On the effects of smoothing for deterministic QoS

Edward W Knightly†§ and Paola Rossaro‡‖

† ECE Department, Rice University, Houston, TX 77251-1892, USA
‡ Wind River Systems, 1010 Atlantic Avenue, Alameda, CA 94501, USA

Abstract. In order for a network to provide end-to-end guarantees on quality of service (QoS), it must allocate its resources according to the traffic characteristics and performance requirements of its clients. However, the burstiness of typical variable bit rate traffic streams makes it difficult to provide the QoS guarantees that the network's clients require and simultaneously make efficient use of network resources. In this paper, we investigate the impact that smoothing traffic at the network's edge has on both the client's QoS and on the network's utilization. Considering the case of end-to-end deterministic QoS guarantees, we explore the analytical foundations of smoothing and identify the scenarios in which smoothing is beneficial to the network or its clients. Moreover, we quantify the potential benefits of smoothing with a set of experiments based on traces of MPEG-compressed video in heterogeneous multi-hop networking environments.

1. Introduction

Emerging distributed real-time applications such as real-time video and audio and distributed medical imaging have stringent requirements on the quality of service (QoS) that they obtain from the network. In order for these applications to be usable, they require the network to *guarantee* their performance parameters such as throughput, delay, and delay-jitter. Integrated services networks that employ a resource reservation scheme together with priority service disciplines inside the network (as in [5]) provide the means for giving network clients the end-to-end QoS guarantees that they require.

The class of applications with the most stringent QoS requirements will need a *deterministically* guaranteed service which ensures that all packets of a connection will meet the guaranteed end-to-end delay bound, and that no packets will be dropped due to buffer overflows. This is as opposed to a *statistical* QoS guarantee such as in [9] and [19], in which *probabilistic* performance guarantees are provided, e.g. a loss probability guarantee of 10^{-6}. 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 [11], and with more accurate admission control conditions as in [2], [3], [12], and [22], significant utilization improvements are possible. Hence, a deterministic service has the means to support variable bit rate (VBR) traffic.

If all of the traffic streams obtaining a guaranteed service are constant bit rate (CBR), then full utilization of network resources can be easily achieved. However, realistic multimedia traffic streams such as compressed video and medical imaging are *variable* bit rate in nature with a high degree of burstiness. For bursty VBR traffic sources, it is generally difficult to provide the good QoS that network clients desire, and to simultaneously achieve high network utilization.

Intuitively, *smoothing* a traffic stream shapes the stream to be more like a CBR stream. Smoother traffic streams ostensibly achieve higher utilization or better QoS. However, reducing a stream's burstiness comes at a cost in that 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 [8]. 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 [16]. Finally, a source can be smoothed by adding variable delays to packets, i.e. by spreading out bursts over time.

Thus, while reducing the burstiness of VBR sources through traffic *smoothing* has potential benefits in terms of network utilization and end-to-end delay bounds, these benefits must be weighed against the *costs* of smoothing: loss of perceptual quality, loss of information, or added smoothing delays.

In this paper, we study the net end-to-end effect of smoothing in the context of a deterministic service. We consider smoothing elements which buffer, but do not drop, incoming packets. Thus, unlike the aforementioned

§ E-mail address: knightly@ece.rice.edu
‖ E-mail address: paola@wrs.com

0967-1846/97/010003+13$19.50 © 1997 The British Computer Society, The Institution of Electrical Engineers and IOP Publishing Ltd

bandwidth- and loss-smoothing approaches, our smoothing scheme does not degrade a stream's perceptual quality or allow packet losses. We also avoid violations of end-to-end delay-bound by controlling the delay incurred inside the smoothing elements. We utilize both analytical and empirical investigations to study the factors that influence the effectiveness of smoothing. We identify the set of scenarios under which smoothing results in a net benefit to either the network client, via improved QoS, or to the network itself, via improved utilization. We take into account both the costs of smoothing, i.e. added delays at the network edge, as well as the benefits of smoothing, i.e. more efficient allocation of resources inside the network. Our analysis focuses on realistic networking scenarios, with end-to-end QoS considerations, heterogeneous traffic mixes, and incorporation of end-system smoothing with network performance.

We first investigate the primary components of a deterministic service that are required to ascertain the end-to-end impact of smoothing. These factors include the parameterized traffic model that streams use to describe their traffic, the packet service discipline that the network uses to schedule packets, and the connection admission control (CAC) algorithm that is used to determine whether or not a new connection can be admitted such that all connections obtain their promised QoS guarantees.

Based on these components of a deterministic service, we then investigate the analytical foundations for smoothing. We begin with a definition of 'smoother', or a partial ordering of smoothness and burstiness. Unlike previous burstiness measures (e.g. [7, 14, 18]), we show that our new definition of burstiness and smoothness relates directly to a stream's maximum resource requirements inside the network and hence the QoS that the connection can obtain and the utilization that the network can achieve. We relate this burstiness definition to various traffic models and show how simple mechanisms can be used to reduce the burstiness of streams in a controlled manner, and we show how the delay inside the smoothing traffic-shaper can be upper-bounded.

With bounds on a stream's delay inside a traffic shaper, and assurance that no packets are dropped by the shaper, we provide a bound on the total end-to-end delay, including smoothing delay and queueing delay at each hop along the connection's path. We show how smoothing streams at the network edge reduces their queueing delay inside the network, and we present a scheme that allows network clients to determine the smoothing policy which yields the maximum benefit. If smoothing is beneficial, the network client can realize the benefits via either an improved *quality* of service (i.e. a reduction in end-to-end delay bound) or by improving the network's utilization, which may result in an improved *price* of service if the network's services are priced according to the quantity of resources reserved.

We show that when 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 streams, and the desired end-to-end delay bounds. 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 signalling 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 quantify the impact of smoothing on end-to-end QoS and network utilization. We perform a set of admission control experiments with heterogeneous traffic streams that traverse multiple hops in a network consisting of 155 and 622 Mbps links. We show how in certain scenarios, proper traffic shaping can result in substantial benefits, either with network clients obtaining a reduction in the end-to-end delay bound for a given set of admissible connections, or with the network achieving an increase in the number of admissible connections for a given QoS requirement. The experiments also confirm the analytical results which identify scenarios in which smoothing is *not* beneficial, i.e. one-hop scenarios where the added smoothing delay outweighs the reduction in queueing delay. Thus, our experiments quantify the potential benefits of traffic smoothing in realistic networking environments. For example, in a trace experiment with 85 MPEG connections multiplexed on a 155 Mbps link, smoothing according to our scheme resulted in a 71% net reduction in end-to-end delay bound, from 392 ms to 112 ms.

2. Deterministic service

A deterministic service provides network clients with a QoS guarantee that avoids any packet losses or delay-bound violations. In this section we review the components of a deterministic network service: the models which clients use to characterize their traffic to the network, the service disciplines that determine the order in which packets are scheduled, and the admission control algorithms, which determine whether or not a new connection can be admitted to the network.

2.1. Deterministic traffic models

In order for the network to deliver a deterministic service, it needs an upper bound on the arrivals of all sources which obtained the service. This bound is determined by a parameterized traffic model that sources use to specify their traffic characteristics to the network.

