

Optimal Multidimensional Bit-Rate Control for Video Communication

Eric C. Reed, *Member, IEEE*, and Jae S. Lim, *Fellow, IEEE*

Abstract—In conventional bit-rate control, the buffer level is controlled by adapting the quantization step size with a fixed frame rate and spatial resolution. We consider a multidimensional (M-D) bit-rate control where the frame rate, spatial resolution and quantization step size are jointly adapted for buffer control. We introduce a fundamental framework to formalize the description of the M-D buffer-constrained allocation problem. Given a set of operating points on a M-D grid to code a nonstationary source in a buffer-constrained environment, we formulate the optimal solution. The formulation allows a skipped frame to be reconstructed from one coded frame using any temporal interpolation method and is shown to be a generalization of formulations considered in the literature. In the case of intraframe coding, a dynamic programming algorithm is introduced to find the optimal solution. The algorithm allows one to compare operational rate-distortion bounds of the M-D and conventional approaches. We also discuss how a solution can be obtained for the case of interframe coding using the optimal dynamic programming algorithm for intraframe coding by making an independent allocation approximation. We illustrate that the M-D approach can provide bit-rate reductions over 50%. We also show that the M-D approach with limited-lookahead provides a slightly suboptimal solution that consistently outperforms the conventional approach with full-lookahead.

Index Terms—Bit-rate control, buffer control, rate-distortion optimization, variable frame rate, variable spatial resolution, very low bit rate, video transmission.

I. INTRODUCTION

IN DIGITAL VIDEO communications, buffering is necessary to absorb variations between the source rate and the channel rate. Due to the broad number of applications, the problem of allocating bits in a buffer-constrained environment has been studied extensively [1], [2]. Most of the emphasis has been placed on the conventional bit-rate control approach where the problem is how to choose quantizers under a buffer constraint while the frame rate and spatial resolution processed by the encoder remain fixed. The conventional approach is well suited for high-bit-rate applications where overhead uses a small fraction of the bit rate and high-quality video is achieved by coding at full frame rate and spatial resolution. At low bit rates, however, it is necessary to code at a reduced frame rate

and/or reduced spatial resolution due to the required transmission of overhead bits. The conventional approach is often applied to low-bit-rate applications for simplicity [3]. However, using the conventional approach at low bit rates requires the frame rate and spatial resolution processed by the encoder to be chosen *a priori*. The choice of these parameters is critical since they have a direct impact on the quantization and overall video quality. Quite often, frames must be dropped arbitrarily to prevent buffer overflow. Furthermore, since these parameters remain fixed, they are not adapting to a nonstationary source.

In this paper, we consider a more general multidimensional (M-D) bit-rate control where the buffer level is controlled by jointly adapting the frame rate, spatial resolution and quantization stepsize [4]. Some variable frame rate and spatial resolution bit-rate control schemes have been considered. For example, the frame rate is adjusted based on the histogram of difference (HOD) measure in [5], [6]. The basic idea is to reduce the frame rate when motion becomes faster and increase the frame rate when motion becomes slower. The HOD measure is useful for detecting motion and was first introduced in [7]. A source model is used in [8] to predict rate-distortion (R-D) characteristics and the frame rate is adjusted to ensure a minimum picture quality of the coded frames. A statistically based approach is taken in [9] where buffer control is performed by vertical subsampling and quantization. While these approaches are ad hoc, they can yield better video quality over conventional approaches.

This paper formalizes the M-D bit-rate control process. In particular, we define the M-D buffer-constrained allocation problem and present an integer programming formulation. The problem involves selecting which frames to code (and which frames to skip), along with the spatial resolution and the quantization stepsize for each coded frame in a buffer-constrained environment such that the reconstructed video sequence is as close as possible to the original according to some objective measure. Our formulation of the problem allows a skipped frame to be reconstructed from one coded frame determined by the choice of a reconstruction pattern and reduces to the formulation in [2] for the special case of conventional bit-rate control where the video is coded at full frame rate and spatial resolution. The added flexibility of the M-D bit-rate control approach allows the controller to be more adaptive to a nonstationary source. For example, the controller has the flexibility to skip more frames when the temporal correlation is high and to code more frames when the temporal correlation is low. Similarly, the controller has the flexibility to spatially subsample frames prior to coding when the spatial correlation is high. There are many interesting questions to consider with the M-D approach. For example, how much coding gain can be achieved using

Manuscript received February 27, 2002. This work was supported by Draper Laboratory and Motorola Broadband Communications Sector. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. A. Enis Cetin.

E. C. Reed is with the Convergent Systems Division, Harmonic, Inc., White Plains, NY 10601 USA (e-mail: ereed@alum.mit.edu).

J. S. Lim is with the Advanced Telecommunications and Signal Processing Group, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: jslim@mit.edu).

Publisher Item Identifier 10.1109/TIP.2002.801122.

1057-7149/02\$17.00 © 2002 IEEE

Authorized licensed use limited to: Fenwick & West. Downloaded on January 24, 2025 at 18:45:03 UTC from IEEE Xplore. Restrictions apply.

the M-D approach compared with the conventional approach? What is the optimal video format as a function of bit rate and buffer size?

A dynamic programming algorithm is presented to obtain an optimal solution for the case of intraframe coding, a special case of dependent coding. Dependency still exists with the case of intraframe coding since skipped frames are reconstructed from coded frames. In the special case of conventional bit-rate control where the video is coded at full frame rate and spatial resolution, our algorithm reduces to the optimal independent algorithm in [2]. Extending our previous work in [10], we also consider the case of interframe coding and discuss how a solution can be obtained using the optimal dynamic programming algorithm for intraframe coding by making an independent allocation approximation. While our algorithm is computationally expensive, it can be directly used for nonreal-time encoding, for benchmarking, and as an aid in the development of suboptimal algorithms.

The optimal solution allows one to compare the operational R-D bounds of the M-D and conventional approaches. To obtain an optimal solution, however, it is assumed that one has access to the entire video sequence. While this is true for non-real-time encoding applications, it is not true for real-time encoding applications due to delay requirements. Therefore, we compare the performance of the M-D bit-rate control in the case of limited-lookahead with the optimal solution (full-lookahead). For the conventional bit-rate control approach where the video is coded at full frame rate and spatial resolution, it is demonstrated in [2] that slightly suboptimal solutions can be obtained with limited-lookahead. The conventional approach with limited-lookahead is also analyzed in [11] where the buffer constraints are viewed as a set of bit budget constraints.

Section II defines the M-D buffer-constrained allocation problem. An integer programming formulation is presented in Section III and the optimal algorithm is presented in Section IV. Experimental results for both intraframe and interframe coding are discussed in Section V.

II. PROBLEM DEFINITION

The M-D bit-rate control approach is illustrated in Fig. 1. Video enters a pre-processor with a delay of ΔK frames so the controller has knowledge of ΔK future frames to achieve better bit allocation. The pre-processor performs temporal and spatial subsampling operations on the video input and therefore can be represented by a cascade connection of a skipped/coded switch followed by a spatial subsampler. The encoder produces a compressed bitstream representation of the subsampled video. Bits produced by the encoder are placed into the buffer at a variable rate and pulled from the buffer at an assumed constant rate. To control the level of the buffer, the M-D bit-rate controller jointly operates the skipped/coded switch, spatial subsampler and the quantizer used in the encoder. The optimal operation of the pre-processor and the quantizer is obtained by solving the M-D buffer-constrained allocation problem which is defined as follows:

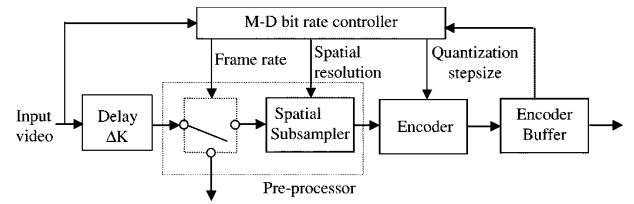


Fig. 1. Illustration of M-D bit-rate control approach. Controller jointly adapts four encoding parameters: frame rate parameter i , spatial subsampling parameters s_h , s_v and quantizer parameter q . The video enters the preprocessor with a delay of ΔK frames to achieve better bit allocation.

Formulation 1: (M-D Buffer-Constrained Allocation Problem)
Given a set of operating points on a M-D grid, a sequence of frames, a finite buffer, and spatial and temporal interpolation methods to be used at the receiver, the goal is to select the operating points (i.e., select which frames to code, and which frames to skip, along with the spatial resolution and quantizer for each coded frame) such that 1) the buffer is never in overflow and 2) some global distortion metric is minimized.

Section II-A defines the encoding parameters that represent each operating point. Section II-B defines a fundamental set of reconstruction patterns used to reconstruct skipped frames from coded frames. Section II-C discusses the delay imposed on the system.

A. Encoding Parameters

Each operating point defines the choice of four encoding parameters:

- 1) temporal subsampling or frame rate parameter, i , which defines the distance from the last coded frame;
- 2) quantizer parameter, q , which represents one-half the quantization stepsize;
- 3) horizontal spatial subsampling parameter, s_h , which defines the horizontal spatial resolution;
- 4) vertical spatial subsampling parameter, s_v , which defines the vertical spatial resolution.

These parameters can be defined at the frame or block level. We consider a frame layer rate control where i , q , s_h , and s_v are defined at the frame level. Our frame layer rate control determines an optimal bit allocation among each coded frame. The resulting bit allocation can then be used by a block layer rate control where q , s_h , and s_v are defined at the block level. Given the bit budget for a coded frame resulting from our frame layer rate control, the frame can be coded in various ways as long as the bit budget is not exceeded. In the case of MPEG-4 [12] where a video sequence is comprised of multiple video objects, operating points can be chosen separately for each video object. The methods discussed in this paper can be used to obtain an optimal bit allocation among the different objects.

