In Proceedings of IFIP IWQoS'96, Paris, France

# Improving QoS through Traffic Smoothing

Edward W. Knightly
knightly@tenet.berkeley.edu
EECS Department, U.C. Berkeley

Paola Rossaro
rossaro@wrs.com
Wind River Systems, Alameda, CA

**Abstract**

*The burstiness of variable bit rate traffic makes it difficult to simultaneously make efficient use of network resources and provide end-to-end performance guarantees. In this paper, we present a smoothing scheme that allows network clients to optimize their quality- or price-of-service to respectively reduce their end-to-end delay bound or decrease their resource requirements from the network. Whether or not smoothing results in a net benefit to the network client depends on the the network load, the number of hops traversed, the burstiness of the stream, and the desired end-to-end delay bound. Hence, our scheme tells network clients* if *they should smooth, and if so,* how much *they should smooth to achieve the maximum benefit. We first derive the analytical foundations for the scheme, and then perform empirical investigations using traces of MPEG compressed video.*

## 1 Introduction

Future packet-switching integrated services networks must support applications with diverse traffic characteristics and performance requirements. Many of these applications are delay- and loss-sensitive and have stringent requirements in terms of Quality of Service (QoS). A resource reservation scheme together with priority service disciplines inside the network (as in [5]) provide the means for giving network clients end-to-end guarantees on delay, loss, and throughput. A *deterministic* guaranteed service ensures that all packets of a connection meet the promised QoS. This provides a no-loss, no-delay-violation service and a better QoS than the probabilistic performance bounds provided by a *statistical* guaranteed service. Contrary to conventional wisdom, a deterministic service does not require a peak-rate-allocation scheme: through better traffic models such as the Deterministic Bounding Interval Dependent (D-BIND) traffic model [10], and with more accurate admission control conditions as in [2, 3, 14], significant utilization improvements are possible.

For bursty VBR traffic sources, it is generally difficult to provide the good QoS that the network clients desire, and to simultaneously achieve high network utilization. While reducing the burstiness of VBR sources through traffic *smoothing* may increase the network utilization, it also introduces a reduction in the stream's end-to-end quality of service. Specifically, smoothing can be viewed along three dimensions: bandwidth, loss, and delay. First, a stream can be smoothed by reducing its bandwidth, i.e., by sending less information over time. For VBR video, this could be achieved by reducing the perceptual quality of the video [7]. Second, introducing loss, or dropping packets during a stream's bursts, may also produce a smoother traffic stream. However, dropping packets during bursts can have an especially disastrous effect on QoS for VBR video streams such as MPEG, since intra-coded frames are both the largest and the most important, due to inter-frame dependencies [12]. Finally, a source can be smoothed by adding variable delays to packets, i.e., by spreading out bursts over time.

In this paper, we utilize this third dimension of smoothing and propose a smoothing scheme that allows network clients to improve their quality- or price-of-service. We introduce bounded-delay traffic shapers which buffer, but do not drop, incoming packets. We then combine this smoothing scheme with an end-to-end deterministic service as in [5]. Unlike the aforementioned bandwidth- and loss-smoothing schemes, our scheme provides a constant-perceptual and no-loss quality of service, with bounded end-to-end delay.

We first investigate the analytical foundations for smoothing in the context of deterministic performance guarantees and provide a partial ordering of "smoothness" or burstiness of streams. Unlike previous burstiness measures (e.g., [6, 11, 13]), this ordering directly relates to a stream's maximum resource requirements inside the network. We then investigate two traffic smoothing mechanisms: a new D-BIND traffic shaper that can be used to obtain specific traffic shapes, and a simple FIFO (first-in first-out) traffic shaper. We bound the stream's delay inside the traffic shapers and ensure that no packets are dropped by the shaper. We then bound the total end-to-end delay, including smoothing delay and queueing delay at each hop.

Next, we provide a scheme that allows network clients to determine which smoothing rate, if any provides them with the maximum benefit. If smoothing is beneficial, the network client can realize the benefits via either an improved *quality* of service or an improved *price* of service, where the latter benefit assumes that network resources are priced according to the source's burstiness, and hence, resource usage. Via analysis, we show that if homogeneous streams traverse only a single hop, or if they traverse multiple hops with only one *congested* hop, then smoothing is an *ineffective* means for improving their end-to-end QoS. The reason for this is that any savings in queueing delay achieved by transmitting smoother traffic is outweighed by the smoothing delay introduced at the traffic shaper.

For the case in which streams traverse multiple congested hops, smoothing can indeed provide significant benefits to the network-clients. The degree to which smoothing is beneficial is determined by various factors: the network load, the number of hops traversed, the burstiness of the stream, and the desired end-to-end delay bound. Thus, our scheme incorporates these factors to tell network clients if they should smooth, and if so, how much they should smooth to achieve the maximum benefit. We also show how the scheme can be integrated with a signaling protocol to provide a stream with its optimal smoothing rate at connection-setup time.

