To appear in 1997 IEEE Data Compression Conference

A Lexicographic Framework for MPEG Rate Control
(extended abstract)

Dzung T. Hoang*
Digital Video Systems, Inc.
2710 Walsh Ave.
Santa Clara, CA 95051
dth@dvsystems.com

Elliot L. Linzer†
C-Cube Microsystems
One Water Street, 2nd Floor
White Plains, NY 10601
elliot.linzer@c-cube.com

Jeffrey Scott Vitter‡
Duke University
Box 90129
Durham, NC 27708–0129
jsv@cs.duke.edu

Abstract

We consider the problem of allocating bits among pictures in an MPEG video coder to equalize the visual quality of the coded pictures, while meeting buffer and channel constraints imposed by the MPEG Video Buffering Verifier. We address this problem within a framework that consists of three components: 1) a bit production model for the input pictures, 2) a set of bit-rate constraints imposed by the Video Buffering Verifier, and 3) a novel lexicographic criterion for optimality. Under this framework, we derive simple necessary and sufficient conditions for optimality that lead to efficient algorithms.

1 Introduction

In any lossy coding system, there is an inherent trade-off between the rate of the coded data and the distortion of the reconstructed signal. Often the transmission (storage) medium is bandwidth (capacity) limited. The purpose of rate control is to allocate bits to coding units and to regulate the coding rate to meet the bit-rate constraints imposed by the transmission or storage medium while maintaining an acceptable level of distortion. In this paper, we consider rate control in the context of the MPEG-1 and MPEG-2 standards for video coding.

In addition to specifying a syntax for the encoded bitstream and a mechanism for decoding it, the MPEG standards define a hypothetical decoder, called the Video Buffering Verifier (VBV), that puts quantifiable limits on the variability in the coding rate. The VBV is an integral part of the MPEG standards and every compliant MPEG bitstream must be decodable by the VBV.

Previous work in optimal buffer-constrained rate control generally seeks to minimize a sum-distortion measure, typically mean-squared error (MSE), averaged over

*Work was begun when the author was at IBM T. J. Watson Research Center.

†Work was performed when author was at IBM T. J. Watson Research Center.

‡Support was provided in part by Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grants F49620–92–J–0515 and F49620–94–1–0217, by Army Research Office grant DAAH04–93–G–0076, and by an associate membership in CESDIS.

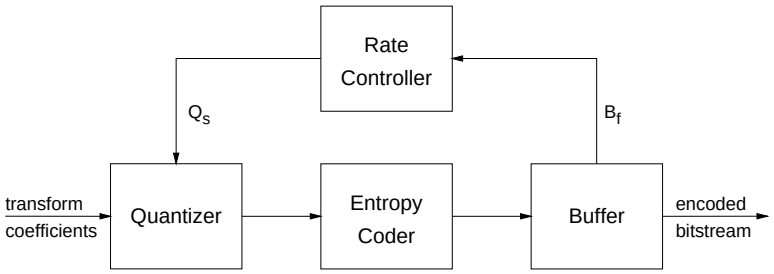


Figure 1: Block diagram of rate control in a typical video coding system.

coding blocks [1]. While this approach leverages the wealth of tools from optimization theory and operations research, it does not guarantee the constancy in quality that is generally desired from a video coding system. For example, a video sequence with a constant or near-constant level of distortion is more desirable than one with lower average distortion but higher variability. This is because human viewers tend to find frequent changes in quality more noticeable and annoying.

We propose a novel optimality criterion that better expresses the desired constancy of quality. The idea is to minimize the maximum block distortion and then minimize the second highest block distortion, and so on. The intuition is that doing so would equalize distortion by limiting peaks in distortion to their minimum. This criterion is referred to as *lexicographic optimality* in the literature [2].

2 Lexicographic Framework

In this paper, we introduce a framework for bit allocation for video coding under VBV constraints and with a total bit budget. The framework consists of three components: 1) a bit-production model, 2) a set of buffer constraints for constant and variable bit rate operation, and 3) a novel lexicographic optimality criterion. We formalize bit allocation as a resource allocation problem with continuous variables and non-linear constraints, to which we apply a global lexicographic optimality criterion.

Analysis of the framework for constant and variable bit rate operation reveals a set of simple and elegant conditions for optimality that admit efficient polynomial-time algorithms. In this paper, we provide a summary of and intuitions behind the analysis. For details of the analysis and algorithms, the reader is referred to [3].

