CS–1996–02

Lexicographically Optimal Rate Control for Video Coding with MPEG Buffer Constraints

Dzung T. Hoang, Elliot Linzer, and Jeffrey S. Vitter

Department of Computer Science

Duke University

Durham, North Carolina 27708–0129

February 1996

Lexicographically Optimal Rate Control for Video Coding with MPEG Buffer Constraints

Dzung T. Hoang[1]
Dept. of Computer Science
Brown University
Box 1910
Providence, RI 02912-1910

Elliot Linzer
IBM T. J. Watson Research Center
Yortown Heights, NY 10598

Jeffrey S. Vitter[2]
Dept. of Computer Science
Duke University
Box 91029
Durham, NC 27708-0129

February 14, 1996

[1]Supported in part by an NSF Graduate Fellowship and by Air Force Office of Strategic Research grant F49620–92–J–0515

[2]Support was provided in part by Air Force Office of Strategic Research grant F49620–92–J–0515, Army Research Office grant DAAH04–93–G–0076, and by an associate membership in CESDIS.

Abstract

In this paper, we address 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 a Video Buffer Verifier. We address this problem within a framework consisting of 1) a bit production model for the input pictures, 2) a novel lexicographic criterion for optimality, and 3) a set of bit-rate constraints imposed by the transmission channel and Video Buffer Verifier. Using this framework, we analyze the bit allocation problem for both constant and variable bit rate operation. The analysis results in a set of simple necessary and sufficient conditions for optimality that leads to efficient algorithms.

1 Introduction

In any lossy coding system, there is an inherent trade-off between the rate of the transmitted data and the distortion of the reconstructed signal. Often the transmission (storage) medium is bandwidth (capacity) limited. The purpose of rate control is to regulate the coding rate to meet the bit-rate requirements imposed by the transmission or storage medium while maintaining an acceptable level of distortion. We consider rate control in the context of the MPEG-1 and MPEG-2 standards. The standards specify a syntax for the encoded bitstream and a mechanism for decoding it. Furthermore, the standards define a hypothetical decoder called the Video Buffer Verifier, diagrammed in Figure 1, that verifies that an encoded bitstream is decodable with specified limitations on the decoding buffer size B_{vbv} and input bit rate R. In this paper, we develop a framework for rate control for video coding under VBV constraints, with an additional constraint on the total number of bits coded. This framework consists of a bit-production model, an optimality criterion, and a set of buffer constraints for constant and variable bit rate operation. Using this framework, we derive necessary and sufficient conditions for optimality that are used to construct efficient algorithms for optimum rate control.

2 Perceptual Quantization

As shown in Figure 2, the output bit rate of a video coder can be regulated by adjusting the 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. Therefore by varying Q_{S}, we can trace out a rate vs. distortion curve such as that shown in Figure 3. Substantial work has been done on computing rate-distortion functions both analytically (e.g., [1]) and operationally (e.g., [2]).

Although Q_{S} can be used to control rate and distortion, coding with a constant value of Q_{S} does not necessarily result in constant bit rate or constant perceived quality. Both of these factors are dependent on 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). These studies lead to various quantization techniques, called perceptual quantization, that take into account properties of the Human Visual System (HVS) [3]. Based on this body of work, we propose a separation of the quantization parameter into two multiplicative factors, a nominal quantization term, Q, and a perceptual quantization term, Q_{P}, so that $Q_{\mathrm{S}} = Q \cdot Q_{\mathrm{P}}$. The idea is to use Q_{P} to compensate for the spatial characteristics of the block to be quantized so that all blocks quantized with the same value of Q will have the same perceptual quality even though the actual quantizer scale used may be different. In this way, the nominal quantization parameter Q would correspond directly to the perceptual quality, and can serve as the object for optimization.

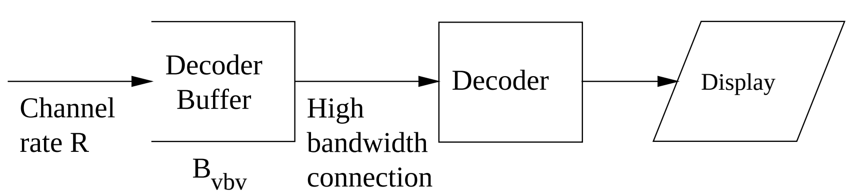

Figure 1: Block diagram of a video buffer verifier.

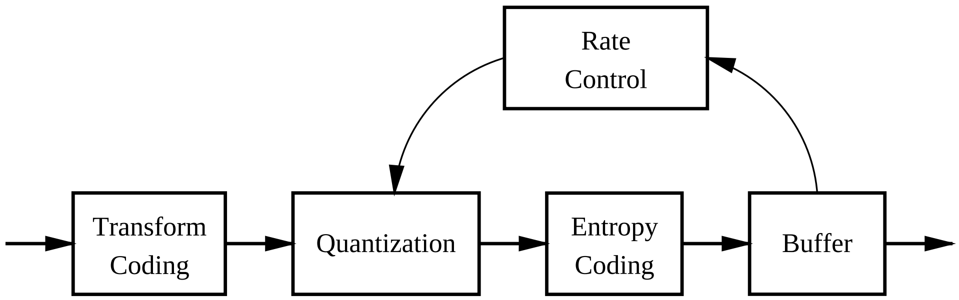

Figure 2: Block diagram of rate control in a typical video coding system.

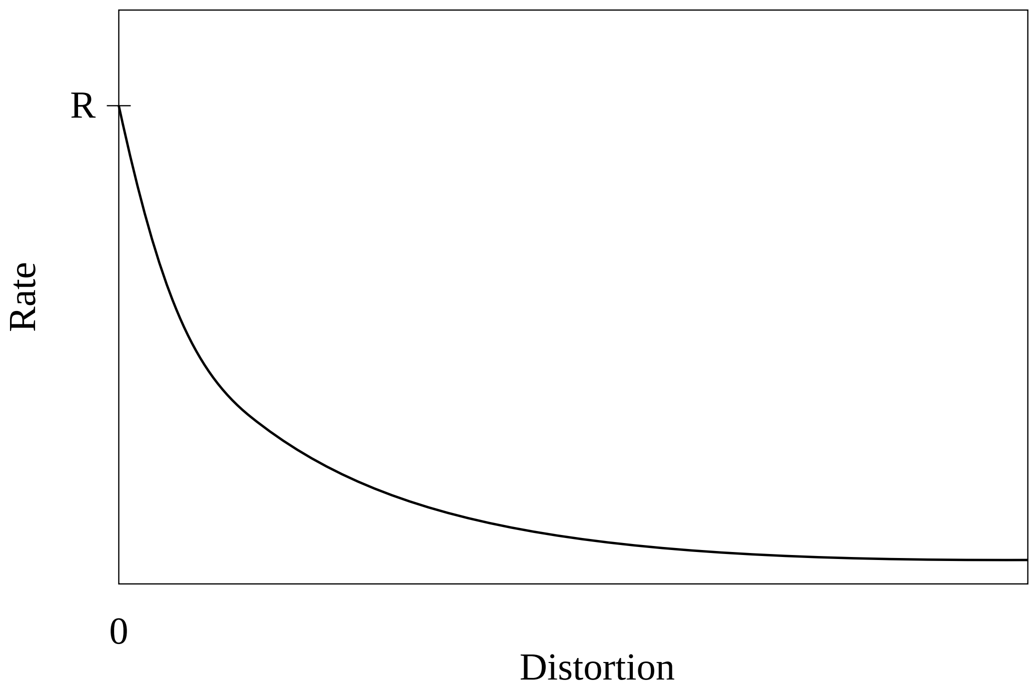

Figure 3: Example of a rate-distortion curve. With no distortion, the source is coded at its entropy R.

(In [4], an additive formulation is proposed.)

The problem of determining Q_{P} has been widely studied elsewhere. Here, we address the computation of Q to give constant or near-constant quality 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 different Q for each block, this method results in constant perceptual quality within each picture since perceptual quantization is employed within each picture. Certainly, our results can be generalized to other coding units.

3 Optimality Criterion

Previous work in rate control generally seek to minimize a distortion measure, typically MSE, averaged over coding blocks (pictures) [5, 6]. While this approach leverages the wealth of tools from optimization theory and operations research, it does not guarantee the constancy of quality that is generally desired in 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 annoying. Moreover, a long sequence will typically contain segments which, even if encoded at a fairly low bit rate, will not contain any disturbing quantization artifacts, so that improving the quality of pictures in those segments is far less important than improving the quality of pictures in segments that are more difficult to encode. To address this issue, we propose a lexicographic optimality criterion that better expresses the desired constancy of quality. The idea is to minimize the maximum (perceptual) distortion of a block (picture). Furthermore, the second highest block distortion should be minimized, and so on. The intuition is that doing so would equalize distortion by limiting peaks in distortion to their minimum. As we will show later, if a constant quality allocation is feasible, then it is necessarily lexicographically optimal.

As discussed in Section 2, we shall use the nominal quantizer Q as the object of optimization.

4 Bit-Production Modeling

For simplicity in modeling, we assume frame-oriented pictures; i.e., we consider only non-interlaced pictures. 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 assumes that the coding of one picture is independent of any other. This is true for an encoding that uses only intra-frame (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 to be a non-negative real number. In practice, the quantization scale, Q_{S}, is a positive integer so that $Q_{\mathrm{S}} = \lfloor Q \cdot Q_{\mathrm{P}} \rfloor$. However, to facilitate analysis, we assume that there is a continuous function that maps Q to B.

For a sequence of N pictures, the bit-production model is specified by N functions, $\{f_1, f_2, \ldots, f_N\}$, that map (nominal) quantization scale to bits: $b_i = f_i(q_i)$, where $f_i : [0, \infty] \mapsto [l_i, u_i]$, with $0 \le l_i < u_i$. (The pictures are numbered in the order in which they are processed by the encoder and decoder, not in the temporal order.) We require the models to have the following properties:

$$
\begin{array}{ll}
1. & f_i(0) = u_i, \\
2. & f_i(\infty) = l_i, \\
3. & f_i \text{ is continuous and monotonically decreasing.}
\end{array}
\tag{1}
$$

From these conditions, we have 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, these cases do not occur often and monotonicity is a generally accepted assumption.

In video coding systems, the number of bits produced for a picture also depends on a myriad of coding choices including motion compensation and the mode used for coding individual blocks. We assume that these choices are made independent of quantization and prior to performing rate control.

5 Video Buffer Verifier

The MPEG-2 standard [7] specifies that an encoder should produce a bitstream that can be decoded by a hypothetical decoder referred to as the Video Buffer Verifier (VBV), as shown in Figure 1. Data can be transferred to the VBV either at a constant or variable rate. In either mode of operation, the number of bits produced by each picture must be controlled so as to satisfy constraints imposed by the operation of the decoder buffer, whose size B_{vbv} is specified in the bitstream by the encoder. The encoder also specifies the maximum transfer rate R into the VBV buffer and the amount of time the decoder should wait before decoding the first picture. In this section, we consider constraints on the number of bits produced in each picture that follow from analysis of the VBV buffer.

5.1 Constant Bit Rate (CBR)

We first examine the mode of operation in which the compressed bitstream is to be delivered at a constant bit rate R. Let T be the amount of time to display one picture. Then RT is the number of bits that enter the decoder buffer in the time required to display one picture.

Definition 1 *Given a sequence of N pictures, an* allocation *$s = \langle s_1, s_2, \ldots, s_N \rangle$ is an N-tuple containing bit allocations for all N pictures, so that s_i is the number of bits allocated to picture i.*

Let B_{vbv} be the size of the decoder buffer. Let $B_{\mathrm{f}}(s, n)$ denote the buffer fullness (the number of bits in the VBV buffer), using allocation s, just *before* the n^{th} picture is removed from the buffer. Let $B_{\mathrm{f}}^*(s, n)$ denote the buffer fullness, using allocation s, just *after* the n^{th} picture is removed. Then

$$B_{\mathrm{f}}^*(s, n) = B_{\mathrm{f}}(s, n) - s_n. \tag{2}$$

Let R be the rate at which bits enter the decoding buffer. Let T_i be the amount of time required to display picture i. Let $B_{\mathrm{a}}(i) = RT_i$. Then $B_{\mathrm{a}}(i)$ is the number of bits that enter the buffer in the time it takes to display picture i.

