

Efficient Bandwidth Resource Allocation for Low-Delay Multiuser Video Streaming

Guan-Ming Su, *Student Member, IEEE*, and Min Wu, *Member, IEEE*

Abstract—This paper studies efficient bandwidth resource allocation for streaming multiple MPEG-4 fine granularity scalability (FGS) video programs to multiple users. We begin with a simple single-user scenario and propose a rate-control algorithm that has low delay and achieves an excellent tradeoff between the average visual distortion and the quality fluctuation. The proposed algorithm employs two weight factors for adjusting the tradeoff, and the optimal choice of these factors is derived. We then extend to the multiuser case and propose a dynamic resource allocation algorithm with low delay and low computational complexity. By exploring the variations in the scene complexity of video programs as well as dynamically and jointly distributing the available system resources among users, our proposed algorithm provides low fluctuation of quality for each user, and can support consistent or differentiated quality among all users to meet applications' needs. Experimental results show that compared to traditional look-ahead sliding-window approaches, our algorithm can achieve comparable visual quality and channel utilization at a much lower cost of delay, computation, and storage.

Index Terms—Dynamic resource allocation, fine granularity scalability (FGS) coding, multiuser video communications, rate control, visual quality fluctuation.

I. INTRODUCTION

THE CAPABILITY of real-time transmission of video over network paves way to a number of emerging applications, which allows us to communicate and entertain from almost every corner of the world. In such applications as digital video on-demand service, broad-band wireless video streaming and conferencing, and direct broadcast satellite (DBS) service, multiple encoded video programs will be transmitted or relayed through a central server. The overall bandwidth of the outbound video streams is limited by the server's outbound communication capacity. To efficiently share critical resources and meet a set of quality of service (QoS) requirements, a major concern for the server is how to allocate the bandwidth resource to each stream. There are two different strategies of resource allocation for multiple users, namely, a collection of single-user subsystems with static resource allocation among users, and a dynamic joint resource allocation system [1]–[3]. In the first system, each user is allocated a fixed amount of bandwidth and the video transmitted for each user is kept

below the bound. This bit allocation strategy treats each user individually without dynamically sharing resources with other users. Since different video programs have different content complexity, at a given bit rate some may have unnecessarily high perceptual quality, while others may have low perceptual quality. In contrast, a system with dynamic resource allocation can leverage the variation of the content complexity in different video programs, aggregate the resource from all users into a common pool, and jointly allocate bandwidth resource to each user to achieve consistent perceptual quality to each other [4]. In this paper, we focus on the dynamic bandwidth resource allocation for a multiuser video streaming system.

As with every resource allocation mechanism, we are concerned with the efficiency and effectiveness of the allocation. In particular, we are dealing with dynamically allocating bandwidth resource to a potentially large number of users in real-time video streaming applications. In order for a system to have good scalability to handle many users as well as to do real-time processing, the computational complexity of a resource allocator should be sufficiently low. The end-to-end delay is often stringent for accommodating interactive applications such as video conferencing. These motivate us to investigate efficient resource allocation strategies with low computational complexity, high scalability, and low delay.

Resource allocation strategies are tied to the system's service objective. In a video transmission system, the perceptual quality of the received video is one of the most important aspects of quality of service experienced by the end users. There are two types of visual quality concern. The most common concern is the average visual quality, often measured in terms of the average mean-square-error (MSE) of all video frames, or the corresponding peak signal-to-noise ratio (PSNR) [5]–[7]. The other important concern is the quality fluctuation, as substantial quality differences between nearby frames can bring annoying flickering and other artifacts to viewers even when the average PSNR is satisfactory. In many systems that employ a set of frames as an encoding unit (known as a group of pictures/frames), severe quality fluctuation may also appear at the boundaries between groups of frames [8]. The quality fluctuation can be measured by the mean absolute difference of the MSE between adjacent frames [6], [7], [9]. Most prior work targeted at optimizing one of the two measures. If the rate-distortion (R-D) characteristics of all video frames are identical, the bit rate allocated to each frame will be equal, leading to identical perceptual quality between frames and the above two measures can be simultaneously optimized [1]. In reality, however, a video has varying R-D characteristics, making it difficult to optimize the average quality and the quality fluctuation at the same time.

Manuscript received December 31, 2003; revised July 8, 2004. This work was supported in part by the U.S. National Science Foundation under Award CCR-0133704. Part of this work was presented at the IEEE International Conference on Communications, Paris, France, June 2004. This paper was recommended by Associate Editor H. Sun.

The authors are with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA (e-mail: gmsu@eng.umd.edu; minwu@eng.umd.edu).

Digital Object Identifier 10.1109/TCSVT.2005.852626

Our work aims at reaching an excellent tradeoff between these two quality criteria through a real-time low-delay algorithm.

In addition to perceptual quality criteria, resource allocation strategies are also closely related to a system's adjustability on resources. For video transmission, the adjustability concerns how a video system can change video encoding/transmission rate to achieve desired visual quality or vice versa. A highly scalable video codec is desirable since it provides flexibility and convenience in reaching the desired visual quality and/or the desired bit rate. Recently, the fine granularity scalability (FGS) coding [10]–[12] and fine granular scalability temporal (FGST) coding [13] have been added into the MPEG-4 video coding standard. The encoder generates a base layer at a low bit rate using a large quantization step and computes the residues between the original frame and the base layer. The bit planes of discrete cosine transform (DCT) transformed coefficients of these residues are then encoded sequentially as an enhancement layer, which is the FGS layer. The decoder can decode any truncated segment of the bit stream corresponding to each frame. The more bits the decoder receives and decodes, the higher the perceptual quality of video we observe. We adopt FGS codec in this work to allow convenient adjustment of rate and distortion.

While a number of works have been devoted to bandwidth allocation for video, only a small amount of them address the multiuser problem. As we will review in Section II, the existing approaches do not always provide a good tradeoff on the two types of visual quality concerns (low average distortion and low fluctuation), or do not always keep delay low and scale well to accommodate a large number of users. In this paper, we propose an efficient algorithm for dynamic bandwidth resource allocation that addresses the above issues and provides superior performance over the prior art. Our work starts with a simple scenario where there is only a single user in the system. We employ two weight factors for adjusting the tradeoff between the overall distortion and the quality fluctuation, and derive the optimal choice of these factors for achieving an excellent tradeoff. We then extend the strategy to the multiuser case and propose a multiuser real-time resource allocation algorithm with low delay and low computational complexity. By exploring the variation in the scene complexity of each video program and jointly redistributing the system resources among users, our proposed algorithm provides low fluctuation of quality for each user, and depending on the applications' needs, it can provide consistent or differentiated quality among all users. Our experimental results on 15 sequences of a total of nearly 6000 frames show that compared to conventional look-ahead sliding-window approaches, the proposed algorithm can achieve comparable perceptual quality and channel utilization at a much lower cost of delay, computation, and storage, and therefore is suitable for a variety of multiuser broad-band applications.

The paper is organized as follows. We first review the prior work in Section II and provide preliminaries on the R-D model for FGS video in Section III. Section IV discusses the simple case of single-user bandwidth resource allocation and proposes an efficient real-time algorithm. We then extend the strategy to the multiuser scenario and present a new algorithm in Section V. Experimental results are shown in Section VI and conclusions drawn in Section VII.

II. PRIOR WORK

Rate control for single user can be considered as a special case of multiuser resource allocation. In general, a video encoded in variable bit rate (VBR) bitstream gives better perceptual quality than in constant bit rate (CBR) bitstream due to the variation of the scene complexity [14]. Most applications deliver a VBR video bitstream through a channel with a fixed amount of bandwidth (known as a CBR channel) because of its predictable traffic pattern as well as simple network management. However, VBR transmission has been shown to provide better source quality and network utilization [15]. To smoothen the traffic and alleviate the jitter caused by VBR coding and transmission, the system allocates buffers on both the transmitter side and the receiver side. The dynamics of the buffer is subject to two constraints to maintain the QoS. When the buffer overflows, we will start to lose data, which degrades the received visual quality; and when the decoder buffer underflows, the decoder has no data to keep up the decoding, which causes jitters. Therefore, a rate control algorithm must be applied to prevent the buffers from overflowing and underflowing [16].

For systems employing MPEG-1/2, H261, or H.263, the encoding rate is often changed by adjusting the quantization step size [16], [17]. To achieve high overall perceptual quality in the single-user scenario, rate control was formulated as an optimization problem in [5], [18]–[20]. These approaches are suitable for off-line applications where the entire video content is known to the transmitter. The computation cost for handling a long video sequence is high due to the nature of integer and dynamic programming. To facilitate solving the rate control problems, several R-D models of existing video codecs have been exploited in the literature. An R-D based approach was proposed in [21] under the assumption that the DCT coefficients of a motion-compensated residue frame are uncorrelated and Laplacian distributed. A R-D model using intra-frame approximation and inter-frame dependency within one GOP was proposed in [6] to meet the perceptual requirement. A quadratic R-D model and rate control for MPEG-4 was studied in [22] and [23]. A rate control algorithm employing a linear correlation model was proposed in [24], whereby the correlation between the rate and the percentage of zeros among the quantized transform coefficients was explored. To simplify the selection of encoding and channel rates, wavelet-based embedded codecs were considered in the rate control problems of [7] and [25].

Sliding window is a general approach that can be used to keep track and allocate system resources. The work in [26] took advantage of the fine granularity of the MPEG-4 FGS codec and proposed a variable-size sliding window scheme to control how much FGS layer data is sent under different channel conditions. An R-D based rate control scheme for prestored video was studied in [9] using a three-level bit allocation for the base layer and employing a sliding window for the FGS layer rate control. An online algorithm using a look-ahead sliding window to achieve constant perceptual quality was proposed in [1]. To apply this scheme for transmitting real-time encoded video, we need to allocate extra storage to store several frames ahead, and perform bit allocation for the current frame by solving such an optimization problem that all frames within a

look-ahead window have consistent and the highest possible perceptual quality subject to a given rate budget. Our studies show that to obtain a low fluctuation of quality, the window size should be no smaller than the size of half to one GOP, which leads to a nontrivial amount of delay that is often too long for real-time interactive applications. We will investigate in this paper how to overcome the problems of long delay and extra storage associated with the sliding window approach.

Several works on joint rate control for multiple video programs employed MPEG-1/2 codecs [2]–[4]. And the extension of the sliding window approach to multiple MPEG-4 FGS video programs was proposed in [1], employing a two-dimensional (2-D) window to address the multiuser problem. However, the computational complexity and extra storage for the look-ahead frames of the sliding window approach go up with the increase of the window size and the number of users. As the number of users increases in the system, the required computational resources to achieve a low fluctuation of quality become formidable. Our work in the current paper will overcome the problem of high computational complexity of the 2-D sliding window approach and improve the system scalability to accommodate many users.