2.1 Perceptual Quantization

In a typical video coder, as shown in Figure 1, the output bit rate can be regulated by adjusting a *quantization scale* Q_{s}. Increasing Q_{s} reduces the output bit rate but also decreases the visual quality of the compressed pictures. Similarly, decreasing Q_{s} increases the output bit rate and increases the picture quality.

Although Q_{s} can be used to control rate and distortion, coding with a constant value of Q_{s} generally does not result in either constant bit rate or constant perceived quality. Both of these factors depend upon the scene content as well. Studies into human visual perception suggest that perceptual distortion is correlated to certain

spatial (and temporal) properties of an image (video sequence) [4, 5]. These studies lead to various quantization techniques, called *perceptual quantization* or *adaptive quantization*, that take into account properties of the Human Visual System [6, 7, 8].

Based upon this body of work, we propose a separation of the quantization scale Q_s into a *nominal quantization* Q and a *perceptual quantization function* $P(I, Q)$ such that $Q_s = P(I, Q)$, where I denotes the block being quantized. The function P is chosen so that if the same nominal quantization Q were used to code two blocks then the blocks would have the same perceptual distortion. In this way, the nominal quantization parameter Q would correspond directly to the perceived distortion and can serve as the object for optimization. We favor a multiplicative model where $P(I, Q) = \alpha_I Q$,[1] such that α_I is large where quantization noise is less noticeable. Our bit rate allocation, however, works with any perceptual quantization function.

The problem of determining $P(I, Q)$ has been studied elsewhere [6, 11]. Here, we address the assignment of Q to each picture to give constant or near-constant quality among pictures while satisfying rate constraints imposed by the channel and decoder. We propose to compute Q at the picture level—that is, we compute one Q for each picture to be coded. Besides decreasing the computation over computing a different Q for each block, this method results in constant perceptual quality within each picture since perceptual quantization is employed at the block level.

2.2 Bit-Production Modeling

We assume that each picture has a bit-production model that relates the picture's nominal quantization Q to the number of coded bits B. This implies that the coding of one picture is independent of any other. This independence holds for an encoding that uses only intraframe (I) pictures, but not for one that uses forward predictive (P) or bidirectionally predictive (B) pictures, for example. In practice, the extent of the dependency is limited to small groups of pictures. We specify Q and B to be non-negative and real-valued. In practice, the quantization scale Q_s and B are positive integers, with $Q_s = \lfloor \alpha_I \cdot Q \rfloor$. However, to facilitate analysis, we assume that there is a continuous function for each picture that maps Q to B.

For a sequence of N pictures, we define N corresponding bit-production models $\{f_1, f_2, \ldots, f_N\}$ that map nominal quantization scale to bits: $b_i = f_i(q_i)$, where $f_i : [0, \infty] \mapsto [l_i, u_i]$, with $0 \leq l_i < u_i$.[2] We require the models to have the following properties:

1. $f_i(0) = u_i$,
2. $f_i(\infty) = l_i$,
3. f_i is continuous and monotonically decreasing.

[1] The MPEG-2 Test Model 5 [9] also uses a multiplicative formulation while an additive formulation is proposed in [10].

[2] We number pictures in encoding order and not in display order.

From these conditions, it follows that f_i is invertible with $q_i = g_i(b_i)$, where $g_i = f_i^{-1}$ and $g_i : [l_i, u_i] \mapsto [0, \infty]$. We note that g_i is also continuous and monotonically decreasing. Although there are specific cases where monotonicity does not hold in practice, it is a generally accepted assumption.

In video coding systems, the number of bits produced for a picture also depends upon a myriad of coding choices besides quantization scale, such as motion compensation and the mode used to code each block. We assume that these choices are made independently of quantization and prior to performing rate control.

2.3 VBV Buffer Constraints

The MPEG standards specify that an encoder should produce a bitstream that can be decoded by a hypothetical decoder referred to as the Video Buffering Verifier (VBV). The VBV consists of a decoder buffer, a decoder, and a display unit. The decoder buffer stores the incoming bits for processing by the decoder. At regular display intervals, the decoder *instantaneously* removes, decodes, and displays the earliest picture in the buffer.

