

Quality-of-Service Mapping Mechanism for Packet Video in Differentiated Services Network

Jitae Shin, JongWon Kim, *Senior Member, IEEE*, and C.-C. Jay Kuo, *Fellow, IEEE*

Abstract—This research presents a futuristic framework for quality-of-service (QoS) mapping between practically categorized packet video and relative differentiated service (DiffServ or DS) network employing unified priority index and adaptive packet forwarding mechanism under a given pricing model (e.g., DiffServ level differentiated price/packet). Video categorization is based on the relative priority index (RPI), which represents the relative preference per each packet in terms of loss and delay. We propose an adaptive packet forwarding mechanism for a DiffServ network to provide persistent service differentiation. Effective QoS mapping is then performed by mapping video packets onto different DiffServ levels based on RPI. To verify the efficiency of proposed strategy, the end-to-end performance is evaluated through an error resilient packet video transmission using ITU-T H.263+ codec over a simulated DiffServ network. Results show that the proposed QoS mapping mechanism can exploit the relative DiffServ advantage and result in the persistent service differentiation among DiffServ levels and the enhanced end-to-end video quality with the same pricing constraint.

Index Terms—Differentiated services (DiffServ), network pricing, packet video, quality-of-service (QoS), relative priority index (RPI), relative service differentiation.

I. INTRODUCTION

INTERNET applications have very diverse requirements on the network service, thus making the current best-effort Internet model less than sufficient. The emerging continuous media (CM) application demands more stringent quality-of-service (QoS) requirements than traditional TCP-based applications. Under the best-effort model, video applications at end-systems may adjust their rates through spatial/temporal quality adjustment in response to packet loss and delay feedback under TCP-friendly congestion control [1], [2]. However, maintaining the end-to-end video quality is too challenging to be accommodated by the best-effort Internet, since the video stream is inherently variable bit rate (VBR) and the Internet is an unpredictable time-varying channel. An alternative is to let the network provide a different level of assurance in terms of network QoS parameters within its resource capacity. Two representative approaches in the Internet engineering task force (IETF) are the integrated services (IntServ) with the resource reservation protocol (RSVP) and the differentiated services

(DiffServ or DS) [3], [4]. These Internet protocol (IP)-QoS methods are more suitable in accommodating various QoS requirements of different applications than the best-effort model. Between the two main IP-QoS approaches, the DiffServ scheme provides a less complicated and scalable solution since IntServ requires to maintain per-flow state across the whole path for resource reservation. In the DiffServ model, resources are allocated differently for various aggregated traffic flows based on a set of bits (i.e., DS byte). Consequently, the DiffServ approach allows different QoS grades to different classes of aggregated traffic flows. DiffServ working group in IETF have defined two services: 1) a premium service (PS) [5], which expects the virtual leased line service to support low loss and delay/jitter, and 2) an assured service (AS) [6], which provides better than best-effort but without guarantee.

While it is relatively clear how to build a predictable application by using protocols and mechanisms of RSVP and IntServ, the way to establish DiffServ-aware application in fine-granularity is still an open issue. Since the DiffServ specifies only local forwarding behaviors, the biggest challenge is to identify the way to best utilize the DiffServ for emerging applications. Ongoing research efforts in service differentiation can be divided into two directions based on absolute [7], [8] and relative [9], [10] concepts. In absolute service differentiation for guaranteed service, Stoica *et al.* propose a state-less core architecture that uses QoS parameter carried by packet header to provide fair-queuing [7] or guaranteed delay [8]. With per-flow state only at boundary nodes, the absolute differentiation seeks to provide an end-to-end absolute performance without per-flow state in the network core. In contrast, Dovrolis *et al.* [9], [10] promote the relative differentiation that provides a proportional service gap with their own proprietary scheduling. That is, a higher *DS level*¹ provides better (or at least not worse) queuing delays and packet losses.

The absolute service differentiation causes more complexity due to QoS provisioning overhead. It also trades off the flexibility for more guarantees. If we can limit the absolute portion of DiffServ as much as possible, it will enhance the flexibility and scalability of whole DiffServ architecture. Regardless of absolute or relative differentiation, the DiffServ already assumes intelligence at the boundary nodes of the DiffServ domain. As the Internet evolves toward the QoS model and as networked CM applications becomes more adaptive to QoS parameters, it appears that relative service differentiation can be a more attractive choice. Except for several conversational CM applications (e.g., Internet telephony including video conferencing), the majority of

Manuscript received March 6, 2001. This paper was presented in part at Packet Video Workshop 2000, Sardinia Italy, May 2000. The associate editor coordinating the review of this paper and approving it for publication was Dr. John Aa. Sorensen.

The authors are with the Integrated Media Systems Center and Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, CA 90089-2564 USA (e-mail: jitaeshi@sipi.usc.edu; jongwon@sipi.usc.edu; cckuo@sipi.usc.edu).

Publisher Item Identifier S 1520-9210(01)04323-1.

¹*DS Level* can be interpreted as the grade of quality provided to a group of packets having an identical DS codepoint in the IP header.

networked CM applications are tolerant to occasional delay/loss violations. Hence, they do not require tight delay/loss bounds, which can be better provided by DiffServ PS². For streaming video applications, whose encoding/decoding is more resilient to the loss-rate and delay fluctuations, the DiffServ AS seems better match. Thus, we can focus on the relative service differentiation and the AS of DiffServ for streaming video applications.

With an appropriate pricing rule, how to exploit relative differentiation for CM applications is a key factor to the successful deployment of DiffServ. The persistency of service differentiation is related to the pricing infrastructure of the Internet service provider (ISP). Only with persistent QoS provisioning in place, the service level can be pre-specified in the service level agreement (SLA) between ISP and end users. Thus, under a subscription-based pricing model for differentiated service quality among DS levels specified in the SLA, a futuristic framework for QoS mapping between practically categorized packet video and relative DiffServ network employing unified priority index and adaptive packet forwarding mechanism is proposed in this paper. In this framework, the video application at the source grades the chunks of its content by certain indexes (i.e., categories for packets) according to their importance in end-to-end QoS (e.g., in terms of loss probability and delay). Since these indexes reflect the desired service preference of a packet compared to others in fine-granularity, we denote it as relative priority index (RPI), which is further divided into a relative loss priority index (RLI) and a relative delay priority index (RDI). Then, the QoS control takes place in the form of assigning each packet with an appropriate DS level, which we call *QoS mapping*. Via the RPI association for each packet, an efficient (i.e., content-aware) mapping can be coordinated either at the end application or at the boundary node. Note that the efficiency of QoS mapping is also dependent on the persistent network differentiation for aggregated flows based on the DS level. That is, regardless of network load fluctuations, the DiffServ needs to provide persistent packet forwarding for an efficient QoS mapping, based on queue management and scheduling. With more persistent DiffServ network, the CM applications including streaming video can be built more reliably and less costly.

The main contribution of this paper is to propose a relative service differentiation framework connecting CM application, especially streaming video application, through the proposed RPI. It is divided into the following issues:

- 1) a relative priority based per-packet video categorization in terms of delay and loss;
- 2) an adaptive packet forwarding mechanism for aggregate flows to provide persistent DS levels, and
- 3) an effective QoS mapping between application categories and DS level under the pricing cost constraint of relative service differentiation network.

Actually, this framework belongs to the joint source-channel coding, more specifically unequal error protection (UEP) technique. Commonly, UEP enables a prioritized protection for source layers (e.g., layered streams of video). It can be realized at the transport-end with different levels of forward

error correction (FEC) and/or automatic repeat request (ARQ) for each layer [11]–[14]. However, to the best of our knowledge, none of the UEP approaches have touched the issue of packet-level fine-grained prioritization of the proposed RPI instead of layered protection. Especially for DiffServ network, only layered prioritization in absolute differentiation sense has recently been proposed in [15], utilizing the video object layer of MPEG-4 and different packet discarding mechanism.

The rest of this paper is organized as follows. The overall framework of the proposed QoS mapping framework is described in Section II. Video categorization with RPI according to several criteria is examined in Section III for the case of ITU-T H.263+ video [16]. By investigating the error resilient version of H.263+ stream, the RPI is assigned so that different video packets can be tied with the relative loss-rate/delay differentiation of DiffServ networks. Then, the adaptive packet forwarding mechanism is proposed in Section IV to support persistent DS levels, adopting the commonly accepted random early detection (RED) [17] or weighted fair queuing (WFQ) [18] combination. An effective QoS mapping guidance using proposed content-aware forwarding mechanism is presented for several possible DiffServ deployment scenarios in Section V. Performance assessments by using network simulator (ns) [19] and H.263+ video are given in Section VI, where the implications of experimental results are also discussed. Concluding remarks and future works are given in Section VII.

II. OVERALL PROPOSED QoS MAPPING FRAMEWORK IN DIFFSERV NETWORK

A. Overall QoS Mapping Framework

QoS parameters defined to measure service quality include latency (delay and delay jitter), packet loss-rate, and throughput. Among them, throughput (or allocated bandwidth) represents the requested rate to initiate and sustain a CM application. Once the CM application is established and operational, it becomes less critical if the application has the adaptation capability to the available bit rate. With the relative differentiation in mind and the throughput is per-flow QoS parameter, we currently exclude the throughput from the per-packet QoS parameters. Instead, we are assuming that the traffic conditioning entities of the DiffServ network is taking care of the throughput negotiation. Note that the relative DiffServ concept is more flexible in terms of bandwidth guarantee. Thus, in our QoS mapping mechanism, the basic QoS 2-tuple *delay, loss* is considered and requested to different degrees by end-user applications, anticipating different guarantee (or assurance) levels for the QoS tuple based on the price that the end-user is willing to pay. Each user application will demand its loss rate/delay preference by marking its DS field with the RLI and RDI for loss-rate and delay, respectively.

