

Optimal Smoothing for Guaranteed Service

Jean-Yves Le Boudec, *Member, IEEE*, and Olivier Verscheure, *Member, IEEE*

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 σ ; in return, it receives a service guarantee expressed by a network service curve β . Functions σ and β 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 σ and the service curve β . 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 constant bit rate (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.

Index Terms—Network calculus, playback delay, video transmission.

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]. The guaranteed service requires that the flow produced by the output device conform with a traffic envelope σ , namely over any window of size t , the amount of data does not exceed $\sigma(t)$. With the resource reservation protocol (RSVP), σ 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 [2]. The function σ is also called an arrival curve.

In our framework, the video source must thus produce an output conforming with the arrival curve constraint. One ap-

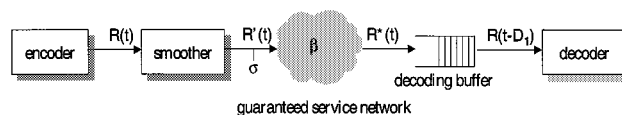


Fig. 1. Scenario and notation used in this paper.

proach for achieving this is called *rate control*. 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 [3]. In this paper we focus on the latter scenario.

A number of results exist on smoothing. In [3], smoothing is studied from the viewpoint of reducing the required network resources, with the assumption that connections are of the renegotiated constant bit rate (CBR) type. Optimality is sought in the sense of reducing the variability of the connection rate. In [4] 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. First, 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. Second, 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 parsimoniously. In [4], 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.

One smoothing strategy is called *shaping* (it is called “optimal shaping” in [5]). It consists of 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 [5] 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.

Our scenario is illustrated in Fig. 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. (Fig. 2 shows an

Manuscript received November 25, 1998; revised May 6, 1999 and February 8, 2000; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor R. Guerin.

J.-Y. Le Boudec is with the Institute for Computer Communications and Applications (ICA), EPFL, Lausanne 1015, Switzerland (e-mail: leboudec@epfl.ch).

O. Verscheure is with the IBM T. J. Watson Research Center, Yorktown Heights, NY, 10598, USA.

Publisher Item Identifier S 1063-6692(00)06792-3.

example of such a function, for an MPEG-2 video sequence.) The smoother output must satisfy the traffic envelope constraint given by some function σ 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 prefetching), namely, we do not require that $R'(t) \leq R(t)$. Look-ahead is commonly used with prerecorded 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 $R' \rightarrow R^*$ 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 queueing 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 extent on traffic conditions; however, the guarantee provided by the network can be formalized by a condition of the form [6], [7], [5], [8]

$$\forall t \geq 0, \exists s \leq t, \text{ such that } R^*(t) \geq R'(s) + \beta(t - s). \quad (1)$$

In the condition, β 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 ρ the rate. It is further assumed that the latency parameter L depends on the rate ρ according to $L = (C_0/\rho) + D_0$ 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 [8], [9]. We consider smoothing strategies that ignore the details of the network, but do know the service curve β .

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

$$\bar{D} = \inf \{t \geq 0 | \forall u \geq 0, v \geq 0: R(u+v-t) \leq \sigma(u) + \beta(v)\}.$$

We also give in this paper a simple formula to compute \bar{D} in practical cases. Second, there exists one smoother output \bar{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, $R'(t) \geq \bar{R}'(t - (D - \bar{D}))$. In other words, if we time-shift the optimal solution \bar{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 \bar{R}' is, at every time instant, no earlier than R' . The optimum $\bar{R}'(t)$ 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 \bar{R}' is also minimum. The optimal output \bar{R}' is given by

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

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 solution found in Section II to another strategy based on delay equalization. We show that in the CBR case, the latter is able to attain the optimal playback delay; in contrast, in the VBR 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 [10] 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.