III. FGS RATE-DISTORTION MODEL AND SIMILARITY

Existing rate control schemes for a single-layer video stream often employ an intra-frame R-D model. Laplacian and Gaussian distribution are typical approximations of DCT coefficients, leading to the frequent use of an exponential or a polynomial R-D model [1], [21]. In contrast to single-layer codecs, FGS codec is a two-layer embedded scheme with an enhancement layer encoded bit plane by bit plane. There is a need to model the statistical distribution of DCT bit planes and their R-D characteristics. Furthermore, due to the nature of the temporal redundancy in video, the predictively encoded frames within one scene have highly similar R-D characteristics. In this section, we present R-D models for intra-frame and inter-frame of a FGS layer, which will be used in our work.

A. Intra-frame Rate-Distortion Model

As reviewed earlier, the MPEG-4 FGS standard employs bit-plane coding of the DCT residue between the original frame and the base layer. For a given bit plane in a frame, if the video is spatially stationary so that the length of the entropy encoded FGS symbols in all blocks is similar to each other, the decoded bit rate and the corresponding amount of reduced distortion will have an approximately linear relationship over the bit rate range of this bit plane. Previous studies in [1] and [9] and our experiments show that a piecewise linear line is a good approximation to the R-D curve of FGS video in the frame level. This piecewise linear line model can be described as

$$D_j(r_j) = \left(\frac{\text{MSE}_j^{k+1} - \text{MSE}_j^k}{\text{Rate}_j^{k+1} - \text{Rate}_j^k} \right) (r_j - \text{Rate}_j^k) + \text{MSE}_j^k,$$

for $\text{Rate}_j^k \leq r_j \leq \text{Rate}_j^{k+1}$ and $k = 0, \dots, p-1$. (1)

Here, $D_j(r_j)$ represents the MSE between the j th original frame and the decoded frame with rate r_j , MSE_j^k the distortion of

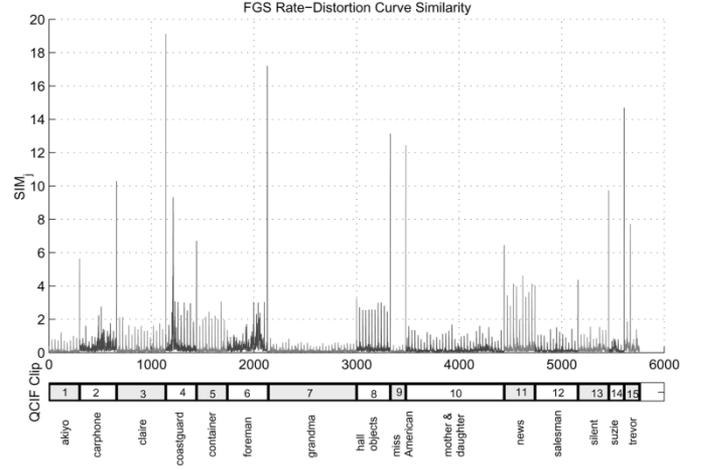


Fig. 1. Inter-frame similarity of FGS R-D characteristics. The results for the odd and even scenes are presented in alternating colors.

the j th frame measured in mean square error after completely decoding the first k DCT bit planes, Rate_j^k the corresponding bit rate, and p the total number of bit planes. We use MSE_j^0 and Rate_j^0 to represent the distortion and rate of the base layer, respectively. Since DCT is a unitary transform, measuring the mean square error between an original frame and its partially decoded version from FGS encoded stream is equivalent to calculating the average energy of the undecoded DCT bit planes in the FGS data stream, along with the residue between the original frame and the complete FGS data. Thus, all MSE_j^k 's and Rate_j^k 's can be obtained during the encoding process.

B. Similarity in Interframe R-D Characteristics

Another important characteristic of FGS video is that the R-D curves of FGS layer between two consecutive frames are similar when they are within the same scene. The rationale is as follows: for a video segment within a scene, the energy of the motion compensation residues between two adjacent frames are comparable. As the base layer is generated using a set of large quantization steps, it leaves most motion residues to be coded by the FGS layer. Therefore, after FGS encoding, the overall R-D characteristics between two adjacent frames are similar.

We quantify the similarity of the R-D characteristics between frame j and $j+1$ using

$$\text{SIM}_j = \frac{1}{T} \sum_{t=1}^T |D_j(t \cdot c) - D_{j+1}(t \cdot c)| \quad (2)$$

where c is a bit rate sampling interval, $T = \lfloor ((\min \{r_j^{f_{\max}}, r_{j+1}^{f_{\max}}\}) / c) \rfloor$, and $r_j^{f_{\max}}$ is the maximal available amount of FGS data for the j th frame. A low value of SIM_j implies high similarity in the R-D characteristics of the j th and $(j+1)$ th frames. Fig. 1 shows the SIM_j for a long video sequence consisting of 15 different standard QCIF clips. As we can see, the R-D models within each clip show a strong similarity. The SIM value becomes large and suggests low similarity when transiting from one clip to another.

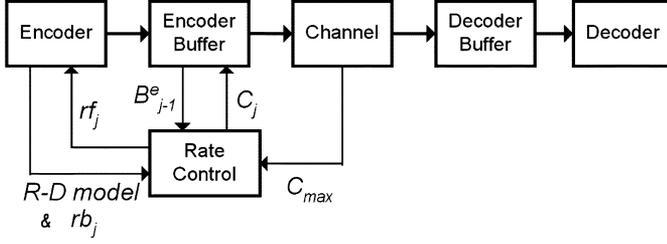


Fig. 2. Block diagram of a single-user video streaming system.

IV. LOW-DELAY BANDWIDTH RESOURCE ALLOCATION FOR SINGLE USER

To facilitate the investigation of the resource allocation problem in a multiuser system, we first study in this section a special case that concerns only a single user in the system. We begin with a discussion on the mechanism of a single-user FGS streaming video system and the corresponding constraints. We formulate this system as a resource allocation problem with two perceptual objectives subject to the system constraints. An online bandwidth resource allocation algorithm with low delay and low fluctuation of quality is then proposed to achieve a tradeoff point between these two perceptual criteria.

A. System Constraints

Illustrated in Fig. 2 is a typical streaming video system. There are two subcomponents in the encoder. One is the base layer encoder and the other is the FGS layer encoder. We discretize the time line by dividing one second into F time slots, where F is the video frame rate. For the simplicity in system design and providing a primitive quality with low fluctuation, we set a large fixed quantization step for all frames in the base layer codec and only perform the rate control for the FGS layer. We denote the base layer rate as rb_j , i.e., a total of rb_j bits must be sent at the j th time slot to ensure the baseline quality. The FGS encoder encodes the bit planes of the residue. Both encoders analyze the R-D characteristics of the incoming video frame and pass the necessary information, such as the R-D pairs $(Rate_j^k, MSE_j^k)$, to the rate control module. After the rate control module determines the amount of FGS data to be transmitted, the encoded base layer and the truncated FGS layer bitstream are moved to the encoder buffer, where we denote the FGS data rate at the j th time slot as rf_j . The channel then delivers video bitstream from the encoder buffer to the decoder buffer. Here we assume that the channel has a maximum rate for reliable transmission, C_{max} , although it is not necessarily in its full load all the time. The amount of channel transmission rate at the j th time slot, denoted as C_j , is also determined by the rate control module. For simplicity, we assume that the transmission delay of every packet is fixed at d^c time slots [5], [19]: if a packet is sent from the encoder buffer at the j th time slot, it will arrive at the decoder buffer at the $(j + d^c)$ th time slot. The decoder fetches data from the decoder buffer, decodes it, and displays each decompressed video frame at its desired instant. Therefore, the major task of the rate control module is to determine rf_j and C_j . To ease the discussion, we summarize the notations in Table I.

 TABLE I
SUMMARY OF NOTATIONS

B_{max}^e	maximal size of encoder buffer
B_j^e	encoder buffer occupancy at the j^{th} time slot
B_{max}^d	maximal size of decoder buffer
B_j^d	decoder buffer occupancy at the j^{th} time slot
C_{max}	maximal channel capacity
C_j	channel transmission rate at the j^{th} time slot
rb_j	base layer rate of the j^{th} frame
rf_j^{max}	maximal available FGS layer rate of the j^{th} frame
rf_j	transmitted FGS layer rate of the j^{th} frame
r_j	total effective encoding rate of the j^{th} frame, $r_j = rb_j + rf_j$
d^c	transmission delay (in unit of frames)
d^d	number of pre-stored frames in the decoder buffer at the beginning of the service
$D_j(\cdot)$	rate-distortion function of the j^{th} frame
R_j	upper bound of FGS rate budget at the j^{th} time slot
R_j^f	fractional FGS rate budget at the j^{th} time slot
R_j^p	required rate of j^{th} frame to keep the same distortion as the previous frame
β	FGS budget factor for determining R_j^f
w_p	weight factor for combining R_j^f and R_j^p

There are three constraints imposed in this system, as studied in the literature [15], [16]. The first constraint is to prevent the encoder buffer of a limited size from overflow. At the j th time slot, a data segment of size C_j is taken from an encoder buffer and sent through the channel, and then a newly encoded frame with size $r_j = rb_j + rf_j$ is added to the encoder. The dynamics of the encoder buffer can thus be expressed as

$$B_j^e = \max \{B_{j-1}^e + rb_j + rf_j - C_j, 0\} \leq B_{max}^e \quad (3)$$

where B_j^e is nonnegative and describes the occupancy of encoder buffer, and B_{max}^e the maximal size of encoder buffer. In addition, the FGS rate rf_j should be nonnegative. For a given C_j , we can rearrange inequality (3) as a constraint for rf_j

$$0 \leq rf_j \leq B_{max}^e + C_j - B_{j-1}^e - rb_j. \quad (4)$$

The second constraint is on the channel transmission rate, C_j . It is nonnegative and cannot exceed the maximal channel capacity, C_{max} . That is

$$0 \leq C_j \leq C_{max}. \quad (5)$$

The third constraint is on the occupancy of the decoder buffer, which should neither overflow nor underflow. We assume that the decoder fetches all the data that belongs to the next frame from the decoder buffer and decodes it within one time slot. In addition, we assume playback buffering of d^d frames, i.e., the first d^d frames are received and stored in the decoder buffer before the playback is started. The total end-to-end delay from the encoder buffer through the channel and decoder buffer to the decoder is thus $d^c + d^d$ frames delay. The decision on how much data is sent into the channel at the j th time slot will directly affect the decoder buffer occupancy at the $(j + d^c)$ th time slot. To meet the constraint imposed on the decoder buffer occupancy at the $(j + d^c)$ th time slot, there is a corresponding limit on how much data can be sent through the channel at the j th time slot. Denote the decoder buffer occupancy as B_j^d , and the maximal

size of decoder buffer as B_{\max}^d . We summarize the constraint on the decoder buffer at the $(j + d^c)$ th time slot as

$$B_{j+d^c}^d = B_{j+d^c-1}^d + C_j - r_{j-d^d} \in [0, B_{\max}^d]. \quad (6)$$

Combining (5) and (6), we arrive at the following constraint for the channel transmission rate C_j

$$\begin{aligned} \max \{r_{j-d^d} - B_{j+d^c-1}^d, 0\} \\ \leq C_j \leq \min \{B_{\max}^d - B_{j+d^c-1}^d + r_{j-d^d}, C_{\max}\}. \end{aligned} \quad (7)$$

In summary, inequalities (4) and (7) are the fundamental constraints for a single-user FGS video streaming system.