Finally, we use 30-minute traces of MPEG compressed video to empirically investigate the scheme. These investigations confirm our result regarding the ineffectiveness of smoothing over a single hop. Additionally, the empirical results show that through proper selection of a smoothing rate, smoothing can result in considerable benefits in multiple-hop scenarios.

## 2 Deterministic Smoothing

In this section, we first review the D-BIND traffic model of [10], which we will use in the remainder of this paper to characterize traffic streams. We then provide a formal definition of "smoother", which directly relates to a stream's worst-case resource requirements. From this definition, we introduce a D-BIND traffic shaper which smooths a traffic stream. Finally, we show how the smoothing delay and buffer requirements of a traffic shaper can be bounded.

### 2.1 The Deterministic-BIND Model

Compared to statistical service, deterministic service provides better QoS in that it provides a no-loss, no-delay-violation service. For the network to deliver such a service, it needs a deterministic upper bound on the arrivals of all sources receiving the service. This bound is determined by a parameterized traffic model that sources use to specify their traffic characteristics to the network.

As shown in [10], previous deterministic traffic models such as the $(\sigma, \rho)$ model [3] and the $(Xmin, Xave, I, Smax)$ model [5] cannot capture the property that sources exhibit burstiness over a wide variety of interval lengths. The Deterministic Bounding Interval Dependent (D-BIND) traffic model was introduced to address this issue. The key components of the D-BIND model are that it is *bounding*, required to provide deterministic QoS guarantees, and *interval-dependent*, needed to capture important burstiness properties of sources. This more accurate traffic characterization then allows for a higher network utilization for a given delay bound (see [10]).

Each deterministic traffic model uses parameters to define a traffic constraint function $b(t)$, which constrains or bounds the source over every interval of length $t$. Denoting by $A[t_1, t_2]$ the number of arrivals in the interval $[t_1, t_2]$, the traffic constraint function $b(t)$ requires that $A[s, s+t] \leq b(t)$, $\forall s, t > 0$. Note that $b(t)$ is a time-invariant deterministic bound since it constrains the traffic source over every interval of length $t$. For example, the $(\sigma, \rho)$ model is defined so that $A[s, s+t] \leq \sigma + \rho t$ for all $s, t > 0$.

The D-BIND model is defined via multiple rate-interval pairs $\{(R_k, I_k) | k = 1, 2, \cdots, P\}$, so that the constraint function is given by a piece-wise linear function:

$$b(t) = \frac{R_k I_k - R_{k-1} I_{k-1}}{I_k - I_{k-1}} (t - I_k) + R_k I_k, \quad I_{k-1} \leq t \leq I_k \tag{1}$$

with $b(0) = 0$. Thus, the rates $R_k$ can be viewed as an upper bound on the rate over every interval of length $I_k$, so that $A[t, t + I_k]/I_k \leq R_k \quad \forall\, t > 0, k = 1, 2, \cdots, P$.

Figure 1(a) shows a plot of the D-BIND rate-interval pairs for a 30 minute trace of an MPEG-compressed action movie (the trace is further described in Section 4). Plotting the bounding rate $R_k$ vs. the interval length $I_k$, the figure shows that the model captures the source's burstiness over multiple interval lengths. For example, for small interval lengths, $R_k$ approaches the source's peak rate, 5.87 Mbps. For longer interval lengths, $R_k$ decreases to the long term average rate of 583 kbps, which is total number of bits in the MPEG sequence divided by the length of the sequence.

Figure 1(b) shows the movie's D-BIND constraint function $b(t)$. As described by Equation (1), it is piece-wise linear and is based on the rate-interval pairs of Figure 1(a).

### 2.2 Definition of "Smoother"

Smoothing traffic in the case of the D-BIND model may be viewed from either of two equivalent viewpoints: the $(R_k, I_k)$ rate-interval pairs or the $b(t)$ constraint function. As shown in Figure 1, a source may be described by bounding rates over multiple interval lengths. Equation (1) indicates that a lower rate for a given interval length will result in a lower constraint curve, and hence, as we will show in Section 3.1, possibly more admissible connections in the network. Thus, smoothing can be viewed as transforming a stream with upper bounds $\{(R_k, I_k) \mid k = 1, 2, \cdots, P\}$ into one with upper bounds


(a) Rate-Interval Pairs (b) Constraint Function

Figure 1: D-BIND Characterization for Action Movie

$\{(\hat{R}_k, \hat{I}_k) \mid k = 1, 2, \cdots, P\}$, with $\hat{R}_k \leq R_k$ if $\hat{I}_k = I_k$. This transformation may be realized with the traffic shapers described in Section 2.3.

A second view of traffic shaping may be seen from the smoothed source's new D-BIND constraint function $\hat{b}(t)$:

$$\hat{b}(t) = \frac{\hat{R}_k \hat{I}_k - \hat{R}_{k-1} \hat{I}_{k-1}}{\hat{I}_k - \hat{I}_{k-1}}(t - \hat{I}_k) + \hat{R}_k \hat{I}_k, \quad \hat{I}_{k-1} \leq t \leq \hat{I}_k. \tag{2}$$

With $\hat{R}_k \leq R_k$ and Equations (1) and (2), we have that $\hat{b}(t) \leq b(t)\ \forall t$.

