Real-Time VBR Rate Control of MPEG Video Based upon Lexicographic Bit Allocation
(extended abstract)

Dzung T. Hoang
Sony Semiconductor Company of America
3300 Zanker Road, MS SJ3C3
San Jose, CA 95134
dzung@acm.org

Abstract

The MPEG-2 Video Standard describes a bitstream syntax and a decoder model but leaves many details of the encoding process unspecified, such as encoder bit rate control. The standard defines a hypothetical decoder model, called the Video Buffering Verifier, that can operate in either constant-bit-rate or variable-bit-rate modes. We present a low-complexity algorithm for variable-bit-rate control suitable for low-delay, real-time applications. The algorithm is motivated by recent results in lexicographic bit allocation. The basic algorithm switches between constant-quality and constant-bit-rate modes based on changes in the fullness of the decoding buffer in the Video Buffering Verifier. We show how the algorithm can be applied either to produce a desired quality level or to meet a global bit budget. Simulation results show that the algorithm compares favorably to the optimal lexicographic algorithm.

1 Introduction

We present a new low-complexity VBR rate control algorithm for MPEG-2 video coding based upon recent results in bit allocation under a lexicographic framework [1]. The framework assumes knowledge of rate-distortion (R-D) characteristics of the input picture and requires offline processing with three computational passes over the input video sequence. In the first pass, the R-D characteristics of each picture are measured and modeled. In the second pass, an optimal global bit allocation is computed. In the third pass, the computed bit allocation is used to encode the video sequence. With the new real-time rate control algorithm, we skip the first two passes and perform bit allocation "on-the-fly" in a single encoding pass, while maintaining some of the desirable properties of the optimal algorithm.

2 Overview of Lexicographic Bit Allocation

The lexicographic bit-allocation framework of [1] contains three components: 1) a lexicographic optimality criterion, 2) a set of buffering constraints, and 3) R-D modeling of input pictures. Analysis of CBR and VBR bit-allocation problems under this framework yields necessary and sufficient conditions for optimality that are simple and intuitive. We now review the underlying assumptions of the framework and summarize the optimality results.

2.1 Lexicographic Optimality Criterion

At the heart of the lexicographic framework is the lexicographic optimality criterion. The criterion states that an optimal bit allocation results in an encoding whose maximum distortion (measured on some basic unit such as a picture) is the least among all admissible allocations. Furthermore, the second-highest distortion should also be the least among those admissible allocations whose maximum distortion is lowest, and so on. This is a refinement of the well-known minimax criterion. The term "lexicographic" arise from the similarity to the sorting of alphabetic strings. The analogy relies upon associating with each admissible allocation a string that consists of the distortion values sorted in decreasing order, so that the first element of the string is the highest distortion value. The allocation whose corresponding string is first in sorted order (lexicographically minimum) is optimal.

2.2 VBV Buffering Constraints

For budget-constrained bit allocation without buffering constraints, an allocation that results in the lowest constant distortion would be lexicographically optimal. The lexicographic criterion is relevant when buffering constraints do not admit the trivial constant-distortion allocation. Buffering constraints arising from the MPEG-2 VBV are derived in [1]. We now briefly describe the operation of the MPEG-2 VBV.

In CBR mode, the VBV takes in data from the channel at a constant rate. The data is stored in the VBV buffer before being processed by the decoder. At regular display intervals, the decoder retrieves all the bits needed to decode the next picture *instantaneously* from the buffer. Before retrieving bits for the first picture, the decoder waits until the buffer fullness reaches a specified level. The channel rate R, the VBV buffer size B_{V}, and the initial buffer fullness B_1 are parameters encoded in the compressed video bitstream. A compliant MPEG-2 bitstream must be decodable by the VBV model without causing the buffer to overflow or underflow.