The diagram of the proposed QoS mapping framework is shown in Fig. 1, where service differentiation is expressed in terms of loss-rate/delay associated with forwarding queues. Each video flow of a user application has to be classified in the loss-rate/delay preference and each packet is associated with RPI composed of two normalized indexes, RLI and RDI, that are described in detail in Section III. These RPI associated

²Note that DiffServ PS can be regarded more in-line with the absolute differentiation than the relative differentiation.

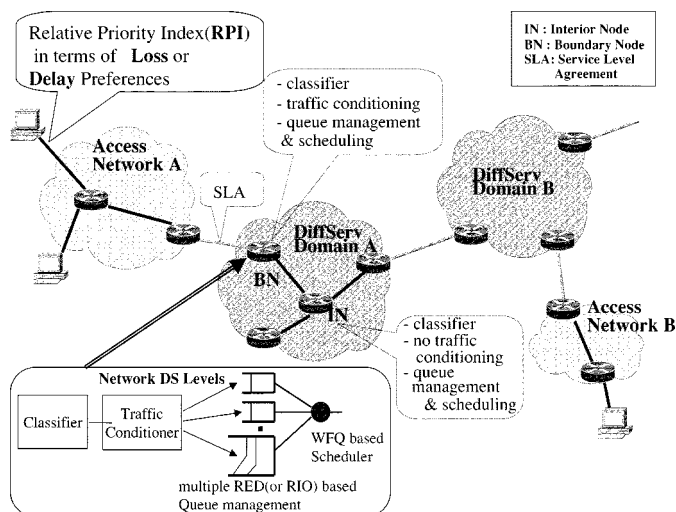


Fig. 1. Overall QoS mapping diagram based on the relative priority index and network DS levels.

packets are categorized into intermediate *DS categories* in fine-grained manner, albeit independent of underlying network if required. Then, pre-marked packets (i.e., RPI categorized) are conveyed into the DiffServ-aware node for QoS mapping, which can be located at the end-system itself, a special version of DiffServ boundary node, and both. Thus, given a video application and the responding DiffServ network, their QoS mapping is accomplished by mapping (i.e., marking) the relative prioritized packets to maximize end-to-end video quality under a given cost constraint. Then, at the DiffServ junction (i.e., DiffServ boundary node), the packets are classified, conditioned, and re-marked to certain network DS levels by considering the traffic profile based on the SLA and the current network status. Finally, the packets with DS level mapping are forwarded toward the destination through a packet forwarding mechanism that is mainly composed of queue management and scheduling scheme. The forwarding mechanism in a DiffServ network should provide the corresponding network DS levels in the relative proportional differentiation sense. The desired differentiation in queuing may be realized by adopting multiple queues (MQs) with several drop curves such as multiple RED and RED with in and out bit (RIO) [6], [20]. Furthermore, by adopting different weighting factors, a modified version of WFQ scheduling can be used to complement the queue management to provide the desired loss-rate/delay differentiation.

The above QoS mapping framework assumes the existence of a futuristic DiffServ network that supports prioritized variable-rate delivery and the associated pricing mechanism. Since a video codec has several options to trade the compression efficiency for flexible delay manipulation, error resilience, and network friendliness, the QoS coordination has to provide a simplified QoS mapping process between the video encoder and the target network. The purpose of introducing RPI is to abstract and isolate coding details from the network adaptation. By assigning RPI to each packet in an appropriate manner (i.e., keeping the fine-granularity as much as possible), the proposed delivery system can accommodate the demand of each packet to achieve the best end-to-end performance in adapting to network fluctuations.

B. Deployment Issues and Scenarios

Typical DiffServ architecture defines a simple forwarding mechanism at interior network nodes while pushes most of the complexity to network boundaries [3], [4]. The traffic conditioner (composed of the meter, the marker, the shaper, and the dropper) is placed at the boundary of DiffServ network domains. Given this functionality at the boundary, interior nodes use a packet forwarding mechanism with queue management and scheduling for incoming packets to deliver differentiated services to various packets. This DiffServ architecture can bring benefit to both end-user and ISP by providing better service quality for CM application at the willingness to pay more for higher quality. Thus, the design principles for QoS mapping should consider interests of both end-user and ISP. That is, an end-user should get benefit from his DiffServ-aware application in end-to-end quality while ISP should enjoy the merit of flexible charging based on the end-user satisfaction. In order to handle this negotiation, we need to measure the QoS demand of CM application and the QoS supply of DiffServ networks in terms of pre-defined granularity. With a pre-defined granularity, the service differentiation can be demanded by marking differently at end-system to ask for targeted DS levels. It then may be adjusted (i.e., re-marking) in the DiffServ network and handled (i.e., forwarding) accordingly.

Each DS level is identified by means of type of service or DS byte (i.e., DS codepoints) defined in IP headers. DiffServ working group also defines per-hop behaviors (PHBs) upon DS byte to specify the required forwarding behavior for the packets according to DS levels. Among initial PHBs being standardized are the expedited forwarding (EF) PHB for DiffServ PS [21] and the assured forwarding (AF) PHB for DiffServ AS [22]. The EF PHB group specifies a forwarding behavior in which packets see very small losses and queuing delays. The EF PHB, based on the priority queuing, better suits latency stringent applications at the cost of higher price. The AF PHB group specifies a forwarding behavior to preferentially drop best-effort and/or out-of-profile packets when congestion occurs. By limiting the amount of AF flows and by managing the best-effort traffic appropriately, network nodes can ensure a lower loss behavior to AF marked packets. As a result, the DiffServ provides DS levels of different losses and delays. For example, one EF queue, four AF queues with three drop preferences, and one BE queue may be defined as depicted in the network side of Fig. 2. We can draw three equivalent cost lines in Fig. 2, imagining several pricing model possibilities of ISP. Line (a) considers only loss-rate, while line (c) depends only on delay. Line (b) relies on both loss-rate and delay and it is flexible. That is, at the same cost, it provides various service combinations such as one with higher delay but a lower loss-rate and vice versa.

Different DS levels are to be provided based on the marking (on DS byte) of an application packet and different loss and delay is expected according to the requested DS level. Thus, it is natural to think of associating the packet with both loss and delay priorities (i.e., RLI/RDI) rather than loss alone, albeit in a fine-grained manner. For streaming video applications, the RLI association for each packet should reflect the loss impact of each packet to the receiving end-to-end video quality.

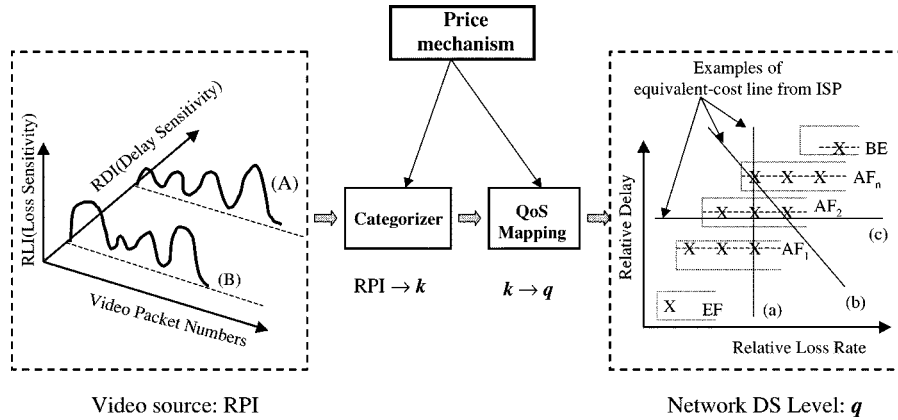


Fig. 2. QoS mapping from source RPI into network DS levels.

For RDI, classification of video streams depends more on the application context (e.g., video-conferencing or video on demand) rather than video contents within a stream. For example, as shown in the source side of Fig. 2, the quality request of two video applications (A) and (B) have clear distinction in terms of RDI depending on application usage, with added variability expressed by RLI. Within a stream, RLI and RDI attributes of packets are however not completely orthogonal, rendering a reasonable classification somewhat difficult (to be discussed further in Section III). Considering complexity in varying RDI per packet, we believe an appropriately fixed RDI with varying RLI for each packet within an application is more than sufficient. Thus, in this paper, we just assign a fixed RDI for all packets of an application as shown in Fig. 2 and try to find out satisfactory range of DS levels for the given RDI.

Given RPI for each packet, our goal is to explore the best QoS mapping for video packets with the content-aware forwarding under a cost constraint. At the end-system, RPI is associated for each packet and it is then categorized into k th category among K DS categories. However, it is still up to a specific deploying environment where the QoS mapping is conducted. If the QoS mapping is conducted at the network adaptation unit of the end-system, it can take advantage of its content-awareness (i.e., original RPI) to its extreme. Also, it can cover the early stage of DiffServ deployment, since it does not require any additional supporting network node except for prioritized DS levels. However, it lacks the knowledge on the dynamics of network as well as the aggregation effect of competing flows, which can impede the efficiency of mapping without proper feedback mechanism in place. To better fit into the access network scenario shared by multiple DiffServ-aware applications, it may be worthwhile to introduce a special version of DiffServ boundary node to handle the proposed QoS mapping. With this, we can treat effective QoS mappings between aggregated CM packets and network DS levels (with fluctuating, but bounded service levels) under the traffic condition agreement in the SLA. By adjusting the QoS mapping dynamically by the coordinated interaction with end-systems, one can expect sophisticated exploitation of the DiffServ advantage.

However, note that in this paper we want to promote the futuristic concept of proposed QoS mapping framework while deferring the discussion of the practical issues such as the dynamic

QoS mappings and their aggregation effect.³ Focusing on the QoS mapping problem of each single flow, the potential of the proposed framework is intensively investigated. Especially, how to build the required building component such as the RPI association/categorization, the persistent packet forwarding mechanism desired, and the practical formulation of QoS mapping are to be discussed. Thus, with several DiffServ network deployment scenarios and corresponding QoS mappings (to be detailed in Section V), both network and end-to-end video performances of the proposed mapping framework are evaluated through the end-to-end streaming video over the simulated DiffServ networks.