For constant bit rate (CBR) operation, the state of the VBV buffer is described by the recurrence:

$$\begin{aligned} B_{\mathrm{f}}(s, 1) &= B_1, \\ B_{\mathrm{f}}(s, n+1) &= B_{\mathrm{f}}(s, n) + B_{\mathrm{a}}(n) - s_n, \end{aligned} \tag{3}$$

where B_1 is the initial buffer fullness. Unwinding the recurrence, we can also express (3) as

$$B_{\mathrm{f}}(s, n+1) = B_1 + \sum_{j=1}^{n} B_{\mathrm{a}}(j) - \sum_{j=1}^{n} s_j. \tag{4}$$

To prevent the decoder buffer from overflowing we must have

$$B_{\mathrm{f}}(s, n+1) \le B_{\mathrm{vbv}}. \tag{5}$$

The MPEG-2 standard allows pictures to be skipped in certain applications (when the "low_delay" bit is set to one). We assume that all pictures are coded, in which case all of picture n arrives at the decoder by the time it is removed from the buffer to be displayed; i.e., we must have

$$B_{\mathrm{f}}(s,n) \geq s_n, \tag{6}$$

or equivalently,

$$B_{\mathrm{f}}^{*}(s,n) \geq 0. \tag{7}$$

A violation of this condition is called a buffer *underflow*.

We now have an upper bound and can derive a lower bound for the number of bits that we can use to code picture n. From (3) and (5) we have

$$s_n \geq B_{\mathrm{f}}(s,n) + B_{\mathrm{a}}(n) - B_{\mathrm{vbv}}.$$

Since we cannot produce a negative number of bits, the lower bound on s_n is

$$s_n \geq \max(B_{\mathrm{f}}(s,n) + B_{\mathrm{a}}(n) - B_{\mathrm{vbv}}, 0). \tag{8}$$

In summary, for constant bit rate operation, to pass VBV buffer verification, an allocation s must satisfy the following for all n:

$$\max(B_{\mathrm{f}}(s,n) + B_{\mathrm{a}}(n) - B_{\mathrm{vbv}}, 0) \leq s_n \leq B_{\mathrm{f}}(s,n). \tag{9}$$

An example plot of the evolution of the buffer fullness over time is shown in Figure 4. In this example, the decoder waits T_0 seconds before decoding the first picture, at which time the buffer fullness is B_1. The time to display each frame is a constant T seconds. In the plot, the upper and lower bounds for the number of bits to code picture 2 are shown as U_2 and L_2, respectively.

5.2 Variable Bit Rate (VBR)

We now examine the scenario where the compressed video bitstream is to be delivered at a variable bit rate (VBR) and with a target total number of bits B_{tgt}. Specifically, we adopt the MPEG-2 convention where bits always enter the buffer at the peak rate R_{max} until the buffer is full. Depending on the state of the buffer, bits enter during each display interval at a rate that is effectively variable up to the peak rate R_{max}. For VBR, the maximum number of bits entering the buffer in the time it takes to display picture n is $B_{\mathrm{a}}(n) = R_{\mathrm{max}} T_n$.

For VBR operation, the state of the VBV buffer is described by:

$$\begin{aligned} B_{\mathrm{f}}(s,1) &= B_1, \\ B_{\mathrm{f}}(s,n+1) &= \min(B_{\mathrm{vbv}}, B_{\mathrm{f}}(s,n) + B_{\mathrm{a}}(n) - s_n). \end{aligned} \tag{10}$$

Unlike the CBR case, the decoder buffer is prevented from overflowing by the minimization in (10). Like the CBR case, underflow is possible and to prevent it (6) must hold.

An example plot of the evolution of the buffer fullness over time is shown in Figure 5. In this example, the decoder waits T_0 seconds before decoding the first picture, at which time the buffer fullness is B_1. The time to display each frame is a constant T seconds. As shown in the plot, a variable number of bits enter the buffer during each display interval.

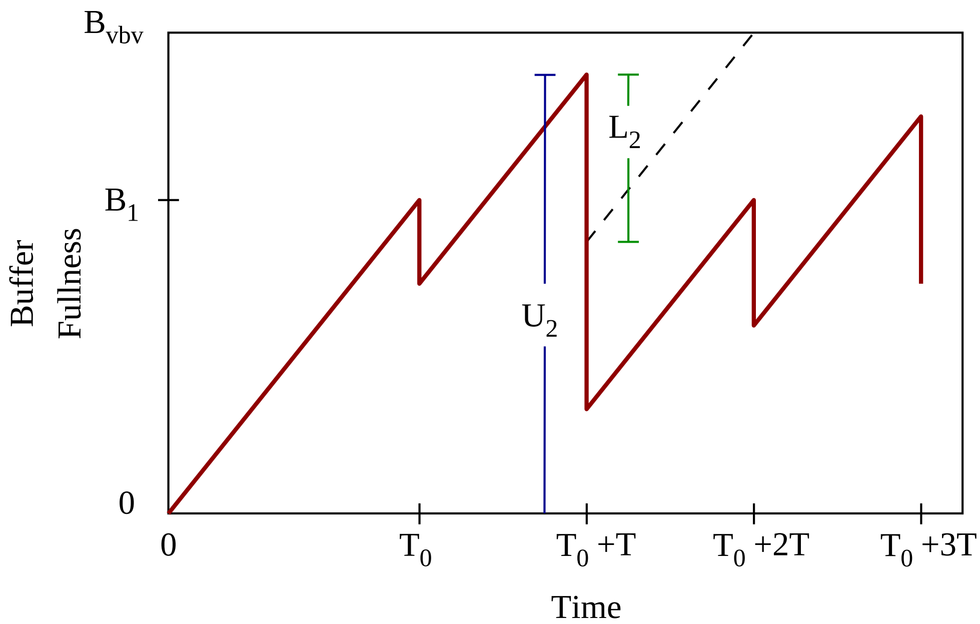

Figure 4: Sample plot of buffer fullness for CBR operation. Bits enter the decoder buffer at a constant rate until time T_0, when the first picture is removed. Successive pictures are removed after a time interval T. Upper and lower bounds on the number of bits that can be produced for the second picture are shown as U_2 and L_2, respectively.

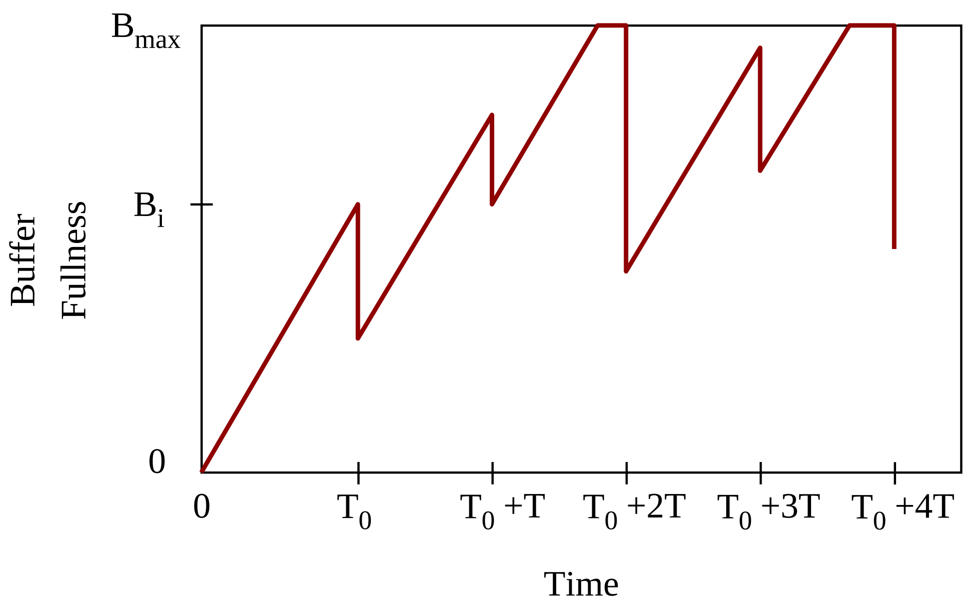

Figure 5: Sample plot of buffer fullness for VBR operation. Bits enter the decoder buffer at a constant rate until time T_0, when the first picture is removed. Successive pictures are removed after a time interval T. When the buffer becomes full, no more bits enter until the next picture is removed.

6 Bit-Allocation Problem

Using the bit-production model and VBV constraints defined above, we now formalize the bit-allocation problem.

Definition 2 *A bit-allocation problem P is specified by a tuple*

$$P = (N, F, B_{\mathrm{tgt}}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}}),$$

where N is the number of pictures; $F = \langle f_1, f_2, \ldots, f_N \rangle$ is a sequence of N functions that model the relationship between the quantization and the number of bits for each picture, as specified in Section 4; B_{tgt} is the target number of bits to code all N pictures; B_{vbv} is the size of the VBV buffer in bits; B_1 is the number of bits initially in the VBV buffer; B_{a} is a function that gives the maximum number of bits that can enter the decoding buffer while each picture is being displayed.

Definition 3 *Given a bit-allocation problem $P = (N, F, B_{\mathrm{tgt}}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}})$, an allocation s is a* legal allocation *if the following conditions hold:*

1. $\displaystyle\sum_{j=1}^{N} s_j = B_{\mathrm{tgt}}$
2. *Equation (6) holds:* $B_{\mathrm{f}}(s, n) \geq s_n$.
3. *For CBR, (8) holds:* $s_n \geq \max(B_{\mathrm{f}}(s, n) + B_{\mathrm{a}}(n) - B_{\mathrm{vbv}}, 0)$.

In order for a CBR bit-allocation problem to have a legal allocation, we must have

$$B_{\mathrm{vbv}} \geq \max_j B_{\mathrm{a}}(j). \tag{11}$$

Also, the buffer fullness at the end of the sequence must be within bounds. For an allocation s, from (2) and (4) we have that $B_{\mathrm{f}}^*(s, N) = B_1 + \sum_{j=1}^{N-1} B_{\mathrm{a}}(j) - B_{\mathrm{tgt}}$. The bounds on $B_{\mathrm{f}}^*(s, N)$ is thus $0 \leq B_{\mathrm{f}}^*(s, N) \leq B_{\mathrm{vbv}}$. This translates to the following CBR bounds on B_{tgt}:

$$B_1 + \sum_{j=1}^{N-1} B_{\mathrm{a}}(j) - B_{\mathrm{vbv}} \leq B_{\mathrm{tgt}} \leq B_1 + \sum_{j=1}^{N-1} B_{\mathrm{a}}(j). \tag{12}$$

A VBR bit-allocation problem does not have a lower bound for the target bit rate B_{tgt} since the VBV does not impose a lower bound on the number of bits produced by each picture. The upper bound for B_{tgt} depends on whether $\max\limits_{1 \leq j \leq N} B_{\mathrm{a}}(j) > B_{\mathrm{vbv}}$. In general, the VBR upper bound on B_{tgt} is

$$B_{\mathrm{tgt}} \leq B_1 + \sum_{j=1}^{N-1} \min(B_{\mathrm{a}}(j), B_{\mathrm{vbv}}). \tag{13}$$

For the rest of this paper, we assume that problems are given so that a legal allocation exists.

7 Lexicographic Optimality

We now formally define the lexicographic optimality criterion. As mentioned in Section 2, the optimality criterion operates on the nominal quantizer Q to be assigned to each picture.