II. OPTIMAL SMOOTHING

A. Formal Definition of the Admissible Smoother Output

Consider again the model illustrated in Fig. 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

$$R'(t) = 0, \quad \text{if } t \leq 0. \quad (2)$$

We assume that R' is constrained by the traffic envelope σ , namely

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

We also assume that the network offers a service curve β to the flow, namely, (1) is satisfied. It is more convenient to rewrite (1) as follows

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

As a convenient notation, the right-hand side in (4) is also traditionally written as $(R' \otimes \beta)(t)$, and is called the “min-plus” convolution of functions R' and β [5], [11], [12], [7]. This gives the equivalent writing for (4) as

$$R^*(t) \geq (R' \otimes \beta)(t). \quad (5)$$

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

$$R^*(t) \geq R(t - D_0 - D) \quad (6)$$

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 β . Thus, R' must be such that (6) is true for *any* realization R^* satisfying (4). Now remember that we have reduced our study to the case where the fixed part of the transfer delay is zero. 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 $R^*(t) = (R' \otimes \beta)(t)$. Thus for all $t > 0$ we must have

$$(R' \otimes \beta)(t) \geq R(t - D). \quad (7)$$

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 constraint for the smoother is to produce an output R' which satisfies simultaneously (2), (3), and (7).

B. Minimal Playback Delay

The first result in this paper is the following theorem.

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

$$\bar{D} = \inf\{t \geq 0 \mid \forall u \geq 0, v \geq 0: R(u + v - t) \leq \sigma(u) + \beta(v)\}.$$

The proof of the theorem is given in [13]. We give a numerical example later in this section (see Fig. 2). 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 σ , given that the network service curve guaranteed to the flow is β . The minimum delay \bar{D} can be better interpreted using the concept of horizontal deviation [8], which we now recall. Fig. 3 gives an intuitive definition.

Definition II.1: For two functions α and β , define the horizontal deviation $h(\alpha, \beta)$ by

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

It is shown in [13] that the value of the minimum playback delay \bar{D} in the theorem is given by

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

In the formula, $\sigma \otimes \beta$ is the min-plus convolution defined as in the discussion following (4), and which can be interpreted as follows [5], [8], [7]. Consider for a second a hypothetical shaper, as

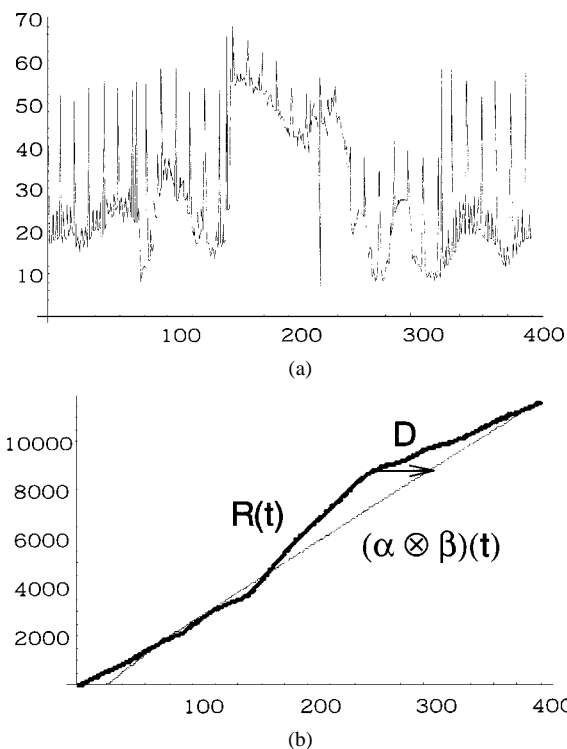


Fig. 2. MPEG-2 trace used as illustration. (a) Number of RTP packets per frame, or, equivalently, per 40 ms timeslot. (b) Cumulative function $R(t)$ counted in packets per timeslot (thick line), as well as the min-plus convolution $\sigma \otimes \beta$ of arrival and service curves (thin line). The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

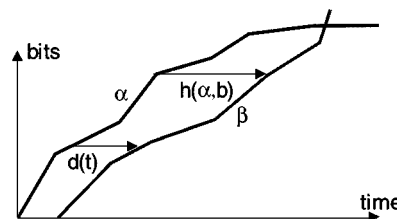


Fig. 3. Definition of horizontal deviation for two functions α and β . Determine $d(t)$ for all t by drawing the horizontal distance from α to β . The horizontal deviation $h(\alpha, \beta)$ is the maximum of all $d(t)$.

defined in the Introduction, with traffic envelope σ . Assume that σ is a “good” function, namely sub-additive, as explained for example in [5]. The arrival curves used with RSVP or for ATM VBR connections are good functions. We know from [5], [8], [7] 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 σ . 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 $(\sigma \otimes \beta)(t)$.