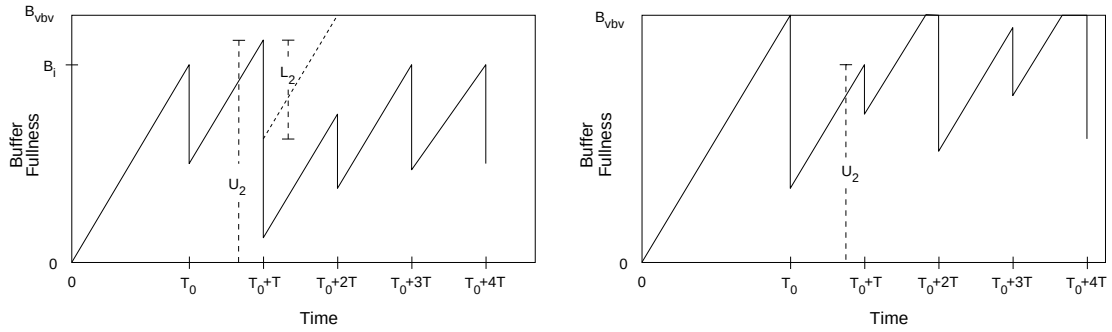
The VBV has two prescribed modes of operation: *constant bit rate* (CBR) and *variable bit rate* (VBR). MPEG-1 supports only CBR mode while MPEG-2 supports both modes. In CBR mode, as illustrated in Figure 2(a), bits enter the decoder buffer at a constant rate $R_{\max}$, specified in the compressed stream. Initially, the buffer is empty and fills for a prespecified time before bits for the first picture are removed and decoded. Afterwards, the buffer continues to fill at the channel rate $R_{\max}$ while the decoder removes bits for coded pictures at regular display intervals. The CBR mode models operation of a decoder connected to a constant-bit-rate channel with a fixed channel delay.

Denoting the buffer fullness just before picture i is removed from the VBV buffer using allocation s by $B_{\mathrm{f}}(s, i)$, we can describe CBR operation of the VBV buffer by the following recurrence.

$$\begin{aligned} B_{\mathrm{f}}(s, 1) &= B_1 \\ B_{\mathrm{f}}(s, i+1) &= B_{\mathrm{f}}(s, i) + B_{\mathrm{a}}(i) - s_i \end{aligned}$$

Here, B_1 is the initial buffer fullness, $B_{\mathrm{a}}(i)$ is the number of bits that enter the buffer after picture i is removed, and s_i is the number of bits allocated to picture i. For progressive (non-interlaced) pictures, $B_{\mathrm{a}}(i) = R_{\max} T$.

In VBR mode, as shown in Figure 2(b), the decoder buffer is initially filled to capacity at the peak rate $R_{\max}$, before the first picture is removed. Thereafter, in each display interval of period T, bits enter the buffer at the peak rate until the buffer is full; bits stop entering the buffer until the next picture has been decoded. When the buffer is full, bits are not discarded, however. Since the decoder buffer stops receiving bits when it is full, a potentially variable number of bits can enter the buffer in each display period. The VBR mode can be thought of as modeling the operation of a decoder connected to a channel or device, a disk drive for example, that can transfer data at a variable rate up to the peak rate $R_{\max}$.



(a) CBR Operation (b) VBR Operation

Figure 2: Sample plots of buffer fullness for CBR and VBR operation.

VBR operation of the VBV buffer can be described with the following recurrence.

$$\begin{aligned} B_{\mathrm{f}}(s,1) &= B_1 \\ B_{\mathrm{f}}(s,i+1) &= \min\{B_{\mathrm{vbv}}, B_{\mathrm{f}}(s,i) + B_{\mathrm{a}}(i) - s_i\} \end{aligned}$$

The effect of the minimization is to prevent the buffer from overflowing.

For proper operation in either CBR or VBR mode, the decoder buffer should not exceed its capacity B_{vbv}.[3] Also, the buffer should contain at least the number of bits needed to decode the next picture at the time it is to be decoded; i.e., $B_{\mathrm{f}}(s,i) \geq s_i$. These requirements impose upper and lower bounds on the number of bits that the encoder can produce for each picture. The upper and lower bounds are indicated in Figure 2 for the second picture as U_2 and L_2, respectively. Note that the lower bound for VBR mode is implicitly zero.

