Allocation of Layer Bandwidths and FECs for Video Multicast Over Wired and Wireless Networks

T.-W. Angus Lee, S.-H. Gary Chan, Qian Zhang, *Member, IEEE*, Wen-Wu Zhu, *Senior Member, IEEE*, and Ya-Qin Zhang, *Fellow, IEEE*

***Abstract*—Layered multicast is an efficient technique to deliver video to heterogeneous receivers over wired and wireless networks. In this paper, we consider such a multicast system in which the server adapts the bandwidth and forward-error correction code (FEC) of each layer so as to maximize the overall video quality, given the heterogeneous client characteristics in terms of their end-to-end bandwidth, packet drop rate over the wired network, and bit-error rate in the wireless hop. In terms of FECs, we also study the value of a gateway which "transcodes" packet-level FECs to byte-level FECs before forwarding packets from the wired network to the wireless clients. We present an analysis of the system, propose an efficient algorithm on FEC allocation for the base layer, and formulate a dynamic program with a fast and accurate approximation for the joint bandwidth and FEC allocation of the enhancement layers. Our results show that a transcoding gateway performs only slightly better than the nontranscoding one in terms of end-to-end loss rate, and our allocation is effective in terms of FEC parity and bandwidth served to each user.**

***Index Terms*—Layered video multicast, optimal bandwidth allocation, optimal FEC, transcoding and nontranscoding gateways, wireless Internet.**

I. Introduction

Layered multicast is an efficient technique to deliver video to its end users [1], [2]. In such a system, the server encodes the videos (stored or captured live) into a certain fixed number of layers (i.e., a base layer and several enhancement layers) and multicasts the layers via several multicast groups to end-users distributed over a network. The base layer guarantees a certain minimum video quality, and hence has to be received with rather low loss [3]. Depending on the end-to-end bandwidth between a client and the server,[1] a client may progressively improve the video quality by getting a number of enhancement layers via joining their multicast groups. This is the so-called "receiver-driven layered multicast" [4], [5]. As a client would not join more layers than its end-to-end bit rate can accommodate (except for some transient attempts), the data is delivered in a "TCP-friendly" manner over the wired network [6]–[8].

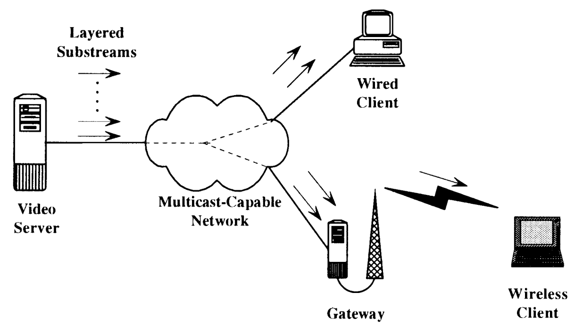


Fig. 1. Architecture of the video system.

We show in Fig. 1 the video system considered in this paper. There are both wired and wireless clients. In the case of wireless access, the base station is connected to a gateway. In its simplest form, the gateway forwards whatever packets it receives to the wireless clients without any re-packetization or fragmentation. On the other hand, a more sophisticated gateway can do some repacketization (e.g., by adding or removing error redundancy codes) before forwarding the packets from the wired infrastructure to the end clients. The data packets are said to be "transcoded" in the process. This kind of "transcoding" gateway may be beneficial since the error characteristics are different in the wired and wireless networks: in wired networks (such as the Internet), packets are dropped mainly due to congestion at the routers, while in the wireless hop, packets are often lost due to random *bit* errors caused by fading or multipath effect [9].

Due to the heterogeneous nature of channel conditions in terms of bandwidth and error rate among the clients and such conditions may vary over time, the source has to continuously adapt the error-recovery mechanisms and bit rate of each layer in order to optimize the overall video quality delivered to the clients. Such joint rate and error control across the layers is a major challenge in system design and is the topic of concern in this work.

To recover packet-loss, feedback recovery or forward-error correction code (FEC) may be used [3], [10]–[13]. In general, feedback recovery does not work very well over long distance with real-time guarantee. FEC such as the Reed–Solomon (RS) code, on the other hand, is more appropriate in this case [14]. It consists of arranging the data and redundancy bits in such a way that even when not all of the bits are received, the original data may still be recovered. Adapting FECs according to the network error conditions can be very efficient to maintain

Manuscript received September 4, 2001; revised September 2, 2001. This work was supported in part by the Areas of Excellence (AoE) Scheme on Information Technology under the University Grant Council of Hong Kong (AoE/E-01/99) and by the Sino Software Research Institute (SSRI00/01.EG04) at Hong Kong University of Science and Technology. This work was previously presented at the IEEE Globecom'01, San Antonio, TX. This paper was recommended by Associate Editor F. Pereira.

T.-W. A. Lee and S.-H. G. Chan are with the Department of Computer Science, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong (e-mail: gchan@cs.ust.hk).

Q. Zhang, W.-W. Zhu, and Y.-Q. Zhang are with the Microsoft Research Asia, Haidian District, Beijing 100080, China.

Digital Object Identifier 10.1109/TCSVT.2002.806816

[1]We have used "bandwidth" and "bit rate" interchangeably in this paper.

video quality. In this paper, we will hence focus on FECs as the error-recovery mechanism. Note that FEC strategies are different for the wireless and wired networks due to their different error characteristics. In the wired network, packet-level FEC in the form of parity *packets* should be used in order to recover packet loss [9], [13], [15]–[17]. On the other hand, byte-level FECs in the form of parity *bytes* should be used over the wireless hop in order to recover bit error [9]. Furthermore, different FECs may be applied for different layers (i.e., unequal error protection) to achieve error-resilience.

We consider that a client periodically reports to the source its current estimated end-to-end bandwidth, packet drop rate, and bit error rate (in the case of a wireless client) between itself and the source by means of some scalable feedback mechanism [6], [7]. (How to estimate these parameters accurately is beyond the scope of this work. Interested readers are referred to [18] and references therein.) It is, therefore, of particular interest to address the following issue: given the heterogeneous error and bandwidth characteristics of its end clients, how should the server allocate the bandwidth and the corresponding packet-and-byte-level FEC of each video layer in order to maximize the overall video quality? Furthermore, are there any differences in performance between a simple "nontranscoding" gateway and the more complicated "transcoding" gateway? Note that the network condition may be nonstationary over time (i.e., packet drop or loss may be bursty). Due to periodic feedback from the clients, the server continuously adapts its FEC and bandwidth allocation according to the network conditions at that time. We, however, assume that the packet drop and bit error are independent between any two adaptation periods, as normally considered in the multistate Markov model in the literature.

Traditionally, video quality is measured in terms of PSNR [19]. In order to offer a good video quality, the packet-loss rate after error correction (i.e., the "residual" packet-loss rate) has to be below a certain (low) value, e.g., $\leq$1% for the base layer and $\leq$2% for enhancement layers. Such a loss rate is essential for effective error concealment. Under such a loss rate, it has been widely observed that the peak signal-to-noise ratio (PSNR) is proportional to the video goodput defined as the useful data bits per second (after error correction) received by a client [15], [20], [21]. Therefore, our objective of maximizing video quality is equivalent to maximizing the overall goodput of the system, subject to a certain low loss constraint for each layer. Since the base layer is the most important layer that all clients must receive, we first optimize its overall quality. Given that, we then jointly allocate bandwidth and FECs for each of the enhancement layers so that the overall goodput in the network is maximized. As the server has to continuously adapt the bandwidth and FEC of the layers, the optimization has to be fast and, in the case of approximation, has to be accurate.

Our contributions in this paper are hence as follows.

1) We study a video multicast system over wired and wireless networks with joint bandwidth and FEC allocation for each layer in order to maximize the overall video quality.
2) We present an analytic model of the system, an efficient algorithm on optimal FEC allocation for the base layer, and a dynamic program formulation with a fast and accurate approximation on the optimal allocation of the enhancement layers.
3) We investigate the advantages of using a gateway which transcodes from packet-level FECs to byte-level FECs for the wireless link.

We briefly present some previous works as follows. There has been much work on using error control for *unicast* video delivery [6], [7], [22]. We study video multicast here. Several error-recovery schemes have been studied for video multicast, such as the limited retransmission and FEC (the so-called hybrid ARQ-FEC) and sending delayed version of parity packets over different multicast groups [23]–[25]. An evaluation of the application of FECs on unequal packet-loss protection for video streaming has been studied in [26]. While all these address *how* FEC can be introduced and applied in a video system, in this paper, we address *how much* FECs is required, and other important issues such as the optimal bandwidth of each layer and the value of a transcoding gateway. All of the above has not considered the issue of "mixed" media (wired and wireless networks) in which packet-level and byte-level FECs should be combined for optimal system operation. Bandwidth allocation at receivers for layered multicast has been examined in [27]. Here, we address a different system (a receiver-driven multicast system) with allocation at the sender. There is also much other work on layered multicast [28]–[31]. Our work differs from all of this in that we examine layered multicast over mixed media with joint bandwidth and FEC allocation, and advantages of using a transcoding wireless gateway [32].