C. QoS Mapping Problem Formulation and Guidance Solution

QoS mapping of the relative prioritized packet on to the DS level can be formulated into the following optimization problem per each single flow as follows. Each packet i of the flow, mapped to a certain network DS level $q \in \{0, 1, \dots, Q-1\}$, gets an average packet loss-rate $\bar{l}_{q(i)}$ by paying unit price $p_{q(i)}$. For the sake of simplicity, we currently enforce constraint based on a fixed RDI instead of incorporating it into the optimizing objective. That is, a fixed RDI per flow is used to limit the allowed range of $[q_{\min}, q_{\max}]$ to which packets can be mapped without violating the delay requirement. Thus, if RLI can represent the loss impact to the video quality successfully, the effort to achieve the best end-to-end quality while satisfying the total cost P can be formulated into the minimization of the quality degradation QD

$$\min_{\vec{q}_i} QD = \min_{\vec{q}_i} \left(\sum_{i=1}^N RLI_i \cdot \bar{l}_{q(i)} \right)$$

subject to

$$\sum_{i=1}^N p_{q(i)} \leq P, \quad (1)$$

where total packet numbers of flow is N , a QoS mapping is denoted by $\vec{q}_i = \{q(1), q(2), \dots, q(N)\}$, and $q(i)$ is DS level to which i th packet is mapped.

³Regarding dynamic QoS mapping issue, some preliminary investigation is done in our latest work on the specially designed DiffServ boundary node [23].

However, since individual mapping choice per each packet is too complex, we are trying to solve a simplified problem by fixing the mapping decision for all packets of DS category k ($\in \{0, 1, \dots, K-1\}$) to a single DS level q . That is, (1) becomes simplified to (2). Here, we only consider an ordered mapping based on the DS category k such that packet with higher k only be mapped to at least equal or larger q than that of lower k packet. Note that the QD in (2) is an expected QD , where the loss impact of a packet belonging to category k is represented by the average loss effect \overline{RLI}_k

$$\min_{\vec{q}_k} \sum_{k=0}^{K-1} QD_k = \min_{\vec{q}_k} \left(\sum_{k=0}^{K-1} \overline{RLI}_k \cdot \bar{l}_{q(k)} \cdot n_k \right)$$

subject to

$$\sum_{k=0}^{K-1} p_{q(k)} \cdot n_k \leq P$$

for

$$\sum_{k=0}^{K-1} n_k = N \quad (2)$$

where n_k is the packet number of k category. A QoS mapping is now denoted by $\vec{q}_k = \{q(0), q(1), \dots, q(K-1)\}$, and $q(k)$ is DS level to which k category is mapped.

To solve the above optimization problem, we should consider the following factors. First, in real situation we have to limit the resource allocation based on the traffic conditioning agreement specified in the SLA with the DiffServ network. Thus, the above formulation has to be constrained further by the traffic conditioning agreement before practical utilization. Next, the cost function in the above may better reflect real situation if the added complexity and the out-of-order arrival handling cost due to dynamically scattered packets over multiple DS levels are included. These factors are assumed negligible at the current stage. Finally, $p_{q(k)}$ depends on the DiffServ provisioning, which will be decided by the service provider and negotiated in the SLA. Persistent service differentiation implies that average delay and loss rate are proportional to the DS level q choice. Note, in order to provide the persistent DS levels regardless of the dynamically changing network load, an adaptive packet forwarding mechanism comprising intelligent traffic conditioning, queue management and packet scheduling is highly required.

To use as the guidance solution, assuming that the network DS levels maintain persistent differentiation, optimal solution for idealized situation can be derived as follows. For the idealized network situation, the packet loss-rate \bar{l}_q might decrease exactly as DS level q increases. Typically, loss-rate \bar{l}_q increases inversely proportional to DS level q increase.⁴ Unit price p_q is then assumed to be proportional to DS level q . Then, the optimal mapping solution $\vec{q}_k^* = \{q^*(0), q^*(1), \dots, q^*(K-1)\}$ to (2) can serve as a guidance. This constrained optimization problem

⁴Note that the loss-rate and DS level may take diverse relationship in real practice. However, this typical relationship seems to provide a reasonable approximation.

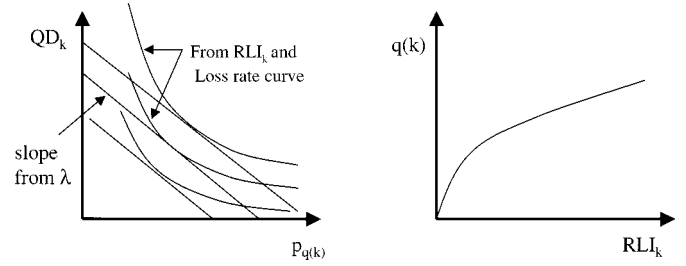


Fig. 3. Mapping from RLI to the DS level. (Given relationship for \overline{RLI}_k categorization, loss-rate per DS level, and the pricing strategy, the Lagrangian formulation leads to optimal mapping.)

can be solved by finding the QoS mapping \vec{q}_k^* that minimizes the Lagrangian formula

$$J_k(\lambda) = \left[\overline{RLI}_k \cdot \bar{l}_{q(k)} + \lambda \cdot p_{q(k)} \right] \cdot n_k. \quad (3)$$

Then, by searching around the convex hull from the graph of QD_k ($\equiv \overline{RLI}_k \cdot \bar{l}_{q(k)} \cdot n_k$) versus cost, we can get the optimal mapping solution as depicted in Fig. 3. To be more specific, the solution is to set total price equal to the price budget P at which the slope λ line intersects the quality degradation curve by adjusting λ under given loss curve versus price (or DS level).

In particular, we can even get closed form solution with further assumption on several relationships for RLI categorization, loss rate per DS level, and the pricing strategy. If average loss-rate of DS level q is inversely proportional to DS level q [i.e., $\bar{l}_{q(k)} = L/q(k)$], and unit price per DS level q is linearly proportional to DS level [i.e., $p_{q(k)} = P_1 + P_2 \cdot q(k)$]—where L , P_1 and P_2 are given constants—the Lagrangian formula of (3) becomes

$$J_k(\lambda) = \left[\overline{RLI}_k \cdot \frac{L}{q(k)} + \lambda \cdot (P_1 + P_2 \cdot q(k)) \right] \cdot n_k. \quad (4)$$

By trying $\partial J_k(\lambda) / \partial q(k)$ and the constraint in (2), the resulting closed form solution is expressed by

$$\lambda = \frac{L \cdot P_2}{(P - P_1 \cdot N)^2} \left[\sum_{k=0}^{K-1} \sqrt{\overline{RLI}_k} \cdot n_k \right]^2, \quad (5)$$

$$q(k) = \frac{P - P_1 \cdot N}{P_2} \cdot \frac{\sqrt{\overline{RLI}_k}}{\sum_{k=0}^{K-1} \sqrt{\overline{RLI}_k} \cdot n_k}.$$

This result means that optimal QoS mapping $q(k)$ (i.e., the mapping k to q) is proportional to $\sqrt{\overline{RLI}_k}$ with the weighting factor coming from total budget P and pricing function for $p_{q(k)}$ since the sum of denominator is a constant.

III. VIDEO PACKET CATEGORIZATION BASED ON RELATIVE PRIORITY INDEX (RPI)

A. Desired Characteristics for Prioritization

In our approach, the preference over packet loss-rate and delay is determined based on the following criteria. First, it focuses on relative and fine-grained priorities inside an application, relying a linkage to an absolute metric at later stages. When initially marked in the DS field with RLI/RDI,

video application packets at the user-end have a limited knowledge about the dynamic status of the network and competing applications within the same boundary. The assigned relative priority for each packet is supposed to be interpreted at the DiffServ-aware node. Next, since more assured but not guaranteed service is to be provided, the video application should cope with possible packet losses and delays. The assigned RLI/RDI parameters basically assume that the video stream has already been generated with error resilience features so that the loss or delay of each packet can be tolerated to a certain degree. Finally, the resulting prioritization should exhibit some kind of clustering behavior in the RLI/RDI space so that it can keep its distinction when mapped to the limited DS byte.

As discussed before, delay preference association of video streams depends more on the application context. If we consider different delay for each packet within a stream, different demands on delay are usually connected with the layered coding of video compression. For example, the I, P, B frames of ISO/IEC MPEG-1/2/4 frames has variant demands on delay as well as loss. The situation is similar for the spatial scalable, SNR scalable, and data partitioned layers of MPEGs or H.261/H.263 with the exception of temporal scalable layer. However, as the video application becomes network aware, the trend seems to move gradually toward delay variation even within a stream. A good example is the asynchronous media transmission scenario, where flexible delay margins can be exploited throughout the transmission. This idea is later denoted as delay cognizant video coding in [24] by employing an extended form of region-based scalability to H.263 video. Multiple region layers within a stream are constructed in [24], where varying delay requirements are associated by perceiving a motion event such as nodding, gestures or maintaining lip synchronization. To be more specific, it assigns the most visually significant region to the lowest delay and vice versa. Since packets with the longer delay demand may arrive late, this presents the opportunity where the network service can route packets through less congested but longer routes and charges at a lower price. Another example is packets of the MPEG-4 system that encompass multiple elementary streams with different demands in one umbrella. This kind of integrated stream can justify the demand for inter-media RDI/RLI differentiation. Thus, we propose to distinguish a packet based on the loss-rate and delay tuple, leaving flexibility to subsequent stages. Note however, that only fixed RDI for a flow is employed currently.

B. Proposed RLI Association and Categorization

Under packetized video transmission, the RLI assignment for a packet would be best if it can precisely represent its error propagation impact to the receiving video quality. However, the specific method for RLI prioritization is totally application (furthermore video compression scheme) dependent. Thus, we have chosen ITU-T H.263+ video [16] as the evaluation codec, considering its wide acceptance for low-latency video conferencing and its potential for video streaming. Note that there is not much difference among the motion-compensated prediction codecs, which includes both MPEGs and H.261/H.263, regarding this RLI association.

