the connections j, $j \in I(n)$, has a regulator function $\mathcal{E}_j(t)$ and QoS parameters d_j and ϵ_j. Now regard the arrival process of stream j to its smoother as a stochastic process. Let $(A_j(t),\ t \geq 0)$ denote the arrival process of the unsmoothed stream j, and let $(A_j(t,\omega),\ t \geq 0)$ denote a realization of the stochastic process. Also let $\mathbf{A}_n(t) = (A_j(t),\ j \in I(n))$, and let $(\mathbf{A}_n(t),\ t \geq 0)$ be the associated vector stochastic arrival process. We say that the vector arrival process $(\mathbf{A}_n(t),\ t \geq 0)$ is *feasible* if (i) the component arrival processes $(A_j(t),\ t \geq 0)$, $j \in I(n)$, are independent, and (ii) for each $j \in I(n)$, each realization $(A_j(t,\omega),\ t \geq 0)$ satisfies the regulator constraint

$$A_j(t+\tau,\omega) - A_j(\tau,\omega) \leq \mathcal{E}_j(t) \quad \forall\ \tau \geq 0, \quad t \geq 0.$$

Denote $\mathcal{A}_n$ for the set of all feasible vector arrival processes $(\mathbf{A}_n(t),\ t \geq 0)$. For a fixed feasible vector arrival process $(\mathbf{A}_n(t),\ t \geq 0)$, let $U_j(t)$ be the rate at which traffic from connection j leaves the associated smoother at time t, and let U_j be the corresponding steady–state random variable. Note that the streams U_j, $j \in I(n)$, may have traversed a number of bufferless nodes before reaching node n. The bufferless nodes do not delay or alter the traffic streams (except for miniscule losses due to link overflow which are negligible in typical networking scenarios). Consider multiplexing the streams U_j, $j \in I(n)$, onto the bufferless link

of capacity C_n. The long–run average fraction of traffic lost by connection j is

$$P_{\text{loss}}^{\text{info,n}}(j) = \frac{E\left[(\sum_{i\in I(n)} U_i - C_n)^+ \frac{U_j}{\sum_{i\in I(n)} U_i}\right]}{E[U_j]}, \qquad (5)$$

where $(x)^+ = \max(0, x)$. The definition of $P_{\text{loss}}^{\text{info,n}}(j)$ relies on the natural assumption that traffic loss at multiplexer n is split between the sources in a manner proportional to the rate at which the sources send traffic into the multiplexer. Note that $P_{\text{loss}}^{\text{info,n}}(j)$ keeps track of loss for each individual connection.

Although $P_{\text{loss}}^{\text{info,n}}(j)$ is an appealing performance measure, we have found it to be mathematically unwieldy. Instead of $P_{\text{loss}}^{\text{info,n}}(j)$ we shall work with a bound on $P_{\text{loss}}^{\text{info,n}}(j)$ which is more tractable and which preserves the essential characteristics of the original performance measure. Noting that the term in the expectation of the numerator is non–zero only when $\sum_{i\in I(n)} U_i > C_n$, we obtain the following bound on $P_{\text{loss}}^{\text{info,n}}(j)$:

$$P_{\text{loss}}^{\text{info,n}}(j) \le \frac{E\left[(\sum_{i\in I(n)} U_i - C_n)^+ U_j\right]}{C_n \cdot E[U_j]} := P_{\text{loss}}^{n}(j). \qquad (6)$$

In most practical circumstances the QoS requirement specifies traffic loss to be miniscule, on the order of $\epsilon_j = 10^{-6}$ or less. Thus we expect the bound to be very tight: In the rare event when the aggregate demand for bandwidth $\sum_{i\in I(n)} U_i$ exceeds the link capacity C_n, $\sum_{i\in I(n)} U_i$ is typically very close to C_n. In Section III-B we provide numerical results which show that $P_{\text{loss}}^{n}(j)$ is very nearly equal to the actual loss probability $P_{\text{loss}}^{\text{info,n}}(j)$. Henceforth, we focus on the bound $P_{\text{loss}}^{n}(j)$, and we refer to $P_{\text{loss}}^{n}(j)$ as the *loss probability for connection j at node n*. We emphasize here that the bound (6) is a crucial and important step for the techniques taken in this paper. To our knowledge, no other authors have made direct use of this important bound.

By taking the supremum over all the feasible vector stochastic processes, we obtain the following worst–case loss probability for connection j at node n:

$$\phi_j^{*n} = \sup_{\mathcal{A}_n} \frac{E\left[(\sum_{i\in I(n)} U_i - C_n)^+ U_j\right]}{C_n \cdot E[U_j]}. \qquad (7)$$

The loss probability of connection j at node n is guaranteed to be bounded by ϕ_j^{*n} for all feasible vector arrival processes in $\mathcal{A}_n$, that is, for all independent arrival processes whose sample paths satisfy the regulator constraints.

As a first step in computing the ϕ_j^{*n}'s, we need to explicitly determine the random variables $U_j,\ j \in I(n)$, that attain the supremum in (7).

Lemma 1: Let $U_j^*,\ j \in I(n)$, be independent random variables, with U_j^* having distribution

$$U_j^* = \begin{cases} c_j^* & \text{with probability } \frac{\rho_j}{c_j^*} \\ 0 & \text{with probability } 1 - \frac{\rho_j}{c_j^*} \end{cases}.$$

There exists a feasible vector arrival process which produces the steady–state rate variables $U_j^*,\ j \in I(n)$, at the smoother outputs.

Proof: The proof is by construction. For each $j \in I(n)$ let

$$t_j = \frac{\sigma_j^2}{\rho_j^1 - \rho_j^2}$$

and

$$T_j = \frac{\rho_j^1 \sigma_j^2}{(\rho_j^1 - \rho_j^2)\rho_j}.$$

Also let $\theta_j,\ j \in I(n)$, be independent random variables with θ_j uniformly distributed over $[0, T_j]$. For each $j \in I(n)$ let $b_j(t)$ be a deterministic periodic function with period T_j such that

$$b_j(t) = \begin{cases} \rho_j^1 & 0 \le t < t_j \\ 0 & t_j \le t \le T_j \end{cases}.$$

For each $j \in I(n)$ define an stochastic arrival process as

$$A_j(t) = \int_0^t b_j(s + \theta_j) ds.$$

Thus each component arrival process $(A_j(t),\ t \ge 0)$ is generated by a periodic on–off source; process j has peak rate ρ_j^1 and average rate ρ_j. By sending each component process $(A_j(t),\ t \ge 0)$ into its respective smoother, we obtain an on–off process whose peak rate is c_j^* and whose average rate is ρ_j. This on–off process is not altered by passing through bufferless nodes. Also, the component processes are independent; thus the vector arrival process produces the steady–state random variables $U_j^*,\ j \in I(n)$, at the smoother outputs.

It remains to show that each realization of $(A_j(t),\ t \ge 0)$ satisfies the regulator constraint (1). It follows immediately from the definition of $b_j(t)$ that

$$\int_0^t b_j(s) ds \le \mathcal{E}_j(t) \text{ for all } 0 \le t \le T_j. \qquad (8)$$

We can, in fact, show that

$$\int_0^t b_j(s) ds \le \mathcal{E}_j(t) \text{ for all } t \ge 0. \qquad (9)$$

To see this consider any arbitrary $t = nT_j + s$, where n is some non–negative integer and $0 \le s \le T_j$. We have

$$\begin{aligned} \int_0^t b_j(s) ds &= \int_0^{T_j} b_j(s) ds + \cdots + \int_{(n-1)T_j}^{nT_j} b_j(s) ds + \\ &\qquad \int_{nT_j}^{nT_j + s} b_j(s) ds \\ &\le nT_j\rho_j + \mathcal{E}_j(s) \\ &\le [\mathcal{E}_j(nT_j + s) - \mathcal{E}_j(s)] + \mathcal{E}_j(s) \\ &= \mathcal{E}_j(t)\ . \end{aligned}$$

The first inequality follows from (8) and from the fact that the average rate of $b_j(t)$ over any period of length T_j is ρ_j.

The second inequality follows because the slope of $\mathcal{E}_j(t)$ is never less than ρ_j. This establishes (9). Finally because $b_j(t)$ is non–increasing over each of its periods, we have

$$\int_\tau^{t+\tau} b_j(s)ds \le \int_0^t b_j(s)ds \text{ for all } \tau \ge 0, \quad t \ge 0. \qquad (10)$$

Combining (9) and (10) proves that each realization of $(A_j(t),\ t \ge 0)$ satisfies the regulator constraint (1). ■

We now show that the random variables $U_j^*,\ j \in I(n)$, attain the supremum in (7). This result will lead to a simple procedure for calculating the worst–case loss probabilities $\phi_j^*,\ j \in I(n)$. To this end, we will need to make use of a concept from stochastic ordering. A random variable X is said to be smaller than a random variable Y in the sense of the *increasing convex stochastic (ics) ordering*, written as $X \le_{icx} Y$, if $E[h(X)] \le E[h(Y)]$ for all increasing, convex functions $h(\cdot)$.

Theorem 1: For each $j \in I(n)$, the worst–case loss probability for connection j at node n is

$$\phi_j^{*n} = \frac{E\left[(\sum_{i\in I(n)} U_i^* - C_n)^+ U_j^*\right]}{C_n \cdot E[U_j^*]}$$

Proof: Let $\mathcal{U}_n$ be the set of all random vectors $(U_j,\ j \in I(n))$ such that
1. $U_j,\ j \in I(n)$, are independent.
2. $0 \le E[U_j] \le \rho_j$ and $0 \le U_j \le c_j^*$ for all $j \in I(n)$.
All feasible vector arrival processes in $\mathcal{A}_n$ give steady–state rate variables that belong to $\mathcal{U}_n$. Let $(U_j,\ j \in I(n))$ be a random vector in $\mathcal{U}_n$. Let $U = \sum_{i\in I(n)} U_i$ and $U^* = \sum_{i\in I(n)} U_i^*$. We need to show that

$$\frac{E[(U - C_n)^+ U_j]}{C_n \cdot E[U_j]} \le \frac{E[(U^* - C_n)^+ U_j^*]}{C_n \cdot E[U_j^*]}. \qquad (11)$$

Fix k, with $k \in I(n)$, and consider the random vector $(\hat{U}_j,\ j \in I(n))$ such that $\hat{U}_k = U_k^*$ and $\hat{U}_j = U_j$ for $j \ne k$. Note that $(\hat{U}_j,\ j \in I(n)) \in \mathcal{U}_n$. We first show that for each fixed j,

$$\frac{E[(U - C_n)^+ U_j]}{C_n \cdot E[U_j]} \le \frac{E[(\hat{U} - C_n)^+ \hat{U}_j]}{C_n \cdot E[\hat{U}_j]}. \qquad (12)$$

Consider the case $i \ne j$. Let $V = U - U_i - U_j$. Let $F_V(\cdot)$ and $F_{U_j}(\cdot)$ be the distribution functions for V and U_j. Noting that U_i, U_j and V are independent, we have

$$\begin{aligned} E[(U - C_n)^+ U_j] &= E[(U_i + V + U_j - C_n)^+ U_j] \\ &= \int_0^\infty \int_0^\infty E[(U_i + v + u - C_n)^+ u] \\ &\qquad dF_V(v) dF_{U_j}(u) \end{aligned}$$

The function $f(x) = (x + v + u - C_n)^+ u$ within the expectation is an increasing, convex function in x for each fixed v and u. Thus, because $U_i \le_{icx} \hat{U}_i$ (e.g., see Proposition 1.5.1 in [52]), we have

$$E[(U_i + v + u - C_n)^+ u] \le E[(\hat{U}_i + v + u - C_n)^+ u]$$

for all v and u. Combining the above two equations gives

$$E[(U - C_n)^+ U_j] \le E[(\hat{U} - C_n)^+ \hat{U}_j],$$

which, when combined with $E[\hat{U}_j] = E[U_j]$, gives (12).