This paper is organized as follows. We first present the packet- and byte-level FEC allocation schemes, and analyze how the schemes can be applied to the base layer for optimal quality in Section II. Then we present the dynamic program formulation for the joint allocation of bandwidth and FEC across enhancement layers in Section III. We conclude in Section IV.

II. Base-Layer Transmission and Its Optimization

Since every client has to receive the base layer, the bit rate allocated to the base layer (including FEC encoding) should be equal to the minimum end-to-end bit rate (Those clients with higher end-to-end bit rate may join the enhancement layers to improve further their video quality). Thus, the only concern in base-layer transmission is how much error control should be applied to serve both wireless and wired clients so that their overall quality is optimized. As noted before, the quality is measured by the aggregate goodput in the system, or equivalently, average goodput of the clients.

In this section, we first describe packet-level and byte-level FEC schemes in Section II-A. In Section II-B, we analyze and optimize video quality in terms of system goodput for nontranscoding and transcoding gateway, given client packet-loss and bit-error rate. Finally, we present some illustrative numerical examples and discuss the effectiveness of the error control schemes in Section II-C.

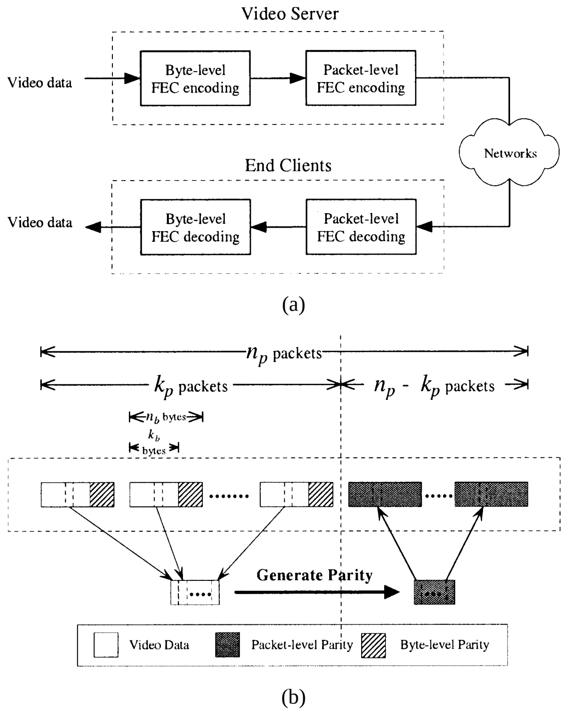


Fig. 2. Packet-level and byte-level FEC scheme for nontranscoding gateway. (a) Data flow of the scheme. (b) FEC generation of the scheme.

A. FEC Scheme Descriptions

We propose mixed packet-level and byte-level FECs to protect the base layer and study the scheme with and without a transcoding gateway.

1) Nontranscoding Gateway: With a nontranscoding gateway, both packet-level and byte-level FEC encodings have to be done at the video server, and error correction are only done at the end clients. We show the flow of FEC encoding and decoding in Fig. 2(a). At the server, the compressed stream is first encoded with byte-level FEC followed by packet-level FEC. The decoding part is the reverse of the encoding process. Note that with this system, the byte-level FEC does not really helps those wired clients (where packet drops occur) in improving their error resilience capability.

In Fig. 2(b), we show how to generate the two levels of FEC based on RS code. For the byte-level FEC, the encoder processes in symbols, where each symbol consists of m bits ($m = 8$ in general). Given a packet of size n_b bytes, $k_b(\geq 1)$ bytes of source data are packed with $n_b - k_b$ parity bytes, where $k_b = n_b, n_b - 2, \ldots$. This is the so-called $\mathrm{RS}(n_b, k_b)$ code, which is able to correct up to t_b symbol errors in a packet, where

$$t_b = \left\lceil \frac{n_b - k_b}{2} \right\rceil . \tag{1}$$

The packet size n_b is limited by $2^m - 1$ symbols; therefore, for $m = 8, n_b \leq 255$.

With every k_p of these byte-encoded video packets, a packet-level FEC is then applied to generate $n_p - k_p$ parity packets to form a block of n_p packets. This is generated as follows. The ith byte of each of the k_p video packets $(1 \leq i \leq n_b)$ is taken out to generate $n_p - k_p$ parity bytes. The generated parity bytes are then redistributed as the ith byte of each of the $n_p - k_p$ parity packets. Since all the packets are sequenced, up to

$$t_p = n_p - k_p \tag{2}$$

packet losses in a block can be corrected. Clearly, as a block of packets has to be ready before packet-level FEC is done, the delay of the system increases with n_p. Therefore, in reality, the client delay requirement determines the n_p that can be used.

The server computes the optimal allocation between the video data rate, the packet-level FEC rate (defined as the number of packet-level FEC parity bits per second), and the byte-level FEC rate (defined as the number of byte-level FEC parity bits per second) given the feedbacks from the end clients. Let G be the multicast group size. The feedbacks for client $g(1 \leq g \leq G)$ are in terms of the estimated end-to-end available bit rate $\hat{B}_g$ and the packet drop rate $\hat{P}_{l,g}(\hat{P}_{l,g}$ may be estimated by the missing sequence numbers of the packets) [2] and, for wireless clients, the bit-error rate of the wireless hop $\hat{e}_{b,g}$ (note that $\hat{e}_{b,g}$ may be estimated after accounting for the limited ARQ recovery in the wireless hop or by using a two-state Markov process as given in [33]).

Given the feedback information, the server has to first decide the packet-level and byte-level FEC rates for the base layer, with its transmission rate, including all the redundant bits is equal to the least end-to-end bit rate in the multicast group (i.e., $R_0 = \min_g \hat{B}_g$). Let the packet-level FEC rate be R_p and the byte-level FEC rate be R_b. Given (n_p, k_p) and $(n_b, k_b), R_p$ and R_b are clearly given by

$$R_p = R_0 \left(\frac{n_p - k_p}{n_p} \right) \tag{3}$$

and

$$R_b = R_0 \left(\frac{k_p}{n_p} \right) \left(\frac{n_b - k_b}{n_b} \right) . \tag{4}$$

The video source rate R_s, defined as the data rate excluding all the FEC, is then given by

$$\begin{aligned} R_s &= R_0 - R_p - R_b \\ &= R_0 \left(\frac{k_b}{n_b} \right) \left(\frac{k_p}{n_p} \right) . \end{aligned} \tag{5}$$

The nomenclature used in this paper is shown in Table I.

2) Transcoding Gateway: A transcoding gateway transcodes video packets from packet-level FEC to byte-level FEC before forwarding the packets to the wireless clients. We show in Fig. 3(a) the block diagram of a system with such gateway. The gateway first recovers any dropped packets by the packet-level FEC and then pads the video packets with byte-level FEC parity. Note that the wired clients need to perform packet-level FEC operations only, and, in contrast with the nontranscoding gateway, byte-level FEC encoding is done at the gateway rather than the server. We consider a simple transcoding gateway which does not do any packet fragmentation or reassembly. We see that a transcoding gateway achieves slightly lower bit-rate requirement than the nontranscoding one

[2] We define a packet as "dropped" if the packet is in error during its transmission over the network. A dropped packet is (permanently) "lost" if it cannot be recovered after packet-level FEC.

TABLE I
NOTATIONS USED IN THIS PAPER

N	:	Total number of clients in the system
L	:	Number of video layers
G	:	Size of multicast group (number of clients)
R_i	:	Transmission rate of the ith layer (bits/s)
$\hat{B}_g$	:	Estimated end-to-end available bitrate for client g (bits/s)
$\hat{P}_{l,g}$	:	Estimated packet drop rate in the wired networks for client g
$\overline{P}_l$	:	Average packet drop rate in the wired network
$\hat{e}_{b,g}$	:	Estimated bit error rate over the wireless hop for client g
$\overline{e}_b$	:	Average bit error rate over the wireless hop
$e_{s,g}$	:	Symbol error rate in the wireless hop for client g
n_b, n_p	:	Packet size of the byte-level FEC (bytes) and block size of the packet-level FEC (packets), respectively
k_b, k_p	:	Data bytes in a byte-level FEC packet and number of packets in a packet-level FEC block, respectively
ε_o	:	Constraint/Requirement on end-to-end packet loss rate (after error correction)
ϵ_g	:	End-to-end packet loss rate (after error correction) for client g for non-transcoding gateway
η_g	:	End-to-end packet loss rate (after error correction) for client g for transcoding gateway
Γ_g	:	Goodput for client g (bits/s)
Γ	:	Total goodput $\triangleq \sum \Gamma_g$ (bits/s)

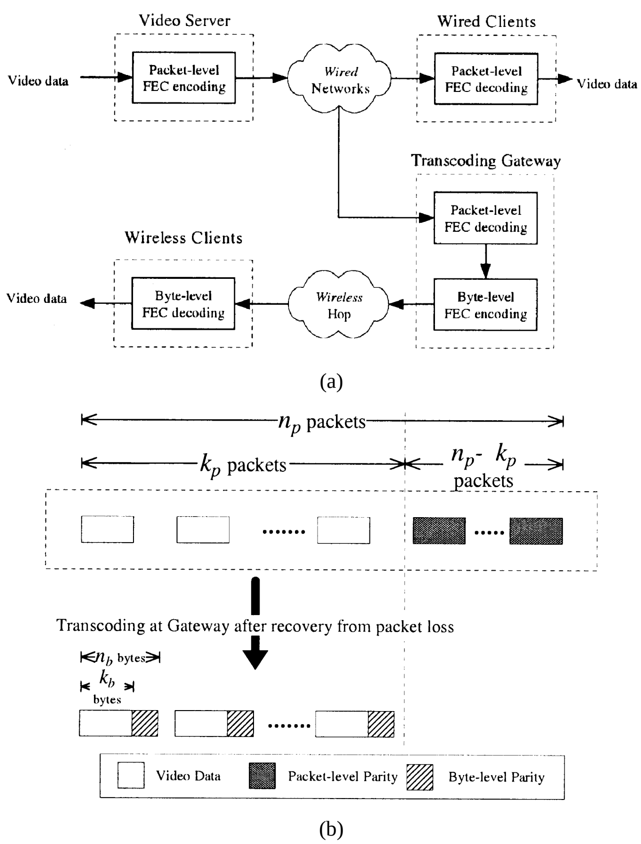


Fig. 3. Packet-level and byte-level FEC scheme for transcoding gateway. (a) Data flow of the scheme. (b) FEC generation of the scheme.