Given frame size and target bit rate, each packet has a different loss effect on the end-to-end video quality due to inter-frame prediction. Impact of packet loss may spread within a frame up to the next resynchronization (e.g., the picture or the GOB headers) due to differential coding, run-length coding, and variable length coding. This is referred to as spatial error propagation and may damage any type of frame. For temporal error propagation, damaged macroblocks (MBs) affect the non intra-coded MBs in subsequent frames which use corrupted MBs as references. Recent researches on the corruption model [25] attempt to model the loss effect. The corruption model is a tool used to estimate the packet loss impact to the overall received video quality. However, most modeling efforts have focused on the statistical side of the loss effect while a dynamic solution is required in our approach. Thus, instead of computationally complex options, we have devised ways to associate RLI to H.263+ encoded packet stream with a simple and on-line calculation method.

Basically, we use the error resilient H.263+ stream, compressed at a target rate of 384 kbps for a common intermediate format (CIF) test sequence with 10 frames per second. Several error resilience and compression efficiency options (“Annexes D, F, I, J, and T” with random intra refresh) are used in the generation of the so-called “Anchor” (i.e., GOB) mode stream. The random intra-refresh rate is set to 5% to cover the network packet loss. It is then packetized by one or more packets per GOB, which is dependent on the maximum transfer unit (MTU) size of the IP network. Thus, we propose a simple yet effective RLI association scheme for this H.263+ video stream, which is calculated for each GOB packet.

Hence, the following video factors are taken into consideration for the proposed RLI association. First, the magnitude and direction of the motion vector for each MB is included to reflect the loss strength and temporal loss propagation due to motion. Then, encoding types (intra, intra-refreshed, inter, etc.) are considered. That is, the refreshing effect is added by counting the number of intra-coded MBs in the packet. Lastly the initial error due to packet loss is considered assuming the normative error concealment adopted at the decoder. Then, RLI for a packet may take into account these video factors together by summarizing the normalized video factors with an appropriate weighting⁵ as

$$RLI_i = \sum_{n=1}^{N_{VF}} W^n \times \frac{VF_i^n}{m_i^n} \quad (6)$$

where N_{VF} stands for the number of video factors to be considered, W^n for the corresponding weight factors, VF_i^n for the magnitude of video factor VF activity n , and m_i^n for sampling mean of VF^n for i -th packet. The normalization is done on-line through updating sampling mean, $m_i^n = \frac{i-1}{i} \cdot m_{i-1}^n + \frac{VF_i^n}{i}$ at i -th packet generated time. Fig. 4 shows the resulting RLI assignment example for “Foreman” sequence.

⁵Recently, we have extended our investigation to provide more systematic RPI association and have devised dynamic version of corruption model. By extending the above video factors such as initial error, temporal dependency (by motion vector and MB encoding type), and spatial filtering effect, this MB-based corruption model provides RPI association to approximate the actual loss impact in mean square error (MSE) [26].

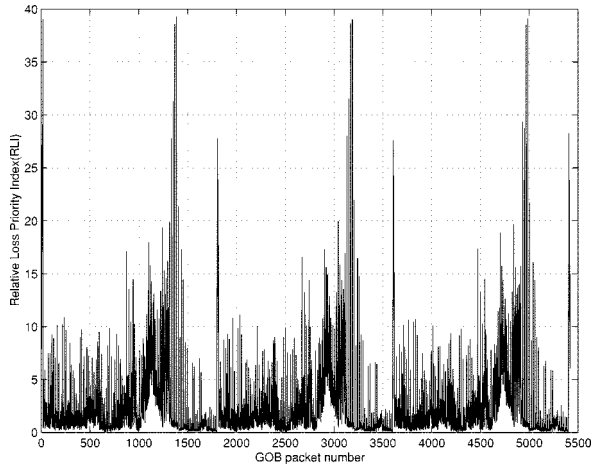


Fig. 4. RLI for “Foreman” sequence obtained by applying W^{rs} s (0.15, 0.15, and 0.7) for the three video factors of such as motion vector, the number of intra-coded MB, and initial mean square error.

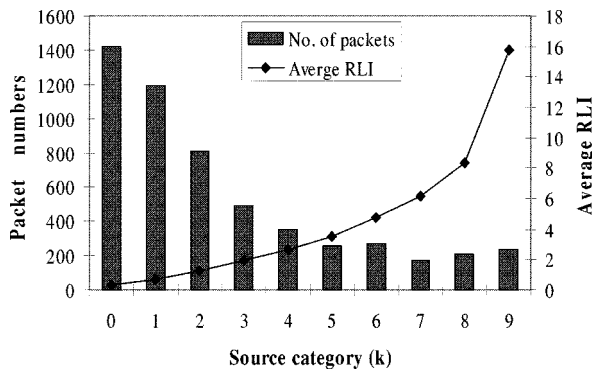


Fig. 5. Packet distribution and average RLI of each video DS category for Foreman sequence.

Finally, RLIs are categorized into K DS categories to enable mapping to limited DS field space (or eventually to be ready for mapping to more limited network DS levels). In our approach, simple nonuniform quantization of RLI is performed for this categorization. That is, as shown in Fig. 5, the categorization is done with gradually increasing step as the category k increases. Another possible approach is to categorize packets into equal number (i.e., uniform distribution). After categorization, all packets belong to category k may be represented by their average RLI value \overline{RLI}_k .

IV. PROPOSED ADAPTIVE PACKET FORWARDING MECHANISM

In this section, we present a packet forwarding mechanism to provide persistent network service differentiation in the QoS tuple *delay, loss* even though the network load condition is time-varying. The probable candidate for the required packet forwarding mechanism can be based on the popular weighted fair queuing (WFQ), which is already being deployed in some CISCO routers for IP QoS. However, previous work [27], [28] shows that WFQ with static weighting factors (static WFQ) is not capable of providing the persistent service differentiation in a finer time-scale. A proprietary scheduling is instead proposed for a proportional DiffServ [9], [10], limiting its practical usability. Thus, in this paper, we propose a mechanism to convert

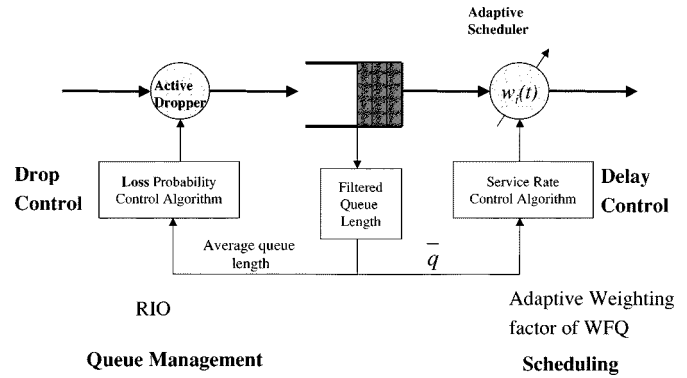


Fig. 6. Control loop for the proposed proportional loss/delay service differentiation (adaptive WFQ).

the combination of WFQ scheduling and multiple RED (more specifically RED with in/out bit: RIO) queue management [6], [20] into an adaptive packet forwarding mechanism named an *adaptive WFQ*. WFQ provides a weighted portion of the shared bandwidth to each class queue according to its weighting factor. Within a class queue, RIO provides a different drop preference through drop probability control algorithm using a different RIO parameter set consisting of the minimum threshold, the maximum threshold, and the maximum drop probability (\min_{th} , \max_{th} , \max_p) for each IN or OUT control curve, respectively. Thus, in order to provide differentiated services on distinct classes persistently, we are interpreting the already admitted traffic and adapting to the dynamic network condition. Among several network status indexes such as packet loss-rate, the round trip time (RTT), and the filtered queue length, we select the filtered queue length and its change rate. Then, focusing on the proportional packet loss/delay differentiation, the control loop algorithm is devised, as shown in Fig. 6.

To maintain the relative loss/delay differentiation, weighting factors of WFQ with RIO are dynamically adjusted. The serviced delay/loss ratio per each DS level is dependent on the allocated portion of shared bandwidth at each time interval. If the desired differentiation among M class queues are quantified as the ratios, δ_i , $i = 0, 1, \dots, M - 1$. We found the measured loss-rate and delay depend together on the weighting factor $w_i(t)$ of WFQ heavily. Drop rate on each DS level is proportionally adjustable by setting the RIO parameters (\min_{th} , \max_{th} , \max_p) without frequent updating since average queue length reflects the queue build-up condition. Then it is sufficient to change $w_i(t)$ dynamically in order to get stable and persistent service differentiation in our goal.

