# Optimal Smoothing for Guaranteed Service

Jean-Yves Le Boudec and Olivier Verscheure

EPFL- DSC Technical Report DSC2000/014

2000 March 16

*This is the extended version, with proofs, of a paper published in ACM/IEEE Transactions on Networking under the same title.*

**Abstract**

We consider the transmission of *variable bit rate* (VBR) video over a network offering a guaranteed service such as ATM VBR or the guaranteed service of the IETF. The guaranteed service requires that the flow accepted by the network has to be conforming with a traffic envelope $\sigma$; in return, it receives a service guarantee expressed by a network service curve $\beta$. Functions $\sigma$ and $\beta$ are derived from the parameters used for setting up the reservation, for example, from the T-SPEC and R-SPEC fields used with the Resource Reservation Protocol (RSVP). In order to satisfy the traffic envelope constraint, the output of the encoder is fed to a smoother, possibly with some look-ahead. The resulting stream is transported by the network; at the destination, the decoder waits for an initial *playback delay* and reads the stream from the receive buffer. We consider the problem of whether there exists one optimal strategy at the smoother which minimizes the playback delay and the receive buffer size, given the traffic envelope $\sigma$ and the service curve $\beta$. We show that there does exist such an optimal smoothing, and give an explicit representation for it. We also obtain a simple expression for the smallest playback delay and playback buffer size which can be achieved over all possible smoothing and playback strategies. We show that the computation of optimal smoothing and minimum playback delay do not depend on the past. We show that separate delay equalization is optimal in the CBR case, but not otherwise. We also apply the theory to the analysis of which T-SPEC should be requested by a source-destination pair, given some playback delay and buffer constraint, and given the path characteristics advertised in RSVP PATH messages.

## I. Introduction

We consider the transmission of *variable bit rate* (VBR) video over a network offering a guaranteed service such as ATM VBR or the guaranteed service of the IETF [1], [2]. The guaranteed service requires that the flow produced by the output device conform with a traffic envelope $\sigma$, namely over any window of size $t$, the amount of data does not exceed $\sigma(t)$. With the Resource Reservation Protocol (RSVP), $\sigma$ is derived from the T-SPEC field in messages used for setting up the reservation, and is given by $\sigma(t) = \min(M + pt, rt + b)$, where $M$ is the maximum packet size, $P$ the peak rate, $r$ the sustainable rate and $b$ the burst tolerance [3]. The function $\sigma$ is also called an arrival curve.

In our framework, the video source must thus produce an output conforming with the arrival curve constraint. One approach for achieving this is called *rate control* [4], [5], [6]. It consists in modifying the encoder output, by acting on the quantization parameters. Rate control is a delicate issue in video coding since it significantly affects the video quality. An alternative approach is to smooth the video stream, using a smoother fed by the encoder [7], [8], [9]. In this paper we focus on the latter scenario.

A number of results of results exist on smoothing. In [8], smoothing is studied from the viewpoint of reducing the required network resources, with the assumption that connections are of the renegotiated CBR type. Optimality is sought in the sense of reducing the variability of the connection rate. In [10], [11] the authors go one step further and address, among others, the issue of minimizing playback delay and buffer, for the case of a CBR connection. They also study the cascaded scenario where playback and smoothing is performed at multiple points, typically as would occur with internetworking. Our results differ from these in two directions. Firstly, we are interested only in the end-system viewpoint, assuming that the sole information obtained by a source is what is available by signalling or by a protocol such as RSVP. Secondly, we focus on VBR rather than CBR or renegotiated CBR. Moving from CBR to VBR requires some sophistication in the method, which we try to use parsimonously. In [10], the authors find a representation of the latest optimal smoother output in the particular case of a CBR traffic envelope and a null network. As discussed in Section II-C, we find a generalization of this result to the VBR case; we also give a simple, physical interpretation of this result in terms of time inversion.

J.-Y. Le Boudec is with the Department for Communication Systems (DSC), EPFL, Switzerland.
O. Verscheure is with the IBM T.J. Watson Research Center, New York, USA.

One smoothing strategy is called *shaping* (it is called "optimal shaping" in [12]). It consists in putting the encoded flow $R(t)$ into a buffer, and outputting bits as soon as doing so does not violate the arrival curve constraint. It is shown in [12] that an optimal shaper minimizes the buffer requirement and the delay experienced in the smoother. However, a shaper is optimal only at the sender side. In this paper we consider another problem, namely, we would like to minimize the playback delay $D$ and the buffer size at the receiver. Another difference with shaping is that we allow our smoothing strategy to look-ahead, which a shaper does not.

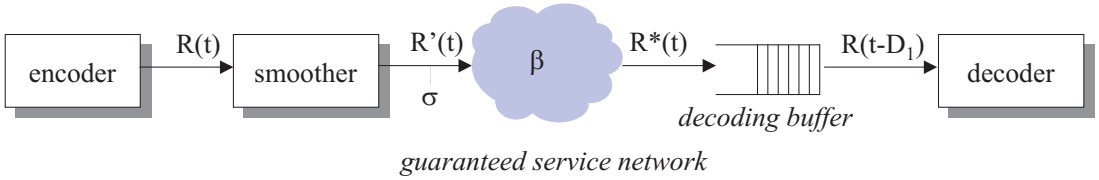


Fig. 1. Scenario and notation used in this paper.

Our scenario is illustrated on Figure 1. A multimedia stream is encoded, and then input into a smoother. The smoother writes the stream into a network for transmission. We call $R(t)$ the total number of bits observed on the encoded flow, starting from time $t = 0$, and $R'(t)$ the output of the smoother. Figure 4 shows an example of such a function, for an MPEG-2 video sequence. The smoother output must satisfy the traffic envelope constraint given by some function $\sigma$ negotiated with the network, which can be expressed as $R'(t+u) - R'(t) \leq \sigma(u)$ for all $u \geq 0$. At the destination, the receiver stores incoming bits into a decoding buffer before passing them to the decoder. The decoder starts reading from the decoding buffer after a delay $D$, and then reads the decoding buffer so as to reproduce the original signal, shifted in time. Thus the output of the decoding buffer is equal to $R(t - D_1)$, where $D_1$ is equal to $D$ plus the transfer time for the first packet of the flow. The delay $D$ is called *playback delay* at the receiver.

We are interested in scheduling strategies *at the smoother* which minimize the playback delay and the required decoding buffer size *at the receiver*. We allow the smoother to perform some look-ahead (also called pre-fetching), namely, we do not require that $R'(t) \leq R(t)$. Look-ahead is commonly used with pre-recorded streams, for which the smoother is composed of both a disk server and a scheduler.

We assume that the network offers to the flow $R'$ a guaranteed service, such as defined for example by the IETF. Call $R^*(t)$ the cumulative function at the output of the network. The transformation $R' \to R^*$ can be decomposed into a fixed delay, and a variable delay. Without loss of generality, we can reduce to the case where the fixed delay is zero, since it does not impact the smoothing method. The variable delay is due to queuing in, for example, guaranteed rate schedulers. The relationship between $R'$ and $R^*$ cannot be known exactly by the sending side, because it depends to some extend on traffic conditions; however, the guarantee provided by the network can be formalized by a condition of the form [13], [14], [12], [15]

$$\text{for all } t \geq 0, \text{ there exists some } s \leq t \text{ such that } R^*(t) \geq R'(s) + \beta(t-s) \tag{1}$$

In the condition, $\beta$ is a function, called the network service curve, which is negotiated during the reservation setup phase. For example, the Internet guaranteed service assumes the form $\beta(t) = \rho(t-L)^+$ where $L$ is called the latency and $\rho$ the rate. It is further assumed that the latency parameter $L$ depends on the rate $\rho$ according to $L = \frac{C_0}{\rho} + D_0$ for some constants $C_0$ and $D_0$. With RSVP, the values of $C_0$ and $D_0$ are contained in the AD-SPEC fields [15], [16]. We consider smoothing strategies that ignore the details of the network, but do know the service curve $\beta$.

Our main result can be summarized as follows. Firstly, there exists a minimal playback delay $\bar{D}$. It is equal to

$$\bar{D} = \inf\{t \geq 0 \text{ such that } \quad \text{for all } u \geq 0, v \geq 0 : R(u+v-t) \leq \sigma(u) + \beta(v)\}$$

We also give in the paper a simple formula to compute $\bar{D}$ in practical cases. Secondly, there exists one smoother output $\overline{R'}$ which is optimal in the following sense. Consider some other smoothing strategy, using a playback delay $D$, and with resulting function $R'$. Since $\bar{D}$ is the minimum playback delay, we must have $D \geq \bar{D}$. Then, necessarily, $R'(t) \geq \overline{R'}(t - (D - \bar{D}))$. In other words, if we time-shift the optimal solution $\overline{R'}$ so that the first

packet for this solution is played back at the same time as the first packet for the other solution $R'$, then $\overline{R'}$ is, at every time instant, no earlier than $R'$. The optimum $\overline{R'}(t)$ thus gives the latest time at which *every* packet of the flow should be scheduled. As a consequence, we show that the size of buffer required at the decoder with solution $\overline{R'}$ is also minimum. The optimal output $\overline{R'}$ is given by

$$\overline{R'}(t) = \sup_{u \geq 0, v \geq 0} \{R(t+u+v-\bar{D}) - \sigma(u) - \beta(v)\}$$

Our result shows that there is no smoothing strategy which can do better than the bounds, and the bounds can be attained. Now the optimal solution which attains the bounds requires the knowledge of the entire encoded sequence $R(t)$, which for very long sequences is not practical. However, this can be used as a benchmark for evaluating practical scheduling strategies.