More formally, we define a partial ordering of "smoothness" in the following manner.

**Definition 1** *If traffic source $j$ has traffic constraint function $b_j(t)$ and source $k$ has traffic constraint function $b_k(t)$, then source $j$ can be considered* smoother *or less bursty than source $k$ if $lim_{t\to\infty} b_j(t)/t = lim_{t\to\infty} b_k(t)/t$ and $b_j(t) \leq b_k(t)\ \ \forall t$.*

The above definition allows us to compare the smoothness of sources that have the same long-term (upper) average rate. It also points out that while the smoothed curve will be less bursty than the original one, it will not in general be constant bit rate. The key advantage of our definition over previous measures of smoothness is that it directly relates to the maximum network resources required by one or more streams (this will be further discussed in Section 3). Previous measures of smoothness (or burstiness) utilize inter-packet arrival variances [6], majorization techniques [13], and $\sigma(\rho)$ "burstiness curves" [11]. While such smoothness measures achieved the goals of these respective works, they do not most accurately determine the maximum network resources required by the individual and multiplexed streams. For example, statistical measures of burstiness such as variance cannot be used to provide absolute upper bounds on a stream's resource requirements. As well, the majorization metric arranges a video stream's frame sizes in order from the largest to the smallest, which loses the correlation structure of the stream. Finally, the "burstiness curve" burstiness measure requires a concave $b(t)$, which in turn less accurately bounds a stream's required resources than the D-BIND model considered here.

### 2.3 The Shaper

Figure 2 illustrates the role of a traffic shaper in a network that supports guaranteed service. In this model, a traffic stream may be smoothed before being transmitted into the network. Inside the network, additional mechanisms may exist that perform rate control on the traffic. Such rate control may also shape the traffic stream, but its purpose is rather to ensure that the stream conforms to its specified traffic parameters such as D-BIND or $(\sigma, \rho)$. That is, rate-control inside the network can be viewed as a policing mechanism that protects the network and other traffic streams from connections that exceed their specified bounds. If a violation does occur, the network may buffer, mark, or drop the violating packet, depending on the rate-controlling mechanisms. Alternatively, the traffic *shaper* that we are considering will only buffer, but not drop or mark packets. Its purpose is to smooth the traffic when needed, in order to reduce the stream's network requirements. The stream will then traverse the policing mechanism unaffected if it has properly specified and shaped its traffic.

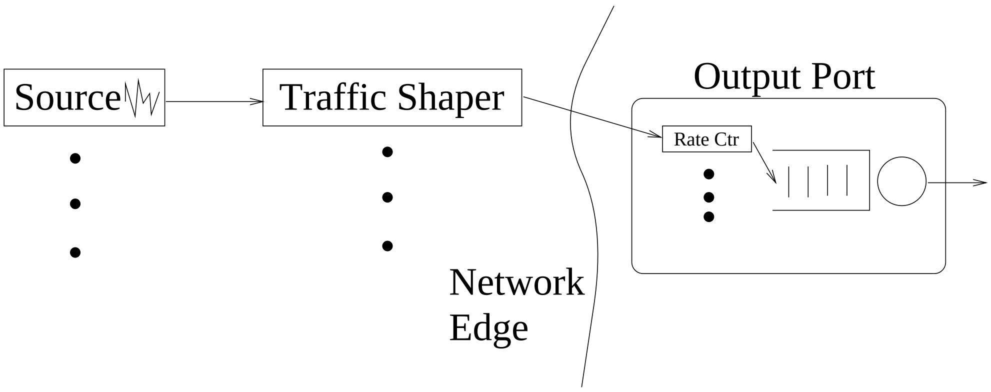



Figure 2: Role of the Traffic Shaper

Different mechanisms can be used to shape the traffic, with the complexity of the shaper affecting the range of shapes or constraint functions $b(t)$, that can be achieved. Two such mechanisms are a D-BIND shaper and a FIFO (a first-in first-out queue).

We define a D-BIND *shaper* as a buffer, together with the D-BIND *policing* mechanism described in [10]. The D-BIND policer enforces $P$ rate-interval pairs with $P$ "non-concave buckets" – mechanisms similar to leaky buckets. Thus, if a source with unsmoothed D-BIND parameters $\{(R_k, I_k) \mid k = 1, 2, \cdots, P\}$ wishes to smooth to parameters $\{(\hat{R}_k, \hat{I}_k) \mid k = 1, 2, \cdots, P\}$, with $\hat{R}_k \leq R_k$ if $\hat{I}_k = I_k$, it can use a *buffered* D-BIND policer, where the policer has parameters $\{(\hat{R}_k, \hat{I}_k), k = 1, \cdots, P\}$. The buffer is needed since a policer alone would drop "violating" packets, i.e., those sent beyond the $(\hat{R}_k, \hat{I}_k)$ rates. The buffer allows packets to be delayed until they can be transmitted without violating the new smoothed D-BIND parameters, $(\hat{R}_k, \hat{I}_k)$. Thus, any desired piece–wise linear constraint curve $\hat{b}(t)$ can be constructed with the D-BIND smoother.