B. Criteria for Visual Quality

We adopt two visual quality criteria for video sequences to measure the average distortion and the quality fluctuation. More specifically, the average received quality is measured by the average mean square error (*aveMSE*) of all M frames in a video sequence

$$\text{aveMSE} = \frac{1}{M} \sum_{j=1}^M D_j(r_j) \quad (8)$$

where $D_j(r_j)$ represents the MSE between the j th original frame and the decoded frame with rate r_j . To account for the fluctuation of quality between consecutive frames, a large value of which can be objectionable to viewers, we use the mean absolute difference of consecutive frames' mean square error (*madMSE*) to measure the perceptual fluctuation

$$\text{madMSE} = \frac{1}{M-1} \sum_{j=2}^M |D_j(r_j) - D_{j-1}(r_{j-1})|. \quad (9)$$

The higher the *madMSE* is, the larger the perceptual fluctuation is. We also define the corresponding PSNR version of these two criteria and denote as *avePSNR* and *madPSNR*, respectively.

C. Problem Formulation

Our objective is to design a rate control strategy to achieve both low *aveMSE* (high *avePSNR*) and low *madMSE* (low *madPSNR*) subject to the constraints of (4) and (7). For offline applications where the entire video content is readily available before the transmission, all R-D information is known and we can formulate this system as (10), shown at the bottom of the page. In this formulation, $f(\cdot, \cdot)$ is a function reflecting the importance and relevance of the average distortion and the quality fluctuation in the human perceptual system. For ex-

ample, a linear combination function of *aveMSE* and *madMSE* is a simple choice of $f(\cdot, \cdot)$.

An optimal solution can be found for the above offline problem using standard nonlinear programming with penalty functions. The complexity for searching for optimal solution would, however, be formidable except for short video clips. In addition, the offline solution is not applicable to online applications where the video content is not entirely available beforehand. If the variations of the R-D characteristics of video sources can be well captured by a finite-state Markovian chain, we can model this system using stochastic three-machine flowshop with finite buffers [27] and obtain an optimal rate control policy using dynamic programming techniques. However, it has been shown that a compressed video sequence trace has long-range dependence [28], which is different from the short-range dependence such as a Markovian process and cannot be handled well using existing solutions. Thus, in this paper, we focus on a sequential resource allocation solution that has a moderate amount of computational complexity and can accommodate online video applications.

The strategy of choosing the effective encoding rate for the FGS layer $\{r_{f_j}\}$ and the channel transmission rate $\{C_j\}$ closely depends on the relative weights of the average distortion and the perceptual fluctuation in the objective function. To achieve low *aveMSE* alone, one may employ a greedy strategy to make the encoder buffer as full as possible all the time and make full use of the available channel bandwidth. This may lead to the desire to select C_j at the upper bound in (7), namely

$$C_j = \min \{B_{\max}^d - B_{j+d^c-1}^d + r_{j-d^d}, C_{\max}\} \quad (11)$$

and to set the FGS rate at the upper bound in (4), which is denoted as R_j and defined as

$$R_j \triangleq B_{\max}^e + C_j - B_{j-1}^e - rb_j. \quad (12)$$

When the encoder buffer is always full, the amount of incoming data cannot exceed the maximal amount of data allowed to be sent through the channel at each time slot. This is equivalent to assigning the same bandwidth resource for transmitting each frame. When encountering intra-coded frames (or I-frames), which have a larger amount of data at the base layer than predictively coded frames, we will have very limited budget left for sending their associated FGS enhancement layers. The MSE of I-frames will thus be larger than the MSE of the other types of frames. This leads to a potential increase in *madMSE*.

On the other hand, low *madMSE* may be achieved by assigning each frame a rate that corresponds to the same distortion, MSE_s . To do so, we extract the R-D pairs from the FGS

$$\begin{aligned} \min_{\{r_{f_j}, C_j\}} & f(\text{aveMSE}, \text{madMSE}) \\ \text{subject to} & \begin{cases} 0 \leq r_{f_j} \leq B_{\max}^e - B_{j-1}^e - rb_j + C_j, & \forall j \\ \max \{r_{j-d^d} - B_{j+d^c-1}^d, 0\} \leq C_j \leq \min \{B_{\max}^d - B_{j+d^c-1}^d + r_{j-d^d}, C_{\max}\}, & \forall j \\ B_j^e = \max \{B_{j-1}^e + rb_j + r_{f_j} - C_j, 0\} \leq B_{\max}^e, & \forall j \\ 0 \leq B_{j+d^c}^d = B_{j+d^c-1}^d + C_j - r_{j-d^d} \leq B_{\max}^d, & \forall j \end{cases} \end{aligned} \quad (10)$$

encoder, approximate the R-D curve for each frame, and assign the rate for the FGS enhancement layer as $rf_j = D_j^{-1}(\text{MSE}_s)$. To prevent encoder buffer from overflowing when encountering I-frames or a new complex scene, we would have to allocate a small amount of data rate for the FGS layers of these I-frames. To keep the lowest madMSE , other frames will also have a small amount of FGS-layer data. As a result, this second approach would not give a low aveMSE . Next, we present a new resource allocation algorithm that can achieve an improved tradeoff between the average distortion and the quality fluctuation.

D. Proposed Resource Allocation Algorithm

We introduce two weight factors in our proposed resource allocation algorithm to solve the above-mentioned problems. To overcome the quality fluctuation problem in the lowest- aveMSE scheme, we propose to use a fraction of the maximally allowed FGS data rate (determined by the buffer constraints) as the effective FGS encoding rate, i.e., $R_j^f \triangleq \beta R_j$, where $\beta \in [0, 1]$ is a budget factor. Compared to adopting the full budget R_j , the fractional budget can keep the encoder buffer occupancy low to accommodate future I-frames and other complex frames. As such, the rate budget available to the incoming I-frames will be close to the maximal encoder buffer size plus the full channel bandwidth, allowing for more FGS data of the I-frames to be sent to avoid a high increase in the madMSE .

To overcome the problem of low overall perceptual quality as in the lowest- madMSE scheme, we relax the requirement of zero madMSE fluctuation by taking partial consideration of both the rate that maintains zero madMSE and the current occupancy of the encoder buffer. We quantify this strategy using a weight factor $w_p \in [0, 1]$ and allocate the FGS rate for the j th frame as

$$\begin{aligned} rf_j &= \min \left\{ w_p R_j^p + (1 - w_p) R_j^f, R_j \right\} \\ &= \min \left\{ w_p R_j^p + (1 - w_p) \beta R_j, R_j \right\} \end{aligned} \quad (13)$$

where R_j^p is the amount of FGS data needed to achieve the same perceptual quality as the previous frame and can be determined by

$$R_j^p \triangleq D_j^{-1}(D_{j-1}(r_{j-1})) - rb_j. \quad (14)$$

As we can see, the allocated FGS rate rf_j is determined using the two factors β and w_p . The lowest aveMSE scheme and madMSE scheme are two special cases of this new strategy: when $w_p = 0$ and $\beta = 1$, (13) becomes the lowest- aveMSE scheme; and when $w_p = 1$, (13) becomes the lowest- madMSE scheme.

We now examine how to select appropriate β and w_p to achieve a good tradeoff between low aveMSE and low madMSE .

1) *Selection of β* : We first fix w_p and study the impact of β on avePSNR and madPSNR when consecutive video frames have similar R-D characteristics. In this situation and with a fixed w_p , when β becomes larger, both the madPSNR and the avePSNR will increase. However, after β passes a specific value, β_0 , the improvement of avePSNR is dramatically reduced while the quality fluctuation becomes more significant. This phenomenon is demonstrated in Fig. 3(a), where we use the first 200

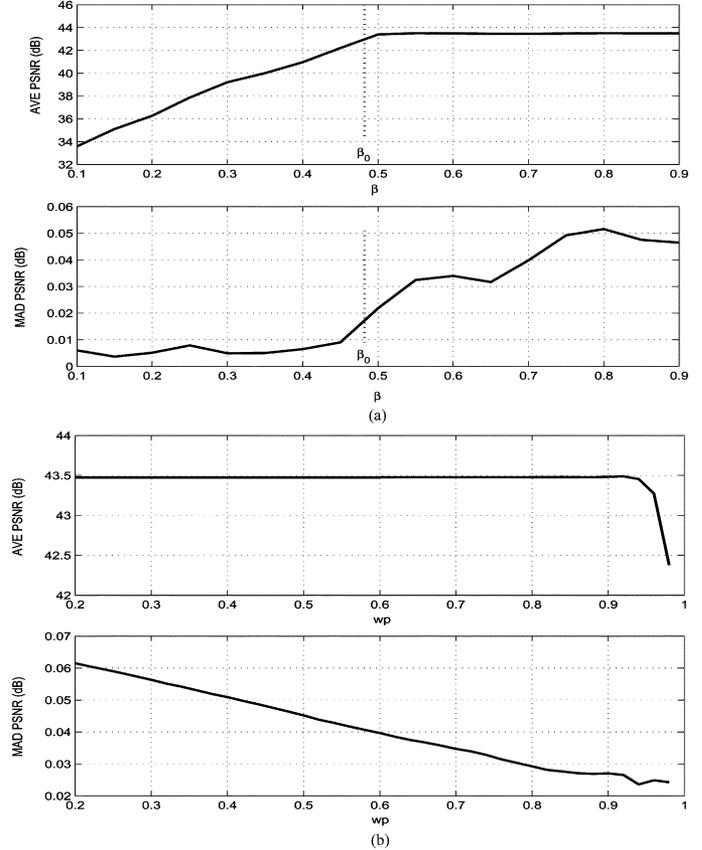


Fig. 3. Impact of β and w_p on visual quality for the first 400 frames of the *grandmother* sequence. (a) avePSNR and madPSNR for different β . (b) avePSNR and madPSNR for different w_p .

frames from the QCIF video clip of the *grandmother* as an example and set the w_p factor at 0.95. Given such trends of avePSNR and madPSNR for different β , the β_0 value, indicated by a vertical line in Fig. 3(a), provides a good tradeoff between avePSNR and madPSNR . As shown in Appendix I, β_0 can be expressed as

$$\beta_0 = \frac{C_{\max} - \overline{rb}}{B_{\max}^e + C_{\max} - \overline{rb}} \quad (15)$$

where \overline{rb} represents the average rate of the base layer, which can be approximated using a moving average of the bit rate statistics of the past L frames. We can see that β_0 is an equilibratory operating point to keep the encoder buffer near empty and the channel utilization near full.

We should notice that in reality, the consecutive video frames do not have exactly the same R-D characteristics. So if β is set to be exactly β_0 , the system is on the verge between stable and unstable operation: the encoder buffer is nearly empty, and as the video content fluctuates, the buffer may underflow. Thus, to ensure a high utilization of channel bandwidth and high avePSNR , we should select a β that is slightly above β_0 such that $\beta = \beta_0 + \Delta\beta$, where $\Delta\beta$ is a small positive constant.