Now consider the case $i = j$. Let $W = U - U_i$. Using $U_i \le c_i^*$, the independence of W and U_i, and the independence of W and $\hat{U}_i$, we obtain

$$\begin{aligned} \frac{E[(U - C_n)^+ U_i]}{C_n \cdot E[U_i]} &= \frac{E[(W + U_i - C_n)^+ U_i]}{C_n \cdot E[U_i]} \\ &\le \frac{E[(W + c_i^* - C_n)^+]}{C_n} \frac{E[U_i]}{E[U_i]} \\ &= \frac{E[(W + c_i^* - C_n)^+]}{C_n} \frac{E[\hat{U}_i]}{E[\hat{U}_i]} \\ &= \frac{E[(W + c_i^* - C_n)^+ \hat{U}_i]}{C_n \cdot E[\hat{U}_i]}. \end{aligned}$$

Also

$$\begin{aligned} E[(\hat{U} - C_n)^+ \hat{U}_i] &= E[(W + \hat{U}_i - C_n)^+ \hat{U}_i] \\ &= E[(W + c_i^* - C_n)^+ \hat{U}_i]. \end{aligned}$$

Combining the above two equations gives (12) for $i = j$.

Thus (12) holds for all $i \in I(n)$. Therefore, starting with the original vector $(U_0, U_1, \ldots, U_{|I(n)|-1}) \in \mathcal{U}_n$ we can replace U_0 with U_0^* and obtain a new vector in $\mathcal{U}_n$ such that (12) holds. Rename this new vector as $(U_0, U_1, \ldots, U_{|I(n)|-1})$. We can repeat the procedure, this time replacing U_1 with U_1^*, and again obtaining a new vector in $\mathcal{U}_n$ such that (12) holds. Performing this procedure for all $i = 0,\ 1, \ldots, |I(n)| - 1$ gives (11). ■

Exploiting the fact that the U_j^*'s are Bernoulli random variables, we can simplify the expression for ϕ_j^{*n}:

$$\phi_j^{*n} = \frac{E\left[(\sum_{i\in I(n)-\{j\}} U_i^* + c_j^* - C_n)^+\right]}{C_n}. \qquad (13)$$

These bounds can be computed by convolving the distributions of the independent random variables. An approximate convolution algorithm is described in [28]. However, convolution often leads to numerical problems. We therefore apply the Large Deviation (LD) approximation, which is known to be accurate and also computationally very efficient [43], [13], [15], [40], to the expectation in the numerator. Towards this end, let $\mu_{U_i^*}(s)$ denote the logarithm of the moment generating of U_i^*:

$$\mu_{U_i^*}(s) := \ln E[e^{sU_i^*}].$$

We define

$$U^* := \sum_{i\in I(n)-\{j\}} U_i^*.$$

Note that

$$\mu_{U^*}(s) = \sum_{i\in I(n)-\{j\}} \mu_{U_i^*}(s)$$

by the independence of the U_i^*'s. The large deviation (LD) approximation gives the following approximation for ϕ_j^{*n} [43]

$$\frac{1}{C_n s^{\star 2}\sqrt{2\pi\mu''_{U^*}(s^\star)}} e^{-s^\star(C_n - c_j^*)+\mu_{U^*}(s^\star)},$$

where $s^\star$ is the unique solution to

$$\mu'_{U^*}(s^\star) = C_n - c_j^*.$$

In summary, (13) is a simple expression for the worst–case loss probability of connection j at node n; this expression involves the independent Bernoulli random variables $U_j^*,\ j \in I(n)$, whose distributions we know explicitly. The LD approximation for (13) is highly accurate and is easily calculated. We note that an admission rule based on on–line traffic measurements for the smoothing/bufferless multiplexing scheme proposed in this article is studied in [39].

At this juncture we note some important related work by Doshi [11], [12]. He studies worst–case, unsmoothed traffic that maximizes an aggregate loss ratio, where the aggregation is taken over all sources. For this criterion he discovers a number of anomalies; in particular, extremal on–off sources are not always worst case. With our bound $P_{\text{loss}}^n(j)$ (6) the loss is maximized by the extremal on–off sources, which greatly simplifies admission control. Furthermore, as we show in this article, smoothing of traffic can significantly expand the admission region.

A. The Optimal Smoother

Up to this point we have assumed that the smoother for each connection j consists of a single buffer that limits the peak rate of the smoother output to c_j^*. In this subsection we study more general smoothers, namely, smoothers that consist of a cascade of leaky buckets. The smoother for connection j, defined by a function $S_j(t)$, constrains the amount of traffic that can enter the network over any time interval. Specifically, if $B_j(t)$ is the amount of traffic leaving smoother j over the interval $[0,t]$, then $B_j(t)$ is required to satisfy

$$B_j(t+\tau) - B_j(\tau) \le S_j(t) \quad \text{for all } t \ge 0,\ \tau \ge 0.$$

We assume throughout this section that the smoother functions are of the form

$$S_j(t) = \min_{1 \le k \le M_j} \{s_j^k + r_j^k t\} \tag{14}$$

with $r_j^1 > r_j^2 > \cdots > r_j^{M_j}$ and $0 = s_j^1 < s_j^2 < \cdots < s_j^{M_j}$. These piecewise linear, concave smoother functions can be easily implemented by a cascade of leaky buckets. The single–buffer smoother defined in Section 2 is a special case with $M_j = 1,\ s_j^1 = 0$ and $r_j^1 = c_j^*$.

We say that a set of smoothers $(S_j(t),\ j \in I(n))$ is *feasible* if the maximum delay incurred at smoother j is $\le d_j$ for all $j \in I(n)$. By definition the set of smoothers $(c_j^* t,\ j \in I(n))$ studied earlier is feasible. Now fix a feasible set of smoothers $(S_j(t),\ j \in I(n))$, and let the regulated traffic from the connections in $I(n)$ pass through these smoothers. Let

$$\phi_j^n = \sup_{\mathcal{A}_n} \frac{E\left[(\sum_{i\in I(n)}^J U_i - C_n)^+ U_j\right]}{C_n \cdot E[U_j]} \tag{15}$$

be the associated worst–case loss probability for connection j at node n. Recall that ϕ_j^{*n} is the same worst–case loss probability but with the traffic passing through the set of smoothers $(c_j^* t,\ j \in I(n))$. The proof of the following result is provided in the appendix.

Theorem 2: $\phi_j^{*n} \le \phi_j^n$ for all $j \in I(n)$. Thus the single–buffer smoothers with $c_j = c_j^*$ minimize the worst–case loss probability over all feasible sets of smoothers.

It follows from Theorem 2 that the more complex smoothers consisting of cascaded leaky buckets do not increase the connection carrying capacity of node n. Thus without loss of performance, we may use the simple smoothers of the form $(c_j t,\ j \in I(n))$. Furthermore, Theorem 2 verifies the intuition that in order to maximize the admission region of node n the smoother rates are as small as the delay constraints permit, that is, $c_j = c_j^*$ for $j \in I(n)$.

B. Numerical Experiments for a Single Node

In this section we evaluate the smoothing/bufferless multiplexing scheme in the context of a single node. We set $N = 1$ and focus on the network consisting of smoothers and one bufferless multiplexer as depicted in Figure 2. We set the capacity of the output link to $C_1 = 45$ Mbps. In this single node scenario admission control is particularly simple: we evaluate ϕ_j^{*1} (13) using the LD approximation and verify whether $\phi_j^{*1} \le \epsilon_j\ \forall j \in I(1)$. We evaluate our scheme using traces from MPEG encoded movies. We obtained the frame size traces, which give the number of bits in each video frame, from the public domain [45]. (We are aware that these are low resolution traces and some critical frames are dropped; nevertheless, the traces are extremely bursty.) The movies were compressed with the Group of Pictures (GOP) pattern IBBPBBPBBPBB at a frame rate of $F = 24$ frames/sec [45]. Each of the traces has $M =$ 40,000 frames, corresponding to about 28 minutes. The mean number of bits per frame and the peak–to–mean ratio are given in Table I. Let $x_m,\ m = 1, \ldots, M$, denote

TABLE I
STATISTICS OF MPEG–1 TRACES.

Trace	Mean (bit) bits/frame	Mean kbits/sec	Peak/Mean
lambs	7,312	171.2	18.4
mr.bean	17,647	423.5	13.0

the size of the mth frame in bits. We convert the discrete frame size trace to a fluid flow by transmitting the mth frame at rate $x_m F$ over the interval $[(m-1)/F, m/F]$.

We compute the empirical envelope and the concave hull of each trace using the algorithms of Wrege *et al.* [54]. Based on the concave hull of each video we compute the minimal smoother rate c_j^*. We also apply the heuristic of Appendix B to the concave hull in order to find the optimal leaky bucket characterization with 2 and more leaky buckets. We then compute the minimal smoother rate c_j^* based on these concise leaky bucket characterizations.

Assuming worst–case on–off traffic, the smoother outputs are statistically multiplexed onto the bufferless link. We set $\epsilon_j = 10^{-7}$ for all connections. In Figure 3 we plot the number of admissible video connections as a function of the delay bound. The graph gives the number of admissible video connections when the videos are characterized by the concave hull or the optimal leaky bucket characterization with 2 leaky buckets (which is obtained with the heuristic of Appendix B). We observe from the plots that the optimal leaky bucket characterization with 2 leaky buckets admits almost as many video connections as the more accurate concave hull characterization. The curves for 3 or more leaky buckets coincide with the curve for the concave hull.

In the next experiment we compare the admission region of our approach with the admission region obtained with the deterministic admission control condition of Wrege *et al.* [54]. The approach of Wrege *et al.* is to feed the unsmoothed traffic into a buffered multiplexer. The deterministic admission control condition guarantees that no bit is delayed by more than the prespecified delay limit in the multiplexer buffer (and it also guarantees that no bit is lost). Our approach, on the other hand, exploits the independence of traffic emanating from the connections in $I(1)$. The videos are passed through simple smoothers with $c_j = c_j^*$. The smoother outputs — assuming worst–case on–off traffic — are then statistically multiplexed onto the bufferless link (see Figure 2). We set $\epsilon_j = 10^{-7}$ for all connections. Losses this small have essentially no impact on the perceived video quality and can be easily hidden by error concealment techniques [53].

In Figure 4 we plot the number of admissible lambs connections as a function of the delay bound. The graph gives the number of lambs connections that are admitted with the our approach (RRR) when 2 or 3 leaky buckets (LB) are used to characterize the video trace. As we just saw in Figure 3 the optimal leaky bucket characterization with 3 leaky buckets admits as many connections as the concave hull, the most accurate, concave characterization of the video; using more leaky buckets does not increase the admission region. We also plot the number of lambs connections that are admitted with the buffered deterministic multiplexing approach of Wrege *et al.* (KLZ) when 2, 3, 8 or 16 leaky buckets are used to characterize the trace. We observe that for delays on the order of 0.5 seconds or more, the number of admissible connections significantly increases as the number of leaky buckets used to describe the trace increases. The approach of Wrege *et al.* thus greatly benefits from a more accurate characterization of the video — achieved by more leaky buckets.

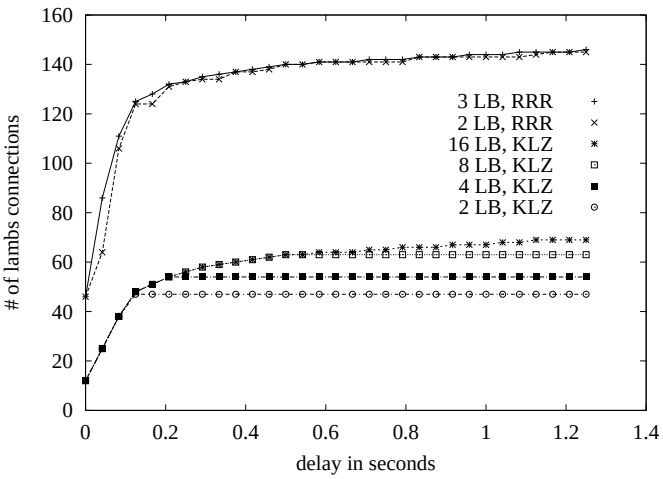


Fig. 4. Number of lambs connections as a function of the delay bound and the number of leaky buckets (LB). Plots shown are for Wrege *et al.* (KLZ) and our approach (RRR).