The MPEG-2 Video Standard defines two VBR modes of operation. In the first mode, the $\mathbf{vbv\_delay}$ parameter, which indicates to the decoder the VBV buffer fullness at which to start decoding the next picture, is set to a constant value of 0xFFFF. In this mode, the buffer is filled at the peak rate R until the buffer is full. This mode is suited for modeling a decoder connected to a storage device with a peak rate of at least R. In the second VBR mode, bits enter the VBV buffer at a piecewise-constant rate up to the peak rate R. The instantaneous transmission rate for a picture is determined by the coded $\mathbf{vbv\_delay}$ parameter and the number of bits in the encoding of the picture. This mode is suited for modeling a channel with a variable delivery rate. Only the first mode is analyzed in [1], and we restrict our attention to this mode.

Given a bit allocation $\mathbf{s} = \langle s_1, s_2, \ldots, s_N \rangle$ to the N pictures in a video sequence, the fullness of the VBV buffer can be described by the following recurrence:

$$\begin{aligned} B_{\mathrm{f}}(\mathbf{s}, 1) &= B_{\mathrm{V}}, \\ B_{\mathrm{f}}(\mathbf{s}, n+1) &= \min\{B_{\mathrm{V}}, B_{\mathrm{f}}(\mathbf{s}, n) + B_{\mathrm{a}}(n) - s_n\}. \end{aligned} \tag{1}$$

The minimization in (1) prevents the buffer from overflowing. When $B_{\mathrm{f}}(\mathbf{s}, n) + B_{\mathrm{a}}(n) - s_n > B_{\mathrm{V}}$, we say that picture n results in a *virtual overflow*. The buffer may underflow if the encoding rate exceeds the peak rate for an extended period.

2.3 Rate-Distortion Modeling

The lexicographic framework relies on an explicit R-D model for each picture. The model takes the form of a rate-quantization function that relates the number of bits produced with a given quantization setting. Analysis in [1] assumes that the models are monotonic and independent. Although the independence assumption does not hold for MPEG-2 encoders that employ motion-compensated prediction, simulations gave good results when steps were taken to correct for deviations that arise from coding dependencies.

2.4 CBR Optimality Results

Analysis of the buffer-constrained allocation problem with CBR constraints shows that the quantization used in an optimal allocation only changes from one picture to the next when the buffer is either full or empty, increasing when the buffer is full and decreasing when the buffer is empty. This result can be understood by considering the complexity of successive scenes. The relationship between quantization and average rate can be thought of as a measure of the encoding complexity of a scene. More complex scenes require higher quantizers than scenes of lower complexity, for the same average rate. With the available buffering, we can code segments of pictures with an average rate higher or lower than the channel rate R for a limited time. To minimize the maximum distortion we code a locally complex scene at a higher rate than the channel rate. To offset for the increased rate, we code adjacent scenes of lower complexity at less than the channel rate. This is achieved by starting a complex scene with the buffer full and ending with the buffer empty.

2.5 VBR Optimality Results

VBR mode differs from CBR in that there is no buffer overflow condition, thus allowing segments of video to be coded at an average rate lower than the peak for an extended time. In typical applications, the local average rate is lower than the peak for a large portion of the sequence. To equalize quality, most of the video should be coded at a constant quality using less than the peak rate. Complex scenes should be coded at the peak rate.

Analysis of the lexicographic framework supports the above intuition. As in the CBR case, the quantization in an optimal allocation only changes when the buffer is full or empty. In addition, pictures that result in a virtual overflow are coded with a globally minimum quantization that, in a sense, defines the base quality for the sequence. One implication of these results is that for a segment of pictures to be coded at higher than the base quantization, it must either start at the beginning of the sequence or with the buffer full. To return to coding at the base quantization, the buffer must empty. This implies that such a segment uses the maximum capacity of the VBV buffer. Informally, we call the segments that are coded at the base quality “easy” and the others “hard.”

3 Real-Time VBR Rate-Control

In this section, we develop a real-time VBR rate control algorithm based upon the optimal results described above. We first outline the optimal VBR algorithm presented in [1] and then show how to modify this algorithm to work in a single pass.

3.1 Optimal Algorithm

Below is an algorithm that computes an optimal VBR allocation using a total of B_{T} bits.