Our study is restricted to the guaranteed service; we do not consider other frameworks, such as the best effort of the differentiated service of the IETF, where multiple video streams would share the same resources without individual guarantees.

The paper is organized as follows. Section II derives the main results. Section III gives applications to some practical cases. We first show that the computation of optimal smoothing and minimum playback delay do not depend on the past. Second, we show that the minimum required buffer size at the decoder depends only on the minimum traffic envelope of the original signal, whereas the minimum playback delay depends on the complete signal. Then we compare the theoretical optimal found in Section II to another strategy based on delay equalization. We show that in the constant bit rate (CBR) case, the latter is able to attain the optimal playback delay in the constant bit rate case; in contrast, in the variable bit rate case, this is generally not true. Lastly we consider the problem of which T-SPEC should be requested by a source-destination pair, given the playback delay and buffer constraints, and given the path characteristics advertised in RSVP PATH messages. This is different from the analysis of feasible arrival curves [17] in that we consider the allocation of the arrival curve on a given Intserv path, for which the path characteristics are known. We think that this is a real problem with which a source is confronted when using the guaranteed service.

Appendix A gathers the proofs for the results in Sections II and III. Appendix B shows how the optimal smoother output corresponds in the time inverted domain to the output of an optimal shaper. The appendices are based on what is called "Network Calculus", which is mainly an application of min-plus algebra. The interested reader will find there some original contribution to the "filtering theory" developed in [12], in particular, the use of min-plus deconvolution as a smoothing operator, and a representation of deconvolution with time inversion.

## II. Optimal Smoothing

### *A. A formal definition of the admissible smoother output*

Consider again the model illustrated in Figure 1. Assume first that we fix the value of the playback delay $D$. The job of the smoother is to produce an output whose cumulative function is $R'$. We take as time origin the beginning of the operation of the smoother, thus we must have

$$R'(t) = 0 \text{ if } t \leq 0 \tag{2}$$

We assume that $R'$ is constrained by the traffic envelope $\sigma$, namely

$$R'(t) - R'(s) \leq \sigma(t-s) \text{ for all } s \leq t \tag{3}$$

We also assume that the network offers a service curve $\beta$ to the flow, namely, Equation (1) is satisfied. It is more convenient to re-write Equation (1) as follows

$$R^*(t) \geq \inf_{0 \leq s \leq t} \{R'(s) + \beta(t-s)\} \tag{4}$$

As a convenient notation, the right-handside in the above equation is also traditionally written as $(R' \otimes \beta)(t)$, and is called the "min-plus" convolution of functions $R'$ and $\beta$ [12], [18], [19], [14]. This gives the equivalent writing for Equation (4):

$$R^*(t) \geq (R' \otimes \beta)(t) \tag{5}$$

The system must also satisfy the real-time constraint at the decoding buffer. This is expressed by

$$R^*(t) \geq R(t - D_0 - D) \tag{6}$$

where $D$ is the playback delay and $D_0$ the transfer time for the first packet of the flow. Now we assume that the smoother cannot know the individual packet delays, but only the network service curve $\beta$. Thus, $R'$ must be such that Equation (6) is true for *any* realization $R^*$ satisfying Equation (4). Now remember that we have reduced our study to the case where the fixed part of the transfer delay is 0. Consider a particular realization $R^*$ such that the first packet has a zero transfer delay, and for the rest (namely $t \geq t_1 =$ the arrival time of the second packet) satisfies the worst case $R^*(t) = (R' \otimes \beta)(t)$. We must thus have, for all $t > 0$:

$$(R' \otimes \beta)(t) \geq R(t - D) \tag{7}$$

Conversely, if this equation holds, then clearly $R^*(t) \geq R(t - D) \geq R(t - D_0 - D)$ and thus the real time condition is satisfied.

In summary, the constraints for the smoother is to produce an output $R'$ which satisfies simultaneously Equations (2), (3) and (7).

### *B. Minimal Playback Delay*

The first result in this paper is the following theorem.

*Theorem II.1* (Minimum Playback Delay) There exists one minimum value of the playback delay $D$ for which the smoother equations (2), (3) and (7) have a solution. It is given by

$$\bar{D} = \inf\{t \geq 0 \text{ such that } \quad \text{for all } u \geq 0, v \geq 0 : R(u + v - t) \leq \sigma(u) + \beta(v)\}$$

The proof of the theorem is given in Appendix A. We give a numerical example later in this Section (Figure 4). We now discuss the content and the implications of the theorem.

The theorem gives the smallest value of the playback delay that can be obtained by any smoothing strategy satisfying the arrival curve constraint $\sigma$, given that the network service curve guaranteed to the flow is $\beta$. The minimum delay $\bar{D}$ can be better interpreted using the concept of horizontal deviation, which we now recall. Figure 2 gives an intuitive definition.

*Definition II.1* (Horizontal Deviation $h$ [15]) For two functions $\alpha$ and $\beta$, define the horizontal deviation $h(\alpha, \beta)$ by

$$h(\alpha, \beta) = \sup_{s \geq 0} \left(\inf \{T : T \geq 0 \text{ and } \alpha(s) \leq \beta(s + T)\}\right) \tag{8}$$

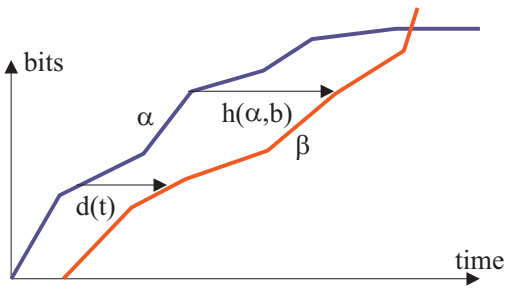


Fig. 2. Definition of horizontal deviation for two functions $\alpha$ and $\beta$. Determine $d(t)$ for all $t$ by drawing the horizontal distance from $\alpha$ to $\beta$. The horizontal deviation $h(\alpha, \beta)$ is the maximum of all $d(t)$.

It is shown in Appendix A that the value of the minimum playback delay $\bar{D}$ in the theorem is given by

$$\bar{D} = h(R, \sigma \otimes \beta) \tag{9}$$

In the formula, $\sigma \otimes \beta$ is the min-plus convolution defined as in the discussion following Equation (4), and which can be interpreted as follows [12], [15], [14]. Consider for a second a hypothetical shaper, as defined in the

Introduction, with traffic envelope $\sigma$. Assume that $\sigma$ is a "good" function, namely sub-additive, as explained for example in [12]. The arrival curves used with RSVP or for ATM VBR connections are good functions. We know from [12], [15], [14] that, if the input flow to the shaper is $S(t)$, and if the shaper is large enough to avoid losing data, then the output is equal to $(\sigma \otimes S)(t)$. Thus we can interpret $\sigma \otimes \beta$ as follows. Imagine a flow with cumulative function $S(t) = \beta(t)$; put this imaginary flow into a shaper in order to make it conform to the traffic envelope $\sigma$. The resulting, shaped flow is $\sigma \otimes \beta$. Then the minimum playback delay achievable with a look-ahead smoother is the horizontal deviation between the original signal $R(t)$ and the curve $(\sigma \otimes \beta)(t)$. Figure 3 illustrates this interpretation.

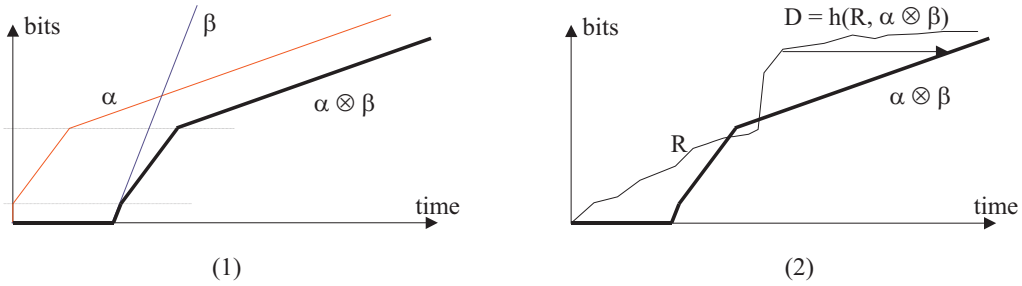


Fig. 3. Computation of minimum playback delay $D$. (1) compute $\sigma \otimes \beta$, the service curve shaped by the arrival curve; (2) $D$ is the horizontal deviation from the original signal $R(t)$ to the curve $(\sigma \otimes \beta)(t)$.