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 $A[s_1, s_2]$ the number of arrivals in the interval $[s_1, s_2]$, the traffic constraint function $b(t)$ requires that

$$A[s, s+t] \le b(t), \qquad \forall s, t > 0. \tag{1}$$

Hence, $b(t)$ is a *time-invariant* deterministic bound since it constrains the traffic source over every interval of length t.

In this paper, we consider two such worst-case traffic models: the (PCR, SCR, MBS) and the D-BIND model. We choose these two because the former model is the current standard of the ATM Forum [6], and the latter model's increased accuracy has been shown to lead to considerable improvements in network utilization [11].

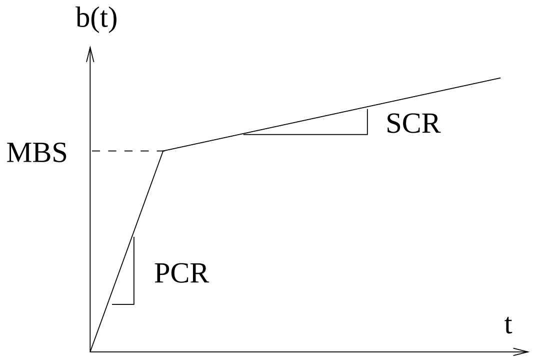


Figure 1. Constraint curve for (PCR, SCR, MBS) model.

2.1.1. (PCR, SCR, MBS) model. In [6], a traffic model is proposed that consists of three parameters: a stream's peak cell rate (PCR), its sustainable cell rate (SCR), and its maximum burst size (MBS).

These three parameters define a constraint function as follows:

$$b(t) = \begin{cases} PCR \cdot t & t \le \frac{MBS}{PCR} \\ & \\ SCR \cdot t + MBS(1 - \frac{SCR}{PCR}) & t > \frac{MBS}{PCR}. \end{cases} \tag{2}$$

The constraint function for the (PCR, SCR, MBS) model is shown in figure 1. As depicted by the figure, this traffic model allows sources to send at their specified peak cell rate of PCR for an interval of up to $\frac{MBS}{PCR}$. Over longer interval lengths the source is constrained by its sustainable cell rate SCR, a rate at which the source can send for an indefinite period of time.

Note that $b(t)$ is not an arrival sequence $A[0, s]$ *per se*, since t is an interval length and not time. For example, if a source *does* transmit exactly according to its constraint function and has $A[0, t] = b(t)$, it will send at its peak rate PCR for only $\frac{MBS}{PCR}$, and then at rate SCR for the remaining duration of the connection. As an alternative example, if a source sends at its peak rate for $\frac{MBS}{PCR}$ seconds and then remains idle for the next $MBS(\frac{1}{SCR} - \frac{1}{PCR})$ seconds, it may then transmit at PCR for another interval of length $\frac{MBS}{PCR}$, and repeat these peak-rate bursts and idle periods indefinitely. Clearly, an infinite number of arrival sequences are possible that are bounded by a given constraint function $b(t)$, and $b(t)$ must not be viewed strictly as an arrival sequence.

2.1.2. D-BIND model. As shown in [11], a deterministic traffic model such as the (PCR, SCR, MBS) model does not capture the property that sources exhibit burstiness over a wide variety of interval lengths. If a traffic model does not accurately bound the real traffic, then resources may be unnecessarily overallocated for the connection. 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.

With the D-BIND model, clients specify their traffic to the network via multiple rate–interval pairs $(R_k,\ I_k)$, where a rate R_k is a bounding or worst-case rate over every interval of length I_k. With P rate–interval pairs, the D-BIND model defines a constraint function that is piecewise linear:

$$b(t) = \frac{R_k I_k - R_{k-1} I_{k-1}}{I_k - I_{k-1}} (t - I_k) + R_k I_k, \qquad I_{k-1} \le t \le I_k \tag{3}$$

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 $A[t, t + I_k]/I_k \le R_k,\ \forall\ t > 0,\ \ k = 1, 2, \ldots, P$.

Figure $2(a)$ 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 5). Plotting the bounding rate R_k versus 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 the total number of bits in the MPEG sequence divided by the length of the sequence. From all of the possible rate–interval pairs shown in the figure, a source specifies P of these to the network at connection setup time.

Figure $2(b)$ shows the movie's D-BIND constraint function $b(t)$. As described by equation (3), it is piecewise linear and is based on the rate–interval pairs of figure $2(a)$. The figure shows the maximum number of kilobits that the source transmits over any interval of length t, and indicates that the D-BIND model is capturing the temporal properties of the MPEG video. For example, the peak rate shown in figure $2(a)$ is caused by transmission of the largest I frame of the sequence. This can be seen in the constraint function with the large slope (slopes indicating rates) between $t = 0$ and $t = 42$ ms (the frame rate is 24 frames per second). Importantly, even in the worst case, a large I frame is followed by two typically smaller B frames, which is captured by the constraint curve's slope decreasing in the interval $t = 42$ ms to $t = 125$ ms. Next, a P frame is transmitted, and these tend to be smaller than I frames but larger than B frames. In [11], it was shown that the D-BIND model's ability to capture both micro- and macro-level burstiness of the video sequence leads to considerably higher utilization than that achieved with previous models.

2.2. Packet service discipline

A second component of deterministic service is the packet service discipline, or the rules or algorithms that determine the order in which packets are serviced when they are queued at a multiplexer.

Two facets of the service discipline are relevant to our discussion here: its ability to support a wide variety of

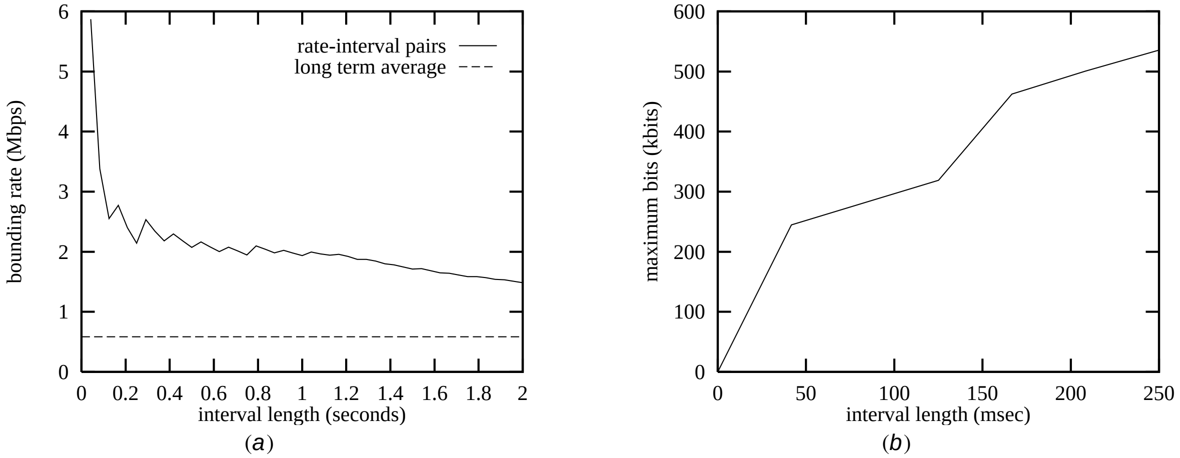


Figure 2. D-BIND characterization for action movie. (*a*) Rate–interval pairs; (*b*) constraint function.