The main result of this experiment, however, is that our approach allows for more than twice the number connections than does the approach of Wrege *et al.* For example, for a delay bound of 1.1 seconds, Wrege *et al.* admit 69 connections (= 29.6 % average link utilization) with 16 leaky buckets while our approach admits 146 connections (= 62.7 % average link utilization) with 3 leaky buckets. We obtain this dramatic increase in the admission region by exploiting the independence of the sources and allowing for a small loss probability.

In Figure 5 we consider multiplexing two different movies, beans and lambs, each with its own delay constraint. We again consider a single node with $C_1 = 45$ Mbps. We use delay bounds of $d_{\text{lambs}} = 125$ ms or 1.25 seconds and $d_{\text{bean}} = 125$ ms or 1.25 seconds, giving four combinations. Both videos are characterized by 3 leaky buckets. We assume that both video connections have the QoS requirement that the fraction of traffic that is delayed by more than the imposed delay limit is less than 10^{-7}. For the Wrege *et al.* plot we use Earliest Deadline First (EDF) scheduling. We see that for all four cases, the admission region for our approach is dramatically larger.

In Figure 6 we compare the actual loss probability at node 1, $P_{\text{loss}}^{\text{info},1}(j)$ given by (5) with our bound for loss probability, $P_{\text{loss}}^{1}(j)$, given by (6). We obtain $P_{\text{loss}}^{\text{info},1}(j)$ and $P_{\text{loss}}^{1}(j)$ by simulation, and assume worst–case on–off traffic. We also verify the accuracy of the large deviation approximation for $P_{\text{loss}}^{1}(j)$. In Figure 6 we plot the loss probabilities as a function of the number of connections being multiplexed onto the $C_1 = 45$ Mbps link. We consider the scenario where the videos have a delay bound of 1 second and are characterized by 3 leaky buckets. We observe that the bound on the loss probability $P_{\text{loss}}^{1}(j)$ (solid line) tightly bounds the actual loss probability $P_{\text{loss}}^{\text{info},1}(j)$ (dotted line). We also observe that the LD approximation (dashed line) closely approximates the simulation results.

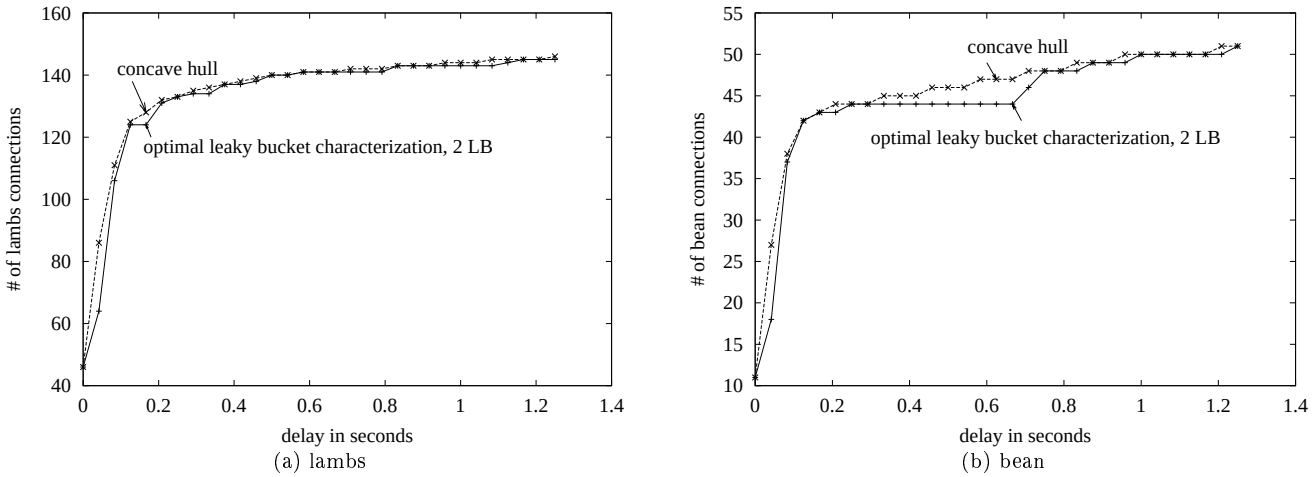


Fig. 3. Number of video connections as a function of the delay bound. The videos are characterized by the concave hull or the optimal leaky bucket characterization with 2 leaky buckets. The bound on the loss probability is 10^{-7}.

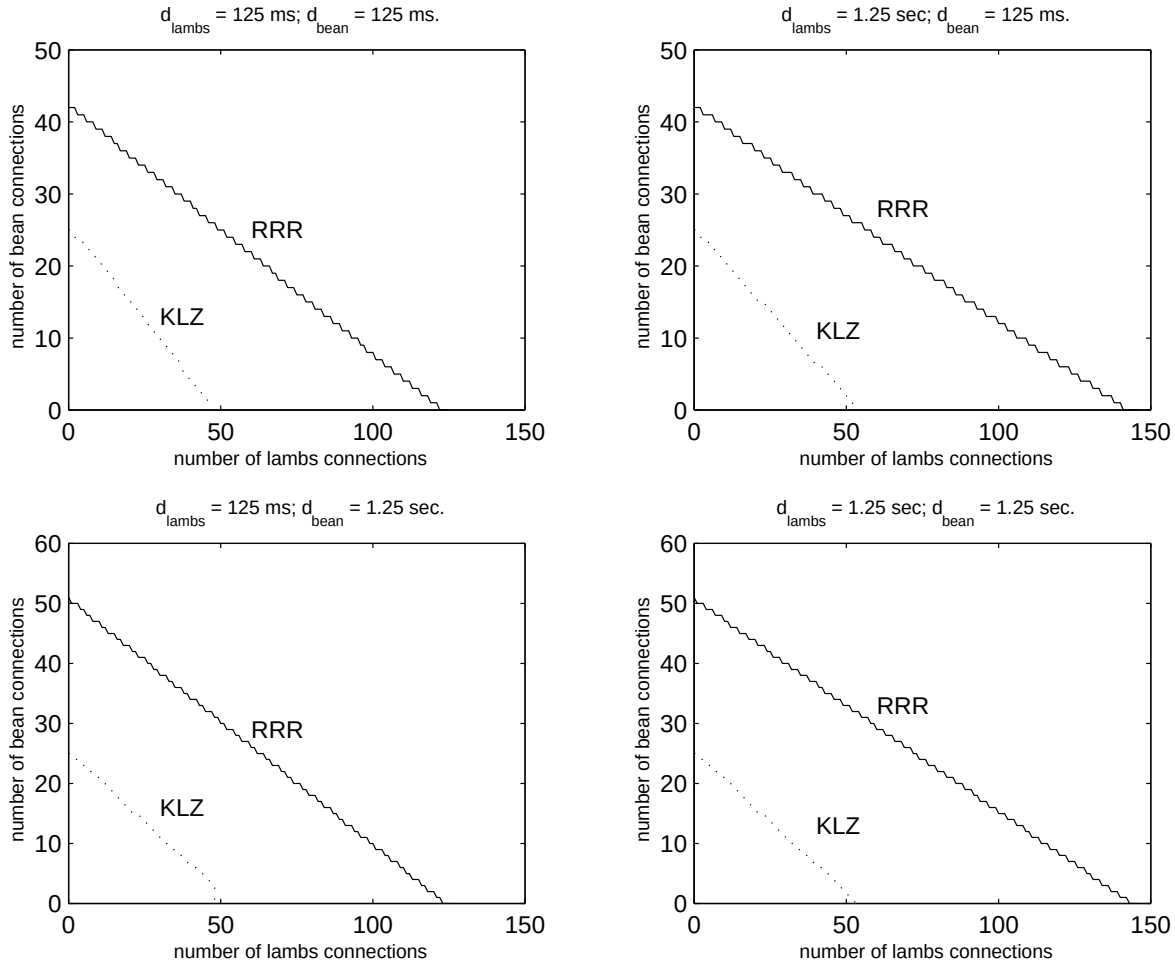
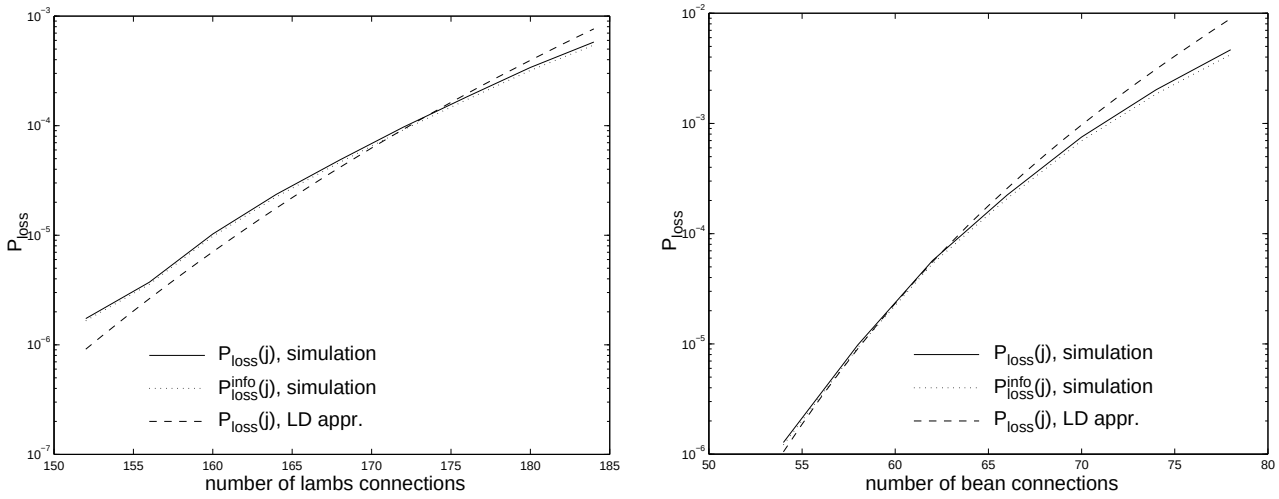


Fig. 5. Admission region for the multiplexing of lambs and bean connections over a 45 Mbps link.

C. Comparison with Buffered Statistical Multiplexing

The numerical results of the previous section show that for a single node our approach allows for dramatically more connections than buffered deterministic multiplexing. In this section we briefly consider buffered multiplexing with an allowance of small loss probabilities, which we refer to as *buffered statistical multiplexing*. Consider the buffered analogy of the single–link bufferless system studied in Sec-



(a) lambs (b) bean

Fig. 6. The simulation verifies that the bound on the loss probability $P^1_{\text{loss}}(j)$ tightly bounds the actual loss probability $P^{\text{info},1}_{\text{loss}}(j)$. The plots further confirm the accuracy of the Large Deviation (LD) approximation. We use a delay bound of 1 second and characterize the videos by 3 leaky buckets. The link rate is 45 Mbps. The plots give the loss probability as a function of the number of ongoing connections.

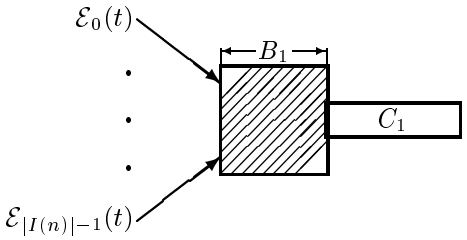


Fig. 7. The traffic of connection j is characterized by the regulator function $\mathcal{E}_j(t)$ and fed directly, i.e. unsmoothed, into a buffered multiplexer.

tion III-B. The link has capacity C_1 and is preceded by a finite buffer of capacity B_1. Let the same connections in $I(1)$ arrive to this system; specifically the connections in $I(1)$ are independent and connection j, $j \in I(1)$, is regulated by a given regulator function $\mathcal{E}_j(t)$. The traffic from the connections in $I(1)$ passes directly into the buffered multiplexer, i.e., the traffic is not pre–smoothed before arriving at the buffer. This buffered system is illustrated in Figure 7. Assuming that traffic is served FIFO, the maximum delay in this system is $d = B_1/C_1$. Suppose that the buffer overflow probability is constrained to be no greater than ϵ.

