Rate Control for VBR Video Coders in Broad-Band Networks

Maher Hamdi, James W. Roberts, and Pierre Rolin

***Abstract*— We present a rate control algorithm adapted to MPEG video coders ensuring that output conforms to the parameters of a leaky-bucket network access controller. The algorithm avoids unpredictable rate variations without the rigidity and systematic coding delay of constant bit-rate (CBR) coders, and makes possible resource provision for guaranteed quality of service. A relatively large burst tolerance parameter allows considerable scope for variation at GoP scale, and only restricts the natural rate when necessary to avoid long-term overloads. Possible multiplexing schemes are discussed distinguishing buffer provision for cell-scale and burst-scale congestion.**

***Index Terms*—ATM networks, MPEG, quality of service guarantee, scales of congestion, shaping, statistical multiplexing, VBR video.**

I. Introduction

THE majority of traffic in broad-band networks is likely to be generated by video applications, whether it be for interactive videophone conversations or videoconferencing, consultation of prerecorded video sequences in multimedia databases, remote viewing of live events (conventions, sport), or simply watching a movie. While some rate adaptive video communications can be provided by best effort networks (e.g., low-resolution videoconference software on the Internet), most video applications impose quite severe constraints on network delay and throughput performance. To respect these constraints in packet-switched networks, including the ATM-based B-ISDN, it is necessary to implement preventive traffic control procedures to avoid congestion.

Preventive traffic control is based on the notion of a traffic contract [22]. A requested connection is described by means of a set of traffic parameters, and the network must decide if it can be accepted without violating quality of service (QoS) constraints. To ensure that the traffic characteristics of an accepted connection are as declared, the network must police the traffic parameters. It proves particularly difficult to identify traffic parameters for variable rate connections which are both significant in determining required network resources and easy to control. The choice in network standards has been to favor the second requirement over the first and to define traffic parameters through a rule, namely, the leaky bucket (known as the generic cell rate algorithm in ATM standards [22]). There are basically two approaches to using these rule-based parameters to describe video connections: either determine the values which most accurately characterize the output of a particular coder used in a particular context, or oblige the coder to make its output conform to predefined parameters by means of rate control.

A large number of studies have been performed on the characterization of coder output (e.g., [8], [10], [15], [16], [24], [26], and [29]). For certain applications such as videoconferencing, it may be possible to characterize the output succinctly in terms of a small number of parameters such as the first two moments of the per-frame bit rate and the coefficient of an assumed exponential autocorrelation function [17]. These parameters can be used in Markovian models to evaluate the performance of a network multiplexer (e.g., [6]). However, such statistical parameters cannot be efficiently policed. For less stereotyped video sequences, even these parameters are inadequate since the distribution of output rate can vary substantially for different minutes long segments of the same sequence [15], [10]. Indeed, long video sequences seem to systematically exhibit long-range dependence whose significant detrimental impact on performance is beginning to be well understood [2], [10], [8]. The incompatibility between the complexity of the parameters needed to describe the traffic and the very limited description provided by leaky bucket parameters is particularly marked for video connections.

The alternative approach of actively shaping the coder output so that it becomes more compliant and predictable has been considered by fewer authors. Reibman and Haskell [33] present bit-rate constraints that prevent codec buffer overflow in the case of a leaky-bucket-controlled channel. Heeke [15] and Coelho [4] aim to make the output behave like a predefined Markov chain. Pickering and Arnold [31] propose a rate control algorithm that produces a variable bit-rate (VBR) traffic lying between predefined upper and lower bounds. Pancha and El Zarki [30] also advocate using rate control with an algorithm similar to that of [33] where a few frames long bucket is used to control traffic variability.

The present contribution belongs to the second category. We pretend that a network cannot efficiently provide performance guarantees if video coder output results from unconstrained open-loop coding. We propose a closed-loop rate control algorithm which reacts on coder parameters according to the value of a leaky-bucket counter forcing the output to conform to

Manuscript received April 10, 1996; revised October 8, 1996. This paper was presented in part at the International Conference on Multimedia Networking (MmNet'95), Aizu-Wakamatsu, Japan, September 1995 and the Symposium on Multimedia Communications and Video Coding, Brooklyn, NY, October 1995.

M. Hamdi is with ENST de Bretagne, BP 78, 35512 Cesson Sévigné, France, on leave from the Telecommunication Services Laboratory, Swiss Federal Institute of Technology, EPFL, Lausanne, Switzerland.

J. W. Roberts is with France Télécom—CNET, 92131 Issy-les-Moulineaux, France.

P. Rolin is with ENST de Bretagne, BP 78, 35512 Cesson Sévigné, France.

Publisher Item Identifier S 0733-8716(97)04196-6.

sustainable rate and burst tolerance parameters used to describe the traffic characteristics of a required connection. Unlike the rate control suggested in [33], we propose to use a relatively large burst tolerance parameter allowing substantial short-term fluctuations while eliminating long-range dependence due to low-frequency (scene scale) variations. The algorithm is devised to fully exploit the potential for short-term variations using a prediction of open-loop rate to determine coder quantization settings. We show that a large burst tolerance is not incompatible with very short network delays, and that, indeed, end-to-end delay is considerably shorter than that of constant bit-rate (CBR) coding. The rate control algorithm is studied for use within the MPEG standards, although implementation in other coders is straightforward.

We first discuss in Section II how variable rate connections can be multiplexed in ATM networks while respecting quality of service guarantees, recognizing the critical distinction between cell scale and burst scale congestion. In Section III, we describe the essential features of the MPEG standards, and argue for the use of a shaping algorithm to control the burstiness of the VBR output. This algorithm is presented in some detail in Section IV. Statistical properties of the shaped traffic are presented in Section V based on simulated coder output, and the realized efficiency of statistical multiplexing is discussed in Section IV.

II. Multiplexing VBR Video

Video coding for communication cannot be studied in isolation from its impact on network performance. There clearly arises a need for compromise between network efficiency and image quality. In the following, we consider the relation between traffic characteristics and network multiplexer performance, and we discuss the possibilities for traffic control to guarantee QoS standards. We base the presentation on ATM multiplexing, but the considerations would also apply to a packet-switched network.

A. Quality of Service

The effect on end-to-end image quality of packet loss is not yet well defined. In early MPEG reference models, cell loss rates lower than 10^{-9} were proposed, but rates of 10^{-4} are currently being considered as acceptable. The effect of cell loss is not only dependent on the average cell loss rate, but also on the distribution of cell losses over time. Periods of high cell loss due to network congestion can have a serious detrimental impact on image quality.

Delay requirements clearly vary depending on the application. For interactive video communication applications, a maximum end-to-end delay of some 100 ms is appropriate [35], while a much longer delay would be tolerable for a user simply watching a recorded clip or movie in a video playback application. Delay requirements clearly have a strong impact on the type of network service to be provided. In the case of video playback, considerable variation in network delays of successive cells can be absorbed by a large buffer in a set top box, for example. The tight delay constraints for real-time communications, on the other hand, severely limit the possibility of dealing with congestion on network links by cell buffering. We note further that coding delays must be included in the overall delay budget, thus limiting the scope for rate smoothing in a closed-loop coder producing CBR output.

B. Multiplexer Performance

Studies on the performance of ATM multiplexers handling VBR traffic show that there are broadly two types of congestion leading to cell delays (e.g., [23], [28]). We assume that the instantaneous bit rate of a variable rate connection is well defined as, for example, when the traffic source is of the on/off type or when the output of a video codec is smoothed to a constant level over a frame duration. When the combined rate of all multiplexed sources is less than the multiplexer output rate, delays can occur due to the coincidental arrival of cells from different sources. These delays are of short duration, generally less than the time required to transmit a few tens of cells (i.e., around 1 ms). This type of congestion is referred to as cell scale congestion. The second type of congestion occurs when the combined rate exceeds the output rate. Such an overload is typically persistent, and the ensuing delays are much longer than those occurring in cell scale congestion. This type of congestion is known as burst scale congestion.