Let S be the set of all legal allocations for a bit-allocation problem P. For an allocation $s \in S$, let $Q^s = \langle Q_1^s, Q_2^s, \ldots, Q_N^s \rangle$ be the values of Q to achieve the bit allocation specified by s. Thus $Q_i^s = g_i(s_i)$, where g_i is as defined in Section 6. Ideally, we want an optimal allocation to have a constant value for Q_i. However, this may not be feasible owing to buffer constraints. We could consider minimizing an l_k norm of Q^s. However, as discussed earlier, such an approach may result in some pictures having extreme values of Q_i. Instead, we would like to minimize the maximum Q_i. Additionally, we want the second largest Q_i to be as small as possible, and so on. This is referred to as *lexicographic optimality* in the literature (e.g., [8]).

We define a permutation DEC on Q^s such that for $\text{DEC}(Q^s) = \langle q_{j_1}, q_{j_2}, \ldots, q_{j_N} \rangle$, we have $q_{j_1} \geq q_{j_2} \geq \cdots \geq q_{j_N}$. Let $\text{rank}(s,k)$ be the k^{th} element of $\text{DEC}(Q^s)$, i.e., $\text{rank}(s,k) = q_{j_k}$. We define a binary relation, $\succ$, on allocations such that $s = \langle s_1, \ldots, s_N \rangle \succ s' = \langle s'_1, \ldots, s'_N \rangle$ 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 a binary relation, $\prec$ such that $s \prec s'$ if and only if $s' \succ s$. We also define $\asymp$ such that $s \asymp s'$ if and only if $\text{rank}(s,j) = \text{rank}(s',j)$ for all j. Similarly we have $s \succeq s'$ 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 4 *An legal allocation $s^* \in S$ is an* optimal allocation *if for all other legal allocation $s \in S$ we have $s^* \preceq s$.*

Lemma 7.1 (Constant Q) *Given a bit-allocation problem $P = (N, F, B_{\text{tgt}}, B_{\text{vbv}}, B_1, B_{\text{a}})$, if there exists a legal allocation s such that $g_i(s_i) = q$ for all i, then s is the only lexicographically optimal allocation.*

Proof: First we prove that s is optimal. Since s is a legal allocation,

$$\sum_{j=1}^{N} s_j = \sum_{j=1}^{N} f_j(q) = B_{\text{tgt}}.$$

Suppose that s is not optimal. Let s' be an optimal allocation. Then $\text{rank}(s',k) < \text{rank}(s,k) = q$ for some k. Then $\text{rank}(s',j) < q$ for $j > k$. Therefore $s'_j > f_j(q)$ for $j \geq k$ since f_j is a decreasing function. Thus

$$\sum_{j=1}^{N} s'_j > \sum_{j=1}^{N} f_j(q) = B_{\text{tgt}}.$$

So s' is not a legal allocation, a contradiction. Therefore s is optimal.

Now we show that s is the only optimal allocation. Let s' be an optimal allocation. Then $\text{rank}(s,j) = \text{rank}(s',j)$ for all j. Therefore $\text{rank}(s',j) = q$ for all j. Thus $s' = s$. □

This lemma establishes a desirable property of the lexicographic optimality criterion: if a constant-Q allocation is legal, it is the only lexicographically optimal allocation. This meets our objective of obtaining a constant-quality allocation (via perceptual quantization and constant-Q) when it is feasible.

8 Constant Bit Rate Allocation

In this section, we analyze the bit-allocation problem under constraints of constant bit rate operation. The results from the theoretical analysis lead to an efficient dynamic programming algorithm for computing the lexicographically optimal solution.

Before proceeding with a formal theoretical treatment, we first present some intuition for the results that follow. 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 a following segment of "hard" pictures if we code each segment with a constant bit rate. Since we have a decoding buffer, we can vary the bit rate to some degree, depending on the size of the buffer. If we can somehow "move" bits from the easy segment to the hard segment, we can code the easy segment at a lower quality and the hard segment at a higher quality, thereby reducing the difference in quality between the two segments. In terms of the decoding buffer, this corresponds to filling up the buffer during the coding of the easy pictures so that the hard pictures can be coded with more than the average bit rate.

Similarly, supposed we have a hard segment followed by an easy segment. We would like to empty the buffer during the coding of the hard pictures so that we use as many bits as the buffer allows to code the hard pictures above the average bit rate, while simultaneously leaving room in the buffer to accumulate excess bits generated by coding the easy pictures below the average bit rate.

This behavior of emptying and filling the buffer is intuitively desirable since this means that we are taking advantage of the full capacity of the buffer. In the following analysis, we will show that such a behavior is indeed exhibited by a lexicographically optimal bit allocation.

8.1 Theory

We first seek to prove necessary conditions for a legal allocation to be lexicographically optimal. To do this, we need the following lemma.

Lemma 8.1 *Suppose that the two allocations s and s' of size N satisfy*

$$s_k = s'_k \text{ if and only if } k \notin \{u, v\}.$$

If

$$\max(g_u(s'_u), g_v(s'_v)) < \max(g_u(s_u), g_v(s_v))$$

we have $s' \prec s$.

The following lemma gives necessary conditions for an optimal allocation.

Lemma 8.2 (CBR Switching) *Given a CBR bit-allocation problem $P = (N, F, B_{\mathrm{tgt}}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}})$, if s^* is an optimal allocation, the following are true:*

1. *If $g_j(s^*_j) > g_{j+1}(s^*_{j+1})$ for some $1 \le j < N$ then $B_{\mathrm{f}}(s^*, j) = s^*_j$. In other words, if the Q used for picture j is greater than that used for picture $j+1$, then it must be that the VBV buffer is as empty as possible without causing an underflow when picture j is removed.*

2. *If $g_j(s^*_j) < g_{j+1}(s^*_{j+1})$ for some $1 \le j < N$ then $B_{\mathrm{f}}(s^*, j+1) = B_{\mathrm{vbv}}$. In other words, if the Q used for picture j is less than that used for picture $j+1$, then it must be that the VBV buffer is as full as possible without overflowing before picture $j+1$ is removed.*

Proof:

Claim 1. We prove by contradiction. Suppose $B_{\mathrm{f}}(s^*, j) \neq s_j^*$. Let $\Delta = B_{\mathrm{f}}(s^*, j) - s_j^*$. Then by (6), $\Delta > 0$. Consider an allocation s' that differs from s^* only for pictures j and $j+1$; that is,

$$s'_k = s^*_k \text{ for } k \in \{1, \ldots, N\} \setminus \{j, j+1\} \text{ and } s'_k \neq s^*_k \text{ for } k \in \{j, j+1\}\,. \tag{14}$$

In order to show a contradiction, we would like to find an assignment to s'_j and s'_{j+1} that would make s' a legal allocation and "better" than s^*. By "better" we mean:

$$g_j(s'_j), g_{j+1}(s'_{j+1}) < g_j(s^*_j). \tag{15}$$

Equivalently, we want

$$s'_j > s^*_j \tag{16}$$

and

$$s'_{j+1} > f_{j+1}(g_j(s^*_j)). \tag{17}$$

To meet the target bit rate, we must have

$$s'_j + s'_{j+1} = s^*_j + s^*_{j+1}. \tag{18}$$

Let $\delta = s'_j - s^*_j$. Then $s^*_{j+1} - s'_{j+1} = \delta$. By (16), we want $\delta > 0$. We want to show that s' is a legal allocation for some value of $\delta > 0$.

To avoid VBV violations, (9) must hold for all pictures under the allocation s'. From (14) and (18), we have

$$B_{\mathrm{f}}(s', k) = B_{\mathrm{f}}(s^*, k) \text{ for } k \neq j+1. \tag{19}$$

Since s^* is a legal allocation, there are no VBV violations for pictures 1 to $j-1$ under s'. Furthermore, if our choice for s'_j does not cause a VBV violation for picture j, then we are assured that there would be no VBV violations in pictures $j+1$ to N.

So we must choose s'_j subject to (9) and (16). Therefore

$$s^*_j < s'_j \leq s^*_j + \Delta. \tag{20}$$

Therefore if $0 < \delta \leq \Delta$, then s' is a legal allocation. We also want for (17) to hold. For this we need

$$\delta < s^*_{j+1} - f_{j+1}(g_j(s^*_j)). \tag{21}$$

Since $g_j(s^*_j) > g_{j+1}(s^*_{j+1})$, $f_{j+1}(g_j(s^*_j)) < s^*_{j+1}$. Therefore $s^*_{j+1} - f_{j+1}(g_j(s^*_j)) > 0$. So for

$$0 < \delta \leq \min(\Delta, s^*_{j+1} - f_{j+1}(g_j(s^*_j))) \tag{22}$$

s' is a legal allocation that meets condition (15). By Lemma 8.1, $s^* \succ s'$ and s^* is not an optimum allocation, a contradiction.

Claim 2. We prove by contradiction. Suppose $B_{\mathrm{f}}(s^*, j+1) \neq B_{\mathrm{vbv}}$. Let $\Delta = B_{\mathrm{vbv}} - B_{\mathrm{f}}(s^*, j+1)$. Then by (5), $\Delta > 0$. Consider an allocation s' that differs from s^* only for pictures j and $j+1$; that is,

$$s'_k = s^*_k \text{ for } k \in \{1, \ldots, N\} \setminus \{j, j+1\}\,. \tag{23}$$

We would like to find an assignment to s'_j and s'_{j+1} that would make s' a legal allocation and "better" than s^*, in order to show a contradiction. By "better" we mean:

$$g_j(s'_j), g_{j+1}(s'_{j+1}) < g_{j+1}(s^*_{j+1}). \tag{24}$$

Equivalently, we want

$$s'_{j+1} > s^*_{j+1} \tag{25}$$

and

$$s'_j > f_j(g_{j+1}(s^*_{j+1})). \tag{26}$$

To meet the target bit rate, we must have

$$s'_j + s'_{j+1} = s^*_j + s^*_{j+1}. \tag{27}$$

Let $\delta = s'_{j+1} - s^*_{j+1}$. Then $s^*_j - s'_j = \delta$. By (25), we want $\delta > 0$. We want to show that s' is a legal allocation for some value of $\delta > 0$.

To avoid VBV violations, (9) must hold for all pictures under the allocation s'. From (23) and (27), we have

$$B_{\mathrm{f}}(s', k) = B_{\mathrm{f}}(s^*, k) \text{ for } k \neq j+1. \tag{28}$$

Since s^* is a legal allocation, there are no VBV violations for pictures 1 to $j-1$ under s'. Furthermore, if our choice for s'_j does not cause a VBV violation, then we are assured that there would be no VBV violations in pictures $j+1$ to N.

So we must choose s'_j subject to (9) and (25).

$$\begin{aligned}
B_{\mathrm{f}}(s', j+1) &= B_{\mathrm{f}}(s', j) + B_{\mathrm{a}}(j) - s'_j \\
B_{\mathrm{f}}(s', j+1) &= B_{\mathrm{f}}(s^*, j) + B_{\mathrm{a}}(j) - s'_j \\
B_{\mathrm{f}}(s^*, j+1) &= B_{\mathrm{f}}(s^*, j) + B_{\mathrm{a}}(j) - s^*_j \\
B_{\mathrm{f}}(s', j+1) &= B_{\mathrm{f}}(s^*, j+1) + s^*_j - s'_j \\
B_{\mathrm{f}}(s', j+1) &= B_{\mathrm{vbv}} - \Delta + \delta \leq B_{\mathrm{vbv}}
\end{aligned}$$

From the above, we require $\delta \leq \Delta$.