The quantization performed during the coding process is significantly affected by the video format chosen for coding. Suppose the source has an original frame rate of f_0 f/s and a spatial resolution of $M_1 \times M_2$ pixels per frame. The frame rate chosen for coding is given by

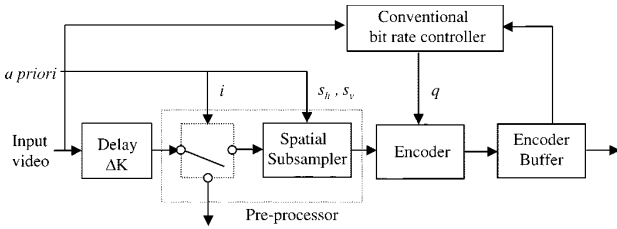


Fig. 2. Illustration of conventional bit-rate control approach. Controller adapts quantizer parameter q while frame rate parameter i and spatial subsampling parameters s_h, s_v are fixed at levels chosen *a priori*. The video enters the preprocessor with a delay of ΔK frames to achieve better bit allocation.

$$f_i = \frac{f_o}{i}, \quad i = 1, 2, \dots, i_{\max} \quad (1)$$

where i_{\max} represents the maximum allowable frame rate parameter. Similarly, the spatial resolution chosen for coding is given by $(M_1/s_h) \times (M_2/s_v)$, $s_h, s_v = 1, 2, \dots$. Subsampling the video in both the temporal and spatial dimensions prior to coding increases the bit allocation to pixels that are coded by a factor of $s_h \cdot s_v \cdot i$.

In the M-D approach, i, s_h, s_v and q are jointly adapted to control the buffer level as illustrated in Fig. 1. The conventional bit-rate control approach, a special case of the M-D approach, is illustrated in Fig. 2. In the conventional approach, q is adapted for buffer control while i, s_h and s_v remain fixed. The fixed levels of i, s_h and s_v are chosen *a priori* independent of the quantization performed during the coding process. The frame rate parameter i is typically determined based on experience and the spatial subsampling parameters s_h, s_v are often set to 1 for the luminance component and 2 for the chrominance components. A theoretical approach is taken in [13] to obtain the optimal frame rate. Since the frame rate and spatial subsampling parameters are chosen automatically during the coding process with the M-D approach, there is no need to choose these parameters *a priori*. In both the M-D and conventional approaches, any encoder can be used for compression.

B. Reconstruction Patterns

When the frame rate parameter i is selected, there are $i - 1$ skipped frames that must be reconstructed from coded frames. There are multiple ways to reconstruct the skipped frames from coded frames. This section establishes a fundamental framework that allows a skipped frame to be reconstructed from one coded frame through the choice of a reconstruction pattern. Using the frame rate parameter i , the controller can select from one of i reconstruction patterns defined and illustrated in Fig. 3. Reconstruction pattern n , for $0 \leq n \leq i - 1$, corresponds to using the previously coded frame to reconstruct the next n future skipped frames. With this set of reconstruction patterns, it is possible to obtain an optimal solution to the M-D buffer-constrained allocation problem for the case of intraframe coding.

The shaded frames in Fig. 3 will be referred to as boundary frames. A frame is a boundary frame if it has an adjacent frame which is reconstructed from a different coded frame. A coded frame is also a boundary frame when it is not used for backward

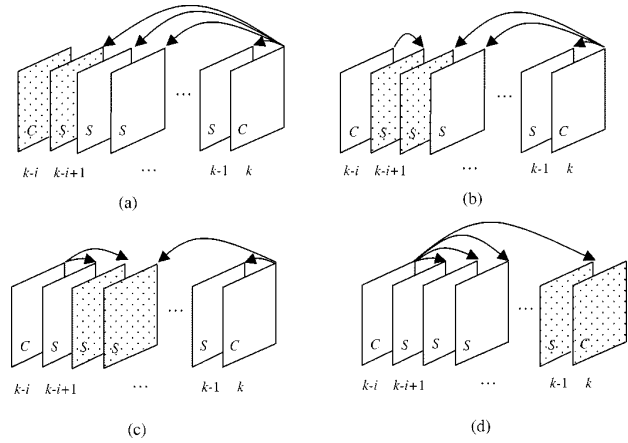


Fig. 3. Illustration of i reconstruction patterns between coded frames $k - i$ and k . Reconstruction pattern n , for $0 \leq n \leq i - 1$, corresponds to using frame $k - i$ to reconstruct n future skipped frames: (a) reconstruction pattern 0, (b) reconstruction pattern 1, (c) reconstruction pattern 2, and (d) reconstruction pattern $i - 1$. In the figure, we assume $i > 3$. Shaded frames represent boundary frames.

and/or forward reconstruction. It is convenient to illustrate the selected reconstruction patterns resulting from the optimization by showing the boundary frames.

When a frame is skipped, it is reconstructed at the receiver from a coded frame defined by the selected reconstruction pattern using some form of temporal interpolation. Typically, zero-order hold temporal interpolation is used. When zero-order hold temporal interpolation is used, the reconstruction patterns illustrated in Fig. 3 represent the most relevant patterns. Since the difference between skipped and coded frames generally increases with increasing distance, other reconstruction patterns are likely to result in a suboptimal solution. With additional complexity at the receiver, skipped frames can be reconstructed using motion-compensated temporal interpolation [14]–[16]. When motion-compensated temporal interpolation is used, bi-directional reconstruction may be useful, especially when motion vectors used to reconstruct skipped frames are estimated at the receiver from the motion detected between two coded frames. Bi-directional reconstruction is not considered in this paper and is left as a problem for future research. Allowing bi-directional reconstruction or additional reconstruction patterns significantly increases the complexity of the problem making it difficult to guarantee an optimal solution.

Typically, conventional and M-D bit-rate control algorithms reconstruct skipped frames using the reconstruction pattern in Fig. 3(d) which involves forward reconstruction only. In low delay applications, the frame reorder delay associated with the additional reconstruction patterns is not acceptable. However, in streaming video applications where a significant delay is tolerable, frame reorder delay is acceptable. In our experiments, the controller can select from any of the reconstruction patterns defined in Fig. 3 for both the M-D and conventional bit-rate control approaches. If desired, the optimization can be performed with any subset of reconstruction patterns. In the M-D case, the reconstruction patterns have a significant effect on the optimization. If the skipped frames are reconstructed more efficiently,

Authorized licensed use limited to: Fenwick & West. Downloaded on January 24, 2025 at 18:45:03 UTC from IEEE Xplore. Restrictions apply.

the number of coded frames resulting from the optimization will decrease.

C. Delay Considerations

The components of a generic video transmission system include the encoder, encoder buffer, transmission channel, decoder buffer and decoder. Delay is introduced into the system in a variety of ways, including:

- 1) delayed encoding ΔT_k ;
- 2) encoder processing delay ΔT_e ;
- 3) encoder buffer delay ΔT_{eb} ;
- 4) frame reorder delay ΔT_{fr} ;
- 5) channel transmission delay ΔT_c ;
- 6) decoder buffer delay ΔT_{db} ;
- 7) decoder processing delay ΔT_d .

Delay requirements depend on the application. In video transmission applications, decoding is performed in real-time while encoding can be performed either in real-time or non-real-time. In real-time encoding applications, communication can be interactive (e.g., video conferencing) where low delay is required or noninteractive (e.g., streaming live video) where delay requirements are relaxed. In nonreal-time encoding (e.g., streaming stored video), the video is transmitted from storage and similar to the noninteractive real-time encoding case, delay is introduced into the system only (ideally) at the beginning of transmission. Since the user notices delay only at the beginning of transmission, the delay can be significant (i.e., 200 ms or greater). The methods developed in this paper are most appropriate for streaming video where delay requirements are relaxed.

In the case of real-time encoding, the total end-to-end delay ΔT through the system is defined as the time at which a frame is generated to the time at which it is displayed. In this case, we are concerned with the delay introduced by delayed encoding, frame reordering and encoder/decoder buffering. If we assume that processing and channel transmission delay are negligible, the total end-to-end delay ΔT is given by

$$\Delta T = \Delta T_k + \Delta T_{fr} + \Delta T_{eb} + \Delta T_{db}. \quad (2)$$

Given a constant end-to-end delay ΔT , a frame input into the system at time t will be displayed at the receiver at time $t + \Delta T$. If $T = 1/f_o$ is the time interval for one video frame, $\Delta T/T$ represents the total end-to-end delay in video frames.

When encoding is performed in real-time, only a finite window of the entire sequence is known at each decision instant due to delay requirements. To account for the complexity of future video frames, the encoder can perform delayed encoding with a delay of ΔK frames. In this case, the encoder makes a decision for frame k using the knowledge of frames k to $k + \Delta K$ to achieve better bit allocation.

In the context of MPEG video [17], frame reorder delay occurs from the backward prediction associated with the use of bi-directionally predicted frames (B-frames). From Fig. 3, frame reorder delay occurs in our work from the use of backward reconstruction to reconstruct skipped frames from coded frames. Since the number of skipped frames reconstructed using backward reconstruction varies, the frame reorder delay

is variable. The total end-to-end delay can be made constant by setting it to the maximum level at the beginning of transmission. Therefore, we can use $\Delta T_{fr} = \Delta T_{fr, \max}$ in (2), where $\Delta T_{fr, \max}$ represents the maximum frame reorder delay imposed on the system. If the maximum allowable distance between coded frames is i_{\max} frames, then the maximum frame reorder delay is $i_{\max} - 1$ frames and $\Delta T_{fr, \max} \leq (i_{\max} - 1)T$.