One option for controlling multiplexer performance is to ensure that the probability of a rate overload leading to burst scale congestion is negligible, and to provide the limited buffering necessary to avoid cell loss in case of cell scale congestion. Cell delays are then very small, with periods of burst scale congestion leading to cell loss. Consider a set of variable rate sources offered to a multiplexer of link capacity C, and let their combined bit rate at time t be Λ_t. The cell loss ratio can then be estimated by the fluid approximation

$$\mathrm{CLR} = \frac{E\{(\Lambda_t - C)^+\}}{E\{\Lambda_t\}} \tag{1}$$

where $E(\cdot)$ represents the expectation operator and X^+ is defined as $X^+ = X$ if $X > 0$ and $X^+ = 0$ if $X < 0$. The expectations can be calculated if the probability distributions of the rates of individual sources are known, and cell loss can be maintained within a target level by performing admission control: a new connection is refused if, according to formula (1), it would lead to cell loss greater than the corresponding QoS constraint. Delay and loss are much less easily controlled when the multiplexer has a large buffer designed to absorb burst scale congestion. It has been shown that the delay distribution and buffer saturation probability then depend significantly on the way the rates of individual sources vary in time. In particular, the autocorrelation function of the rate of successive video frames is known to have a very significant impact on the duration and severity of burst scale congestion. The traffic characteristics necessary to predict performance here are generally unknown at the start of a communication and cannot be policed by the network [1]. Long-term dependence observed in certain types of video sequences [2], [10] can lead to extreme congestion which can hardly be avoided by buffer dimensioning [27].

C. Traffic Control

Preventive traffic control relies on the network being able to perform admission control based on the declared values of traffic parameters, and then to police these parameters during a connection. Difficulties arise in defining traffic parameters which are sufficient for describing the traffic and can at the same time be policed. Apart from the source peak rate p, the only parameters so far agreed on are the sustainable cell rate and burst tolerance which can be controlled by the generic cell rate algorithm [22]. It is well known that the GCRA is equivalent to a leaky bucket, given an appropriate transformation of parameters.

For present purposes, we use leak rate r and bucket size b as traffic parameters, defining the leaky bucket $\mathrm{LB}(r,b)$ to be a counter incremented at rate r bits/s up to the maximum b and decremented as data are admitted to the network by the corresponding number of bits. We assume that data are discarded when the counter would otherwise decrease below zero. Traffic passing through $\mathrm{LB}(r_i,b_i)$ thus satisfies a burstiness constraint. Let $N_i(s,t)$ be the number of bits input to the network by source i in an interval (s,t). The leaky bucket then ensures the inequality

$$N_i(s,t) \le r_i(t-s)+b_i. \tag{2}$$

The overall mean input rate is thus bounded by r_i, and the maximum burst size at peak rate p_i is equal to $b_i p_i/(p_i-r_i)$. The way constraint (2) can be used in connection admission control depends on whether multiplexer buffers are provided for cell scale or burst scale congestion. In the former case, the cell loss ratio may be approximated using (1) provided the stationary distribution of the input rate Λ_t is known. If we do not know the distribution, but only that the input of all sources satisfies burstiness constraints, a conservative approach is to assume that offered traffic is the worst possible from the point of view of causing congestion while remaining compatible with (2). It is commonly accepted that such worst case traffic is of the on/off type with a source emitting bursts at peak rate p_i separated by silence intervals and such that the realized mean rate is r_i. If, for example, N similar connections are multiplexed with $r_i=r$ and $p_i=p$ for $i=1,2,\cdots,N$, the rate distribution is binomial

$$P\{\Lambda_t = np\} = \binom{N}{n}\alpha^n(1-\alpha)^{N-n}, \quad \text{where } \alpha = r/p. \tag{3}$$

It is known that buffering for cell scale congestion is efficient if the peak rates p_i are just a small fraction of the multiplexer output rate (less than 1/50, say) [3]. In this case, the relative variation of rate about the mean value is sufficiently small that link occupancy can attain 70%, say, while still satisfying a given low cell loss ratio. If the peak rate is higher ($>$1/10th of the link rate, say), a low cell loss ratio, estimated by (1), can only be achieved at a relatively low mean multiplexer utilization, implying higher transmission costs. Greater occupancy for high peak rate traffic can be achieved if buffering is provided for burst scale congestion. If all sources are constrained as in (2), it is known that cell loss can be avoided by providing a buffer of length $\Sigma\, b_i$ [5]. If an individual source i can be assured a minimum service rate of r_i (e.g., by implementing a scheduling mechanism such as weighted fair queueing [32]), then its cell delay is bounded by b_i/r_i. These bounds can be tightened somewhat by accounting for the finite peak rate. They are, however, still very conservative since they apply to a particularly pessimistic assumption where all sources simultaneously begin to transmit a burst of maximal size. Less conservative worst case assumptions for admission control are discussed in [7].

The choice of an appropriate multiplexing scheme for video applications depends on their particular performance requirements. For interactive applications, the use of cell scale buffering only has the significant advantage of providing very small network delay variation. Conversely, to efficiently use network resources in this case requires that the multiplexed video signal have a peak rate which is a small fraction of the link rate. Burst scale buffering can improve network efficiency, but the disadvantage is that delay variation can be wide and must be compensated for by an appropriately dimensioned playout buffer in the video receiver. The delay bounds can be rather large, and even then are only strictly appropriate if the network implements non-FIFO queueing disciplines guaranteeing a certain throughput per connection.

We have only considered leaky-bucket-defined traffic parameters as the basis for traffic control. The appropriateness of such a description remains debatable for many traffic sources. For video connections, in particular, the practicality and the usefulness of describing traffic in this way depend on the type of coding algorithm, as discussed in the next section. We return to the question of network multiplexing efficiency in Section VI.

III. Rate Control Algorithms in MPEG Coding

MPEG (the Motion Picture Expert Group of ISO/IEC) has defined video standards to satisfy a large variety of applications. MPEG-1 is suited to mass video storage and retrieval systems at rates up to 1.5 Mbit/s. More recently, MPEG-2 was standardized as a broadcast TV quality recommendation. Full details can be found in the standards [18]–[21]; for a more readable presentation, see [9]. The standards are becoming very popular, and are currently used in a number of video communication services including video on demand and World Wide Web browsers. In the following, we describe the essential features of the MPEG standard, and argue for the use of a shaping algorithm to be implemented in the coder to produce variable-bit-rate output with controlled burstiness.

A. MPEG Coding

The MPEG video syntax defines the group of pictures (GoP) structure containing three types of frames: I frames are intraframe coded (i.e., without reference to other frames) using two-dimensional discrete cosine transform; an I frame begins a GoP; P frames (predictive frames) are coded with reference to previous I and P frames using interframe coding; they achieve a better compression ratio than I frames; B frames (bidirectional frames) are coded with reference to the next and previous I or P frame. B frames achieve the highest compression ratio. The GoP is a sequence such as

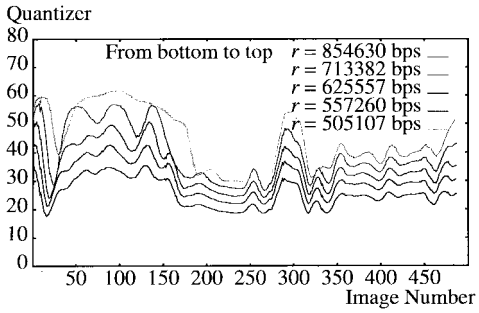

Fig. 1. CBR coding, the quantizer variation.

$IBBPBBPBB$. The number of P and B frames is set by the user. In particular, real-time video services may dispense entirely with B frames whose coding introduces additional delay. The use of these three frame types allows MPEG to be both robust and efficient. The coding algorithm is based on a division of each picture into blocks, groups of blocks and macroblocks. For present purposes, we assume that each macroblock is coded as an entity, notably with respect to the choice of a quantization parameter Q which determines spatial resolution. Bit rate and image quality decrease with increasing Q. The MPEG standard offers two coding options: CBR coding allowing the generated signal to be transmitted at constant rate with bounded delay and VBR coding where output rate variations are only constrained by the peak rate. While the precise rate control algorithm is not specified (it depends on the implementer), a reference CBR rate control algorithm was proposed in [18] for tests and comparison purposes.