1. Mark all pictures as *easy*. Let $B_{\mathrm{easy}} \leftarrow B_{\mathrm{T}}$.
2. Allocate B_{easy} bits to easy pictures using a constant quantizer $Q_{\min}$.
3. Simulate the VBV to identify *hard* segments of pictures. A hard segment leads to a buffer underflow when $Q_{\min}$ is used and consists of pictures that follow the most recent virtual overflow up to and including the picture the caused the underflow. After identifying a hard segment, reduce the bit allocation to the picture that caused the underflow to just prevent underflow. Reset the buffer fullness to empty and continue the simulation, adding new pictures to the existing hard segment if the buffer continues to underflow.
4. Let B_{hard} be the total number of bits allocated to the hard pictures. Let $B_{\mathrm{easy}} \leftarrow B_{\mathrm{T}} - B_{\mathrm{hard}}$.
5. If a new hard segment has been identified in Step 3, goto Step 2.
6. Allocate bits to maximal segments of hard pictures using an optimal CBR algorithm.

The algorithm loops when a new hard segment has been detected. Since the number of hard segments is bounded by the length N of the sequence, the algorithm terminates after at most N iterations. With each iteration, $Q_{\min}$ is reduced. Halting the algorithm at the end of an iteration results in an optimal allocation for the bit budget used. In effect, additional iterations only refine the allocation to meet the original bit budget B_{T}.

3.2 Single-Pass Algorithm

To transform the above algorithm to operate in a single pass, we can perform only one iteration and need remove any offline processing. Step 2 and the identification of hard pictures in Step 3 require lookahead and cannot be performed online. Also, the optimal CBR algorithm in Step 6 is an offline algorithm. We propose to remove the offline processing identified above and use an online CBR algorithm for Step 6.

The limitation of a single iteration and the lack of knowledge of rate-quantization characteristics impact the bit budget and the choice of $Q_{\min}$. One approach is not to impose a bit budget and set $Q_{\min}$ based on the desired quality. This would result in an average bit rate that depends on the complexity of the input sequence. Another approach is to monitor and control the average bit rate to meet a specified target. Which approach to use would depend on the particular application. For example, storage applications that are capacity-limited would require the latter approach, while quality-critical applications would favor the former. We first describe an algorithm to deliver video at a desired base quality and later show how to modify it to meet a desired bit budget.

3.2.1 Basic VBR Algorithm

The single-pass VBR algorithm below removes the offline processing identified above and is suitable for low-complexity encoding with a specified base quality.

1. Initially the VBV buffer is set to full. In addition to the VBV buffer size B_{V}, peak input bit rate R, and base quantization scale $Q_{\min}$, the algorithm has three extra parameters: a CBR trigger threshold t_{CBR}, a VBR trigger threshold t_{VBR}, and a target buffer fullness B_{tgt}, with $B_{\mathrm{V}} > t_{\mathrm{VBR}} \geq t_{\mathrm{CBR}} > B_{\mathrm{tgt}} > 0$.
2. The encoder simulates operation of the VBV and keeps track of the VBV buffer fullness B_{f}. The encoder operates in VBR mode and encodes the input pictures using $Q_{\min}$ until $B_{\mathrm{f}} \leq t_{\mathrm{CBR}}$. With this event, the encoder switches to CBR mode and allocates bits to the next K pictures so that the target fullness B_{tgt} would be reached after those pictures have been encoded. The parameter K may correspond to the number of pictures in a small number of GOPs, for example. In CBR mode, the peak rate R is used as the target rate.
3. A CBR rate-control algorithm, such as the TM5 algorithm, can be used in CBR mode with two modifications: the perceptually-adjusted quantization scale used to code any picture cannot be lower than $Q_{\min}$, and the VBV buffer is to be operated near the target fullness B_{tgt}.
4. The encoder switches from CBR mode to VBR mode when $B_{\mathrm{f}} > t_{\mathrm{VBR}}$. When this occurs the base quantizer $Q_{\min}$ is again used.