(or equivalently, higher video quality given a bit-rate constraint) by trading off some system complexity.

We show the detail of the encoding process in Fig. 3(b). The gateway first recovers the k_p data packets (out of the n_p FEC block), each of k_b bytes, and then transcodes the packets to n_b bytes by padding them with some byte-level FEC. Given the base-layer transmission rate of R_0 bits/s, the packet-level and byte-level FEC rates are clearly given by the same expressions as (3) and (4), respectively. The source rate is, however, given by

$$R_s = R_0 \left(\frac{k_p}{n_p} \right). \tag{6}$$

B. Quality Optimization

Here, we analyze the systems with nontranscoding and transcoding gateways and consider how the video quality can be maximized over all the clients in the system. As mentioned before, we consider maximizing the sum of PSNR over all the clients. For a low loss rate (i.e., $<10\%$), this is equivalent to maximizing the aggregate goodput Γ (bits/s), defined as the useful data bits delivered per second over all clients *after* error correction. Furthermore, let Γ_g be the goodput of the gth client.

Therefore, we study the following byte-level and packet-level FEC allocation problem: Given n_b and n_p, find the optimal k_p, and k_b in order to maximize

$$\Gamma = \sum_{g=1}^{G} \Gamma_g \tag{7}$$

such that the end-to-end packet-loss rate (after error correction) is no more than a certain value ε_o (say, 0.01–0.03) over all clients. Here, we consider the sum of the individual goodput, i.e., all the clients in the system have the same priority or importance. If this is not the case, we need to assign some weight to each Γ_g (and thereof each PSNR). This extension is straightforward and will not be pursued further here.

1) Optimization for Nontranscoding Gateway: Let us consider a particular client g (and hence the subscript "g" in some of our equations) and obtain its goodput given $\hat{P}_{l,g}$ and $\hat{e}_{b,g}$. In the wireless hop, a symbol is considered in "error" if any of the m bits in the symbol are transmitted in error. Clearly, given the bit-error rate $\hat{e}_{b,g}$ in the wireless channel, the symbol error rate is

$$e_{s,g} = 1 - (1 - \hat{e}_{b,g})^m. \tag{8}$$

Since the $\mathrm{RS}(n_b, k_b)$ code corrects up to t_b symbol errors, the probability that a random packet cannot be recovered by byte-level FEC is given by

$$\alpha_g = \sum_{j=t_b+1}^{n_b} \binom{n_b}{j} e_{s,g}{}^j (1 - e_{s,g})^{n_b - j}. \tag{9}$$

Note that for the wired clients, $\alpha_g = 0$ as $\hat{e}_{b,g} = 0$ by definition.

A packet is "dropped" if it is dropped in the wired networks (with rate $\hat{P}_{l,g}$), or if it is in unrecoverable error (with probability α_g) over the wireless hop. Since the two events are independent, the end-to-end packet drop rate from the source to the client is given by

$$\beta_g = 1 - (1 - \hat{P}_{l,g})(1 - \alpha_g). \tag{10}$$

Note that the dropped packets may be recovered by the packet-level FEC [see Fig. 2(a)]. Since up to $t_p = n_p - k_p$ dropped packets in the same block can be recovered by packet-level FECs. By considering the number of packet drops in a FEC block, the probability that a random packet is permanently "lost" (i.e., the end-to-end packet-loss rate after error correction) is given by (see [6] and [7])

$$\epsilon_g = \sum_{k=t_p+1}^{n_p} \frac{k}{n_p} \binom{n_p}{k} \beta_g{}^k (1 - \beta_g)^{n_p - k}. \tag{11}$$

The goodput of the client g is hence given by

$$\Gamma_g = R_0 \left(\frac{k_b}{n_b}\right) \left(\frac{k_p}{n_p}\right) \times (1 - \epsilon_g). \tag{12}$$

The allocation problem is a two-dimensional search on k_p and k_b, which is of complexity $O(Gn_pn_b)$ and is not efficient. Validated by extensive runs, we found that packet-level FEC optimization can be done independently with that of byte-level FEC without affecting the results much (less than 1%). Therefore, we can greatly reduce the complexity to $O(G(n_p + n_b))$ by means of the following two-step procedures.

- *Packet-level FEC optimization*— First, we compute k_p^* so that the residual loss rate over the wired network is no more than ε_o by the following. We ignore the wireless links by setting $\alpha_g = 0$ for all clients. Let $P_l = \max_g \hat{P}_{l,g}$ be the maximum packet drop rate for all the clients. If $P_l \le \varepsilon_o$, STOP and proceed to the next step (The packet drop rate is so low that $k_p^* = n_p$). Otherwise, for all the clients with $\hat{P}_{l,g} > \varepsilon_o$, search for the largest $k_p < n_p$ (i.e., for minimum parities) such that the end-to-end residual loss rate ϵ_g [in accordance with (11)] of all these clients are no more than ε_o. This is the k_p^* required.
- *Byte-Level FEC optimization*— Now, we reintroduce α_g for all wireless clients. Given k_p^*, find the largest $k_b < n_b$, such that ϵ_g [in accordance with (11)] for all the wireless clients are no more than ε_o. This is the k_b^* required.

TABLE II
CLIENTS' PROFILE USED FOR TRANSCODING AND NON-TRANSCODING GATEWAYS

	$\hat{P}_{l,g}$ (%)	$\hat{e}_{b,g}$ (10^{-4})
client 1	2.0572	0.9993
client 2	1.7179	0.5460
client 3	2.4790	0.8594
client 4	1.8248	1.3363
client 5	2.7698	1.0134
client 6	1.3341	0.0
client 7	2.1079	0.0
client 8	2.7578	0.0
client 9	1.1049	0.0
client 10	2.4529	0.0

2) Optimization for Transcoding Gateway: Consider a client g. The probability that a random packet is permanently lost over the wired network is clearly given by

$$\gamma_g = \sum_{k=t_p+1}^{n_p} \frac{k}{n_p} \binom{n_p}{k} \hat{P}_{l,g}^k (1 - \hat{P}_{l,g})^{n_p - k}. \tag{13}$$

If it is a wireless client, the packets corrected after packet-level FEC are transmitted over the wireless hop. The probability that these packets cannot be recovered due to wireless error has already been obtained as α_g in (9) (again, for the wired client, $\alpha_g = 0$). Therefore, the end-to-end packet-loss rate after error correction is given by (by the independence of error rates in the wired and wireless networks)

$$\eta_g = 1 - (1 - \gamma_g)(1 - \alpha_g) \tag{14}$$

and hence, the goodput of the client is

$$\Gamma_g = R_0 \left(\frac{k_p}{n_p}\right) \times (1 - \eta_g). \tag{15}$$

As in the nontranscoding case, we again observe that the packet-level FEC can be done independently of the byte-level FEC in this case. The optimization procedure is, hence, the same as that of the nontranscoding case, except that ϵ_g is replaced by η_g in (14).

C. Illustrative Numerical Examples and Results

In this section, we compare the performance of transcoding and nontranscoding gateways. We consider a baseline system of $G = 10$ clients, with half of them being wireless clients. We show in Table II $\hat{P}_{l,g}$ and $\hat{e}_{b,g}$ of each client, which are generated by assuming that they are uniformly distributed with mean $\bar{P}_l = 2\%$ and $\bar{e}_b = 10^{-4}$, respectively. Note that clients 1–5 are wireless users, while the remaining are wired. The other baseline parameters are $\varepsilon_o = 1\%, n_b = 255, n_p = 40$, and $R_0 = 100$ kbits/s. Optimal FEC allocation will first be performed given these parameters. Then, in our sensitivity analysis, we vary the other parameters, one at a time.