2) *Selection of w_p* : In general, a system with a high value of w_p has low fluctuation of visual quality. When consecutive frames within a video segment have similar R-D characteristics, increasing w_p affects only the madPSNR while the avePSNR has

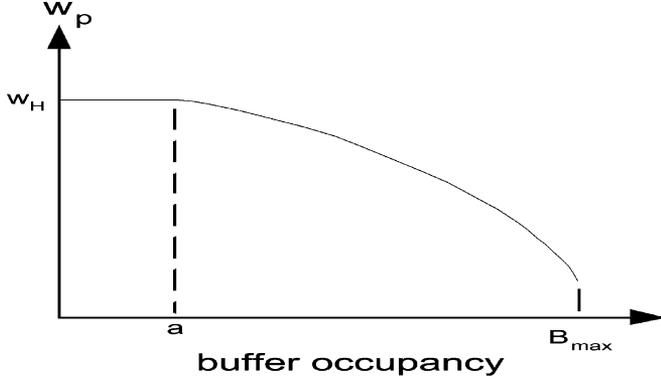


Fig. 4. Selection of w_p according to the encoder buffer occupancy.

little decrease until w_p is close to one. To achieve low fluctuation of quality, high value of w_p is preferred. This trend is illustrated in Fig. 3(b), where we again use the above-mentioned *grandmother* video clip as an example and set β to a fixed value of 0.5.

When two adjacent frames exhibit significant difference in R-D characteristics such as when arriving at scene boundary, we need to make adjustment to this system to handle the following frames. We consider two cases here. The first case is that the video sequence enters a new segment with more complex R-D characteristics than that of the previous segment, whereby the FGS rate required to maintain the same PSNR level as before is higher than the FGS rate for the previous sequence. To balance between the need of preventing the encoder buffer from overflowing and controlling the fluctuation of perceptual quality, we dynamically adjust the weight factor w_p with respect to the encoder buffer occupancy. When the encoder buffer occupancy is lower than a threshold a , we set w_p at a high value w_H to keep the distortion similar to that of the previous frame. When the encoder buffer occupancy is higher than threshold a , we try to drain out the data from the buffer quickly by choosing w_p as a concave and decreasing function of the buffer occupancy as shown in Fig. 4, so that the higher the buffer occupancy is, the lower w_p is. As an example, the overall selection of w_p can be chosen as

$$\begin{aligned}
 w_p &= W(B_{j-1}^e) \\
 &= w_L + (w_H - w_L) \\
 &\quad \times \left(1 - u(B_{j-1}^e - a) \frac{B_{j-1}^e - a}{B_{\max}^e} \right)^b \quad (16)
 \end{aligned}$$

where $u(\cdot)$ is the step function, and a and b are positive constants.

The second case is that the video sequence enters a new segment with simpler R-D characteristics than that of the past segment, whereby the FGS rate required to maintain the same PSNR level as before is lower than the FGS rate for the previous sequence. To balance between fully utilizing the available channel bandwidth resource and maintaining low fluctuation of quality, after detecting a change in R-D characteristics, we immediately adjust w_p to a low value w_L to utilize more available bandwidth and maintain this value for the following

TABLE II
PROPOSED SINGLE-USER RESOURCE ALLOCATION ALGORITHM (*S-LDLF*)

1. Initialization:
 $j = RDSC = 1$, $B_0^e = B_1^d = \dots = B_{dc}^d = 0$, $R_0^p = rb_1$,
 $w_p = w_L$, $D_0(rb_0) = D_1(rb_1)$.

2. While the last frame of this video is not reached:

a) Calculate the budget factor and weight factor
 $\bar{r}b = \frac{1}{j - \max(j-L, 0)} \sum_{k=\max(j-L, 0)+1}^j rb_k$,
 $\beta = \frac{C_{max} - \bar{r}b}{B_{max}^e + C_{max} - \bar{r}b} + \Delta\beta$, $R_j^p = D_j^{-1}(D_{j-1}(r_{j-1})) - rb_j$.
If $|R_j^p - R_{j-1}^p| / R_{j-1}^p \geq S_T$, then $RDSC = j$.
If $j \in [RDSC, RDSC + P_T]$, then $w_p = w_L$,
else $w_p = W(B_{j-1}^e)$.

b) Select the channel transmission rate
 $C_j = \min\{B_{max}^d - B_{j+dc-1}^d + r_{j-d}, C_{max}\}$.

c) Select FGS rate
 $R_j = B_{max}^e + C_j - B_{j-1}^e - rb_j$,
 $rf_j = \min\{w_p R_j^p + (1 - w_p)\beta R_j, R_j\}$.

d) Update the encoder buffer occupancy information:
 $B_j^e = \max\{B_{j-1}^e + rb_j + rf_j - C_j, 0\}$.
If $B_j^e = 0$, then $C_j = B_{j-1}^e + rb_j + rf_j$.

e) Update the decoder buffer occupancy information:
 $B_{j+dc}^d = B_{j+dc-1}^d + C_j - r_{j-d}$.

f) $j = j + 1$.

P_T frames. As the scene transition is complete and the channel bandwidth becomes highly utilized again, we can adjust w_p back to a high value to maintain constant quality.

In summary, we adjust w_p dynamically according to the encoder buffer occupancy and the detection of significant change in R-D characteristics. The changes in R-D characteristics can be identified by calculating the relative rate change between R_j^p and R_{j-1}^p , i.e., we check whether $|R_j^p - R_{j-1}^p| / R_{j-1}^p$ is greater than a threshold S_T . The β parameter will be chosen to be right above β_0 as in (15) and the channel transmission rate according to (11). We present the detailed algorithm in Table II.

V. LOW-DELAY BANDWIDTH RESOURCE ALLOCATION FOR MULTIPLE USERS

In this section, we extend the proposed bandwidth resource allocation algorithm from handling single user to multiple users. A simple way to deal with multiple users/sequences is to allocate a fixed amount of resource, including various buffers and channel bandwidth, to each user, and apply our proposed single-user approach to each individual user. We shall call this strategy *multiple single-user approach*. A more sophisticated approach allows for dynamically allocating resource among users and has the potential to improve the utilization of critical resources. Multiple users share the total channel bandwidth and buffer capacity, and a central resource allocation system dynamically distribute these system resources to handle the transmission of the video sequences from all users. We shall call this class of strategies *dynamic multiuser approaches*. We will focus on the dynamic multiuser approach and aim at achieving high average visual quality and low fluctuation of quality for each user. We will examine the scenarios of uniform quality of service among all users versus differentiated service. The performance of the dynamic multiuser strategy will be compared with the multiple single-user strategy through simulations in Section VI.

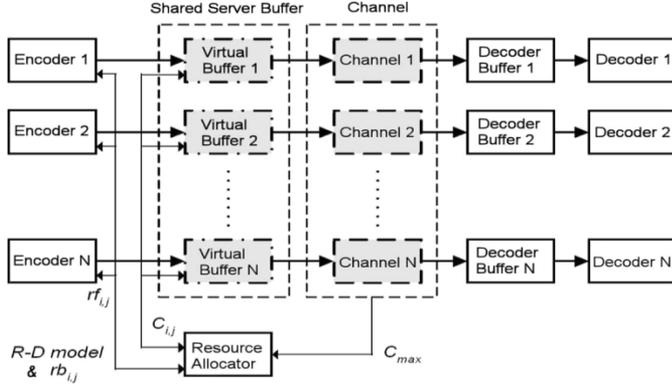


Fig. 5. Block diagram of a multiuser video streaming system.

A. System Constraints

An N -user system is depicted in Fig. 5. At the server side, each user has his/her own video encoder to encode a different video program in real time. For the i th user, the corresponding encoder sends the measured parameters of the R-D model of the current j th frame to the resource allocation module. The parameters are in the form of $(\text{Rate}_{i,j}^k, \text{MSE}_{i,j}^k)$ for the first k th bitplane. Using the R-D model, the resource allocation module determines the amount of FGS data to be transmitted. The encoder of each user then moves both the base layer data at the rate of $rb_{i,j}$ and the FGS layer bitstream truncated at the allocated rate $rf_{i,j}$ to the shared server buffer whose maximal capacity is B_{\max}^e . Denote the occupancy of the shared server buffer at the time slot j as B_j^e and the amount of data left by the i th user in the server buffer as $B_{i,j}^e$. We can treat $B_{i,j}^e$ as a virtual encoder buffer for the i th user, and the sum of all virtual encoder buffers' occupancy equals to B_j^e . All users also share a channel whose maximal outbound capacity is C_{\max} . The resource allocation module needs to determine the channel transmission rate allocated for each user's data at the time slot j , which we denote as $C_{i,j}$ for the i th user. Upon receiving the data packets of the video program intended for him/her, each end-user first stores them temporarily in the decoder buffer, then decodes and renders each frame on time. In summary, similar to the single user case, the duty of the multiuser resource allocation module is to determine $rf_{i,j}$ and $C_{i,j}$ jointly for all users.

In parallel to the single-user case, there are three sets of system constraints for multiuser resource allocation. The first set of constraints is on the server buffer, which should not overflow. In particular, the sum of all virtual encoder buffers should not exceed the capacity of the server buffer. The dynamics of the buffer occupancy can be extended from the single-user case. The constraints can be described as

$$\begin{aligned} B_{i,j}^e &= \max \{ B_{i,j-1}^e + rb_{i,j} + rf_{i,j} - C_{i,j}, 0 \} \leq B_{\max}^e, \quad \forall i \\ B_j^e &= \sum_{i=1}^N B_{i,j}^e \leq B_{\max}^e. \end{aligned} \quad (17)$$

The constraints of the FGS layer rates for each user can be extended from the single-user problem and described as

$$0 \leq rf_{i,j} \leq B_{\max}^e - B_{i,j-1}^e - rb_{i,j} + C_{i,j}, \quad \forall i \quad (18)$$

and

$$0 \leq \sum_{i=1}^N rf_{i,j} \leq B_{\max}^e - B_{j-1}^e - \sum_{i=1}^N rb_{i,j} + \sum_{i=1}^N C_{i,j}. \quad (19)$$

Since all users share the overall bandwidth, both the individual and the aggregate channel transmission rate should be nonnegative and not exceed the maximal capacity. These channel transmission rate constraints can be described as

$$0 \leq C_{i,j} \leq C_{\max}, \quad \forall i \quad (20)$$

$$0 \leq \sum_{i=1}^N C_{i,j} \leq C_{\max}. \quad (21)$$

The constraints of the decoder buffer are the same as the single-user case

$$B_{i,j+d_i^c}^d = B_{i,j+d_i^c-1}^d + C_{i,j} - r_{i,j-d_i^d} \in [0, B_{i,\max}^d] \quad (22)$$

where $B_{i,\max}^d$ is the maximal size of the i th decoder buffer, d_i^c the channel transmission delay for user i , and d_i^d the prestored frame delay in the decoder buffer for user i . Rearranging and combining (20) and (22), we obtain a simplified constraint for the individual channel transmission rate

$$C_{i,j}^L \leq C_{i,j} \leq C_{i,j}^U, \quad \forall i \quad (23)$$

where

$$\begin{aligned} C_{i,j}^U &\triangleq \min \{ B_{i,\max}^d - B_{i,j+d_i^c-1}^d + r_{i,j-d_i^d}, C_{\max} \}, \\ C_{i,j}^L &\triangleq \max \{ r_{i,j-d_i^d} - B_{i,j+d_i^c-1}^d, 0 \}. \end{aligned}$$

Inequalities (18), (19), (21), and (23) are fundamental constraints in a multiuser system. Under these constraints, we determine the rate of the FGS data and the channel transmission rate for each user in the system to achieve low fluctuation of perceptual quality of each program as well as the desired uniform or differentiated perceptual quality among all programs.