B. Constant-Bit-Rate Coding

Codecs for video transmission have traditionally aimed to produce a CBR stream suitable for transport over circuit-switched telecommunications networks. The MPEG closed-loop algorithm is essentially based on the quantization parameter Q determining the resolution of the currently coded macroblock. A fixed quantity of bits is allocated to each GoP and apportioned progressively to successive pictures and, within pictures, to successive macroblocks. Bit-rate variability persists even at the GoP scale since the number of bits used may be different from the *a priori* assignment. The difference is taken into account in fixing the bit allocation of the next macroblock or GoP. Details of the algorithm are given in [18].

Fig. 1 shows how the quantization parameter Q changes from GoP to GoP for a particular video sequence. This is a 500-frame sequence from the TV program *Spitting Images* in "384 $\times$ 288" format coded with GoP structure $IBBPBBPBBPBB$. To smooth out high-frequency variations we have calculated the moving average of Q over the 12 frames of a GoP. The figure shows how the image resolution varies widely over the sequence, the amplitude of the variations being roughly independent of the target constant bit rate. The drawback of CBR coding is that the same bit rate is generated independently of the scene contents, thereby resulting in variable visual quality. A further disadvantage for real-time communications is the delay introduced by a smoothing buffer whose role is to compensate for residual variability. This residual variability is due to the natural differences between different macroblocks within a frame and between different frames, and is essential for picture quality. There is a need for compromise between the amplitude of this residual variability and the delay introduced by the smoothing buffer (and the compensating playout buffer at the receiver). Current coders introduce a systematic delay of around 200 ms.

C. Open-Loop Coding

In a packet-switched or ATM-based network, there is not necessarily an advantage to be gained from eliminating the natural variability of the signal generated by a coder since the signal bit rate is not constrained as in circuit switching. The MPEG variable-bit-rate coding algorithm uses open-loop coding where the same quantization parameter, q say, is used for all macroblocks. The rate depends on image complexity and activity, and image quality is said to be constant since the quality reduction is assumed to be the same for all scenes. When observing VBR video traffic, we can distinguish variability occurring over a range of time scales.

- *Packet Scale*—The data of a given frame may be packetized in different ways: as and when they are generated, macroblock by macroblock; all at once at the end of the frame at some peak bit rate for a fraction of the frame duration; at a constant rate calculated to fill the entire frame duration.
- *Frame Scale*—The MPEG algorithm introduces systematic variations from frame to frame due to the pattern of I, P, and B frames.
- *GoP Scale*—The bit rate averaged over a GoP varies in a correlated way from GoP to GoP as the image content changes; changes can be gradual within a scene or sharp in the event of a change of scene.
- *Scene Scale*—This kind of variation is responsible for the generation of large rate surges of uncontrolled duration; it has seldom been taken into account in VBR traffic characterization studies (see [8] for an example of a scenic model).

Variations at multiple time scales make it particularly difficult for the network to accommodate VBR video traffic in a guaranteed QoS environment. Fig. 2 shows how the output rate, averaged over a GoP, varies for different quantization parameters for the same test sequence. The parallel between the rate variations in open-loop coding and quantization variation in closed-loop coding is obvious, and it gives an intuition for the approximate relationship considered in Section IV-B.

D. Controlled Burstiness VBR Coding

As discussed in Section II, to efficiently multiplex VBR video streams requires the knowledge of traffic parameters describing the rate variations. Even for stereotyped video applications such as videoconferencing, the characteristics of the rate generated by an open-loop coder will depend on factors such as the number of participants, their activity, and even the way they are dressed. Movies have a rate which varies widely as scenes change with persistent periods of heavy

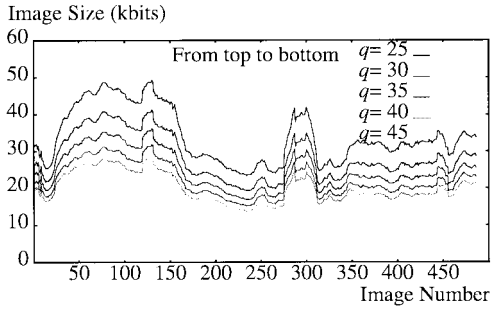

Fig. 2. Open-loop coding, bit-rate trace.

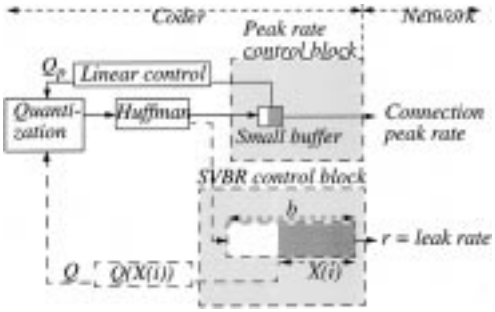

Fig. 3. Burstiness control using a virtual buffer.

traffic in case of high activity and complex image structure. These low-frequency variations are difficult to foresee, and can cause congestion over significant time periods. On the other hand, frame scale variations are an essential result of the MPEG intra- and interframe coding algorithm, and must be preserved. Furthermore, there is no necessity to eliminate variations at GoP scale. We propose, therefore, to seek a compromise between open-loop coding and CBR coding. We pretend that the full variability of open-loop coding is not necessary to maintain the subjective quality of video sequences containing scenes of different types. Quality from the user point of view depends mainly on the visual capacity to capture the information displayed on the screen. In fast moving scenes with complex image structure, the human eye does not have enough time to discover all image details. We suggest that the high bit rate generated for such scenes by an open-loop coder is unnecessarily generous. On the other hand, scenes with little motion and simple structure are more sensitive to signal degradations. Their quality should be maintained at a satisfactory level. We believe that subjective quality may actually be improved by restricting the scope for scene scale variations: for a given mean rate, higher resolution in low-activity scenes more than compensates for poorly perceived detail in fast-moving and complex sequences. It is for this reason that CBR coders generally produce acceptable visual quality. However, we would argue further that CBR is unnecessarily restrictive, and that the use of an appropriate rate control algorithm can provide "network-friendly" output with controlled rate variability. In fact, VBR rate control algorithms are known to be necessary to fit bit-rate profiles/levels defined by the MPEG-2 standard. Such algorithms (e.g., [31]) are designed to optimize the perceived quality with no regard to traffic burstiness. We propose to use a closed-loop algorithm to ensure that the volume of data emitted satisfies the following burstiness constraint: in any sequence of k successive GoP's, the number of bits emitted $N(k)$ satisfies

$$N(k) \leq rk + b. \tag{4}$$

This choice is motivated by the widespread use of leaky-bucket-like algorithms to control the traffic offered to packet-switched and ATM-based networks. We consider the leaky bucket $\mathrm{LB}(r,b)$ defined in Section II-C where, for notational convenience, r is measured in bits/GoP and b in bits. By maintaining an image of the counter, the coder can implement a closed-loop control to ensure its output conformance, and thus avoid data loss at the network interface. We assume for present purposes that the counter is adjusted on a GoP-by-GoP basis, and say the coder conforms to $\mathrm{LB}(r,b)$ when its output satisfies the above burstiness constraint. (In practice, an access controller typically works on a packet or cell basis, and it would be necessary to allow for packet scale variations to ensure conformity.) The task of the closed-loop control is to maintain the leaky-bucket counter within the range $[1, b-1]$. A zero value would mean the risk of packet discard, while a counter value of b means that not all of the available rate r is being used.

The leaky bucket may be viewed as a fictitious buffer whose current state is given by the value of the counter. The control thus parallels that employed in the CBR algorithm where a smoothing buffer is inserted between the coder and the output line. The essential difference is that, in the present case, the cells are not actually delayed. The size of the counter is thus not constrained by the need to reduce coding delay, and can be as large as necessary to maintain the quality levels provided by GoP scale variations. Furthermore, the rate adjustment algorithm can be much simpler than that employed in the CBR coder. Instead of minute macroblock-by-macroblock variations, the quantization parameter can be fixed on a frame or GoP basis since the fictitious buffer is much greater than the real CBR buffer, allowing wider variations about the given mean rate.