It is a difficult and challenging problem to accurately characterize the admission region for a buffered multiplexer which multiplexes regulated traffic and which allows for statistical multiplexing. Elwalid *et al.* in [15] made significant progress in this direction. They consider the buffered multiplexer for the *special case of regulators with two leaky buckets*, i.e., for $\mathcal{E}_j(t) = \min\{\rho^1_j t,\ \sigma_j + \rho_j t\}$. (In our numerical comparisons, we extend their theory to the case of multiple cascaded leaky buckets.) In order to make the buffered multiplexer mathematically tractable they assign each connection its own virtual buffer/trunk system. Each virtual buffer/trunk system is allocated buffer $b_{0,j}$ and bandwidth $e_{0,j}$. The allocations are based on the buffer and bandwidth resources (B_1 and C_1, respectively) and on the regulator parameters (ρ_j, ρ^1_j, and σ_j) for the input traffic. It turns out that the bandwidth $e_{0,j}$ is exactly the c^*_j obtained by setting $d_j = d = B_1/C_1$ in (4). After some analysis Elwalid *et al.* obtain the following bound on the fraction of time during which loss occurs at the buffered multiplexer:

$$P^{\text{EMW}}_{\text{loss}} = P(\sum_{j \in I(1)} U^*_j > C_1).$$

where U^*_j, $j \in I(1)$, are exactly the same random independent random variables that occur in Theorem 1. (To calculate the associated c^*_j, $j \in I(1)$, set $d_j = d = B_1/C_1$ for each connection j.)

This observation indicates that our smoother/bufferless multiplexer system has remarkable similarities with the buffered system in [15]. Specifically, for a fixed maximum delay d in the buffered system, we can design a bufferless system with pre–smoothers which has the same maximum delay and which has an admission region based on the same set of independent random variables U^*_j, $j \in I(1)$. The pre–smoothers essentially implement the virtual buffer/trunk systems introduced by Elwalid *et al.* For a maximum loss probability of ϵ the admission region for the buffered multiplexer is defined by

$$P(\sum_{j \in I(1)} U^*_j > C_1) \le \epsilon,$$

whereas the admission region for the bufferless system is

$$\frac{E[(\sum_{i \in I(1)} U^*_i - C_1)^+ U^*_j]}{C_1 \cdot E[U^*_j]} \le \epsilon.$$

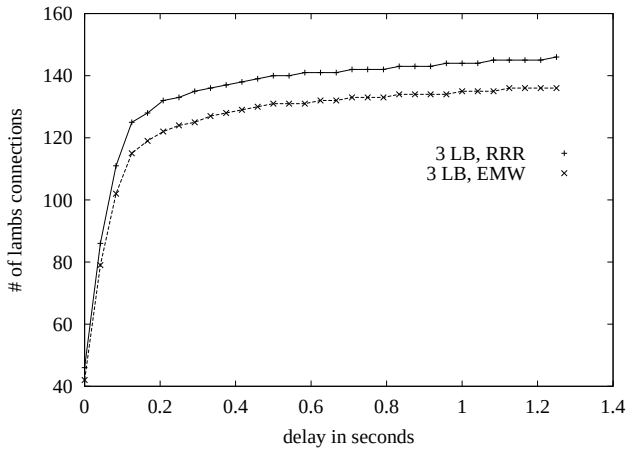


Fig. 8. Number of lambs connections as a function of the delay bound. The lambs video is described by 3 leaky buckets. Plots shown are for Elwalid *et al.* (EMW) and our approach (RRR). The difference in the number of admissible connections is due to the different notions of loss probability.

Although these admission regions are different, they are based on exactly the same independent random variables $U_j^*,\ j \in I(1)$. The difference in these admission regions is an artifact of using two different notions of loss probability: while in this article we use "fraction of traffic lost", the article [15] uses "the fraction of time during which loss occurs". If the same notions of loss were used, then the admission regions would be identical. Figure 8 gives the number of lambs connections that are admitted with the approach of Elwalid *et al.* (EMW) [15] and our approach (RRR) when 3 leaky buckets are used to characterize the trace. We assume $C_1 = 45$ Mbps and set $\epsilon_j = 10^{-7}$ for all connections.

Thus, in the context of a single node our bufferless system has essentially the same admission region as the buffered system in [15] for a fixed worst–case delay d and loss probability ϵ. While being no more difficult to perform call admission, we believe that the bufferless system has some important advantages over the buffered system: (*i*) no buffer is needed at the multiplexer (for packetized traffic, a relatively small buffer would be needed); (*ii*) the bufferless approach allows for a per–connection QoS requirement, whereas the buffered system imposes the same QoS requirement on all connections; and (*iii*), perhaps most importantly, networks are quite tractable for bufferless links, as we can reasonably approximate a connection's traffic at the output of the multiplexer as being identical to its traffic at the input to the multiplexer. This fact is exploited in the next section where we analyze our scheme for general multihop networks.

We conclude this section by noting that the buffered system does have some advantages over the bufferless system. First, although both systems have the same worst–case delay, the buffered system has a lower average delay. (Note, however, that multimedia applications are typically designed for a delay bound.) Second, due to statistical buffer sharing among streams, the buffered system has the potential to admit more streams (see [56] for a quantitative evaluation of this potential). However, exploiting this potential requires admission rules that are typically more complex (e.g., [33], [37], [55]).

IV. Guaranteeing Statistical QoS: Multihop Analysis

We now turn our attention to the entire multihop network. Without loss of generality we focus on connection 0 traversing nodes 1 through N. At the output of any of the nodes, connection 0 has a peak rate no larger than c_0^* and an average rate no larger than ρ_0. We can therefore use (13) to calculate the worst–case loss probability ϕ_0^{*n} at any of the bufferless multiplexers $n,\ n = 1, \ldots, N$. The end–to–end loss probability of connection 0 is bounded by the sum of the worst–case loss probabilities of the individual hops along connection 0's path, that is, the loss in the network is bounded by $\sum_{n=1}^{N} \phi_0^{*n}$.

We note here that the single buffer serving traffic at rate c_j^* which was shown to minimize ϕ_j^{*n} at a single node n in Theorem 2 also minimizes the sum of the ϕ_j^{*n}. To see this, recall that the design of the smoother for connection j depends only on the connection parameters (the regulator function $\mathcal{E}_j(t)$ and the delay limit d_j). Therefore, the *same* smoother minimizes the ϕ_j^{*n} at every node n along connection j's path. As a consequence the single buffer smoother with rate c_0^* minimizes $\sum_{n=1}^{N} \phi_0^{*n}$, the bound on the overall fraction of overflowing connection–0 traffic in the network.

The end–to–end QoS requirement of connection 0 is met if

$$\sum_{n=1}^{N} \phi_0^{*n} \leq \epsilon_0. \tag{16}$$

For admission control, we must ensure that (16) holds for all connections. Specifically, we must partition — either statically or dynamically — the loss constraint ϵ_j among the nodes traversed by each of the connections. This problem is of independent interest and is discussed in Sections 5.10 and 5.11 of [46].

We have thus provided a framework for providing end–to–end statistical QoS guarantees for a multihop network. The framework consists of input smoothers at the network ingresses and bufferless statistical multiplexing within the network. Increasing the number of nodes a connection traverses increases the loss probability but not the delay. Roughly speaking, the network loss probability for a connection is approximately the loss probability of a typical node multiplied by the number of nodes through which a connection passes. Because the loss probability of a node is dimensioned to be on the order of 10^{-6} or less, the increased loss is only of minor importance.

We note at this juncture that $\sum_{n=1}^{N} \phi_0^{*n}$ also provides a bound on the probability that a bit of connection 0 experiences an end–to–end delay of more than d_0 in the network. More formally, with D_0 denoting the end–to–end delay incurred by a bit of connection 0 in the network, we have

$$P(D_0 > d_0) \leq \sum_{n=1}^{N} \phi_0^{*n}. \tag{17}$$

Recall from Section II that by design a bit of connection 0 is delayed by at most d_0 in the smoother. Bits that do not overflow at any of the bufferless links in the network incur no additional delay while bits that do overflow are considered to have infinite delay. The bound (17) follows by noting that $\sum_{n=1}^{N} \phi_0^{*n}$ is a bound on the fraction of bits that do overflow. We emphasize that the bound on the probability that a bit violates a given delay limit is minimized by smoothing as much as the delay limit permits at the network ingress. We compare the performance of our smoothing/bufferless multiplexing scheme with that of deterministic traffic management schemes in the next subsection. These deterministic schemes are lossless and guarantee that a specific delay limit d_j is never violated, that is, they guarantee that $D_j \leq d_j$ with probability one.

In order to facilitate the comparison of the performance with the deterministic benchmarks we make the following simplifying assumptions about the traffic streams and the network. First, we assume that all streams are regulated by a single leaky bucket; for the single leaky bucket, the regulator function takes the following form:

$$\mathcal{E}_j(t) = \sigma_j + \rho_j t.$$

Note that the single leaky bucket regulator constrains the long–run average rate of connection j to be no greater than ρ_j. The multihop analysis of our traffic management scheme for more complex regulators consisting, for instance, of a cascade of leaky buckets is a straightforward extension of the analysis presented here. However, GPS which we shall use as a benchmark to evaluate our scheme, has been analyzed extensively in [34], [35] for single leaky bucket regulators. We will make use of some of those analytical results in our performance evaluation and focus therefore on single leaky bucket regulators throughout this section. For the regulator function $\mathcal{E}_j(t) = \sigma_j + \rho_j t$ and the delay limit d_j we obtain from (4) the smoother rate

$$c_j^* = \max\left(\frac{\sigma_j}{d_j},\ \rho_j\right).$$

To further simplify the performance comparison we assume that all streams in the network are homogeneous, that is, all streams have the same leaky bucket parameters and QoS requirement. (We emphasize that this assumption is not needed in our framework; we only make it here to facilitate the comparison.) We set $\sigma_j = \sigma$, $\rho_j = \rho$, $d_j = d$ and $\epsilon_j = \epsilon$ for all streams j in the network. This implies that all connections have the same smoother rates, that is, $c_j^* = c^*$ for all streams j. Also, all of the Bernoulli random variables U_j^* are now identically distributed (but still independent). When comparing the performance we again focus on connection 0 traversing nodes 1 through N. We assume that each of the nodes $n,\ n = 1, \ldots, N$, serves J streams, that is, $|I(n)| = J\ \forall n = 1, \ldots, N$. We also assume that all output links in the network have the same capacity C. With these simplifying assumptions the worst–case loss probability of connection 0 at a node is

$$\phi_0^{*n} = \frac{E\left[(\sum_{i=1}^{J-1} U_i^* + c^* - C)^+\right]}{C} := \phi^*. \qquad (18)$$

The end–to–end loss probability of connection 0 is given by $N\phi^*$. Now assume that connection 0 is new and requests a connection traversing nodes 1 through N. The QoS requirement of the new connection 0 is satisfied if $\phi^* \leq \epsilon/N$. Suppose that all other streams that traverse one of the nodes $n,\ n = 1, \ldots, N$, have allocated a loss constraint larger than ϵ/N to that node n. With this assumption the QoS requirements of all other streams will continue to hold if $\phi^* \leq \epsilon/N$. Hence connection 0 can be admitted if $\phi^* \leq \epsilon/N$.

We use the maximum number of connections each of the multiplexers 1 through N can carry without violating any QoS commitment as a measure of the performance of our scheme. Let J^* denote this maximum number of connections. We clearly have:

$$J^* = \max_{J \in \mathbb{N}} \{J : \phi^* \leq \frac{\epsilon}{N}\},$$

where $\mathbb{N}$ denotes the set of natural numbers. Note that in the described networking scenario each of the multiplexers 1 through N is serving connection 0 and $J^* - 1$ fresh connections.