In Fig. 4, we show k_p^* versus ε_o for the transcoding and nontranscoding cases. Clearly, both cases have the same optimal k_p^* (due to the same optimization step in k_p^*). k_p^* increases with ε_o in

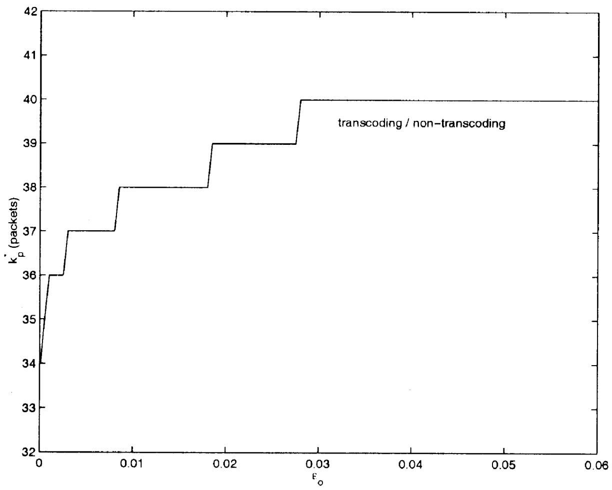


Fig. 4. k_p^* versus ε_o given k_b^* for transcoding and nontranscoding gateways ($\bar{e}_b = 10^{-4}, \bar{P}_l = 2\%, n_b = 255$ bytes, $n_p = 40$ packets).

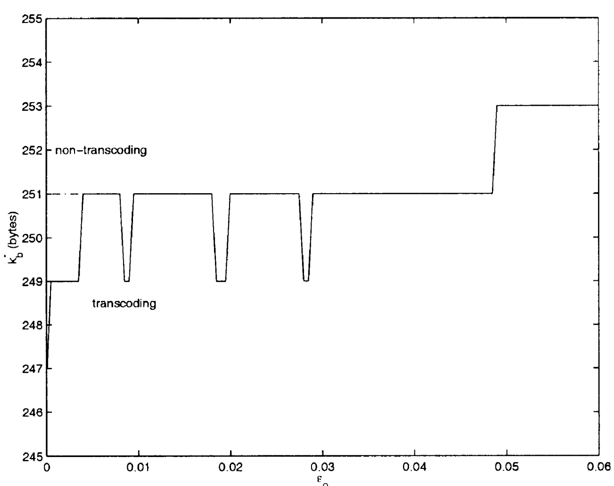


Fig. 5. k_b^* versus ε_o given k_p^* for transcoding and nontranscoding gateways ($\bar{e}_b = 10^{-4}, \bar{P}_l = 2\%, n_b = 255$ bytes, $n_p = 40$ packets).

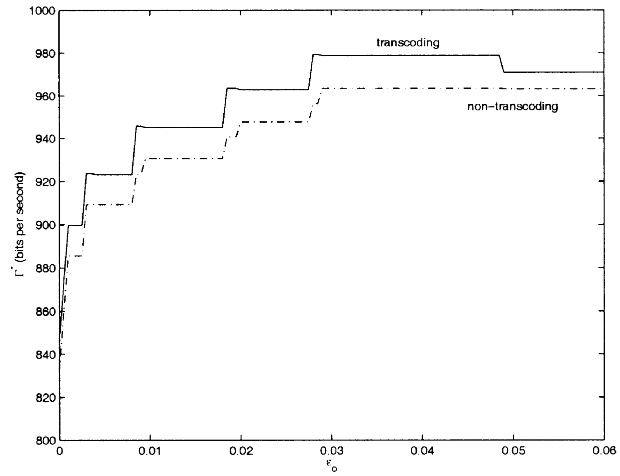


Fig. 6. Γ^* versus ε_o for transcoding and nontranscoding gateways ($\bar{e}_b = 10^{-4}, \bar{P}_l = 2\%, n_b = 255$ bytes, $n_p = 40$ packets).

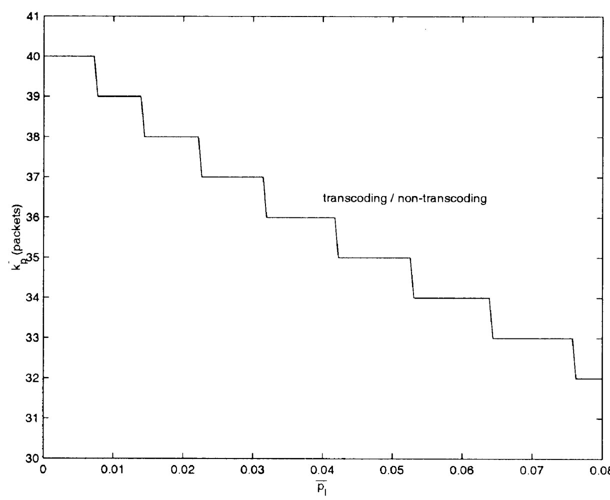


Fig. 7. k_p^* versus $\bar{P}_l$ for transcoding and nontranscoding gateways ($\bar{e}_b = 10^{-4}, \varepsilon_o = 1\%, n_b = 255$ bytes, $n_p = 40$ packets).

a stepwise manner (due to the constraint on integral value). Note that k_p^* is already very close to n_p, indicating that little packet-level FEC is necessary to achieve a low end-to-end packet loss. The packet-level FEC is so effective that even though most of the $\hat{P}_{l,g}$'s are greater than 2%, only a few parity packets (two, in this case) are needed to bring ε_o to as low as 1%. No parity is necessary when $\varepsilon_o \geq \max_g \hat{P}_{l,g}$ (as all client has $\hat{P}_{l,g} < \varepsilon_o$).

Next, in Fig. 5, we show the corresponding k_b^* versus ε_o for transcoding and nontranscoding gateways. Both cases share almost the same k_b^*. As compared with k_p^*, k_b^* is quite insensitive to ε_o; it increases relatively slowly. Therefore, as ε_o changes, k_p is a more important parameter to adjust. Note that for $\hat{e}_{b,g} = 10^{-4}$, a random packet without any byte-level FEC is in error occurs with probability of $1-(1-\hat{e}_{b,g})^{mn_b} = 0.18$. Even with this packet-loss rate, only a few parity bytes (about 4–6 in our plot) are enough to bring this error rate down to a low level given by ε_o. This again indicates the efficiency of byte-level FEC. The "dips" in the figure corresponds to the "rises" in Fig. 4. This is because once k_p^* is increased, the packet-level error-correction capability decreases and hence a lower k_b^* (and thereof a stronger byte-level correction capability) is needed to compensate. As ε_o increases, the k_b^* jumps back up as the system can tolerate more end-to-end packet loss.

We show in Fig. 6 the corresponding optimal goodput Γ^* according to (12) (i.e., with k_p^* and k_b^*) versus ε_o for the transcoding and nontranscoding cases. Though the goodput for the transcoding case is higher, there is no much difference between them (only about a 2% difference here). This is expected because, from (15) and (12), the ratio of the nontranscoding and transcoding goodputs for client g is given by $(k_b^*(\text{nontranscoding})/n_b) \times (1-\epsilon_g)/(1-\eta_g) \approx k_b^*/n_b$. From Fig. 5, we have already seen that k_b^* for the nontranscoding gateway is very close to n_b, and hence the difference is small. As ε_o increases, Γ^* in general first increases and then decreases (the decrease is shown for the transcoding case). This is due to the following. Γ is affected by two factors: 1) the end-to-end packet-loss rate ϵ_g and η_g (Γ decreases with them) and 2) k_p and k_b (Γ increases with them). From Fig. 5, we see that when ε_o is small, the effect of k_p^* and k_b^* dominate, while when ε_o is higher, the error rate dominates and Γ^* decreases. For the cases of interest (i.e., $\varepsilon_o \leq 5\%$), Γ^* increases with ε_o.

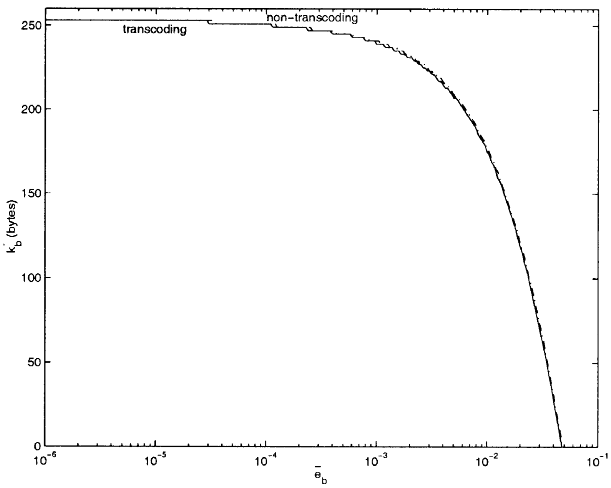


Fig. 8. k_b^* versus $\bar{e}_b$ for transcoding and nontranscoding gateways ($\bar{P}_l = 2\%, \varepsilon_o = 1\%, n_b = 255$ bytes, $n_p = 40$ packets).

In Fig. 7, we show how k_p^* varies with $\bar{P}_l$. Clearly, k_p^* decreases with $\bar{P}_l$ as more error protection is necessary. The packet-level FEC is quite effective, as only a few parity packets (given by $n_p - k_p^*$) are able to bring a high $\bar{P}_l$ (say, 7%–8%) to a low ε_o value (1%).