IV. The Shaping Algorithm

In this section, we describe a rate control algorithm to be implemented in the MPEG coder to ensure that its output conforms to a leaky bucket defined by its leak rate r and virtual buffer size b. The proposed algorithm produces a "shaped bit-rate" output stream, and we give it the acronym SVBR for *shaped variable bit rate*, to distinguish it from CBR and open-loop coding. The proposed control algorithm is to be used in addition to the peak rate control algorithm that is necessary to fit the coder output to the link capacity. Fig. 3 shows the SVBR control block to be added to a classical MPEG coder. It should be noted that, since rate control features are not subject to standardization, SVBR coders remain fully MPEG compatible. In particular, SVBR control is completely transparent to decoders, and there is no need for them to be modified.

A. Principle

The SVBR algorithm operates in open loop to code the different frames and macroblocks of a GoP, while the quantization parameter Q is adjusted from GoP to GoP to control the extent of bit-rate variations. Note that, since we allow rate variations over several GoP's, there is no need for present purposes to adjust Q on a finer scale as in the CBR algorithm. The SVBR algorithm is thus considerably less complex than that described in [18].

The adjustments are derived from the value of a counter $X(k)$ which records the number of leaky-bucket credits spent at the start of the kth GoP. Let $R(k)$ be the number of bits generated in GoP-k (i.e., $R(k)$ is the rate measured in bits/GoP). $X(k)$ then evolves as follows:

$$X(k+1) = \min\{b, (\max\{0, X(k) - r\} + R(k))\}. \quad (5)$$

We have $0 \leq X(k) \leq b$, for all k. The initial value $X(0)$ is arbitrarily chosen to be $b/2$.

The algorithm aims to adjust the GoP-k quantization parameter $Q(k)$ which determines the rate $R(k)$ to ensure that $X(k)$ is neither too close to b nor too close to zero. In the former case, the coding tends to CBR coding at rate r; in the latter, the coder does not fully use the available average bit rate r. The adjustments to $Q(k)$ should allow flexible, open-loop-like control when $X(k)$ is in a middle range around $b/2$ while attracting it back to this range if it tends to approach either extreme, zero or b.

B. GoP Scale Rate Prediction

Generated bit rate decreases with increasing quantization parameter, but the exact relationship between Q and R varies in time and depends on instantaneous activity and motion. If known, the function $Q(R)$ gives the appropriate value of Q corresponding to a desired output rate. When performing CBR coding, the control loop acts on the macroblock scale. In this case, the precise $Q(R)$ function is very complex, and needs to be approximated using predefined codebooks, for instance, [31]. In SVBR coding, the control loop acts on a time scale larger than that of the macroblock, and it proves possible to derive a much simpler relationship between $R(k)$ and $Q(k)$. Based on the analysis of a 500-frame sequence from the TV program *Spitting Images*, we derive the following approximate relationship between Q and R:

$$Q(k) \approx \frac{K(k)}{R(k)}. \quad (6)$$

$K(k)$ is a constant that depends only on the scene complexity (i.e., depends on k). In Fig. 4, we plot $R(k)^{-1}$ as a function of $Q(k)$ (ranging from 2 to 61) for six randomly chosen, open-loop coded GoP's (GoP-k starts at image 100^*k). Clearly, the curves of Fig. 4 can be approximated by linear functions of the form (6). Note that the approximate principle stated above is significant at the considered GoP time scale since detailed rate/distortion properties observed for CBR coding are masked by the averaging operation.

Expression (6) implies that the product RQ for a given GoP is independent of the rate control algorithm used by the coder,

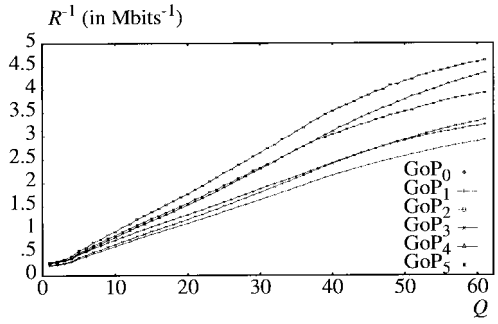

Fig. 4. Linear approximation of the Q-R^{-1} relationship.

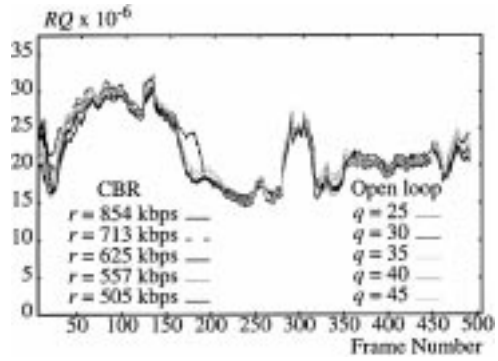

Fig. 5. RQ product.

TABLE I
THE CONSTANT RQ PRODUCT

Open Loop		CBR		RQ
q	$E[R_{open}(k)]$	R (in bps)	$E[Q]$	$Rxqx10^{-6}$
25	854630	854630	25.62	21.36
30	713382	713382	30.44	21.40
35	625557	625557	34.79	21.89
40	557260	557260	39.80	22.29
45	505107	505107	44.86	22.72

and for a given algorithm, is independent of the quantization value used for coding the GoP. To verify this property, we compressed the video sequence described above in both CBR (using five different bit rates) and open-loop modes (using five values for the quantization parameter). Each time, the RQ product is plotted versus the frame number (see Fig. 5). If expression (6) were exact, curves of Fig. 5 would be identical. The observed discrepancies are caused by the nonlinear parts of the curves of Fig. 4. Table I shows the matching of the averaged values of Q and R for the two coding modes (CBR and open loop). The parameter q denotes the constant quantizer used in open-loop coding, and R_{open} is the corresponding average rate. Furthermore, as a function of the GoP number $k, K(k)$ is a highly correlated process (as shown in Fig. 5). It can, in fact, be considered as a global measure of scene complexity because it depends only on GoP-k spatial and temporal activity.

This relationship is used as a GoP rate prediction method. Consider a rate control operating at the GoP scale to satisfy some traffic constraints, i.e., before coding GoP-$(k+1)$, an

algorithm gives the target bit allocation $R(k+1)$ (in bits) of that GoP. Using expression (6) and approximating $K(k+1)$ by $K(k)$, we obtain

$$Q(k+1) = Q(k)R(k)/R(k+1). \tag{7}$$

This expression gives the quantization parameter value to be used to obtain the desired bit allocation $R(k+1)$.

C. Rate Adjustment

In order to respect the burstiness constraint (4), it is necessary to adjust the quantization parameter of GoP-k to ensure that the counter value $X(k+1)$, determined from (5) using the current frame rate $R(k)$, remains less than b. Given the empirical relationship (7), any number of rate adjustment algorithms can achieve this objective. A simple solution would be to make the rate change in proportion to the distance of $X(k)$ from a median value of $b/2$: the quantization parameter changes most drastically as the leaky-bucket counter tends to its limits zero or b. Such an algorithm pays no attention to the current scene activity except indirectly through the current value of $X(k)$. We have preferred to develop an alternative algorithm where rate adjustments take account of the fluctuations which would occur with classical open-loop coding. To do so, we need to introduce a supplementary coding parameter q. This is the constant quantization which in open-loop coding would produce an average rate equal to r. How appropriate values of the three coder parameters r, b, and q could be chosen in practice is discussed in Section IV-D below.