Therefore if $0 < \delta \leq \Delta$, then s' is a legal allocation. We also want for (26) to hold. For this we need

$$\delta < s^*_j - f_j(g_{j+1}(s^*_{j+1})). \tag{29}$$

Since $g_{j+1}(s^*_{j+1}) > g_j(s^*_j)$, $f_j(g_{j+1}(s^*_{j+1})) < s^*_j$. Therefore $s^*_j - f_j(g_{j+1}(s^*_{j+1})) > 0$. So for

$$0 < \delta \leq \min(\Delta, s^*_j - f_j(g_{j+1}(s^*_{j+1}))) \tag{30}$$

s' is a legal allocation that meets condition (26). By Lemma 8.1, $s^* \succ s'$ and s^* is not an optimal allocation, a contradiction.

□

Lemma 8.2 gives us a set of necessary *switching* conditions for optimality. It states that an optimal allocation consists of segments of constant Q, with changes in Q occurring only at buffer boundaries. Also, Q must change in a specific manner depending on whether the buffer is full or empty. We observe that in an optimal allocation, the decoder buffer is full before decoding starts on a difficult scene. This policy makes the entire capacity of the decoder buffer available to code the more difficult pictures. On the other hand, before decoding an easy scene, the buffer is emptied in order to provide the most space to accumulate bits when the easy scene uses less than the average bit rate.

We note that Lemma 7.1 follows directly from Lemma 8.2.

The theorem that follows is the main result of this section and shows that the switching conditions are also sufficient for optimality. But first we prove a useful lemma that will be helpful in the proof of the theorem.

Lemma 8.3 *Given bit-allocations s and s', if $s_j \le s'_j$ for $u \le j \le v$ and $B_{\mathrm{f}}(s,u) \ge B_{\mathrm{f}}(s',u)$ then $B_{\mathrm{f}}(s,v+1) = B_{\mathrm{f}}(s',v+1)$ if and only if $B_{\mathrm{f}}(s,u) = B_{\mathrm{f}}(s',u)$ and $s_j = s'_j$ for $u \le j \le v$.*

Proof: We use (4) to express $B_{\mathrm{f}}(s,v+1)$ in terms of $B_{\mathrm{f}}(s,u)$.

$$\begin{aligned}
B_{\mathrm{f}}(s,v+1) &= B_1 + \sum_{j=1}^{v} (B_{\mathrm{a}}(j) - s_j) \\
B_{\mathrm{f}}(s,u) &= B_1 + \sum_{j=1}^{u-1} (B_{\mathrm{a}}(j) - s_j) \\
B_{\mathrm{f}}(s,v+1) &= B_{\mathrm{f}}(s,u) + \sum_{j=u}^{v} (B_{\mathrm{a}}(j) - s_j) \,.
\end{aligned}$$

Similarly,

$$B_{\mathrm{f}}(s',v+1) = B_{\mathrm{f}}(s',u) + \sum_{j=u}^{v} \left(B_{\mathrm{a}}(j) - s'_j\right).$$

First we prove the "if" part. Suppose $B_{\mathrm{f}}(s,u) = B_{\mathrm{f}}(s',u)$ and $s_j = s'_j$ for $u \le j \le v$. Then

$$\begin{aligned}
B_{\mathrm{f}}(s,v+1) &= B_{\mathrm{f}}(s,u) + \sum_{j=u}^{v} (B_{\mathrm{a}}(j) - s_j) \\
&= B_{\mathrm{f}}(s',u) + \sum_{j=u}^{v} \left(B_{\mathrm{a}}(j) - s'_j\right) \\
&= B_{\mathrm{f}}(s',v+1).
\end{aligned}$$

Now we prove the "only if" part. Suppose $B_{\mathrm{f}}(s,v+1) = B_{\mathrm{f}}(s',v+1)$. Then

$$\begin{aligned}
B_{\mathrm{f}}(s,v+1) &= B_{\mathrm{f}}(s',v+1) \\
B_{\mathrm{f}}(s,u) + \sum_{j=u}^{v} (B_{\mathrm{a}}(j) - s_j) &= B_{\mathrm{f}}(s',u) + \sum_{j=u}^{v} \left(B_{\mathrm{a}}(j) - s'_j\right) \\
B_{\mathrm{f}}(s,u) - B_{\mathrm{f}}(s',u) &= \sum_{j=u}^{v} (s_j - s'_j).
\end{aligned}$$

But $B_{\mathrm{f}}(s,u) \ge B_{\mathrm{f}}(s',u)$ and $s_j \le s'_j$ for $u \le j \le v$. Therefore $B_{\mathrm{f}}(s,u) - B_{\mathrm{f}}(s',u) \ge 0$ and $\sum_{j=u}^{v}(s_j - s'_j) \le 0$. But this means that $B_{\mathrm{f}}(s,u) = B_{\mathrm{f}}(s',u)$ and $\sum_{j=u}^{v} s_j = \sum_{j=u}^{v} s'_j$, which is true only if $s_j = s'_j$ for $u \le j \le v$. □

Theorem 8.1 *Given a CBR bit-allocation problem $P = (N, F, B_{\mathrm{tgt}}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}})$, a legal allocation s is optimal if and only if the following conditions hold. Also, the optimal allocation is unique.*

1. *If $g_j(s_j) > g_{j+1}(s_{j+1})$ for some $1 \le j < N$, then $B_{\mathrm{f}}(s,j) = s_j$.*
2. *If $g_j(s_j) < g_{j+1}(s_{j+1})$ for some $1 \le j < N$, then $B_{\mathrm{f}}(s,j+1) = B_{\mathrm{vbv}}$.*

Proof: Lemma 8.2 established these as necessary conditions. Now we need to show that these conditions are also sufficient for optimality and imply uniqueness. Let s be a legal allocation that meets both conditions of the theorem.

Let s^* be an optimal allocation for P. Let $Q_{\max} = \max\limits_{1 \le j \le N} g_j(s_j)$. Consider the segments of consecutive pictures that are assigned the maximum Q by allocation s. Let u be the index of the start of such a segment. There are two cases:

$u = 1$: $B_{\mathrm{f}}(s, u) = B_{\mathrm{f}}(s^*, u) = B_1$.

$u > 1$: Since u is the index of the start of the segment, $g_{u-1}(s_{u-1}) < g_u(s_u)$ which implies that $B_{\mathrm{f}}(s, u) = B_{\mathrm{vbv}}$ by condition 2. Since s^* is a legal allocation, $B_{\mathrm{f}}(s^*, u) \le B_{\mathrm{vbv}} = B_{\mathrm{f}}(s, u)$.

In either case we have

$$B_{\mathrm{f}}(s, u) \ge B_{\mathrm{f}}(s^*, u). \tag{31}$$

Let v be the index of the end of the segment. Since s^* is optimal, $g_j(s_j^*) \le Q_{\max}$ for all j, and thus

$$s_j^* \ge s_j \quad \text{for } u \le j \le v. \tag{32}$$

Therefore

$$B_{\mathrm{f}}(s^*, j) \le B_{\mathrm{f}}(s, j) \quad \text{for } u \le j \le v. \tag{33}$$

There are two cases for v:

$v = N$: $B_{\mathrm{f}}(s, v+1) = B_{\mathrm{f}}(s^*, v+1) = B_1 + \sum_{j=1}^{N-1} B_{\mathrm{a}}(j) - B_{\mathrm{tgt}}$.

$v < N$: Since v is the index of the end of the segment, $g_v(s_v) > g_{v+1}(s_{v+1})$ which implies that

$$B_{\mathrm{f}}(s, v) = s_v \tag{34}$$

by condition 1. Since s^* is a legal allocation, $B_{\mathrm{f}}(s^*, v) \ge s_v^*$. Combine this with (34) and (32) we have

$$B_{\mathrm{f}}(s^*, v) \ge B_{\mathrm{f}}(s, v). \tag{35}$$

Combining (33) and (35), we have that $B_{\mathrm{f}}(s^*, v) = B_{\mathrm{f}}(s, v)$ and $s_v^* = s_v$. As a result, $B_{\mathrm{f}}(s^*, v+1) = B_{\mathrm{f}}(s, v+1)$.

In either case, $B_{\mathrm{f}}(s^*, v+1) = B_{\mathrm{f}}(s, v+1)$. From Lemma 8.3, $B_{\mathrm{f}}(s^*, u) = B_{\mathrm{f}}(s, u)$ and $s_j^* = s_j$ for $u \le j \le v$. As a consequence, $B_{\mathrm{f}}(s, j) = B_{\mathrm{f}}(s^*, j)$ for $u \le j \le v$.

Partition pictures 1 through N into segments, where, according to allocation s, the pictures in a segment use the same value of Q, either the first picture in the segment is picture 1 or the first picture in the segment uses a value of Q different from the previous picture, and either the last picture in the segment is picture N or the last picture in the segment uses a value of Q different from the next picture. Let M be the number of such segments. Order the segments so that segment k uses a value of Q greater than or equal to the value of Q used in segment j if $j < k$, and denote the value of Q used by segment k as $Q_k^{(s)}$.

We will show that s^* uses the same number of bits as s for each picture in segment k, for $1 \le k \le M$. This will establish the conditions in the theorem as necessary and show that the optimal allocation is unique. We will prove this by induction on k, the case of $k = 1$ having already been proven.

Inductive Hypothesis: For all segments of pictures u to v with $u = 1$ or $g_{u-1}(s_{u-1}) \ne g_u(s_u)$, $v = N$ or $g_v(s_v) \ne g_{v+1}(s_{v+1})$, and $g_j(s_j) = Q_k^{(s)}$, we have that $s_j^* = s_j$ and $B_{\mathrm{f}}(s, u) = B_{\mathrm{f}}(s^*, u)$ for $u \le j \le v$.

Assume that the hypothesis is true for $1 \le k < m$. We will show that is is also true for $k = m$. Consider a segment of consecutive pictures that are assigned quantization $Q_m^{(s)}$ by allocation s. Let u be the index of the start of the segment and v the index of the end of the segment.

By the inductive hypothesis, s and s^* use the same values of Q for all pictures for which s uses $Q > Q_m^{(s)}$. Because s^* is optimal, $g_j(s_j^*) \le g_j(s_j) = Q_m^{(s)}$ for $u \le j \le v$, and thus

$$s_j^* \ge s_j \quad \text{for } u \le j \le v. \tag{36}$$

We consider all cases for the segment boundaries. For the left segment boundary there are three cases:

$u = 1$: $B_\mathrm{f}(s^*, u) = B_\mathrm{f}(s, u) = B_1$.

$g_{u-1}(s_{u-1}) > g_u(s_u)$: From the inductive hypothesis, we have $B_\mathrm{f}(s^*, u-1) = B_\mathrm{f}(s, u-1)$ and $s_{u-1}^* = s_{u-1}$. Therefore $B_\mathrm{f}(s^*, u) = B_\mathrm{f}(s, u)$.

$g_{u-1}(s_{u-1}) < g_u(s_u)$: From condition 2, we have $B_\mathrm{f}(s, u) = B_\mathrm{vbv}$. Since s^* is a legal allocation, $B_\mathrm{f}(s^*, u) \le B_\mathrm{vbv} = B_\mathrm{f}(s, u)$.

For all three cases we have

$$B_\mathrm{f}(s^*, u) \le B_\mathrm{f}(s, u). \tag{37}$$

For the right segment boundary there are three cases:

$v = N$: $B_\mathrm{f}(s^*, v+1) = B_\mathrm{f}(s, v+1) = B_1 + \sum_{j=1}^{N-1} B_\mathrm{a}(j) - B_\mathrm{tgt}$.

$g_v(s_v) > g_{v+1}(s_{v+1})$: From condition 1 we have $B_\mathrm{f}(s, v) = s_v$. From (36) and (37) we have

$$B_\mathrm{f}(s^*, j) \le B_\mathrm{f}(s, j) \quad \text{for } u \le j \le v. \tag{38}$$

Since s^* is a legal allocation,

$$B_\mathrm{f}(s^*, v) \ge s_v^* \ge s_v = B_\mathrm{f}(s, v). \tag{39}$$

Combining (38) and (39), we have that $B_\mathrm{f}(s^*, v) = B_\mathrm{f}(s, v)$ and $s_v^* = s_v$. Therefore $B_\mathrm{f}(s^*, v+1) = B_\mathrm{f}(s, v+1)$.