B. Proposed Resource Allocation Algorithm

Our proposed multiuser resource allocation algorithm first allocates the channel transmission rate for each user subject to (21) and (23). With a selected channel transmission rate, we extend rate control strategy that we have proposed for the single-user case to the multiuser case to determine the feasible range for FGS layer data of each user according to (18) and (19). Specifically, we use two weight factors β and w_p to achieve a tradeoff between average perceptual quality and quality fluctuations.

1) *Selection of Channel Transmission Rate:* As all users share the overall channel bandwidth in multiuser system, we need to dynamically adjust the transmission rate allocated for each user. Our strategy consists of two steps: first, we assign each user a lower bound of channel transmission rate $C_{i,j}^L$ to prevent all decoder buffers from underflowing. Second, to help drain out the virtual encoder buffers, we distribute the rest of the available bandwidth $C_{\max} - \sum_{i=1}^N C_{i,j}^L$ to each user proportional to his/her previous encoding rate $r_{i,j-1}$. Thus, when a video program encounters an I-frame and leaves a large amount of data in its virtual encoder buffer at the previous

time slot, our strategy will assign the corresponding user a high channel transmission rate to drain his/her virtual encoder buffer at the current time slot.

2) *Selection of FGS Rate*: As in the single-user strategy proposed in Section IV-D, to balance between low fluctuation of quality and high average quality, we introduce two weight factors to our multiuser algorithm, namely, β and w_p .

We first take an aggregated view on how much total bit rate are spent in the base layer for all users ($\sum_i r b_{i,j}$), and on what the upper bound on total FGS rate is at the j th time slot (R_j) according to (19). This is as if the aggregated rates are applied to a single "super-user." The β factor is applied to R_j to obtain a fractional FGS rate budget R_j^f that helps overcome the quality fluctuation.

Next, we distribute R_j^f to each user. For applications that desire uniform quality among users, the fractional rate budget for each user $\{R_{i,j}^f\}$ is determined through the following optimization formulation

$$\begin{aligned} & \min_{R_{1,j}^f, \dots, R_{N,j}^f} D_j \\ & \text{subject to} \begin{cases} D_j \triangleq D_{1,j}(R_{1,j}^f) = \dots = D_{N,j}(R_{N,j}^f) \\ \sum_{i=1}^N R_{i,j}^f \leq \beta R_j. \end{cases} \end{aligned} \quad (24)$$

Since the R-D functions are monotonically decreasing, this optimization problem with equality constraints can be easily solved using bisection search. The search algorithm calculates the total required rates to achieve a target distortion, and then increases the target distortion at the next iteration if the total required rates is higher than the rate constraint and vice versa.

Finally, we determine the allocated FGS rate for each user $r f_{i,j}$ using a similar linear combination as in (13)

$$r f_{i,j} = \min \left\{ w_p R_{i,j}^p + (1 - w_p) R_{i,j}^f, R_{i,j} \right\} \quad (25)$$

where $R_{i,j}^p$ represents the FGS rate for the i th user in the j th frame (time slot) in order to maintain the same quality as the previous frame, and $R_{i,j}$ is the upper bound in (18).

C. Differentiated Service (DS)

Differentiated service (DS) refers to a service in which each user receives different quality according to his/her service agreement with the server. We consider a scenario that at the beginning of the service, each user submits his/her priority request, quantified by $DS_i \in (0, 1]$, such that the average distortion received by each user normalized by DS_i is constant:

$$\frac{\text{aveMSE}_1}{DS_1} = \frac{\text{aveMSE}_2}{DS_2} = \dots = \frac{\text{aveMSE}_N}{DS_N}. \quad (26)$$

In other words, a user who specifies a smaller value of DS_i (and possibly pays a premium fee in return) will receive a higher overall perceptual quality. This can be achieved by modifying the optimization problem in (24) as follows:

$$\begin{aligned} & \min_{R_{1,j}^f, \dots, R_{N,j}^f} \bar{D}_j \\ & \text{subject to} \begin{cases} \bar{D}_j \triangleq \frac{D_{1,j}(R_{1,j}^f)}{DS_1} = \dots = \frac{D_{N,j}(R_{N,j}^f)}{DS_N} \\ \sum_{i=1}^N R_{i,j}^f \leq \beta R_j. \end{cases} \end{aligned} \quad (27)$$

TABLE III
PROPOSED MULTIUSER RESOURCE ALLOCATION ALGORITHM (*M-LDLF*)

1. Initialization:
$j = 1$, Set $RDSC=1$, $B_0^e = 0$, $B_{i,0}^e = 0$, $B_{i,1}^d = \dots = B_{i,d_i^c}^d = 0$, $R_{i,0}^p = r b_{i,1}$, $r b_{i,0} = r f_{i,0} = 0$, $w_p = w_L$, and $MSE_{i,0} = D_{i,1}(r b_{i,1})$, $\forall i$.
2. While the last frame of this video is not reached:
a) Calculate the budget factor and weight factor $\bar{r}b = \frac{1}{j - \max(j-L, 0)} \sum_{i=1}^N \sum_{k=\max(j-L, 0)+1}^j r b_{i,k}$, $\beta = \frac{C_{max} - \bar{r}b}{B_{max}^e + C_{max} - \bar{r}b} + \Delta\beta$, $R_{i,j}^p = D_{i,j}^{-1}(MSE_{i,j-1}) - r b_{i,j}$, $\forall i$. If $\frac{1}{N} \sum_{i=1}^N (R_{i,j}^p - R_{i,j-1}^p / R_{i,j-1}^p) \geq S_T$, then $RDSC=j$. If $j \in [RDSC, RDSC + P_T]$, then $w_p = w_L$, else $w_p = W(B_{j-1}^e)$.
b) Select the channel transmission rate $C_{i,j}^U = B_{i,max}^d - B_{i,j+d_i^c-1}^d + r_{i,j-d_i^d} \forall i$, $C_{i,j}^L = \max\{-B_{i,j+d_i^c-1}^d + r_{i,j-d_i^d}, 0\} \forall i$, If $\sum_{i=1}^N C_{i,j}^U \geq C_{max}$, then $C_{i,j} = \min\{C_{i,j}^L + (C_{max} - \sum_{i=1}^N C_{i,j}^L) \frac{r_{i,j-1}}{\sum_{i=1}^N (r_{i,j-1})}, C_{i,j}^U\}$, Else $C_{i,j} = C_{i,j}^U$.
c) Select FGS rate
i) $R_{i,j} = B_{max}^e + C_{i,j} - B_{i,j-1}^e - r b_{i,j}$, $\forall i$.
ii) $R_j = B_{max}^e + \sum_{i=1}^N C_{i,j} - \sum_{i=1}^N B_{i,j-1}^e - \sum_{i=1}^N r b_{i,j}$.
iii) Solve the optimization problem: $\min_{R_{1,j}^f, \dots, R_{N,j}^f} D_j$, subject to Quality Constraint $\bar{D}_j \triangleq \frac{D_{1,j}(R_{1,j}^f)}{DS_1} = \dots = \frac{D_{N,j}(R_{N,j}^f)}{DS_N}$; Rate Constraint $\sum_{i=1}^N R_{i,j}^f \leq \beta R_j$.
iv) $r f_{i,j} = \min\{w_p R_{i,j}^p + (1 - w_p) R_{i,j}^f, R_{i,j}\}$, $\forall i$.
d) Update the encoder buffer occupancy information: $B_{i,j}^e = \max\{B_{i,j-1}^e + r b_{i,j} + r f_{i,j} - C_{i,j}, 0\}$. If $B_{i,j}^e = 0$, then $C_{i,j} = B_{i,j-1}^e + r b_{i,j} + r f_{i,j}$, $\forall i$.
e) Update the decoder buffer occupancy information: $B_{i,j+d_i^c}^d = B_{i,j+d_i^c-1}^d + C_{i,j} - r_{i,j-d_i^d} \forall i$.
f) $j = j + 1$.

The uniform quality problem of (24) is a special case of (27) when all DS_i 's are the same. This generalized optimization problem can also be solved using bisection search. We present the complete multiuser algorithm in Table III.

VI. EXPERIMENTAL RESULTS

In this section, we examine the performance of the proposed low-delay resource allocation algorithm with low-fluctuation of quality (*LDLF*), and compare it with two alternatives. The first alternative is the constant-bitrate (CBR) approach, which assigns a constant bit rate to each frame. The second alternative is a look-ahead sliding-window algorithm (*SWLF*) with buffer constraints adapted from [1], the details of which are given in Appendix II. Three statistics are used to evaluate the proposed algorithm and the two alternatives: the average PSNR (*avePSNR*), the mean of absolute difference of PSNR (*madPSNR*), and the overall channel utilization (*ChUtiliz*).

A. Experiment Setup

We concatenate 15 QCIF (176×144) video sequences to form one testing video sequence of 5760 frames. The 15 sequences are 300-frame *Akiyo*, 360-frame *carphone*, 480-frame *Claire*, 300-frame *coastguard*, 300-frame *container*, 390-frame *foreman*, 870-frame *grandmother*, 330-frame *hall objects*,

150-frame *Miss American*, 960-frame *mother and daughter*, 300-frame *MPEG4 news*, 420-frame *salesman*, 300-frame *silent*, 150-frame *Suzie*, and 150-frame *Trevor*. The base layer is generated by MPEG-4 encoder with a fixed quantization step of 30 and the GOP pattern is 29 P-frames after one I-frame. All frames of FGS layer have up to six bit planes. For N users in this system, we allocate $N * 80$ K bits for the server buffer and the shared maximal channel capacity is $N * 960$ kb/s. Each user has a small decoder buffer of 400 kb. For each user, the transmission delay, d_i^c , is 3 frames and initial playback delay, d_i^d , is 3 frames. The parameters $(a, b, P_T, S_T, \Delta\beta, L)$ used in the *LDLF* algorithm are set to $(B_{\max}^e/4, 0.75, 3, 0.3, 0.01, 30)$.

B. Experimental Results for Single-User Rate Control

For the single-user system, the video content is picked from frame 301 to 2100, corresponding to the video sequences of *carphone*, *Claire*, *coastguard*, *container*, and *foreman*. Fig. 6 shows the *avePSNR*, *madPSNR*, and *ChUtiliz* using the three different algorithms.

