Special Issue

Dynamic QoS Mapping Control for Streaming Video in Relative Service Differentiation Networks*

JITAE SHIN, JIN-GYEONG KIM, JONGWON KIM AND C.-C. JAY KUO
Integrated Media Systems Center and Department of Electrical Engineering-Systems
University of Southern California, Los Angeles, California 90089-2564
{*jitaeshi,jingyeok,jongwon,cckuo*}*@sipi.usc.edu*

Abstract. A dynamic quality of service (QoS) mapping control scheme, which includes feedforward and feedback QoS control, is proposed for the differentiated services (DiffServ) networks in this work. To achieve reliable and consistent end-to-end video streaming with relative service differentiation, the proposed solution consists of two parts: (1) relative priority-based indexing and categorization of streaming video content at the sending end-system and (2) dynamic and aggregate QoS mapping control with packet, session, and class-based granularity level for categorized packets at the edge of the DiffServ domain, called the video gateway (VG), based upon the load variation of the network. In particular, we focus on dynamic solutions to handle QoS demand variations of continuous media applications (e.g. varying priorities from aggregated/categorized packets) and QoS supply variations of the DiffServ network (e.g. varying loss/delay due to fluctuating network loads). Thus, with the proposed dynamic QoS mapping control, video streaming with enhanced quality is demonstrated under a pricing model.

1 INTRODUCTION

Internet applications have very diverse requirements on the network service, thus making the current best-effort (BE) Internet model less than sufficient. The emerging continuous media (CM) applications demand more stringent QoS requirements than traditional TCP-based applications. Under the BE model, maintaining the end-to-end CM quality is extremely challenging due to two reasons. First, the CM stream is inherently a variable bit rate (VBR) data stream. Second, the Internet is an unpredictable time-varying channel. Emerging differentiated services (DiffServ or DS) [1, 2, 3] schemes have been extensively studied to achieve IP-QoS in recent years. It provided more consistent QoS services to meet CM application's needs than the current best-effort Internet service in a simple and scalable manner. On-going research efforts in DiffServ can be divided into two classes: absolute differentiation [4, 5] and relative differentiation [6, 7]. An absolute service differentiation scheme attempts to guarantee QoS for a set of aggregated flows regardless of background traffics. With absolute service differentiation, per-flow QoS is typically achieved through admission control. A relative service differentiation scheme tries to maintain a quality gap among DS levels (e.g. behaviour aggregates [1]) in the environment, where traffic loads of different behavior aggregates change dynamically. As the Internet evolves towards the DiffServ model and as networked CM applications become more network-aware and adaptive, relative service differentiation will be more attractive due to its simplicity and flexibility.

The DiffServ architecture can bring benefits to both end-users and ISP by providing better service quality for CM applications at the willingness of users to pay more for better services. Thus, for DiffServ-aware applications which include video streaming as a special example, the architecture design should consider interests of both end-users and ISP. That is, an end-user should get a better price/performance tradeoff for his/her DiffServ-aware application while ISP benefits in providing different charging and service policies to maximize end-users' satisfaction. In order to perform negotiation, we should measure the QoS demand of CM applications and the QoS supply of DiffServ networks in terms of predefined granularity. With a predefined granularity, service differentiation can be demanded by tagging (or marking) source units differently at the end-system. These units will be treated differently according to their tags. Furthermore, these tags can be adjusted (i.e. re-marking) dynamically according to traffic conditions and handled differently according to new tags.

Generally speaking, there are three differentiation gran-

*This research was funded in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, under the Cooperative Agreement No. EEC-9529152.

ularities. At the application side, a end-user may want to demand differentiation per session or per packet depending on application requirements. The session-based scheme matches per-flow QoS control within the access network. This DiffServ granularity has also been promoted under the name of user-based allocation [8] and user-share differentiation [9]. Differentiation can however be carried out at the packet level to enable intra-flow differentiation, i.e. the packet-based scheme, which will be promoted in this paper. For packet-based marking, individual packets are prioritized with respect to each other for classification, queuing, rate-limiting, and so on. For example, sessions of CM applications usually require further granularity for a better match with the different priority of each packet in terms of quality as employed in video streaming [10, 11]. In addition, there is class-based service differentiation associated with the DiffServ network. That is, beyond the access and the boundary networks, the DiffServ domain handles only aggregated flows of the class-based granularity, i.e. flows with the same DS levels. The class-based DiffServ scheme demands some adaptive packet forwarding mechanism [6, 7, 12] for consistent and proportional service differentiation.

Under the relative service differentiation paradigm, this work presents a solution in which streaming video senders, receivers, and a special boundary node called the video gateway (VG), located at the border of the DiffServ domain, control QoS of streaming videos interacting with one another under a cost constraint. In this method, a video application at the source assigns chunks of its content (often in the unit of packets) by certain indices according to their impacts to end-to-end QoS in terms of loss and delay. Since these indices reflect the desired service preference of one portion with respect to other portions, we call it the relative priority index (RPI), which can be further divided into two indices, i.e. the relative loss index (RLI) and the relative delay index (RDI). Then, dynamic QoS management takes place in the form of assigning an appropriate DS level to each packet at the ingress of the DiffServ domain (i.e. VG), which is called QoS mapping. Our focus is the dynamic QoS mapping control with feedforward and feedback QoS control at VG. The system can operate in two different modes, i.e. feedforward and feedback. For the feedforward mode, we perform dynamic QoS mapping between aggregate categorized CM packets and network DS levels. Feedback in terms of end-to-end quality as well as the associated delay/loss status is relayed back to VG of the corresponding sender. By adjusting packet categorization, RPI assignment, and QoS mapping based upon the feedback, the quality of streaming video can be better managed.