A second possible smoothing mechanism is a FIFO, i.e., a buffer together with a constant-rate server. After traversing the FIFO, the stream will be "smoother" as in Definition 1. For example, the stream will have its peak rate reduced to the FIFO's service rate.

### 2.4 Bounding Smoothing Delay $\tau_r$

Different traffic shapers allow different transformations of constraint functions from $b(t)$ to $\hat{b}(t)$. As described in Section 1, we are allowing the smoothers to buffer, but not drop packets. The maximum buffer size required by the smoothers and the maximum smoothing delay $\tau_r$ can be calculated with the techniques of [3]. Specifically, $\tau_r$, the worst case smoothing delay when smoothing from constraint function $b(t)$ to $\hat{b}(t)$ may be calculated as the maximum horizontal time distance between the two curves

$b$ and $\hat{b}$. That is,

$$\tau_r = \max_{t_2 > t_1} \{ t_2 - t_1 | \hat{b}(t_2) = b(t_1) \}. \tag{3}$$

In the remainder of this paper, we consider the FIFO smoothing mechanism. We use a FIFO rate $r$ that is between the peak-rate and the average-rate of the unsmoothed source, and bound the maximum delay in the smoother using Equation (3).

# 3 A Smoothing Scheme for Optimizing End-to-End QoS

In this section, we provide a scheme for determining which smoothing rate, if any, provides the maximum benefit to a network client, where clients may realize this benefit by receiving either a better quality- or price-of-service. For example, some network clients may be interested only in minimizing their total end-to-end delay bound. Other clients may want a certain end-to-end delay bound, for example, 200 msec for human inter-activity, and obtaining delay bounds below this value may not be useful. In this case, the client may choose to transmit its traffic in the smoothest possible way so that its end-to-end delay requirement is met. By transmitting smoother traffic, the network can improve its utilization (shown below in Lemma 1) so that this connection will receive a better price (assuming that a stream's price increases with its resource requirements).

In the analysis below, we approach the problem from a QoS or delay-bound perspective by considering a fixed number of connections or utilization, and providing a scheme for determining the optimal smoothing rate. This optimal rate is the one that provides the maximum *net* reduction in end-to-end delay bound. As we will show, smoothing is not always beneficial, in that, under some circumstances, neither delay bound nor utilization can be improved through smoothing. Thus, the scheme will determine *if* smoothing is beneficial, and if so, what *rate* provides the maximum benefit. The approach can also be applied to the corresponding utilization- or price-perspective.

First, we show how *end-to-end* delay can be bounded including smoothing, queueing, and propagation delays. Second, we show that over a single hop, smoothing never results in a net reduction in delay bound for the network client or a net increase in utilization for the network. Third, we show that over multiple hops, smoothing can indeed have an advantageous effect and we provide a condition that allows a network client to decide if smoothing is beneficial in a general multi-hop scenario. Finally, we describe our scheme that allows network clients to find their optimal smoothing rate. We also describe how the scheme can be integrated with a signaling or connection-establishment protocol.

## 3.1 Bounding End-to-End Delay

At a single server, queueing delay can be bounded using the delay analysis techniques of [2, 3, 14]. These works show that the delay bound provided to a connection by a server is a function of the server speed and of the traffic constraint functions of all the connections sharing the server. This function varies with different service disciplines. For example, if a FCFS scheduler with link speed $l$ is serving $N$ connections, each bounded by its respective constraint functions $b_j(t), j = 1, \cdots, N$, then an upper bound on delay for all connections is given by:

$$d = \frac{1}{l} \max_{t \in I_k} \{ \sum_{j=1}^{N} b_j(t) - lt \}. \tag{4}$$

With a bound on queueing delay, we can now state formally the intuitive notion that a less bursty source can better utilize network resources. We use $d$ to represent the delay bound for the *unsmoothed* stream that has constraint function $b(t)$, and $\hat{d}$ to represent the delay bound for the *smoothed* stream with constraint function $\hat{b}(t)$.

**Lemma 1** *If a source is smoothed as in Definition 1 from a constraint function $b(t)$ to a constraint function $\hat{b}(t)$, then the queueing delay bound $d$ for every source traversing a FCFS scheduler is reduced to $\hat{d} \leq d$. Equivalently, with smoother or less bursty sources, more connections can be multiplexed at a FCFS scheduler for a given queueing delay bound $d$.*

Proof: The FCFS queueing delay bound is given by Equation (4) as $d = \frac{1}{l}\max_{t \geq 0}\{\sum_{j=1}^{N} b_j(t) - lt\}$. If source $j$ is smoothed, then the $j^{th}$ term of the summation $b_j(t)$ is replaced with $\hat{b}_j(t)$. Since $\hat{b}(t) \leq b(t)$ for all $t$, it can only reduce the delay bound to $\hat{d} \leq d$. For the "equivalent" statement, consider the homogeneous sources with constraint function $b(t)$ so that Equation (4) can be rewritten as

$$N(d) = max\{n \mid nb(t) \leq l(d+t), t \geq 0\} \tag{5}$$

where $N$ is the number of admissible connections for the delay bound $d$. If $b(t)$ is reduced, $N$ can only increase. □

Lemma 1 can easily be applied to other service disciplines since queueing delay bounds can be expressed in terms of constraint functions.