QoS requirements and its implementation complexity. For example, the FCFS service discipline is the simplest to implement, since when the scheduler decides which packet to service next, it simply selects the packet at the head of the queue. However, FCFS can effectively support only *one* QoS level such as delay bound or packet loss probability. Thus, if the clients' QoS requirements are diverse, resources will be wasted. Experiments along these lines can be found in [20].

Alternatively, a service discipline such as earliest deadline first (EDF) can support a full range of packet delay bounds and loss probabilities since it is based on a dynamic priority mechanism. However, implementation of EDF is considerably more complex than FCFS since when the scheduler selects the next packet for transmission it must search among all packets for the one with the lowest deadline.

In this paper, we consider a static priority (SP) scheduler as a middle ground between FCFS and EDF. SP schedulers are simpler than EDF since the priorities are fixed and no searching or sorting of packets is required. Moreover, SP supports more QoS types than FCFS, with the number of QoS levels corresponding to the number of priority queues in the scheduler. For example, in [21], a rate-controlled static priority scheduler is proposed in which each priority level has an associated delay bound and connections are shaped, or rate-controlled, before being queued. Further discussion of the impact of service disciplines on deterministic service can be found in [20].

2.3. Admission control

Connection admission control (CAC) for deterministic service requires calculation of the worst-case buffer size and delay bound for a collection of streams aggregating at a multiplexer. A specific CAC algorithm is a function of the packet service discipline and we first consider FCFS. The following theorem provides an absolute upper bound on delay for a FCFS scheduler.

Theorem 1. *Consider a scheduler that services packets in first-come first-serve order. For $j = 1, \ldots, N$ multiplexed connections constrained by their respective constraint functions $b_j(t)$, and with a link speed l, a deterministic upper bound on delay for all connections is given by*

$$d = \frac{1}{l} \max_{t \geq 0} \left\{ \sum_{j=1}^{N} b_j(t) - lt \right\}. \tag{4}$$

Proof. The evolution of the multiplexer's behaviour over time can be described by Lindley's equation [13]:

$$q(s+\delta) = \left(q(s) + \sum_j A_j[s, s+\delta] - l\delta \right)^+ \tag{5}$$

where $q(s)$ is the backlog or queue length at time s, $(x)^+ = maximum(0, x)$, and $A_j[s, s+\delta]$ are the *actual* arrivals from connection j in the interval $[s, s+\delta]$.

From equation (5,) it follows that

$$q(s) = \max_{\tau \leq s} \left\{ \sum_{j=1}^{N} A_j[\tau, s] - l(s-\tau) \right\}. \tag{6}$$

For the FCFS service discipline, an upper bound on the length of time that a packet is delayed in the multiplexer is given by

$$d = \max_s q(s)/l \tag{7}$$

or the maximum queue length over all time s divided by the link speed. Hence, we have that

$$d = \frac{1}{l} \max_s \max_{\tau \leq s} \left\{ \sum_{j=1}^{N} A_j[\tau, s] - l(s-\tau) \right\}. \tag{8}$$

Next, we utilize that

$$\max_{\tau \leq s} \left\{ \sum_{j=1}^{N} A_j[\tau, s] \right\} \leq \sum_{j=1}^{N} \max_{\tau \leq s} A_j[\tau, s] \tag{9}$$

and

$$A_j[\tau, s] \leq b_j(s-\tau) \tag{10}$$

for $s \geq \tau$.

Combining equations (9) and (10), equation (4) follows. □

The techniques in the above proof are based on the results of [3]. Delay bounds for other service disciplines more suited to providing integrated services can also be derived. In [20], admission control conditions for earliest-due-date, static priority, and FCFS are reported.

For example, for a static priority scheduler with n priority levels and a link speed l, the maximum delay of any packet at priority level k is bounded above by:

$$d_k = \max\left\{t \geq 0 | b'_k(t) \geq lt\right\} \tag{11}$$

and $b'_k(\alpha)$ is defined for all α by

$$b'_k(\alpha) = \max_{\beta \geq 0}\left\{\overline{s} + \sum_{j \in C_k} b_{k,j}(\beta) + \sum_{q=1}^{k-1}\sum_{j \in C_q} b_{q,j}(\alpha+\beta) - l\beta\right\} \tag{12}$$

where the maximum packet size is $\overline{s}$, C_q is the set of connections at level q, and the jth connection in C_q satisfies the traffic constraint function $b_{q,j}(\cdot)$.

Equations (4) and (11) can be used as CAC tests for FCFS and SP schedulers in that they can test whether or not a set of connections can be multiplexed so that each packet of each connection can be delivered within the delay bound that is guaranteed to the network client. For CAC, the theorems may be used to test if a *new* connection can be admitted so that all connections, including the new one, obtain their respective delay guarantees. Hence, the CAC tests and delay bounds of equations (4) and (11) can also be viewed in terms of the maximum number of admissible connections for a given QoS. For example, for FCFS, equation (4) can be rewritten to express the maximum number of admissible connections as a function of the delay bound:

$$N(d) = \max\left\{n \left\| \frac{1}{l}\max_{t \geq 0}\left\{\sum_{j=1}^{n} b_j(t) - lt\right\} \leq d\right.\right\}. \tag{13}$$

Note also that the CAC tests are written in terms of the constraint function $b(t)$ rather than traffic model parameters such as PCR or rate–interval pairs. Thus, the system has flexibility in how network clients characterize their traffic.

In summary, a deterministic traffic model bounds the arrivals of the traffic streams and parameterizes a constraint function $b(t)$. Once the service discipline at a multiplexer is defined, CAC tests can be derived to determine the maximum number of connections that can be multiplexed such that all connections obtain their required QoS.

3. Deterministic smoothing

As motivated in section 1, the goal of smoothing is to reduce a stream's resource requirements from the network so that clients can obtain either a better QoS or a cheaper price-of-service from the network. However, these improvements must be weighed against the costs of smoothing such as the delay incurred at the entrance of the network.

In this section we provide a formal definition of 'smoother' which directly relates to a stream's worst-case resource requirements. From this definition, we introduce a traffic shaper which smooths a traffic stream. Lastly, we show how the smoothing delay and buffer requirements of a traffic shaper can be bounded so that, as a consequence, end-to-end delay can also be bounded.

To distinguish between parameters of the smoothed and unsmoothed stream, we will denote smoothed parameters with a hat (e.g. $\hat{R}_k$) and unsmoothed parameters without (e.g. R_k).

3.1. definition of 'smoother'

In order to investigate the impact of smoothing on service provisioning, we first define what it means for one stream to be *smoother* than another, or what it means to smooth a traffic stream. We define a partial ordering of 'smoothness' as the following.