A. Comparison with Deterministic Service Disciplines

In this section we compare the performance of our smoothing/bufferless multiplexing scheme with that of deterministic service disciplines. These deterministic service disciplines provide lossless service and guarantee a deterministic end–to–end delay bound. Of the deterministic service disciplines discussed in the literature, the Generalized Processor Sharing (GPS) [34], [35] and Rate–Controlled Service (RCS) [18] disciplines guarantee the smallest delay bounds. GPS considers the route of a connection as a whole and is thus able to guarantee tighter bounds than are achievable by adding worst–case delays at each hop [7], [8]. RCS, which is at the heart of the Guaranteed Service framework of the Internet [47], relies on traffic shaping at every hop and can guarantee the same delay bounds as GPS. In fact it is shown in [18] that RCS has the potential of providing tighter delay bounds than GPS. However, the problem of how to choose the parameters of the RCS discipline in order to achieve these tighter delay bounds is not addressed. Instead, the authors suggest to use the parameters induced by the GPS discipline. This ensures that RCS can accept as many connections as GPS (and some more in a heterogeneous network). With the networking scenario that we have chosen for the performance comparison — homogeneous connections, homogeneous nodes, fluid model — GPS and RCS have exactly the same performance. We shall therefore compare our scheme's performance with that of GPS. For this purpose we modify the network depicted in Figures 1 and 2. We remove the buffered smoothers at the network ingresses and replace the bufferless multiplexers with buffered GPS servers.

Review of GPS

First, we briefly review GPS [34], [35] and adapt the notation of [34], [35] to our network model. The GPS server n serving the streams in $I(n)$ is characterized by positive real numbers $w_j^n,\ j \in I(n)$. These numbers govern the allocation of service to each of the streams. Let $S_j^n(\tau,t)$ denote the amount of stream j traffic served by server n during an interval $[\tau,t]$. The GPS policy guarantees that for any connection $j \in I(n)$ that is continuously backlogged in the interval $[\tau,t]$, that is, has a positive amount of traffic in server n's buffer throughout the interval $[\tau,t]$,

$$\frac{S_j^n(\tau,t)}{S_i^n(\tau,t)} \geq \frac{w_j^n}{w_i^n}, \quad i \in I(n).$$

A connection j that is backlogged is thus guaranteed a minimum service rate called *connection j backlog clearing rate* of

$$g_j^n = \frac{w_j^n}{\sum_{i\in I(n)} w_i^n} C_n$$

by server n. The minimum connection–0 backlog clearing rate along its route traversing nodes 1 through N is

$$g_0 = \min_{1\leq n\leq N} g_0^n.$$

Let $D_0(t)$ be the end–to–end delay incurred in the network by a connection–0 bit that arrives at time t. Furthermore, let D_0^* denote the maximum end–to–end delay of connection 0 over all time and all feasible arrival processes of all streams sharing a server with connection 0, formally:

$$D_0^* = \sup_{\cup_{1\leq n\leq N}\mathcal{A}_n} \max_{t\geq 0} D_0(t).$$

A key result of [35] is the following deterministic bound on the maximum end–to–end delay for connection 0: if $g_0 \geq \rho_0$ then

$$D_0^* \leq \frac{\sigma_0}{g_0}.$$

We note that this bound does not require the independence of the served traffic streams. The independence of the traffic streams, however, is a prerequisite for our bound on the loss probability. Given a specific delay bound d_j, finding the corresponding weights of the general GPS policy is a very tedious procedure. This procedure is greatly simplified by setting $w_j = \rho_j$ for all traffic streams. GPS with this special assignment of weights is referred to as Rate Proportional Processor Sharing (RPPS). With RPPS the connection j backlog clearing rate at server n is given by

$$g_j^n = \frac{\rho_j^n}{\sum_{i\in I(n)} \rho_i^n} C_n.$$

For ease of comparison with our smoothing/bufferless multiplexing scheme we make the same simplifying assumptions we made at the end of Section IV. In particular, we set $\rho_j = \rho$, $\sigma_j = \sigma$ and $d_j = d$ for all connections. We assume that all servers 1 through N serve J connections and have a capacity of C. With these simplifications, the minimum back log clearing rate of connection 0 along its route from node 1 to N is $g_0 = C/J$. The end–to–end delay bound of connection 0 is

$$D_0^* \leq J\sigma/C, \tag{19}$$

provided the stability condition $C/J \geq \rho$ is satisfied. We are interested in the maximum number of connections each server along the route of connection 0 can serve without violating the delay limit of connection 0 or any other connection. Let J^* denote this maximum number of connections. From (19) and the the stability condition we have:

$$J^* = \lfloor \min\left\{\frac{Cd}{\sigma}, \frac{C}{\rho}\right\} \rfloor.$$

Note that J^* does not depend on N, the number of nodes connection 0 traverses. We remark that for the example at hand, consisting of homogeneous connections with homogeneous delay bounds, J^* is the absolute maximum number of connections a deterministic service discipline can support; no matter what deterministic service discipline (GPS, RCS, etc.) is employed.

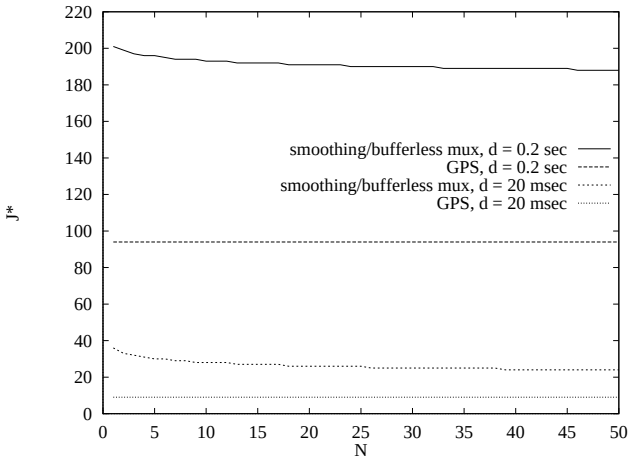


Fig. 9. Maximum number of connections J^* as a function of the number of hops N for smoothing/bufferless multiplexing and GPS.

Numerical Results

In this section we compare the performance of the smoothing/bufferless multiplexing scheme with that of GPS in multihop networks numerically. We have chosen the parameters $\sigma = 11{,}925$ Bytes, $\rho = 150$ Kbit/sec and $C = 45$ Mbit/sec. For our smoothing/bufferless multiplexing scheme we set the loss bound to $\epsilon = 10^{-7}$. (These parameters are also used for some some of the numerical examples in [15].) In Figure 9 we plot the maximum number of connections J^* that can be supported by the nodes 1 through N without violating any QoS requirements as a function of the number of hops, N. We do this for two delay bounds, $d = 20$ msec and $d = 0.2$ seconds. The maximum number of connections that can be supported by GPS is independent of N; $J^* = 9$ for $d = 20$ msec and $J^* = 94$ for $d = 0.2$ seconds.

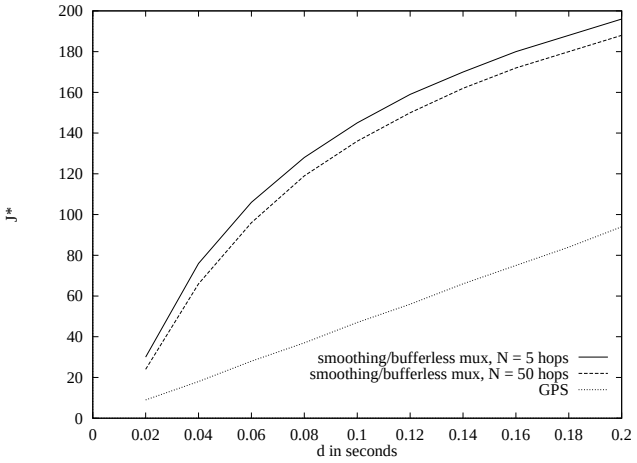


Fig. 10. Maximum number of connections J^* as a function of the delay bound d for smoothing/bufferless multiplexing and GPS. The GPS performance is independent of the number of traversed hops.

Figure 10 depicts J^* as a function of the delay bound d for $N = 5$ hops and $N = 50$ hops. Again, note that the GPS performance is independent of the number of hops.

Two points are especially noteworthy about the plots. First, with our smoothing/bufferless multiplexing scheme the number of allowable connections, J^*, drops off only slowly as the number of traversed hops, N, increases. Secondly, our smoothing/bufferless multiplexing scheme dramatically increases the connection–carrying capacity of the network. We observe from Figure 9, for instance, that for a delay bound of $d = 20$ msec and $N = 15$ hops our scheme can support more than three times the number of connections that GPS — or any other deterministic service discipline — can support. We achieve this remarkable performance by first smoothing the traffic at the network edges and then statistically multiplexing the smoothed traffic streams with miniscule loss probabilities within the network. The miniscule losses of the order of 10^{-7} can be effectively hidden by applying error concealment techniques to the multimedia streams [53]. The losses will therefore not be noticed by the viewers/listeners.

V. Interaction between Application and Network

In this section we discuss how the responsibilities of smoothing, call admission control and traffic policing can be shared by the application and the network when our smoothing/bufferless multiplexing scheme is employed. Call admission control is the responsibility of the network. Before accepting a new connection, the network has to ensure that the QoS requirements continue to hold for all established connections and the new connection. Policing is also a network responsibility. The network edge has to police all established connections in order to ensure that all connections comply with their respective regulator function advertised at connection establishment. While call admission control and traffic policing are responsibilities of the network, smoothing can be performed by either the application or the network. We refer to the case where the application performs the smoothing and sends the smoothed traffic to the network edge as *application smoothing*. The case where the application sends its unsmoothed traffic to the network edge and the network edge performs the smoothing is referred to as *network smoothing*.

With application smoothing the application internally smoothes its traffic. Based on the regulator function of its traffic and the maximum delay it can tolerate, the application finds the minimum smoother rate by applying (4). Since the smoothing is done by the application, there is no need to reduce the number of leaky buckets used to characterize the traffic by applying the heuristic outlined in Appendix B. Instead, the concave hull of a prerecorded source is used directly for dimensioning its smoother. The application advertises the regulator function $\mathcal{E}_j(t) = \min\{c_j^* t,\ \sigma_j^{L_j} + \rho_j^{L_j} t\}$ and the delay bound $d_j = 0$ to the network. We remark that this dual leaky bucket regulator function has been adopted by the ATM Forum [17] and is being proposed for the Internet [48]. The network does not have to be aware of the smoothing done by the application. The network edge dimensions its own smoother based on $\mathcal{E}_j(t)$ and $d_j = 0$. Since $d_j = 0$ the network's smoother degenerates to a server with rate c_j^* preceded by a buffer of size zero.

With network smoothing the application advertises its regulator function and maximum tolerable delay to the network. Prerecorded sources apply the heuristic of Appendix B when the network restricts the number of leaky buckets to a number smaller than the number of segments in the concave hull. The network edge dimensions the smoother based on the regulator function and delay bound supplied by the application. Call admission control is based on the assumption of worst–case on–off traffic at the smoother output. The network edge polices the applications' traffic before it enters the smoother and drops violating traffic.

VI. Final Remarks

In this article we have developed a framework for providing end–to–end statistical QoS guarantees in a network. We have argued that it is preferable to smooth the traffic at the ingress and to perform bufferless statistical multiplexing within the network than to use shared–buffer multiplexing. For our scheme we have determined the worst–case traffic and have outlined an admission control procedure based on the worst–case traffic. We have also explicitly characterized the optimal smoother.

Our results are particularly relevant in light of the current debate on service disciplines for the Internet. Our results indicate that an Internet offering exclusively Guaranteed Service based on the RCS service discipline will be severely underutilized. An Internet service allowing for small losses — such as the Predictive Service framework proposed in [5] — would be able to make efficient use of the Internet resources and still provide the receivers with an enjoyable multimedia experience. Such a statistical Internet service could be based on our smoothing/bufferless

multiplexing traffic management scheme.

ACKNOWLEDGMENT: We gratefully acknowledge interactions with Jim Roberts at the early stages of this research.

APPENDIX A: PROOF OF THEOREM 2

The purpose of this appendix is to provide a proof for Theorem 2. But first we need to establish two lemmas.

Lemma 2: A necessary condition for $(S_j(t),\ j \in I(n))$ to be feasible is $r_j^1 \geq c_j^*$ for all $j \in I(n)$.