The single-pass VBR algorithm does not attempt to model the complexity of the coded pictures. Instead, it *reacts* to changes in complexity of pictures that are manifested as changes in the VBV buffer level. One implication of this reactive nature is a delay in switching between VBR and CBR modes compared to the optimal algorithm. Whereas the optimal algorithm switches to CBR mode when the buffer is full, the single-pass algorithm must wait until the buffer reaches t_{CBR} before switching. This leaves less buffering for the CBR-coded pictures. Also, setting $B_{\mathrm{tgt}} > 0$ further reduces the buffering for hard segments. Another difference is that for segments with occasional spikes in complexity, the single-pass algorithm may temporarily switch to CBR coding when the optimal algorithm would continue to operate in VBR mode. The switching from CBR mode to VBR mode is less problematic. Because the CBR algorithm in Step 3 enforces a minimum quantization of $Q_{\min}$, the switching to VBR mode can actually take place before the buffer reaches t_{VBR}. In fact, it is this enforcement of the minimum quantization that enables the buffer to fill up as the complexity of coded pictures decreases.

3.2.2 VBR Algorithm with Bit Budget

The basic single-pass VBR algorithm above does not attempt to control the coding rate to meet a specified bit budget. In a sense, meeting a total bit budget is a long-term process in which a target rate is specified over a long interval, whereas the basic VBR algorithm above can be viewed as a short-term process in which coding decisions are made in response to local variations in buffer state. This is similar to the approach taken in [2]. Enforcing a bit budget is akin to specifying a constant bit rate with the rate averaged across the entire sequence, or a large portion of the sequence. In this context, a budget-constrained VBR algorithm can be viewed as performing the function of a CBR algorithm in which the encoder buffer is sized large enough to sufficiently smooth the variation in coding rate across a large portion of the sequence. In a practical VBR encoder, the size of the encoder buffer is fixed and limited. However, we can use this intuition to design a VBR rate control algorithm that incorporates a bit-budget constraint.

In the TM5 encoder model, a simple buffer-feedback mechanism is used to perform CBR rate control. In this scheme, a base quantizer scale is computed as a function of the fullness of a "virtual" encoder buffer that empties at a constant rate. Denoting the fullness of the virtual buffer as B_{v}, the buffer-feedback function takes the form $Q = clip\left(31\frac{B_{\mathrm{v}}}{r}\right)$, where r is a normalization factor and $clip\left(\cdot\right)$ clips the quantization scale to the range of $[1, 31]$. The base quantizer scale is then modulated with an *activity factor* that attempts to compensate for the difference in visibility of quantization errors among blocks with different levels of spatial detail.

For simulation, we use the TM5 buffer-feedback mechanism to perform the long-term rate control. However, instead of controlling the quantizer scale directly, we control $Q_{\min}$ with the feedback function: $Q = clip\left(31\frac{B_{\mathrm{v}}}{r}\right)$. The size of virtual buffer determines how quickly $Q_{\min}$ can change and also affects the accuracy of the rate control. Conceptually, the size of the virtual buffer need not be constant. For example, to come close to the specified bit budget, we can reduce the size of the buffer gradually to constrain the variance in rate near the end of the sequence. In the simulation results presented below, the virtual buffer size is held constant. The constant emptying rate of the virtual buffer is computed as the total bit budget divided by the number of pictures in the sequence.

4 Simulation Results

We implemented the VBR rate control algorithms using the ISO MPEG-2 software encoder [3] and provide simulation results with the same 3,660 frame sequence used in [1]. As in [1], we use the TM5 adaptive quantization scheme to provide a mapping between quantization and actual quantization scale. The quantization scale is modulated by an *activity factor* that is computed from the spatial variance of the luminance blocks within a macroblock. In TM5, the actual quantization scale used for coding a particular macroblock is determined from an initially computed reference quantization scale, a feedback factor that is dependent on the state of a virtual encoding buffer, and the normalized activity factor for that block. In reporting results, we define the quantization for a picture as the average of the product of the reference quantization scale and the buffer-feedback factor.