$g_v(s_v) < g_{v+1}(s_{v+1})$: From the inductive hypothesis, we have $B_\mathrm{f}(s^*, v+1) = B_\mathrm{f}(s, v+1)$.

For all three cases we have

$$B_\mathrm{f}(s^*, v+1) = B_\mathrm{f}(s, v+1). \tag{40}$$

From (36), (37), (40), and Lemma 8.3, we have that $s_j^* = s_j$ for $u \le j \le v$. It follows that $B_\mathrm{f}(s, j) = B_\mathrm{f}(s^*, j)$ for $u \le j \le v$.

By induction, we have that $s_j^* = s_j$ for all j, and so $s = s^*$. □

8.2 CBR Allocation Algorithm

Theorem 8.1 is a powerful result. To find the optimal allocation we need only find a legal allocation that meets the switching conditions stated in the theorem. We use the technique of dynamic programming to develop an algorithm that runs in polynomial time and uses linear space.

The idea 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 8.1 for each subproblem we solve. We do this by constructing optimal bit allocations for pictures 1 to k that end up with the VBV buffer in one of two states: *full* or *empty*. 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 end up with the VBV buffer *full*, if such an allocation exists. Similarly, let Bot^k be the optimal allocation for pictures 1 to k that end up with the VBV buffer *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 concatenate a constant-Q segment, resulting in a legal allocation s, that ends with the buffer full (i.e., $B_{\mathrm{f}}(s, k+1) = B_{\mathrm{vbv}}$) and meets the switching conditions. Similarly, for Bot^{k+1} we search for a previously computed allocation that, when extended by constant-Q segment that meets the switching condition, results in the buffer being empty, i.e., $B_{\mathrm{f}}(s, k+1) = s_{k+1}$.

Once we have computed Top^{N-1} and Bot^{N-1}, we can compute the optimal allocation in a process similar to the one above for computing Top^k, except that the constant-Q allocation to be concantenated should give the desired target number of bits B_{tgt}.

When computing Top^k and Bot^k for $1 \le k \le N-1$, we have insured that the conditions of Theorem 8.1 are met. Additionally in the final computation, the conditions are also met. Therefore we end up with a legal allocation that meets the conditions of Theorem 8.1 and is therefore optimal.

We use the concept of a constant-Q segment extensively in the above discussion. We now formalize this concept. First, we define a family of bit-production functions, $F_{i,j}(q)$, that gives the number of bits resulting from allocating a constant value of Q for pictures i to j, inclusive:

$$F_{i,j}(q) = \sum_{i \le k \le j} f_k(q). \tag{41}$$

What we are really interested in, though, is the inverse of $F_{i,j}$. We denote the inverse as $G_{i,j}$ so that $G_{i,j} = F_{i,j}^{-1}$. Then $G_{i,j}(b)$ gives the constant Q that results in b bits being produced by pictures i to j.

If we assume that $G_{i,j}$ can be computed and the constant-Q allocation verified in time linear in the number of pictures involved, then we can show that to compute Top^k and Bot^k takes $O(k^2)$ time. This can be done for a large number of functional forms of the bit-production model f_i. In such cases, to compute an optimal allocation for a sequence of N pictures would take $O(N^3)$ time. If we store pointers for tracing the optimal sequence of concatenations, the algorithm requires $O(N)$ space.

For some special cases of the bit-production model f_i, we can compute $G_{i,j}$ in constant time for each invocation with $O(N)$ preprocessing time and space. An example is $f_i(q) = \alpha_i/q + \beta_i$, where

$$G_{i,j}(b) = \frac{\sum_{i \le k \le j} \alpha_k}{b - \sum_{i \le k \le j} \beta_k}.$$

We can precompute the cummulative sums for α and β in linear time and space and then use these to compute $G_{i,j}$ in constant time. We can also verify a legal allocation in constant amortized time. Other functional forms for f_i with a closed-form solution for $G_{i,j}$ can be found in [9]. In these special cases, to compute an optimal CBR allocation for a sequence of N pictures would require $O(N^2)$ time and $O(N)$ space.

9 Variable Bit Rate Allocation

In this section, we analyze the bit-allocation problem under VBR constraints, as described in Section 5.2. The analysis leads to an efficient iterative algorithm for computing the lexicographically optimal solution.

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 is upper bounded by the peak bit rate and the buffer size. Consequently, VBR is useful and is most advantageous over CBR when the average bit rate is 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.

For typical VBR applications, then, the average bit rate is much lower than the peak. In this case bits enter the decoder buffer at an effective bit rate that is less than the peak during the display interval of many pictures. In interesting cases, there will be some pictures that will be coded with a bit rate that is higher than the peak. This is possible due to the buffering. These pictures are, in a sense, "harder" to code than the other "easy" pictures.

So, in order to equalize quality, the easy pictures should be coded at the same quality. Also, it does not pay to code any of the hard pictures 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 instead. Among the hard pictures, there are different levels of coding difficulty. Using the same intuitions from the CBR case, we draw similar conclusions about the buffer emptying and filling behavior among the hard pictures.

In the following analysis, we show that a lexicographically optimal VBR bit allocation possesses the properties described above. In particular, the hard segments of pictures in a VBR bit allocation behave as if they are allocated in a CBR setting. As a result, the VBR algorithm we present performs multiple passes over the video sequence to determine the quality at which to code the easy pictures and invokes the CBR algorithm to allocate for the hard pictures.

9.1 Theory

The following two lemmas characterize the "easy" pictures in an optimal allocation, that is, the pictures that are coded with the best quality (lowest Q).

Lemma 9.1 *Given a VBR bit-allocation problem $P = (N, F, B_{\mathrm{tgt}}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}})$ and an optimal allocation s^*, if $B_{\mathrm{f}}(s^*, j) + B_{\mathrm{a}}(j) - s^*_j > B_{\mathrm{vbv}}$ for $1 \le j \le N$, then $g_j(s^*_j) = \min_{1 \le k \le N} g_k(s^*_k)$.*

Proof: Let $Q_{\min} = \min_{1 \le k \le N} g_k(s^*_k)$. Let j be an index such that $B_{\mathrm{f}}(s^*, j) + B_{\mathrm{a}}(j) - s^*_j > B_{\mathrm{vbv}}$. Let $\Delta = B_{\mathrm{f}}(s^*, j) + B_{\mathrm{a}}(j) - s^*_j - B_{\mathrm{vbv}}$. Since $B_{\mathrm{f}}(s^*, j) + B_{\mathrm{a}}(j) - s^*_j > B_{\mathrm{vbv}}$, $\Delta > 0$. Suppose that $g_j(s^*_j) > Q_{\min}$. Let u be an index such that $g_u(s^*_u) = Q_{\min}$. Consider an allocation s that differs from s^* only for pictures j and u. We want to assign values to s_j and s_u that would make s a legal allocation with $g_j(s_j), g_u(s_u) < g_j(s^*_j)$, from which $s^* \succ s$, thereby arriving at a contradiction. Let $s_j = s^*_j + \delta$ and $s_u = s^*_u - \delta$ with $\delta > 0$. Then $g_j(s_j) < g_j(s^*_j)$ and $\sum_{k=1}^N s_k = \sum_{k=1}^N s^*_k = B_{\mathrm{tgt}}$. We now need to show that s does not result in a VBV buffer underflow, that is, $B_{\mathrm{f}}(s, k) \ge s_k$ for $1 \le k \le N$. There are two cases to consider: $u < j$ and $u > j$.

$u < j$**:** Since $s_k = s^*_k$ for $k < u$, $B_{\mathrm{f}}(s, k) = B_{\mathrm{f}}(s^*, k)$ for $1 \le k \le u$. Since $s_u < s^*_u$ and $s_k = s^*_k$ for $u < k < j$, $B_{\mathrm{f}}(s, k) \ge B_{\mathrm{f}}(s^*, k)$ for $u < k \le j$. Therefore pictures 1 to $j - 1$ cannot cause any VBV buffer underflows. If we choose $0 < \delta < \Delta$, then $B_{\mathrm{f}}(s, j + 1) = B_{\mathrm{vbv}}$ and picture j also cannot cause a VBV buffer underflow. Since $s_k = s^*_k$ for $k > j$, and $B_{\mathrm{f}}(s, j+1) = B_{\mathrm{f}}(s^*, j+1)$, pictures $j + 1$ to N also cannot cause any VBV buffer underflows.

$u > j$**:** Since $s_k = s^*_k$ for $k < j$, $B_{\mathrm{f}}(s, k) = B_{\mathrm{f}}(s^*, k)$ for $1 \le k \le j$. If we choose $0 < \delta < \Delta$, then $B_{\mathrm{f}}(s, j + 1) = B_{\mathrm{vbv}}$ and picture j also cannot cause a VBV buffer underflow. Since $s_k = s^*_k$ for $j < k < u$, and $B_{\mathrm{f}}(s, j + 1) = B_{\mathrm{f}}(s^*, j + 1)$ (by a suitable choice of δ), pictures $j + 1$ to $u - 1$ also cannot cause any VBV buffer underflows. Since $s_u < s^*_u$, $B_{\mathrm{f}}(s, k) \ge B_{\mathrm{f}}(s^*, k)$ for $k : k \ge u$. Therefore pictures u to N also cannot cause any VBV buffer underflows.

Therefore s is a legal allocation with $g_j(s_j) < g_j(s_j^*)$. We need to guarantee that $g_u(s_u) < g_j(s_j^*)$. Let $\gamma = g_j(s_j^*) - g_u(s_u^*)$. Since $g_j(s_j^*) > g_u(s_u^*)$, we have $\gamma > 0$. Let $\alpha = s_u^* - f_u(g_u(s_u^*) + \gamma/2)$. Since f_u is decreasing and $\gamma > 0$, we have $\alpha > 0$ and

$$\begin{aligned} s_u^* - \alpha &= f_u(g_u(s_u^*) + \gamma/2) \\ g_u(s_u^* - \alpha) &= g_u(s_u^*) + \gamma/2 \\ g_u(s_u^* - \alpha) &< g_u(s_u^*) + \gamma = g_j(s_j^*) \\ g_u(s_u^* - \alpha) &< g_j(s_j^*). \end{aligned}$$

Consider the assignment $\delta = \min(\alpha, \Delta/2)$. There are two cases: $\alpha \leq \Delta/2$ and $\alpha > \Delta/2$.

$\alpha \leq \Delta/2$: We have $\delta = \alpha$ from which $g_u(s_u) = g_u(s_u^* - \delta) = g_u(s_u^* - \alpha) < g_j(s_j^*)$. Since $0 < \delta < \Delta$, the allocation s is legal.

$\alpha > \Delta/2$: We have $\delta = \Delta/2$. Since g_u is decreasing and $\alpha > \Delta/2$, we have $g_u(s_u^* - \Delta/2) < g_u(s_u^* - \alpha)$ and thus $g_u(s_u) = g_u(s_u^* - \delta) = g_u(s_u^* - \Delta/2) < g_j(s_j^*)$. Since $0 < \delta < \Delta$, the allocation s is legal.