At the beginning of transmission, buffer delay is introduced by having the decoder wait a certain time after the first bit of the bitstream is received before starting to decode it. The buffer delay in video frames is ΔL when the decoder waits ΔL frame intervals (or $\Delta L T$ s). A detailed analysis of buffer relationships can be found in [18] and [19].

Given a frame reorder delay of ΔK frames, a buffer delay of ΔL frames, and a maximum allowable distance between coded frames of i_{\max} frames

$$\Delta T \leq (\Delta K + \Delta L + i_{\max} - 1)T. \quad (3)$$

In the case of nonreal-time encoding, encoding is performed off-line and the total end-to-end delay ΔT through the system is defined as the time at which transmission begins to the time at which the first frame is displayed. Here, we are concerned with the delay introduced by frame reordering and decoder buffering. Assuming that decoder processing and channel transmission delay are negligible

$$\Delta T = \Delta T_{fr} + \Delta T_{db} \leq (\Delta L + i_{\max} - 1)T. \quad (4)$$

III. INTEGER PROGRAMMING FORMULATION

Since the encoding parameters and reconstruction patterns are integer variables, the M-D buffer-constrained allocation problem can be solved using techniques in the field of integer programming [20]. This section presents an integer programming formulation of the M-D buffer-constrained allocation problem. The formulation allows a skipped frame to be reconstructed from one coded frame using the reconstruction patterns in Fig. 3.

Suppose the controller can choose from I frame rate parameters, Q quantizer parameters, S_h horizontal spatial subsampling parameters and S_v vertical spatial subsampling parameters. Let i_{\max} denote the maximum frame rate parameter and let N denote the length of the video sequence. The combination of all parameters defines the set of operating points on a M-D grid. Let the index $j = 1, \dots, IQS_hS_v$ represent one of the operating points ordered into a 1-D vector.

Define $x(k)$ to be the index for the operating point used to code frame k . Coding frame k with operating point $x(k)$ produces a rate $r_{k, x(k)}$ ¹, distortion $d_{k, x(k)}$ and buffer state $B(k)$ given by

$$B(k) = B(k - i) + r_{k, x(k)} - iC, \quad k > 0 \quad (5)$$

where C is channel rate per frame and i is the frame rate parameter associated with operating point $x(k)$. If frame k is skipped, $x(k)$ is set to zero and $r_{k, 0} = 0$. Since overhead bits required

¹The rate includes the overhead bits required to specify that operating point $x(k)$ is selected.

to reconstruct a skipped frame are negligible, they are included with the rate of a coded frame.² Alternatively, the overhead bits can simply be neglected. In the interval between coded frames, the buffer state decreases linearly at the rate of C bits per frame. If the buffer state falls to zero at any given time, stuffing bits are used to maintain the buffer at the zero level. It is assumed that the first frame is always coded and the channel turns on after the bits for the first frame are released to the buffer. Therefore, $B(0) = B(-1) + r_{0,x(0)}$, where $B(-1)$ is the initial buffer state.

Since skipped frames are reconstructed from coded frames defined by the choice of a reconstruction pattern, the sequence $p(k)$ is introduced where $p(k)$ is set to k if frame k is coded and set to r if frame k is skipped and reconstructed from frame r . Therefore, $d_{k,x(p(k))}$ represents the distortion of frame k reconstructed from frame $p(k)$ which has been coded with operating point $x(p(k))$. If frame k is coded, it is reconstructed from itself (i.e., $x(p(k)) = x(k)$). Given i_{\max} , $p(k) \in [\max(k - i_{\max} + 1, 0), \dots, k, \dots, \min(k + i_{\max} - 1, N - 1)]$.

Formulation 2: (Integer Programming Formulation)

Given spatial and temporal interpolation methods to be used at the receiver, find the sequences $x(k)$ (operating points) for $k = 0, \dots, N - 1$ and $p(k)$ (reconstruction patterns) for $k = 1, \dots, N - 1$ that solves

$$\begin{aligned} \min \sum_{k=0}^{N-1} w_k d_{k,x(p(k))} \\ \text{subject to } B(k) \leq B_{\max}, \quad k = 0, \dots, N - 1 \end{aligned}$$

where w_k represents a temporal weighting factor for frame k , B_{\max} is the buffer size and $p(0) = 0$.

The weighting factors can be chosen to take into account perceptual effects. For example, the weights can be chosen from statistical measures such as those defined in [7] to account for temporal masking effects. Weights can also be chosen to achieve different tradeoffs between quantization noise and temporal resolution. For this purpose, it is useful to consider the weighted distortion metric given by

$$w \cdot \sum_{k \in \mathcal{C}} d_{k,x(p(k))} + (1 - w) \cdot \sum_{k \in \mathcal{C}'} d_{k,x(p(k))}, \quad w \in [0, 1] \quad (6)$$

where \mathcal{C} is the set of coded frames and \mathcal{C}' is the set of skipped frames. Coded frames can be weighted more heavily by setting $w > 1/2$. This has the effect of reducing the number of frames that are coded which, in turn, reduces quantization noise. This is useful when the temporal correlation is small. Setting $w_k = 1, \forall k$ results in the total unweighted distortion. It is worthwhile to mention that minimizing the maximum distortion of a frame does not yield a desirable solution at low bit rates. In this case, the algorithm will try to code every frame since skipped frames have the largest distortion.

²Overhead bits are required to specify the reconstruction patterns and any transmitted motion vectors in the case motion-compensated temporal interpolation is used. Overhead bits are non-negligible if multiple motion vectors are transmitted for each skipped frame. In this case, frames are no longer skipped.

Conventional bit-rate control is a special case of the M-D approach. To obtain an optimal solution for the conventional approach, the frame rate and spatial subsampling parameters are fixed at some specified level (i.e., $I = S_h = S_v = 1$) and the optimization is performed with respect to quantizer parameter and reconstruction pattern selection. In the special case of conventional bit-rate control where every frame is coded at full resolution, $p(k) = k$ and Formulation 2 reduces to the familiar formulations previously presented in the literature [1], [2], [19]. While we focus on the fixed-rate channel, Formulation 2 can be easily extended to the case of a variable-rate channel [1], [19].

IV. OPTIMAL ALGORITHM

In this section, we show that forward dynamic programming [21] can be used to solve the M-D buffer-constrained allocation problem for the case of intraframe coding which is a special case of dependent coding. The case of intraframe coding corresponds to coding frames independently of other coded frames, however, dependency still exists since skipped frames are reconstructed from coded frames. Section IV-A defines a trellis that represents all feasible buffer paths. Section IV-B presents the optimal algorithm. Section IV-C describes a M-D bit-rate control algorithm for the case of limited-lookahead. Finally, Section IV-D discusses the complexity of growing the trellis. While any additive distortion metric may be used, we will assume throughout this section for notational convenience that $w_k = 1, \forall k$.

A. Trellis

It is useful to begin by defining a trellis to represent all the feasible buffer paths. The following definitions describe the trellis.

- **Stage:** Each stage represents a frame that will be either skipped or coded.
- **Node:** Each node is a triplet (k, b, n) where $k \in 0, \dots, N - 1$ is the stage number, $b \in 0, \dots, B_{\max}$ is the buffer state and $n \in 0, \dots, \min(i_{\max} - 1, N - k - 1)$ represents the number of future skipped frames that are reconstructed from coded frame k . In the remainder of this section, we will assume for notational convenience, unless otherwise stated, that $\min(i_{\max} - 1, N - k - 1) = i_{\max} - 1$.
- **Branch:** A branch links a node in one stage with a node in another stage as illustrated in Figs. 4 and 5. If operating point j (which uses frame rate parameter i) at stage k has R-D characteristics $(r_{k,j}, d_{k,j})$, then node $(k - i, b, n)$ will be linked to node $(k, b + r_{k,j} - i \cdot C, m)$ by a branch of cost weight $d_{k-i+n+1,j} + \dots + d_{k,j} + \dots + d_{k+m,j}$, for $0 \leq n \leq i - 1, 0 \leq m \leq i_{\max} - 1$, provided no overflow occurs [see Fig. 4(a)]. Here, n corresponds to using reconstruction pattern n as illustrated in Fig. 4(b) between coded frames $k - i$ and k . Notice that the branch cost includes the distortion of coded frame k and all the skipped frames reconstructed from it.³
- **Path:** A path is a concatenation of branches (see Fig. 5). A feasible buffer path is a path linking nodes at the initial stage to nodes at the final stage.

³Frames in the interval $[k - i + n + 1, \dots, k + m]$ can be considered as a unit which is coded independently of other units. If bi-directional reconstruction is allowed, this is no longer the case.