2.4 Lexicographic Optimality

We now formally define the lexicographic optimality criterion. Let S be the set of all legal allocations for a bit-allocation problem P. For an allocation $s \in S$, let $\mathbf{Q}^s = \langle Q_1, Q_2, \ldots, Q_N \rangle$ be the sequence of Q-values to achieve the bit allocation specified by s. Thus $Q_i = g_i(s_i)$, where g_i is as defined in Section 2.2 and s_i is the number of bits allocated to picture i. Ideally, we would like an optimal allocation to use a constant nominal quantization scale. However, this may not be feasible because of buffer constraints. We could consider minimizing an l_k norm of $\mathbf{Q}^s$. However, as discussed earlier, such an approach does not guarantee constant quality where possible and may result in some pictures having extreme values of Q_i. Instead, we would like to minimize the maximum Q_i. Additionally, given that the maximum Q_i is minimized, we want the second largest Q_i to be as small as possible, and so on.

We define a permutation DEC on $\mathbf{Q}^s$ such that for $\mathrm{DEC}(\mathbf{Q}^s) = \langle q_{j_1}, q_{j_2}, \ldots, q_{j_N} \rangle$, we have $q_{j_1} \geq q_{j_2} \geq \cdots \geq q_{j_N}$. Let $\mathrm{rank}(s,k)$ be the k^{th} element of $\mathrm{DEC}(\mathbf{Q}^s)$, i.e.,

[3] By definition, this requirement is always met in VBR mode.

$\text{rank}(s,k) = q_{j_k}$. We define a binary relation $\succ$ on allocations as follows: $s \succ s'$ if and only if $\text{rank}(s,j) = \text{rank}(s',j)$ for $j = 1, 2, \ldots, k-1$ and $\text{rank}(s,k) > \text{rank}(s',k)$ for some $1 \leq k \leq N$. We define $s \prec s'$ if and only if $s' \succ s$. We also define $s \asymp s'$ if and only if $\text{rank}(s,j) = \text{rank}(s',j)$ for all j. Similarly we define $s \succeq s'$ if and only if $s \succ s'$ or $s \asymp s'$, and $s \preceq s'$ if and only if $s \prec s'$ or $s \asymp s'$.

Definition 1 A legal allocation s^* is *lexicographically optimal* if $s^* \preceq s$ for all other legal allocation s.

Lemma 1 (Constant-Q) *Given a bit-allocation problem P of length N, if there exists a legal allocation $s = \langle s_1, s_2, \ldots, s_N \rangle$ such that $g_n(s_n) = q$ for all n, then s is the only lexicographically optimal allocation for P.*

This lemma establishes a desirable property of lexicographic optimality: If a constant-Q allocation is legal, it is the only lexicographically optimal allocation. This meets our objective of obtaining a constant-quality allocation when feasible.

3 CBR Analysis

For a desired level of quality, the number of bits needed to code a picture depends upon the content of the picture as well as its similarity to neighboring pictures. We can associate a coding difficulty with each picture where an "easy" picture would require fewer bits to code than a "hard" picture at the same quality. If we consider a video sequence as being composed of segments of differing coding difficulty, a segment of easy pictures can be coded at a higher quality (lower distortion) than an immediately following segment of hard pictures if we code each segment at a constant bit rate. Since we have an encoder buffer and a decoder buffer, we can vary the bit rate locally to some degree, depending upon the size of the buffers. If we could somehow "move" bits from the easy segment to the hard segment, we would be able to code the easy segment at a lower quality than before and the hard segment at a higher quality, thereby reducing the difference in quality between the two segments. In terms of the decoder buffer, this process corresponds to filling up the buffer during the coding of the easy pictures, which are coded at lower than the average bit rate, so that the hard pictures can be coded at higher than the average bit rate.

Similarly, suppose we have a hard segment followed by an easy segment. We would like to empty the decoder buffer during the coding of the hard pictures to use as many bits as the buffer allows to code the hard pictures at above the average bit rate. This simultaneously leaves room in the buffer to accumulate excess bits resulting from coding the easy pictures below the average bit rate.

Analysis of the lexicographic framework under CBR constraints yields a set of necessary and sufficient conditions that validate the above intuitions. The analysis is summarized in the following theorem.

Theorem 1 *Given a CBR bit-allocation problem P of length N, a legal allocation $s = \langle s_1, s_2, \ldots, s_N \rangle$ is optimal if and only if the following conditions hold. Also, the optimal allocation is unique.*

1. *If* $Q_j > Q_{j+1}$ *for some* $1 \le j < N$, *then* $B_{\rm f}(s, j) = s_j$.
2. *If* $Q_j < Q_{j+1}$ *for some* $1 \le j < N$, *then* $B_{\rm f}(s, j+1) = B_{\rm vbv}$.