Since s is a legal allocation that differs from s^* only for pictures u and j with $g_u(s_u), g_j(s_j) < g_j(s_j^*)$, from Lemma 8.1, we have that $s^* \succ s$, and s^* is not optimal, a contradiction. Therefore $g_j(s_j^*) = \min_{1 \leq k \leq N} g_k(s_k^*)$. □

Lemma 9.2 *Given a VBR bit-allocation problem $P = (N, F, B_{\text{tgt}}, B_{\text{vbv}}, B_1, B_{\text{a}})$ and an optimal allocation s^*, if $B_{\text{f}}(s^*, N) > s_N^*$ then $g_N(s_N^*) = \min_{1 \leq k \leq N} g_k(s_k^*)$.*

Proof: Let $Q_{\min} = \min_{1 \leq k \leq N} g_k(s_k^*)$. Let j be an index such that $B_{\text{f}}(s^*, j) + B_{\text{a}}(j) - s_j^* > B_{\text{vbv}}$. Let $\Delta = B_{\text{f}}(s^*, N) - s_N^*$. Since $B_{\text{f}}(s^*, N) > s_N^*$, $\Delta > 0$. Suppose that $g_N(s_N^*) > Q_{\min}$. Let u be an index such that $g_u(s_u^*) = Q_{\min}$. Now consider an allocation s that differs from s^* only for pictures u and N. We want to assign values to s_N and s_u that would make s a legal allocation with $g_N(s_N), g_u(s_u) < g_N(s_N^*)$, from which $s^* \succ s$, thereby arriving at a contradiction. Let $s_N = s_N^* + \delta$ and $s_u = s_u^* - \delta$ with $\delta > 0$. Then $g_N(s_N) < g_N(s_N^*)$ and $\sum_{k=1}^{N} s_k = \sum_{k=1}^{N} s_k^* = B_{\text{tgt}}$. We now need to show that s does not result in a VBV buffer underflow, that is, $B_{\text{f}}(s, k) \geq s_k$ for $1 \leq k \leq N$.

Since $s_k = s_k^*$ for $k < u$, $B_{\text{f}}(s, k) = B_{\text{f}}(s^*, k)$ for $1 \leq k \leq u$. Since $s_u < s_u^*$ and $s_k = s_k^*$ for $u < k < N$, $B_{\text{f}}(s, k) \geq B_{\text{f}}(s^*, k)$ for $u < k \leq N$. Therefore pictures 1 to $N - 1$ cannot cause any VBV buffer underflows. For picture N, we have $B_{\text{f}}(s, N) \geq B_{\text{f}}(s^*, N) = \Delta + s_N^* = \Delta + s_N - \delta$. Therefore if we choose $\delta < \Delta$, then $B_{\text{f}}(s, N) > s_N$ and picture N also cannot cause a VBV buffer underflow.

Therefore s is a legal allocation with $g_N(s_N) < g_N(s_N^*)$. We need to guarantee that $g_u(s_u) < g_N(s_N^*)$. Let $\gamma = g_N(s_N^*) - g_u(s_u^*)$. Since $g_N(s_N^*) > g_u(s_u^*)$, we have $\gamma > 0$. Let $\alpha = s_u^* - f_u(g_u(s_u^*) + \gamma/2)$. Since f_u is decreasing and $\gamma > 0$, we have $\alpha > 0$ and

$$\begin{aligned} s_u^* - \alpha &= f_u(g_u(s_u^*) + \gamma/2) \\ g_u(s_u^* - \alpha) &= g_u(s_u^*) + \gamma/2 \\ g_u(s_u^* - \alpha) &< g_u(s_u^*) + \gamma = g_N(s_N^*) \\ g_u(s_u^* - \alpha) &< g_N(s_N^*). \end{aligned}$$

Consider the assignment $\delta = \min(\alpha, \Delta/2)$. There are two cases: $\alpha \leq \Delta/2$ and $\alpha > \Delta/2$.

$\alpha \le \Delta/2$**:** We have $\delta = \alpha$ from which $g_u(s_u) = g_u(s_u^* - \delta) = g_u(s_u^* - \alpha) < g_N(s_N^*)$. Since $0 < \delta < \Delta$, the allocation s is legal.

$\alpha > \Delta/2$**:** We have $\delta = \Delta/2$. Since g_u is decreasing and $\alpha > \Delta/2$, we have $g_u(s_u^* - \Delta/2) < g_u(s_u^* - \alpha)$ and thus $g_u(s_u) = g_u(s_u^* - \delta) = g_u(s_u^* - \Delta/2) < g_N(s_N^*)$. Since $0 < \delta < \Delta$, the allocation s is legal.

Since s is a legal allocation that differs from s^* only for pictures u and N with $g_u(s_u), g_N(s_N) < g_N(s_N^*)$, from Lemma 8.1, we have that $s^* \succ s$, and s^* is not optimal, a contradiction. Therefore $g_N(s_N^*) = \min_{1 \le k \le N} g_k(s_k^*)$. □

The next lemma gives a set of *switching* conditions for changes in Q that are similar to the results of Lemma 8.2.

Lemma 9.3 (VBR Switching) *Given a VBR bit-allocation problem $P = (N, F, B_{\mathrm{tgt}}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}})$, if s^* is an optimal allocation, the following are true:*

1. *If $g_j(s_j^*) > g_{j+1}(s_{j+1}^*)$ for $1 \le j < N$, then $B_{\mathrm{f}}(s^*, j) = s_j^*$.*
2. *If $g_j(s_j^*) < g_{j+1}(s_{j+1}^*)$ for $1 \le j < N$, then $B_{\mathrm{f}}(s^*, j+1) = B_{\mathrm{vbv}}$ and $B_{\mathrm{f}}(s^*, j+1) + B_{\mathrm{a}}(j+1) - s_{j+1}^* \le B_{\mathrm{vbv}}$.*

Proof:

Claim 1: The proof is identical to the proof of Claim 1 of Lemma 8.2, except that condition (6) now holds instead of (9).

Claim 2: Suppose that the claim is false. Then either $B_{\mathrm{f}}(s^*, j+1) < B_{\mathrm{vbv}}$ or $B_{\mathrm{f}}(s^*, j+1) + B_{\mathrm{a}}(j+1) - s_{j+1}^* > B_{\mathrm{vbv}}$. Suppose that $B_{\mathrm{f}}(s^*, j+1) + B_{\mathrm{a}}(j+1) - s_{j+1}^* > B_{\mathrm{vbv}}$. Then by Lemma 9.1, $g_{j+1}(s_{j+1}^*) \le g_j(s_j^*)$, a contradiction. Therefore $B_{\mathrm{f}}(s^*, j+1) + B_{\mathrm{a}}(j+1) - s_{j+1}^* \le B_{\mathrm{vbv}}$.

Suppose that $B_{\mathrm{f}}(s^*, j+1) < B_{\mathrm{vbv}}$. Let $\Delta = B_{\mathrm{vbv}} - B_{\mathrm{f}}(s^*, j+1)$. Then $\Delta > 0$. Now consider an allocation s that differs from s^* only for pictures j and $j+1$. We want to assign values to s_j and s_{j+1} that would make s a legal allocation with $g_j(s_j), g_{j+1}(s_{j+1}) < g_{j+1}(s_{j+1}^*)$, from which $s^* \succ s$, thereby arriving at a contradiction.

Taking cues from Lemma 9.1, let $\gamma = g_{j+1}(s_{j+1}^*) - g_j(s_j^*)$ and $\alpha = s_j^* - f_j(g_j(s_j^*) + \gamma/2)$. Since $g_j(s_j^*) < g_{j+1}(s_{j+1}^*)$, $\gamma > 0$. Since f_j is decreasing and $\gamma > 0$, we have $\alpha > 0$ and

$$
\begin{aligned}
s_j^* - \alpha &= f_j(g_j(s_j^*) + \gamma/2) \\
g_j(s_j^* - \alpha) &= g_j(s_j^*) + \gamma/2 \\
g_j(s_j^* - \alpha) &< g_j(s_j^*) + \gamma = g_{j+1}(s_{j+1}^*) \\
g_j(s_j^* - \alpha) &< g_{j+1}(s_{j+1}^*).
\end{aligned}
$$

Consider the assignments $s_j = s_j^* - \delta$ and $s_{j+1} = s_{j+1}^* + \delta$, where $\delta = \min(\alpha, \Delta/2)$. There are two cases: $\alpha \le \Delta/2$ and $\alpha > \Delta/2$.

$\alpha \le \Delta/2$**:** We have $\delta = \alpha$ from which $g_j(s_j) = g_j(s_j^* - \delta) = g_j(s_j^* - \alpha) < g_{j+1}(s_{j+1}^*)$.

$\alpha > \Delta/2$**:** We have $\delta = \Delta/2$. Since g_j is decreasing and $\alpha > \Delta/2$, we have $g_j(s_j^* - \Delta/2) < g_j(s_j^* - \alpha)$ and thus $g_j(s_j) = g_j(s_j^* - \delta) = g_j(s_j^* - \Delta/2) < g_{j+1}(s_{j+1}^*)$.

We now need to show that allocation s as defined above is a legal allocation. Since $s_k = s_k^*$ for $k < j$, $B_{\mathrm{f}}(s,k) = B_{\mathrm{f}}(s^*,k)$ for $1 \le k \le j$. Therefore there are no VBV buffer violations in pictures 1 to $j-1$. Since $s_j < s_j^*$, $B_{\mathrm{f}}(s,j+1) > B_{\mathrm{f}}(s^*,j+1)$. Therefore picture j could not cause a VBV buffer underflow.

Now we need to show that pictures $j+1$ to N also cannot cause a VBV buffer underflow. Since $B_{\mathrm{f}}(s^*,j+1) < B_{\mathrm{vbv}}$, $B_{\mathrm{f}}(s^*,j+1) = B_{\mathrm{f}}(s^*,j) + B_{\mathrm{a}}(j) - s_j^*$ and

$$\begin{aligned}
B_{\mathrm{f}}(s,j) + B_{\mathrm{a}}(j) - s_j &= B_{\mathrm{f}}(s^*,j) + B_{\mathrm{a}}(j) - (s_j^* - \delta) \\
&= B_{\mathrm{f}}(s^*,j) + B_{\mathrm{a}}(j) - s_j^* + \delta \\
&= B_{\mathrm{f}}(s^*,j+1) + \delta \\
&< B_{\mathrm{f}}(s^*,j+1) + \Delta = B_{\mathrm{vbv}} \\
&< B_{\mathrm{vbv}}.
\end{aligned}$$

Thus $B_{\mathrm{f}}(s,j+1) = B_{\mathrm{f}}(s,j) + B_{\mathrm{a}}(j) - s_j$. We have already shown that $B_{\mathrm{f}}(s^*,j+1) + B_{\mathrm{a}}(j+1) - s_{j+1}^* \le B_{\mathrm{vbv}}$. Therefore $B_{\mathrm{f}}(s^*,j+2) = B_{\mathrm{f}}(s^*,j+1) + B_{\mathrm{a}}(j+1) - s_{j+1}^*$. Now,

$$\begin{aligned}
B_{\mathrm{f}}(s,j+1) + B_{\mathrm{a}}(j+1) - s_{j+1} &= B_{\mathrm{f}}(s,j) + B_{\mathrm{a}}(j) - s_j + B_{\mathrm{a}}(j+1) - s_{j+1} \\
&= B_{\mathrm{f}}(s^*,j+1) + \delta + B_{\mathrm{a}}(j+1) - (s_{j+1}^* + \delta) \\
&= B_{\mathrm{f}}(s^*,j+1) + B_{\mathrm{a}}(j+1) - s_{j+1}^* \\
&= B_{\mathrm{f}}(s^*,j+2) \le B_{\mathrm{vbv}}.
\end{aligned}$$

Therefore $B_{\mathrm{f}}(s,j+2) = B_{\mathrm{f}}(s^*,j+2)$. Since $s_k = s_k^*$ for $k > j+1$, we have that $B_{\mathrm{f}}(s,k) = B_{\mathrm{f}}(s^*,k)$ for $k > j+1$. Therefore pictures $j+1$ to N cannot cause a VBV buffer underflow and s is a legal allocation.

Since s is a legal allocation that differs from s^* only for pictures j and $j+1$ with $g_j(s_j), g_{j+1}(s_{j+1}) < g_{j+1}(s_{j+1}^*)$, from Lemma 8.1, we have that $s^* \succ s$, and s^* is not optimal, a contradiction.