B.0.a Numerical Example. We now illustrate the result on a numerical example. We consider a video sequence encoded with MPEG-2, transported over UDP and IP using the real time transport protocol (RTP) [20]. Our example is a 400 frame-long sequence conforming to the ITU-R 601 format (720*576, 25 frames per second). The sequence is composed of 3 video scenes that differ in terms of spatial and temporal complexities. It has been encoded in an open-loop VBR mode, as interlaced video, with a structure of 11 images between each pair of I-pictures and 2 B-pictures between every reference picture. For this purpose, the widely accepted TM5 video encoder [21] has been utilized. According to the MPEG-2 standard, a TS packet is a 188-byte length packet, which encapsulates both video and system information. We consider, as is common place, that two transport stream packets are palced in one RTP packet. Since the size of the MPEG-2 transport stream packet is 188 bytes and the overhead of RTP is 40 bytes, the packets sent throughout the IP network contain $2*188+40 = 416$ bytes. Figure 4 shows the trace we use. We apply Theorem II.1 with the following parameters. The arrival

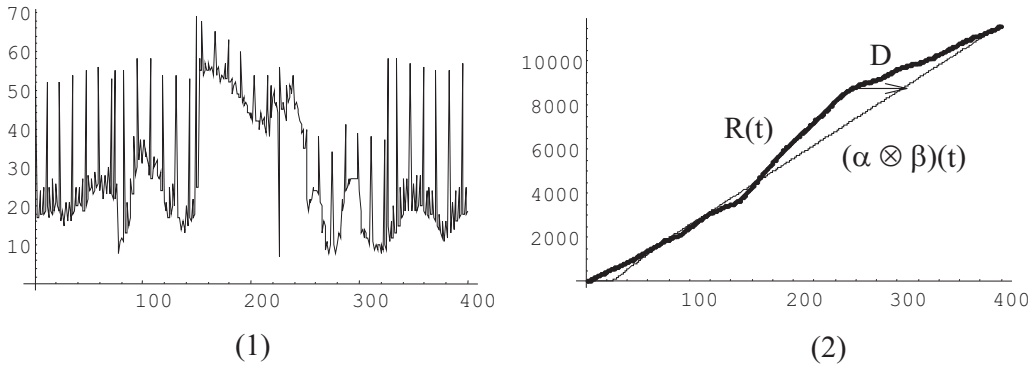


Fig. 4. The MPEG-2 trace used as illustration: (1) number of RTP packets per frame, or, equivalently, per 40 ms timeslot; (2) the cumulative function $R(t)$ counted in packets per timeslot (thick line), as well as the min-plus convolution $\sigma \otimes \beta$ of arrival and service curves (thin line). The minimum playback delay ($\bar{D} = 2.05$ seconds) is indicated by the arrow.

curve has the form $\sigma(t) = \min(M + pt, rt + b)$ given in the introduction. As usual, $M$ is the maximum packet size, thus is equal to 1 packet. The peak and sustainable rates are, respectively, the peak and the average rates of the MPEG-2 stream ($P = 4.38$ Mbits/s and $r = 2.7$ Mbits/s). The burst tolerance $b = 332$ packets corresponds to roughly 1 Mbits. The service curve is as with the Internet guaranteed service, with a latency

$L = 1$s and a rate $\rho =$ equals to, respectively, $1s$ (25 frames) and 3 Mbits/s (slightly more than the average bit rate but less than the peak rate).

In the case where the arrival curve $\sigma$ and the service curve $\beta$ have the standard form used with the Internet integrated services, the computation of $D$ can be simplified as follows.

*Proposition II.1* (Computation of $\bar{D}$ in practice) Assume that the arrival curve $\sigma$ has the form $\sigma(t) = \min(M+pt, rt+b)$, and the service curve $\beta$ has the form $\beta(t) = \rho(t-L)^+$. For a given signal $R(t)$, the minimum playback delay $\bar{D} = h(R, \alpha \otimes \beta)$ is also given by

$$\bar{D} = \sup_{t \geq 0} \{F(R(t)) - t\}$$

with

$$F(k) = L + \max\left(\frac{k-M}{p}, \frac{k-b}{r}, \frac{k}{\rho}\right)$$

The proof is given in Appendix A. This shows that the complexity of computing $\bar{D}$ is $O(n)$, where $n$ is the number of samples in the trace $R(t)$.

### C. Optimal Smoother Output

So far we have given a result for the minimum playback delay. We now show a more global result, namely, there exists one smoother output which is better than any other output, at any time instant, in a sense which we define now.

*Definition II.2* (Time shifted optimal output $R^-$) For a given signal $R(t)$, define $R^-(t)$ for all $t \in \mathbb{R}$ by

$$R^-(t) = \sup_{u \geq 0, v \geq 0} \{R(t+u+v) - \sigma(u) - \beta(v)\}$$

Note that, unlike $R$, the function $R^-$ is non-zero even for some negative times. After appropriate time-shifting, $R^-$ is the optimal smoother output, as the following theorem shows.

*Theorem II.2:* 1. The minimal delay defined in Theorem II.1 is the smallest $t$ such that $R^-(-t) \leq 0$

2. For any admissible smoother output $R'$, with playback delay $D$, we have, for all $t \geq 0$:

$$R'(t) \geq R^-(t - D)$$

The proof is given in Appendix A. We can interpret the theorem as follows. The first item relates the minimal delay $\bar{D}$ to the optimal output. It says that $\bar{D}$ is the smallest time shift which is necessary to make the flow described by $R^-$ start at time 0. Second, note that, since $\bar{D}$ is the minimum playback delay, we must have $D \geq \bar{D}$. Now call $\overline{R'}(t) = R^-(t - \bar{D})$ the optimal output, namely the shifted version of $R^-$ that starts at time 0. Then the theorem means that if we time-shift $\overline{R'}$ so that the first packet for this solution is played back at the same time as the first packet for some other solution $R'$, then $\overline{R'}$ is, at every time instant, no earlier than $R'$. The shifted optimal output $\overline{R'}(t - (D - \bar{D})) = R^-(t - D)$ thus gives the latest time at which *every* packet of the flow should be scheduled. Figure 5 illustrates this.

C.0.b Representation of Optimal Smoother Output with Time Inversion . The shifted optimal output $R^-$ can be computed using its definition; however, we can reduce its complexity with a time inversion transformation. At this point we need to introduce a classical min-plus construct, called min-plus deconvolution, noted $\oslash$, and defined [22], [23] by:

$$(f \oslash g)(t) = \sup_{u \in \mathbb{R}} \{f(t+u) - g(u)\} \tag{10}$$

Note that $f \oslash g$ may be non-zero for negative times even if this is not the case for $f$ and $g$. With this notation, the function $R^-(t)$ can be written in a more compact way as $R^- = R \oslash (\sigma \otimes \beta)$.

It is shown in Theorem B.1 in Appendix that min-plus deconvolution can be computed easily by means of time inversion. Thus, $R^-$ can be computed as follows. First invert time; then compute, in the inverted time domain, the min-plus convolution of the resulting function on one hand, of $\sigma \otimes \beta$ on the other hand; lastly, invert time again and obtain $R^-$. Figure 6 illustrates this representation on a very simplified scenario. The signal $R(t)$ consists of one large burst of $B$ bits at time $\theta$, and the network offers a constant delay (null network case; thus we drop $\beta$ in the rest of this example). This scenario is extreme, but it represents an interesting

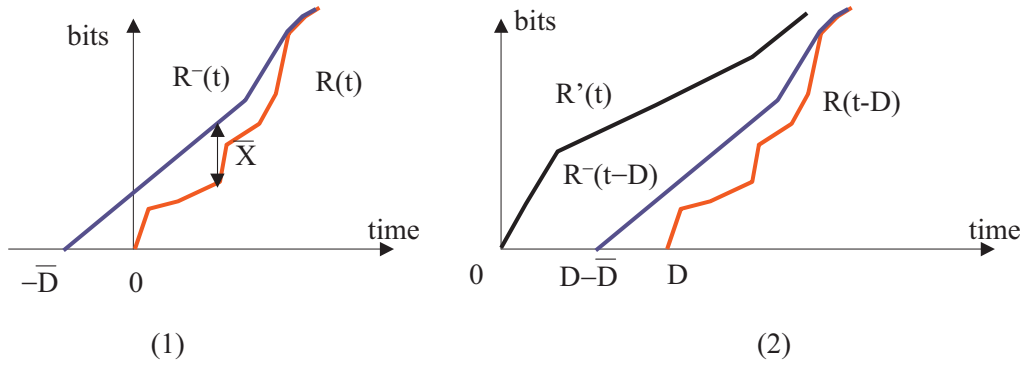


Fig. 5. Optimal smoothing: (1) computation of $R^-(t)$ from the encoded signal $R(t)$. The minimum playback delay $\bar{D}$ is the point where $R^-(-t)$ hits 0. (2) For any admissible smoother output $R'(t)$ with playback delay $D$, the shifted version $R^-(t-D)$ is no earlier that $R'$.

limiting case. The figure shows the shifted optimal smoother output $R^- = R \oslash \sigma$, assuming the arrival curve $\sigma$ has the standard form $\sigma(t) = \min(M + pt, rt + b)$.

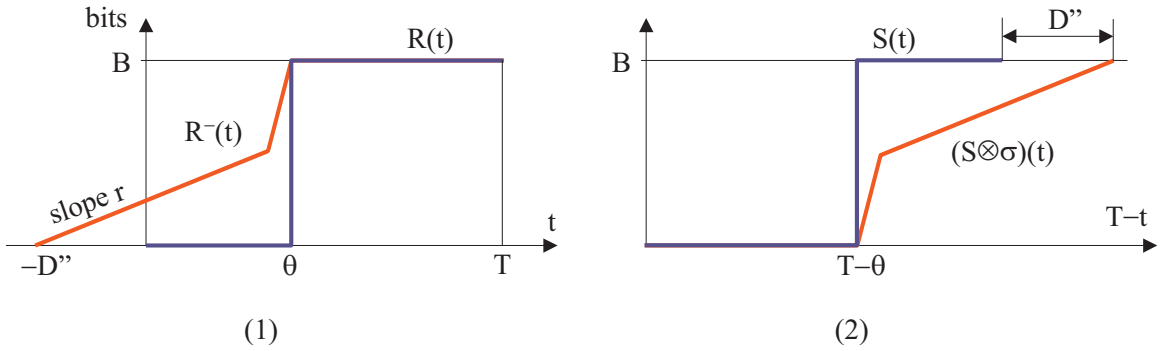


Fig. 6. (1) A bursty scenario for delay equalization, showing $R(t)$ and the shifted optimal smoother output $R^-$ for this scenario. (2) $R^-(t)$ is obtained by reverting time. Define $S(t) = R(T) - R(T-t)$; the curve for $S$ is obtained from $R$ by a rotation of $180^o$ around the center $(\frac{T}{2}, \frac{R(T)}{2})$. Obtain $S \otimes \sigma$ by shaping $S$ according to the arrival curve $\sigma(t) = \min(M + pt, rt + b)$. Then $R^-(t) = (S \otimes \sigma)(T) - (S \otimes \sigma)(T-t)$ is obtained by reverting time again.