Proof: From [8], [9], [18] the maximum delay at smoother j is

$$\tilde{d}_j = \max_{t\geq 0}\{\max_{1\leq k\leq M_j} \frac{\mathcal{E}_j(t) - s_j^k}{r_j^k} - t\}. \tag{20}$$

Suppose $r_j^1 < c_j^*$ for some $j \in I(n)$. Because $s_j^k \geq 0$ and $r_j^k \leq r_j^1$ for all k, it follows from (20) that

$$\tilde{d}_j \geq \max_{t\geq 0}\{\frac{\mathcal{E}_j(t)}{r_j^1} - t\}. \tag{21}$$

And because, by assumption, $r_j^1 < c_j^*$, it follows from (21) that

$$\tilde{d}_j > \max_{t\geq 0}\{\frac{\mathcal{E}_j(t)}{c_i^*} - t\} = d_j,$$

where the last equality follows from (4). ∎

Lemma 3: There exists a stochastic vector arrival process in $\mathcal{A}_n$ that produces the steady-state rate variables $\tilde{U}_j,\ j \in I(n)$, with $\tilde{U}_j$ having distribution

$$\tilde{U}_j = \begin{cases} \min(r_j^1, \rho_j^1) & \text{with probability } \frac{\rho_j}{\min(r_j^1,\rho_j^1)} \\ 0 & \text{with probability } 1 - \frac{\rho_j}{\min(r_j^1,\rho_j^1)} \end{cases}$$

at the smoother outputs.

Proof: For each $j \in I(n)$, let $t_j = \sigma_j^2/(\rho_j^1 - \rho_j^2)$ and $\delta_j = s_j^2/(r_j^1 - r_j^2)$. At $t = t_j$ the slope of $\mathcal{E}_j(t)$ changes form ρ_j^1 to $\rho_j^2 < \rho_j^1$. Consequently, $\mathcal{E}_j(t_j) = \rho_j^1 t_j$ is the maximum size burst that can be transmitted at rate ρ_j^1, provided successive maximum size bursts are spaced at least $\mathcal{E}_j(t_j)/\rho_j - t_j$ apart. Similarly, at $t = \delta_j$ the slope of $S_j(t)$ changes form r_j^1 to $r_j^2 < r_j^1$. Consequently, $S_j(\delta_j) = r_j^1\delta_j$ is the maximum size burst the smoother can pass at rate r_j^1, provided successive maximum size bursts are spaced at least $S_j(\delta_j)/r_j^{M_j} - \delta_j$ apart.

Let $\tilde{b}_j(t)$ be a deterministic periodic function such that

$$\tilde{b}_j(t) = \begin{cases} \rho_j^1 & 0 \leq t < t_{\text{on}_j} \\ 0 & t_{\text{on}_j} \leq t \leq T_j\ . \end{cases}$$

with on–time t_{on_j} and period T_j given in Table II. Also, let $\theta_j,\ j \in I(n)$, be independent random variables with θ_j uniformly distributed over $[0, T_j]$ and define the stochastic arrival process j as

$$\tilde{A}_j(t) = \int_0^t \tilde{b}_j(s + \theta_j)ds.$$

Thus each component arrival process $(\tilde{A}_j(t),\ t \geq 0)$ is generated by a periodic on–off source; process j has peak rate ρ_j^1 and average rate ρ_j. The argument in the proof of Theorem 1 shows that the vector process $(\tilde{\mathbf{A}}(t),\ t \geq 0)$ is a feasible process in $\mathcal{A}_n$.

It remains to show that by sending each component process $(\tilde{A}_j(t),\ t \geq 0)$ into its respective smoother we obtain an on–off process whose peak rate is $\min(r_j^1, \rho_j^1)$ and whose average rate is ρ_j. Specifically, we now show that $\tilde{A}_j(t)$ produces $\tilde{O}_j(t) = \int_0^t \tilde{o}_j(s + \theta_j)ds$ at the smoother output where

$$\tilde{o}_j(t) = \begin{cases} \min(r_j^1, \rho_j^1) & 0 \leq t < \tau_{\text{on}_j} \\ 0 & \tau_{\text{on}_j} \leq t \leq T_j\ , \end{cases}$$

where the periods and on–times are given in Table II.

First, consider the case $\rho_j^1 \geq r_j^1$ and $\mathcal{E}_j(t_j) \geq S_j(\delta_j)$. Clearly, $t_{\text{on}_j} \leq t_j$ since $t_{\text{on}_j} = S_j(\delta_j)/\rho_j^1$ and $t_j = \mathcal{E}_j(t_j)/\rho_j^1$ and by assumption $S_j(\delta_j) \leq \mathcal{E}_j(t_j)$. This implies that $\mathcal{E}_j(t_{\text{on}_j}) = \rho_j^1 t_{\text{on}_j}$. Hence

$$S_j(\tau_{\text{on}_j}) = \mathcal{E}_j(t_{\text{on}_j}). \tag{22}$$

Note furthermore that

$$t_{\text{on}_j} \leq \tau_{\text{on}_j} \tag{23}$$

since $t_{\text{on}_j} = S_j(\delta_j)/\rho_j^1 = r_j^1\delta_j/\rho_j^1$ and by assumption $r_j^1 \leq \rho_j^1$. Because of (22) and (23) and $\tau_{\text{on}_j} = \delta_j$ the smoother bursts at rate r_j^1 for a duration of τ_{on_j} when fed with an input burst at rate ρ_j^1 for a duration of $t_{\text{on}_j} \leq t_j$. Also, note that the smoother output has average rate $\mathcal{E}_j(t_{\text{on}_j})/T_j = \rho_j \leq r_j^{M_j}$, where the last inequality follows from the stability condition.

Next, consider the case $\rho_j^1 \geq r_j^1$ and $\mathcal{E}_j(t_j) < S_j(\delta_j)$. We have

$$\tau_{\text{on}_j} \leq \delta_j \tag{24}$$

since $\tau_{\text{on}_j} = \mathcal{E}_j(t_j)/r_j^1$ and $\delta_j = S_j(\delta_j)/r_j^1$ and by assumption $S_j(\delta_j) > \mathcal{E}_j(t_j)$. Thus $S_j(\tau_{\text{on}_j}) = r_j^1\tau_{\text{on}_j}$. Hence

$$S_j(\tau_{\text{on}_j}) = \mathcal{E}_j(t_{\text{on}_j}). \tag{25}$$

Also,

$$t_{\text{on}_j} \leq \tau_{\text{on}_j} \tag{26}$$

since $t_{\text{on}_j} = \mathcal{E}_j(t_j)/\rho_j^1$ and $\tau_{\text{on}_j} = \mathcal{E}_j(t_j)/r_j^1$ and by assumption $\rho_j^1 > r_j^1$. Because of (24), (25) and (26) the smoother bursts at rate r_j^1 for a duration of τ_{on_j} when fed with an input burst at rate ρ_j^1 for a duration of t_{on_j}. The average rate of the smoother output is $\mathcal{E}_j(t_{\text{on}_j})/T_j = \rho_j \leq r_j^{M_j}$, where the last inequality follows from the stability condition.

Now consider the case $\rho_j^1 < r_j^1$ and $\mathcal{E}_j(t_j) \geq S_j(\delta_j)$. We have $t_{\text{on}_j} \leq t_j$ since $t_{\text{on}_j} = S_j(\delta_j)/\rho_j^1$ and $t_j = \mathcal{E}_j(t_j)/\rho_j^1$ and by assumption $S_j(\delta_j) \leq \mathcal{E}_j(t_j)$. This implies that $\mathcal{E}_j(t_{\text{on}_j}) = \rho_j^1 t_{\text{on}_j}$. Hence

$$S_j(\delta_j) = \mathcal{E}_j(t_{\text{on}_j}). \tag{27}$$

TABLE II
ON–TIMES AND PERIODS OF $\tilde{b}_j(t)$ AND $\tilde{o}_j(t)$.

	$\rho_j^1 \geq r_j^1$		$\rho_j^1 < r_j^1$	
	$\mathcal{E}_j(t_j) \geq S_j(\delta_j)$	$\mathcal{E}_j(t_j) < S_j(\delta_j)$	$\mathcal{E}_j(t_j) \geq S_j(\delta_j)$	$\mathcal{E}_j(t_j) < S_j(\delta_j)$
T_j	$S_j(\delta_j)/\rho_j$	$\mathcal{E}_j(t_j)/\rho_j$	$S_j(\delta_j)/\rho_j$	$\mathcal{E}_j(t_j)/\rho_j$
t_{on_j}	$S_j(\delta_j)/\rho_j^1$	t_j	$S_j(\delta_j)/\rho_j^1$	t_j
τ_{on_j}	δ_j	$\mathcal{E}_j(t_j)/r_j^1$	$S_j(\delta_j)/\rho_j^1$	t_j

Note furthermore that

$$\delta_j \leq t_{\text{on}_j} \tag{28}$$

since $\delta_j = S_j(\delta_j)/r_j^1$ and $t_{\text{on}_j} = S_j(\delta_j)/\rho_j^1$ and by assumption $r_j^1 > \rho_j^1$. Because of (27), (28) and $\rho_j^1 < r_j^1$ (by assumption) the smoother passes the input burst at rate ρ_j^1 for a duration of t_{on_j} unchanged. The average rate of the smoother output is $\mathcal{E}_j(t_{\text{on}_j})/T_j = \rho_j \leq r_j^{M_j}$, where the last inequality follows from the stability condition.

Finally, consider the case $\rho_j^1 < r_j^1$ and $\mathcal{E}_j(t_j) < S_j(\delta_j)$. These two assumptions imply that the smoother can pass the input burst of size $\mathcal{E}_j(t_j)$ at rate ρ_j^1. The average rate of the smoother output is $\mathcal{E}_j(t_{\text{on}_j})/T_j = \rho_j \leq r_j^{M_j}$, where the last inequality follows from the stability condition. ■

Proof of Theorem 2: Using Lemma 3 and mimicking the proof of Theorem 1 we obtain

$$\phi_j^n = \frac{E\left[(\sum_{i\in I(n)} \tilde{U}_i - C_n)^+ \tilde{U}_j\right]}{C_n \cdot E[\tilde{U}_j]},$$

where $\tilde{U}_j$, $j \in I(n)$, are defined in Lemma 3. Using the fact that $\tilde{U}_j$ is a Bernoulli random variable, we obtain from the above expression

$$\begin{aligned} \phi_j &= \frac{E\left[(\sum_{i\in I(n)-\{j\}} \tilde{U}_i + \min(r_j^1, \rho_j^1) - C_n)^+\right]}{C_n} \\ &\geq \frac{E\left[(\sum_{i\in I(n)-\{j\}} \tilde{U}_i + c_j^* - C_n)^+\right]}{C_n}, \end{aligned} \tag{29}$$

where the last inequality follows from Lemma 2.

From (13) and (29) it remains to show that

$$\begin{aligned} E[(\sum_{i\in I(n)-\{j\}} U_i^* + c_j^* - C_n)^+] \leq \\ E[(\sum_{i\in I(n)-\{j\}} \tilde{U}_i + c_j^* - C_n)^+]. \end{aligned} \tag{30}$$

From Lemma 2 and Proposition 1.5.1 in [52]

$$U_i^* \leq_{icx} \tilde{U}_i \quad \text{for all } i \in I(n). \tag{31}$$

The inequality (30) follows from (31), the independence of U_j^*, $j \in I(n)$, and an argument that parallels the argument in the proof of Theorem 1. □

APPENDIX B: A HEURISTIC FOR FINDING A LEAKY BUCKET CHARACTERIZATION OF PRERECORDED SOURCES

In this appendix we discuss how to obtain a good characterization $\mathcal{E}_j(t)$ of a source for a given restriction L_j on the number of leaky buckets. For any given characterization $\mathcal{E}_j(t)$ we use at the network edge a single–buffer smoother with rate c_j^* given by (4). Our goal is to find a characterization $\mathcal{E}_j(t)$ that has at most L_j slopes (i.e., L_j cascaded leaky buckets) and attempts to minimize both ρ_j and c_j^*. From Theorem 2 we know that minimizing ρ_j and c_j^* minimizes the worst–case loss probabilities, and thereby maximizes the connection–carrying capacity of a particular node.