The control principle is shown in Fig. 6. Scene activity is measured using a prediction of the equivalent open-loop bit rate R_{open} defined as follows: $R_{\text{open}}(k)$ is the rate of *GoP-k* which would result from an open-loop coding with a quantization parameter q initially fixed by the user. A scene with reasonable activity and duration is coded at the rate R_{open} while excessively long and/or active scenes are "truncated" and their bit rate is reduced to r. This means that for periods where R_{open} conforms to the traffic contract, the shaping algorithm behaves like open-loop control. On the other hand, during overload periods (those where R_{open} does not conform to the traffic contract), the algorithm aims to bring the rate down to r. During these periods, image quality may be reduced to that of CBR coding. However, because network resources are dimensioned based on the leaky-bucket conformance, this shaping avoids cell loss which could otherwise occur at rate up to $R_{\text{open}} - r$. Thus, only harmful scenes are shaped. When $X(k)$ approaches zero and the open-loop rate would typically be less than r, the algorithm provides a lower quantization than q to attain rate r. A higher rate is not necessary here, and the leaky-bucket counter can remain at a low level in anticipation of a future change in activity which can thus be more readily accommodated. The algorithm is described more explicitly below.

- For high-activity scenes $(R_{\text{open}}(k) \geq r)$:
  when $X(k) \approx 0$, the algorithm behaves as in open loop, i.e., $R(k)$ is set to $R_{\text{open}}(k)$
  when $X(k) \approx b$, the algorithm behaves like CBR, i.e., $R(k)$ is set to r.
- For low-activity scenes $(R_{\text{open}}(k) \leq r)$:
  when $X(k) \approx 0$, the algorithm behaves like CBR, i.e., $R(k)$ is set to r
  when $X(k) \approx b$, the algorithm behaves as in open loop, i.e., $R(k)$ is set to $R_{\text{open}}(k)$.

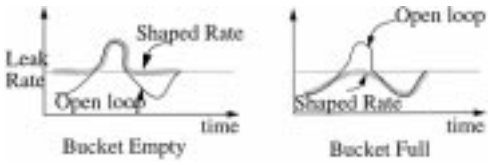

Fig. 6. Principle of the shaping algorithm.

When the bucket is partially filled, $R(k)$ is set to a linear combination of the two extreme cases stated above. The SVBR algorithm is intended to produce a satisfactory compromise between CBR and open-loop coding with $R(k)$ lying between r and $R_{\text{open}}(k)$, closer to the minimum of r and $R_{\text{open}}(k)$ when $X(k)$ is near b and closer to their maximum when $X(k)$ is near zero. These requirements are satisfied by the following relations:

$$R(k) = (1-\varepsilon_1(x))R_{\text{open}}(k) + \varepsilon_1(x)r \quad \text{if } R_{\text{open}}(k) > r \tag{8}$$

$$R(k) = \varepsilon_2(x)R_{\text{open}}(k) + (1-\varepsilon_2(x))r \quad \text{if } R_{\text{open}}(k) \leq r \tag{9}$$

where $x = X(k)/b$.

The functions $\varepsilon_1(x)$ and $\varepsilon_2(x)$, satisfying $0 \leq \varepsilon_i(x) \leq 1$, are increasing functions of the buffer fullness. Their explicit form is subject to tuning.

To realize relations (8) and (9), we must act on the quantization parameter $Q(k)$. Applying expression (6) for the GoP-k rate $R(k)$ and quantization $Q(k)$ and for $R_{\text{open}}(k)$ and q gives

$$R(k)Q(k) = R_{\text{open}}(k)q. \tag{10}$$

Finally, using (7)–(9), the algorithm becomes

$$Q(k+1) = \frac{qR_{\text{open}}(k)}{(1-\varepsilon_1(x))R_{\text{open}}(k) + \varepsilon_1(x)r} \quad \text{if } R_{\text{open}}(k) > r \tag{11}$$

$$Q(k+1) = \frac{qR_{\text{open}}(k)}{\varepsilon_2(x)R_{\text{open}}(k) + (1-\varepsilon_2(x))r} \quad \text{if } R_{\text{open}}(k) \leq r \tag{12}$$

where $R_{\text{open}}(k) = (R(k)Q(k)/q)$.

Note that the recurrence relations (11) and (12) are entirely defined by the three coding parameters r, b, and q.

D. Parameter Settings and Visual Quality

As with CBR and open-loop coding, it is necessary to choose coding parameters according to cost and quality criteria. In CBR coding, the rate parameter r must be chosen to achieve a satisfactory compromise between image quality of the cost of transmission. Typically, a different rate would be chosen according to the video content (e.g., sport, person

talking, music clips, etc.) and the capacity of the receiver (PC, TV, wide screen, etc.). In open-loop coding, it is the quantization parameter q which determines quality, and an appropriate value depends again on video content and the desired cost/quality tradeoff. For the SVBR coding algorithm proposed in Section IV-C, it is necessary to fix r and q conjointly, bearing in mind that they are, in theory, related through the open-loop coding algorithm. For stored videos, the appropriate choice of q can be determined exactly by simulating the open-loop algorithm in a preliminary phase. For real-time applications, it should be possible to establish an empirical relation $q(r)$ for a given type of communication (videoconference, lecture, etc.), thus reducing the problem to choosing a single rate parameter r as for CBR. The optimal choice of r and q remains an open issue, whether it be for CBR, open-loop, or SVBR coders, and it is largely beyond the scope of the present work. We note, however, that while the choice of q will have an impact on image quality, it does not change the traffic characteristics of the coder output insofar as these will always satisfy the burstiness constraint (4).

SVBR introduces the additional parameter b. In fact, r and b determine the leaky-bucket parameters which, together with the peak rate, determine the traffic characteristics of the connection to be established in the network. The value of b has an impact on video quality and network performance. The larger the value of b, the greater the scope for rate variability leading, in principle, to higher quality coding. Note that b has absolutely no impact on the coding delay since there is no physical buffering. Indeed, the absence of delay constitutes a major advantage of SVBR compared to CBR where a physical smoothing buffer must be implemented and variable delays must be compensated for in a playout buffer. In the network, the impact of b depends on the type of multiplexing employed, as discussed in Section II. In particular, if cell scale multiplexing is employed, b has no effect on the average cell loss ratio since this depends uniquely on the stationary rate distribution, but it does influence the way cells are lost: a larger value of b increases the probability of prolonged overload, leading to grouped cell losses. If multiplexing relies on large buffers to absorb burst scale congestion, the value of b directly determines the delay bounds which can be guaranteed, as discussed in Section II. An alternative networking solution based on shaping the coder output to reduce its peak rate is discussed in Section VI below: in this case, the choice of b determines the size of the buffer necessary to perform the shaping. The optimal compromise between allowed variability and facility of traffic control depends on the result of subjective tests, which are again beyond the scope of this paper. However, we expect a value of b equal to the content of several GoP's to be a judicious choice, allowing the frame scale and GoP scale variations which are necessary for coding quality while eliminating the possibility of sustained overloads due to scene scale variations and their undesirable impact on network performance.

V. Numerical Results

We implemented the SVBR control algorithm in the software MPEG-1 coder developed by the MPEG Group. Tests were performed on a 2000-frame-long video sequence taken from the *Spitting Image* TV program. The sequence was coded in CBR with target rate $r = 0.78$ Mbit/s, in open loop with $q = 35$ (producing an average rate of 0.78 Mbit/s), and in SVBR with parameters $r = 0.78$ Mbit/s, $q = 35$, and $b = 564710$ bits (equivalent to 18 average frames). The frame rate was 25 frames/s and the GoP size was 12 frames.

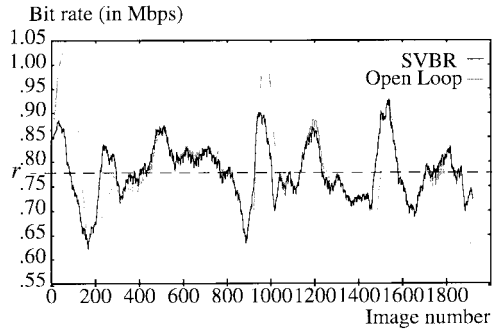

Fig. 7. Instantaneous bit rate of SVBR and open-loop traces.