In [10], the authors find a representation of the optimal smoother output in the particular case of a CBR traffic envelope and a null network. Their representation can be easily interpreted as the time inverted signal, shaped to a constant bit rate. Thus, their representation is a particular case of our result.

C.0.c Required Buffer at the Decoder: . Consider now the buffer size that must be provisioned at the decoder. Remember that we can remove any fixed delay. Thus, for a given scheduler output $R'(t)$, all we can know about the decoder input decoder $R^*$ is that $R(t-D) \leq R^*(t) \leq R'(t)$. The decoder buffer content at some time $t$ is $R^*(t) - R(t-D)$. Thus the buffer size that must be provisioned is $\sup_{t \geq 0}\{R'(t) - R(t-D)\}$. A simple examination of Figure 5 shows the following corollary.

*Corollary II.1:* The buffer size that need to be provisioned at the decoder is minimum for solution $\overline{R'}(t) = R^-(t - \bar{D})$. It is equal to

$$\bar{X} = \sup_{t \geq 0}\{R^-(t) - R(t)\} = \sup_{t \geq 0, u \geq 0, v \geq 0}\{R(t+u+v) - R(t) - \sigma(u) - \beta(v)\}$$

We show in Appendix A that the formula for $\bar{X}$ can be interpreted in terms of network calculus abstractions, which leads to the following simplification.

*Proposition II.2* (Computation of $\bar{X}$ in practice) Assume that the arrival curve $\sigma$ has the form $\sigma(t) = \min(M + pt, rt + b)$, and the service curve $\beta$ has the form $\beta(t) = \rho(t-L)^+$. For a given signal $R(t)$, the minimum buffer

that needs to be provisioned at the decoder, $\bar{X}$ is also given by

$$\bar{X} = \sup_{t \geq 0} \left\{ A(t) - \min \left[ (p(t-L)+M)^+, (r(t-L)+b)^+, (\rho(t-L))^+ \right] \right\}$$

where $A(t)$ is the empirical envelope for $R$, defined by:

$$A(t) = \sup_{u \geq 0} \{R(t+u) - R(u)\}$$

The complexity of computing $\bar{X}$ with this method is $O(n^2)$, where $n$ is the number of samples in the trace $R(t)$. In appendix A-C we give an alternative method using the time inversion representation, which has a complexity of $O(n)$. It is the same representation as in [10], Section IV.A., for the particular case of a null network and a CBR traffic envelope.

### *D. Null network case*

Consider the case where the network service provides a constant transfer delay. This occurs for example with a circuit switched service, or, as an approximation, with ATM constant bit rate (CBR) services if the delay variation if very small. In our framework, a constant delay network is equivalent to a null network.

The null network case is a straightforward application of the general case, by letting $\beta(t) = +\infty$ for all $t \geq 0$. Equivalently, simply remove $\beta$ from all formulas: for example, the minimum playback delay becomes

$$\bar{D} = h(R, \sigma) = \inf\{t \geq 0 \text{ such that } \quad \text{for all } u \geq 0 : R(u-t) \leq \sigma(u)\}$$

For a circuit switched network service, $\sigma$ is given by $\sigma(t) = ct$, where $c$ is the bit rate of the circuit or the peak rate of the CBR connection. Thus, applying Proposition II.1, we obtain the minimum playback delay for a flow $R(t)$ transmitted over a circuit with rate $c$:

$$\bar{D}_{CBR} = \sup_{t \geq 0} \{\frac{R(t)}{c} - t\} = -\frac{1}{c} \check{R}(c)$$

where $\check{R}(x) = \inf_{s \geq 0}\{xs - R(s)\}$ is the concave conjugate of $R$. Figure 7 shows an example.

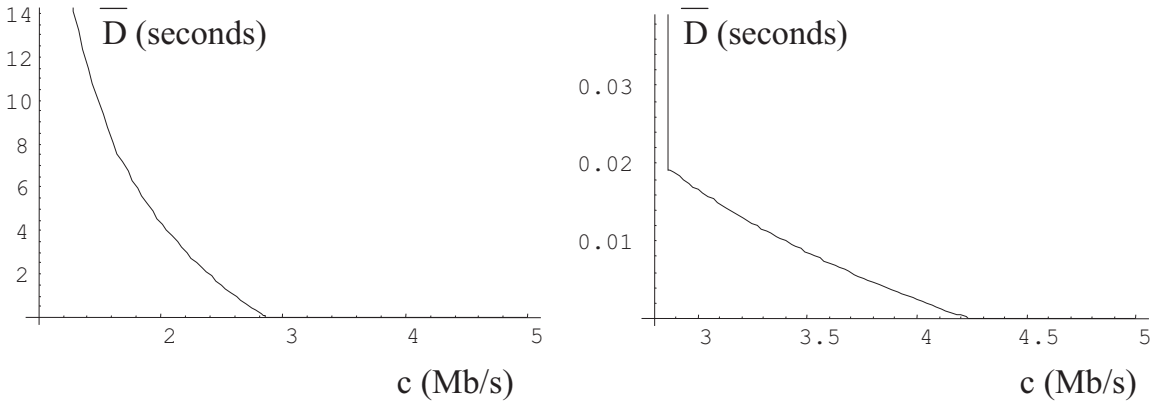


Fig. 7. Minimum playback delay $\bar{D}$ for the MPEG trace of Figure 4, when it is transported over a circuit or CBR connection with rate $c$. The figure shows the delay as a function of $c$, over two time scales. The delay is 0 for when $c$ is larger than the peak rate of the scene.

## III. Applications

### *A. Optimal Smoothing versus Optimal Shaping*

The previous section has shown that there is one optimal scheduling which minimizes the decoder buffer and playback delay. In this subsection we give some insight into the optimal smoother output that leads to this

solution. To that end, we restrict our discussion to the null network case, and compare the optimal smoother output to another scenario called shaping [12].

Optimal shaping is the standard method used to make an arbitrary flow conform to some traffic envelope $\sigma$. A shaper, with shaping curve $\sigma$, is a system which takes a flow as input, possibly keeps the bits in a buffer, and outputs the bits in such a way that the output conforms to the traffic envelope $\sigma$. An optimal shaper is one which sends the bits as early as possible. A well known example of optimal shaper is the leaky bucket controller. For an optimal shaper with input function $R$, the output $R'$ is given by $R'(t) = (R \otimes \sigma)(t)$. The formula is true under the assumption that $\sigma$ is sub-additive (namely $\sigma(s+t) \leq \sigma(s) + \sigma(t)$) and $\sigma(t) = 0$ for $t \leq 0$. It is known that these technical conditions on $\sigma$ are not a restriction, since any arrival curve can be replaced by one which satisfies them. The arrival curves defined for Internet integrated services or for ATM and mentioned above do satisfy these assumptions, as do any concave arrival curves [15]. It is known that an optimal shaper minimizes buffer and delay on the shaper side.

Back to our original problem, consider the optimal smoother output in the null network case. More precisely, let us focus on the time shifted function $R^-(t)$ given in Definition II.2. Using min-plus deconvolution recalled in Equation (10), we can write $R^- = R \oslash \sigma$. We call *optimal smoothing* the transformation $R \mapsto R \oslash \sigma$. There is some similarity with the transformation associated with an optimal shaper. Indeed, for a shaper with service curve $\sigma$ (with $\sigma$ sub-additive and $\sigma(0) = 0$), the output is equal to $S \otimes \sigma$ [12], [15], if $S$ is the input. The transformation $S \to S \oslash \sigma$ is also a smoothing operation, and like the other one, it is idempotent, namely, $(S \oslash \sigma) \oslash \sigma = S \ominus \sigma$.

Since optimal smoothing minimizes buffer and delay requirements at the decoder side, we should expect in general that a smoother that would be implemented by shaping the encoded flow $R(t)$ (thus producing a function $R' = R \otimes \sigma$) will yield a larger playback delay and buffer requirement at the decoder. Figure 8 shows one example.

Note that a smoother that would be implemented as a shaper would first read the bits in its buffer in real time as they are produced by the encoder, before delivering them to the network. We say that optimal shaping is *causal*: the scheduling of packets requires only the knowledge of the present and the past, and is independent of the future. In contrast, the optimal smoother can look ahead, and this is what allows it to obtain a smaller playback delay; the optimal smoother output needs to know the future of the signal $R(t)$ in order to determine the optimal scheduling.

Now the representation of optimal smoothing with min-plus deconvolution gives us more insight. It is shown in Appendix B that min-plus deconvolution can be obtained by min-plus convolution, after time inversion. In other words, if we call $S(t) = R(T) - R(T-t)$, where $T$ is the end of the trace, then the optimal smoother output $R^- = R \oslash \sigma$ is equal to the time inverted version of $S \otimes \sigma$. Figure 6 illustrates that this graphically corresponds to a rotation of $180^o$ around the point $(\frac{T}{2}, \frac{R(T)}{2})$. Since $S \otimes \sigma$ can be interpreted as the result of optimal shaping applied to $S$ in the inverted time domain, it follows that optimal smoothing is *anti-causal*. This means that the computation of the optimal smoother output is *independent of the past and the present*, and depends only on the future of the signal. Thus, in some sense, minimizing the playback delay is based exclusively on the ability to look-ahead in the original encoded signal $R(t)$.