We develop the heuristic for determining the characterization $\mathcal{E}_j(t)$ in the context of prerecorded sources. These sources include full–length movies, music video clips and educational material for video–on–demand (VoD) and other multimedia applications. It is well known how to compute the empirical envelope for prerecorded sources [32], [54], [31]. The empirical envelope gives the tightest bound on the amount of traffic that can emanate from a prerecorded source over any time interval. The empirical envelope is however not necessarily concave, and therefore we may not be able to characterize it by a cascade of leaky buckets. However, applying the algorithms of Wrege *et al.* [54] or Grahams Scan [6], we can compute the concave hull of the empirical envelope. The concave hull for connection–j traffic, denoted by $\mathcal{H}_j(t)$, takes the form

$$\mathcal{H}_j(t) = \min_{1\leq i\leq K_j} \{\sigma_j^i + \rho_j^i t\}. \tag{32}$$

Here, K_j denotes the number of piecewise linear segments in the concave hull. Without loss of generality we may assume $\sigma_j^1 < \sigma_j^2 < \cdots < \sigma_j^{K_j}$ and $\rho_j^1 > \rho_j^2 > \cdots > \rho_j^{K_j}$.

The number of segments in the concave hull can be rather large. The *"Silence of The Lambs"* video segment used in our numerical experiments, for instance, has a concave hull consisting of 39 segments. This implies that 39 leaky bucket pairs are required to police the tightest concave characterization of the *"Silence of The Lambs"* video segment. Our goal is to find a more succinct characterization of prerecorded sources in order to simplify call admission control and traffic policing.

Suppose that a source is allowed to use L_j ($L_j < K_j$) leaky buckets to characterize its traffic. We now present a heuristic for the following problem: Given a source's concave hull $\mathcal{H}_j(t) = \min_{1\leq i\leq K_j}\{\sigma_j^i + \rho_j^i t\}$ and the delay limit

d_j, find L_j leaky buckets (out of the K_j leaky bucket pairs in the concave hull) that maximize the admission region.

We illustrate our heuristic for the case $L_j = 2$. For $L_j = 2$ the traffic constraint function takes the form

$$\mathcal{E}_j(t) = \min\{\sigma_j^{a_j} + \rho_j^{a_j} t,\ \sigma_j^{b_j} + \rho_j^{b_j} t\} \quad \text{with} \quad 1 \le a_j, b_j \le K_j, \tag{33}$$

where the indices a_j and b_j are yet to be specified. Our strategy is to first choose the leaky bucket that has the tightest bound on the average rate (i.e., minimize ρ_j), and then choose another leaky bucket which minimizes the smoother rate c_j^*. Let r_j^{ave} denote the average rate of the prerecorded source. We found in our numerical experiments that some of the leaky bucket pairs in the concave hull (particularly those with high indices) may have slopes less than r_j^{ave}. We set $b_j = \max\{i : \rho_j^i \ge r_j^{\text{ave}},\ 1 \le i \le K_j\}$. In words, we use the highest indexed leaky bucket with a slope larger than r_j^{ave} to specify the source's average rate.

In order to find the leaky bucket indexed by a_j we consider all leaky buckets $(\sigma_j^i,\ \rho_j^i)$ with $1 \le i < b_j$. We compute the smoother rates obtained by combining each of the leaky buckets $(\sigma_j^i,\ \rho_j^i)$, $1 \le i < b_j$ with the leaky bucket $(\sigma_j^{b_j}, \rho_j^{b_j})$ and select the index i that gives the smallest smoother rate — and thus the largest admission region. More formally, let c_j^{*i}, $1 \le i < b_j$, denote the minimal smoother rate for traffic with regulator function $\mathcal{E}_j(t) = \min\{\sigma_j^i + \rho_j^i t,\ \sigma_j^{b_j} + \rho_j^{b_j} t\}$ and delay bound d_j. By (4) we have

$$c_j^{*i} \quad = \quad \max_{t \ge 0} \frac{\min\{\sigma_j^i + \rho_j^i t,\ \sigma_j^{b_j} + \rho_j^{b_j} t\}}{d_j + t}.$$

We can obtain a more explicit expression for c_j^{*i}. Since

$$\min\{\sigma_j^i + \rho_j^i t,\ \sigma_j^{b_j} + \rho_j^{b_j} t\} = \begin{cases} \sigma_j^i + \rho_j^i t & \text{for } 0 \le t \le t_i \\ \sigma_j^{b_j} + \rho_j^{b_j} t & \text{for } t \ge t_i \end{cases}$$

with $t_i = (\sigma_j^{b_j} - \sigma_j^i)/(\rho_j^i - \rho_j^{b_j})$, we have

$$c_j^{*i} \quad = \quad \max\left[\max_{0 \le t \le t_i} \frac{\sigma_j^i + \rho_j^i t}{d_j + t},\ \max_{t \ge t_i} \frac{\sigma_j^{b_j} + \rho_j^{b_j} t}{d_j + t}\right].$$

The expressions inside the max[·] can be further simplified. It can be shown that

$$\max_{0 \le t \le t_i} \frac{\sigma_j^i + \rho_j^i t}{d_j + t} = \begin{cases} \frac{\sigma_j^i}{d_j} & \text{if } d_j \le \frac{\sigma_j^i}{\rho_j^i} \\ \frac{\sigma_j^i + \rho_j^i t_i}{d_j + t_i}, & \text{if } d_j \ge \frac{\sigma_j^i}{\rho_j^i} \end{cases}$$

and

$$\max_{t \ge t_i} \frac{\sigma_j^{b_j} + \rho_j^{b_j} t}{d_j + t} = \begin{cases} \frac{\sigma_j^i + \rho_j^i t_i}{d_j + t_i}, & \text{if } d_j \le \frac{\sigma_j^{b_j}}{\rho_j^{b_j}} \\ \frac{\sigma_j^{b_j}}{d_j} & \text{if } d_j \ge \frac{\sigma_j^{b_j}}{\rho_j^{b_j}}. \end{cases}$$

We set the smoother rate to $\min_{1 \le i < b_j} c_j^{*i}$ and set a_j to the index that attains this minimum.

We now briefly discuss how to find the optimal regulator function consisting of 3 or more leaky buckets. First, note that there are $\begin{pmatrix} b_j - 1 \\ L_j - 1 \end{pmatrix}$ combinations of leaky bucket pairs to consider. This can be computationally prohibitive. The heuristic can be sped up by considering only regulator functions consisting of $L_j - 1$ consecutive leaky buckets of the concave hull and the leaky bucket $(\sigma_j^{b_j},\ \rho_j^{b_j})$. In the case $L_j = 3$, for instance, we compute the minimal smoother rates only for the regulator functions $\mathcal{E}_j(t) = \min\{\sigma_j^i + \rho_j^i t,\ \sigma_j^{i+1} + \rho_j^{i+1} t,\ \sigma_j^{b_j} + \rho_j^{b_j} t\}$ with $1 \le i < b_j - 1$. This speed–up of the heuristic can produce a suboptimal regulator function. Our numerical experiments (see Section III-B), however, indicate that it works surprisingly well.

We evaluate the heuristic using the traces of Section III-B. The heuristic produced the optimal leaky bucket characterizations given in Table III for the lambs trace. The

TABLE III

PARAMETERS OF THE OPTIMAL LEAKY BUCKET CHARACTERIZATION WITH 2 LEAKY BUCKETS AS A FUNCTION OF THE DELAY BOUND FOR THE LAMBS TRACE. THE AVERAGE RATE IS CHARACTERIZED BY THE 34TH LEAKY BUCKET, I.E., $b_{\text{lambs}} = 34$, WITH PARAMETERS $\sigma_{\text{lambs}}^{b_{\text{lambs}}} = 3{,}157.8$ KBYTE AND $\rho_{\text{lambs}}^{b_{\text{lambs}}} = 208.8$ KBIT/SEC FOR ALL DELAY BOUNDS.

d_{lambs} sec.	a_{lambs}	$\sigma_{\text{lambs}}^{a_{\text{lambs}}}$ kByte	$\rho_{\text{lambs}}^{a_{\text{lambs}}}$ kbit/sec	c_{lambs}^* kbit/sec
0	1	0	3474.8	3474.8
0.042	2	13.3	939.3	2535.5
0.125	2	13.3	939.3	939.0
0.250	4	23.5	802.2	801.9
0.500	8	43.5	711.0	710.8
1.000	10	69.9	676.9	674.7

table gives the index a_{lambs} and the parameters of the leaky bucket $(\sigma_{\text{lambs}}^{a_{\text{lambs}}},\ \rho_{\text{lambs}}^{a_{\text{lambs}}})$ for various delay bounds. The average rate is characterized by the 34th leaky bucket in the concave hull, i.e., $b_{\text{lambs}} = 34$, for all delay bounds. The table also gives the minimal smoother rates for the various delay bounds. For a delay bound of zero, the smoother rate is set to the rate of the first leaky bucket, i.e., the peak rate of the trace. For $d_{\text{lambs}} = 0.042$ sec $(= 1/F)$ the trace is characterized by the 2nd and 34th leaky bucket of the concave hull $(a_{\text{lambs}} = 2,\ b_{\text{lambs}} = 34)$. Note that $d_{\text{lambs}} < \sigma_{\text{lambs}}^{a_{\text{lambs}}}/\rho_{\text{lambs}}^{a_{\text{lambs}}}$ in this case and $c_{\text{lambs}}^* = \sigma_{\text{lambs}}^{a_{\text{lambs}}}/d_{\text{lambs}}$. For $d_{\text{lambs}} \ge 0.125$ sec we have $d_{\text{lambs}} > \sigma_{\text{lambs}}^{a_{\text{lambs}}}/\rho_{\text{lambs}}^{a_{\text{lambs}}}$ and $c_{\text{lambs}}^* = (\sigma_j^{a_{\text{lambs}}} + \rho_j^{a_{\text{lambs}}} t_{a_{\text{lambs}}})/(d_j + t_{a_{\text{lambs}}})$.

REFERENCES

[1] M. Andrews. Probabilistic end–to–end delay bounds for earliest deadline first scheduling. In *Proceedings of IEEE Infocom 2000*, Tel Aviv, Israel, March 2000.

[2] R. Boorstyn, A. Burchard, J. Liebeherr, and C. Oottamakorn. Statistical service assurances for traffic scheduling algorithms. *IEEE Journal on Selected Areas in Communications*, 18(13):2651–2664, December 2000.

[3] C.-S. Chang. Stability, queue length and delay of deterministic and stochastic networks. *IEEE Transactions on Automatic Control*, 39(5):913–931, May 1994.

[4] C.-S. Chang. *Performance Guarantees in Communication Networks*. Springer Verlag, 2000.

[5] D. Clark, S. Shenker, and L. Zhang. Supporting real–time applications in an integrated services packet network: Architecture and mechanism. In *ACM SIGCOMM '92*, pages 14–26, 1992.

[6] T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 1990.

[7] R. Cruz. A calculus for network delay, part I: Network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–131, January 1991.

[8] R. Cruz. A calculus for network delay, part II: Network analysis. *IEEE Transactions on Information Theory*, 37(1):132–141, January 1991.

[9] R. Cruz. Quality of service guarantees in virtual circuit switched networks. *IEEE Journal on Selected Areas in Communications*, 13(6):1048–1056, August 1995.

[10] R. L. Cruz. Quality of service management in integrated services networks. In *Proceedings of 1st Semi–Annual Research Review*, CWC, UCSD, June 1996.

[11] B. T. Doshi. Deterministic rule based traffic descriptors for broadband ISDN: Worst case behavior and connection acceptance control. In J. Labetoulle and J. W. Roberts, editors, *Proceedings of International Teletraffic Congress (ITC) 14*, pages 591–600. Elsevier Science B. V., 1994.

[12] B. T. Doshi. Deterministic rule based traffic descriptors for broadband ISDN: Worst case behavior and its impact on connection acceptance control. *International Journal of Communication Systems*, 8:91–109, 1995.

[13] A. Elwalid, D. Heyman, T. Lakshman, D. Mitra, and A. Weiss. Fundamental bounds and approximations for ATM multiplexers with application to video teleconferencing. *IEEE Journal on Selected Areas in Communications*, 13(6):1004–1016, August 1995.

[14] A. Elwalid and D. Mitra. Design of generalized processor sharing schedulers which statistically multiplex heterogeneous QoS classes. In *Proceedings of IEEE Infocom '99*, pages 1220–1230, New York, NY, March 1999.