The rest of the paper is organized as follows. In Section 2, we briefly describe the proposed dynamic QoS mapping control composed of feedforword and feedback QoS control in VG with three-level different granularities. In Section 3, video categorization with source importance is examined and used later for QoS mapping between applications and network DS levels. The QoS mapping problem is formulated in detail and a mapping guidance derived from the ideal situation is given in Section 4. Section 5 presents dynamic QoS control through traffic conditioning and feedback reaction at VG. Various sets of performance evaluation based on computer simulation are presented in Section 6. Concluding remarks are given in Section 7.

2 Proposed QoS management architecture

In this work, we propose a QoS-based media delivery scheme with relative differentiation services as illustrated in Figure 1. The system includes feedforward QoS mapping and feedback control at VG with support of three granularities for end-systems (e.g. the sender and the receiver). The service quality parameters considered are delay, packet loss and bandwidth.

For the bandwidth management issue, the maximally allowed transmitting rate with token bucket (TB) policing for each flow can be enforced for simplicity. It is worthwhile to point out that we seek solutions that avoid explicit admission control, where application-based bandwidth allocation and rate limiting decision have to be involved, with relative service differentiation here. To achieve this, we may rely on the rate adaptation capability of end applications to some extent. Even though we have not yet include source-rate adaptation in our evaluation at the current stage, it is assumed that error resilience of CM applications will play a role in case of insufficient bandwidth for a given video session. Thus, leaving the bandwidth management issue to adaptive rate adaptation of end applications, we will focus on the proper QoS mapping and traffic control with relative differentiation. Thus, our scheme only considers parameters of delay and packet loss. They are requested by various user applications in different degrees, anticipating different levels of guarantee (or assurance) according to the price paid.

Each CM application will demand its loss rate/delay preference by marking the DS field of its packets based on RPI, which comprises both RLI and RDI. That is, CM packets are categorized by RPI and their DS bytes are marked accordingly based on discussion given in section 3. With this, the packet-level differentiation granularity is supported. Categorized CM flows are then aggregated into VG, but they are blind of other flows that share the same VG. Each packet from a CM session is undergoing the bandwidth restriction enforced by TB, especially for the peak rate.

At the ingress of a DiffServ domain, VG is responsible for QoS mapping and feedback control. As shown in Figure 1, the system has two different operating modes, i.e. feedforward and feedback. With feedforward QoS control (the default mode of operation), we perform QoS mapping between aggregate categorized CM packets and network

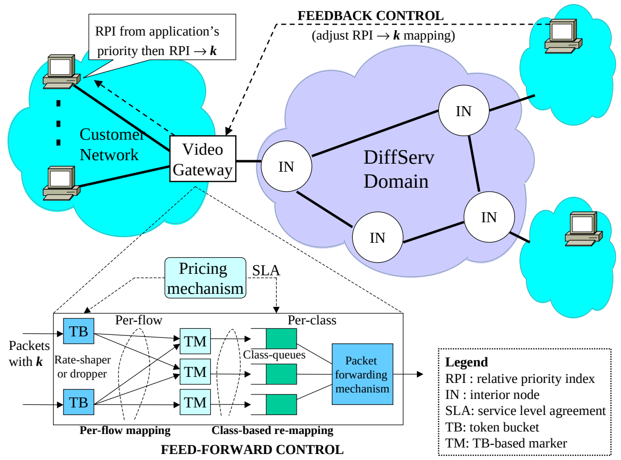


Figure 1: *The overall proposed dynamic QoS mapping control scheme.*

DS levels under the service level agreement (SLA) with this DiffServ domain. The feedback QoS control is achieved by using the feedback information from the receiving end-system. Feedback in terms of end-to-end quality as well as the associated delay/loss status is relayed back to VG of the corresponding sender. By adjusting packet categorization, RPI assignment and QoS mapping based upon the feedback, the quality of streaming video can be better managed. The feedback mechanism provides dynamic QoS mapping adjustment between connected end-systems.

Furthermore, VG is responsible for traffic management according to the traffic conditioning agreement (TCA) specified in SLA. The three granularities discussed above have to be manipulated intelligibly to increase the end-to-end service quality of DiffServ-aware video streaming. For the video streaming application, the quality performance should be performed under the cost constraint of each end-user. For VG in the DiffServ network boundary, it should coordinate conflicting demands of all end-users while attempting to maximize the aggregated quality and resource utilization under the constraint of SLA. In our approach, we has adopted a version of token bucket-based marker (TM) in Section 5.1.2 to handle traffic conditioning for aggregated flows while providing a dynamic and effective QoS mapping to realize more consistent service quality to pay-more-willingness user[1].