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). Our example is a 400-frame-long sequence conforming to the ITU-R 601 format. The sequence

is composed of three video scenes that differ in terms of spatial and temporal complexities. It has been encoded in an open-loop VBR mode. For this purpose, the widely accepted TM5 video encoder has been utilized. Fig. 2 shows the trace we use. We apply Theorem II.1 with the following parameters. The arrival 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 one 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 Mbit. The service curve is as with the Internet guaranteed service, with a latency L and a rate ρ equal 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 σ and the service curve β have the standard form used with the Internet integrated services, the computation of \bar{D} can be simplified as follows.

Proposition II.1: Assume that the arrival curve σ has the form $\sigma(t) = \min(M + pt, rt + b)$, and the service curve β 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 $F(k) = L + \max((k - M/p), (k - b/r), (k/\rho))$.

The proof is given in [13]. 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: 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 nonzero even for some negative times. After appropriate time-shifting, R^- is the optimal smoother output, as the following theorem shows.

Theorem II.2: This theorem is divided in two parts:

- 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 [13]. 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 $\bar{R}'(t) = R^-(t - \bar{D})$ the optimal output, namely the shifted version of R^- that starts at time 0. Then the theorem means that if we time-shift \bar{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 \bar{R}' is, at

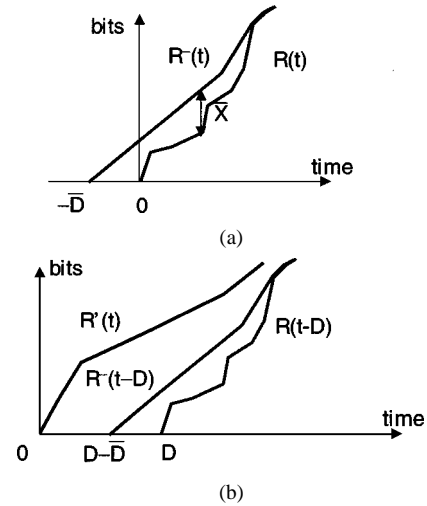


Fig. 4. Optimal smoothing. (a) Computation of $R^-(t)$ from the encoded signal $R(t)$. The minimum playback delay \bar{D} is the point where $R^-(-t)$ hits 0. (b) For any admissible smoother output $R'(t)$ with playback delay D , the shifted version $R^-(t - D)$ is no earlier than R' .

every time instant, no earlier than R' . The shifted optimal output $\bar{R}'(t - (D - \bar{D})) = R^-(t - D)$ thus gives the latest time at which every packet of the flow should be scheduled. Fig. 4 illustrates this.

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 [14] by

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

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

It is shown in [13] 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^- . Fig. 5 illustrates this representation on a very simplified scenario. The signal $R(t)$ consists of one large burst of B bits at time θ , and the network offers a constant delay (null network case; thus we drop β in the rest of this example). This scenario is extreme, but it represents an interesting limiting case. The figure shows the shifted optimal smoother output $R^- = R \oslash \sigma$, assuming the arrival curve σ has the standard form $\sigma(t) = \min(M + pt, rt + b)$.

In [4], 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 CBR. Thus, their representation is a particular case of our result.