The solid line with triangle makers represents the results of single-user *SWLF* algorithm (*S-SWLF*) with different window sizes. The solid line represents the results of CBR approach, which encodes each frame at a constant bit rate of 32 kb. The dotted and dashed lines represent the results of the proposed single-user *LDLF* algorithm (*S-LDLF*) with $(w_H, w_L) = (0.95, 0.3)$ and $(0.98, 0.3)$, respectively. As the CBR and *S-LDLF* algorithms do not have the window parameter, we plot their results as horizontal lines to allow for the performance comparison among CBR, *S-LDLF*, and *S-SWLF* of different window sizes.

As we can see from Fig. 6, CBR approach provides the highest average perceptual quality and channel utilization. However, CBR has the worst fluctuation of visual quality. In contrast, our experiment shows that variable rate control for video, such as *S-LDLF* and *S-SWLF*, can provide more consistent quality. For the proposed *S-LDLF* algorithm, a higher value of w_H gives smaller *madPSNR* and a little lower *avePSNR* as expected. We also observe that *madPSNR* decreases when the window size increases in *S-SWLF* algorithm. To provide sufficient smoothing, the window size L of the *S-SWLF* algorithm needs to be at least the size of half to one GOP to cover an I-frame of high data rate, which is 15–30 frames in our experiment. We compare *S-SWLF* with window size 30 frames with *S-LDLF* with $(w_H, w_L) = (0.98, 0.3)$, and present the PSNR results for each frame in Fig. 7. We can see that *S-SWLF* and *S-LDLF* have similar performance in terms of visual quality. However, to achieve this comparable performance, *S-SWLF* requires about one-second more delay ($L = 30$ frames) and a corresponding large amount of extra storage for the look-ahead frames on the encoder side, while the proposed *S-LDLF* algorithm requires no extra delay and storage on the encoder side.

C. Experimental Results for Multiuser Resource Allocation

For the multiuser system, the content program for each user is 1200 frames long and starts from a randomly selected I-frame of the testing video source. If the length of this video source

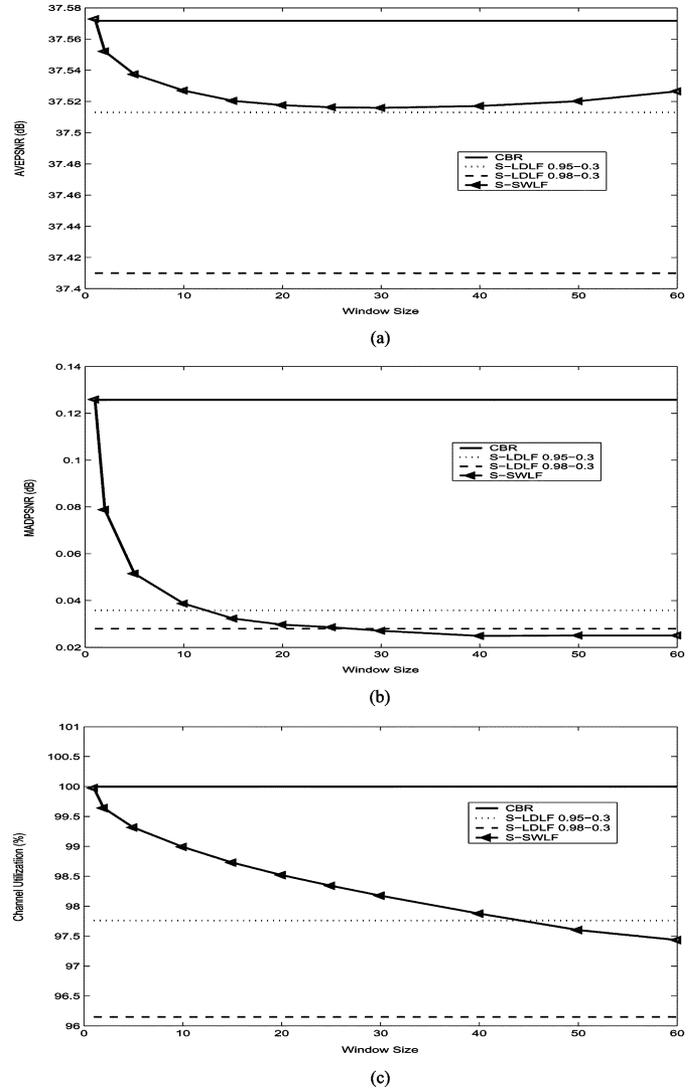


Fig. 6. Comparison of CBR, *S-SWLF*, and the proposed *S-LDLF* rate control algorithms for video frame 301–2100 from the testing sequence. (a) *avePSNR*. (b) *madPSNR*. (c) Channel utilization.

is not long enough, we loop from the beginning of the testing sequence. We repeat the simulation multiple times for a total of about 25 000 user cases to obtain the averaged results for the systems with 4, 8, 16, 32, 64, and 128 users.

We first demonstrate the performance when all users request the same level of visual quality. Fig. 8 shows the average of all users' *avePSNR*, *madPSNR*, and *ChUtiliz* for different number of users using four algorithms, namely, the CBR algorithm (*CBR*), the multiuser *SWLF* algorithm (*M-SWLF*) with different window sizes, the multiple single-user approach using the above *S-LDLF* (*M S-LDLF*) with $(w_H, w_L) = (0.98, 0.3)$, and the proposed multiuser *LDLF* (*M-LDLF*) approach with $(w_H, w_L) = (0.98, 0.3)$. The CBR approach assigns each user a fixed encoding rate, 32 kb/frame. The *M S-LDLF* system provides individual encoder buffer (80 kb) and channel bandwidth (960 kb/s) to each user using *S-LDLF* algorithm. As we can see from Fig. 8, the CBR approach suffers from much higher fluctuation of visual quality than any other approaches, suggesting once again the need of variable rate control. Among

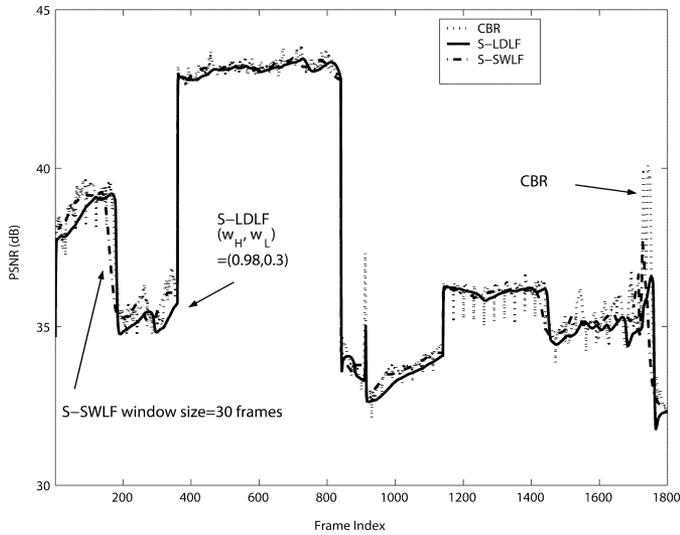


Fig. 7. Frame-by-frame PSNR comparison of the CBR, S-SWLF, and the proposed S-LDLF approaches.

the three approaches providing VBR video, the two joint resource allocation approaches for multiuser system, *M-SWLF* and *M-LDLF*, provide smaller fluctuation of visual quality than the individual resource allocation scheme (*M S-LDLF*). The more users a system has, the higher possibility we can take the advantage of the variations of video content similar to those in multiplexing [4] and offer desired quality to each user through dynamic bandwidth allocation. Comparing these two dynamic multiuser algorithms, the fluctuation of visual quality of *M-LDLF* algorithm is between the quality fluctuations of the *M-SWLF* algorithm with window sizes of 15 and 30 frames; and the performance of both visual quality measurements, *avePSNR* and *madPSNR*, of the *M-LDLF* algorithm approaches the results of *M-SWLF* with window size 30 frames when the number of users increases.

To compare the frame-by-frame PSNR of the three algorithms, *M S-LDLF*, *M-SWLF*, and *M-LDLF*, we simulate the scenario in which the content program for user i is 1200 frames long and starts from frame $600 \times (i - 1) + 1$ of the testing video source. Fig. 9 shows the frame-by-frame PSNR of the first and tenth users in the *M S-LDLF*, *M-SWLF*, and *M-LDLF* systems when there are 16 and 32 users in the systems, respectively. Again, we see that the dynamic multiuser approaches (*M-LDLF* and *M-SWLF*) can provide more uniform quality than the multiple single-user approach both within a scene and when crossing scene boundaries. When the number of users increases, the gain from joint resource allocation is more significant, providing more uniform quality and less quality fluctuation. Between the two dynamic multiuser approaches, our proposed *M-LDLF* approach can achieve similar perceptual quality to that of *M-SWLF* approach with large window size (30 frames); however, similar to the single-user case, the prior work *M-SWLF* needs a longer delay (1 s) and a substantially larger storage than the propose approach. This additional storage is for keeping the look-ahead data of all users. In the example illustrated above, *M-SWLF* needs an extra storage of 30 frames/user \times 32 users = 960 frames. As a result, the proposed approach

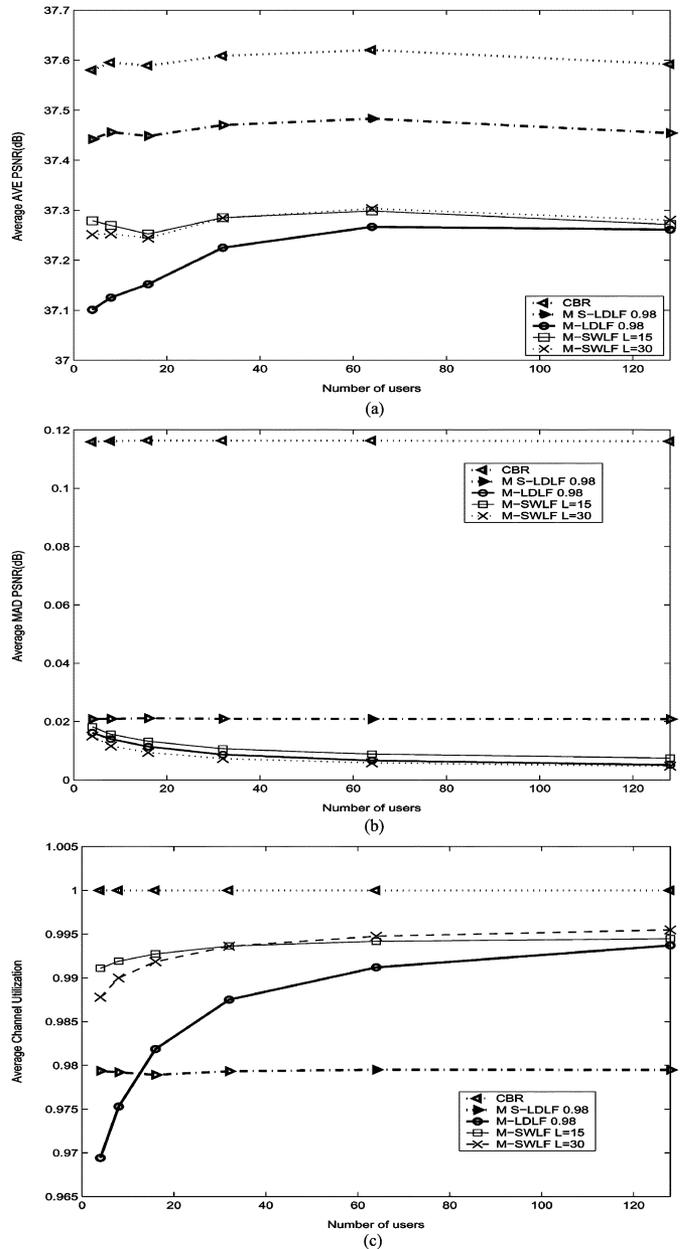


Fig. 8. Comparison of CBR, M-SWLF, M S-LDLF, and M-LDLF systems under uniform service for all users. Shown here are three statistics averaged among all users. (a) Average *avePSNR* among all users. (b) Average *madPSNR* among all users. (c) Average channel utilization among all users.

has higher system scalability than the *M-SWLF* approach. This makes the proposed scheme an attractive choice for building a large system to accommodate many users.