[15] A. Elwalid, D. Mitra, and R. H. Wentworth. A new approach for allocating buffers and bandwidth to heterogeneous regulated traffic in an ATM node. *IEEE Journal on Selected Areas in Communications*, 13(6):1115–1127, August 1995.

[16] D. Y. Eun and N. B. Shroff. The impact of aggregation in simplifying network analysis. In *submitted*, July 2001.

[17] ATM Forum. *ATM User–Network Interface Specification, Version 3.0*. Prentice–Hall, 1993.

[18] L. Georgiadis, R. Guerin, V. Peris, and K.N. Sivarajan. Efficient network QoS provisioning based on per node traffic shaping. *IEEE/ACM Transactions on Networking*, 4(4):482–501, August 1996.

[19] I. Hsu and J. Walrand. Admission control for ATM networks. In *IMA Workshop on Stochastic Networks*, Minneapolis, Minnesota, March 1994.

[20] E. W. Knightly. H-BIND: A new approach to providing statistical performance guarantees to VBR traffic. In *Proceedings of IEEE Infocom '96*, San Francisco, CA, April 1996.

[21] E. W. Knightly and H. Zhang. Providing end–to–end statistical performance guarantees with bounding interval dependent stochastic models. In *Proceedings of ACM Sigmetrics '94*, 1994.

[22] K. Kumaran, G. E. Margrave, D. Mitra, and K. R. Stanley. Novel techniques for the design and control of generalized processor sharing schedulers for multiple QoS classes. In *Proceedings of IEEE Infocom 2000*, Tel Aviv, Israel, March 2000.

[23] J. Kurose. On computing per–session performance bounds in high–speed multi–hop computer networks. In *Proceedings of ACM Sigmetrics '92*, pages 128–139, 1992.

[24] S.-K. Kweon and K. G. Shin. Traffic–controlled rate–monotonic priority scheduling. In *Proceedings of IEEE Infocom 1996*, pages 655–662, San Francisco, CA, March 1996.

[25] S.-K. Kweon and K. G. Shin. Real–time transport of MPEG video with a statistically guaranteed loss ratio in ATM networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(4):387–403, April 2001.

[26] J. Y. LeBoudec and P. Thiran. *Network Calculus, Lecture Notes in Computer Science, Vol. 2050*. Springer Verlag, 2001.

[27] J.Y. LeBoudec. Applications of network calculus to guaranteed service networks. *IEEE Transactions on Information Theory*, 44(3):1087–1096, May 1998.

[28] T. Lee, K. Lai, and S. Duann. Design of a real–time call admission controller for ATM. *IEEE/ACM Transactions on Networking*, 4(5):758–765, October 1995.

[29] C. Li and E.W. Knightly. Coordinated network scheduling: A framework for end–to–end services. In *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, Osaka, Japan, November 2000.

[30] J. Liebeherr, S. Patek, and A. Burchard. A calculus for end–to–end statistical service guarantees. Technical Report CS–2001–19, University of Virginia, August 2001.

[31] J. Liebeherr and D. Wrege. An efficient solution to traffic characterization of VBR video in quaility–of–service networks. *ACM Multimedia Systems Journal*, 6(4):271–284, July 1998.

[32] J. Liebeherr, D. Wrege, and D. Ferrari. Exact admission control for networks with a bounded delay service. *IEEE/ACM Transactions on Networking*, 4(6):885–901, December 1996.

[33] F. LoPresti, Z. Zhang, D. Towsley, and J. Kurose. Source time scale and optimal buffer/bandwidth trade–off for regulated traffic in a network node. *IEEE/ACM Transactions on Networking*, 7(4):490–501, August 1999. A shorter version has appeared in *Proc. of IEEE Infocom*, Kobe, Japan, April 1997.

[34] A. Parekh and R. Gallager. A generalized processor sharing approach to flow control in integrated services networks: The single node case. *IEEE/ACM Transactions on Networking*, 1(3):344–357, June 1993.

[35] A. Parekh and R. Gallager. A generalized processor sharing approach to flow control in integrated services networks: The multiple node case. *IEEE/ACM Transactions on Networking*, 2(2):137–150, April 1994.

[36] J. Qiu and E. Knightly. Inter–class resource sharing using statistical service envelopes. In *Proceedings of IEEE Infocom '99*, New York, NY, March 1999.

[37] S. Rajagopal, M. Reisslein, and K. W. Ross. Packet multiplexers with adversarial regulated traffic. *Computer Communications*, 25(3):239–253, February 2001. a shorter version appeared in *Proceedings of IEEE Infocom '98*, pages 347–355, San Francisco, CA, April 1998.

[38] R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne. Adaptive playout mechanisms for packetized audio applications in wide-area networks. In *Proceedings of IEEE Infocom '94*, pages 680–688, San Francisco, CA, April 1994.

[39] M. Reisslein. Measurement–based admission control for bufferless multiplexers. *International Journal of Communication Systems (Wiley Interscience)*, 14, 2001. appeared in part in *Proceedings of the IEEE International Symposium on Computers and Communications (ISCC)*, pages 462–467, Antibes, France, July 2000.

[40] M. Reisslein and K. W. Ross. Call admission for prerecorded sources with packet loss. *IEEE Journal on Selected Areas in Communications*, 15(6):1167–1180, August 1997.

[41] M. Reisslein, K. W. Ross, and S. Rajagopal. Guaranteeing statistical QoS to regulated traffic: The multiple node case. In *Proceedings of 37th IEEE Conference on Decision and Control (CDC)*, pages 531–538, Tampa, FL, December 1998.

[42] M. Reisslein, K. W. Ross, and S. Rajagopal. Guaranteeing statistical QoS to regulated traffic: The single node case. In *Proceedings of IEEE Infocom '99*, pages 1060–1071, New York, NY, March 1999.

[43] J. Roberts, U. Mocci, and J. Virtamo (Eds.). *Broadband Network Traffic: Performance Evaluation and Design of Broadband Multiservice Networks, Final Report of Action COST 242, (Lecture Notes in Computer Science, Vol. 1155)*. Springer Verlag, 1996.

[44] J. W. Roberts. Realizing quality of service guarantees in multiservice networks. In T. Hasegawa, H. Takagi, and Y. Takahashi, editors, *Performance and Management of Complex Communication Networks — Proceedings of IFIP Conference PMCCN '97*, Tsukuba, Japan, November 1997. Chapmann Hall.

[45] O. Rose. Statistical properties of MPEG video traffic and their impact on traffic modelling in ATM systems. Technical Report 101, University of Wuerzburg, Germany, February 1995.

[46] K. W. Ross. *Multiservice Loss Models for Broadband Telecommunication Networks*. Springer–Verlag, 1995.

[47] S. Shenker, C. Partridge, and R. Guerin. Request for comments 2212: Specification of guaranteed quality of service. September 1997.

[48] S. Shenker and J. Wroclawski. Request for comments 2215: Gen-

eral characterization parameters for integrated service network elements. September 1997.

[49] V. Sivaraman and F. M. Chiussi. Providing end–to–end statistical delay guarantees with earliest deadline first scheduling and per–hop traffic shaping. In *Proceedings of IEEE Infocom 2000*, Tel Aviv, Israel, March 2000.

[50] V. Sivaraman, F. M. Chiussi, and M. Gerla. End–to–end statistical delay service under GPS and EDF scheduling: A comparison study. In *Proceedings of IEEE Infocom 2001*, Anchorage, Alaska, April 2001.

[51] D. Starobinski and M. Sidi. Stochastically bounded burstiness for communication networks. In *Proceedings of IEEE Infocom '99*, New York, NY, March 1999.

[52] D. Stoyan. *Comparison methods for queues and other stochastic models*. Wiley, 1983.

[53] Y. Wang and Q. Zhu. Error control and concealment for video communication: A review. *Proceedings of the IEEE*, 86(5):974–997, May 1998.

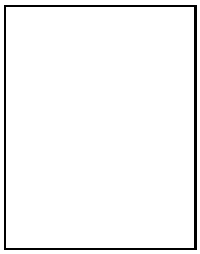
[54] D. Wrege, E. Knightly, H. Zhang, and J. Liebeherr. Deterministic delay bounds for VBR video in packet–switching networks: Fundamental limits and tradeoffs. *IEEE/ACM Transactions on Networking*, 4(3):352–362, June 1996.

[55] T. Wu and E. W. Knightly. Enforcable and efficient service provisioning. *Computer Communications*, 23(14–15):1377–1388, August 2000.

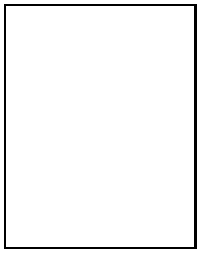
[56] T. Wu and E.W. Knightly. Buffering vs. smoothing for end–to–end QoS: Fundamental issues and comparison. In *Proceedings of Performance '99*, Istanbul, Turkey, August 1999.

[57] H. Zhang. Providing end–to–end performance guarantees using non–workconserving disciplines. *Computer Communications*, 18(10), October 1995.

[58] H. Zhang and D. Ferrari. Rate–controlled service disciplines. *Journal of High Speed Networks*, 3(4):389–412, 1994.



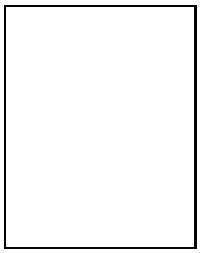
Martin Reisslein (M '95/ACM '98) is an Assistant Professor in the Department of Electrical Engineering at Arizona State University, Tempe. He is affiliated with ASU's Telecommunications Research Center. He received the Dipl.-Ing. (FH) degree from the Fachhochschule Dieburg, Germany, in 1994, and the M.S.E. degree from the University of Pennsylvania, Philadelphia, in 1996. Both in electrical engineering. He received his Ph.D. in systems engineering from the University of Pennsylvania in 1998. During the academic year 1994–1995 he visited the University of Pennsylvania as a Fulbright scholar. From July 1998 through October 2000 he was a scientist with the German National Research Center for Information Technology (GMD FOKUS), Berlin. While in Berlin he was teaching courses on performance evaluation and computer networking at the Technical University Berlin. He has served on the Technical Program Committees of IEEE Infocom, IEEE Globecom, and the IEEE International Symposium on Computer and Communications. He has organized sessions at the IEEE Computer Communications Workshop (CCW). He maintains an extensive library of video traces for network performance evaluation, including frame size traces of MPEG–4 and H.263 encoded video, at `http://www.eas.asu.edu/trace`. He is co–recipient of the Best Paper Award of the *SPIE Photonics East 2000 — Terabit Optical Networking* conference. His research interests are in the areas of Internet Quality of Service, video traffic characterization, wireless networking, and optical networking. He is particularly interested in traffic management for multimedia services with statistical Quality of Service in the Internet and wireless communication systems.



Keith Ross received his Ph.D. from the University of Michigan in 1985 (Program in Computer, Information and Control Engineering). He was a professor at the University of Pennsylvania from 1985 through 1997. At the University of Pennsylvania, his primary appointment was in the Department of Systems Engineering and his secondary appointment was in the Wharton School. He joined the Multimedia Communications Dept. at Institut Eurecom in January 1998, and became department chairman in October 1998. In Fall 1999, while remaining a professor at Institut Eurecom, he co–founded and became CEO of Wimba.

Keith Ross has published over 60 papers and written two books. He has served on editorial boards of five major journals, and has served on the program committees of major networking conferences, including Infocom and Sigcomm. He has supervised more than ten Ph.D. theses. His research and teaching interests include multimedia networking, asynchronous teaching, Web caching, streaming audio and video, and traffic modeling.

Along with Jim Kurose, Keith Ross is the co–author of *Computer Networking: A Top–Down Approach Featuring the Internet*, Addison–Wesley, 2000.



Srinivas Rajagopal received his Ph.D. in Operations Research from the University of North Carolina in 1995. He was an Assistant Professor at the University of Pennsylvania, Systems Engineering until 1997. His areas of research included Stochastic Processes and Telecommunications. Since September 1997 he has been with Manugistics Group, Inc. As Technical Director his present interests are in the area of Logistics and Supply Chain Management.