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

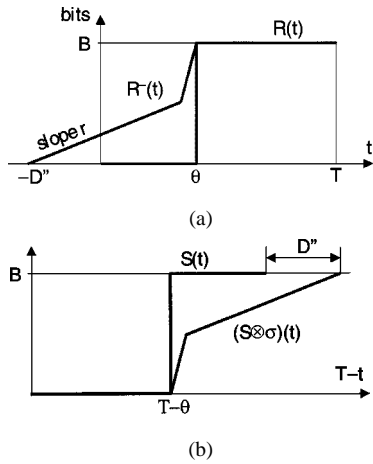


Fig. 5. (a) Bursty scenario for delay equalization, showing $R(t)$ and the shifted optimal smoother output $R^-(t)$ for this scenario. (b) $R^-(t)$ is obtained from R by a rotation of 180° around the center $((T/2), (R(T)/2))$. Obtain $S \otimes \sigma$ by shaping S according to the arrival curve $\sigma(t) = \min(M + pt, rt + b)$. Then $R^-(t) = (S \otimes \sigma)(T) - (S \otimes \sigma)(T - t)$ is obtained by inverting time again.

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 Fig. 4 shows the following corollary.

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

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

We show in [13] that the formula for \bar{X} can be interpreted in terms of network calculus abstractions, which leads to the following simplification.

Proposition II.2: Assume that the arrival curve σ has the form $\sigma(t) = \min(M + pt, rt + b)$ and the service curve β 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

$$\bar{X} = \sup_{t \geq 0} \{A(t) - \min[\sigma(t - L)^+, \beta(t)]\}$$

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 [13] we give an alternative method using the time-inversion representation, which has a complexity of $O(n)$. It is the same representation as in [4], 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 cir-

cuit-switched service, or, as an approximation, with ATM CBR services if the delay variation is 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 β from all formulas: for example, the minimum playback delay becomes

$$\bar{D} = h(R, \sigma) = \inf \{t \geq 0 \mid \forall u \geq 0: R(u - t) \leq \sigma(u)\}.$$

For a circuit-switched network service, σ 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 :

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

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

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 [5].

Optimal shaping is the standard method used to make an arbitrary flow conform to some traffic envelope σ . A shaper, with shaping curve σ , 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 σ . 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 σ 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 σ 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 [8]. 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 (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 σ (with σ sub-additive and $\sigma(0) = 0$), the output is equal to $S \otimes \sigma$ [5], [8] if S is the input. The transformation $S \rightarrow S \otimes \sigma$ is also a smoothing operation, and like the other one, it is idempotent, namely, $(S \otimes \sigma) \otimes \sigma = S \otimes \sigma$.

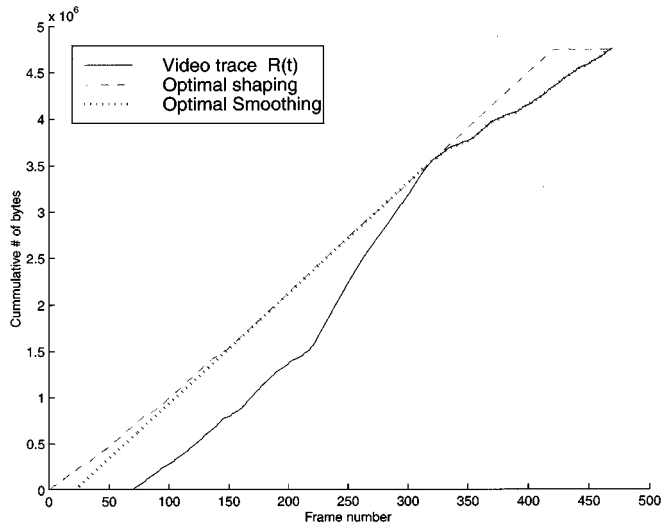


Fig. 6. Example of optimal shaping versus optimal smoothing for one MPEG trace. The example is for a network with constant delay, for a traffic envelope with $M = 1$ MPEG TS-packet, $p = 4.8$ Mb/s, $b = 80$ KB, $r = 2.4$ Mb/s. The figure shows the optimal shaper [resp. smoother] output and the original signal $R(t)$ shifted in time appropriately. The playback delay is 2.76 s for optimal shaping and 1.92 s for optimal smoothing.