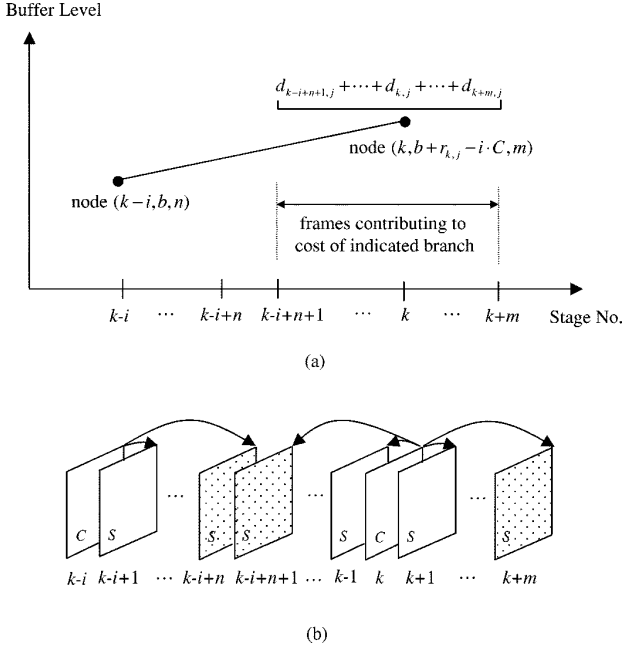


Fig. 4. Illustration of a branch linking node $(k-i, b, n)$ to node $(k, b+r_{k,j}-i.C, m)$ using operating point j with corresponding frame rate parameter i : (a) Using an unweighted distortion metric, the branch cost is given by $d_{k-i+n+1,j} + \dots + d_{k,j} + \dots + d_{k+m,j}$. (b) Corresponding reconstruction patterns and boundary frames. Frames contributing to cost of indicated branch include coded frame k and all skipped frames reconstructed from it.

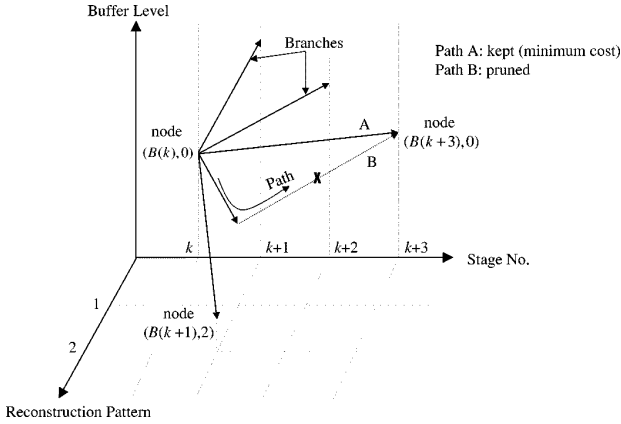


Fig. 5. Illustration of optimization. When the frame rate parameter i is used, nodes at stage k will be linked to nodes at stage $k+i$. Of all the paths arriving at a given node, only the minimum cost path has to be kept. For example, A is the minimum cost path arriving to node $(B(k+3), 0)$ at stage $k+3$. Therefore, path B can be pruned without loss of optimality.

B. Global Optimization: $\Delta K = N - 1$

Given an initial buffer state, the algorithm described below can be used to generate the shortest path through the trellis for any given final buffer state. In the special case of conventional bit-rate control where every frame is coded at full resolution, $i_{\max} = 1$ and our algorithm reduces to the algorithm in [2] which solves the case of purely independent coding. Fig. 5 illustrates the optimization.

Authorized licensed use limited to: Fenwick & West. Downloaded on January 24, 2025 at 18:45:03 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1: (Global Optimization)

Step 0: Choose an initial buffer state $B(-1)$. The algorithm begins by coding the first frame with all quantizer and spatial subsampling parameter combinations. For each parameter combination, the first frame is used to reconstruct the next n frames, $0 \leq n \leq i_{\max} - 1$, to populate all the achievable nodes at stage 0. If any parameter combinations achieve the same rate, only the combination producing the minimum distortion will be kept. Set the stage count k to zero.

Step 1: At stage k add permissible branches (no buffer overflow) to the end nodes of all surviving paths. At each node, a branch is grown for all operating points and reconstruction patterns, and the cost of that branch is added to the total accumulated cost of the path arriving to the node in a future stage. If an operating point has a frame rate parameter i , branches will be grown linking nodes at stage k with nodes at stage $k+i$. If $k+i > N-1$, then branches will be grown linking nodes $(k, b, N-k-1)$ with nodes $(N-1, b-(N-k-1).C, 0)$.

Step 2: Of all the paths arriving at a node in stage $k+i$, the minimum cost path is chosen and the rest are pruned. Note that a path surviving the current iteration may be pruned in a future iteration.

Step 3: Increment k by 1 and go to **Step 1** until $k = N - 1$.

Of all the paths arriving at a given node, only the path with the minimum aggregate cost may be part of the overall optimal solution. Paths with higher aggregate cost cannot be part of the overall optimal solution. The aggregate cost of a path arriving at node (k, b, n) represents the total distortion of reconstructed frames $0, \dots, k+n$. With the reconstruction patterns defined in Fig. 3, the distortion of future reconstructed frames $k+n+1, \dots, N-1$ is independent of the distortion of the first $k+n+1$ frames. Since all paths arriving at node (k, b, n) have the same resources available to code future frames $k+n+1, \dots, N-1$, a path with higher aggregate distortion can be discarded without loss of optimality.

C. Limited-Lookahead Optimization: $\Delta K \ll N - 1$

To obtain an optimal solution, one needs to grow the full trellis before allocating bits to any frame. For a length N sequence, a trellis of depth N is generated which requires the entire sequence to be available for processing. With real-time encoding, only a finite window of the entire sequence is known at each decision instant due to delay requirements. For this case, the optimization can be performed in a sliding window fashion where paths are grown and released based on a limited number of frames. The optimal solution obtained using Algorithm 1 can be used as a benchmark to assess performance.

Suppose the encoder performs delayed encoding with a delay of ΔK frames. In this case, a decision for frame k is determined based on the optimal path from k to $k+\Delta K$. A decision involves determining whether frame k is coded or skipped and whether it is reconstructed using backward or forward reconstruction in the case it is skipped. The following algorithm can be used to generate a feasible buffer path with limited-lookahead. In the algorithm, we assume $\Delta K < N - 1$.

Algorithm 2: (Limited-Lookahead Optimization)

Step 0: Choose an initial buffer state $B(-1)$. The first frame is coded as follows: Determine the optimal path through the trellis from stages 0 to ΔK for some final buffer state. The encoding parameters used for the first frame in the chosen optimal path is final. Set the stage count k to 1. Set the last coded frame l to zero.

Step 1: Determine the optimal path through the trellis from stages k to $\min(k + \Delta K, N - 1)$ for some final buffer state. The trellis is grown starting from stage l with buffer state defined by the recursion in (5). Let k' represent the first coded frame in the chosen optimal path.

Step 2: If $\min(k + \Delta K, N - 1) = N - 1$, the decision for the remaining $N - k - 1$ frames in the chosen optimal path is final and the algorithm terminates. Otherwise, repeat **Step 1** with k and l determined as follows: If frame k is skipped and reconstructed from frame l using forward reconstruction in the chosen optimal path, the corresponding decision for frame k is final. In this case, k is incremented to $k + 1$ and l is unchanged. If frame k is coded (i.e., $k' = k$), the corresponding decision for frame k is final and both k and l are incremented by 1. If frame k is skipped and reconstructed from frame k' using backward reconstruction in the chosen optimal path, the corresponding decision for frames $k, k + 1, \dots, k'$ is final. In this case, k is incremented to $k' + 1$ and l is incremented to k' .

When an optimal path is chosen from k to $k + \Delta K$ in Algorithm 1, the choice of the final buffer state at stage $k + \Delta K$ is arbitrary. As a result, increasing ΔK does not guarantee a lower global cost.

D. Complexity

In this section, we estimate the order of complexity of growing the trellis for the M-D and conventional bit-rate control approaches. The order of complexity refers to the number of comparisons that need to be performed to compute an optimal solution. Let B_{\max} denote the buffer size, i_{\max} denote the maximum frame rate parameter and N denote the length of the video sequence. Suppose there are I frame rate parameters, Q quantizer parameters, S_h horizontal spatial subsampling parameters and S_v vertical spatial subsampling parameters.

Let us first consider the M-D approach. There are at most $B_{\max}N i_{\max}$ nodes in the trellis to be considered. A branch is grown to each node for all feasible operating points. There are a total of IQS_hS_v operating points. For a given operating point, there are at most i_{\max} reconstruction patterns to consider. Therefore, the order of complexity of growing the trellis for the M-D approach is given by

$$C_{M-D} = \mathcal{O}(B_{\max}NIQS_hS_v i_{\max}^2). \quad (7)$$

For the conventional approach, there are at most $B_{\max}N$ nodes in the trellis to be considered due to uniform subsampling in time. There are a total of Q operating points and i_{\max} reconstruction patterns to consider for each operating point.

Therefore, the order of complexity of growing the trellis for the conventional approach is given by

$$C_C = \mathcal{O}(B_{\max}NQi_{\max}). \quad (8)$$

To gain some insight into the complexity, assume B_{\max} and N are (approximately) 10^4 and 10^2 , respectively. Also, assume that I , Q , S_hS_v , and i_{\max} are all (approximately) 10. In this case, the number of comparisons needed to compute an optimal solution for the M-D and conventional bit-rate control approaches is (approximately) 10^{11} and 10^8 , respectively. The complexity of the M-D approach is roughly three orders of magnitude larger.

It is possible to reduce complexity at the cost of a slightly sub-optimal solution by reducing the number of nodes in the trellis. As stated earlier, each node is defined as a triplet (k, b, n) . In one approach, the number of nodes can be reduced by buffer state clustering as discussed in [2]. For a fixed value of n , only the minimum cost path of those arriving to a set of buffer states in a local neighborhood is kept. The clustering factor determines the size of the neighborhood. The number of nodes can also be reduced by ignoring the dependency introduced with frame skipping. In this case, only one path (i.e., the minimum cost path for $n = n_0$) of those arriving to a given buffer state is kept. Each node then becomes a pair (k, b) and the number of nodes is reduced by a factor of i_{\max} .