□

The following theorem is the main result of this section. It shows that the minimal-Q and switching conditions stated in the previous lemmas are also sufficient for optimality.

Theorem 9.1 *Given a VBR bit-allocation problem $P = (N, F, B_{\mathrm{tgt}}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}})$, a legal allocation s 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 $g_j(s_j) = \min_{1 \le k \le N} g_k(s_k)$.*
2. *If $B_{\mathrm{f}}(s^*,N) > s_N^*$ then $g_N(s_N^*) = \min_{1 \le k \le N} g_k(s_k^*)$.*
3. *If $g_j(s_j) > g_{j+1}(s_{j+1})$ for $1 \le j < N$, then $B_{\mathrm{f}}(s,j) = s_j$.*
4. *If $g_j(s_j) < g_{j+1}(s_{j+1})$ for $1 \le j < N$, then $B_{\mathrm{f}}(s,j+1) = B_{\mathrm{vbv}}$ and $B_{\mathrm{f}}(s,j+1) + B_{\mathrm{a}}(j+1) - s_{j+1} \le B_{\mathrm{vbv}}$.*

Proof: Lemmas 9.1, 9.2, and 9.3 establish these as necessary conditions. Now we need to show that these conditions are also sufficient for optimality and imply uniqueness.

The proof for sufficiency and uniqueness is similar to that of Theorem 8.1 except for segments with the minimum Q. Let s^* be an optimal allocation, $Q_{\min} = \min_{1 \le j \le N} g_j(s_j)$, and $J_{\min} = \{j : g_j(s_j) = Q_{\min}\}$. By condition 2, if $g_N(s_N) > Q_{\min}$ then it must be that $B_{\mathrm{f}}(s,N) = s_N$, or equivalently, $B_{\mathrm{f}}(s, N+1) = B_{\mathrm{a}}(N)$. Therefore $B_{\mathrm{f}}(s, N)$ is known if picture N does not use the minimum Q, and we can use arguments of Theorem 8.1. Following the steps of Theorem 8.1, we can show that $s_j^* = s_j$ for $j : g_j(s_j) > Q_{\min}$.

Since s^* is optimal, $g_j(s_j^*) \le g_j(s_j)$ for $j \in J_{\min}$. Therefore

$$s_j^* \ge s_j \text{ for } j \in J_{\min}. \tag{42}$$

Since the total number of bits allocated is the same for s and s^*, we have that the number of bits to be allocated to pictures in J must also be the same. That is,

$$\sum_{j \in J_{\min}} s_j^* = \sum_{j \in J_{\min}} s_j. \tag{43}$$

But (42) and (43) both hold if and only if $s_j^* = s_j$ for $j \in J_{\min}$. Therefore $s = s^*$. □

Although Theorem 9.1 is an important result, it does not show us how to compute the minimum Q with which to code the "easy" pictures. The following lemmas and theorem show that, if we relax the bit budget constraint, we can find the minimum Q, and therefore the optimal allocation, to meet the bit budget by a interative process. Furthermore, the iterative process converges to the optimal allocation in a finite number of steps.

Lemma 9.4 *Given two VBR bit-allocation problems*

$$\begin{aligned} P^{(1)} &= (N, F, B_{\mathrm{tgt}}^{(1)}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}}) \\ P^{(2)} &= (N, F, B_{\mathrm{tgt}}^{(2)}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}}) \end{aligned}$$

that have optimal allocations $s^{(1)}$ and $s^{(2)}$, respectively, with $B_{\mathrm{tgt}}^{(1)} < B_{\mathrm{tgt}}^{(2)}$, then $s^{(1)} \succ s^{(2)}$.

Proof: Let $Q_{\min}^{(1)} = \min_{1 \le j \le N} g_j(s_j^{(1)})$, $\Delta = B_{\mathrm{tgt}}^{(2)} - B_{\mathrm{tgt}}^{(1)}$, and $J_{\mathrm{over}} = \left\{ j : B_{\mathrm{f}}(s^{(1)}, j) + B_{\mathrm{a}}(j) - s_j^{(1)} > B_{\mathrm{vbv}} \right\}$. If we start with $s^{(1)}$, it is clear that we can use more bits for the pictures in J_{over} without changing the buffer fullness. Let $B_{\mathrm{over}} = \sum_{j \in J_{\min}} B_{\mathrm{f}}(s^{(1)}, j) + B_{\mathrm{a}}(j) - s_j^{(1)} - B_{\mathrm{vbv}}$. Then B_{over} is the maximum number of bits we can add to the pictures in J_{over} without changing the buffer fullness. There are two cases to consider: $\Delta \le B_{\mathrm{over}}$ and $\Delta > B_{\mathrm{over}}$.

$\Delta \le B_{\mathrm{over}}$: Consider an allocation s for problem $P^{(2)}$ constructed as follows. Let $s_j = s_j^{(1)}$ for $j \notin J_{\mathrm{over}}$. We then distribute Δ bits to the pictures in J_{over} without changing the buffer fullness. Then $s_j \ge s_j^{(1)}$ which implies that $g_j(s_j) \le g_j(s_j^{(1)})$. Since $\Delta > 0$, we also have that $s_j > s_j^{(1)}$ for some $j \in J_{\mathrm{over}}$. Since $B_{\mathrm{f}}(s, j) = B_{\mathrm{f}}(s^{(1)}, j)$ for all j, s does not cause any buffer underflows. Since we used $B_{\mathrm{tgt}}^{(1)} + \Delta = B_{\mathrm{tgt}}^{(2)}$ bits in s, s is a legal allocation for $P^{(2)}$. Since $g_j(s_j) \le g_j(s_j^{(1)})$ for all j and $g_j(s_j) < g_j(s_j^{(1)})$ for some j, from Lemma 8.1, we have that $s^{(1)} \succ s$. Since $s^{(2)}$ is the optimal allocation for $P^{(2)}$, we have $s \succeq s^{(2)}$. Therefore $s^{(1)} \succ s^{(2)}$.

$\Delta > B_{\mathrm{over}}$: Consider an allocation s for problem $P^{(2)}$ constructed as follows. Let $s_j = s_j^{(1)}$ for $j \notin J_{\mathrm{over}} \cup \{N\}$. We then distribute B_{over} bits to pictures in J_{over}. We do this with the assignments: $s_j = s^{(1)} + (B_{\mathrm{f}}(s^{(1)}, j) + B_{\mathrm{a}}(j) - s_j^{(1)} - B_{\mathrm{vbv}})$ for $j \in J_{\mathrm{over}}$. Finally, we distribute the remaining $\Delta - B_{\mathrm{over}}$ bits to picture N with $s_N = s_N^{(1)} + \Delta - B_{\mathrm{over}}$.

We have shown how to create a legal allocation s for $P^{(2)}$. When we add more bits to s, we strictly increase the Q for the pictures that we add bits to and never increase Q anywhere. Therefore $s^{(1)} \succ s$. Since $s^{(2)}$ is the optimal allocation for $P^{(2)}$, we have $s \succeq s^{(2)}$. Therefore $s^{(1)} \succ s^{(2)}$.

$\square$

Lemma 9.5 *Given two VBR bit-allocation problems*

$$\begin{aligned} P^{(1)} &= (N, F, B_{\mathrm{tgt}}^{(1)}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}}) \\ P^{(2)} &= (N, F, B_{\mathrm{tgt}}^{(2)}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}}) \end{aligned}$$

that have optimal allocations $s^{(1)}$ and $s^{(2)}$, respectively, with $B_{\mathrm{tgt}}^{(1)} < B_{\mathrm{tgt}}^{(2)}$, then $s_j^{(1)} = s_j^{(2)}$ for j such that $g_j(s_j^{(1)}) > \min_{1 \le k \le N} g_k(s_k^{(1)})$.

Proof: We provide an inductive proof similar to that used to prove Theorem 8.1. First we assume that $s^{(1)}$ is not a constant Q allocation, for if it were, then the lemma would hold vacuously.

Let $Q_k^{(1)}$ be the k^{th} highest value of Q assigned by allocation $s^{(1)}$. Let $Q_{\min}^{(1)}$ be the minimum value of Q.

Inductive Hypothesis: For all segments of pictures u to v with $u = 1$ or $g_{u-1}(s_{u-1}) \neq g_u(s_u)$, $v = N$ or $g_v(s_v) \neq g_{v+1}(s_{v+1})$, and $g_j(s_j) = Q_k^{(1)} > Q_{\min}^{(1)}$, we have that $s_j^{(1)} = s_j^{(2)}$ and $B_{\mathrm{f}}(s^{(1)}, j) = B_{\mathrm{f}}(s^{(2)}, j)$ for $u \le j \le v$.

We first prove the base case of $k = 1$. Consider the segments of consecutive pictures that are assigned quantization $Q_1^{(1)}$ by allocation $s^{(1)}$. Let u be the index of the start of such a segment. There are two cases:

$u = 1$: $B_{\mathrm{f}}(s^{(1)}, u) = B_{\mathrm{f}}(s^{(2)}, u) = B_1$.

$u > 1$: Since u is the index of the start of the segment, $g_{u-1}(s_{u-1}^{(1)}) < g_u(s_u^{(1)})$ which implies that $B_{\mathrm{f}}(s^{(1)}, u) = B_{\mathrm{vbv}}$ by Lemma 9.3. Since $s^{(2)}$ is a legal allocation, $B_{\mathrm{f}}(s^{(2)}, u) \le B_{\mathrm{vbv}}$.

In either case we have

$$B_{\mathrm{f}}(s^{(1)}, u) \ge B_{\mathrm{f}}(s^{(2)}, u) \tag{44}$$

Let v be the index of the end of the segment. There are two cases:

$v = N$: By the contrapositive of Lemma 9.2, $B_{\mathrm{f}}(s^{(1)}, v) = s_v^{(1)}$. (Here we use the condition that $Q_1^{(1)} > Q_{\min}^{(1)}$.)

$v < N$: Since v is the index of the end of the segment, $g_v(s_v^{(1)}) > g_{v+1}(s_{v+1}^{(1)})$ which implies that $B_{\mathrm{f}}(s^{(1)}, v) = s_v^{(1)}$ by Lemma 9.3.

In either case we have

$$B_{\mathrm{f}}(s^{(1)}, v) = s_v^{(1)}. \tag{45}$$

From Lemma 9.4, we have $s^{(1)} \succ s^{(2)}$. Therefore $g_j(s_j^{(2)}) \le Q_1^{(1)}$ for all j and thus

$$s_j^{(1)} \le s_j^{(2)} \text{ for } u \le j \le v. \tag{46}$$

From (44) and (46) we have

$$B_{\mathrm{f}}(s^{(1)}, j) \ge B_{\mathrm{f}}(s^{(2)}, j) \quad \text{for } u \le j \le v. \ . \tag{47}$$

Since $s^{(2)}$ is a legal allocation,

$$B_{\mathrm{f}}(s^{(2)}, v) \geq s_v^{(2)} \geq s_v^{(1)} = B_{\mathrm{f}}(s^{(1)}, v). \tag{48}$$

Combining (47) and (48), we have that $B_{\mathrm{f}}(s^{(1)}, v) = B_{\mathrm{f}}(s^{(2)}, v)$ and $s_v^{(1)} = s_v^{(2)}$. Therefore $B_{\mathrm{f}}(s^{(1)}, v+1) = B_{\mathrm{f}}(s^{(2)}, v+1)$. Since $Q_1^{(1)} > Q_{\min}^{(1)}$, by the contrapositive of Lemma 9.2, we see that the buffer fullness for pictures u to v is updated the same as with CBR operation. Therefore we can use the results of Lemma 8.3, which implies that $B_{\mathrm{f}}(s_j^{(1)}) = B_{\mathrm{f}}(s^{(2)}, j)$ and $s_j^{(1)} = s_j^{(2)}$ for $u \leq j \leq v$.