We report results here for the following four algorithms: 1) optimal lexicographic VBR, 2) TM5 CBR, 3) single-pass VBR without long-term rate control, and 4) single-pass VBR with long-term rate control. The simulation results for the lexicographic VBR algorithm are taken directly from [1]. To speed up the simulations of the other algorithms, we use approximate motion estimation. This results in a small decrease in the average PSNR of the TM5 results presented here compared to those in [1]. However, approximate motion estimation shows negligible effect on the behavior over time of PSNR, quantizer scale, and VBV buffer fullness that are due to the rate control.

The input sequence consists of 3,660 interlaced frames sampled at 720×480 resolution and 29.97 frames/sec. The sequence contains scenes of varying complexity. The clip starts with several scenes with low motion and low spatial complexity with a variety of transition effects. The clip then cycles through several action-filled scenes including galloping horses,

Table 1: Results of encoding simulations.

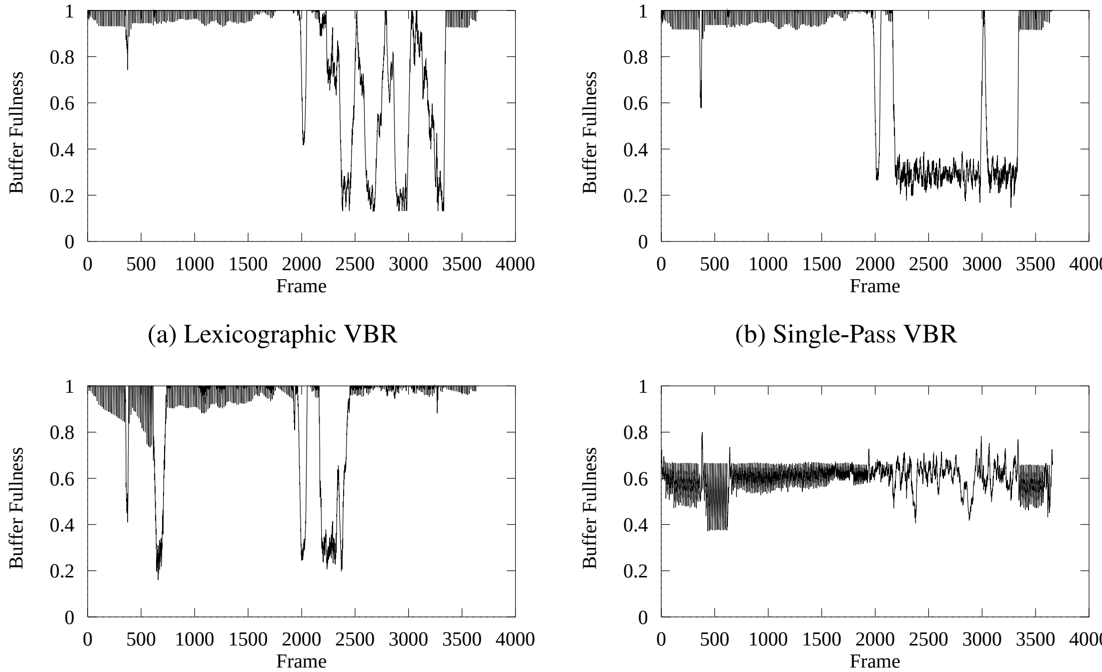
Rate Control Algorithm	Avg. PSNR (dB)	Std. dev. of PSNR	Avg. Nom. Q	Std. dev. of Nom. Q	Max. Nom. Q	Min. Nom. Q
Lexicographic VBR	33.09	2.54	11.74	1.97	16.84	9.02
Single-Pass VBR	32.76	3.09	12.87	4.20	33.59	10.50
Controlled VBR	32.93	4.70	14.59	8.40	29.64	2.14
TM5 CBR	33.28	5.79	16.10	12.81	53.71	1.96

a rotating skydiver, a bicycle race, and highlights from a professional basketball game. For all simulations, we used the GOP pattern `BBIBBPBBPBBPBBP` and a VBV size of 1,835,008 bits. For the VBR algorithms, we specified an average rate of 3.0 Mbps and a peak rate of 4.5 Mbps. For the TM5 CBR algorithm, we specified a constant rate of 3.0 Mbps. For both single-pass VBR algorithms, the initial base quantization parameters were adjusted to give roughly the same encoded file size as TM5 and the following thresholds were used: $t_{\mathrm{VBR}} = 0.85B_{\mathrm{V}}$, $t_{\mathrm{CBR}} = 0.7B_{\mathrm{V}}$ and $B_{\mathrm{tgt}} = 0.3B_{\mathrm{V}}$. For the single-pass algorithm with long-term rate control, the size of the virtual encoder buffer was set to $20B_{\mathrm{V}}$.