The proportional differentiation of delay (d_i) and loss (l_i) can be stated as

$$\frac{d_i}{d_j} = \frac{\delta_i}{\delta_j}, \quad \frac{l_m}{l_n} = \frac{\delta_m}{\delta_n}. \quad (7)$$

To provide the desired differentiation, the adaptive weighting factors of WFQ, $w_i(t)$, $i = 0, 1, \dots, M - 1$, for the time interval $[t - \Delta\tau, t]$ are

$$\frac{w_j(t)}{w_i(t)} = \frac{\delta_i \bar{q}_j(t)}{\delta_j \bar{q}_i(t)} \quad (8)$$

TABLE I
PERFORMANCE COMPARISON BETWEEN STATIC WFQ AND ADAPTIVE WFQ

Classes	Static WFQ			Adaptive WFQ		
	AF ₁	AF ₂	BE	AF ₁	AF ₂	BE
delay(msec)	11.8	17.6	33.1	10.7	18.5	35.2
deviation(msec)	11.78	17.37	22.33	9.96	15.53	57.27
delay ratio between classes	1.49		1.88	1.73		1.91
loss rate(%)	6.2	11.4	34.0	6.1	11.5	25.6
loss ratio between classes	1.84		2.98	1.89		2.23

where $\bar{q}_i(t)$ is the filtered queue length at time t . With the assumption $\sum_i w_i(t) = 1$, we have

$$\left(1 + \frac{\delta_0 \cdot \bar{q}_1(t)}{\delta_1 \cdot \bar{q}_0(t)} + \dots + \frac{\delta_0 \cdot \bar{q}_{M-1}(t)}{\delta_{M-1} \cdot \bar{q}_0(t)}\right) \times w_0(t) = 1.$$

Then, we can obtain the adaptive weighting factors via

$$w_i(t) = \frac{\bar{q}_i(t)}{\sum_{l=0, l \neq i}^{M-1} \frac{\delta_i}{\delta_l} \cdot \bar{q}_l(t)}. \quad (9)$$

To demonstrate the effectiveness of the proposed adaptive forwarding, the loss/delay differentiation performance of static and adaptive WFQ is compared by network simulation. Comparison is conducted for a shared node with three class queues (AF1, AF2, and BE) with 4:2:1 proportional differentiation target. To generate network loads, each queue has been injected with five TCP and five UDP (jittered CBR) traffics. The results have been tabulated in Table I, which shows the enhanced persistence of the proposed forwarding scheme. Also, the average queuing delays calculated per 100 received packets are depicted in Fig. 7 and static WFQ fails to provide the highest priority class queue with the proportional lowest relative delay. In contrast, the proposed adaptive WFQ provides much consistent delay service differentiation between class queues without inversion of QoS parameters. Finally, note that the proposed scheme is compatible with any kind of scheduling schemes, and their weighting factor can adjust the relative delays/losses among different class queues adaptively based on the network conditions.

V. QoS MAPPING SCENARIOS

Suppose that there are K categories of video packets indexed by RLI/RDI and Q DS levels. We now examine the QoS mapping from k to q in an open loop control environment (i.e., without feedback) under a cost constraint. Our goal is to provide a practical performance to the solution of the optimization problem given in (2). Remember that each video category indexed by k is to be mapped into the same q DS level without splitting. The following three scenarios are considered. First, the simplified situation, where every packet is assigned to a single queue with different drop preference. In second scenario, content forwarding is extended to MQs. The approach of feeding a video flow into a single q DS level will be compared to that of spreading packets to MQs. Finally, to check the effect of RDI constraint, MQ situation is evaluated with different delay constraint enabled.

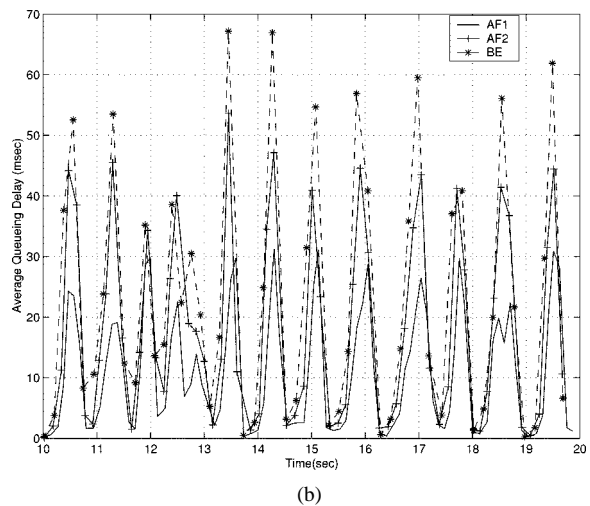
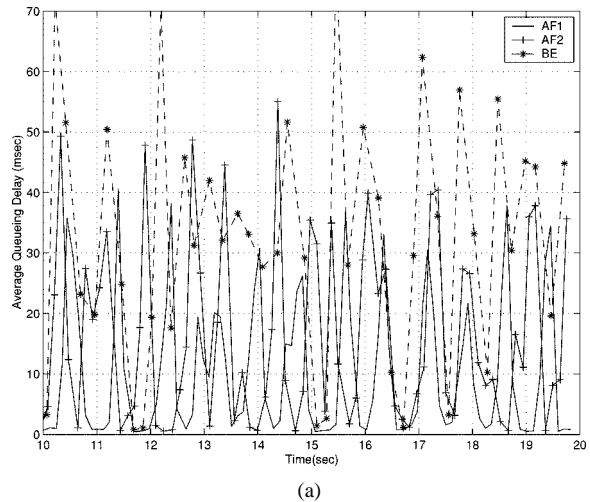


Fig. 7. Average queuing delay comparison of three class queues (AF₁, AF₂, and BE) under (a) static WFQ and (b) adaptive WFQ scheduling schemes.

A. Single Queue Scenario with RLI Prioritization

Current Internet routing still relies on the mixture of FIFO and RED queues with a single queue, which is basically content-blind. Thus, a straightforward extension to enable the loss differentiation by a single queue may be an interim solution toward the DiffServ. Under this scenario, each video stream is assigned to a queue with several DS levels according to its price, which provides the required loss differentiation by adopting multiple drop preference only. Only loss differentiation is considered in this case, since en-queued packets is served as first-in first-out (FIFO) manner and delay differentiation is not possible. This scenario is useful in verifying the appropriateness of the proposed RLI. By comparing it against the content-blind forwarding, we can confirm the effectiveness of our RLI prioritization. Content-based video categorization plays an important role if there exists a cost constraint. This situation represents a certain type of user flows, which cannot afford to pay more, to experience gradual quality loss as traffic increases.

The traffic conditioner based on the static traffic profile from the SLA is not efficient in handling bursty video flows since it is basically content-blind. But with the support of the proposed RLI QoS mapping, the situation changes. By adjusting several

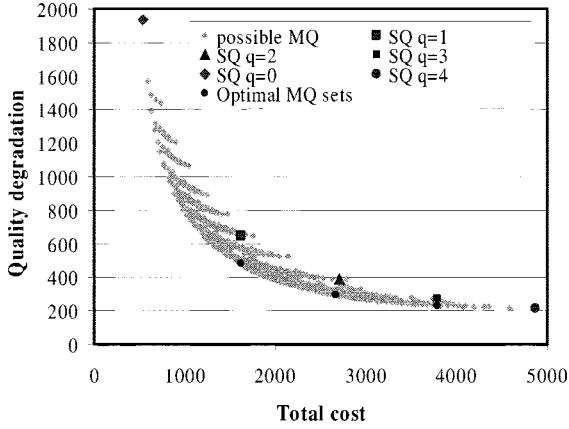


Fig. 8. Quality degradation versus the pricing budget for RLI-based service differentiation with the “Foreman” sequence (Ideal guidance case).

thresholds on the RLI, as shown in Fig. 4 and coupling a target rate with threshold adjustment, loss-differentiation with RLI can be implemented through modified RIO with three-level drop rates. The packets with higher RLI below a target rate will get the desired q DS level while those with lower RLI are pushed to lower q DS levels under the cost constraint. This case has simple QoS mapping connecting the RLI threshold with the target rate, and will be evaluated in Section VI-A with experimental results.

B. Multiple Queue Scenario with RLI Prioritization

The MQ scenario can provide delay as well as loss differentiation. We examine the loss differentiation of MQ here and the loss/delay combined differentiation is to be described next. The QoS mapping is attempted without constraining a stream into a single queue any more. It is, however, assumed that an intelligent receiver is capable of correcting the negative effect (i.e., out-of-order packet arrival), which might be caused by different delays according to the DS levels.

As discussed in Section II-C, we can get a guidance for the effective QoS mapping assuming several input relationships. This guidance mapping is an ideal mapping to minimize (2). If RLI used in this guidance mapping is effective, the resulting QoS mapping can lead to the best end-to-end video quality under the given constraints. First, the categorized RLI relation, depicted in Fig. 5, is used to represent the fine-grained priorities in $K = 10$ categories. Then, total $Q = 5$ DS levels are provided as depicted in Fig. 9(a). These DS levels are differentiated according to the assumed cost versus DS level relation (i.e., $p_q = 0.1 + 0.2 \cdot q$) and the loss-rate versus cost relation (i.e., $\bar{l}_q = 0.015/p_q$). Thus, for these pre-defined condition, quality degradation versus the cost budget is calculated and plotted in Fig. 8. From each point corresponding to a unique mapping, we can observe the different quality degradation and total cost combination, which as a whole set provide the desired convex hull.

In fact, by checking the convex hull of Fig. 8, we can locate the best mappings for the target pricing budgets. The cases of mapping all packet of k DS category into a fixed DS level only, denoted by SQ (single queue) mapping, usually produce worst quality degradation. Thus, we have chosen three most efficient mappings that use the same total budgets with the corresponding

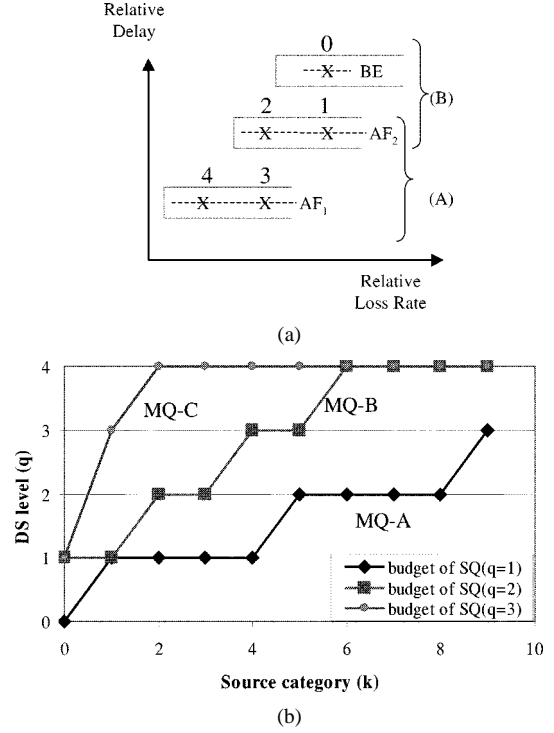


Fig. 9. (a) Utilized network DS levels for MQ scenarios: case (A) for delay-stringent case and case (B) for delay-tolerant case. (b) Efficient guidance QoS mappings (MQ-A, MQ-B, and MQ-C) for different total cost budgets, which consume equal budgets for SQ $q = 1, 2,$ and $3,$ respectively.