V. RESULTS

This section illustrates the optimization results for the M-D and conventional bit-rate control approaches. In all experiments, the objective is to minimize the total unweighted distortion with the sum of absolute error (SAE) as the distortion measure. Mean square error (MSE) is not used since the squaring operation places more emphasis on the larger distortion associated with skipped frames which results in the coding of more frames.

The encoder used in the experiments is similar to H.263 [22] with all advanced options turned off. We experiment only with the luminance component of the test sequences. Therefore, the overhead associated with the chrominance components is removed. Zero-order hold temporal interpolation is used to reconstruct skipped frames from coded frames and bilinear interpolation is used to reconstruct coded frames that are spatially subsampled [23]. We experiment with global motion-compensated temporal interpolation in [10]. In all experiments, the initial buffer state is set to zero (i.e., $B(-1) = 0$) and the buffer size is set to $B_{\max} = \Delta L \cdot C$, for some integer ΔL corresponding to a buffer delay of ΔL frames.

We experiment with two video sequences which have a length of 80 frames (i.e., $N = 80$) and size of 160×128 pixels⁴ at 30 f/s and 8 bits/pixel. Therefore, the raw data rate is approximately 5 Mb/s. The two test sequences will be referred to as *Carphone* and *Resource*. *Carphone* is the well-known head-and-shoulders sequence with no scene changes and *Resource* is a movie trailer with two scene changes. Relative to each other, *Carphone* is inactive while *Resource* is highly active with varying characteristics in different scenes. The first scene change occurs between

⁴The sequences were clipped from their original QCIF versions.

frames 23, 24 and the second scene change occurs between frames 65, 66. The first scene has the highest activity (lowest temporal correlation) while the middle scene has the lowest activity (highest temporal correlation).

The added flexibility of the M-D bit-rate control approach allows the controller to be more adaptive to the source characteristics. One would expect the benefit of this added flexibility to increase with the variability of the source characteristics. As a result, one should expect the M-D approach to provide larger coding gains over the conventional approach for *Resource*.

A. Intraframe Coding Experiments

In these experiments, all coded frames are intraframes (I-frames). We consider bits rates ranging from 20–100 kb/s corresponding to compression factors ranging from 50–250. With the M-D approach, the controller can choose from 1) the set of frame rate parameters given by $i \in \{1, \dots, i_{\max}\}$ for some specified i_{\max} ; 2) the set of spatial subsampling parameters given by $s_h, s_v \in \{1, 2\}$, allowing each coded frame to be subsampled by factor of 1 (no subsampling) or 2 in either direction; and 3) the set of quantizer parameters to be used for each coded frame given by $q \in \{1, \dots, 31\}$, ordered from finest to coarsest. With the conventional approach, $i, s_h,$ and s_v are fixed at specified levels and the same set of quantizer parameters are used for control.

1) *Global Optimization*: This section compares the optimal solution of the M-D and conventional bit-rate control approaches using Algorithm 1. Since the entire sequence is assumed to be known, the total end-to-end delay ΔT is given by (4). To make a fair comparison, the delay ΔT is set equal for the two approaches at any given bit rate.

To obtain the same total delay ΔT , the optimization is first performed for the M-D approach with a given ΔL . The total delay is then given by the sum of the buffer and maximum frame reorder delay resulting from the optimization. For a given buffer size, the maximum frame reorder delay imposed on the system tends to decrease with increasing channel rate since more frames are coded as the channel rate increases. Once the total delay is determined for the M-D approach, ΔL is chosen (typically increased) for the conventional approach such that the total delay is equal to that achieved with the M-D approach. Using the conventional approach with frame rate parameter $i < i_{\max}$, the maximum possible frame reorder delay is $i - 1$ frames. If the maximum frame reorder delay imposed on the system is larger than $i - 1$ frames using the M-D approach, the buffer size will be increased to achieve the same total delay.

A comparison of the operational R-D bounds for the M-D and conventional bit-rate control approaches is shown in Figs. 6 and 7. The horizontal axis represents the channel transmission rate and the vertical axis represents the average SAE over a frame. Operational R-D bounds are obtained by choosing the final buffer state that yields the minimum global cost. R-D curves are shown for the conventional approach at three different frame rates ($i = 3, 4, 6$) with s_h and s_v set to 1. R-D data that is missing in the figures for the conventional approach at the low rates indicates that no solution exists for the selected

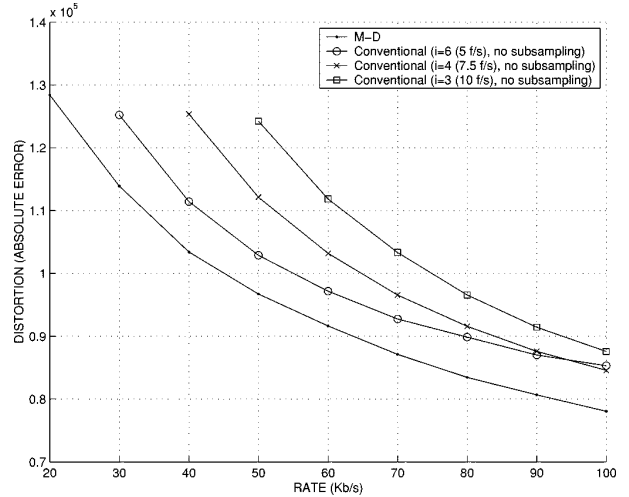


Fig. 6. Operational R-D bounds for *Carphone*. M-D approach ($\Delta L = 10$, $i_{\max} = 9$) and conventional approach for $i = 3, 4, 6$ with $s_h = s_v = 1$.

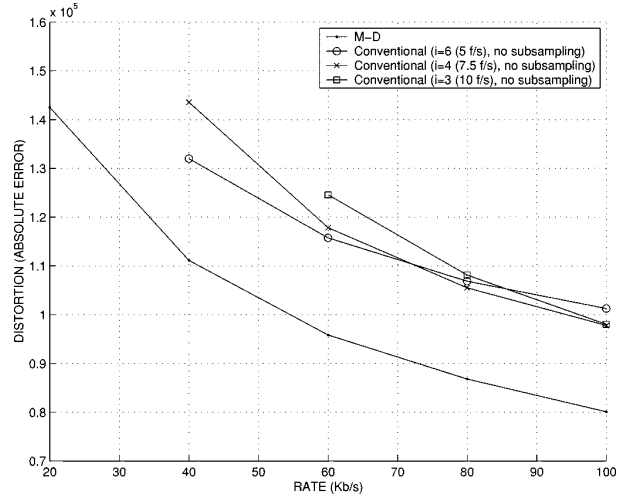


Fig. 7. Operational R-D bounds for *Resource*. M-D approach ($\Delta L = 10$, $i_{\max} = 9$) and conventional approach for $i = 3, 4, 6$ with $s_h = s_v = 1$.

video format at the given rate. The R-D curves for the M-D approach were generated with $\Delta L = 10$ and $i_{\max} = 9$.⁵ Then, ΔL is chosen for the conventional approach at each bit rate to achieve the same total end-to-end delay as achieved with the M-D approach. For example, the frame reorder delay using the M-D approach at 100 kb/s for *Carphone* is 5 frames (see Fig. 8). Hence, the total buffer and frame reorder delay is 15 frames. To compare the M-D approach with the conventional approach at 100 kb/s and $i = 4$, ΔL is set to at least 12 since the maximum possible frame reorder delay is 3 frames when $i = 4$.

Figs. 6 and 7 illustrate that significant coding gains are achieved with the M-D approach. Fig. 6 shows bit-rate

⁵Given $B_{\max} = \Delta L \cdot C$, $i_{\max} \leq \Delta L$ to prevent encoder buffer underflow. If $i_{\max} > \Delta L$ between two coded frames, the encoder buffer would underflow resulting in inefficient use of the channel.

TABLE I
OPTIMAL VIDEO FORMAT FOR *CARPHONE* USING M-D BIT RATE CONTROL WITH $\Delta L = 10$ AND $i_{\max} = 9$ AS A FUNCTION OF BIT RATE

Channel rate (kb/s)	Number of frames coded	Spatial resolution of coded frames ($s_h \times s_v$ subsampling)		
		1x1 subsampling	2x1 or 1x2 subsampling	2x2 subsampling
20	12	0	8	4
40	12	8	3	1
60	13	12	1	0
80	16	15	1	0
100	18	18	0	0

TABLE II
OPTIMAL VIDEO FORMAT FOR *RESOURCE* USING M-D BIT RATE CONTROL WITH $\Delta L = 10$ AND $i_{\max} = 9$ AS A FUNCTION OF BIT RATE

Channel rate (kb/s)	Number of frames coded	Spatial resolution of coded frames ($s_h \times s_v$ subsampling)		
		1x1 subsampling	2x1 or 1x2 subsampling	2x2 subsampling
20	14	0	6	8
40	22	4	4	14
60	23	5	8	10
80	23	7	13	3
100	26	8	17	1

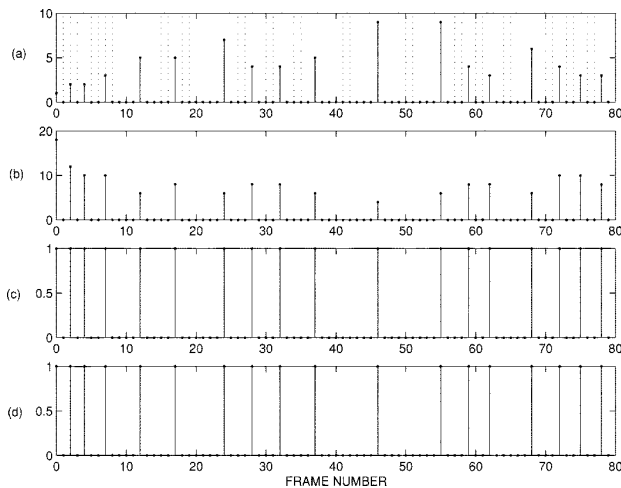


Fig. 8. Optimal parameter and reconstruction pattern selection for *Carphone* using M-D bit-rate control with $\Delta L = 10$ and $i_{\max} = 9$ at 100 kb/s. (a) Frame rate parameter and boundary frames (represented by dotted lines), (b) quantizer parameter, (c) horizontal and (d) vertical spatial subsampling parameters (2 = subsampled, 1 = not subsampled). Frame reorder delay is 5 frames due to backward reconstruction of skipped frames 19–23 from coded frame 24.