Another implication is the following. With an optimal shaper, the effect of a large burst at the beginning of a sequence tends to disappear with time. Thus, we have a converse result for optimal smoothing: the influence on the minimum playback delay of large bursts located at the end of a sequence tends to disappear if the sequence is long. Thus, a sub-optimal smoothing strategy based on limited look-ahead should be able to provide results close to the optimal. A detailed analysis of this statement is the object of future research.

### *B. Playback Delay versus Decoder Buffer*

Let us consider again the required buffer $\bar{X}$ defined in Corollary II.1. We can rewrite the Equation in the corollary as

$$\bar{X} = \sup_{t \geq 0} \{A(t) - [(\sigma \otimes \beta)(t)]\}$$

where $A(t)$ is, as defined in Proposition II.2, the empirical envelope for $R(t)$. Thus, the minimum required buffer depends only on the empirical envelope $A(t)$ of the original signal. This means that two sequences with the same envelope, but which distribute their bursts at different times, have the same minimum required buffer.

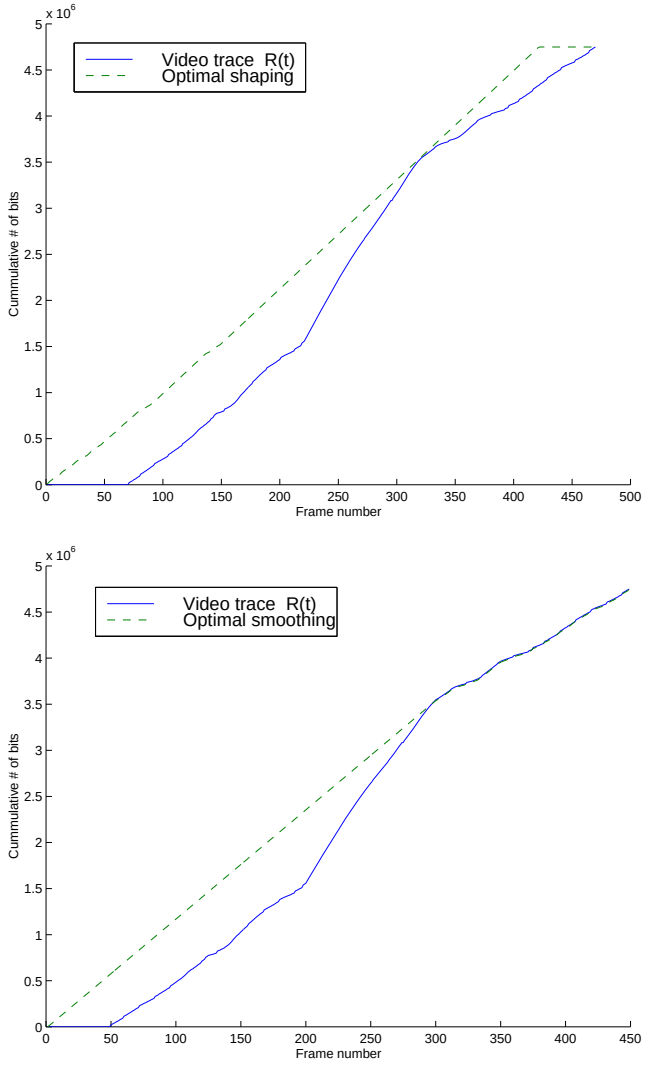


Fig. 8. Example of optimal shaping versus optimal smoothing for one MPEG trace. The example is for a network with constant delay, for a traffic envelope with $M$ =1 MPEG TS-packet, $p = 4.8$Mb/s, $b = 80$KB, $r = 2.4$Mb/s. The figure shows the optimal shaper [resp. smoother] output and the original signal (video trace), shifted by the required playback delay. The playback delay is 2.76s for optimal shaping (top) and $1.92s$s for optimal smoothing (bottom).

In contrast, the minimum playback delay, as given by Equation (9), does depend on the complete sequence, and not on the traffic envelope. Figure 9 shows two sequences with the same envelope, thus the same required buffer, but with different minimum playback delays.

### *C. Comparison with delay equalization*

A common method to implement a decoder is to first remove any delay jitter caused by the network, by delaying the arriving data in a delay equalization buffer; then we use a playback buffer to compensate for fluctuations due to pre-fetching. Figure III-C shows such a system. If the delay equalization buffer is properly configured, its combination with the guaranteed service network results into a fixed delay network, which, from the viewpoint in this paper, is equivalent to a null network. Compared to the original scenario in Figure 1, we have now separate buffers for delay equalization and for compensation of pre-fetching. We would like to understand the impact of this separation on the minimum play back delay. The delay equalization buffer operates by delaying the first bit of data by an initial delay $D'$, equal to the worst case delay though the network. Call $D$" the initial delay at the decoding buffer. The total playback delay for this scenario is $D' + D$".

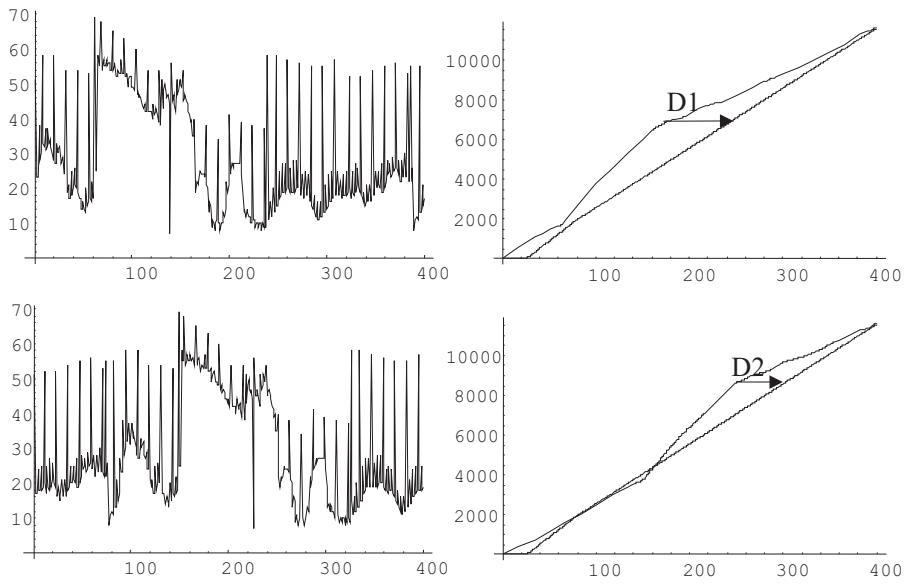


Fig. 9. The two traces have the same envelope, thus the same minimum buffer requirement (here, 928KB), however the second trace has its bursts later, thus, has a smaller minimum playback delay ($D_2 = 2.05$s versus $D_1 = 2.81$s). The example is for the same network parameters as Figure 4.

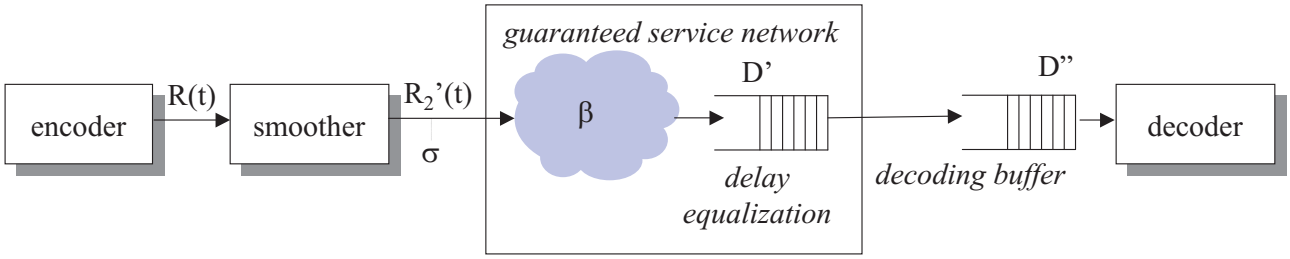


Fig. 10. Delay equalization at the receiver.

Of course, we must have $D' + D" \leq \bar{D}$, where $\bar{D}$ is the playback delay for the optimal smoother of the original scenario, since we have proven that $\bar{D}$ is the minimum playback delay that can ever be obtained. Thus, we should expect that, at least in general, separate delay equalization is not optimal. However, we can get some more insight, as follows.

Firstly, in order to understand where non-optimality might come form, consider again the simplified scenario illustrated on Figure 6 (in the rest of this discussion we call $R_2^-$ what is shown as $R^-$ on Figure 6). We simplify the rest of the discussion by considering the limiting case where $p = +\infty$ and $M = 0$. From Theorem II.2, the pure playback delay $D$" is the value shown on the figure and is equal to $D" = \frac{B-b}{r} - \theta$. The buffer equalization delay $D'$ is the worst case delay obtained when the input to the network is $R_2^-$; assume that the network service curve has the standard form $\beta(t) = \rho(t - L)^+$. It is equal to $D' = L + \frac{b}{R}$. The minimum playback delay $\bar{D}$ is obtained using the method presented in Figure 3; we have $\bar{D} = \frac{B-b}{r} - \theta$ and finally:

$$D' + D" = \bar{D} + \frac{b}{R} \tag{11}$$

Thus, with this scenario, separate delay equalization indeed gives a larger overall playback delay. A detailed examination of the formulas shows that if we combine delay equalization and compensation for pre-fetching in one single buffer, then, if the smoother output is optimal, the playback delay accounts for burstiness only once. This is another instance of the"pay bursts only once" phenomenon [3], [15].