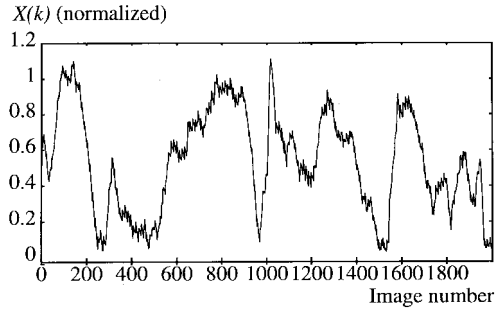

Fig. 8. SVBR algorithm, the virtual buffer fullness.

A. Rate Variation

Fig. 7 shows the bit rate generated by open-loop and SVBR algorithms. To remove high-frequency variations, the plotted rates are the moving average of seven consecutive GoP's. The SVBR algorithm generates less traffic than open loop in active scenes (frames 1–150 and 900–1000), and compensates by providing higher rates in calmer periods (200–300, 1400–1500). Compared to open-loop coding, rate variability is maintained, but with smaller amplitude. Corresponding variations in the leaky-bucket counter are illustrated in Fig. 8. This curve confirms that the algorithm indeed exploits the full range of variability provided by burstiness parameter b.

B. Quantizer and PSNR Variations

Fig. 9 compares quantization parameter variations of SVBR and CBR coding. Quantizer variations are much more stable with the shaping algorithm. The quantizer varies only during very active scenes to shape their bit rate or during very low-activity scenes to enhance their quality.

Although only psychovisual tests can decide about the visual quality, we have plotted the peak-signal-to-noise ratio of the three algorithms in Fig. 10. For reasons of clarity, only results for frames 50–300 have been plotted. First, we note that the PSNR of the shaped output is always higher than the minimum

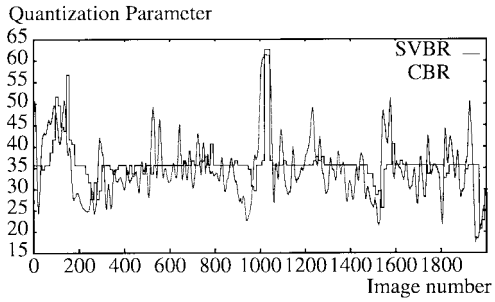

Fig. 9. Quantization parameter variation.

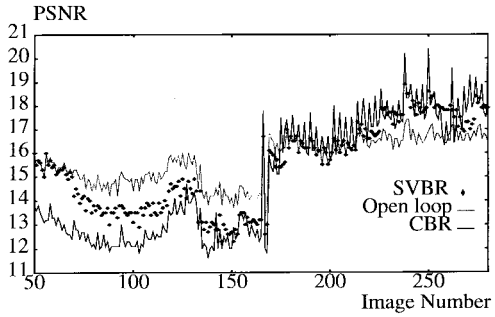

Fig. 10. PSNR comparison.

of the open-loop and CBR PSNR. It is equal to their maximum when the virtual bucket is empty, and is equal to the minimum when the virtual buffer is full. If the coding parameters r and b are chosen correctly, the PSNR is most of the time equal to the maximum of that of open loop and CBR since only scenes that cause congestion (buffer full) are shaped.

C. Statistical Characteristics

A second sequence (10 000 frames long) has been used to test the effect of shaping on the coder bit-rate distribution and autocorrelation function. This sequence was captured in CIF format, and represents a music video clip showing high-image complexity, a wide range of colors, numerous scene changes, zooms, and almost no fixed plans.

Fig. 11 illustrates the bit-rate variations observed with open-loop and SVBR coding. We have again eliminated high-frequency variations by averaging the bit rate over a number of GoP's (ten in this case). Fig. 12 shows the stationary distribution of the frame size.

The SVBR algorithm eliminates long-range dependence. This is manifested through the autocorrelation function which decreases much more rapidly than that of the open-loop coding output, as shown in Fig. 13. In fact, it can easily be demonstrated formally that an input stream satisfying a burstiness constraint (4) where the leak rate is equal to the traffic average rate is not self-similar (i.e., does not exhibit long-range dependence) [11].

VI. Multiplexing Efficiency

As the output of an SVBR coder conforms exactly to the traffic parameters p (peak rate), r (the realized mean rate), and

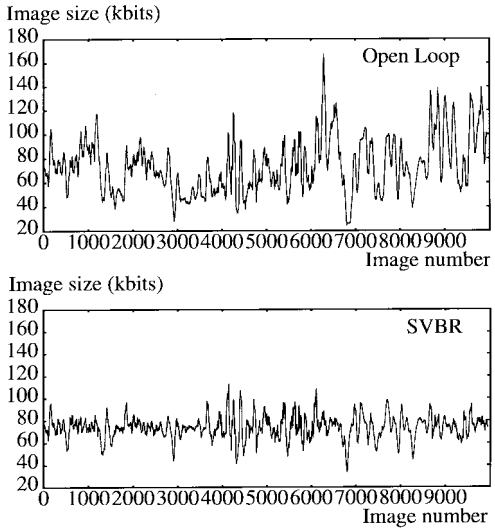

Fig. 11. Instantaneous bit rate.

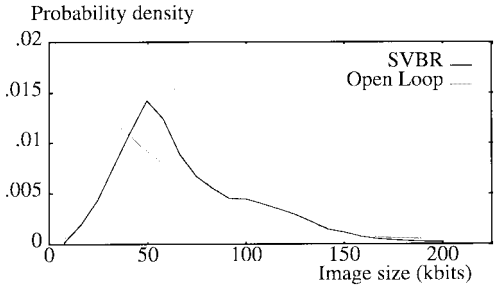

Fig. 12. Probability density for open-loop and SVBR traces.

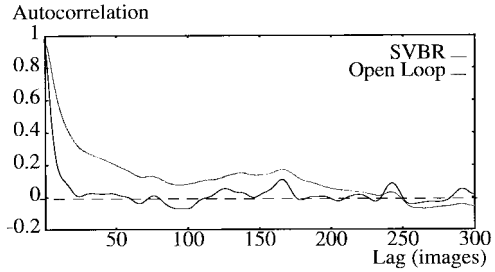

Fig. 13. Autocorrelation functions for open-loop and SVBR traces.

b (determining burst tolerance), statistical multiplexing can be performed as described in Section II-C.

Consider first multiplexers equipped with small buffers designed to take care of cell scale congestion only. For illustration purposes, assume a link carries N independent video connections with rate parameter $r = 1.8$ Mbit/s and peak rate p. Assuming worst case on/off traffic, the cell loss ratio for a given link capacity C can be computed from relation (1) with the binomial rate distribution (3):

$$\mathrm{CLR} = \frac{\sum_{n=\lceil C/p \rceil}^{N} (np - C)\binom{N}{n}\alpha^n(1-\alpha)^{N-n}}{Nr}. \quad (13)$$

It is thus possible to compute the maximum value of N compatible with an assumed target cell loss ratio of 10^{-6},

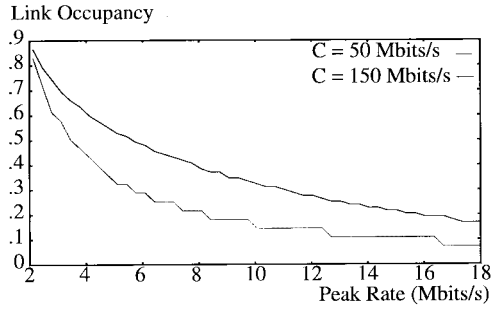

Fig. 14. Cell scale dimensioning for $r = 1.8$ Mbit/s and CLR $= 10^{-6}$.

and consequently deduce achievable link utilization (Nr/C). Fig. 14 plots this utilization for two link rates, 50 and 150 Mbit/s. It is clear from the figure that this kind of multiplexing is efficient for moderate peak rates only. Higher efficiency could be achieved if the distribution of the per-frame rate (assuming this rate is realized by spacing cell emissions over the frame duration) were known and given by a histogram, as in the case illustrated in Fig. 12. Roughly the same multiplexing gain would be achieved with either open-loop or SVBR coding in this particular example. The difference between the two cases is rather in the nature of the guarantees possible. The rate of open-loop coding with a given quantization parameter q can change quite drastically from one video sequence to another so that the rate distribution is not intrinsic to the coder. This is less the case with SVBR coding where the rate control algorithm guarantees the same mean rate r for any sequence and considerably constrains frame-to-frame variations. However, even here, to assume the worst case on/off traffic pattern is the only sure way to strictly guarantee cell loss rates since only the mean of the rate distribution can be policed. For real-time video connections where, we would argue, burst scale congestion should be avoided, both the peak rate and mean rate parameters should be a small fraction of the link rate. In the example considered in Fig. 14, the mean rate of 1.8 Mbit/s is probably already too big for the considered link sizes.