SQ mappings, SQ ($q = 1$), SQ ($q = 2$), and SQ ($q = 3$), respectively. Those are multiple DS level mapping cases, denoted by MQ-A, MQ-B, and MQ-C. These QoS mappings are plotted in Fig. 9(b). Also, in K -tuple $\{q^*(0), q^*(1), \dots, q^*(K-1)\}$ format, they are MQ-A $\{0, 1, 1, 1, 1, 2, 2, 2, 3\}$, MQ-B $\{1, 1, 2, 2, 3, 3, 4, 4, 4, 4\}$, and MQ-C $\{1, 3, 4, 4, 4, 4, 4, 4, 4, 4\}$. The gap between MQ-A and SQ ($q = 1$) in Fig. 8 represents the efficiency of guidance mapping. The same applies to MQ-B and MQ-C over SQ ($q = 2$) and SQ ($q = 3$), respectively. In addition, these mapping sets are in match with the closed form in (5) as a mapping guidance. Finally, note that these guidance mapping sets and their corresponding single DS level mappings are used in the experiment in Section VI-B.

C. Multiple Queue Scenario with Both RLI and RDI

Under this scenario, we consider the MQ mapping with both RLI and RDI together. However, as discussed earlier, RDI is taken into account only at the application level. Users assign RDI to each of their flows according the delay bound requirement of the flow. This limits the use of DS levels for the QoS mapping of a video source as illustrated by case (A) for delay-stringent case or (B) for delay-tolerant case in Fig. 9(a). Within the acceptable range of DS levels, the categorized video stream is mapped. This case has a smaller mapping combination than MQ with RLI only, which actually makes QoS mapping less efficient. In this case, the MQ mapping set is determined as a suboptimal mapping set from the possible mapping combinations of Fig. 8. The experimental result will be presented in Section VI-C.

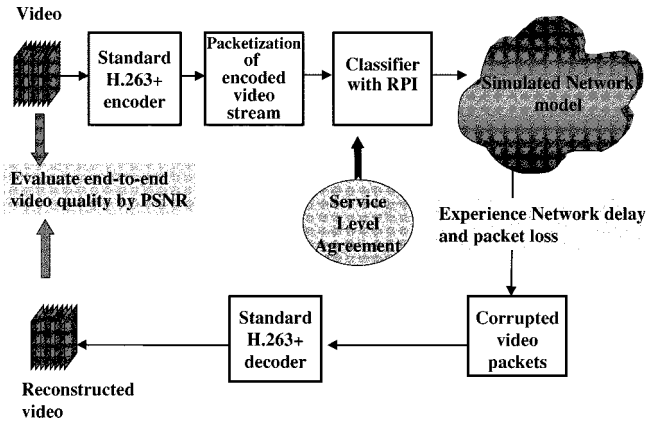


Fig. 10. Overall simulation diagram for H.263+ streaming video over a simulated DiffServ network.

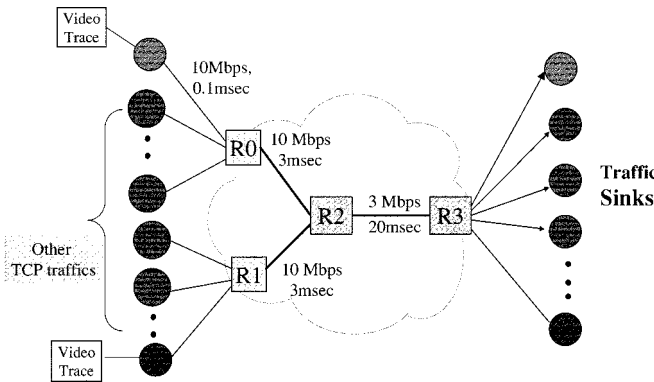
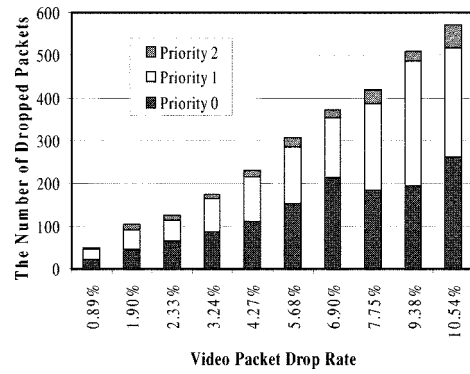


Fig. 11. Network topology model used to generate the DiffServ networks in *ns* simulation.

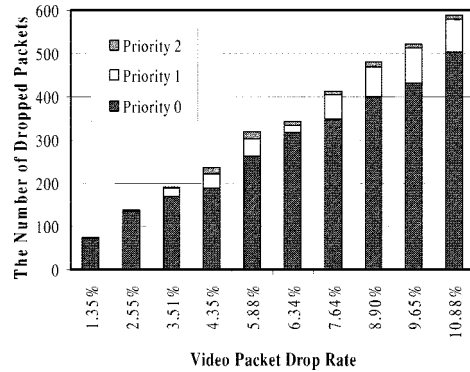
VI. EXPERIMENTAL RESULTS

In this section, we will demonstrate the performance of the proposed system consisting of video packet categorization through RPI, adaptive packet forwarding, and QoS mapping between video applications and the DiffServ network. Three scenarios are considered as mentioned in Section V. One is the single queue system that is a slight modification of the current Internet while the other two cases are MQ systems with RLI only and RLI/RDI together. The overall simulation setup is illustrated in Fig. 10. The test video trace with RLI is transmitted employing UDP over the *ns*-simulated DiffServ network, as shown in Fig. 11. The nodes R0–R3 are DiffServ-enabled with several network DS levels by using RIO and WFQ. While the video flow competes with other TCP flows, an error resilient decoding is applied to the corrupted H.263+ stream. The network is conditioned by different packet drops ranging from 0% to about 10%, which is controlled by the number of best-effort TCP flows and the setting of different target rates in AF queues.

Before going into details of performance evaluation, it is worthwhile to point out that the matching of relative prioritized video packets and network adjustment is dynamic in nature, while the traffic condition agreement in the SLA can only provide the guideline in a static manner. Furthermore, the simulation topology can effect the performance of the proposed algorithm. There are several commonly used topologies such as



(a)



(b)

Fig. 12. Comparison of packet losses under the single queue scenario (a) without DiffServ and (b) with DiffServ by RLI, where the total packet numbers and the average byte are (2675, 205.8), (2381, 204.5), and (351, 243.2) for DS level 0, 1, and 2, respectively.

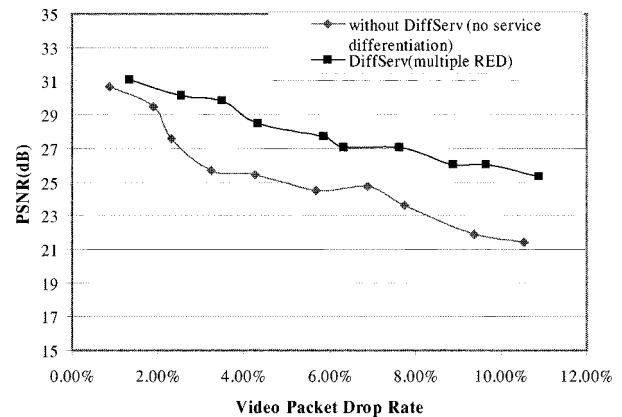


Fig. 13. Average end-to-end video performance change according to increasing packet loss-rate (by average PSNR for 300 frame Foreman sequence under the single queue scenario).

Dumbbells (a single shared link), tree-branch (merging traffics toward sink), chain (a mesh-type with other traversing interfering traffics), and so on. The scalability of PHB enables the effect of various network topologies to be minimized especially in relative service differentiation such as AS in which we are interested. Then, we choose the tree-branch type to compare the performance. We find it sufficient to use the interaction simulation setup given in Figs. 10 and 11 to demonstrate the source-network interaction in the DiffServ scenario shown in Figs. 1 and 2.

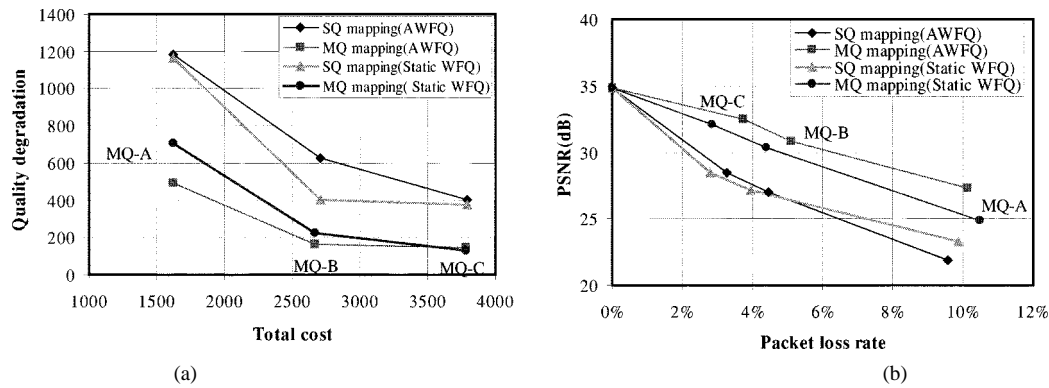


Fig. 14. Performance of the static WFQ and adaptive WFQ with RIO/RED under the MQ scenario for the Foreman test sequence: (a) quality degradation versus cost (b) end-to-end video quality versus network packet loss-rates in SQ mapping cases and corresponding MQ mapping cases.