A summary of the simulation results is listed in Table 1. The buffer fullness, quantization Q, and PSNR are plotted in Figures 1, 2, and 3, respectively. The results show that the VBR algorithms produce less variance in PSNR and quantization compared to TM5 CBR. However, the average PSNR is actually better for TM5 CBR than for the VBR algorithms. We observe that TM5 gives better PSNR for scenes in the first half of the sequence where the the VBR algorithms operate in constant-quality mode and worse PSNR on for complex scenes in the second half. Since the majority of the pictures operate in constant-quality mode, the average PSNR is biased towards TM5. Visually however, the VBR encodings have more even quality and better scene transitions. Between the optimal VBR algorithm and the single-pass algorithm without long-term rate control, the major differences can be attributed to the use of online versus offline CBR algorithms. The single-pass VBR algorithm with long-term rate control shows some similarity with TM5 CBR in long-term variations in quantization, but with smoother local variations in quantization and PSNR.

5 Related Work

In [4], Reibman and Haskell study encoder rate constraints in the context of an ATM network with a leaky-bucket policing function and propose a coding system in which the selections of channel rate and encoder rate are performed jointly. The proposed encoder rate control is based on the Reference Model 8 (RM8) simulation encoder [5]. In order to prevent the use of progressively smaller quantizer scales for low-complexity segments of video, a minimum quantizer scale is enforced, assuming that the user has selected a maximum desired quality setting. This modification is motivated to save some rate for future peaks. The modified RM8 encoder rate control is similar to that specified in Step 3 of the single-pass VBR algorithm of Section 3.2.1.