The worst case rate binomial distribution may be unduly pessimistic. If data emission is smoothed over a frame duration and the actual per-frame rate distribution is known, the cell loss ratio can again be calculated by (1). The multiplexing efficiency is clearly greater. The drawback is that the distribution is not known *a priori* as a function of the shaping parameters r and b.

Video applications with less severe time constraints than interactive communications can certainly be handled more efficiently at the expense of longer (although guaranteed) delays. These delays could occur in the coder or in the network. Note that in both cases, it would then be necessary to compensate for delay variability by an appropriately dimensioned playback buffer in the decoder.

To add delay in the network implies operating multiplexers with burst scale congestion. Burst scale congestion can be controlled using the SVBR algorithm through the burstiness constraints (4) and the bounds on multiplexer delay discussed in Section II-C. For the sake of simplicity, assume that (4) can be replaced by the more precise constraint

$$N(s,t) \leq r(t-s)+b, \qquad \text{for any interval } (s,t) \tag{14}$$

where $N(s,t)$ is the amount of coded data between in the interval (s,t). In addition, we assume that peak rate control implies the following:

$$N(s,t) \leq p(t-s) \qquad \text{for any interval } (s,t). \tag{15}$$

The relationship between constraint (4), evaluated on a GoP basis, and the above fluid approximations of constraints operating at the network input is examined in [11]. We prefer to omit this discussion herein for the sake of conciseness.

Adding α times (14) to $(1-\alpha)$ times (15), with $0 \leq \alpha \leq 1$, we deduce a family of burstiness constraints

$$N(s,t) \leq (\alpha r + (1-\alpha)p)(t-s) + \alpha b. \tag{16}$$

A service rate $\rho(\alpha) \geq \alpha r + (1-\alpha)p$ is sufficient to ensure a delay of $(\alpha b/\alpha r + (1-\alpha)p)$. Multiplexing with cell scale congestion corresponds to choosing $\alpha = 0$, the principle being that the service rate is never less than the sum of peak rates of active sources (multiplexing delay is zero in the fluid approximation). It is possible to meet any delay budget between zero and b/r by an appropriate choice of α if the service rate $\rho(\alpha)$ can be guaranteed. To make such a service rate guarantee to an individual connection does, however, imply that multiplexers are equipped with per-connection queue scheduling schemes such as weighted fair queueing. The alternative of adding delay in the coder to make the traffic more amenable to multiplexing (with cell scale, FIFO buffering) can be realized more simply.

Adding a FIFO queue of service rate p' between coder and network reduces the peak rate of the output from p to p'. Link utilization can thus be improved, as indicated in Fig. 14, while the burstiness constraints still apply so that delay and required buffer size can be calculated. From (16), we deduce $\alpha = (p-p'/p-r)$; the FIFO buffer should thus be of size $(b(p-p')/p-r)$, and its delay is bounded by $D = (b(p-p')/(p-r)p')$. For given mean and peak rates, the delay D is proportional to the burst tolerance b. This clearly expresses a burstiness–interactivity tradeoff that can be controlled by choosing the right value of b. The network only deals with cell scale congestion and its performance in terms of cell-loss ratio is controlled by (13) where the considered peak rate is p'. For a given CLR value, the link utilization depends on the allowed delay D. As shown in Fig. 15, link utilization increases with increasing D. It should be noted that satisfactory utilization (e.g., 0.7) can be achieved at the expense of reasonable delay (150 ms for the 150-Mbit/s link and 200 ms for the 50-Mbit/s link). This multiplexing scheme seems to be suitable for moderately interactive applications such as video on demand or TV distribution. Again, such schemes are based on the prior knowledge of traffic parameters ensured by SVBR coding.

VII. Concluding Remarks

For the transport of video communication application in a broad-band network, it is necessary to find a satisfactory

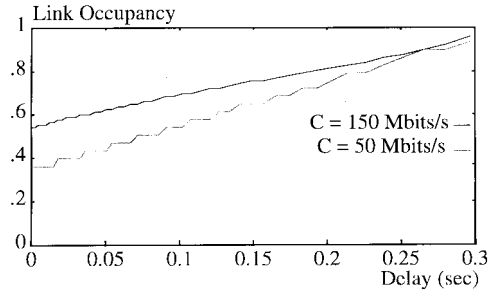

Fig. 15. Link utilization versus delay, $r = 1.8$ Mbit/s, $p = 5$ Mbit/s, $b = 560$ kbit/s, and CLR $= 10^{-6}$.

compromise between the range of rate variability required to ensure high-quality images and the predictability of such variations, necessary to be able to meet network quality of service constraints.

Preventive traffic control standards for B-ISDN rely on describing traffic streams by the parameters r and b of a leaky bucket. In order to ensure that a video communication conforms to such parameters, it is necessary to introduce a closed-loop control algorithm in the coder. In this paper, we present such an algorithm which can be easily implemented in an MPEG coder.

The SVBR algorithm adjusts the coder quantization parameter on a GoP-by-GoP basis to ensure that the output satisfies the burstiness constraint imposed by the leaky-bucket traffic control. The applied adjustments take account of scene activity, and aim to follow the natural variability of open-loop coding except when this would lead to nonconformity with the traffic contract. In addition to the leaky-bucket variables r and b, SVBR requires a further parameter q representing the constant quantization necessary with open-loop coding to achieve the average rate r.

Network resource provision is based on the coding parameters r and b together with the source peak rate p. Since the SVBR algorithm realizes a mean rate r while satisfying the leaky-bucket constraints, multiplexing with performance guarantees can be performed efficiently with buffering for either cell scale or burst scale congestion. Cell scale buffering (delays generally less than a millisecond) is ideal for real-time communications, but only achieves high multiplexer utilization when the source peak rate is a small fraction of link rate. For applications where large delays are acceptable (more than 200 ms, say) burst scale buffering can improve utilization, but network delay guarantees then rely on the use of queueing disciplines like weighted fair queueing which guarantee an individual service rate. An alternative is to shape the coder output to reduce its peak rate before offering it to a network equipped for cell scale congestion only (small FIFO buffers). The SVBR algorithm renders the shaping delay predictable and determines the size of the required receiver playout buffer.

SVBR removes the unpredictability of open-loop VBR coding, but the choice of a sufficiently large burst tolerance parameter b still allows considerable rate variability up to GoP scale. Only potentially damaging scene scale variations are eliminated. Compared to CBR coding, SVBR has the considerable advantage of eliminating the coding delay associated with a smoothing buffer. The quality of SVBR is also higher than that of CBR due to the possibility of rate variations at GoP scale.

The performance claimed for SVBR coding remains loosely theoretical in the absence of subjective tests. These tests are planned, and will allow the tuning of parameter values for optimal performance.

REFERENCES

[1] B. Bensaou, B. Guibert, J. Roberts, and A. Simonian, "Performance of an ATM multiplexer in the fluid approximation using the Benes approach," *Ann. Oper. Res.*, vol. 49, pp. 137–160, 1994.

[2] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger, "Long-range dependence in variable bit rate video traffic," *IEEE Trans. Commun.*, vol. 43, pp. 985–992, 1995.

[3] COST 242 Management Committee, "Methods for the performance evaluation and design of broadband multiservice networks," The COST 242 Final Rep., June 1996.