Definition 1. *If traffic stream j has traffic constraint function $b_j(t)$ and stream k has traffic constraint function $b_k(t)$, then stream j can be considered* smoother *or less bursty than stream k if*

$$b_j(t) \leq b_k(t) \ \ \forall t \tag{14}$$

and

$$\lim_{t\to\infty}\frac{b_j(t)}{t} = \lim_{t\to\infty}\frac{b_k(t)}{t}. \tag{15}$$

The primary motivation for defining a smoothness or burstiness ordering as in definition 1 is that it dictates that a *smoother* or less bursty traffic stream requires fewer network resources than a more bursty traffic stream. This property will be shown below. As a secondary part of the definition, we only compare streams with the same long-term average rate. The reason is that we will be considering *lossless* smoothing in which a stream's packets may be delayed in various manners, but not dropped.

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 4). Previous measures of smoothness (or burstiness) utilize inter-packet arrival variances [7], majorization techniques [18], and $\sigma(\rho)$ 'burstiness curves' [14]. 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.

Note from definition 1 that while the smoothed traffic stream will be less bursty than the original one, it will not in general be constant bit rate.

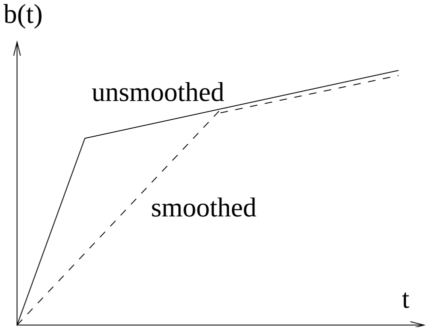


Figure 3. Smoothed traffic for (PCR, SCR, MBS) model.

3.1.1. (PCR, SCR, MBS) smoothing.

Consider a stream with traffic parameters (PCR, SCR, MBS) so that its arrivals $A[0, s]$ are bounded by the constraint function in equation (2). In order to smooth the stream without losing packets, SCR cannot be reduced. A less bursty stream will be one with a reduced PCR, but a possibly greater MBS. An example of such smoothing is shown in figure 3.

Here, the maximum burst size of the smoothed stream $\widehat{MBS}$ is larger than that of the unsmoothed stream, but $\widehat{PCR} < PCR$ and $\widehat{SCR} = SCR$.

Thus, smoothing for the case of the (PCR, SCR, MBS) model amounts to reducing the peak rate in a controlled manner by spacing or delaying packets. Comparatively, the D-BIND model allows greater flexibility in the shape of the smoothed traffic, potentially resulting in better use of network resources.

3.1.2. D-BIND smoothing.

Smoothing traffic in the case of the D-BIND model may be viewed from either of two equivalent view points: the (R_k, I_k) rate–interval pairs or the $b(t)$ constraint function. As shown in figure 2, a source may be described by bounding rates over multiple interval lengths. Equation (3) 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 5, possibly more admissible connections in the network. Thus, smoothing can be viewed as transforming a stream with upper bounds $\{(R_k, I_k) \mid k = 1, 2, \ldots, P\}$ into one with upper bounds $\{(\hat{R}_k, \hat{I}_k) \mid k = 1, 2, \ldots, P\}$, with $\hat{R}_k \le R_k$ if $\hat{I}_k = I_k$ and $\hat{R}_P = R_P$. This transformation may be realized with the traffic shapers described in section 3.2.

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

$$\hat{b}(t) = \frac{\hat{R}_k \hat{I}_k - \hat{R}_{k-1} \hat{I}_{k-1}}{\hat{I}_k - \hat{I}_{k-1}} (t - \hat{I}_k) + \hat{R}_k \hat{I}_k, \quad \hat{I}_{k-1} \le t \le \hat{I}_k. \tag{16}$$

With $\hat{R}_k \le R_k$ and equations (3) and (16), we have that $\hat{b}(t) \le b(t)\ \forall t$.

Thus, the D-BIND model allows sources to be shaped from their original piecewise linear constraint function to a second piecewise linear constraint function, provided that the latter one is less than or equal to the former for all interval lengths.

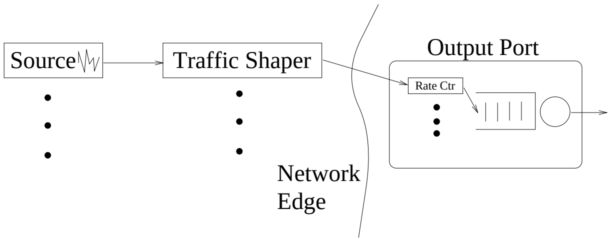


Figure 4. Role of the traffic shaper.

3.2. The shaper

Figure 4 illustrates the role of a traffic shaper in a network that supports guaranteed services. In this scenario, a traffic stream may be smoothed before being transmitted into the network. Inside the network, additional mechanisms may also shape the traffic, but for different purposes. For example, a rate controller inside the network as in figure 4 may be used to *police* traffic, to ensure that traffic streams conform to their specified traffic parameters. Policing therefore protects the network and other traffic streams from connections that exceed their specified traffic 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 with the goal of obtaining a better quality- or price-of-service. The stream will then traverse the policing mechanism unaffected if it has properly specified and shaped its traffic.

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 [11]. 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 $\{(R_k, I_k) \mid k = 1, 2, \ldots, P\}$ wishes to smooth to parameters $\{(\hat{R}_k, \hat{I}_k) \mid k = 1, 2, \ldots, P\}$, with $\hat{R}_k \le R_k$ if $\hat{I}_k = I_k$, it can use a *buffered* D-BIND policer, where the policer has parameters $\{(\hat{R}_k, \hat{I}_k), k = 1, \ldots, P\}$. 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 piecewise 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.

As an example of the effects of smoothing on a traffic stream, figure 5 shows the rate–interval pairs for the action movie before and after being smoothed by a FIFO. With

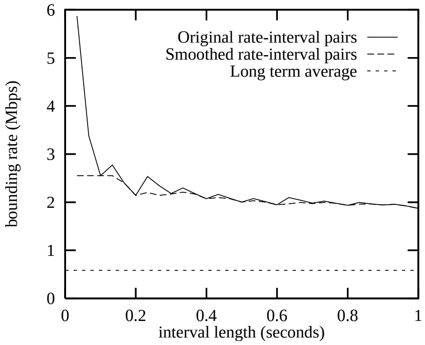


Figure 5. Smoothed D-BIND rate–interval pairs.

the unsmoothed stream, the peak rate is 5.9 Mbps. After smoothing with a FIFO of rate 2.6 Mbps, the peak rate is reduced to the rate of the FIFO. Moreover, as shown in the figure, the stream's bounding rates over all interval lengths are reduced.

3.3. Bounding smoothing delay τ_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 τ_r can be calculated with the delay bounding techniques described in section 2.3. Specifically, τ_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,

$$\tau_r = \max_{t_2 > t_1}\{t_2 - t_1 | \hat{b}(t_2) = b(t_1)\}. \qquad (17)$$

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 (17). We use the FIFO rather than the D-BIND shaper because it is simpler; however, the D-BIND shaper can more precisely shape traffic, and may provide better smoothing gains than those reported here for the FIFO smoothing.