A. Single Queue System with RLI

Each video packet with RLI only is mapped to one of three different network DS levels. Three different losses are provided through the multiple RED queuing (i.e., having three drop preferences). The price per packet is set according to the provided packet loss-rate in this scenario. The end-to-end video quality is measured in terms of PSNR. We compare results of DiffServ case to that of RED queue only (i.e., without DiffServ). Experimental results for various packet loss-rates are shown by the associated dropped packet numbers of Fig. 12. Note that packets with higher RLI results in less packet losses, which enhances the performance. The gain of the proposed mechanism with RLI in terms of end-to-end video quality is also shown in Fig. 13. As expected, the loss differentiation shows clear gain in terms of PSNR under the same budget. It also shows that DiffServ case gets better visual quality and graceful degradation over various total loss-rates corresponding to total spent cost in single class queue case. Even though the DiffServ case experiences slightly higher packet loss-rate or spends less total budget, it can perform better. This result is well matched with the relevant results of unequal error protection area.

B. Multiple Queue System with RLI

The evaluation of the multiple DS level mapping case with RLI is performed. To get a fair comparison, we injected two video traces (from “Foreman”) simultaneously into the network simulator and assigned one trace to the fixed SQ mapping (i.e., all k categories into a q DS level) and the other trace to the corresponding effective MQ mapping with the same total budget. Five DS levels, as depicted in Fig. 9(a), are provided by three class queues in each router R0–R3. The AF_1 queue with RIO processes packets of $q = 3$ and $q = 4$, the AF_2 queue with RIO handles $q = 1$ and $q = 2$, and the BE queue by RED deals with $q = 0$ traffic. First, weighting factors of static WFQ case for AF_1 , AF_2 , and BE are set to be 4, 2, and 1, respectively. For adaptive WFQ, related differentiation ratios δ_i among class queues are also 4, 2, and 1. The guidance MQ mapping sets, MQ-A, MQ-B, and MQ-C in Fig. 9(b), are used in comparison to the corresponding SQ cases. Also, the effect of proposed adaptive packet forwarding (i.e., adaptive WFQ) is also evaluated by the comparison with static WFQ. The results are shown in Table II and Fig. 14.

TABLE II
PACKET LOSS-RATE OF EACH k CATEGORY UNDER DIFFERENT PACKET FORWARDING MECHANISM. EACH COMPONENT IN [] IS THE PACKET LOSS-RATE (%) OF EACH k CATEGORY AND SAME UNDERLINED CATEGORIES ARE MAPPED TO THE SAME DS LEVEL

Mapping Set	Static WFQ	Adaptive WFQ
SQ ($q=1$)	[7.4, 11.9, 11.8, 12.5, 12.1, 6.9, 5.8, 8, 11.1, 6.9]	[9.4, 9.7, 10.9, 10.2, 11.2, 9.2, 10.7, 12.1, 11.9, 5]
MQ-A	[12.4, 13.6, 12.6, 12.9, 12.4, 7.7, 1.5, 2.3, 4.2, 1.3]	[17.5, 9.3, 11.8, 10.6, 11.2, 0.3, 0.4, 0.5, 0.2, 0]
SQ ($q=2$)	[3.7, 3.3, 2.5, 1.2, 1.1, 2.3, 4.0, 4.6, 3.7, 2.6]	[4.0, 3.4, 3.8, 2.5, 3.7, 3.4, 2.6, 4.6, 2.8, 3.9]
MQ-B	[7.1, 1.7, 1.2, 2.1, 1.7, 0.8, 2.6, 2.9, 0.9, 1.3]	[10.0, 10.2, 0.9, 1.1, 0.6, 0.5, 0, 0, 0]
SQ ($q=3$)	[2.0, 2.8, 3.0, 4.5, 3.4, 1.2, 3.7, 5.1, 1.4, 3.3]	[4.5, 3.9, 4.8, 4.5, 4.5, 5.0, 5.1, 5.1, 4.7, 4.5]
MQ-C	[7.1, 1.7, 1.2, 2.1, 1.7, 2.6, 2.8, 0.9, 1.3]	[10.4, 2.0, 1.1, 1.8, 2.2, 1.5, 2.3, 2.2, 1.5, 1.2]

The measured average of experimental packet loss-rates for each DS category are shown in Table II. The resulting loss-rates show some discrepancy to the targeted ideal loss-rate proportion for each DS level (i.e., 4 : 2 : 1), which is highly dependent on the network load dynamics. Fortunately we are interested in verifying the advantage of efficient mapping first of all. Also, we want to see the performance improvement based on the proposed adaptive WFQ over static WFQ. With these in mind, let us interpret the results shown in Table II for three total budget cases. From static WFQ results, we can easily notice several failures in providing persistent loss-rate differentiation. It also shows that adaptive WFQ gives more proportional DS levels, which altogether verifies the efficiency of the proposed adaptive packet forwarding mechanism.

The results in Fig. 14 are obtained using almost the same parameters with Fig. 8 by substituting the measured packet loss-rate of each DS level, which is somewhat different from the ideally targeted rate. Fig. 14(a) shows the relationship between the quality degradation and the used total price budget. Fig. 14(b) then shows the corresponding the end-to-end video quality (i.e., PSNR) in both static WFQ and adaptive WFQ cases. The same benefit of MQ mapping is observed like SQ scenario. Note that MQ mapping cases get better PSNR values than SQ mapping cases, even though SQ cases have lower total packet loss-rates, as shown in Fig. 14(b), which tells again the DiffServ gives the benefit of visual quality through effective QoS mapping rather than content-blind single queue mapping with similar budget spending.

Also we want to point out the advantage of proposed adaptive WFQ in addition to effective QoS mapping (i.e., MQ mapping). The PSNR gains in case of adaptive WFQ for MQ mapping are 4, 3.7, and 5.42 dB than SQ mapping while the gains of static WFQ are 3.65, 3.35, and 1.6 dB than SQ mapping. The adaptive

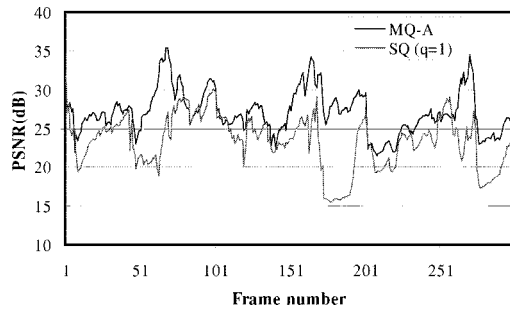


Fig. 15. End-to-end video performance comparison between MQ mapping (MQ-A) and SQ ($q = 1$) with similar cost constraint under the adaptive WFQ.



Fig. 16. Visual comparison of the reconstructed 154th frame of Fig. 15: (a) MQ-A mapping and (b) SQ ($q = 1$) cases.

WFQ gets larger service gaps among DS levels than static WFQ, which explains that the loss rate differentiation in adaptive WFQ is more proportional than that in static WFQ. Next, we consider the pattern of quality degradation gaps between MQ mappings and the corresponding SQ mappings. In relative service differentiation, we just want more persistent quality gap at the cost of same price. Fig. 14(a) shows the gaps in quality degradation between MQ and SQ mapping sets in static WFQ are somewhat irregular than that of adaptive WFQ case. This again implies that the packet loss-rate between categories mapped into the same network DS level is more persistent in adaptive case than that of static, which is obvious, as shown in Table II. In MQ-A of static WFQ case, the loss rates among DS levels show service inversion, which violates the relative service differentiation concept. Also the packet loss-rates among categories mapped into the same DS level in static WFQ have larger variation.

For the sake of completeness, PSNR advantage of multiple over single DS level mapping is shown in Fig. 15. Also, the corresponding visual effect is shown in Fig. 16. The visual quality of MQ mapping case outperforms most of the time in visually noticeable range, in case that MQ-A (average PSNR: 27.3 dB) and SQ ($q = 1$) (average PSNR: 23.3 dB) experience packet-loss 10.14% and 9.55% under the same total budget.

C. Multiple Queue System with Both RLI and RDI

All conditions are the same as previous Section VI-B, except for the restriction on the multiple DS level mapping. The case (A) of Fig. 9(a) represents more delay-stringent conversational application. It has higher RDI and should be assigned only to AF_2 and/or AF_1 . The delay-tolerant case (B) has lower RDI demand, as shown in Fig. 9(a), may be mapped to rather inexpensive queues such as AF_2 and/or BE . The resulting mapping

TABLE III
PERFORMANCE COMPARISON OF DELAY-SENSITIVE APPLICATIONS (CASE A)
AND DELAY-TOLERANT APPLICATIONS (CASE B)

Performance Parameters	case A		case B	
	SQ mapping ($q = 2$)	MQ-B'	SQ mapping ($q = 1$)	MQ-A'
Spent Total Budget in Fig. 8	2709	2667	1625	1521
Avg./Var. PSNR (dB)	26.2/5.17	30.7/2.54	24.9/4.91	28.7/2.61
Avg. Total Loss-rate (%)	3.39	3.47	4.96	5.28
Avg./Var. Delay (ms)	98.4/21.60	89.6/20.45	103.1/18.78	121.7/30.72

are denoted by MQ-B' and MQ-A' for Case (A) and (B), respectively. Case (A) shows the comparison of SQ ($q = 2$) versus MQ-B' (1, 1, 2, 2, 3, 3, 4, 4, 4, 4) while case (B) does that of SQ ($q = 1$) versus MQ-A' (0, 1, 1, 1, 1, 2, 2, 2, 2, 2). Note that they are using different total price budget.

We expect relatively delay-sensitive case (A) gets better delay/jitter performance than corresponding SQ mapping set. Delay-tolerant case (B) gets better visual quality than SQ mapping at the cost of worse delay/jitter. Experimental results are shown in Table III. In case (A) of Table III, MQ mapping gets better PSNR and better delay/jitter through RLI/RDI-aware QoS mapping. Perceptual video quality comparison between MQ and SQ mappings show similar behavior like MQ with RLI only scenario. In case (B), MQ mapping has similar benefit in average PSNR while gets larger delay/jitters than SQ. But it is tolerable to this application.