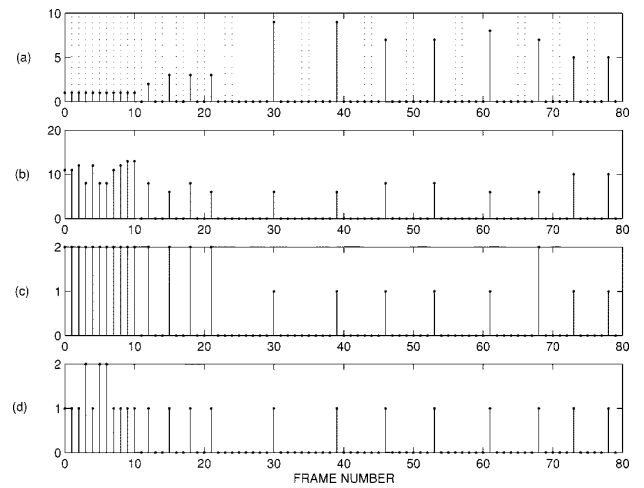


Fig. 9. Optimal parameter and reconstruction pattern selection for *Resource* using M-D bit-rate control with $\Delta L = 10$ and $i_{\max} = 9$ at 80 kb/s. (a) Frame rate parameter and boundary frames (represented by dotted lines), (b) quantizer parameter, (c) horizontal and (d) vertical spatial subsampling parameters (2 = subsampled, 1 = not subsampled). Frame reorder delay is 6 frames due to backward reconstruction of skipped frames 24–29 from coded frame 30. Frames 23, 24 and 65, 66 represent scene change boundaries.

reductions ranging from 20% to 50% for *Carphone* and Fig. 7 shows bit-rate reductions ranging from 30% to above 50% for *Resource*. The larger gains for *Resource* are due to larger variations of the source characteristics.

The M-D bit-rate control approach automatically determines the optimal video format. Tables I and II show the optimal number of coded frames and the chosen spatial resolution of the coded frames as a function of channel rate for *Carphone* and *Resource*, respectively. The tables show that the spatial resolution of the coded frames tends to increase with higher channel rates. They also show that the optimal number of coded frames increases with higher channel rates. This relationship can also be seen in Figs. 6 and 7 by focusing on the R-D curves of the conventional bit-rate control approach. Notice that lower frame rates perform better at lower channel rates and higher frame rates perform better at higher channel rates. This is the reason why the curves intersect at some bit rate.

Fig. 8 illustrates the optimal parameter and reconstruction pattern selection using the M-D bit-rate control for *Carphone* at 100 kb/s. Similarly, Fig. 9 illustrates the optimal parameter and reconstruction pattern selection using the M-D bit-rate control for *Resource* at 80 kb/s. In the figures, all the parameters are set to zero if a frame is skipped. Figs. 8(a) and 9(a) show that the optimal solution skips more frames when the temporal correlation is high and codes more frames when the temporal correlation is low. For example, Fig. 9(a) shows that the smallest frame rate parameters are selected in the first scene, which has the lowest temporal correlation, and the largest frame rate parameters are selected in the middle scene, which has the highest temporal correlation. It is also worthwhile to notice in Fig. 9 that the optimal algorithm invokes spatial subsampling in regions of low temporal correlation. When the temporal correlation is low, the algorithm codes more frames which results in less bits allocated

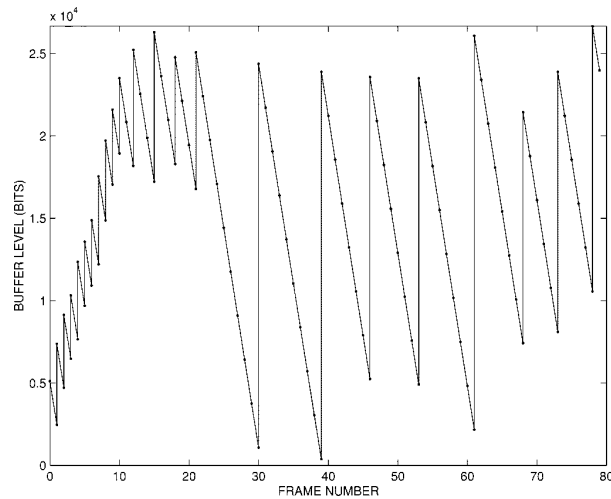


Fig. 10. Optimal buffer path for *Resource* with $B_{\max} = 26,667$ ($\Delta L = 10$) and $i_{\max} = 9$ at 80 kb/s.

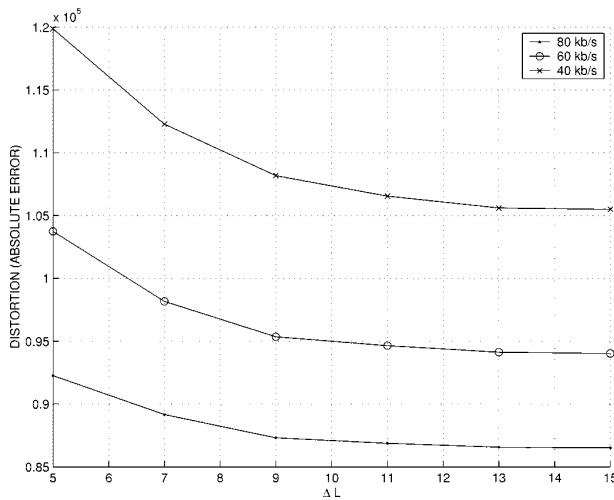


Fig. 11. Performance of M-D bit-rate control for *Carphone* as a function of buffer size ($B_{\max} = \Delta L \cdot C$) at 40, 60 and 80 kb/s with $i_{\max} = \Delta L$.

to each coded frame. Rather than using coarse quantization, the algorithm favors spatial subsampling with finer quantization.

Figs. 8 and 9 illustrate that the optimal solution allocates the largest number of bits to coded frames with the largest dependency range (i.e., coded frames used to reconstruct the most skipped frames) when the objective function is the total unweighted distortion. They also illustrate that the optimal algorithm tends to use a reconstruction pattern that assigns a skipped frame to the nearest coded frame (see Fig. 3(c)). In general, the difference between a skipped and coded frame increases as their distance increases. For example, consider frames 30–39 in Fig. 9(a). Skipped frames 31–34 are reconstructed from coded frame 30 while skipped frames 35–38 are reconstructed from coded frame 39.

Fig. 10 illustrates the optimal buffer path corresponding to the optimal parameter selection in Fig. 9. The buffer path follows the recursion in (5). The discontinuities or jumps represent bits

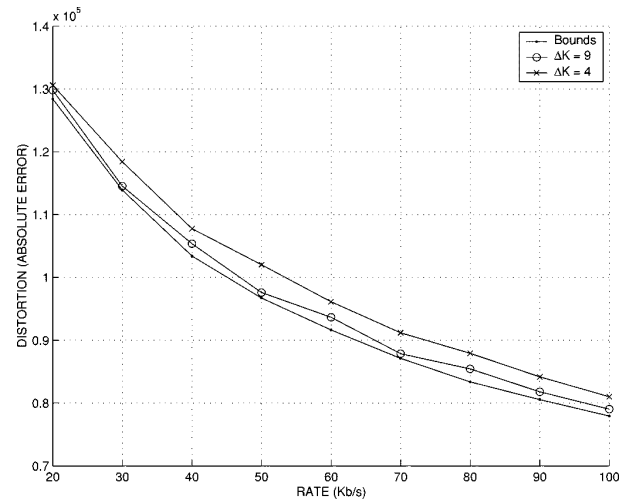


Fig. 12. Performance of M-D bit-rate control approach for *Carphone* with $\Delta L = 10$, $i_{\max} = 9$ and $\Delta K = \{4, 9\}$. Operational R-D bounds serve as a benchmark.

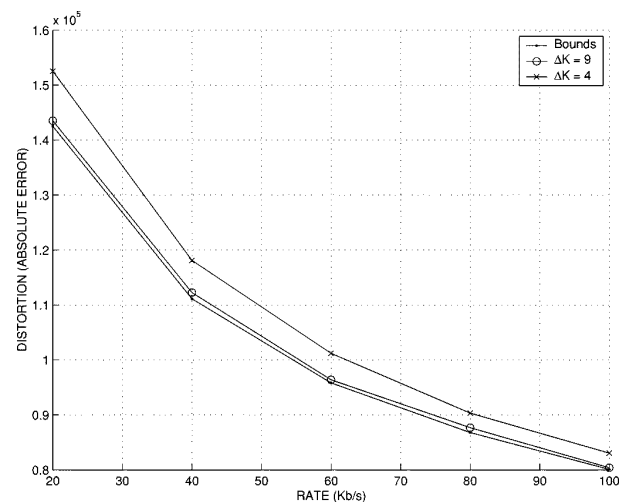


Fig. 13. Performance of M-D bit-rate control approach for *Resource* with $\Delta L = 10$, $i_{\max} = 9$ and $\Delta K = \{4, 9\}$. Operational R-D bounds serve as a benchmark.

of a coded frame being instantaneously placed into the buffer, while the steady declines between coded frames represent bits being extracted from the buffer at the channel rate. The figure illustrates that the optimal algorithm uses the full dynamic range of the buffer. In addition, since the goal is to minimize distortion, the optimal algorithm tries to prevent buffer underflow and therefore utilizes the channel resources efficiently.