4. Smoothing for improved QoS

A network client can realize a benefit from smoothing in one of two manners: improved quality or price-of-service. For example, one network client may want the best possible QoS such as the minimum possible end-to-end delay bound, regardless of the resulting price-of-service. In this case, the network client will smooth its traffic only if it results in a net reduction in end-to-end delay bound.

A second network client may have a certain end-to-end delay bound requirement, for example, 200 ms for human interactivity, and obtaining a delay bound below this value is not useful. In this case the client chooses 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 client will get a lower price, assuming that a stream's price-of-service increases with its resource requirements.

Decreasing a connection's price of service is analogous to improving its quality of service since smoothing can potentially allow for either more admissible connections for a given delay bound, or a lower delay bound for the same number of connections. The latter case improves the connections' QoS and the former their price of service, assuming that price increases with resource requirements. In the remainder of this paper, we will focus on using smoothing to improve *quality* of service.

In this section, we investigate how the smoothing techniques described in the previous sections can be used by network clients to improve their QoS in certain scenarios. We investigate the scenarios under which a net benefit is possible and quantify the potential gains in section 5.

For notation, 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 $\hat{b}(t)$.

4.1. Bounding end-to-end delay

To ascertain the net impact of smoothing, we first describe how end-to-end delay can be bounded from its component delay bounds. At a single server, queueing delay can be bounded using the delay bounding techniques of section 2.3. To extend this single-server delay bound to an *end-to-end* delay bound, we will restrict the class of service disciplines in the network to rate-controlled service disciplines such as rate controlled static priority, earliest deadline first, or hierarchical round robin [23]. This restriction will allow us to have tighter end-to-end delay bounds than if we consider, for example, the class of all work-conserving service disciplines as in [4]. The reason is that if traffic streams are not rate-controlled inside the network, their properties become more and more bursty with each hop traversed, and allocation of resources in this situation is extremely wasteful.

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. Rate-control decouples the network nodes so that the end-to-end delay bound D is bounded by the summation of the delay bounds (d_i or $\hat{d}_i$) at individual nodes [23]. For a source that performs smoothing, the end-to-end delay over H hops is bounded by:

$$\hat{D} = \tau_r + \sum_{i=1}^{H} \hat{d}_i + \sum_{i=1}^{H} \pi_i \qquad (18)$$

where π_i is the propagation delay of the ith hop and τ_r is the smoothing delay bound of equation (17).

4.2. Reducing queueing delay

With the queueing delay bounds of section 2.3, we can now state formally the intuitive notion that a less bursty source can better utilize network resources.

Lemma 1. *If a source is smoothed as in definition 3.1 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 jth 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 equation (13) which states that $N(d) = max\{n \mid \frac{1}{l} \max_{t \geq 0}\{\sum_{j=1}^{n} b_j(t) - lt\} \leq d\}$. Once again, if If $b_j(t)$ is reduced for any connection j, 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.

4.3. Effect on end-to-end delay

Thus, smoothing *reduces* the queueing delay bound, but also introduces an additional delay τ_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 or obtaining an improved QoS, 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 a condition for determining whether smoothing is advantageous to a network client 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:*

$$\tau_r < \sum_{i=1}^{H} (d_i - \hat{d}_i). \tag{19}$$

Proof. The proposition states that if the bound on smoothing delay τ_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. □

Whether or not the inequality of proposition 1 is satisfied depends on factors such as the network load, the number of hops traversed, the burstiness of the stream, and the desired delay bound. For example, in theorem 1 of [10], it was shown that when streams traverse a single hop, smoothing never results in a net reduction in end-to-end delay bound. 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.

Equation (19) indicates that as the number of *congested* hops increases smoothing becomes more important. The reason 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 of [10], 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 [23]. The local delay bounds that constitute the sum in equation (19) may then be calculated by the admission control tests that correspond to the local service discipline (see [20] for several examples).

4.4. Obtaining the optimal rate

Here we develop proposition 1 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 signalling or connection-establishment protocol. The scheme requires additional network support during the establishment phase of a connection, but not during its data-delivery phase. We also note that network clients do not need to provide additional information to the network beyond their usual traffic specifications. The network informs a client about its best smoothing rate based on only the client's traffic specification and the load inside the network. Below, we consider only the D-BIND model, with the (PCR, SCR, MBS) model being a special case consisting of two rate–interval pairs.

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

$$\max_{R_1 \leq r \leq R_P} \{D - \hat{D}(r)\} \tag{20}$$

where D and $\hat{D}(r)$ are calculated as in section 4, and R_1 and R_P are as defined by the D-BIND model in section 2.1. If equation (20) 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 signalling 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 are available ahead of time, such as for stored video, then these parameters can be calculated directly from the trace. Otherwise the algorithms in [24] can be used to determine the appropriate parameters for the unknown source.

Next, this traffic specification is sent to the signalling protocol along with the source's maximum end-to-end delay requirement D and its request for *deterministic* service. For practical reasons, including the latency of the call setup and the length of the signalling message, only a small number of FIFO-smoothing rates can be tested, i.e. the maximization of equation (20) must be tested for a small number of rates r. This number, which we call Υ, will be fixed in practice by the signalling protocol. A larger Υ would increase the granularity of the tested rates to more closely obtain the optimal rate r^*; however, a larger Υ also requires more admission control tests and a larger signalling 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 $r_\upsilon = R_1 - \upsilon/\Upsilon(R_1 - R_P)$ will be tested for $\upsilon = 0, 1, 2, \ldots, \Upsilon$.

Along the forward path of the signalling 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, $\{d_i, \hat{d}_i^1, \hat{d}_i^2, \ldots, \hat{d}_i^\Upsilon\}$, to the local delay bounds calculated by the upstream nodes. The final or destination node then has $\Upsilon + 1$ bounds on end-to-end queueing delay. Using equation (17) this node also calculates the Υ 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

$$D_{min} = \min \left\{ \left(\sum_{i=1}^{H} d_i \right), \left(\tau_1 + \sum_{i=1}^{H} \hat{d}_i^1 \right), \left(\tau_2 + \sum_{i=1}^{H} \hat{d}_i^2 \right), \ldots, \left(\tau_\Upsilon + \sum_{i=1}^{H} \hat{d}_i^\Upsilon \right) \right\}. \qquad (21)$$

If the first term is the minimum then no smoothing is the best alternative. Otherwise a smoothing-FIFO with rate r_υ should be used, where r_υ is the smoothing rate that 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 signalling 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.

5. Experimental investigations

In this section we quantify the effects of smoothing on streams' end-to-end delay bounds and on the network's utilization with a set of experiments based on two 30 minute traces of MPEG compressed video. One trace is of an action movie (a 'James Bond' film) and the other one is of a newscast. Both were digitized to 384 by 288 pixels and compressed at 24 frames per second with frame pattern IBBPBBPBBPBB. The sequences, taken from [17], were compressed using constant-quality MPEG 1 compression performed with the Berkeley MPEG software tool [15]. Further details of the compression parameters and properties of the video streams can be found in [17]. We will refer to these streams as 'movie' and 'news' respectively.