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. Fig. 6 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 [13] 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 $S \otimes \sigma$. Fig. 5 illustrates that this corresponds to a rotation of 180° around the point $((T/2), (R(T)/2))$. 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 *anticausal*. 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

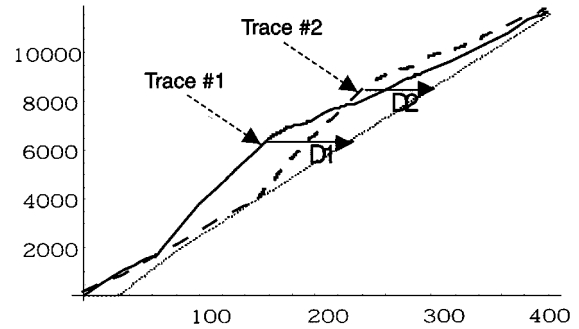


Fig. 7. The two traces have the same envelope, thus the same minimum buffer requirement (here, 928 KB), however the second trace has its bursts later, thus, has a smaller minimum playback delay ($D_2 = 2.05$ s versus $D_1 = 2.81$ s). The example is for the same network parameters as Fig. 2.

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.

In contrast, the minimum playback delay, as given by (9), does depend on the complete sequence, and not on the traffic envelope. Fig. 7 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 prefetching. 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 Fig. 1, we have now separate buffers for delay equalization and for compensation of prefetching. We would like to understand the impact of this separation on the minimum playback delay. The delay equalization buffer operates by delaying the first bit of data by an initial delay D' , equal to the worst-case delay through the network. Call D'' the initial delay at the decoding buffer. The total playback delay for this scenario is $D' + D''$. 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.

First, in order to understand where nonoptimality might come from, consider again the simplified scenario illustrated on Fig. 5 (in the rest of this discussion we call R_2^- what is shown as R^- on Fig. 5). 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 $D'' = (B - b/r) - \theta$. 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 $D' = L + (b/R)$. The minimum playback delay \bar{D} is given by $\bar{D} = (B - b/r) - \theta$ and, finally

$$D' + D'' = \bar{D} + \frac{b}{R}. \quad (11)$$

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 prefetching 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 [2], [8].

Second, (11) suggests a different outcome for the case $b = 0$, namely, the CBR 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 CBR 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 ρ , and further, that the latency parameter L depends on the rate ρ according to $L = (C_0/\rho) + D_0$. Using the IETF terminology, ρ is contained in the list of R-SPEC parameters. The constants C_0 and D_0 depend 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 ρ , provided by the network, is not known *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 σ 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 ρ) 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 Δ 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 Δ . Note that this problem is different from the problem of which arrival curve $\sigma(t) = \min(M + Pt, b + rt)$ is admissible [10], 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 [13]. The result is a procedure to test whether a choice of parameters (σ, T_0) is compatible with the playback delay D , as follows:

Given are a traffic envelope σ , a playback delay budget Δ , 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 $T_0 < D_0 - (b - M/p - r)$ then (σ, T_0) is not admissible,
- else compute ρ_2 as the only positive solution of $\bar{R}(\rho_2) + \rho_2(\Delta - D_0) - C_0 = 0$, where $\bar{R}(\rho) = \inf_u \{\rho u - R(u)\}$. If $r \geq (b + C_0/T_0 - D_0)$ then do the following. If $r \geq \rho_2$ then (σ, T_0) is admissible else not.
- Else (namely if $r < (b + C_0/T_0 - D_0)$), then compute $\rho_1 = (rt_0 + b + C_0/t_0 + T_0 - D_0)$, where $t_0 = (b - M/p - r)$ and do the following. If both $\rho_1 \geq \rho_2$ and $\bar{R}(r) + r(\Delta - D_0) + b - r(C_0/\max(\rho_1, r)) \geq 0$ then (σ, T_0) is admissible, else not.