In an optimal allocation, if Q changes from one picture to the next, the VBV buffer must be in one of two states: *empty* or *full*. If Q increases from picture j to picture $j+1$, the buffer must be be full immediately before picture $j+1$ is removed from the buffer. If Q decreases, the buffer must be empty immediately after picture j is removed. Furthermore, the theorem guarantees that an allocation that meets the specified "switching" conditions and does not cause the buffer to underflow or overflow is lexicographically optimal. Using these results, we can design a dynamic programming algorithm to compute an optimal allocation.

The basic idea behind dynamic programming is to decompose a given problem in terms of optimal solutions to smaller problems. All we need to do is maintain invariant the conditions stated in Theorem 1 for each subproblem we solve. We do this by constructing optimal bit allocations for pictures 1 to k that results in the VBV buffer being either *full* or *empty* after picture k is decoded. These states are exactly the states where a change in Q may occur. Let Top^k be the optimal allocation for pictures 1 to k that results in the VBV buffer being full. Similarly, let Bot^k be the optimal allocation for pictures 1 to k that results in the VBV buffer being empty. Suppose that we have computed Top^i and Bot^i for $1 \le i \le k$. To compute Top^{k+1}, we search for a legal allocation among $\left\{\emptyset, \mathrm{Top}^1, \ldots, \mathrm{Top}^k, \mathrm{Bot}^1, \ldots, \mathrm{Bot}^k\right\}$, where $\emptyset$ denotes the empty allocation, to which we can concatenate a constant-Q segment to give a legal allocation s such that the switching conditions are met and the buffer ends up full; i.e., $B_{\rm f}(s, k+1) = B_{\rm vbv}$. Similarly, for Bot^{k+1} we search for a previously computed allocation that, when extended by a constant-Q segment, meets the switching conditions and results in the buffer being empty; i.e., $B_{\rm f}(s, k+1) = s_{k+1}$.

Once we have computed Top^{N-1} and Bot^{N-1}, we can compute the optimal allocation for all N pictures in a process similar to the one above for computing Top^k and Bot^k, except that the final allocation would result in a final buffer state that gives the desired target number of bits $B_{\rm tgt}$.

With the above algorithm, $O(N)$ space is used and $O(N^2)$ possible allocations are considered. For most practical bit-production models, computing a constant-Q segment requires time linear in the length of the segment, yielding a total running time of $O(N^3)$. For some useful classes of bit-production models, such as the hyperbolic $f_i(q_i) = \alpha_i/q_i + \beta_i$, we can compute a constant-Q segment in constant amortized time, resulting $O(N^2)$ time complexity.

4 VBR Analysis

In CBR operation, the total number of bits that a CBR stream can use is dictated by the channel bit rate and the buffer size. With VBR operation, the total number of bits has no lower bound, and its upper bound is determined by the peak bit rate and the buffer size. Consequently, VBR is useful and most advantageous over

CBR when the average bit rate needs to be lower than the peak bit rate. This is especially critical in storage applications, where the storage capacity, and not the transfer rate, is the limiting factor. Another important application of VBR video coding is for multiplexing multiple video bitstreams over a CBR channel [12]. In this application, statistical properties of the multiple video sequences allow more VBR bitstreams with a given peak rate $R_{\max}$ to be multiplexed onto the channel than CBR bitstreams coded at a constant rate of $R_{\max}$.

For typical VBR applications, then, bits enter the decoder buffer at an effective rate that is less than the peak during the display interval of many pictures. In interesting cases, there will be segments of pictures that are best coded with an average bit rate that is higher than the peak. This is possible because of the buffering. During the display of these pictures, bits enter the VBV buffer at the peak rate. Since these pictures require more bits to code than the average rate, they are conceptually "harder" to code than the other "easier" pictures.

To equalize quality, easy pictures should be coded at the same base quality. It does not pay to code a hard picture at a quality higher than that of the easy pictures; the bits expended to do so could be better distributed to raise the quality of the easy pictures. Therefore the base quality for the easy pictures should also serve as the maximum quality level for the sequence. Among hard pictures, there may be different levels of coding difficulty. Using intuitions from Section 3, we can draw similar conclusions about the desired buffer behavior when coding the hard pictures.