(a) Lexicographic VBR

(b) Single-Pass VBR

(c) Controlled VBR

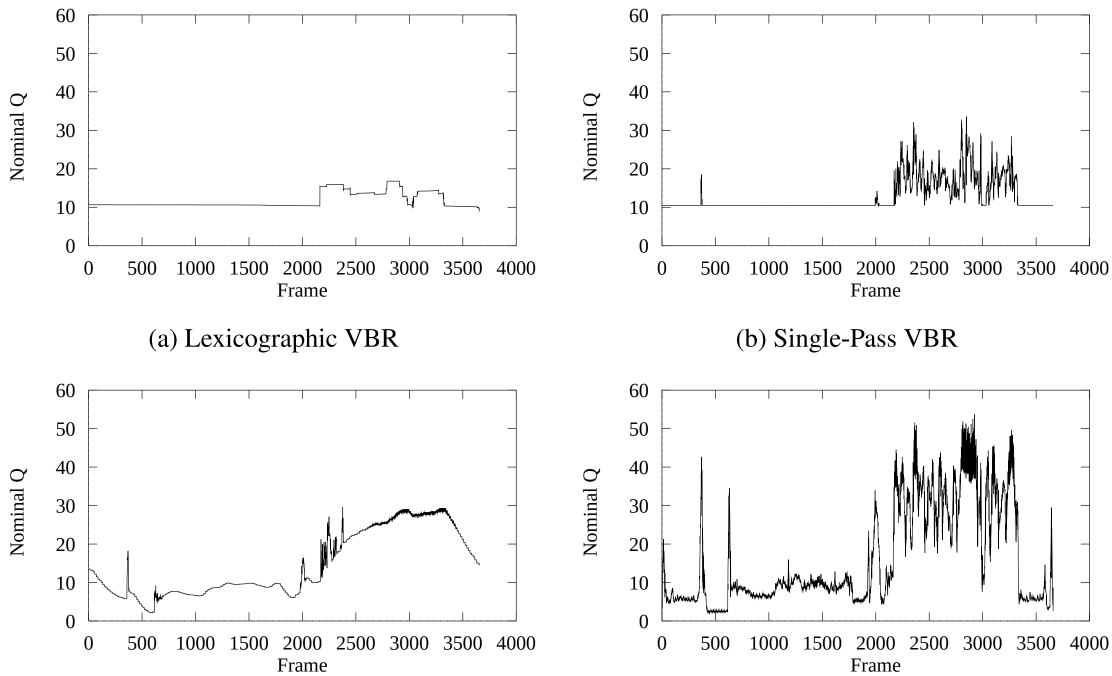
(d) TM5 CBR

Figure 1: Evolution of normalized buffer fullness for a) lexicographically optimal VBR, b) single-pass VBR without long-term rate control, c) single-pass VBR with long-term rate control, and d) TM5 CBR.

The leaky-bucket policing mechanism ensures that the channel rate cannot be sustained above a specified average rate for a period of time related to the size of the bucket. Such constraints are not considered in Equation (1). The "Greedy Leaky-Bucket Rate-Control Algorithm" of [4] selects the highest allowed transmission rate in order to empty the encoder buffer as fast as possible. In the absence of leaky-bucket constraints, this policy corresponds to the MPEG-2 VBR mode described by Equation 1, where the decoder is filled as quickly as possible at the maximum transmission rate.

If we disregard the leaky-bucket constraints, the similarities and differences between the encoder rate control algorithm in [4] and the single-pass VBR algorithm of Section 3.2.1 become more evident. Since RM8 uses a feedback rate control mechanism in which the quantizer scale is determined as a monotonically increasing function of the fullness of the encoder buffer, there is a buffer fullness $B^{\mathrm{e}}_{\mathrm{m}in}$ corresponding to a quantizer scale of Q_{min}. When the encoder's buffer fullness falls below $B^{\mathrm{e}}_{\mathrm{m}in}$, the modified RM8 algorithm is essentially operating in VBR mode with constant quantizer scale of Q_{min}. Since the encoder's buffer mirrors the decoder's buffer in the case under consideration, the modified RM8 algorithm corresponds roughly to our single-pass VBR algorithm with $t_{\mathrm{VBR}} = t_{\mathrm{CBR}} = B_{\mathrm{V}} - B^{\mathrm{e}}_{\mathrm{m}in}$. The modified RM8 algorithm does not have a corresponding B_{tgt}.

In [2], Ding also considers joint encoder and channel rate control over ATM networks. The proposed channel rate control basically performs bitstream smoothing, where the chan-