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 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 formulas 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 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 CBR case, but not in the VBR case. Lastly, we have applied the theory to the practical problem to which a source is confronted when using the Internet guaranteed service.

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 [5], in particular, the use of min-plus deconvolution as a smoothing operator, and a representation of deconvolution with time inversion.

REFERENCES

- [1] T. V. Lakshman, A. Ortega, and A. R. Reibman, “VBR video: Trade-offs and potentials,” *Proc. IEEE*, pp. 952–973, May 1998.
- [2] 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*, vol. 31, no. 3, 1998.
- [3] J. Salehi, Z. Zhang, J. Kurose, and D. Towsley, “Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing,” presented at the ACM SIGMETRICS, May 1996.
- [4] J. Rexford and D. Towsley, “Smoothing variable bit rate video in an internetwork,” *IEEE/ACM Trans. Networking*, vol. 7, pp. 202–215, Apr. 1999.
- [5] C. Chang, “On deterministic traffic regulation and service guarantee: A systematic approach by filtering,” *IEEE Trans. Inform. Theory*, pp. 1096–1107, Aug. 1998.
- [6] R. L. Cruz, “Quality of service guarantees in virtual circuit-switched networks,” *IEEE J. Select. Areas Commun.*, pp. 1048–1056, Aug. 1995.
- [7] R. Agrawal and R. Rajan, “Performance bounds for guaranteed and adaptive services,” IBM, Tech. Rep. 20 649, 1996.
- [8] J.-Y. Le Boudec, “Application of network calculus to guaranteed service networks,” *IEEE Trans. Inform. Theory*, pp. 1087–1096, May 1998.
- [9] B. Braden, D. Clark, and S. Shenker, “Integrated services in the Internet architecture: An overview,” IETF, rfc 1633, June 1994.
- [10] S. H. Low and P. P. Varaiya, “A simple theory of traffic and resource allocation in ATM,” presented at the GLOBECOM, Dec. 1991.
- [11] F. Baccelli, G. Cohen, G. J. Olsder, and J.-P. Quadrat, *Synchronization and Linearity, An Algebra for Discrete Event Systems*. New York, NY: Wiley, 1992.
- [12] R. L. Cruz, “Sced+: Efficient management of quality of service guarantees,” presented at the IEEE INFOCOM, Mar. 1998.
- [13] J.-Y. Le Boudec and O. Verscheure. (2000) Optimal smoothing for guaranteed service. EPFL-DSC, Tech. Rep. DSC/2000/014. [Online]. Available: http://dscwww.epfl.ch/EN/publications/documents/tr00_014.pdf
- [14] R. Agrawal, R. L. Cruz, C. Okino, and R. Rajan, “Performance bounds for flow control protocols,” *IEEE/ACM Trans. Networking*, vol. 7, pp. 310–323, June 1999.

Jean-Yves Le Boudec (M’89) received the Agregation in Mathematics from the Ecole Normale Supérieure de Saint-Cloud, Paris, France, in 1980, and the Ph.D. degree from the University of Rennes, France, in 1984.

He became an Assistant Professor at INSA/IRISA, Rennes, in 1984. In 1987 he joined Bell Northern Research, Ottawa, Canada, as a Member of Scientific Staff in the Network and Product Traffic Design Department. In 1988, he joined the IBM Zurich Research Laboratory, Rüschlikon, Switzerland, where he was Manager of the Customer Premises Network Department. In 1994 he joined EPFL, Switzerland, where he is a Full Professor. His interests are in the architecture and performance of communication systems.

Olivier Verscheure (M’00) received the B.S. degree in electrical engineering from the Ecole Polytechnique de Mons, Belgium, in 1995, and the Ph.D. degree from the Swiss Federal Institute of Technology, Lausanne, in 1999.

He has been a Research Staff Member at the IBM T. J. Watson Research Center, Yorktown Heights, NY, since July, 1999. He was a Visiting Researcher at the Hewlett-Packard Laboratories, Palo Alto, CA, during the summer of 1997. His research lies within the areas of scalable multimedia servers, packet video, and vision science.