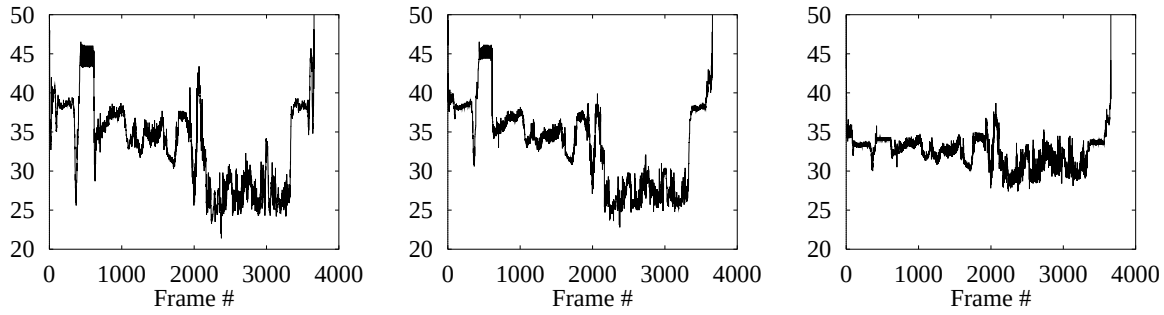
Analysis of the lexicographic framework under VBR constraints yields a set of necessary and sufficient conditions that includes the CBR switching conditions in addition to a condition relating to the base quality for easy pictures. The analysis is summarized in the following theorem.

Theorem 2 *Given a VBR bit-allocation problem P of length N, a legal allocation $s = \langle s_1, s_2, \ldots, s_N \rangle$ is optimal if and only if the following conditions hold. Also, the optimal allocation is unique.*

1. *If $B_{\mathrm{f}}(s,j) + B_{\mathrm{a}}(j) - s_j > B_{\mathrm{vbv}}$ for $1 \le j \le N$, then $Q_j = \min_{1 \le k \le N} \{Q_k\}$.*
2. *If $B_{\mathrm{f}}(s^*, N) > s^*_N$ then $Q_N = \min_{1 \le k \le N} \{Q_k\}$.*
3. *If $Q_j > Q_{j+1}$ for $1 \le j < N$, then $B_{\mathrm{f}}(s,j) = s_j$.*
4. *If $Q_j < Q_{j+1}$ for $1 \le j < N$, then $B_{\mathrm{f}}(s,j+1) = B_{\mathrm{vbv}}$ and $B_{\mathrm{f}}(s,j+1) + B_{\mathrm{a}}(j+1) - s_{j+1} \le B_{\mathrm{vbv}}$.*

In an optimal allocation, a picture that causes the VBV buffer to fill up before the next picture is removed is coded with the globally minimum quantization scale. The last picture is also coded with the minimum quantization scale if it does not cause the buffer to empty upon its removal. In other cases where bits enter the buffer at the peak rate, the CBR switching conditions hold.

Computing an optimal VBR allocation now reduces to identifying easy and hard pictures and finding the minimum quantization scale needed to meet the target bit



(a) TM5 (b) Optimal CBR (c) Optimal VBR

Figure 3: Plots of PSNR (in dB) for different bit allocation algorithms.

budget. Further analysis in [3] shows that this can be done within a finite number of iterations of a simple algorithm that simulates the operation of the VBV to determine hard and easy pictures and invokes the CBR algorithm for hard pictures. Analysis of the VBR algorithm yields the same time and space complexity as the CBR algorithm.