(a) Lexicographic VBR (b) Single-Pass VBR

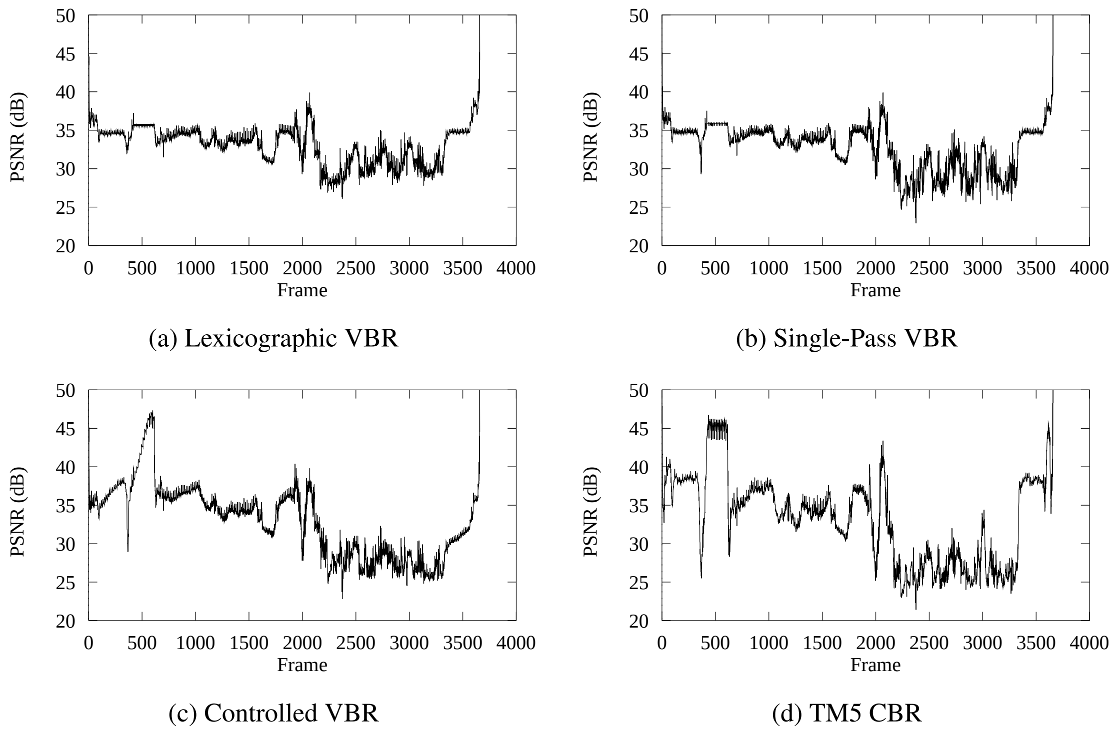
(c) Controlled VBR (d) TM5 CBR

Figure 2: Quantization for a) lexicographically optimal VBR, b) single-pass VBR without long-term rate control, c) single-pass VBR with long-term rate control, and d) TM5 CBR.

nel rate is determined as an average of the encoding rate of some number of past pictures. Encoder rate control is separated into two processes: *encoder instantaneous-rate control* and *encoder sustainable-rate control*. Encoder instantaneous-rate control also uses the concept of a minimum quantization parameter, labeled sequence Q_{s} in [2]. The encoder instantaneous-rate control increases the encoding quantization parameter above Q_{s} only when the upper bound on encoder bit rate is violated. The violation can be determined either by estimating the bit rate of the current picture or by performing a two-pass encoding. Again, if we disregard constraints imposed by ATM policing function, the encoder instantaneous-rate control roughly corresponds to our single-pass VBR algorithm with $t_{\mathrm{VBR}} = t_{\mathrm{CBR}} = E_i$, where E_i is the estimated or computed number of bits to encode the current picture with Q_{s}.

The encoder sustainable-rate control of [2] adjusts the sequence Q_{s} to adapt to the changing local statistics of the video sequence. The sequence Q_{s} is adjusted in discrete increments by monitoring changes in the average fullness of the virtual buffer, which is defined to be the sum of the fullness of the encoder buffer and the leaky bucket.

Considering the previous works described above, our single-pass VBR algorithm, with or without long-term control of $Q_{\min}$, seems especially suited for encoder rate control in an ATM setting when coupled with a suitable channel rate control algorithm. Our algorithm has advantages over these previous algorithms in that better use can be made of the decoder buffer to minimize changes in quantization scale, and consequently in perceived quality.



(a) Lexicographic VBR

(b) Single-Pass VBR

(c) Controlled VBR

(d) TM5 CBR

Figure 3: PSNR for a) lexicographically optimal VBR, b) single-pass VBR without long-term rate control, c) single-pass VBR with long-term rate control, and d) TM5 CBR.

References

[1] D. T. Hoang, E. L. Linzer, and J. S. Vitter, "Lexicographic bit allocation for MPEG video", *J. of Visual Commun. Image Representation*, vol. 8, no. 4, pp. 384–404, Dec. 1997.

[2] W. Ding, "Joint encoder and channel rate control of VBR video over ATM networks", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 266–278, Apr. 1997.

[3] MPEG Software Simulation Group, "MPEG-2 encoder/decoder version 1.2", July 19, 1996, Available URL: `http://www.mpeg.org/MSSG`.

[4] A. R. Reibman and B. G. Haskell, "Constraints on variable bit-rate video for ATM networks", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 4, pp. 361–372, Dec. 1992.

[5] CCITT, "Description of reference model 8 (RM8)", June 1989, Study Group XV—Document 525.