In Fig. 8, we show k_b^* versus $\bar{e}_b$ for transcoding and nontranscoding cases. Clearly, there is not much difference between these cases. In general, k_b^* decreases with $\bar{e}_b$ because more parity bytes are needed. As $\bar{e}_b$ increases, k_b^* remains quite flat at the beginning and then sharply decreases. This indicates that when the bit error is high, many more parity bytes are needed to achieve a certain error rate after FEC. From the figure, we also see that when $\bar{e}_b$ is greater than a certain value (about 5% in this case), the bit-error rate is too high, and thus the byte-level FEC is no longer effective in bringing errors in the wireless hop down to ε_o.

III. Joint Bandwidth and FEC Optimation for the Enhancement Layers

While the quality optimization of the base-layer focuses mainly on FEC allocation, the optimization of the enhancement layers has two dimensions: both FEC and bandwidth allocations. This is the subject of discussion in this section.

In our system, the video is encoded into a total of L enhancement layers (i.e., the video stream has $L+1$ layers including the base layer). Note that in layered encoding, a higher layer can be decoded only if all the lower layers are received. Let the bandwidth of enhancement layer l be R_l bits/s $(1 \le l \le L)$, where the higher the index l is, the higher is the enhancement layer (i.e., a client cannot decode the layer i without receiving all of its preceding $i-1$ layers). Each of the layers is carried by a multicast group. The clients in the network join as many layers as possible; however, none of them joins more layers than its estimated end-to-end bandwidth can accommodate.

We assume that the video quality is enhanced (in terms of PSNR) due to the enhancement layers is linearly dependent on the aggregate goodput of the layers received, i.e., the quality is independent on the number of layers and how the goodput is

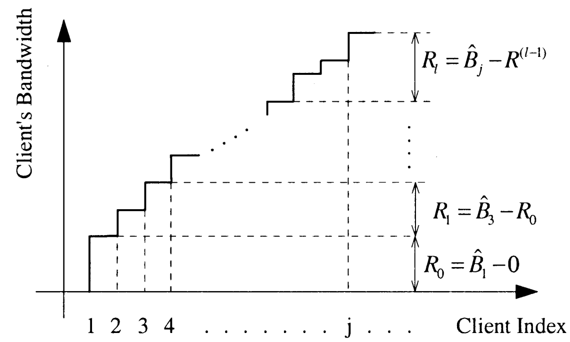


Fig. 9. Mapping of $\hat{B}_g$ to R_l.

partitioned among them. Clearly, if the layer bit rates are set too low, the sum of goodput is low. On the other hand, if the layer bit rates are set too high, many clients can only join a few layers and hence the aggregate goodput is also low. Therefore, there is an optimal allocation such that the aggregate goodput is maximized. The optimization of the enhancement layers then becomes: What are the bandwidth and FEC of each of the enhancement layers in order to maximize the sum of video quality enhanced in terms of the goodput of each client?

In Section III-A, we first formulate the allocation problem and present a dynamic program to solve it. In order to reduce the time complexity of the optimization, an approximation method on the bandwidth partition is discussed in Section III-B. We show some illustrative results in Section III-C.

A. Dynamic Program Optimization

To formulate the optimization problem, we start by ordering the end-to-end bandwidth in increasing order, so that $\hat{B}_i \ge \hat{B}_j$ for $i > j$. We plot $\hat{B}_g$ versus the client index g in Fig. 9. Clearly, $R_0 = \hat{B}_1$ is the base-layer bit rate. R_l's, $1 \le l \le L$, are added on top of each other, one by one. Let

$$R^{(l)} = \sum_{i=1}^{l} R_i \tag{16}$$

be the cumulative transmission rate of the enhancement layers up to and including layer l (by definition, $R^{(0)} = R_0 = \hat{B}_1$); therefore, $R_l = R^{(l)} - R^{(l-1)}$. Then, all clients with $\hat{B}_g \ge R^{(l)}$ would join enhancement layer l.

Obviously, in order to maximize the goodput, we only need to consider

$$R^{(l)} \in \{\hat{B}_2, \ldots, \hat{B}_G\}, \quad l \ge 1. \tag{17}$$

In other words, $R_l = \hat{B}_j - R^{(l-1)}$, for some $R^{(l-1)} < \hat{B}_j \le \hat{B}_G$ ($R^{(i)} \ne R^{(j)}$ for $i \ne j$). For example, in the figure, the first enhancement layer is encoded with transmission rate $R_1 = \hat{B}_3 - R^{(0)} = \hat{B}_3 - \hat{B}_1$.

Obviously, there is no issue in bandwidth assignment when $L \ge G - 1$ (i.e., the number of enhancement layers is larger than the user pool); the server simply encodes the lowest $G-1$ enhancement layers with $R_l = \hat{B}_{l+1} - \hat{B}_l$. Hence, we will only focus on the $L < G - 1$ case in the following.

Let $\mathcal{S}_l$ be the set of clients who join the lth enhancement layer (and hence all layers below). Clearly, the sum of the goodput for all the clients joining enhancement layer l is given by

$$\Gamma^{(l)} = \sum_{g \in \mathcal{S}_l} \Gamma_g(R_l) \tag{18}$$

where $\Gamma_g(R_l)$ is the goodput of client g joining layer l. From the derivation in the previous section, we have

$$\Gamma_g(R_l) = \begin{cases} R_l\left(\frac{k^*_{p,l}}{n_p}\right)(1-\eta_{g,l}), & \text{for transcoding} \\ R_l\left(\frac{k^*_{b,l}}{n_b}\right)\left(\frac{k^*_{p,l}}{n_p}\right)(1-\epsilon_{g,l}), & \text{otherwise} \end{cases} \tag{19}$$

where $k^*_{p,l}$ and $k^*_{b,l}$ are the packet-level and byte-level FEC for layer l, respectively (depending on the loss characteristics of those clients joining the multicast group), while $\epsilon_{k,l}$ and $\eta_{k,l}$ are given according to (11) and (14), respectively.

Therefore, the total goodput of the system due to the L enhancement layers $\Gamma_{[L]}$ is given by

$$\Gamma_{[L]} = \sum_{l=1}^{L} \Gamma^{(l)}. \tag{20}$$

We denote $\Gamma^*_{[l]}(x)$ as the maximum goodput given that there are l enhancement layers and the maximum end-to-end bandwidth is x, where $x \in \{\hat{B}_2, \ldots, \hat{B}_G\}$. Clearly, we are interested in $\Gamma^*_{[L]}(\hat{B}_G)$ (which is equal to $\Gamma_{[L]}$ in (20)). By noting that the goodput for l enhancement layers is the sum of the goodput for the lth layer and $(l-1)$ layers below, $\Gamma^*_{[L]}(\hat{B}_G)$ can be computed recursively with the following dynamic program by solving for $R^{(i)}$'s (and hence R_i's)

$$\begin{aligned} \Gamma^*_{[L]}(\hat{B}_G) &= \max_{\substack{\hat{B}_{L+1} \le x < \hat{B}_G; \\ R^{(L-1)} < R^{(L)} \le \hat{B}_G}} \left(\Gamma^{(L)} + \Gamma^*_{[L-1]}(x)\right) \\ \Gamma^*_{[L-1]}(y) &= \max_{\substack{\hat{B}_{L} \le x < y; \\ R^{(L-2)} < R^{(L-1)} \le y}} \left(\Gamma^{(L-1)} + \Gamma^*_{[L-2]}(x)\right) \\ &\vdots \\ \Gamma^*_{[L-i]}(y) &= \max_{\substack{\hat{B}_{L-i+1} \le x < y; \\ R^{(L-i-1)} < R^{(L-i)} \le y}} \left(\Gamma^{(L-i)} + \Gamma^*_{[L-i-1]}(x)\right) \\ &\vdots \\ \Gamma^*_{[1]}(y) &= \max_{\hat{B}_1 < R^{(1)} \le y} \Gamma^{(1)}. \end{aligned} \tag{21}$$

B. Efficient Approximation on Layer Bandwidths

In each of the recursive steps in the dynamic program above, there are $O(G)$ possibilities of $R^{(l)}$; therefore, the search space of the above bit rate and FEC allocation problem is $O(GL(n_p+n_b))$. Clearly, the complexity becomes excessive for a large number of clients. In this section, we present an approximation of the allocation problem when the user pool is large and the error rate is negligible. The approximation can be done quickly and can be used for initial search. In this way, the search space is greatly reduced to $O(L(n_p+n_b))$. We show that our approximation matches well with the actual computation of the dynamic program with a finite user pool and reasonable error rates.

Consider a large number of clients (i.e., $G \to \infty$) with their end-to-end bandwidth distributed according to some probability density function (pdf) $f(x)$ which ranges from $\hat{B}_{\min}$ to $\hat{B}_{\max}$. Note that a total of $G\int_{R^{(l)}}^{R^{(l+1)}} f(x)\,dx$ clients are with enhancement bit rate of $R^{(l)} - \hat{B}_{\min}$ (since the bit rate of the base layer is $\hat{B}_{\min}$), the aggregate goodput of all the clients for the enhancement layers is then given by

$$\frac{\Gamma_{[L]}}{G} \approx \sum_{l=1}^{L-1}(R^{(l)} - \hat{B}_{\min})\int_{R^{(l)}}^{R^{(l+1)}} f(x)\,dx + (R^{(L)} - \hat{B}_{min})\int_{R^{(L)}}^{\hat{B}_{\max}} f(x)\,dx. \tag{22}$$

The corresponding $R^{(l)*}$'s $(1 \le l \le L)$ that yield the maximum goodput can be obtained by setting $\partial\Gamma_{[L]}/\partial R^{(l)} = 0, \forall l$.