To extend this delay bound at a single server to an end-to-end delay bound, we will restrict the class of service disciplines to rate-controlled service disciplines such as Rate Controlled Static Priority, Earliest Deadline First, or Hierarchical Round Robin [15]. This restriction will allow us to have better end-to-end delay bounds than if we consider, for example, end-to-end delay bounds for all work-conserving service disciplines as in [4].

With rate-controlled service disciplines, the stream's original traffic constraint function $b(t)$ (or $\hat{b}(t)$ if the source is smoothed) is reconstructed at each hop via mechanisms such as the multi-level leaky bucket. Thus, rate-control decouples the network nodes so that the end-to-end delay bound $D$ may be calculated by summing delay bounds $d_i$ at individual nodes [15]. For a source that performs smoothing, the end-to-end delay over $H$ hops is bounded by:

$$\hat{D} = \tau_r + \sum_{i=1}^{H} \hat{d}_i + \sum_{i=1}^{H} \pi_i \tag{6}$$

where $\pi_i$ is the propagation delay of the $i^{th}$ hop and $\tau_r$ is the smoothing delay bound of Equation (3).

Thus, while smoothing does reduce the queueing delay bound, it also introduces an additional delay $\tau_r$, due to buffering at the smoothing-FIFO. This delay contributes to the total end-to-end delay bound obtained by the stream. Therefore, from the perspective of delay bound, a source should smooth if the additional smoothing delay bound is less than the reduction in the queueing delay bound, i.e., if the total delay bound is reduced. Below, we present conditions for determining whether smoothing is advantageous to a network-client.

### 3.2 A Single-Hop Rule

Over a single network hop, the net effect of smoothing on the total end-to-end delay bound is stated in the following theorem.

**Theorem 1** *When homogeneous streams traverse a single-hop, the reduction in queueing delay bound for a FCFS server introduced by smoothing is outweighed by the additional smoothing delay. That is, smoothing will only increase the total delay bound, smoothing plus queueing, or* $\tau_r + \hat{d} \geq d$.

The proof is given in [8]. Theorem 1 therefore states that smoothing of homogeneous sources should never be performed in the single hop case since it will result in a net increase in the total end-to-end delay bound. While the next section addresses more general multi-hop scenarios, it will also show that if a stream traverses multiple hops with a single bottleneck node, the result will be similar to the above single-hop case.

### 3.3 A General Multi-Hop Condition

While the one-hop case above has a simple "never smooth" outcome, no such "always" or "never" statements can be made in multi-hop networks. The effectiveness of smoothing will depend on the network load, the number of hops traversed, the burstiness of the stream, and the desired delay bound. Thus, we provide an algorithm that can be integrated with the network's signaling protocol that tells a stream whether or not to smooth and what smoothing rate is appropriate.

When a stream traverses multiple hops, the analysis has an important additional component that differs from the one-hop case: while the smoothing delay is incurred only once at the stream's traffic shaper, queueing delays may be incurred at multiple nodes in a congested network. Thus, a smoother stream can reduce its queueing delay at *each* congested hop, resulting in a considerable decrease in its end-to-end delay bound. Therefore, in many cases with multiple congested hops, the reduction in queueing delay outweighs the added delay caused by smoothing, so that smoothing can often result in a net benefit to network clients.

The following proposition provides a condition for determining if smoothing provides a net benefit in terms of end-to-end delay bound or utilization for networks that use rate-controlled service disciplines.

**Proposition 1** *Consider a network in which each node $i$ has a rate-controlled service discipline that can provide an upper bound on queueing delay. If $\hat{d}_i$ is the queueing delay bound at hop $i$ for the smoothed source and $d_i$ is the original queueing delay bound at hop $i$, a stream that traverses $H$ hops will obtain a net reduction in end-to-end delay bound due to smoothing if the following condition holds:*

$$\tau_r < \sum_{i=1}^{H} (d_i - \hat{d}_i). \tag{7}$$

Proof: The proposition states that if the bound on smoothing delay $\tau_r$ is less than the total reduction in queueing delay across multiple hops, then smoothing is advantageous to the source. Without smoothing, the end-to-end delay bound is given by $D = \sum_{i=1}^{H} d_i + \sum_{i=1}^{H} \pi_i$. With smoothing, the end-to-end delay bound is $\hat{D} = \tau_r + \sum_{i=1}^{H} \hat{d}_i + \sum_{i=1}^{H} \pi_i$. A stream will have a reduction in its end-to-end delay bound if $\hat{D} < D$, which is the inequality stated in the proposition. □

Equation 7 indicates that as the number of *congested* hops increases, smoothing becomes more important. The reason for this is that the summands in the right hand side of the inequality are always positive (shown in Lemma 1), and therefore the additional benefit of smoothing can only increase with $H$. If only a single hop is congested, then for all but one of the hops, $d_i \approx \hat{d}_i$, and the situation will be similar to the single hop result of Theorem 1, in which smoothing is not beneficial.