VII. CONCLUSION AND FUTURE WORK

Focusing on the relative differentiated version of IP DiffServ model, we have presented a futuristic QoS mapping framework. RPI-based video categorization, adaptive packet forwarding mechanism, and effective QoS mapping under a given cost constraint is proposed. The RPI plays a good bridging role in enabling the network to be content-aware and provides better end-to-end video quality. The proposed adaptive packet forwarding mechanism provides more persistent network DS levels regardless of network load fluctuation. Practical guidelines of effective QoS mapping is also suggested based on the categorized RPI. The performance of content-aware differentiation was demonstrated by extensive experimental results to justify the advantage of the proposed QoS mapping mechanism.

A couple of issues should be elaborated on further and have been under our investigation. First, RDI association should be extended to include more characteristics within a video stream. A proper combination of loss-rate/delay differentiation will provide a more comprehensive prioritization of the media stream in both intra-media and inter-media sense. Thus, by differentiating each packet with different (albeit varying) loss-rate and delay demands, more enhanced content-aware forwarding might be feasible with the help of upcoming intelligent DiffServ-aware applications. Second, if the end-system adjustment with feedback as a closed-loop is combined with the proposed open-loop QoS mapping, we can expect more persistent QoS mapping. This open/closed-loop idea for QoS mapping control is recently discussed in [29], too. Third, for the MQ scenario, we can explore the forwarding method for proportional differentiation in more depth. Finally, it is interesting to investigate the effective pricing mechanism in distributing a flow stream into MQs.

REFERENCES

- [1] J.-C. Bolot and T. Tuletto, "Experience with rate control mechanisms for packet video in the Internet," *Comput. Commun. Rev.*, vol. 28, no. 1, Jan. 1998.
- [2] J. Kim, Y.-G. Kim, H. Song, T. Kuo, Y. Chung, and C.-C. J. Kuo, "TCP-friendly internet video streaming employing variable framerate encoding and interpolation," *IEEE Trans. Circuits Syst. Video Technol. (Special Issue on the Picture Coding Symposium '99: Recent Advances in Picture Compression)*, vol. 10, no. 7, pp. 1164–1177, Oct. 2000.
- [3] K. Nichols, S. Blake, F. Baker, and D. Black, "Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers," RFC 2474, IETF, Dec. 1998.
- [4] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," RFC 2475, IETF, Dec. 1998.
- [5] K. Nichols, V. Jacobson, and L. Zhang, "A two-bit differentiated services architecture for the Internet," RFC 2638, IETF, July 1999.
- [6] D. Clark and J. Wroclawski, "An approach to service allocation in the Internet," Internet Draft, draft-clark-diff-svc-alloc-00.txt, July 1997.
- [7] I. Stoica, S. Shenker, and H. Zhang, "Core-stateless fair queuing: Achieving approximately fair bandwidth allocations in high speed networks," in *Proc. SIGCOMM*, Vancouver, BC, Canada, Sept. 1998.
- [8] I. Stoica and H. Zhang, "Providing guaranteed services without per flow management," in *Proc. SIGCOMM*, Boston, MA, Sept. 1999, pp. 81–94.
- [9] C. Dovrolis, D. Stiliadis, and P. Ramanathan, "Proportional differentiated services: Delay differentiation and packet scheduling," in *Proc. SIGCOMM*, Boston, MA, Sept. 1999.
- [10] C. Dovrolis and P. Ramanathan, "Proportional differentiated services—Part II: Loss rate differentiation and packet dropping," in *Proc. International Workshop on Quality of Service*, Pittsburgh, PA, June 2000.
- [11] J. M. Boyce, "Packet loss resilient transmission of MPEG video over the Internet," *Signal Process. Image Commun.*, vol. 15, pp. 7–24, 1999.
- [12] H. Gharavi and S. M. Alamouti, "Multipriority video transmission for third-generation wireless communication systems," *Proc. IEEE*, vol. 87, no. 10, pp. 1751–1763, 1999.
- [13] W. R. Heinzelman, M. Budagavi, and R. Talluri, "Unequal error protection of MPEG-4 compressed video," in *Proc. ICIP*, Kobe, Japan, Oct. 1999.
- [14] P. Bahl and I. Chlamtac, "H.263 based video codec for real-time visual communications over wireless radio networks," in *Proc. IEEE ICUPCR*, Oct. 1997.
- [15] Y. T. Hou, D. Wu, B. Li, T. Hamada, I. Ahmad, and H. J. Chao, "A differentiated services architecture for multimedia streaming in next generation Internet," *Comput. Netw.*, vol. 32, no. 2, pp. 185–209, Feb. 2000.
- [16] ITU-T, "Draft text of recommendation H.263 version 2('H.263+') for decision," Tech. Rep., International Telecommunication Union, Jan. 1998.
- [17] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.
- [18] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queuing algorithm," *J. Internetworking: Research and Experience*, pp. 3–26, Sept. 1990.
- [19] UCB/LBNL/VINT, "Network Simulator - ns (ver. 2)," <http://www-mash.cs.berkeley.edu/ns>, 1998.
- [20] D. Clark and W. Fang, "Explicit allocation of best effort packet delivery service," *IEEE/ACM Trans. Networking*, vol. 6, no. 4, pp. 362–373, Aug. 1998.
- [21] V. Jacobson, K. Nichols, and K. Poduri, "An expedited forwarding PHB," RFC 2598, IETF, June 1999.
- [22] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, "Assured forwarding PHB group," RFC 2597, IETF, June 1999.
- [23] J. Shin, J.-G. Kim, J. Kim, D. C. Lee, and C.-C. J. Kuo, "Aggregated QoS mapping framework for relative service differentiation-aware video streaming," in *Proc. Packet Video Workshop*, Korea, May 2001, to be published.
- [24] Y. Chang, D. Messerschmitt, and T. Carney, "Expanding network video capacity with delay cognizant video coding," in *Proc. SPIE Visual Communications and Image Processing*, San Jose, CA, Jan. 1999, pp. 783–794.
- [25] G. Reyes, A. Relbman, and S. F. Chang, "A corruption model for motion compensated video subject to bit errors," in *Proc. Packet Video Workshop*, May 1999.
- [26] J.-G. Kim, J. Kim, J. Shin, and C.-C. J. Kuo, "Coordinated packet level protection employing corruption model for robust video transmission," in *Proc. SPIE Visual Communications and Image Processing*, San Jose, CA, Jan. 2001, pp. 410–421.
- [27] C. Dovrolis and P. Ramanathan, "A case for relative differentiated services and the proportional differentiation model," *IEEE Network*, vol. 13, no. 5, pp. 26–35, Sept. 1999.
- [28] C. Dovrolis and D. Stiliadis, "Relative differentiated services in the Internet: Issues and mechanisms," in *Proc. ACM SIGMETRICS*, Atlanta, GA, May 1999.
- [29] K. Park and H. Ren, "Toward a theory of differentiated services," in *Proc. Int. Workshop QoS*, Pittsburgh, PA, June 2000, pp. 211–220.

Jitae Shin received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1986, the M.S. degree in nuclear engineering from Korea Advanced Institute of Science and Technology (KAIST), in 1988, and a second M.S. degree in electrical engineering from the University of Southern California (USC), Los Angeles, in 1998. He received the Ph.D. degree in electrical engineering from USC in May 2001.

He joined the Korea Electrical Power Corporation (KEPCO) and Korea Atomic Energy Research Institute (KAERI), where he worked as a Research Engineer from 1988 to 1996. His research interests include reliable and efficient multimedia transmission over QoS enabled network (i.e., differentiated services network) as a next generation Internet or wireless network focusing on QoS and network control/protocol issues.

JongWon Kim (S'88–M'95–SM'01) was born in Sangju, Korea, on September 4, 1964. He received the B.S., M.S., and Ph.D. degrees from Seoul National University, Seoul, Korea, in 1987, 1989, and 1994, respectively, all in control and instrumentation engineering.

From 1994 to 1999, he was with the Department of Electronics Engineering at the Kongju National University, Kongju, Korea, as an Assistant Professor. From July 1997 to the present, he has been visiting the Signal and Image Processing Institute (SIPI) of Electrical Engineering - Systems Department, the University of Southern California, Los Angeles, where he has been a Research Assistant Professor since December 1998. His current research interests are in the areas of multimedia signal processing and networking systems, where he has authored approximately 80 technical publications in international conferences and journals. Under the slogan of "reliable and flexible delivery for integrated multimedia over wired/wireless IP networks," he is focusing on the multimedia compression, processing, and delivery through Internet and wireless networks.

Dr. Kim is a member of ACM and SPIE.

C.-C. Jay Kuo (S'83–M'86–SM'92–F'99) received the B.S. degree from the National Taiwan University, Taipei, in 1980, and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1985 and 1987, respectively, all in electrical engineering.

He was Computational and Applied Mathematics (CAM) Research Assistant Professor in the Department of Mathematics at the University of California, Los Angeles, from October 1987 to December 1988. Since January 1989, he has been with the Department of Electrical Engineering Systems and the Signal and Image Processing Institute at the University of Southern California, Los Angeles, where he currently has a joint appointment as Professor of Electrical Engineering and Mathematics. His research interests are in the areas of digital signal and image processing, audio and video coding, media communication technologies and delivery protocols, and network computing. He has authored more than 400 technical publications in international conferences and journals and graduated approximately 30 Ph.D. students.

Dr. Kuo is a member of SIAM, ACM, and a Fellow of the IEEE and SPIE. He is Editor-in-Chief for the *Journal of Visual Communication and Image Representation*, Associate Editor for *IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING* and Editor for the *Journal of Information Science and Engineering* and the *Journal of Applied Signal Processing*. He served as Associate Editor for *IEEE TRANSACTIONS ON IMAGE PROCESSING* in 1995–1998 and *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY* in 1995–1997. He received the National Science Foundation Young Investigator Award (NYI) and Presidential Faculty Fellow (PFF) Award in 1992 and 1993, respectively.