Note that our approximation works for all kinds of bandwidth distribution (e.g., Gaussian, uniform, beta, …, etc) as long as the corresponding pdf $f(x)$ is known. The shape of $f(x)$ may depend on some parameter, say θ (e.g., for Gaussian distribution, θ may be the mean and/or variance, while for uniform distribution, θ may be the mean). Therefore, $R^{(l)*}$ obtained is a function of θ. In reality, θ, and hence the actual $f(x)$ may be estimated by curve-fitting (some of) the feedbacks from the clients using regression.

As an example, let us consider that the end-to-end bandwidth of the clients is uniformly distributed between $\hat{B}_{\min}$ and $\hat{B}_{\max}$ (with mean $(\hat{B}_{\max} - \hat{B}_{\min})/2$), i.e., $f(x) = 1/(\hat{B}_{\max} - \hat{B}_{\min})$. Thus

$$\frac{\Gamma_{[L]}}{G} = \sum_{l=1}^{L-1}(R^{(l)} - \hat{B}_{\min})\left(\frac{R^{(l+1)} - R^{(l)}}{\hat{B}_{\max} - \hat{B}_{\min}}\right) + (R^{(L)} - \hat{B}_{\min})\left(\frac{\hat{B}_{\max} - R^{(L)}}{\hat{B}_{\max} - \hat{B}_{\min}}\right) \tag{23}$$

from which we get

$$R^{(l)*} = \hat{B}_{\min} + l \times \frac{\hat{B}_{\max} - \hat{B}_{\min}}{L+1} \tag{24}$$

and hence

$$R^*_l = \frac{\hat{B}_{\max} - \hat{B}_{\min}}{L+1} \tag{25}$$

which is the approximated layered bit rate obtained.

C. Illustrative Numerical Examples and Results

In this subsection, we show the results of the joint bandwidth and FEC optimization. All enhancement layers are transmitted with end-to-end loss requirement of $\varepsilon_o = 2\%$, while that of the base-layer is $\varepsilon_o = 1\%$. We use the same baseline system as considered in the previous section, with the video stream consisting of four enhancement layers. As there is not much difference between the transcoding and nontranscoding scheme, we will use the latter in this subsection. The client end-to-end bandwidths are uniformly distributed between $\hat{B}_{\min} = 100$ kb/s and $\hat{B}_{\max}$, and therefore the standard deviation of their bandwidth is given by $(\hat{B}_{\max} - \hat{B}_{\min})/\sqrt{12}$. The $\hat{e}_{b,g}$ and $\hat{P}_{l,g}$ of each client are independently distributed with mean $\bar{e}_b = 10^{-4}$ and $\bar{P}_l = 5\%$,

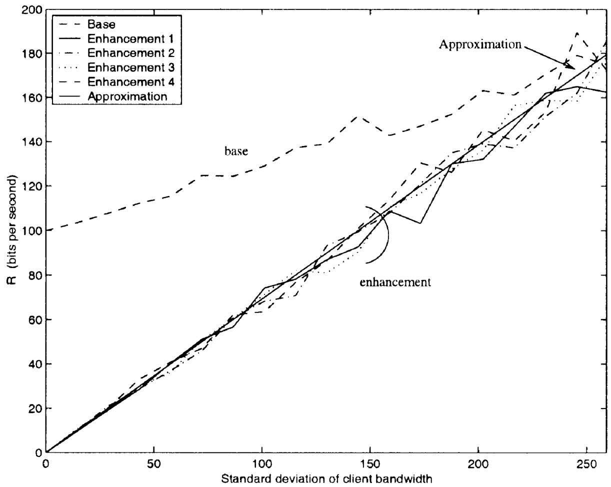


Fig. 10. Transmission rate of enhancement layers R_l versus the standard deviation of the end-to-end bandwidth of the clients ($G = 10$ with five wireless clients, $\bar{e}_b = 10^{-4}, \bar{P}_l = 5\%, L = 4, n_b = 255$ bytes, and $n_p = 40$ packets).

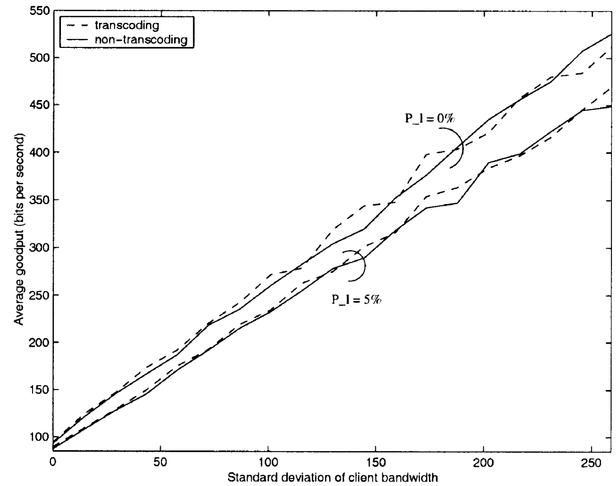


Fig. 12. Average goodput $\Gamma^*_{[L]}/G$ versus the standard deviation of client bandwidth for transcoding and nontranscoding gateways ($G = 10$ with 5 wireless clients, $\bar{e}_b = 10^{-4}, L = 4, n_b = 255$ bytes, and $n_p = 40$ packets).

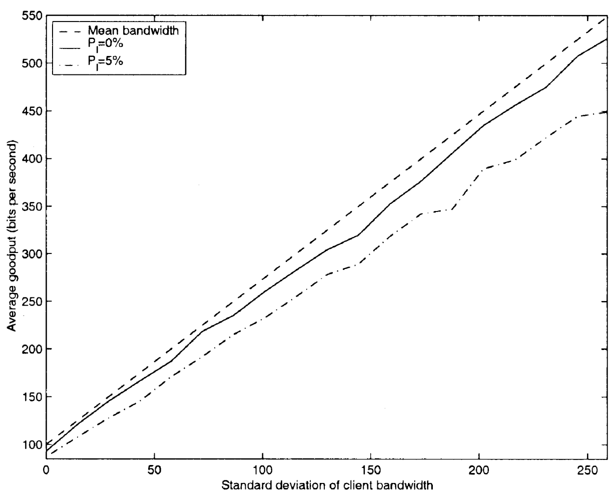


Fig. 11. Average goodput $\Gamma^*_{[L]}/G$ [see (20)] versus the standard deviation of client bandwidth ($G = 10$ with five wireless clients, $\bar{e}_b = 10^{-4}, L = 4, n_b = 255$ bytes, and $n_p = 40$ packets).

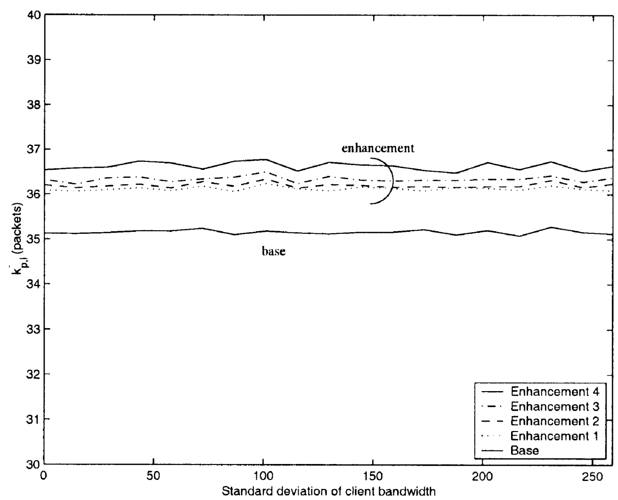


Fig. 13. $k^*_{p,l}$ versus the standard deviation of client bandwidth ($G = 10$ with five wireless clients, $\bar{P}_l = 5\%, \bar{e}_b = 10^{-4}, L = 4, n_b = 255$ bytes, and $n_p = 40$ packets).

respectively. The results are averaged over a number of independent runs (typically 50 runs).

We show in Fig. 10 the optimal bandwidth allocated to layer l, R^*_l, versus the standard deviation of client bandwidth. For the enhancement layers, we show the results for both actual solution of the dynamic program and our approximation from (25). Clearly, our approximation agrees well with the direct computation of the dynamic program. The layered bandwidth increases with the standard deviation because the range of client bandwidth increases. The result shows that our approximation is good even with as few as $G = 10$ clients, and with a packet drop rate as high as $\bar{P}_l = 5\%$. The base-layer bandwidth also increases with the standard deviation, since clients are more likely to have higher end-to-end bandwidth with higher deviation (note that $\hat{B}_{\min}$ does not change). We have also run the case for $\bar{P}_l = 0\%$, but there is not much difference in allocation (not shown here). This indicates that the allocation does not depend sensitively on $\bar{P}_l$, and that bandwidth heterogeneity, rather than error rate and group size, dominates the choice for the layer bandwidths.