Secondly, Equation (11) suggests a different outcome for the case $b = 0$, namely, the constant bit rate case.

We now consider that case in a general setting, namely the signal $R(t)$ has its general form, not just the special case mentioned previously. We assume thus that the arrival curve is of the form $\sigma(t) = \lambda_r(t) = rt$; this is the case for circuit switched services, for a guaranteed service flow with burstiness $b = 0$, or for an ATM constant bit rate connection. Assume as previously that the network service curve has the standard form $\beta(t) = \rho(t-L)^+$. For this case, the pure playback delay $D''$ is now the horizontal distance $D" = h(R, \lambda_r)$. The buffer equalization delay $D'$ satisfies $D' \leq L$; and finally the overall minimum playback delay $\bar{D}$ is horizontal distance $\bar{D} = h(R, \lambda_r \otimes \beta)$. If we assume that $\rho \geq r$, then it is simple to show that $(\lambda_r \otimes \beta)(t) = r(t-L)^+$ and thus $h(R, \lambda_r \otimes \beta) = L + h(R, \lambda_r)$. Thus finally $\bar{D} = D' + D"$, in other words, for the CBR case, separate delay equalization is able to attain the optimal playback delay.

### *D. Determination of optimal T-SPEC*

The Internet guaranteed service assumes that every node offers a service of the form $\beta(t) = \rho(t-L)^+$ for some latency $L$ and rate $\rho$, and further, that the latency parameter $L$ depends on the rate $\rho$ according to $L = \frac{C_0}{\rho} + D_0$. Using the IETF terminology, $\rho$ is contained in the list of R-SPEC parameters. The constants $C_0$ and $D_0$ depends on the route taken by the flow throughout the network. They are both determined during the advertisement phase (in the PATH messages, assuming routing does not change with the traffic parameters). The rate $\rho$, provided by the network, is not know a priori by a source, it is discovered during the advertisement phase using PATH messages, and accumulated in the AdSpec. With the guaranteed service, a source advertises an arrival curve $\sigma$ of the form $\sigma(t) = \min(M + Pt, b + rt)$, and destinations choose a target admissible network delay $T_0$. The choice of a specific service curve $\beta(t) = \rho(t-L)^+$ (or equivalently, of a rate parameter $\rho$) is done during the reservation phase and cannot be known exactly in advance.

We consider the following problem. Assume that an input flow and a fixed maximum playback delay $\Delta$ are given. Assume that source and destination are able to agree on what reservation should be done, by some out-of-band mechanism. The question is: which choices of $\sigma(t) = \min(M + Pt, b + rt)$ and of $T_0$ are admissible in order to guarantee that the reservation that will subsequently be performed ensures a playback delay not exceeding $\Delta$. Note that this problem is different from the problem of which arrival curve $\sigma(t) = \min(M + Pt, b + rt)$ is admissible [17], or of the tradeoff between burst tolerance and rate allocations. Indeed, in our case, we consider the allocation of the arrival curve on a given Intserv path, for which the path characteristics are known. We think that this is the real problem to which a source is confronted when using the guaranteed service.

The solution to this problem is detailed in Appendix A-D. The result is a procedure to test whether a choice of parameters $(\sigma, T_0)$ is compatible with the playback delay $D$, as follows.

D.0.d Procedure to test the acceptability of a traffic envelope $\sigma$ and target network delay $T_0$: . Given are a traffic envelope $\sigma$, a playback delay budget $\Delta$, a target network delay $T_0$ and path characteristics $C_0, D_0$. The algorithm is as follows.

- If $T_0 \geq \Delta$ or $D \leq D_0$ or $T_0 < D_0 - \frac{b-M}{p-r}$ then $(\sigma, T_0)$ is not admissible
- else compute $\rho_2$ as the only positive solution of $\overline{\overline{R}}(\rho_2) + \rho_2(\Delta - D_0) - C_0 = 0$, where $\overline{\overline{R}}(\rho) = \inf_u\{\rho u - R(u)\}$. If $r \geq \frac{b+C_0}{T_0 - D_0}$ then do the following. If $r \geq \rho_2$ then $(\sigma, T_0)$ is admissible else not.
- Else (namely if $r < \frac{b+C_0}{T_0-D_0}$), then compute $\rho_1 = \frac{rt_0+b+C_0}{t_0+T_0-D_0}$, where $t_0 = \frac{b-M}{p-r}$ and do the following. If both $\rho_1 \geq \rho_2$ and $\overline{\overline{R}}(r) + r(\Delta - D_0) + b - r\frac{C_0}{\max(\rho_1, r)} \geq 0$ then $(\sigma, T_0)$ is admissible, else not.

Figure 11 shows an example.

## IV. Conclusion

We have analyzed the scenario where a multimedia source uses the guaranteed service; the flow is assumed to receive a certain fixed network service curve, but has to comply with some traffic envelope. We also assume that the source has the ability to look ahead and deliver information in advance of the real time. We are interested at minimizing playback delay and required buffer at the decoder. In this context, we found that there exists one minimum playback delay, and obtain one scheduling strategy at the source which achieves this minimum. This strategy is also the one that sends data as late as possible. We have given explicit formulae to compute all elements of the strategy for practical cases. This result is of fundamental nature; it is explicit and easy to compute, however, it assumes a complete knowledge of the entire signal. Nonetheless, the existence of and the

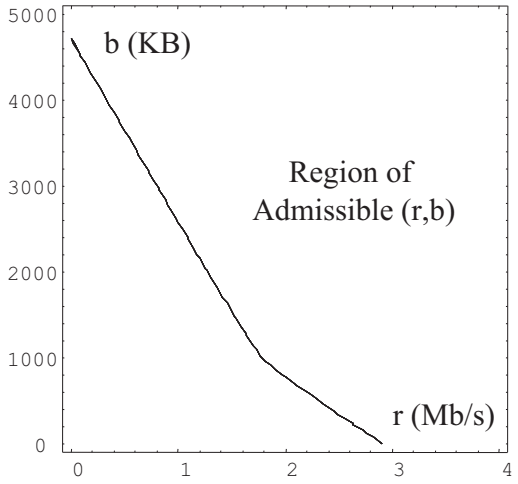


Fig. 11. Set of admissible $(r, b)$ for the MPEG-2 trace of Figure 4, assuming that the peak parameters $P$ =4Mb/s and $M$ = 1MPEG packet are fixed. The target playback delay is $\Delta = 2$s. Assume that the destination chooses (with RSVP) an admissible network delay of 1s and that the network path characteristics are $C_0 = 0.5$Mb/s and $D_0 = 0.2$s. Thus the problem remains to choose acceptable values of the sustainable rate $r$ and the burst tolerance $b$. The figure shows the set of values of $(r, b)$ which guarantee that the playback delay constraint is satisfied, given the path characteristics and the allocation of network delay.

expression for an explicit optimum is a fundamental result which can be used to analyze practical scheduling strategies.

This result also gives us insight into some system aspects. We have obtained the optimal scheduling strategy as the reverse time equivalent of optimal shaping. This leads us to the conjecture that scheduling strategies based on a limited amount of look-ahead should be close to optimal in practice. This also shows that the computation of optimal smoothing and minimum playback delay do not depend on the past. We have shown that the minimum required buffer size at the decoder depends only on the minimum traffic envelope of the original signal, whereas the minimum playback delay depends on the complete signal. We have found that separate delay equalization is optimal in the constant bit rate case, but not otherwise in the variable bit rate case. Lastly, we have applied the theory to the practical problem to which a source is confronted when using the Internet guaranteed service, namely, which T-SPEC should be requested by a source-destination pair, given some playback delay and buffer constraint, and given the path characteristics advertised in RSVP PATH messages.

From a methodological viewpoint, the derivations in the paper are based on min-plus algebra (a "network calculus" approach). We gave some original contribution to the "filtering theory" developed in [12], in particular, the use of min-plus deconvolution as a smoothing operator, and a representation of deconvolution with time inversion.

## Appendix

### I. Proofs

We use the min-plus convolution and deconvolution operations, noted $\otimes$ and $\oslash$, defined respectively in the text following Equation (4) and in Equation (10). We use in particular the following properties of min-plus deconvolution [22], [23], [19]. For any three functions of time $f$, $g$ and $h$:

1. $(f \oslash g) \leq h$ if and only if $f \leq (g \otimes h)$
2. $(f \oslash g)$ is the minimum solution to the problem $f \leq (g \otimes x)$, where $t \to x(t) \in \mathbb{R}$ is the unknown function.
3. $(f \oslash g) \oslash h = f \oslash (g \otimes h)$

Note that we allow negative times and that $f \oslash g$ maybe positive for some negative times even if $f$ and $g$ are zero for negative times.

### A. Proof of Theorems II.1 and II.2

Consider a solution $R'$ to the smoother equations (2), (3) and (7), for some value $D$ of the playback delay . Using the above properties, we can re-write Equations (3) and (7) as

$$R' \geq R' \oslash \sigma \tag{12}$$

$$R' \geq R_D \oslash \beta \tag{13}$$

where $R_D$ is defined by $R_D(t) = R(t-D)$. Substitute Equation (13) into Equation (12), this gives

$$R' \geq (R_D \oslash \beta) \oslash \sigma = R_D \oslash (\beta \otimes \sigma) = R_D \oslash (\sigma \otimes \beta)$$

Now the last term, applied at time $t$, is simply $R^-(t-D)$, where $R^- = R \oslash (\sigma \otimes \beta)$ as in Definition II.2. We have shown that, for any solution $R'$, we have $R'(t) \geq R^-(t-D)$. Now define $\bar{D}$ as in Theorem II.1. It is straightforward to show that $\bar{D}$ is the smallest $t$ such that $R^-(-t) \leq 0$. This proves Theorem II.2.