5 Experimental Results

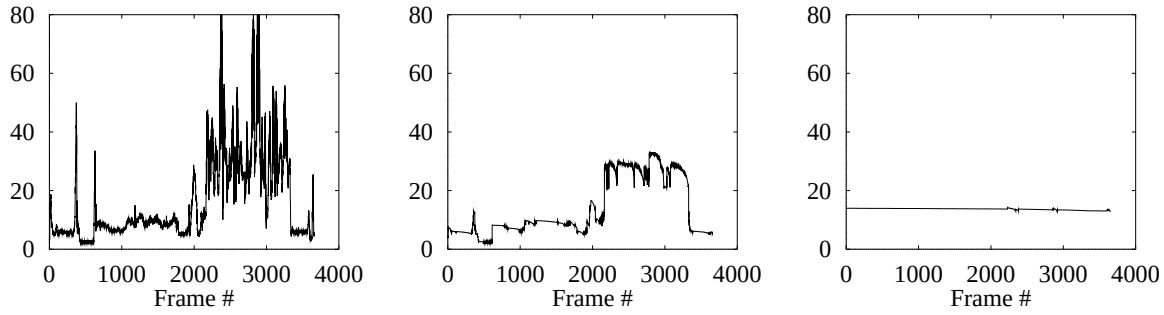
We implemented the CBR and VBR bit allocation algorithms within a software MPEG-2 encoder. A linear spline is used to model the bit production. Multiple encoding passes, each using a fixed nominal quantization, are used to construct the model. In the final encoding pass, the optimal bit allocation algorithms are used to compute the nominal quantization for each picture. To recover from errors in the model, the bit allocation is recomputed after each picture is coded.

To test the effectiveness of lexicographic bit allocation, we performed coding simulations on a two-minute IBM commercial that contains scenes of widely varying complexities. The video sequence is in NTSC CCIR-601 format. For the simulations, we code the sequence in CBR mode at 3 Mbit/s and in VBR mode at 3 Mbit/s average and 6 Mbit/s peak. The VBV buffer size is 1,835,008 bits. The results are summarized in Figures 3 and 4. Results for the optimal CBR allocation show less variance in PSNR and considerably less in nominal quantization when compared to results using the MPEG Test Model 5 [9]. The results for optimal VBR allocation show nearly constant nominal quantization and much less variance in PSNR.

References

[1] A. Ortega, K. Ramchandran, and M. Vetterli, "Optimal trellis-based buffered compression and fast approximations", *IEEE Trans. on Image Processing*, vol. 3, no. 1, pp. 26–40, Jan. 1994.

[2] T. Ibaraki and N. Katoh, *Resource Allocation Problems*, MIT Press, Cambridge, MA, 1988.



(a) TM5 (b) Optimal CBR (c) Optimal VBR

Figure 4: Plots of Nominal Q for different bit allocation algorithms.

[3] D. T. Hoang, E. Linzer, and J. S. Vitter, "Lexicographically optimal rate control for video coding with MPEG buffer constraints", Tech. Rep. CS-1996-02, Duke University, Dept. of Computer Science, 1996.

[4] L. A. Olzak and J. P. Thomas, "Seeing spatial patterns", in *Handbook of Perception and Human Performance*, K. Boff, L. Kaufman, and J. Thomas, Eds. Wiley, 1986.

[5] V. R. Algazi, Y. Kato, M. Miyahara, and K. Kotani, "Comparison of image coding techniques with a picture quality scale", in *Proc. of SPIE, Applications of Digital Image Processing XV*, San Diego, CA, July 1992, pp. 396–405.

[6] E. Viscito and C. Gonzales, "A video compression algorithm with adaptive bit allocation and quantization", in *SPIE Proceedings: Visual Communications and Image Processing*, Nov. 1991, vol. 1605, pp. 58–72.

[7] N. S. Jayant, J. Johnson, and R. Safranek, "Signal compression based on models of human perception", *Proc. of the IEEE*, vol. 81, pp. 1385–1422, Oct. 1993.

[8] T.-Y. Chung, K.-H. Jung, Y.-N. Oh, and D.-H. Shin, "Quantization control for improvement of image quality compatible with MPEG2", *IEEE Trans. on Consumer Electronics*, vol. 40, no. 4, pp. 821–825, Nov. 1994.

[9] ISO-IEC/JTC1/SC29/WG11/N0400, "Test model 5", Apr. 1993, Document AVC-491b, Version 2.

[10] M. R. Pickering and J. F. Arnold, "A perceptually efficient VBR rate control algorithm", *IEEE Trans. on Image Processing*, vol. 3, no. 5, pp. 527–532, Sept. 1994.

[11] K. W. Chun, K. W. Lim, H.D. Cho, and J. B. Ra, "An adaptive perceptual quantization algorithm for video coding", *IEEE Trans. on Consumer Electronics*, vol. 39, no. 3, pp. 555–558, Aug. 1993.

[12] B. G. Haskell and A. R. Reibman, "Multiplexing of variable rate encoded streams", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 4, no. 4, pp. 417–424, Aug. 1994.