Because rate-controlled service disciplines decouple the network nodes from one another, Proposition 1 applies to heterogeneous networks utilizing a wide range of service disciplines at the nodes [15]. The local delay bounds that constitute the sum in Equation (7) may then be calculated by the admission control tests that correspond to the local service discipline (see [9] for several examples).

### 3.4 Obtaining the Optimal Rate

In this section, we develop the previous multi-hop condition into a scheme that allows network clients to determine which smoothing rate provides them with the maximum benefit. We then describe how the scheme can be integrated with a signaling or connection-establishment protocol. Note that the scheme requires additional network support during connection-establishment phase only, but not during the data-delivery phase. Note also that the scheme does not require additional information from the client, since it is based on the constraint function for the traffic as defined in Section 2.1.

The maximum savings in end-to-end delay bound that a source can obtain by smoothing is given by:

$$\max_{R_1 \leq r \leq R_P} \{D - \hat{D}(r)\} \tag{8}$$

where $D$ and $\hat{D}(r)$ are calculated as in Section 3.1, and $R_1$ and $R_P$ are as defined by the D-BIND model in Section 2.1. If Equation (8) is maximized by the smoothing rate $r^*$, then $r^*$ is the optimal smoothing rate in that it attains the minimum end-to-end delay bound for the network client. If $r^* = R_1$, i.e., the optimal smoothing rate is the same as the peak rate, then *no* smoothing is the best policy.

For protocols that provide guaranteed services such as ATM or the Tenet Real-Time Protocol Suite [1], the smoothing scheme can be integrated into the signaling or connection-establishment protocol in the following manner. First, a source will determine its original, unsmoothed D-BIND rate-interval pairs (with the number of pairs set by the protocol). If the data to be transmitted is available ahead of time such as for stored video, then these parameters can be calculated directly from the trace. Otherwise, the algorithms in [16] can be used to determine the appropriate parameters for the unknown source.

Next, this traffic specification is sent to the signaling protocol along with the source's maximum end-to-end delay requirement $D$ and its request for a *deterministic* service. For practical reasons, including the latency of the call setup and the length of the signaling message, only a small number of FIFO-smoothing rates can be tested, i.e., the maximization of Equation (8) must be tested for a small number of rates $r$. This number, which we call $\Upsilon$, in practice will be fixed by the signaling protocol. A larger $\Upsilon$ would increase the granularity of the tested rates to more closely obtaining the optimal rate $r^*$; however, a larger $\Upsilon$ also requires more admission control tests and a larger signaling message. Each node will then calculate the delay bound for the $\Upsilon + 1$ smoothing rates between the source's peak rate $R_1$ and the upper average rate $R_P$, where $R_P$ is the bounding rate over the longest specified interval length, $I_P$. The delay bound calculation for rate $R_1$ is simply the calculation for the *unsmoothed* source since $R_1$ is the stream's peak rate. The smoothing rates to be tested can be chosen uniformly between $R_1$ and $R_P$ so that $r_\upsilon = R_1 - \frac{\upsilon}{\Upsilon}(R_1 - R_P)$ will be tested for $\upsilon = 0, 1, 2, \cdots, \Upsilon$.

Along the forward path of the signaling or connection-establishment message, each hop $i$ of the $H$ hops adds its local delay bound calculations for each of these $\Upsilon + 1$ smoothing rates, $\{d_i, \hat{d}_i^1, \hat{d}_i^2, \cdots, \hat{d}_i^\Upsilon\}$, to the local delay bounds calculated by the up-stream nodes. The final node or destination node then has $\Upsilon + 1$ bounds on end-to-end queueing delay. Using Equation (3), this node also calculates the $\Upsilon$

smoothing delays, so that it can determine which smoothing rate (if any smoothing at all) results in the smallest end-to-end delay bound. The best smoothing rate is the one that achieves

$$D_{min} = \min\{(\sum_{i=1}^{H} d_i), (\tau_1 + \sum_{i=1}^{H} \hat{d}_i^1)\ , (\tau_2 + \sum_{i=1}^{H} \hat{d}_i^2)\ , \cdots,\ (\tau_\Upsilon + \sum_{i=1}^{H} \hat{d}_i^\Upsilon)\}. \tag{9}$$

If the first term is the minimum, then no smoothing is the best alternative. Otherwise a smoothing-FIFO with rate $r_v$ should be used, where $r_v$ is the smoothing rate the achieves the minimum delay bound $D_{min}$.

Because of connection admission control, if the source's required end-to-end delay bound $D$ is less than $D_{min}$, then the call has to be rejected regardless of the smoothing policy. Otherwise, this last node of the path will make a final decision on the smoothing rate and will send a signaling message along the reverse path of the connection. This message will indicate that the connection has been accepted and will contain the source's final traffic specification.