We must also have $D \geq \bar{D}$. Conversely, define $R'$ by $R' = R_{\bar{D}} \oslash (\sigma \otimes \beta)$. It can easily be seen that $R'$ is a solution with playback delay $\bar{D}$. This shows Theorem II.1.

### B. Proof of Equation (9) and Proposition II.1

From the definition of the horizontal deviation $h$, we have, for any number $t$:

$$t \geq h(R, f) \Leftrightarrow \text{ for all } s \geq 0 \; R(s) \leq f(s+t)$$

this in turn is equivalent to $(R \oslash f)(-t) \leq 0$, which proves Equation (9).

Now the horizontal deviation can be computed more easily with inverse functions. For some function $f$, define the pseudo-inverse $f^{-1}$ by [22]

$$f^{-1}(k) = \inf\{t \text{ such that } f(t) \geq k\}$$

Then it is simple to see that

$$h(f, g) = \sup_t \{g^{-1}(f(t)) - t\}$$

Proposition II.1 follows simply from computing $F = (\sigma \otimes \beta)^{-1}$.

### C. Proof and discussion of Proposition II.2

By re-arranging the sup in Corollary II.1, we find

$$\bar{X} = \sup_u \{\sup_t [R(t+u) - R(t)] - (\sigma \otimes \beta)(u)]\} = \sup_u \{A(u) - (\sigma \otimes \beta)(u)]\}$$

Proposition II.2 follows simply from computing $(\sigma \otimes \beta)(t)$.

We would like at this point to discuss the method given in Proposition II.2. There exists an alternative which generally requires fewer computation steps, but is more complex to express. We now outline this alternative method. First note that

$$\bar{X} = \sup_t \{R^-(t) - R(t)\}$$

Now $R^- = R \oslash (\sigma \otimes \beta)$, thus we can exploit the representation in Appendix B. Define $S(t) = R(n) - R(n-t)$, where $n$ is the size of the trace, and assuming time is discrete. Appendix B suggests computing $S \otimes \sigma \otimes \beta$, since $R^-$ is obtained from $S \otimes \sigma \otimes \beta$ by inverting time. With the assumptions in Proposition II.2, a simple computation gives

$$(S \otimes \sigma \otimes \beta)(t) = (S \otimes \sigma_0)(t - L)$$

where $\sigma_0(t) = \min\{\sigma(t), \rho t\}$. Thus $S \otimes \sigma_0$ is the result of shaping the inverted flow $S$ through a combination of three leaky-bucket controllers (two for $\sigma(t)$, one for $\rho t$). From the filtering theory in for example [12] we know that a leaky bucket controller can be implemented with a finite number of operations per time instant. Thus the computation of $(S \otimes \sigma_0)$ has a complexity $O(n + \bar{D})$, and thus so is the computation of $S$ and finally $R^-$. If we can assume (a safe bet) that there exists a constant $K$ such that $\bar{D} \leq Kn$ then the computation of $R^-$ is $O(n)$. Note that the required storage is also $O(n)$.

*D. Computation of Admissible T-SPECs*

For a given (but unknown) $(\sigma, T_0)$, the reservation rate is not known. Let us call $\mathcal{D}(\sigma, \mathcal{T}_\prime)$ the set of rates that may be allocated by the network. A rate $\rho$ is in $\mathcal{D}(\sigma, \mathcal{T}_\prime)$ if and only if

$$\begin{cases} h(\sigma, \beta(\rho)) & \leq T_0 \quad \text{(a)} \\ \rho & \geq r \quad \text{(b)} \end{cases} \tag{14}$$

where $\beta(\rho)(t) = \rho(t-L)^+$ and $\sigma(t) = \min(M+Pt, b+rt)$. Call $\rho_1(\sigma, T_0)$ the solution to Equation 14(a) and define $\rho_{min}(\sigma, T_0) = \max(\rho_1(\sigma, T_0), r)$. It is easy to see that $\mathcal{D}(\sigma, \mathcal{T}_\prime)$ is the interval $[\rho_{min}(\sigma, T_0), +\infty[$.

We now determine the solution $\rho_1$ to Equation 14(a). We easily derive $\rho_1 = \frac{rt_0+b}{t_0+T_0-L(\rho_1)}$ where $L(\rho_1)$ is the delay parameter contained in the R-SPEC and is equal to $\frac{C_0}{\rho_1} + D_0$. Finally, we obtain

$$\rho_1 = \frac{rt_0 + b + C_0}{t_0 + T_0 - D_0} \tag{15}$$

with $t_0 = \frac{b-M}{p-r}$.

Note that that all variables $(r, t_0, b, C_0, T_0, D_0)$ have non-negative values. Thus, Equation (15) has an admissible solution if and only if $T_0 \geq D_0 - \frac{b-M}{p-r}$. Moreover, the condition $\rho_1(\sigma, T_0) \geq r$ is equivalent to $r \leq \frac{b+C_0}{T_0-D_0}$.

Now we proceed with analyzing the conditions on $(\sigma, T_0)$. Every rate $\rho \in \mathcal{D}(\sigma, \mathcal{T}_\prime)$ corresponds to a rate that the network may potentially reserve (returned in its R-SPEC). Thus, we require that every rate $\rho \in \mathcal{D}(\sigma, \mathcal{T}_\prime)$ must necessarily verify the constraint on the playback delay: $h(R, \sigma \otimes \beta(\rho)) \leq \Delta$.

One must notice that $h(R, \sigma \otimes \beta(\rho))$ decreases when $\rho$ increases. Indeed, for a given time $t$, $(\sigma \otimes \beta(\rho))(t)$ increases with the rate $\rho$. Therefore, we can conclude that it is necessary and sufficient that

$$h(R, \sigma \otimes \beta(\rho_{min}(\sigma, T_0))) \leq \Delta \tag{16}$$

which may be rewritten as:

$$(\sigma \otimes \beta(\rho_{min}))(t) \geq R(t-\Delta) \quad \text{for} \quad i = \{1, 2\} \tag{17}$$

where $(\sigma \otimes \beta(\rho_{min}))(t)$ is the shifted version of $(\sigma \otimes \alpha)(t) = \min\{M+Pt, b+rt, \rho_{min}t\}$ by the amount of time $L(\rho_{min})$. Therefore, we obtain that the following must be true for all $t$:

$$\begin{cases} b + rt & \geq & R(t - \Delta + L(\rho_{min})) & \text{(a)} \\ \rho_{min}t & \geq & R(t - \Delta + L(\rho_{min})) & \text{(b)} \end{cases} \tag{18}$$

with $L(\rho_{min}) = \frac{C_0}{\rho_{min}} + D_0$. Now, first, we analyze Equation 18(b). Let $u = t - \Delta + \frac{C_0}{\rho_{min}} + D_0$; Equation 18(b) may be rewritten as:

$$\rho_{min}(\Delta - D_0) - C_0 \geq \sup_{u \geq 0}\{R(u) - \rho_{min}u\}$$

which is equivalent to

$$\rho_{min}(\Delta - D_0) - C_0 \geq -\inf_u\{\rho_{min}u - R(u)\}$$

Call $\overline{\overline{R}}$ the concave conjugate of $R$, namely $\overline{\overline{R}}(\rho_{min}) = \inf_u\{\rho_{min}u - R(u)\}$. Equation 18(b) is equivalent to

$$\overline{\overline{R}}(\rho_{min}) + \rho_{min}(\Delta - D_0) - C_0 \geq 0 \tag{19}$$

From the concavity of $\overline{\overline{R}}$ we can conclude that there exists one $\rho_2$ (independent of $(\sigma, T_0)$) such that Equation 18(b) is equivalent to

$$\rho_{min} \geq \rho_2 \tag{20}$$

The value of $\rho_2$ is the only positive solution of

$$\overline{\overline{R}}(\rho_2) + \rho_2(\Delta - D_0) - C_0 = 0 \tag{21}$$

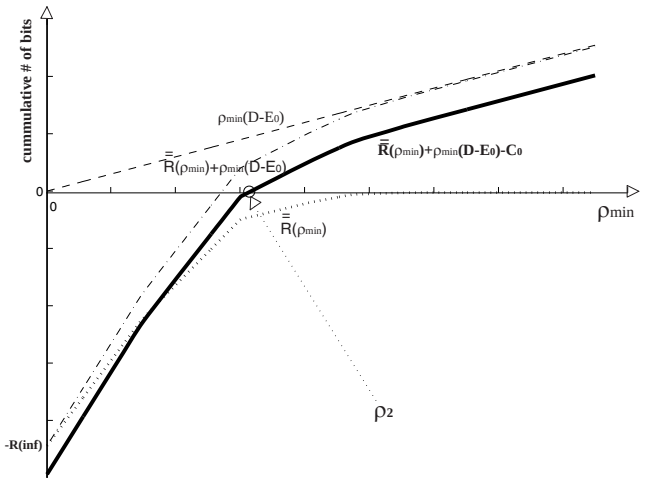


Fig. 12. Graphical solution to Equation 19, showing how to compute $\rho_2$.