[4] R. Coelho and S. Tohme, "Video coding mechanism to predict video traffic in ATM network," in *Proc. IEEE GLOBECOM'93*, pp. 447–451.

[5] R. L. Cruz, "A calculus for network delay, Part I: Network elements in isolation," *IEEE Trans. Inform. Theory*, vol. 37, pp. 114–131, Jan. 1991.

[6] A. Elwalid, D. Heyman, T. V. Lakshman, and A. Weiss, and D. Mitra, "Fundamental bounds and approximations for ATM multiplexers with application to video teleconferencing," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 1004–1016, Aug. 1995.

[7] A. Elwalid, D. Mitra, and R. H. Wentworth, "A new approach to allocating buffers and bandwidth to heterogeneous regulated traffic in an ATM node," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 1115–1128, Aug. 1995.

[8] M. R. Frater, P. Tan, and J. F. Arnold, "Variable bit rate video traffic on the broadband ISDN: Modeling and verification," in *Proc. ITC-14*, Elsevier Science Publisher B.V., North-Holland, 1993, pp. 1351–1360.

[9] D. Le Gall, "MPEG: A video compression standard for multimedia applications," *Commun. ACM*, vol. 4, pp. 305–313, Apr. 1991.

[10] M. W. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic," in *Proc. SigComm.*, ACM, Sept. 1994.

[11] M. Hamdi, "Contrôle de trafic pour source à débit variable dans les réseaux ATM," Ph.D. dissertation, Rep. Télécom Bretagne/Univ. Rennes 1, Nov. 1996.

[12] M. Hamdi and J. W. Roberts, "QoS guaranty for shaped bit rate video connections in broadband networks," in *Proc. Int. Conf. Multimedia Networking, MmNet'95*, Aizu-Wakamatsu, Japan, Sept. 1995.

[13] M. Hamdi, J. W. Roberts, and P. Rolin, "GoP scale rate shaping for MPEG transmission in the B-ISDN," in *Proc. Symp. Multimedia Commun. Video Coding*, New York, NY, Oct. 1995.

[14] M. Hamdi and J. W. Roberts, "Burstiness bounds based multiplexing schemes for VBR video connections in the B-ISDN," in *Proc. Int. Zurich Seminar Digital Commun.*, Zurich, Feb. 1996.

[15] H. Heeke, "A traffic control algorithm for ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, pp. 182–189, June 1993.

[16] ———, "Statistical multiplexing gain for variable bit rate video codecs in ATM networks," *Int. J. Digital Analog Commun. Syst.*, vol. 4, pp. 261–268, 1991.

[17] D. P. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, pp. 49–59, Mar. 1992.

[18] *Coded Representation of Picture and Audio Information*, ISO-IEC/JTC1/SC29/WG11, MPEG Test Model 2, July 1992.

[19] *Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to 1.5 Mbits/s*, ISO-IEC/JTC1/SC29/WG11, DIS11172-1, Mar. 1992.

[20] *Generic Coding of Moving Pictures and Associated Audio: Systems*, ISO-IEC/JTC1/SC29/WG11, Recommendation ITU-T H.222.0, ISO/IEC 13818-1, Nov. 1994.

[21] *Generic Coding of Moving Pictures and Associated Audio: Video*, ISO-IEC/JTC1/SC29/WG11, Recommendation ITU-T H.262, ISO/IEC 13818-2, Nov. 1994.

[22] *Traffic Control and Resource Management in B-ISDN*, ITU-T, I.371 Recommandation, 1992.

[23] H. Kroner, "Statistical multiplexing of sporadic sources—Exact and approximate performance analysis," in *Proc. ITC-13*, Copenhagen, Elsevier Science Publisher B.V., North-Holland, 1991.

[24] D. S. Lee, B. Melamed, A. R. Reibman, and B. Sengupta, "TES modeling for analysis of a video multiplexer," *Perf. Eval.*, no. 16, pp. 21–34, 1992.
[25] D. M. Lucantoni, M. F. Neuts, and A. R. Reibman, "Methods for performance evaluation of VBR video traffic models," *IEEE/ACM Trans. Networking*, vol. 2, pp. 176–180, Apr. 1994.
[26] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J. D. Robbins, "Performance models of statistical multiplexing in packet video communications," *IEEE Trans. Commun.*, vol. 36, pp. 834–844, July 1988.
[27] I. Norros. "A storage system with self-similar input," *Queueing Syst., Theory and Appl.*, vol. 16, pp. 387–396, 1994.
[28] I. Norros, J. Roberts, A. Simonian, and J. Virtamo, "The superposition of variable bit rate sources in an ATM multiplexer," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 378–387, Apr. 1991.
[29] M. Nomura, T. Fujii, and N. Ohta, "Basic characteristics of variable rate video coding in ATM environment," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 752–760, June 1989.
[30] P. Pancha and M. El Zarki, "Leaky bucket access control for VBR MPEG video," in *Proc. IEEE INFOCOM'95*, Boston, MA, Apr. 1995.
[31] M. R. Pickering and J. F. Arnold, "A perceptually efficient VBR rate control algorithm," *IEEE Trans. Image Processing*, vol. 3, pp. 527–532, Sept. 1994.
[32] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in intergated services networks—The single node case," in *Proc. IEEE INFOCOM*, 1992, pp. 914–924.
[33] A. R. Reibman and B. G. Haskell, "Constraints on variable bit rate video for ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, pp. 361–372, Dec. 1992.
[34] J. W. Roberts, "Virtual spacing for flexible traffic control," *Int. J. Commun. Syst.*, vol. 7, pp 307–318, Dec. 1994.
[35] R. Steinmetz and K. Nahestedt, *Multimedia Computing, Communications and Applications.* Englewood Cliffs, NJ: Prentice-Hall, 1995.

Maher Hamdi was born in Mahdia, Tunisia, in 1969. He received the Diploma of Designer Engineer in computer sciences from the Ecole Nationale des Sciences de l'Informatique of Tunis with the Presidential Award in 1992, and the Ph.D. degree from the University of Rennes I in 1996.

His doctoral research was carried at the Networks and Multimedia Department of ENST de Bretagne, and was concerned with video traffic control in ATM networks. He is working as a Senior Researcher at the same department, where he is concerned with the design and evaluation of multimedia services on ATM networks.

James W. Roberts received the B.Sc. degree in mathematics from the University of Surrey, U.K., in 1970 and the Doctorate from the Université Pierre et Marie Curie, Paris, France, in 1987.

Since graduating, he has worked in the field of teletraffic engineering and performance evaluation, first with the British Post Office from 1971 to 1975, with the French company SOCOTEL from 1975 to 1977, and with the Centre National d'Etudes des Télécommunications since 1978. His current work concerns the performance evaluation of ATM multiplexing and the design of traffic controls for the broad-band ISDN.

Dr. Roberts is Co-Chairman of the 3rd IEEE ATM'97 Workshop, and was Technical Chairman of the 14th International Teletraffic Congress, ITC 14. He co-edited the Final Report of Action COST 242, *Broadband Network Teletraffic* (Springer-Verlag), and he is a member of the editorial team of the journal *Computer Networks and ISDN Systems*. He is also an Associate Rapporteur for ITU Study Group 2 activities on traffic engineering standards for B-ISDN.

Pierre Rolin was born in Bourges, France, in 1950. He received the engineer degree from INSA Rennes in 1973, the Ph.D. degree in 1976, and the state doctorate in 1986 in mathematics and computer science.

He teaches networks principles, distributed systems, and security issues at Telecom Bretagne, ENSTA. Since 1991, he has been in charge of the Network Department and multimedia services at Telecom Bretagne, and is concerned with research on high-speed networks, security, and multimedia services. Previously, he was with ENSTA (1988–1991) and INRIA (1976–1987) where he developed research on real-time networks. From 1979 to 1982, he worked on the distributed database system SIRIUS delta, and previously worked on computer system performance measurement. He has authored books on LAN's, MAP, ATM, and network basis.