All the results in this section have been generated with $\Delta L = 10$. Fig. 11 illustrates the R-D performance of the M-D bit-rate control approach as a function of ΔL for *Carphone* at various bit rates. The curves were generated by choosing the final buffer state that yields the minimum global cost with an imposed total budget constraint of $N \cdot C$ bits. The figure illustrates that the marginal gain of increasing the buffer size decreases as the buffer size grows. As the buffer size grows, the buffer constraints eventually become irrelevant. When this happens, the

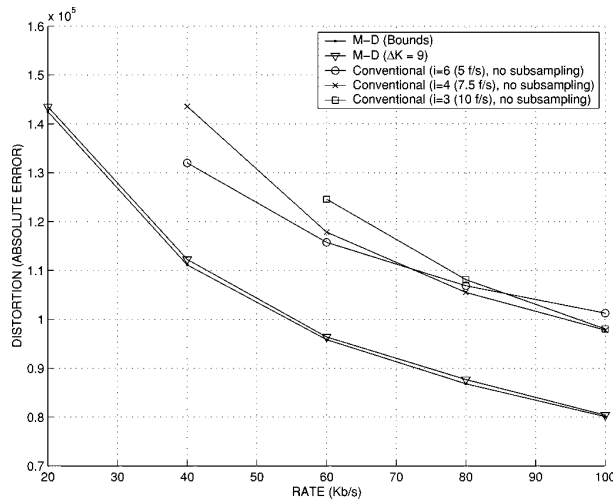


Fig. 14. M-D approach with limited-lookahead outperforms conventional approach with full-lookahead for *Resource*. R-D curves of conventional approach represent full-lookahead case.

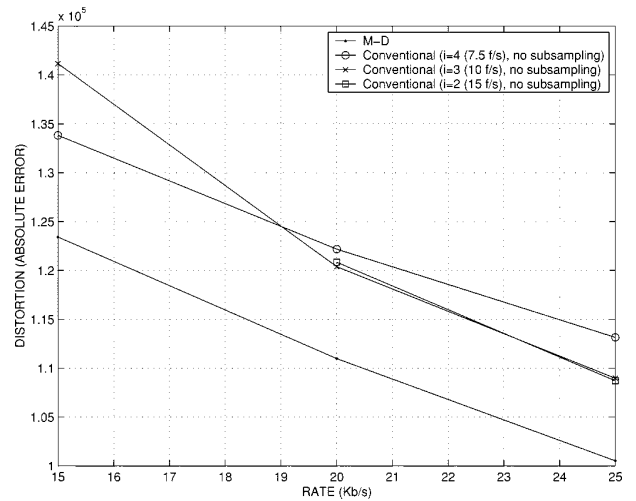


Fig. 16. Interframe coding R-D curves for *Resource* using Algorithm 1. M-D approach ($\Delta L = 10$, $i_{\max} = 9$) and conventional approach for $i = 2, 3, 4$ with $s_h = s_v = 1$.

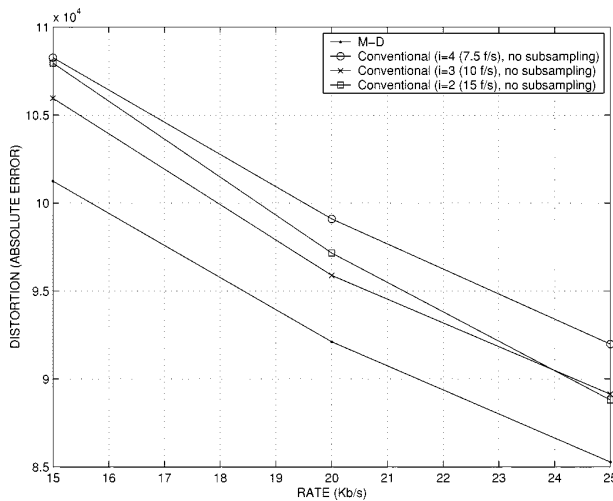


Fig. 15. Interframe coding R-D curves for *Carphone* using Algorithm 1. M-D approach ($\Delta L = 10$, $i_{\max} = 9$) and conventional approach for $i = 2, 3, 4$ with $s_h = s_v = 1$.

optimal budget-constrained solution is reached and the marginal gain becomes zero. It is also worth noting that increasing ΔL has the effect of reducing the number of coded frames.

2) *Limited-Lookahead Optimization*: Up to now, we have only considered the performance of the M-D approach assuming full knowledge of the source. In this section, we consider performance with limited-lookahead using Algorithm 2. In this case, the total end-to-end delay ΔT is given in (2). In each iteration of Algorithm 2, the final buffer state that yields the minimum distortion is chosen. Similar to the experiments in Section IV, ΔL is set to 10 and i_{\max} is set to 9. The operational R-D bounds obtained from Algorithm 1 are used as a benchmark.

The results for *Carphone* and *Resource* are illustrated in Figs. 12 and 13 with $\Delta K = \{4, 9\}$. Operational R-D bounds ($\Delta K = N - 1$) are also shown in the figures to compare performance. The figures illustrate that a slightly suboptimal

solution is obtained with a limited-lookahead of 9 frames which demonstrates that it is not necessary to increase the delay beyond 9 frames. These results illustrate the finite memory of the problem [2]. Since the allocation of bits for the first few frames is less likely to be influenced by the allocation of bits for the last frames as the sequence length grows, the allocations can be chosen independently. The memory of the problem increases with increasing buffer size since there are more buffer states at each stage.

Fig. 14 combines the full- and limited-lookahead results of the M-D and conventional bit-rate control approaches to illustrate that the M-D approach with a limited-lookahead of 9 frames consistently outperforms the conventional approach with full-lookahead.

B. Interframe Coding Experiments

This section presents results of the M-D and conventional bit-rate control approaches for the case of interframe coding. We study the case where the first coded frame is an I-frame and all other coded frames are predicted from the previously coded frame (i.e., P-frames). However, I-frames can be inserted at any desired location (e.g., scene changes) to restart the prediction loop. In addition, B-frames can be inserted to achieve more efficient compression. A solution is obtained using Algorithm 1. Predictive coding dependency is accounted for in Step 1, however, an independent allocation strategy is employed in Step 2 when paths are pruned. Using Algorithm 1 is attractive since the global optimization results in efficient bit allocation.

In Step 1 of Algorithm 1, branches are grown from the end nodes of all surviving paths. Each surviving path at stage k corresponds to a unique allocation of bits to all previously coded frames including frame k . Due to predictive coding dependency, branch costs grown from the end nodes of surviving paths depend on the previously coded frames which vary with each surviving path. When a branch is grown connecting nodes in stages k and $k + i$, frame $k + i$ is predicted from coded frame k which

TABLE III
INTERFRAME CODING VIDEO FORMAT FOR *CARPHONE* USING M-D BIT RATE CONTROL WITH $\Delta L = 10$ AND $i_{\max} = 9$ AS A FUNCTION OF BIT RATE

Channel rate (kb/s)	Number of frames coded	Spatial resolution of coded frames ($s_h \times s_v$ subsampling)		
		1x1 subsampling	2x1 or 1x2 subsampling	2x2 subsampling
15	33	30	2	1
20	40	39	0	1
25	48	47	1	0

TABLE IV
INTERFRAME CODING VIDEO FORMAT FOR *RESOURCE* USING M-D BIT RATE CONTROL WITH $\Delta L = 10$ AND $i_{\max} = 9$ AS A FUNCTION OF BIT RATE

Channel rate (kb/s)	Number of frames coded	Spatial resolution of coded frames ($s_h \times s_v$ subsampling)		
		1x1 subsampling	2x1 or 1x2 subsampling	2x2 subsampling
15	38	17	13	8
20	44	23	14	7
25	47	29	12	6

is different for each surviving path. To obtain the R-D characteristics associated with each surviving path that account for the predictive coding dependency, the most recent coded frame of every surviving path is stored in memory. Once all branches are grown from a given surviving path, the associated frame is removed from memory. To reduce the number of frames that must be stored in memory and the number of R-D data points that need to be computed, the buffer states are clustered by a factor of 100 and frame skipping dependency is ignored by retaining only one path for each buffer state. Reducing the number of nodes leads to little performance loss.

In Step 2 of Algorithm 1, pruning at each node may result in a suboptimal solution since the R-D characteristics of future coded frames depend on the allocation of bits given to frame $k+i$. By pruning at each node in stage $k+i$, the algorithm allocates bits to frame $k+i$ independent of this future dependency (i.e., an independent allocation approximation). The benefit of pruning is that it prevents the complexity of the problem from increasing exponentially. At the cost of increased complexity and memory requirements, more than one path can be kept at each node.