We now use Fig. 10 to demonstrate the differentiated service of a 32-user system by keeping the same system settings as in Fig. 9 except setting the differentiated service priority for the i th user as follows:

$$DS_i = 1 - \frac{0.9(i-1)}{31}. \quad (28)$$

Fig. 10(a) illustrates the average MSE for each user. As we can see, the proposed algorithm can achieve the required differentiated service priority, which is almost a linear line as we have designed for in (28). Fig. 10(b) and (c) highlight the received

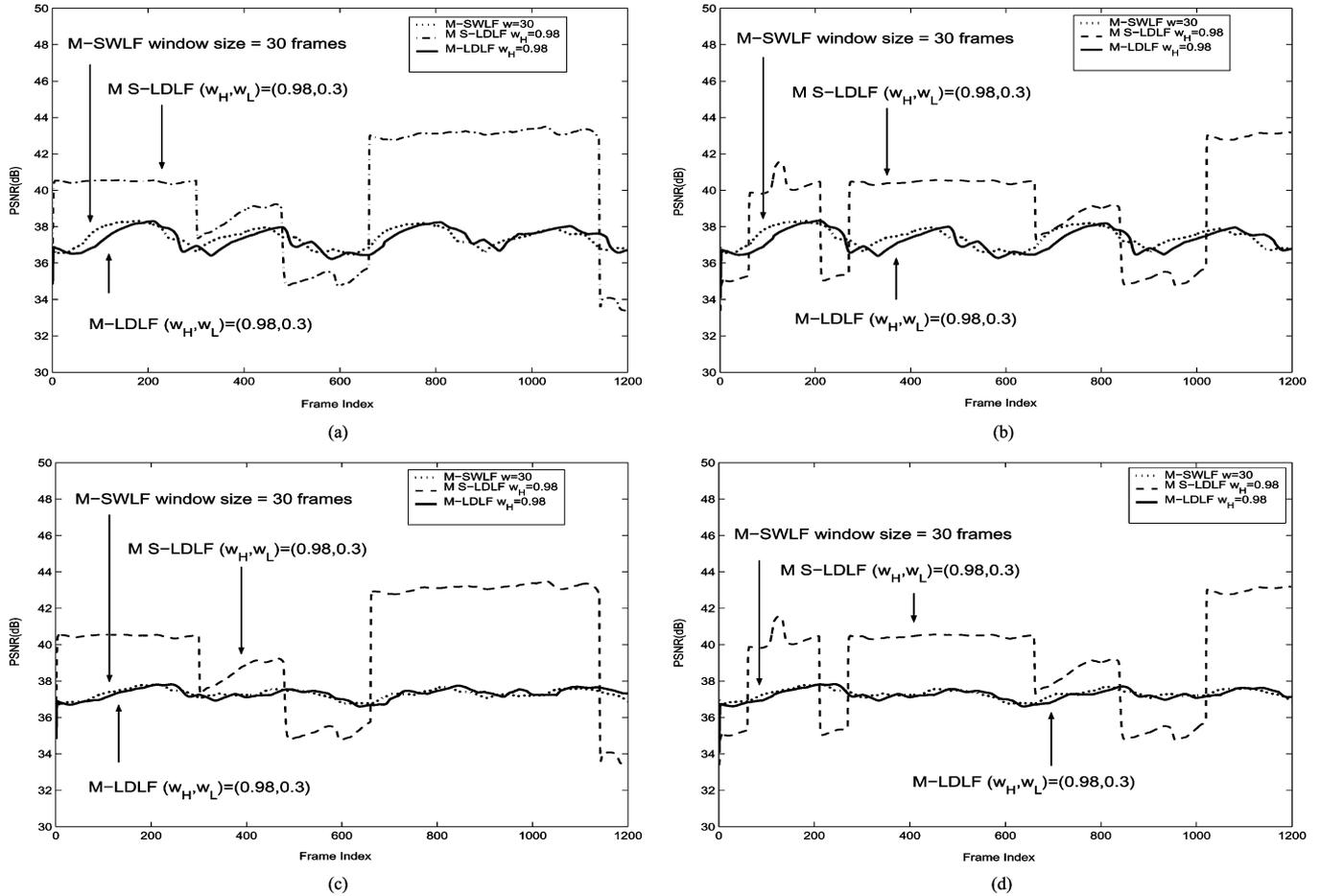


Fig. 9. Frame-by-frame PSNR results of the first and tenth user in the *M-S-LDLF*, *M-SWLF*, and *M-LDLF* systems. (a) First user in a 16-user system. (b) Tenth user in a 16-user system. (c) First user in a 32-user system. (d) Tenth user in a 32-user system.

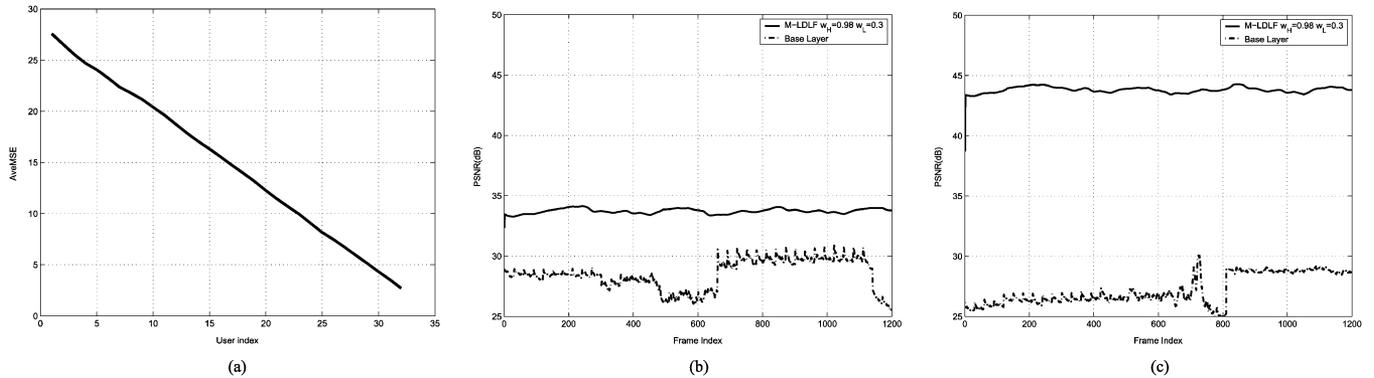


Fig. 10. Results of a 32-user system using *M-LDLF* with differentiated service. (a) *aveMSE* for each user. (b) PSNR for the first user. (c) PSNR for the 32nd user.

PSNR for the first and the last user in this system, who request the lowest and the highest video quality, respectively.

VII. CONCLUSION

In summary, we have proposed an efficient bandwidth resource allocation algorithm for streaming multiple MPEG-4 FGS video sequences. By exploring the intra- and inter-frame R-D characteristics of MPEG-4 FGS codec, we present a control policy to achieve an excellent tradeoff between the average quality and quality fluctuation criteria. We demonstrate

that multiuser systems with dynamic joint resource allocation provide more consistent quality than the multiple single-user approaches that do not dynamically share resources. Evaluating the video quality in terms of the average distortion and the quality fluctuation, our algorithm gives excellent performance comparable to those by the general look-ahead sliding-window approach. But compared to the existing approaches, our algorithm has higher system scalability, as it does not need a delay of dozens of frames' long and does not require extra storage proportional to the number of users. Therefore, the proposed multiuser resource allocation algorithm with low

delay and low fluctuation of quality can serve as an efficient and effective building block for real-time multiuser broad-band communications.

APPENDIX I

IMPACT OF BUDGET FACTOR ON PERCEPTUAL QUALITY

In this appendix, we present detailed rationale behind the bandwidth resource allocation algorithm proposed in Section IV-D. In particular, for a video scene consisting of similar R-D characteristics, we analyze the trend of the *aveMSE* and the *madMSE* as β changes, and derive the result of (15) for β_0 .

Consider an M -frame video clip with similar visual contents, whereby the R-D model of each frame within the clip are similar to each other. As such, it is reasonable to assume that the feasible range of FGS data rates for all frames are within the same bit plane, and $R_j^p \approx r f_{j-1}$. The R-D model can be expressed as follows for rates falling in the range of interest

$$D_j(r_j) = k_1 - k_2 r f_j, \quad \text{for all } j \quad (29)$$

where k_1 and k_2 are constants. We denote \overline{rb} and \overline{rf} as the average rate of base layer and FGS layer within this video sequence, respectively. Here \overline{rb} is fixed due to the use of constant quantization step for the base layer of all M frames.

Let us first consider *madMSE*, which can be represented as follows using (29)

$$\text{madMSE} = \frac{k_2}{M-1} \sum_{j=2}^M |\Delta r f_j| \quad (30)$$

where $\Delta r f_j = r f_j - r f_{j-1}$. We examine the absolute difference of FGS rates between two frames at time slot j

$$|\Delta r f_j| = \left| (1-w_p)\beta[(C_j - C_{j-1}) + (B_{j-2}^e - B_{j-1}^e) + (rb_{j-1} - rb_j)] + w_p \Delta r f_{j-1} \right|. \quad (31)$$

With a fixed w_p , a larger β would usually result in larger $|\Delta r f_j|$ hence larger *madMSE* by (30).