Assume that the inductive hypothesis is true for $1 \leq k < n$. We need to show that it is also true for $k = n$ where $Q_n^{(1)} > Q_{\min}^{(1)}$. Consider a segment of consecutive pictures that are assigned quantization $Q_n^{(1)}$. Let u be the index of the start of the segment and v the index of the end of the segment. We consider all cases for the segment boundaries. For the left segment boundary there are three cases:

$u = 1$:

$B_{\mathrm{f}}(s^{(1)}, u) = B_{\mathrm{f}}(s^{(2)}, u) = B_1$.

$g_{u-1}(s_{u-1}^{(1)}) > g_u(s_u^{(1)})$:

From the inductive hypothesis, we have $B_{\mathrm{f}}(s^{(1)}, u-1) = B_{\mathrm{f}}(s^{(2)}, u-1)$ and $s_{u-1}^{(1)} = s_{u-1}^{(2)}$. Therefore $B_{\mathrm{f}}(s^{(1)}, u) = B_{\mathrm{f}}(s^{(2)}, u)$.

$g_{u-1}(s_{u-1}^{(1)}) < g_u(s_u^{(1)})$:

From Lemma 9.3, we have $B_{\mathrm{f}}(s^{(1)}, u) = B_{\mathrm{vbv}}$. Since $s^{(2)}$ is a legal allocation, $B_{\mathrm{f}}(s^{(2)}, u) \leq B_{\mathrm{vbv}} = B_{\mathrm{f}}(s^{(1)}, u)$.

For all three cases we have

$$B_{\mathrm{f}}(s^{(2)}, u) \leq B_{\mathrm{f}}(s^{(1)}, u). \tag{49}$$

For the right segment boundary there are three cases:

$v = N$: By the contrapositive of Lemma 9.2, $B_{\mathrm{f}}(s^{(1)}, v) = s_v^{(1)}$. (We use the condition that $Q_n^{(1)} \neq Q_{\min}^{(1)}$.)

$g_v(s_v^{(1)}) > g_{v+1}(s_{v+1}^{(1)})$: By Lemma 9.3, $B_{\mathrm{f}}(s^{(1)}, v) = s_v^{(1)}$.

$g_v(s_v^{(1)}) < g_{v+1}(s_{v+1}^{(1)})$: From the inductive hypothesis, we have $B_{\mathrm{f}}(s^{(1)}, v+1) = B_{\mathrm{f}}(s^{(2)}, v+1)$.

For the first two cases, we have

$$B_{\mathrm{f}}(s^{(1)}, v) = s_v^{(1)} \tag{50}$$

From Lemma 9.4, we have $s^{(1)} \succ s^{(2)}$. Therefore $g_j(s_j^{(2)}) \leq Q_n^{(1)}$ for $u \leq j \leq v$ and thus

$$s_j^{(1)} \leq s_j^{(2)} \text{ for } u \leq j \leq v. \tag{51}$$

From (49) and (51) we have

$$B_{\mathrm{f}}(s^{(1)}, j) \geq B_{\mathrm{f}}(s^{(2)}, j) \quad \text{for } u \leq j \leq v. \tag{52}$$

Since $s^{(2)}$ is a legal allocation,

$$B_{\mathrm{f}}(s^{(2)}, v) \geq s_v^{(2)} \geq s_v^{(1)} = B_{\mathrm{f}}(s^{(1)}, v). \tag{53}$$

Combining (52) and (53), we have that $B_{\mathrm{f}}(s^{(1)}, v) = B_{\mathrm{f}}(s^{(2)}, v)$ and $s_v^{(1)} = s_v^{(2)}$. Therefore $B_{\mathrm{f}}(s^{(1)}, v+1) = B_{\mathrm{f}}(s^{(2)}, v+1)$.

So for all three cases for v, we have $B_{\mathrm{f}}(s^{(1)}, v+1) = B_{\mathrm{f}}(s^{(2)}, v+1)$.

Since $Q_n^{(1)} > Q_{\min}^{(1)}$, by the contrapositive of Lemma 9.2, we see that the buffer fullness for pictures u to v is updated the same as with CBR operation. Therefore we can use the results of Lemma 8.3, which implies that $B_{\mathrm{f}}(s_j^{(1)}) = B_{\mathrm{f}}(s^{(2)}, j)$ and $s_j^{(1)} = s_j^{(2)}$ for $u \leq j \leq v$.

By induction, we have that $s_j^{(1)} = s_j^{(2)}$ for all j such that $g_j(s^{(1)}, j) > Q_{\min}^{(1)}$. □

Theorem 9.2 *Given two VBR bit-allocation problems*

$$\begin{aligned} P^{(1)} &= (N, F, B_{\mathrm{tgt}}^{(1)}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}}) \\ P^{(2)} &= (N, F, B_{\mathrm{tgt}}^{(2)}, B_{\mathrm{vbv}}, B_1, B_{\mathrm{a}}) \end{aligned}$$

that have optimal allocations $s^{(1)}$ *and* $s^{(2)}$*, respectively, with* $B_{\mathrm{tgt}}^{(1)} < B_{\mathrm{tgt}}^{(2)}$*, then* $\min_{1 \leq j \leq N} g_j(s_j^{(1)}) > \min_{1 \leq j \leq N} g_j(s_j^{(2)})$.

Proof: Let $Q_{\min}^{(1)} = \min_{1 \leq j \leq N} g_j(s_j^{(1)})$ and $Q_{\min}^{(2)} = \min_{1 \leq j \leq N} g_j(s_j^{(2)})$. From Lemma 9.4 we have $s^{(1)} \succ s^{(2)}$. From Lemma 9.5 we have that the only pictures that can be assigned different Q by $s^{(1)}$ and $s^{(2)}$ are those that are assigned quantization $Q_{\min}^{(1)}$ by $s^{(1)}$. But $s^{(1)} \succ s^{(2)}$ which implies that $s^{(2)}$ must assign to some picture a quantization lower than $Q_{\min}^{(1)}$. Therefore $Q_{\min}^{(1)} > Q_{\min}^{(2)}$. □

9.2 VBR Allocation Algorithm

Theorems 9.1 and 9.2 give us a way to find the optimal allocation for a given VBR allocation problem. If we know the minimum Q that the optimal allocation uses, then it would be straightforward to find the optimal allocation. However, in general we do not know what that minimum Q would be. Theorem 9.2 gives us an iterative way to find the minimum Q.

Our VBR allocation algorithm is sketched below.

1. Mark all pictures as "easy".
2. Compute an upper bound for $Q_{\min}$ based on assigning unallocated bits to easy pictures using a fixed-Q.
3. Run VBV algorithm to determine "hard" and "easy" pictures. Hard pictures are those that lead to buffer underflows.
4. Allocate bits to hard pictures according to CBR algorithm.
5. If extra bits remain, goto Step 2.

Note that the loop terminates when no more easy picture is turned into a hard picture. This implies that the algorithm terminates after at most N iterations, where N is the number of pictures.

Assuming a special form for the bit-production model that results in execution time of $O(N^2)$ for the CBR algorithm, it can be shown that the VBR algorithm also executes in $O(N^2)$ time and uses $O(N)$ space.

10 Limiting Look-Ahead

The above rate control algorithms compute an allocation for the entire video sequence. This may not be feasible when the sequence consists of many pictures, as in a feature-length movie, for example. One way to deal with this is to partition the sequence into blocks consisting of a small number of consecutive pictures. Optimal allocation can then be performed on the blocks separately. In order to do this, the starting and ending buffer fullness must be specified for each block for the CBR case. For the VBR case, the bit budget must be specified for each block. This approach is globally suboptimal; however, it is easy to parallelize since the block allocations are independent of each other.

The rate control algorithms assume that we know the bit-production model for each picture exactly. In practice, we can approximate the models using multiple coding passes or with an adaptive technique. These approximate models will most likely lead to buffer violations if we use the quantization scales that are computed. One approach is to enforce the bit allocation, instead of the quantization scale, with a lower-level rate control algorithm (within a picture). One disadvantage to this is that local errors in the modeling can potentially affect the global allocation.

Another approach is to use limited look-ahead to compute the bit allocation for the first picture. Using this allocation, we code the first picture and update the VBV buffer and bit-production model according to the actual bit-production. We then compute the bit allocation for the second picture using a limited look-ahead and so on. This technique allows for the bit-production models of future pictures to be updated based on the coding of previous pictures, potentially reducing the number of encoding passes.

Another approach is to use the allocation computed from a given model and only recompute the allocation when the buffer fullness breach preset buffer boundaries, such as 10% and 90% of buffer fullness. If the bit-production model is fixed for an entire encoding pass, this approach can be efficiently implemented for the CBR case by first computing a dynamic programming solution (in the reverse direction). Each time the buffer fullness breaches the buffer boundaries, the pre-computed dynamic programming table can be used to continue the allocation.

11 Conclusions and Future Work

We have proposed a framework for the analysis of rate control for constant-quality video compression, in which we formulate rate control as a resource allocation problem with a lexicographic optimality criterion. We derive constraints for a video buffer verifier for operation at both constant and variable bit rates. Using these constraints and an idealized model of video coding, we prove necessary and sufficient conditions for optimality and provide efficient algorithms for computing the optimal allocations.

In previous work, Ortega, Ramchandran, and Vetterli [5] analyze bit allocation with constant bit rate constraints using an additive distortion metric. They formulate the problem as a 0-1 integer programming problem and solve it using the Viterbi dynamic programming algorithm.

The analyses presented here are based on simplications of the actual video coding process. In particular, one of the prerequisites is an accurate bit-production model for each picture of the video sequence. Another prerequisite is proper adaptive perceptual quantization.

In itself, bit-production modeling is a challenging task that needs further work. For preliminary testing of the allocation algorithms, we used multiple encoding passes to build and refine a bit-production model. The rate control algorithms have been implemented within a MPEG-2 encoder and preliminary experiments show that the algorithms do indeed result in near-constant quality video for short (a few hundred pictures) test sequences. More experiments in coding a variety of longer sequences is planned to more accurately access the performance of the algorithms.

The bit-production model described in Section 4 assumes that the coding of a particular picture is independent of the coding of any other picture. As noted earlier, this independence assumption does not hold for video coders that employ a differential coding scheme such as motion compensation. In this case, the coding of a reference picture affects the coding of subsequent pictures that are coded with respect to it. Therefore, the bit-production model for picture i would depend causally not only on the quantization scale used for picture i but also on the quantization scales used for its reference pictures. The dependent coding problem has been addressed in the traditional distorion-minimization framework in [6]. In future work, we propose to formulate a causally dependent coding model and hope to extend the lexicographically optimal results of the previous sections.

References

[1] R. M. Gray, *Source Coding Theory*, Kluwer Academic Publishers, Boston, MA, 1990.

[2] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 9, pp. 1445–1453, Sept. 1988.

[3] N. S. Jayant, J. Johnson, and R. Safranek, "Signal compression based on models of human perception", *Proceedings of the IEEE*, Oct. 1993.

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

[5] A. Ortega, K. Ramachandran, and M. Vetterli, "Optimal buffer-constrained source quantization and fast approximations", in *Proceedings 1992 IEEE International Symposium on Circuits and Systems*, San Diego, CA, May 1992, pp. 192–195.

[6] K. Ramachandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders", *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 533–545, Sept. 1994.

[7] ISO-IEC/JTC1/SC29/WG11/N0802, "Generic coding of moving pictures and associated audio information: Video", Nov. 1994, MPEG Draft Recommendation ITU-T H.262, ISO/IEC 13818-2.

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

[9] H. Luss and S. K. Gupta, "Allocation of effort resources among competitive activities", *Operations Research*, vol. 23, pp. 360–366, 1975.