In these experiments, the M-D controller can choose from 1) the set of frame rate parameters given by $i \in \{1, \dots, i_{\max}\}$ for some specified i_{\max} ; 2) the set of spatial subsampling parameters given by $s_h, s_v \in \{1, 2\}$, allowing each coded frame to be subsampled by factor of 1 (no subsampling) or 2 in either direction; and 3) the set of quantizer parameters given by $q \in \{8, 10, 12, 15, 18, 21, 25, 31\}$ and $q \in \{6, 8, 10, 12, 14, 16, 20, 31\}$ for I- and P-frames, respectively, ordered from finest to coarsest. With the conventional approach, i , s_h , and s_v are fixed at specified levels and the same set of quantizer parameters are available for control.

A comparison of the R-D curves for the M-D and conventional bit-rate control approaches is shown in Figs. 15 and 16. R-D curves are shown for the conventional approach at three different frame rates ($i = 2, 3, 4$) with s_h and s_v set to 1. R-D data that is missing in the figures for the conventional approach at the low rates indicates that no solution exists for the selected video format at the given rate. The R-D curves of the M-D approach were generated with $\Delta L = 10$ and $i_{\max} = 9$. Then, the R-D curves of the conventional approach were generated to achieve the same total end-to-end delay using the methods discussed in Section V-A.

Figs. 15 and 16 illustrate that significant coding gains are achieved with the M-D approach. Fig. 15 shows bit-rate reductions ranging from 10% to 20% for *Carphone* and Fig. 16 shows bit-rate reductions ranging from 15% to 25% for *Resource*. Similar to the results obtained with the intraframe coding case, larger gains are obtained with *Resource* due to larger variations of the source characteristics. Since interframe coding exploits temporal correlations in the source, one can expect smaller coding gains compared to the gains achieved with intraframe coding.

Tables III and IV show the number of coded frames and the chosen spatial resolution of the coded frames as a function of channel rate for *Carphone* and *Resource*, respectively. The tables show that the spatial resolution of the coded frames tends to increase with higher channel rates. They also show that the number of coded frames increases with higher channel rates.

VI. CONCLUSION

Many ad hoc M-D bit-rate control algorithms have been proposed in the past. In this paper, we formalized the M-D bit-rate control problem and developed a dynamic programming algorithm to compute an optimal solution. Our algorithm can be directly used for nonreal-time encoding, for benchmarking, and as an aid in the development of suboptimal algorithms. While our algorithm is optimal only for the intraframe coding case, it can be used to provide a solution for more complex scenarios such as interframe coding.

The work presented in this paper provides the foundation for some interesting future research. For example, it is interesting to consider the M-D bit-rate control problem when channel rates can be chosen by the user under ATM policing constraints [19]. In this case, the problem is to jointly select the source and channel rates to optimize the quality of the transmitted video subject to source buffer and network policy constraints. Extending the work in [19], the optimal solution to this problem can be obtained by extending the trellis defined in this paper where each node would represent a quadruplet rather than a triplet. The added dimension represents the state of the policy function defined by the choice of the channel rates.

The multiplexing of two or more video sources is another area for future research. In the case of MPEG-4 where a video sequence is comprised of multiple video objects, operating points

can be chosen separately for each video object. The methods developed in this paper can be used to jointly select the operating points for each object to obtain an optimal bit allocation under a buffer constraint. Since objects can take on different frame rates, the composition problem would need to be addressed [24].

In addition, the conventional budget-constrained allocation problem [25] can be generalized using the same principles discussed in this paper by defining the M-D budget-constrained allocation problem. Other interesting extensions include the use of bi-directional reconstruction and macroblock level encoding decisions.

ACKNOWLEDGMENT

The authors would like to acknowledge Prof. D. Freeman, Massachusetts Institute of Technology, Cambridge, for many valuable conversations. The authors also acknowledge the anonymous reviewers for their useful comments.

REFERENCES

[1] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Processing Mag.*, vol. 15, pp. 23–50, Nov. 1998.

[2] A. Ortega, K. Ramchandran, and M. Vetterli, "Optimal trellis-based buffered compression and fast approximations," *IEEE Trans. Image Processing*, vol. 3, pp. 23–50, Jan. 1994.

[3] J.-R. Corbera and S. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 172–185, Feb. 1999.

[4] E. Reed, "Multi-dimensional bit rate control for video communication," Ph.D. dissertation, Mass. Inst. Technol., Cambridge, 2001.

[5] H. Song, J. Kim, and C.-C. Kuo, "Improved H.263+ rate control via variable frame rate adjustment and hybrid I-frame coding," in *Int. Conf. Image Processing*, vol. 2, Oct. 1998.

[6] —, "Real-time encoding frame rate control for H.263+ video over the internet," *Signal Process.—Image Commun.*, vol. 15, pp. 127–148, Sept. 1999.

[7] J. Lee and B. Dickinson, "Temporally adaptive motion interpolation exploiting temporal masking in visual perception," *IEEE Trans. Image Processing*, vol. 3, pp. 513–526, Sept. 1994.

[8] H.-M. Hang and J.-J. Chen, "Source model for transform video coder and its application—Part II: Variable frame rate coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 299–311, Apr. 1997.

[9] J. Zdepski, D. Raychaudhuri, and K. Joseph, "Statistically based buffer control policies for constant rate transmission of compressed digital video," *IEEE Trans. Commun.*, vol. 39, pp. 947–957, June 1991.

[10] E. Reed and J. S. Lim, "Multidimensional bit rate control for video communication," *Proc. SPIE*, vol. 4115, pp. 102–112, July 2000.

[11] D. W. Lin, M.-H. Wang, and J.-J. Chen, "Optimal delayed-coding of video sequences subject to a buffer-size constraint," *Proc. SPIE*, vol. 2094, pp. 223–234, Nov. 1993.

[12] *MPEG-4 Verification Model 7. Coding of Moving Pictures and Associated Audio*, ISO-IEC/JTC1/SC29/WG11, Mar. 1997.

[13] Y. Takishima, M. Wada, and H. Murakami, "An analysis of optimal frame rate in low bit rate video coding," *IEICE Trans. Commun.*, vol. E76-B, no. 11, pp. 1389–1397, November 1993.

[14] R. Krishnamurthy, J. W. Woods, and P. Moulin, "Frame interpolation and bidirectional prediction of video using compactly encoded optical-flow fields and label fields," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 713–726, Aug. 1999.

[15] C. K. Wong and O. C. Au, "Fast motion compensated temporal interpolation for video," *Proc. SPIE*, vol. 2, pp. 1108–1118, May 1995.

[16] C. W. Tang and O. C. Au, "Comparison between block-based and pixel-based temporal interpolation for video coding," in *IEEE Int. Symp. Circuits and Systems*, vol. 4, 1998, pp. 122–125.

[17] J. Mitchell, W. Pennebaker, C. Fogg, and D. LeGall, *MPEG Video Compression Standard*. London, U.K.: Chapman & Hall, 1997.

[18] A. R. Reibman and B. G. Haskell, "Constraints on variable bit-rate video for ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, pp. 361–372, Dec. 1992.

[19] C.-Y. Hsu, A. Ortega, and A. Reibman, "Joint selection of source and channel rate for VBR video transmission under ATM policing constraints," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 1016–1028, Aug. 1997.

[20] L. A. Wolsey, *Integer Programming*. New York: Wiley, 1998.

[21] D. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA: Athena, 1995.

[22] "Video codec test model for H.263 (TM5)," *Telenor Res.*, Jan. 1995.

[23] J. S. Lim, *Two-Dimensional Signal and Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1990.

[24] A. Vetro, H. Sun, and Y. Wang, "MPEG-4 rate control for multiple video objects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 186–199, Feb. 1999.

[25] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. Signal Processing*, vol. 36, pp. 1445–1453, Sept. 1988.



Eric C. Reed (S'93–M'01) received the B.S. degree in electrical engineering from Drexel University, Philadelphia, PA, in 1994, and the S.M. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, in 1996 and 2001, respectively.

He was a Member of the Advanced Telecommunications Research Program at MIT from 1994 to 2001. He spent two summers at Compaq's Cambridge Research Laboratory in the Video and Image Processing Group. He is currently a Senior Engineer in the Convergent Systems Division, Harmonic, Inc., White Plains, NY. He holds two patents in the areas of video compression and rate control.

Dr. Reed is the recipient of a DoD National Defense Science and Engineering Graduate Fellowship.



Jae S. Lim (S'76–M'78–SM'83–F'86) received the S.B., S.M., E.E., and Sc.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, in 1974, 1975, 1978, and 1978, respectively.

He joined the MIT faculty in 1978 as an Assistant Professor. He is currently a Professor in the Department of Electrical Engineering and Computer Science and Director of the Advanced Telecommunications Research Program. His research interests include digital signal processing and its applications to image, video and speech processing. He has contributed more than one hundred articles to journals and conference proceedings. He is the holder of more than 30 patents in the areas of Advanced Television Systems and Signal Compression. He is the editor of a reprint book, *Speech Enhancement* (Englewood Cliffs, NJ: Prentice-Hall, 1983), and a co-editor of a reference book *Advanced Topics in Signal Processing* (Englewood Cliffs, NJ: Prentice-Hall, 1987). He is also the author of a textbook *Two-Dimensional Signal and Image Processing* (Englewood Cliffs, NJ: Prentice-Hall, 1990). During the 1990s, he participated in the Federal Communication Commission's Advanced Television Standardization Process. He represented MIT in submitting an MIT/GI System, which became one of the four finalist systems. He also represented MIT when the four finalist systems became a single system through the formation of the Grand Alliance. The Grand Alliance HDTV System became the basis for the U.S. Digital Television Standard adopted by the FCC in December 1996.

Dr. Lim is the recipient of many awards including the Senior Award from the IEEE ASSP Society and the Harold E. Edgerton Faculty Achievement Award.