This section describes experiments with connection admission control, investigating either how the set of admissible connections increases or decreases through various smoothing policies, or how the end-to-end delay bounds for a given set of connections is increased or decreased because of smoothing. Because the service is deterministic, we do not need to simulate the actual transfer of packets since we are assured by CAC tests that all packets will meet their respective delay bound. However, we do report the average utilization of the network which represents the fraction of time that a link is carrying deterministically-guaranteed traffic. This can be calculated as

$$\mu = \frac{\sum_{j \in \mathcal{A}} \psi_j}{l} \qquad (22)$$

where $\mathcal{A}$ is the set of admissible connections, ψ_j is the long term average rate of connection j, and l is the link speed. For a trace, ψ_j is given by the total number of bits transmitted over the entire trace divided by the duration of the trace.

5.1. Network topology

In the experiments we consider the network topology as depicted in figure 6. The traffic shapers are located at the network edge, where they smooth the original traffic streams to the desired shape (as in figure 4). The circles represent ATM switches at the respective OC-3 (155 Mbps) and OC-12 (622 Mbps) link speeds. The switches are assumed to schedule cells according to static priority, where the priority level is determined at connection set-up time. Streams with a deterministic QoS, as we are considering in this paper, are at the highest priority level. Below that are streams with statistical and best-effort services. Within both the deterministic and statistically guaranteed priority levels may be sub-levels for various delay bounds. Lastly, we assume that the streams are 're-shaped' with

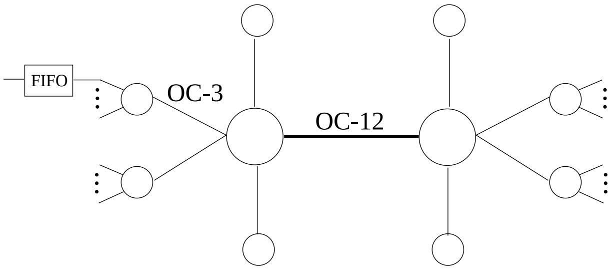


Figure 6. Network topology.

rate controllers inside the network so that the traffic stream does not become burstier inside the network. The scheduler we are considering in these experiments is therefore that of [21].

Note that cross-traffic from streams with statistical or best-effort service (such as file transfers or electronic mail) will not affect the performance obtained by the streams receiving a deterministic service, since these latter streams have priority over the former ones in the SP scheduler. Hence, the experiments below report results only for the streams obtaining deterministic service. Moreover, we emphasize that in the experiments, '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: statistical and best-effort traffic can always utilize remaining network resources.

Our experiments focus on three factors that influence the effectiveness of smoothing: the number of hops traversed by the stream H, the rate of the smoothing-FIFO r, and the effect of the traffic model, D-BIND or (PCR, SCR, MBS).

Our primary performance indices for measuring the effectiveness of smoothing are the average utilization of the network and the net *savings* in end-to-end delay bound due to smoothing, i.e. the difference of the total delay bound without and with smoothing, $D - \hat{D}$.

5.2. Number of hops

In the following experiments we compare the effectiveness of smoothing when streams traverse a single hop compared with three hops. For the single-hop case, a collection of streams are smoothed at the network edge and then traverse a single OC-3 link. In the three-hop case, the streams are again smoothed but this time traverse two OC-3 links and an OC-12 link as depicted in figure 6. Both homogeneous and heterogeneous traffic mixes are considered and the results are reported below.

5.2.1. Homogeneous connections. Figure 7 shows utilization versus the stream's total delay bound as in equation (18). Figure 7(a) depicts the case of a stream traversing one hop and figure 7(b) depicts the case of three hops. The utilization is as calculated by equation (22) and the results are based on the properties of the movie's video trace.

Figure 7(a) confirms the result of theorem 1 of [10], which states that smoothing never results in an increased network utilization for a given delay bound when a collection of streams traverses a single hop. For example, for an unsmoothed source and a delay bound of 80 ms, 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%. Similar results are obtained for different FIFO rates; indeed, no smoothing at all is the best alternative in this one-hop case.

Figure 7(b) depicts the case where the movie video streams traverse three hops rather than one. In this case, smoothing results in a substantial benefit. For example, without smoothing, an average utilization of 17% is achieved when the total end-to-end delay bound is 100 ms. By utilizing smoothing via a FIFO of rate 2.1 Mbps, this utilization is improved to 29%. Thus, in this case, smoothing of the video streams has resulted in a 70% increase in the number of admissible connections for the same end-to-end delay bound of 100 ms. 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.

5.2.2. Heterogeneous connections. Experiments with mixes of movie and news connections are depicted in figures 8(a) and 8(b). In these experiments the maximum end-to-end delay bound is fixed to 100 ms for figure 8(a) and 150 ms for figure 8(b). The figures show the maximum number of admissible movie connections on the vertical axis and news connections on the horizontal axis, both with and without smoothing.

Figure 8(a) shows a one-hop scenario where a collection of streams traverse a single OC-3 link. The solid line depicts the admissibility region without smoothing: 347 movie connections and no news connections can be admitted with $D = 100$ ms, as can 0 movie connections and 329 news. The curve depicts all of the admissible combinations for the given QoS constraint, such as 64 movie and 271 news connections. As the figure indicates, the addition of smoothing has only decreased the admissible region. For example, with 0 movie connections, the number of admissible news connections is reduced to 313 from 329. In this particular experiment the movie trace is smoothed with a FIFO rate of 2.7 Mbps and the news trace with a 2.1 Mbps FIFO rate. Different FIFO rates will shift the smoothing curve slightly but will not result in a greater admissible region for these streams.

Figure 8(b) shows the admissible region for a heterogeneous mix of traffic traversing *three* hops, with both OC-3 and OC-12 links, as in figure 6. This experiment also had FIFO rates of 2.7 and 2.1 Mbps for the movie and news traces respectively. As was the case in figure 7(b), the three-hop scenario results in a substantial benefit from smoothing. In figure 8 this benefit is expressed in terms of a larger admissible region for the same 150 ms end-to-end delay bound. For example, without smoothing if 100

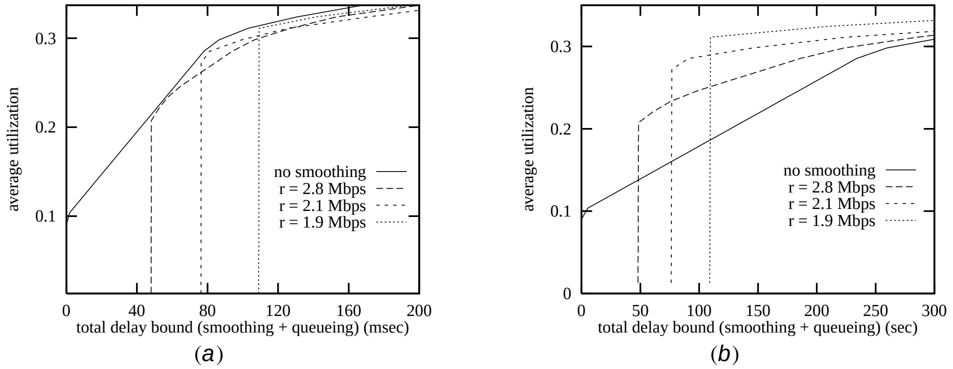


Figure 7. Average utilization plotted against total delay bound for various smoothing rates for (*a*) the one-hop and (*b*) the three-hops case.