As described above, the above framework can also be viewed in terms of price or network utilization. For example, if there is a *price* associated with the network's resources at different hops or at the network edge, this price could be incorporated into the smoothing decision of Equation (9) to encourage or discourage the use of certain resources. If the network has a fee structure for guaranteed-performance connections, this price could be the actual monetary price of the reserved resources, which would presumably encourage some intended type of behavior. Thus, while we have focused on end-to-end delay as the quantity to be minimized, appropriate weighting of reserved resources allows us to use other ways to realize the benefits of smoothing.

# 4 Experimental Results

In this section, we use a 30-minute trace of MPEG compressed video to empirically investigate the smoothing scheme introduced in Section 3. This investigation provides insight into the smoothing-benefits that are possible for real traffic loads. The trace is of an action movie (a "James Bond" film), digitized to 384 by 288 pixels and compressed at 24 frames per second with frame pattern **IBBPBBPBBPBB**.

Our experiments consist of the following steps. First, we calculate the D-BIND rate-interval pairs for the trace as shown in Figure 1(a). We then consider the network of Figure 3, which consists of $H$ hops with rate-controlled FCFS servers at the switching nodes. The switches are output buffered with link speeds of $l = 155$ Mbps. For the unsmoothed source, we calculate the maximum number of connections $N$ that can be multiplexed in this network, so that no packets are dropped or violate their end-to-end delay bound $D$. The calculation is based on the D-BIND parameters as determined above, together with the admission control test of Equation (5).

This number of connections will result in a certain average utilization of the network. This average utilization is the total number of bits transmitted by all of the sources in their 30-minute duration divided by the total number of bits that can be served by the multiplexer in 30 minutes (30 minutes times 155 Mbps). For $N$ homogeneous connections, the utilization can be written as $NR_\infty/l$ where $R_\infty$ is the long-term average rate of the video stream, i.e., the total number of bits in the stream divided by the total time, 30 minutes.

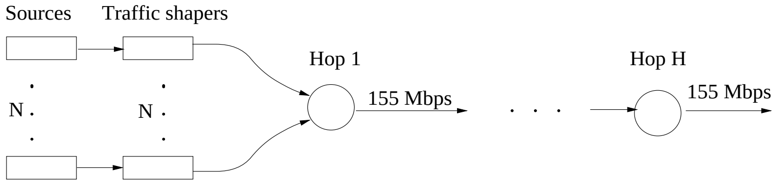
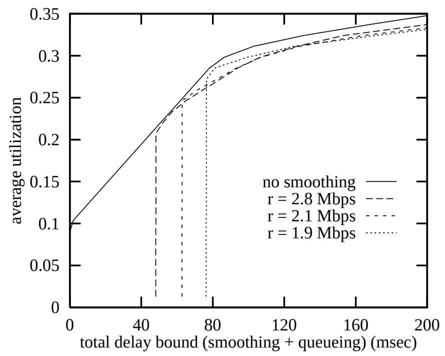



Figure 3: Network Topology

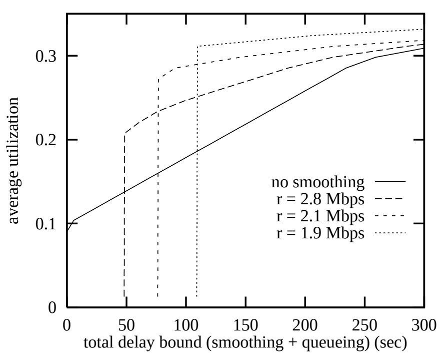
Next, we consider various rates of the smoothing-FIFO $r$ to investigate the range of benefits achieved by the smoothing scheme. Each different rate $r$ transforms the stream's original rate-interval pairs $(R_k, I_k)$ into smoothed pairs $(\hat{R}_k, \hat{I}_k)$. As described in Section 3, smoothing causes a delay inside the traffic shaper, which is upper-bounded by $\tau_r$ as in Equation (3). With the new traffic characterization, we again calculate the maximum number of smoothed streams $\hat{N}$ that can be multiplexed for an end-to-end delay bound $\hat{D}$, where $\hat{D}$ includes both smoothing delay and the queueing delays accumulated at each hop as in Equation (6).

Note that in the above scenario, "worst-case" end-to-end deterministic guarantees are provided, in that even if all sources are exactly synchronized in the worst possible way (e.g., all sources' **I** frames synchronize), the delay-bounds are still met. The utilizations reported are those achieved by deterministic network clients alone: best-effort traffic can always utilize remaining network resources.

A network client is ultimately interested in its *net* benefit from smoothing, or the total end-to-end delay bound including both smoothing and queueing delay. Figure 4 shows utilization versus the stream's total delay bound as in Equation 6. Figure 4(a) depicts the case of a stream traversing one hop and Figure 4(b) depicts the case of three hops. Figure 4(a) confirms the result of Theorem 1 which states that smoothing never results in an increased network utilization for a given delay bound when homogeneous streams traverse a single hop. For example, for an unsmoothed source and a delay bound of 80 msec, a 29% utilization is achievable. Alternatively, when the source is smoothed with a FIFO of rate $r = 2.1$ Mbps, the achievable utilization is reduced to 27%.