The graphical solution to Equation 19 is represented on Figure 12. It illustrates that there exists a solution to Equation 19 if and only if the following conditions are met:

$$\begin{cases} \Delta \geq D_0 \quad \text{and,} \\ C_0 = 0 \quad \text{if} \quad \Delta = D_0 \end{cases}$$

Similarly, Equation 18(a) is equivalent to

$$\overline{\overline{R}}\,(r) + r(\Delta - D_0) + b - r\frac{C_0}{\rho_{min}} \geq 0 \tag{22}$$

It is not clear whether Equation (22) can be simplified. In summary so far, the conditions on $(\sigma, T_0)$ are that both Equations (20) and (22) are satisfied.

Now, if it turns out from the values of $(\sigma, T_0)$, that $\rho_1 \leq r$, then $\rho_{min} = r$ and Equation (22) is redundant. The only condition is thus $r \leq \rho_2$ in that case. This ends the proof of the algorithm in Section III-D.

## II. Representation of Min-Plus Deconvolution By Time Inversion

We show in this appendix how min-plus deconvolution can be represented in the time inverted domain by min-plus convolution. As a consequence, the optimal smoother output can be obtained by shaping the time inverted signal, then inverting time again.

Call $\mathcal{F}$ the set of wide-sense increasing functions of time with values in $[0, +\infty]$, which have a finite lifetime. More precisely, a function $t \to S(t)$ is in $\mathcal{F}$, if it is wide-sense increasing, if $S(t) \geq 0$, if there exist some finite $T_0$ and $T$ such that $S(t) = 0$ if $t \leq T_0$ and $S(t) = S(T)$ for $t \geq T$. It is traditional to consider $T_0 = 0$; in other words, to consider only non-negative times. However, in this paper, it is more convenient to allow some negative times. For a function $S$ in $\mathcal{F}$, we use the notation $S(+\infty)$ as a shorthand for $\sup_{t\in\mathbb{R}} S(t) = \lim_{t\to+\infty} S(t)$.

*Lemma B.1:* Let $f$ be some wide sense increasing function such that $f(t) = 0$ for $t \leq 0$ and $\lim_{t\to+\infty} f(t) = +\infty$. For any $S \in \mathcal{F}$, $S \otimes f$ is also in $\mathcal{S}$ and $(S \otimes f)(+\infty) = S(+\infty)$.

.0.e Proof: . Define $L = S(+\infty)$ and call $T$ a number such that $S(t) = L$ for $t \geq T$. Since $f(0) = 0$ implies that $S \otimes f \leq S$. Thus

$$(S \otimes f)(t) \leq L \text{ for } t \geq T \tag{23}$$

Now since $\lim_{t\to+\infty} f(t) = +\infty$, there exists some $T_1 > T$ such that $f(t) \geq L$ for all $t > T_1$. Now let $t > 2T_1$. If $u > T_1$, then $f(u) \geq L$. Otherwise, $u \leq T_1$ thus $t - u \geq t - T_1 > T_1$ thus $S(t - u) \geq L$. Thus in all cases $f(u) + S(t - u) \geq L$. Thus we have shown that

$$(S \otimes f)(t) \geq L \text{ for } t > 2T_1 \tag{24}$$

Combining (23) and (24) shows the lemma. ■

*Definition B.1* (Time Inversion) For a fixed $T \in [0, +\infty[$, the inversion operator $\Phi_T$ is defined on $\mathcal{F}$ by:

$$\Phi_T(S)(t) = S(+\infty) - S(T-t)$$

Graphically, time inversion can be obtained by a rotation of $180^o$ around the point $(\frac{T}{2}, \frac{S(+\infty)}{2})$. It is simple to check that $\Phi_T(S)$ is in $\mathcal{F}$, that time inversion is symmetrical ($\Phi_T(\Phi_T(S)) = S$) and preserves the total value ($\Phi_T(S)(+\infty) = S(+\infty)$). Lastly, for any $f$ and $T$, $S$ is $f$-smooth if and only if $\Phi_T(S)$ is $f$-smooth.

*Theorem B.1* (Representation of Deconvolution by Time Inversion) Let $S \in \mathcal{F}$ be a function with finite lifetime, and let $T$ be such that $S(T) = S(+\infty)$. Let $f$ be a wide-sense increasing function, with $f(t) = 0$ for $t \leq 0$ and $\lim_{t\to+\infty} f(t) = +\infty$. Then

$$S \oslash f = \Phi_T(\Phi_T(S) \otimes f) \tag{25}$$

The theorem says that $S \oslash f$ can be computed by first inverting time, then smoothing as with an optimal shaper, then inverting time again. Figure 6 shows a graphical illustration. The assumption that $\lim_{t\to+\infty} f(t) = +\infty$ means that the smoothing does not put a limit on the total number of bits that are output.

.0.f Proof: . The proof consists in computing the right handside in Equation (25). Call $\widehat{S} = \Phi_T(S)$. We have, by definition of the inversion

$$\Phi_T(\Phi_T(S) \otimes f) = \Phi_T(\widehat{S} \otimes f) = (\widehat{S} \otimes f)(+\infty) - (\widehat{S} \otimes f)(T-t)$$

Now from Lemma B.1 and the preservation of total value:

$$(\widehat{S} \otimes f)(+\infty) = \widehat{S}(+\infty) = S(+\infty)$$

Thus, the right-handside in Equation (25) is equal to

$$S(+\infty) - (\widehat{S} \otimes f)(T-t) = S(+\infty) - \inf_{u\geq 0}\{\widehat{S}(T-t-u) + f(u)\}$$

Again by definition of the inversion, it is equal to

$$S(+\infty) - \inf_{u\geq 0}\{S(+\infty) - S(t+u) + f(u)\} = \sup_{u\geq 0}\{S(t+u) - f(u)\}$$

■

## References


[1] T. V. Lakhsman, A. Ortega and A. R. Reibman, "VBR video: Trade-offs and potentials," *Proceedings of the IEEE*, July 1997.

[2] D. Reininger, et al., "Variable Bitrate MPEG video : Characteristics, Modeling and Multiplexing," *Proceedings of the ITC-14*, pp. 295–306, 1994.

[3] R. Guérin and V. Peris, "Quality-of-service in packet networks - basic mechanisms and directions," *Computer Networks and ISDN, Special issue on multimedia communications over packet-based networks*, 1998.

[4] M. Hamdi and J. W. Roberts, "QoS Guaranty for Shaped Bit Rate Video Connections in Broadband Networks," *IEEE Computer Society Press*, pp. 153–162, Sept. 1995.

[5] W. Ding and B. Liu, "Joint Encoder and Channel Rate Control of VBR Video over ATM Networks," *SPIE Electronic Imaging - Digital Video Compression*, vol. 2668, Jan. 1996.

[6] A. O. C.-Y. Hsu and A. R. Reibman, "Joint selection of source and channel rate for vbr video transmission under atm policing constraints," *IEEE Journal on Selected Areas in Communications*, 1997.

[7] W. Feng, F. Jahanian, and S. Sechrest, "Optimal Buffering for the Delivery of Compressed Prerecorded Video," *IASTED/ISMM Int'l Conference on Networks*, Jan. 1995.

[8] J. Salehi, Z. Zhang, J. Kurose and D. Towsley, "Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing," *ACM SIGMETRICS*, May 1996.

[9] J. McManus and K. Ross, "Video on demand over atm: Constant-rate transmission and transport," *IEEE INFOCOM*, March 1996.

[10] J. Rexford and D. Towsley, "Smoothing variable bit rate video in an internetwork," *IEEE/ACM transactions on networking*, vol. 23, no. 7, pp. 202–215, 1999.

[11] J. Zhang and J. Hui, "Multi-node buffering and traffic smoothing in vbr video transmission," Tech. Rep. 217, Rutgers University, 1997.

[12] C. Chang, "On deterministic traffic regulation and service guarantee: A systematic approach by filtering," *IEEE Transactions on Information Theory*, pp. 1096–1107, August 1998.

[13] R. L. Cruz, "Quality of service guarantees in virtual circuit switched networks," *IEEE Journal on Selected Areas in Communications*, pp. 1048–1056, August 1995.
[14] R. Agrawal and R. Rajan, "Performance bounds for guaranteed and adaptive services," Tech. Rep. 20649, IBM, 1996.
[15] J.-Y. L. Boudec, "Application of network calculus to guaranteed service networks," *IEEE Transactions on Information Theory*, pp. 1087–1096, May 1998.
[16] B. Braden, D. Clark and S. Shenker, "Integrated services in the internet architecture: an overview," rfc 1633, IETF, June 1994.
[17] S. H. Low and P. P. Varaiya, "A simple theory of traffic and resource allocation in atm," *GLOBECOM*, Dec. 1991.
[18] F. Baccelli, G. Cohen, G. J. Olsder and J.-P. Quadrat, *Synchronization and Linearity, An Algebra for Discrete Event Systems*. John Wiley and Sons, 1992.
[19] R. L. Cruz, "Sced+ : Efficient management of quality of service guarantees," *IEEE INFOCOM*, March 1998.
[20] V. G. D. Hoffman, G. Fernando and R. Civanlar, "Rtp payload format for mpeg1/mpeg2 video," rfc 2250, IETF, Jan. 1998.
[21] C. Fogg, "mpeg2encode/mpeg2decode," tech. rep., MPEG Software Simulation Group, 1996.
[22] R. Agrawal, R. L. Cruz, C. Okino and R. Rajan, "A framework for adpative service guarantees," *Conference on Communications, Control and Computers*, Sept. 1998.
[23] J.-Y. L. Boudec, "Network calculus made easy," Tech. Rep. 218, EPFL-DI, 1996.