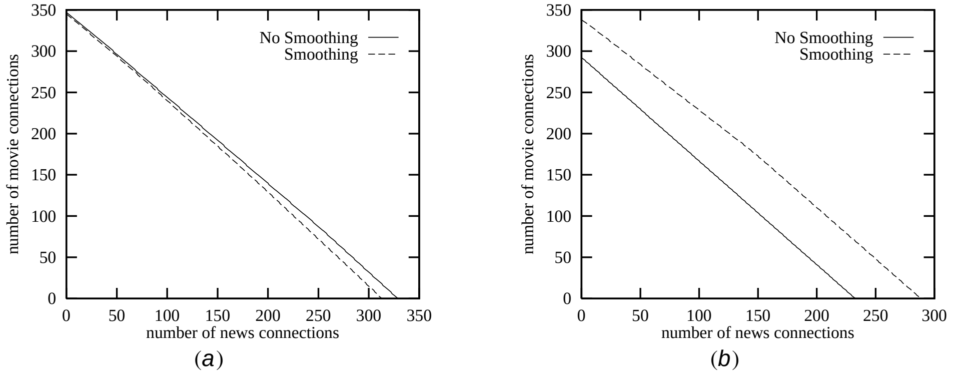


Figure 8. Number of admissible connections for heterogeneous traffic mixes for (*a*) the one-hop and (*b*) the three-hops case.

news connections are admitted, no more than 167 movie connections can be admitted so that all connections obtain an end-to-end delay bound of 150 ms. With the addition of smoothing, an additional 73 movie connections can be admitted—an increase of 50%.

5.3. Rate of the smoothing-FIFO

The implementation of the smoothing scheme proposed in section 4.4 shows that of the Υ smoothing rates that will be tested, the algorithm culminating in equation (21) returns the smoothing rate that achieves the maximum benefit from smoothing. Here, we quantify this benefit by considering the effect of the rate of the smoothing-FIFO r on the total end-to-end delay bound. Each different FIFO rate r transforms the stream's original rate–interval pairs (R_k, I_k) into smoothed pairs $(\hat{R}_k, \hat{I}_k)$, which in turn results in a different end-to-end delay bound.

The results of the experiment are shown in figure 9. In this experiment the network utilization is fixed to 33% and collections of movie streams are considered. The rate of the FIFO, r, is varied up to the stream's peak rate, and is reported on the horizontal axis. The vertical axis shows the resulting end-to-end delay bound.

Since this stream's peak rate is 5.9 Mbps, the right-most points of the curves depict the case of no smoothing, since a stream can traverse such a FIFO nearly unaffected. As the FIFO-rate is *decreased* the stream becomes increasingly smooth. Although the stream's average rate is 583 kbps, smoothing rates below 1 Mbps result in prohibitively high smoothing delays, and these cases are not reported.

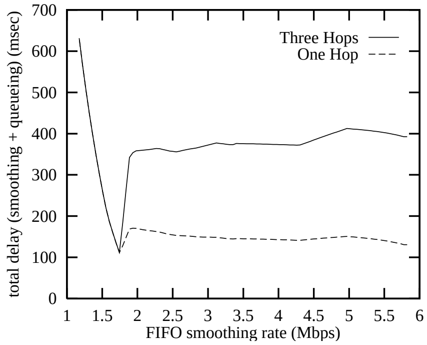
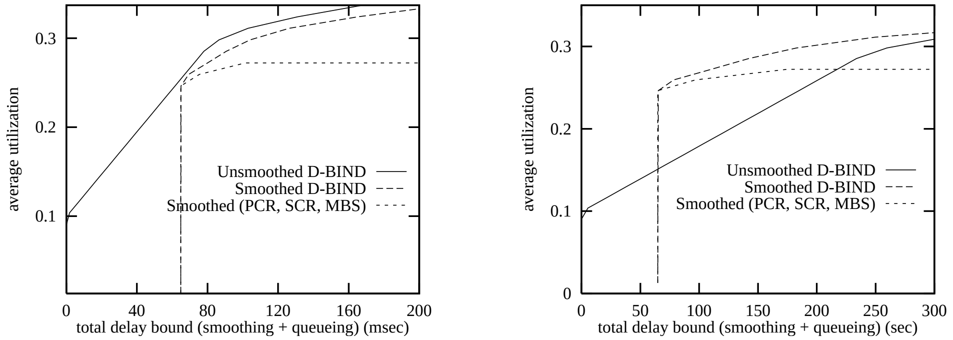


Figure 9. Effect of smoothing rate on total end-to-end delay.

The most striking result of the figure is that there is only a narrow range of FIFO rates in which smoothing is beneficial: indeed smoothing with an improperly chosen FIFO rate can be far worse than not smoothing at all. However, when the best FIFO rate is chosen the benefits can be substantial. For example, if a 200 ms end-to-end delay bound is required by 85 action movie streams traversing three congested hops (85 streams corresponds to a 33% utilization), the figure shows that the streams must be smoothed with a FIFO rate between 1.6 and 1.8 Mbps. Smoothing rates outside this range would result in



(*a*) (*b*)

Figure 10. Average utilization plotted against total delay bound for D-BIND and (PCR, SCR, MBS) models for (*a*) the one-hop and (*b*) the three-hops case.

a rejection of some of the connections by the admission control algorithm, indicating that the requested 200 ms delay bound cannot be guaranteed to 85 streams, for the 'improperly' shaped sources. The figure also illustrates the potential QoS benefits achievable from smoothing. For example, compared to an unsmoothed stream, smoothing to the best FIFO rate results in a 71% reduction in end-to-end delay bound, from 392 ms to 112 ms.

Thus, this experiment indicates the need for a smoothing scheme as in section 4.4 to allow a network client to properly choose its smoothing rate.

5.4. Traffic model

As described in section 2.3, peak-rate resource allocation is not required to provide a deterministic QoS. Indeed, [11] showed that the achievable utilization improvement above a peak-rate resource allocation scheme is highly influenced by the choice of the parameterized traffic model that network clients use to characterize their traffic to the network.

The traffic model also impacts the extent to which *smoothing* traffic streams can improve their QoS. Since both lemma 1 and proposition 1 are valid for any deterministic traffic model, smoothing as in definition 1 can still provide net benefits to the network clients, even with traffic models other than D-BIND.

Figure 10 shows network utilization versus the streams' total end-to-end delay bound for both the D-BIND and (PCR, SCR, MBS) traffic models. Figure 10(*a*) shows the case of a single hop, and figure 10(*b*) plots the case of three hops. The movie trace is used in both figures.

Figure 10(*a*) reconfirms theorem 1 of [10] for the (PCR, SCR, MBS) model. In this one-hop case, the incurred smoothing delay always outweighs the savings in queueing delay obtained inside the network. The figure also shows that when smoothed with the same FIFO rate of 2.2 Mbps, the D-BIND model provides a higher network utilization than the (PCR, SCR, MBS) model. The reason for this is that the D-BIND model better captures the burstiness properties of typical VBR streams. For example, with the (PCR, SCR, MBS) model and a delay bound of 100 ms, a 26% utilization is achievable. With the D-BIND model, the achievable utilization is 30%.