(a) One Hop




(b) Three Hops

Figure 4: Average Utilization vs. Total Delay Bound for Various Smoothing Rates

Figure 4(b) depicts the case where the streams traverse three hops rather than one. In this case, smoothing results in a substantial benefit. For example, for a total end-to-end delay bound of 100 msec, without smoothing, an average utilization of 17% is achieved. With smoothing via a FIFO of rate 2.1 Mbps, this utilization is improved to 29%. Thus, in this case, smoothing has resulted in a 70% increase in the number of admissible connections for the same end-to-end delay bound. The reason for this utilization improvement is explained by Proposition 1: while the smoothing delay occurs only once at the source, there is a queueing delay at each of the congested network nodes, thus providing a net benefit for smoothing.

## 5 Conclusions

In this paper, we have introduced a smoothing scheme that allows a network client to choose a smoothing policy that provides the greatest benefit. This benefit may be realized with a better quality of service (via a reduced delay bound) or a better price of service (via increasing the network utilization). We considered traffic shapers that buffer, but do not-drop, incoming packets. We then bounded the smoothing delay and combined this bound with delay bounds at the network multiplexers to provide an end-to-end no-loss, no-delay-violation QoS.

We showed analytically that when streams traverse only a single congested hop and sources are homogeneous, smoothing never results in a net benefit to the network client. That is, while smoothing does reduce the queueing delay bound, this reduction in delay is outweighed by the added smoothing delay. We confirmed this result by performing experiments with 30-minute traces of MPEG-compressed video.

When streams traverse multiple congested hops, the situation changes to favor smoothing, primarily because while the smoothing delay is incurred only once at the entrance of the network, the queueing delay may be incurred at each congested hop. Our scheme therefore allows a network client to determine if smoothing is beneficial at all, and if so, what smoothing rate will provide the greatest benefit. Using the MPEG traces, we provided insights into typical delay bounds that can be obtained by a realistic workload and found that in many cases smoothing indeed resulted in substantial benefits that can be seen either as reductions in end-to-end delay bound or as increased network utilization.

## References


[1] A. Banerjea, D. Ferrari, B. Mah, M. Moran, D. Verma, and H. Zhang. Tenet real-time prococol suite: Design, implementation, and experiences. *IEEE/ACM Transactions on Networking*, 1995. To appear.

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

[3] R. Cruz. A calculus for network delay, part I : Network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–121, January 1991.

[4] R. Cruz. A calculus for network delay, part II : Network analysis. *IEEE Transactions on Information Theory*, 37(1):121–141, January 1991.

[5] D. Ferrari and D. Verma. A scheme for real-time channel establishment in wide-area networks. *IEEE Journal on Selected Areas in Communications*, 8(3):368–379, April 1990.

[6] R. Gusella. Characterizing the variability of arrival processes with indices of dispersion. *IEEE Journal on Selected Areas in Communications*, 9(2):203–211, February 1991.

[7] H. Kanakia, P. Mishra, and A. Reibman. An adaptive congestion control scheme for real-time packet video transport. In *Proceedings of ACM SIGCOMM'94*, pages 20–31, San Francisco, CA, September 1993.

[8] E. Knightly and P. Rossaro. Smoothing and multiplexing tradeoffs for deterministic performance guarantees to VBR video. Technical Report TR-95-033, International Computer Science Institute, Berkeley, CA, July 1995.

[9] E. Knightly, D. Wrege, J. Liebeherr, and H. Zhang. Fundamental limits and tradeoffs for providing deterministic guarantees to VBR video traffic. In *Proceedings of ACM SIGMETRICS'95*, Ottowa, Ontario, May 1995.

[10] E. Knightly and H. Zhang. Traffic characterization and switch utilization using deterministic bounding interval dependent traffic models. In *Proceedings of IEEE INFOCOM'95*, pages 1137–1145, Boston, MA, April 1995.

[11] S. Low and P. Varaiya. Burstiness bounds for some burst reducing servers. In *Proceedings of IEEE INFOCOM'93*, pages 2–9, San Francisco, CA, March 1993.

[12] I. Richardson and M. Riley. Usage parameter control cell loss effects on MPEG video. In *Proceedings of ICC'95*, pages 970–974, Seattle, WA, June 1995.

[13] J. Salehi, Z. Zhang, J. Kurose, and D. Towsley. Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing. Technical Report TR-95-98, Department of Computer Science, University of Massachusetts, Amherst, MA, November 1995.

[14] H. Zhang and D. Ferrari. Improving utilization for deterministic service in multimedia communication. In *Proceedings of 1994 International Conference on Multimedia Computing and Systems*, pages 295–304, Boston, MA, May 1994.

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

[16] H. Zhang and E. Knightly. RED-VBR: A new approach to support VBR video in packet-switching networks. In *Proceedings of IEEE Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV'95)*, pages 275–286, Durham, NH, April 1995.