In Fig. 11, we show the average optimal goodput $\Gamma^*_{[L]}/G$ versus the standard deviation of client bandwidth, given $\bar{P}_l$. The average goodput increases with the standard deviation. When $\bar{P}_l$ increases, the goodput decreases (due to more FEC). We also show in the figure the ideal case of no packet and bit errors, corresponding to a mean bandwidth over all the clients of $(\hat{B}_{\min}+\hat{B}_{\max})/2$. We see that our allocation is reasonably close to the ideal case (the goodput with $\bar{P}_l = 0\%$ is about 15% lower than the ideal case), indicating the efficiency of our allocation.

We compare the average optimal goodput versus the standard deviation of client bandwidth for both transcoding and nontranscoding gateway in Fig. 12, given $\bar{P}_l$. Clearly, the difference between the goodput by the two schemes is negligible. The figure indicates that the variation of client bandwidth is a more important factor in determining goodput than the choice of the gateway.

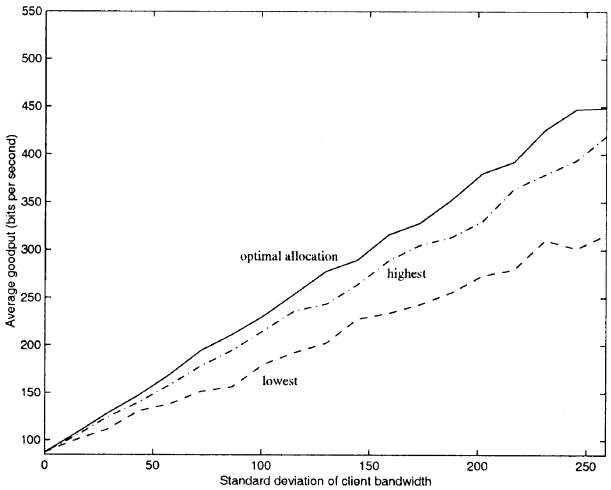


Fig. 14. Average goodput $\Gamma^*_{[L]}/G$ versus the standard deviation of client bandwidth for different allocation strategies ($G = 10$ with five wireless clients, $\bar{e}_b = 10^{-4}$, $\bar{P}_l = 5\%$, $L = 4$, $n_b = 255$ bytes, and $n_p = 40$ packets).

We show in Fig. 13 the corresponding $k^*_{p,l}$ for each layer, which does not depend sensitively on the standard deviation of the client bandwidth. We see that the base layer requires more parity packets (by about one packet) than the enhancement layers, mainly due to its higher ε_o. For the enhancement layers, the higher ones require slightly fewer parity packets than the lower ones. This is because $k^*_{p,l}$ depends on the largest loss rate among the clients joining the layer. Since there are fewer clients joining a higher layer, it is more likely that their maximum loss rate is lower than those of the lower layers. We see that our allocation is effective in terms of FEC parties (only about 10% overhead). The plot for the corresponding $k^*_{b,l}$ shows a similar trend and is not repeated here.

To show the effectiveness of our joint bandwidth and FEC allocation scheme, we compare the average goodput achieved by our scheme with two simple "naive" bandwidth allocation schemes. These schemes allocate layer bandwidths according to either the highest L client bandwidths (i.e., $R^{(l)} = \hat{B}_{G-L+l}$) or the lowest L client bandwidths (i.e., $R^{(l)} = \hat{B}_{l+1}$), respectively. We show the average goodputs of all the schemes in Fig. 14 with $\bar{P}_l = 5\%$. Clearly, our optimal scheme achieves higher goodput than both "naive" schemes. The difference is especially high when the variance of user bandwidth is high (by about 10% for the allocation to the highest set of bandwidths and by about 30% for the allocation to the lowest set of bandwidths in this example). This shows the strength of our allocation scheme, especially in a multicast group with diverse end-to-end bandwidth.

IV. Conclusions

In this paper, we have studied a layered video multicast system over wired and wireless networks with receiver feedback. The main challenge is to optimize the overall video quality by means of layer bandwidth and FEC allocations for the set of clients given their heterogenous bandwidth and error characteristics, subject to a certain overall loss rate requirement. Furthermore, since there may be a transcoding gateway (which transcodes from packet-level FEC to byte-level FEC) between the wireless clients and the wired network, we have studied the value of such a gateway. We have analyzed the system and proposed an efficient allocation policy.

In order to serve all the clients, the bandwidth of the base layer should be equal to the minimum bandwidth of the clients. The issue of the base-layer transmission is hence how to allocate packet-level and byte-level FEC so as to maximize video quality (in terms of goodput). Instead of a two-dimensional search, we have presented an efficient algorithm for such optimal FEC allocation.

Our results show that the transcoding scheme performs only slightly better in terms of system goodput than the nontranscoding scheme (by about 2%). This is mainly due to the efficiency of FEC encoding (which occupies less than 20% of the data for the packet-level FEC, and less than 10% for byte-level FEC). This small difference may not justify the complexity of such a transcoding gateway. A gateway which transcodes data in some other ways may be more useful.

Clients may join the enhancement layers to further improve the video quality beyond the base layer. The issue is then how to allocate bandwidth and FEC across each layer so as to achieve maximum video quality. We have formulated the problem with a dynamic program, and developed a fast approximation for such allocation. The results show that our approximation agrees with the actual computation of the dynamic program, which is much more complex. Our allocation is effective in the sense that it achieves quality close to the ideal case without packet loss or bit error.

References

[1] D. Wu, Y. T. Hou, W. W. Zhu, Y.-Q. Zhang, and J. M. Peha, "Streaming video over the Internet: Approaches and directions," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 282–300, Mar. 2001.

[2] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Transporting real-time video over the Internet: Challenges and approaches," *Proc. IEEE*, vol. 88, pp. 1855–1877, Dec. 2000.

[3] B. Girod and N. Farber, "Feedback-based error control for mobile video," *Proc. IEEE*, vol. 87, no. 10, pp. 1707–1723, Oct. 1999.

[4] L. Vicisano, L. Rizzo, and J. Crowcroft, "TCP-like congestion control for layered multicast data transfer," in *Proc. INFOCOM'98*, vol. 3, San Francisco, CA, Mar./Apr. 1998, pp. 996–1003.

[5] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 983–1001, Aug. 1997.

[6] Q. Zhang, W. Zhu, and Y.-Q. Zhang, "Network-adaptive rate control with TCP-friendly protocol for multiple video objects," *Proc. IEEE Int. Conf. Multimedia and Expo*, vol. 2, pp. 1055–1058, July/Aug. 2000.

[7] ——, "Resource allocation for multimedia streaming over the Internet," *IEEE Trans. Multimedia*, vol. 3, pp. 339–355, Sept. 2001.

[8] R. Rejaie, M. Handley, and D. Estrin, "RAP: An end-to-end rate-based congestion control mechanism for realtime streams in the Internet," *Proc. IEEE INFOCOM'99*, vol. 3, pp. 1337–1345, Mar. 1999.

[9] J. G. Kim and M. M. Krunz, "Bandwidth allocation in wireless networks with guaranteed packet-loss performance," *IEEE/ACM Trans. Networking*, vol. 8, pp. 337–349, June 2000.

[10] G. Ramamurthy and D. Raychaudhuri, "Performance of packet video with combined error recovery and concealment," in *Proc. INFOCOM'95*, vol. 2, Boston, MA, Apr. 1995, pp. 753–761.

[11] C. Papadopoulos and G. Parulkar, "Retransmission-based error control for continuous media applications," in *Proc. 6th Int. Workshop Network Operating Syst. Support Digital Audio Video*, 1996, pp. 5–12.

[12] A. Albanese, J. Blomer, J. Edmonds, M. Luby, and M. Sundan, "Priority encoding transmission," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1737–1744, Nov. 1996.

[13] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1012–1032, June 2000.

[14] S. Wicker and V. Bhargava, *Reed-Solomon Codes and Their Applications*. Piscataway, NJ: IEEE Press, 1994.

[15] S. Gringeri, S. Iren, and R. Egorov, "Transmission of MPEG-4 video over the Internet," *Proc. IEEE Int. Conf. Multimedia and Expo*, vol. 3, pp. 1767–1770, July/Aug. 2000.

[16] P. Frossard and O. Versche, "Joint source/FEC rate selection for optimal MPEG-2 video delivery," *Proc. IEEE Int. Conf. Multimedia and Expo*, vol. 3, pp. 1301–1304, July/Aug. 2000.

[17] V. Parthasarathy, J. W. Modestino, and K. S. Vastola, "Reliable transmission of high-quality video over ATM networks," *IEEE Trans. Image Processing*, vol. 8, pp. 361–374, Mar. 1999.

[18] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," in *Proc. ACM SIGCOMM'98*, vol. 28, Vancouver, BC, Canada, Oct. 1998, pp. 303–314.