In the case of multiple congested hops, figure 10(*b*) reconfirms proposition 1 for the (PCR, SCR, MBS) model. The figure shows the effect of the traffic model on achievable utilization for streams traversing *three* hops, again with a FIFO smoothing rate of $r = 2.2$ Mbps. Over multiple hops, the delay-bound savings inside the network becomes significantly greater than the smoothing delay introduced at the network edge, so that both the (PCR, SCR, MBS) and D-BIND models are able to provide a net benefit for smoothing. However, as was the case with a single network hop, the D-BIND model's increased accuracy allows it to achieve higher utilization than the (PCR, SCR, MBS) model. For example, for a delay bound of 150 ms, a 26% utilization is achieved with the (PCR, SCR, MBS) model. With the D-BIND model and for the same delay bound, the network utilization is 29%. Without smoothing, the maximum utilization is limited to 21%.

Thus, the bounded-delay traffic smoothing that we have considered in this paper is effective with different deterministic traffic models, and the accuracy of the traffic model influences the extent to which smoothing is beneficial.

6. Conclusions

Network clients that have demanding QoS requirements on the network will need a deterministic service that provides an *a priori* guarantee that no packets will be dropped and that no packets will violate their delay bounds.

However, the *burstiness* of typical VBR streams makes it difficult to provide good QoS to the applications, while simultaneously making efficient use of network resources. *Smoothing* traffic streams can alleviate this problem at its origin by reducing the burstiness of the streams. However, smoothing also has consequences in that it necessarily reduces the QoS of a stream, either by reducing the amount of information transmitted or by adding delays to the stream's bursts.

In this paper we have investigated the *net* effects of smoothing, including both the gains obtained inside the network as well as the costs incurred at the network edge. We investigated these effects on an *end-to-end* basis considering bursty and heterogeneous traffic mixes, using both analytical techniques and experimentation with several long traces of MPEG-compressed video.

Our results identify the major factors involved in determining the scenarios in which smoothing is effective, and our experiments quantify the potential benefits of smoothing. For example, over a single network hop, we showed that the incurred smoothing delays necessarily outweigh any reductions in queueing delay, so that smoothing cannot provide a net benefit to the network client. Alternatively, over multiple hops, we showed that smoothing can indeed result in substantial net benefits to clients; the primary reason is that while the smoothing delay is incurred only once at the entrance of the network, queueing delays may be incurred at each congested hop.

We also showed that the effectiveness of smoothing is quite sensitive to the smoothing rate, or the manner in which the traffic is shaped. Hence, we provided guidelines on how this rate can be best selected so that network clients obtain the greatest benefit.

Lastly, we showed how the parameterized traffic model that network clients use to describe their traffic impacts the QoS that they can obtain from the network. We showed that use of the D-BIND traffic model [11] results in higher network utilization since it characterizes streams more accurately than a traffic model based on peak-rate, average rate, and burst length such as [5, 6]. This accuracy in the traffic characterization translates to more effective smoothing policies.

References

[1] Banerjea A, Ferrari D, Mah B, Moran M, Verma D and Zhang H 1996 Tenet real-time prococol suite: design, implementation and experiences *IEEE/ACM Trans. Networking* **4** 1–10

[2] Chang C 1994 Stability, queue length and delay of deterministic and stochastic queueing networks *IEEE Trans. Automatic Control* **39** 913–31

[3] Cruz R 1991 A calculus for network delay, part I: network elements in isolation *IEEE Trans. Informat. Theor.* **37** 114–21

[4] Cruz R 1991 A calculus for network delay, part II: network analysis *IEEE Trans. Informat. Theor.* **37** 121–41

[5] Ferrari D and Verma D 1990 A scheme for real-time channel establishment in wide-area networks *IEEE J. Selected Areas Commun.* **8** 368–79

[6] ATM Forum 1994 ATM user-network interface specification, version 3.1 *ATM Forum document*

[7] Gusella R 1991 Characterizing the variability of arrival processes with indices of dispersion *IEEE J. Selected Areas Commun.* **9** 203–11

[8] Kanakia H, Mishra P and Reibman A 1993 An adaptive congestion control scheme for real-time packet video transport *Proc. ACM SIGCOMM'94 (San Francisco, CA, 1993)* pp 20–31

[9] Knightly E 1996 H-BIND: a new approach to providing statistical performance guarantees to VBR traffic *Proc. IEEE INFOCOM'96 (San Francisco, CA, 1996)* pp 1091–9

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

[11] Knightly E and Zhang H 1995 Traffic characterization and switch utilization using deterministic bounding interval dependent traffic models *Proc. IEEE INFOCOM'95 (Boston, MA, 1995)* pp 1137–45

[12] Liebeherr J, Wrege D and Ferrari D 1996 Exact admission control for networks with bounded delay services *IEEE/ACM Trans. Networking* **4** (6) 385–401

[13] Lindley D 1952 On the theory of queues with a single server *Proc. Camb. Phil. Soc.* **48** 277–89

[14] Low S and Varaiya P 1993 Burstiness bounds for some burst reducing servers *Proc. IEEE INFOCOM'93 (San Francisco, CA, 1993)* pp 2–9

[15] Patel K, Smith B and Rowe L 1993 Performance of a software MPEG video decoder *Proc. 1st ACM Int. Conf. on Multimedia (Anaheim, CA, 1993)* pp 75–82

[16] Richardson I and Riley M 1995 Usage parameter control cell loss effects on MPEG video *Proc. ICC'95 (Seattle, WA, 1995)* pp 970–4

[17] Rose O 1995 Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems *Technical Report 101* Institute of Computer Science, University of Wurzburg

[18] Salehi J, Zhang Z, Kurose J and Towsley D 1996 Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing *Proc. ACM SIGMETRICS'96 (Philadelphia, PA, 1996)*

[19] Shroff N 1996 Improved loss calculations at an ATM multiplexer *Proc. IEEE INFOCOM'96 (San Francisco, CA, 1996)* pp 561–8

[20] Wrege D, Knightly E, Zhang H and Liebeherr J 1996 Deterministic delay bounds for VBR video in packet-switching networks: fundamental limits and practical tradeoffs *IEEE/ACM Trans. Networking* **4** 352–62

[21] Zhang H and Ferrari D 1993 Rate-controlled static priority queueing *Proc. IEEE INFOCOM'93 (San Francisco, CA, 1993)* pp 227–36

[22] Zhang H and Ferrari D 1994 Improving utilization for deterministic service in multimedia communication *Proc. Int. Conf. on Multimedia Computing and Systems (Boston, MA, 1994)* pp 295–304

[23] Zhang H and Ferrari D 1994 Rate-controlled service disciplines *J. High Speed Networks* **3** 389–412

[24] Zhang H and Knightly E 1995 A new approach to support VBR video in packet-switching networks *Proc. IEEE Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV'95) (Durham, NH, 1995)* pp 275–86