Next, we consider *aveMSE*. With the R-D model in (29), we can express *aveMSE* as follows:

$$\text{aveMSE} = k_1 - k_2 \overline{rf}. \quad (32)$$

Summing up from $r f_1$ to $r f_M$ using (12) and (13) and taking the average, we obtain

$$\overline{rf} = w_p \overline{R^p} + (1-w_p)\beta(B_{\max}^e + \overline{C} - \overline{B}^e - \overline{rb}) \quad (33)$$

where $\overline{R^p}$, \overline{C} , \overline{B}^e represent the average of their corresponding rates or buffer occupancy. Bringing $\overline{R^p} \approx \overline{rf}$, the condition of encoder buffer occupancy $\overline{B}^e \approx 0$, and the conservation law of data flow that the overall input flow should be equal to the overall output flow

$$\overline{C} = \overline{rb} + \overline{rf} \quad (34)$$

we arrive at

$$\overline{rf} \approx \frac{\beta B_{\max}^e}{(1-\beta)}. \quad (35)$$

Taking (35) into (32), we can represent *aveMSE* as a function of β

$$\text{aveMSE} \approx k_1 - \frac{k_2 \beta B_{\max}^e}{(1-\beta)}. \quad (36)$$

Thus, if the system has a larger β , the average distortion (*aveMSE*) will decrease. Note that w_p does not affect *aveMSE* as long as $0 \leq q w_p < 1$.

The above analysis shows that a larger β reduces *aveMSE* but leads to larger *madMSE*. To complete the derivation for β_0 , we observe from (34) that since \overline{rb} is fixed, increasing \overline{C} is equivalent to increasing \overline{rf} . But when \overline{C} approaches the channel capacity C_{\max} , \overline{rf} and *aveMSE* cannot be further improved. Thus, there exists β_0 such that its corresponding $\overline{rf}(\beta_0)$ from (35) is equal to $C_{\max} - \overline{rb}$, i.e.,

$$\frac{\beta_0 B_{\max}^e}{(1-\beta_0)} = C_{\max} - \overline{rb}. \quad (37)$$

Recalling the results in (30) and (36), the selection of $\beta = \beta_0$ can give an excellent tradeoff between *aveMSE* and *madMSE*. Solving (37) for β_0 , we arrive at the tradeoff point

$$\beta_0 = \frac{C_{\max} - \overline{rb}}{B_{\max}^e + C_{\max} - \overline{rb}}. \quad (38)$$

APPENDIX II

REVIEW OF AN ALTERNATIVE SLIDING-WINDOW ALGORITHM

The sliding-window algorithm is originally proposed in [1] and only concerns the channel capacity without considering the buffer and delay constraints. To determine the bit rate for a frame, the sliding window algorithm requires the complete bit rate and R-D information of L frames ahead. The algorithm then distributes the FGS rates to the current frame by solving an optimization problem that all frames within the look-ahead window have the uniform and highest possible quality subject to a FGS rate budget for all frames in this sliding window, denoted W_j^F . This FGS rate budget is obtained by subtracting all base layer rates within the window from an overall rate budget, W_j . The rate budget W_j is updated by removing the bandwidth used for the previous frame and adding the currently available channel transmission rate. That is

$$W_j = W_{j-1} - r_{j-1} + C_j \quad (39)$$

$$W_j^F = W_j - \sum_{k=j}^{j+L-1} r b_k. \quad (40)$$

For a fair comparison with our proposed algorithms, we modify the sliding window approach by adding the delay and encoder/decoder buffer constraints. We also make two further modifications to fit in the scenarios considered in this paper.

The first modification is on W_j^F . When the sliding window does not cross the scene boundary, it is reasonable to assume that the frames within the sliding window have similar R-D characteristics. Under this assumption, the estimated total transmission rate for these L frames in the sliding window is $L \cdot C_j^{(U)}$, where $C_j^{(U)}$ is the upper bound in (7). To keep the occupancy

of the encoder buffer low, the data left in the encoder buffer at time slot $j - 1$ is flushed out during the next L time slots. Thus, the modified FGS rate budget for a sliding window is

$$W_j^F = L \cdot C_j^{(U)} - B_{j-1}^e - \sum_{k=j}^{j+L-1} r b_k. \quad (41)$$

Second, we observe that when a sliding window enters a segment with simple R-D characteristics from a past segment with complex R-D characteristics, the encoder buffer may overflow and force the system to drop more FGS layer data, which leads to several severe quality fluctuations near the R-D characteristics dissimilarity boundaries (which often coincide with scene changes). To overcome this problem and allow fair comparison with our proposed schemes, we detect the change in R-D characteristics once it appears in the sliding window and adaptively reduce the window size so that the sliding window does not across this dissimilarity boundary. After passing the boundary, the size of the sliding window is restored to the original size.

REFERENCES

- [1] X. M. Zhang, A. Vetro, Y. Q. Shi, and H. Sun, "Constant quality constrained rate allocation for FGS video coded videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 2, pp. 121–130, Feb. 2003.
- [2] L. Wang and A. Vincent, "Bit allocation and constraints for joint coding of multiple video programs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 6, pp. 949–959, Sep. 1999.
- [3] L. Boroczky, A. Y. Nagi, and E. F. Westermann, "Joint rate control with look-ahead for multi-program video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 7, pp. 1159–1163, Oct. 2000.
- [4] M. Wu, R. Joyce, H.-S. Wong, L. Guan, and S.-Y. Kung, "Dynamic resource allocation via video content and short-term traffic statistics," *IEEE Trans. Multimedia*, vol. 3, no. 2, pp. 186–199, Jun. 2001.
- [5] C.-Y. Hsu, A. Ortega, and A. R. Reibman, "Joint selection of source and channel rate for VBR video transmission under ATM policing constraints," *IEEE J. Sel. Areas Commun.*, vol. 15, pp. 1016–1028, Aug. 1997.
- [6] L.-J. Lin and A. Ortega, "Bit-rate control using piecewise approximated rate-distortion characteristics," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 4, pp. 446–459, Aug. 1998.
- [7] Y. Yang and S. S. Hemami, "Rate control for VBR video over ATM: Simplification and implementation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 9, pp. 1045–1058, Sep. 2001.
- [8] J. Xu, Z. Xiong, S. Li, and Y.-Q. Zhang, "Memory-constrained 3-D wavelet transform for video coding without boundary effects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 9, pp. 812–818, Sep. 2002.
- [9] L. Zhao, J. Kim, and C.-C. J. Kuo, "MPEG-4 FGS video streaming with constant-quality rate control and differentiated forwarding," in *Proc. SPIE Conf. Visual Communications and Image Processing*, 2002, pp. 230–241.
- [10] *ISO/IEC JTC1/SC29/WG11 MPEG-4 Video Verification Model version 18.0 N3908*, Jan. 2001.
- [11] W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 301–317, Mar. 2001.
- [12] H. M. Radha, M. van der Schaar, and Y. Chen, "The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP," *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 53–68, Mar. 2001.
- [13] M. van der Schaar and H. M. Radha, "A hybrid temporal-SNR fine-granular scalability for internet video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 318–331, Mar. 2001.
- [14] T. V. Lakshman, A. Ortega, and A. R. Reibman, "VBR video: Tradeoffs and potentials," *Proc. IEEE*, vol. 86, no. 5, pp. 952–973, May 1998.
- [15] M.-T. Sun and A. R. Reibman, *Compressed Video Over Networks*. New York: Marcel Dekker, 2001.
- [16] A. R. Reibman and B. G. Haskell, "Constraints on variable bit-rate video for ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 4, pp. 361–372, Dec. 1992.
- [17] W. Ding, "Joint encoder and channel rate control of VBR video over ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 266–278, Apr. 1997.
- [18] E. C. Reed and J. S. Lim, "Optimal multidimensional bit-rate control for video communication," *IEEE Trans. Image Process.*, vol. 11, no. 8, pp. 873–885, Aug. 2002.
- [19] C. E. Luna, L. P. Kondi, and A. K. Katsaggelos, "Maximizing user utility in video streaming applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 2, pp. 141–148, Feb. 2003.
- [20] X. S. Zhou and S.-P. Liou, "Optimal nonlinear sampling for video streaming at low bit rates," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 535–544, Jun. 2002.
- [21] J. Ribas-Corbera and S. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 1, pp. 172–185, Feb. 1999.
- [22] H.-J. Lee, T. Chiang, and Y.-Q. Zhang, "Scalable rate control for MPEG-4 video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 6, pp. 878–894, Sep. 2000.
- [23] K. N. Ngan, T. Meier, and Z. Chen, "Improved single-video-object rate control for MPEG-4," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 5, pp. 385–393, May 2003.
- [24] Z. He and S. K. Mitra, "A linear source model and a unified rate control algorithm for DCT video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 11, pp. 970–982, Nov. 2002.
- [25] H. Radha, Y. Chen, K. Parthasarathy, and R. Cohen, "Scalable internet video using MPEG-4," *Signal Process.: Image Commun.*, vol. 15, pp. 95–126, Sep. 1999.
- [26] R. Cohen and H. Radha, "Streaming fine-grained scalable video over packet-based networks," in *IEEE Global Telecommunications Conf.*, vol. 1, 2000, pp. 288–292.
- [27] S. P. Sethi and Q. Zhang, *Hierarchical Decision Making in Stochastic Manufacturing Systems*. Cambridge, MA: Birkhäuser Boston, 1994.
- [28] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger, "Long-range dependence in variable-bit-rate video traffic," *IEEE Trans. Commun.*, vol. 43, no. 2, pp. 1566–1579, Feb. 1995.



Guan-Ming Su (S'04) received the B.S.E. degree from National Taiwan University, Taipei, Taiwan, R.O.C., in 1996 and the M.S. degree from University of Maryland, College Park, in 2001, both in electrical engineering. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering at the University of Maryland, College Park.

He was with the Research and Development Department, Qualcomm, Inc, San Diego, CA, during the summer of 2005. His research interests are multimedia communications and multimedia signal processing.



Min Wu (S'95–M'01) received the B.E. degree in electrical engineering and the B.A. degree in economics from Tsinghua University, Beijing, China, in 1996 (both with the highest honors), and the M.A. and Ph.D. degrees in electrical engineering from Princeton University, Princeton, NJ, in 1998 and 2001, respectively.

She was with NEC Research Institute and Signafy, Inc. in 1998, and with Panasonic Information and Networking Laboratories in 1999. Since 2001, she has been an Assistant Professor of the Department of Electrical and Computer Engineering, the Institute of Advanced Computer Studies, and the Institute of Systems Research at the University of Maryland, College Park. Her research interests include information security, multimedia signal processing, and multimedia communications. She co-authored a book *Multimedia Data Hiding* (New York: Springer-Verlag, 2003) and holds four U.S. patents on multimedia security.

Dr. Wu received a CAREER award from the U.S. National Science Foundation in 2002, and a George Corcoran Education Award from University of Maryland in 2003, a TR100 Young Investigator Award from the MIT Technology Review Magazine in 2004, a EURASIP Best Paper Award, and a U.S. Office of Naval Research Young Investigator Award in 2005. She is a Member of the IEEE Technical Committees on Multimedia Signal Processing and on Multimedia Systems and Applications, Publicity Chair of 2003 IEEE International Conference on Multimedia and Expo, and a guest editor of the Special Issue on Media Security and Rights Management for the EURASIP Journal on Applied Signal Processing.