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

[20] O. Verscheure, P. Frossard, and M. Hamdi, "Joint impact of MPEG-2 encoding rate and ATM cell losses on video quality," *Proc. IEEE Global Telecommunications Conf.*, vol. 1, pp. 71–76, Nov. 1998.

[21] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, pp. 974–997, May 1998.

[22] J. C. Bolot, T. Turletti, and I. Wakeman, "Scalable feedback control for multicast video distribution in the Internet," in *Proc. ACM SIGCOMM'94*, vol. 24, London, U.K., Oct. 1994, pp. 58–67.

[23] R. Puri, K. Ramchandran, and A. Ortega, "Joint source channel coding with hybrid FEC/ARQ for buffer constrained video transmission," *Proc. IEEE 2nd Workshop on Multimedia Signal Processing*, pp. 567–572, Dec. 1998.

[24] S. S. Wang, H. Zheng, and J. A. Copeland, "A QoS enhanced hybrid SR-ARQ for mobile video communications," *Proc. IEEE Int. Conf. Communications*, vol. 1, pp. 526–530, June 2000.

[25] P. A. Chou, A. E. Mohr, A. Wang, and S. Mehrotra, "Error control for reciver-driven layered multicast of audio and video," *IEEE Trans. Multimedia*, vol. 3, pp. 108–122, Mar. 2001.

[26] M. Shhar and H. Radha, "Unequal packet loss resilience for fine-granular scalability video," *IEEE Trans. Multimedia*, vol. 3, pp. 381–394, Dec. 2001.

[27] H. M. Smith, M. W. Mutka, and E. Torng, "Bandwidth allocation for layered multicast video," *Proc. IEEE Int. Conf. Multimedia Computing and Systems*, vol. 1, pp. 232–237, June 1999.

[28] F. Le Leannec, J. Vieron, X. Henocq, and C. Guillemot, "Hybrid sender and receiver driven rate control in multicast layered video transmission," in *Proc. Int. Conf. Image Processing*, vol. 3, Vancouver, BC, Canada, Sept. 2000, pp. 532–535.

[29] M. Gallant and F. Kossentini, "Rate-distortion optimized layered coding with unequal error protection for robust Internet video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 357–372, Mar. 2001.

[30] W.-T. Tan and A. Zakhor, "Video multicast using layered FEC and scalable compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 373–386, Mar. 2001.

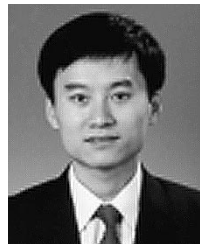
[31] Q. Guo, Q. Zhang, W. Zhu, and Y.-Q. Zhang, "A sender-adaptive and receiver-driven layered multicast scheme for vidoe over Internet," *Proc. 2001 IEEE Int. Symp. Circuits and Systems*, vol. 5, pp. 141–144, May 2001.

[32] T.-W. A. Lee, S.-H. G. Chan, Q. Zhang, W. Zhu, and Y.-Q. Zhang, "Optimal allocation of packet-level and byte-level FEC in video multicasting over wired and wireless networks," *Proc. IEEE Global Telecommunications Conf.*, vol. 3, pp. 1994–1998, Nov. 2001.

[33] H. S. Wang and P.-C. Chang, "On verifying the first-order Markovian assumption for a Rayleigh fading channel model," *IEEE Trans. Veh. Technol.*, vol. 45, pp. 353–357, May 1996.

T.-W. Angus Lee received the B.Eng. degree in computer engineering and the M.Phil. degree in computer science and from the Hong Kong University of Science and Technology, Hong Kong, in 1999 and 2001, respectively.

He was a research intern at Microsoft Research Asia, Beijing, China. His research interests include video delivery in wireless and wired networks, multicast protocol, layered video coding, and error-correction coding.



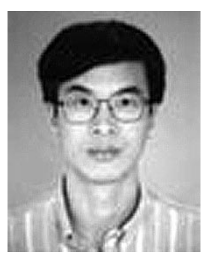
S.-H. Gary Chan received the B.S.E. degree (with highest honors) in electrical engineering from Princeton University, Princeton, NJ, in 1993, and the Ph.D. degree in electrical engineering (with a minor in business administration) from Stanford University, Stanford, CA, in 1999.

He is currently an Assistant Professor with the Department of Computer Science, Hong Kong University of Science and Technology, Kowloon, Hong Kong, and an Adjunct Researcher with Microsoft Research Asia, Beijing, China. He was a Visiting Assistant Professor in networking with the Department of Computer Sscience, University of California at Davis, from September 1998 to June 1999. During 1992–1993, he was a Research Intern at the NEC Research Institute, Princeton, NJ. His research interests include multimedia networking, peer-to-peer networks, high-speed and wireless communications networks, and Internet technologies and protocols.

Dr. Chan was a William and Leila Fellow at Stanford University during 1993–1994. In 1993, he also received the Charles Ira Young Memorial Tablet and Medal and the POEM Newport Award of Excellence from Princeton University. He is a member of Tau Beta Pi, Sigma Xi, and Phi Beta Kappa.

Qian Zhang (M'00) received the B.S., M.S., and Ph.D. degrees from Wuhan University, China, in 1994, 1996, and 1999, respectively, all in computer science.

She joined Microsoft Research, Asia, Beijing, China, in July 1999 as an Associate Researcher in the Internet Media Group and is now a Researcher with the Wireless and Networking Group. She has published over 40 refereed papers and is the inventor of several pending patents. Her current research interests include multimedia delivery over wireless, Internet, next-generation wireless networks, and P2P network/*ad hoc* networks. Currently, she is participating in TCP/IP header compression in ROHC WG in IETF. She is the principal contributor to the IETF ROHC-TCP WG draft.



Wen-Wu Zhu (S'92–M'97–SM'01) received the B.E. and M.E. degrees from the National University of Science and Technology, China, in 1985 and 1988, respectively, the M.S. degree from Illinois Institute of Technology, Chicago, and the Ph.D. degree from Polytechnic University, Brooklyn, NY, in 1993 and 1996, respectively, all in electrical engineering.

From August 1988 to December 1990, he was with the Graduate School, University of Science and Technology of China (USTC), and the Institute of Electronics, Chinese Academy of Sciences, Beijing, China. During 1996–1999, he was with Bell Labs, Lucent Technologies, in Whippany, Holmdel, and Murray Hill, NJ, as a Member of Technical Staff. While he was with Bell Labs, he performed research and development in the areas of Internet video, video conferencing, and video streaming over IP networks. He joined Microsoft Research, Beijing, China, in 1999 as a Researcher in the Internet Media Group, and is currently a Research Manager of the Wireless and Networking Group. He has published over 100 refereed papers in international leading journals and key conferences in the areas of wireless/Internet video transport, wireless/Internet multimedia communications and networking, and multimedia signal processing, and has contributed to the IETF ROHC WG draft on robust TCP/IP header compression over wireless links. He is the inventor of more than a dozen pending patents. His current research interests are in the area of wireless/Internet multimedia delivery and delivery networks.

Dr. Zhu has served as Guest Editor for Special Issues on Streaming Video and Wireless Video in IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (T-CSVT). In 2001, he received the T-CSVT Best Paper Award. He is a member of the Multimedia System and Application Technical Committee of the IEEE Circuits and Systems Society, the Multimedia Communication Technical Committee of the IEEE Communications Society, the Visual Signal Processing and Communication Technical Committee, and Eta Kappa Nu.

Ya-Qin Zhang (S'87–M'90–SM'93–F'98) received the B.S. and M.S. degrees in electrical engineering from the University of Science and Technology of China (USTC), Hefei, Anhui, China, in 1983 and 1985, respectively, and the Ph.D. degree in electrical engineering from George Washington University, Washington, DC, in 1989.

He is currently the Managing Director of Microsoft Research Asia, Beijing, China. Previously, he was the Director of the Multimedia Technology Laboratory, Sarnoff Corporation, Princeton, NJ (formerly David Sarnoff Research Center and RCA Laboratories). From 1989 to 1994, he was with GTE Laboratories Inc., Waltham, MA. He has been engaged in research and commercialization of MPEG2/DTV, MPEG4/VLBR, and multimedia information technologies. He has authored and co-authored over 200 refereed papers in leading international conferences and journals, and has been granted over 40 U.S. patents in digital video, Internet, multimedia, wireless, and satellite communications. Many of the technologies he and his team developed have become the basis for start-up ventures, commercial products, and international standards. He serves on the Board of Directors of five high-tech IT companies and has been a key contributor to the ISO/MPEG and ITU standardization efforts in digital video and multimedia.

Dr. Zhang served as the Editor-In-Chief for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from July 1997 to July 1999. He was the Chairman of the Visual Signal Processing and Communications Technical Committee of the IEEE Circuits and Systems (CAS) Society. He serves on the Editorial Boards of seven other professional journals and over a dozen conference committees. He has received numerous awards, including several industry technical achievement awards and IEEE awards, such as the CAS Jubilee Golden Medal. He was named "Research Engineer of the Year" in 1998 by the Central Jersey Engineering Council for his "leadership and invention in communications technology, which has enabled dramatic advances in digital video compression and manipulation for broadcast and interactive television and networking applications." He also received The Outstanding Young Electrical Engineer Award in 1998.