Then, each packet is serviced by the queuing and the scheduling mechanism to support a stable and persistent network DS level in terms of network QoS parameters within the DiffServ network domain. The extension to commonly accepted random early detection (RED) or weighted fair queuing (WFQ) combination, as proposed in [12], is utilized. The adopted scheme provides better and persistent service differentiation for aggregated flows even though the network load conditions fluctuate dynamically.

3 Delay and loss impact analysis for video source

For streaming video, differentiation granularity can vary in different levels, i.e. the session (or the flow) level, the frame level, and finally the packet level.

We currently adopted the session-based granularity to account for the delay effect of the source. Since the video application context (e.g. video-conferencing vs video e-mail) plays a crucial role in delay prioritization, we simply assign a fixed RDI to each video flow representing its delay sensitivity at the current stage. In other words, RDI is kept constant for whole one video stream so that the applicable range of DS levels is limited. However, it may be useful to vary the RDI value of frames or packets within a session/flow based on video contents and the selected coding modes (e.g., I/P/B frames) in the future. This potential merit requires further study and justification. Packet-based differentiation is adopted to account for the loss effect, where a corruption model is used to quantify the loss impact of a packet. The corruption model provides a tool used to estimate the packet loss impact to the overall received video quality. This impact can be measured by induced distortion in terms of the mean square error (MSE). For more detailed discussion of this topic, we refer to [13]. Key results in [13] are summarized below.

For a motion-compensated video coder such as ITU-T H.263+, the macroblock (MB)-based corruption can be modeled by considering effects of error concealment, temporal dependency (controlled by coding modes and motion vectors), and prediction loop filtering. By assuming that the loss impact of each MB is independent, the impact of one MB loss can be expressed as the sum of the initial error and the propagation error. By assuming that encoder knows the error concealment method at the decoder, the initial error for the MB can be estimated by differentiating reconstructed MB without loss and with loss. According to general error propagation behavior, the initial error propagates to the subsequent frames governed by effects of the temporal dependency and prediction loop filtering. The temporal dependency can be defined as the contribution of the current MB to the MB in the subsequent frame in terms of pixel area, and the prediction loop filtering effect implies the interpolation process of half-pel motion compensation and the spatial loop filtering in the decoder prediction loop. Therefore, the propagation error is attenuated by the prediction loop filtering and weighted by the temporal dependency. If the Intra-MB refresh option is used, then error propagation will be confined. Under such a scenario, we can estimate the total impact (in terms of error energy) of a MB loss in frame n by

$$\sigma^2 = \sigma_u^2 + \sum_{m=1}^{M} \sum_{j=1}^{N} w_{n,j}(m,j)\sigma_v^2(m,j), \tag{1}$$

where σ_u^2 is the initial error and its value depends on

[1] Combined with TB per flow, this traffic conditioning of aggregated flows can be modified to enable the session-based differentiation, which is however not fully investigated in this work yet.

the underlying error concealment scheme, $w_{n,j}(m,j)$ and $\sigma_v^2(m,j)$ stand for the temporal dependency weight and the propagating error impact to the jth MB (among N MBs) of the mth frame (among M subsequent frames), respectively. In addition,

$$\sigma_v^2(m,j) = \sigma_u^2/(1+\gamma_{m,j}),$$

where $\gamma_{m,j}$ is the parameter named $decaying factor$ that is governed by strength of prediction loop filtering and the frequency characteristic of the initial error. To demonstrate the accuracy of the corruption model, we compare the estimated distortion and the actual measurement in Figure 2 in terms of the MSE value. The test video is the 'Foreman' sequence of the common intermediate format (CIF) resolution coded with H.263+, where the 5th packet is lost among every group of 18 packets per frame and each packet consists of one group of blocks (GOB). As shown in the figure, there two curves are very close to each other.

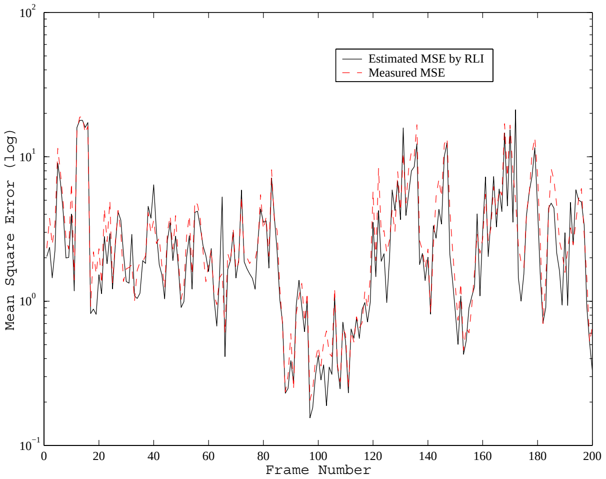


Figure 2: *Comparison of the estimated distortion and the actually measured distortion.*

Then, we can determine the RLI value (in the unit of dB) of each packet by log-scaling the estimated MSE obtained from the corruption model, i.e.

$$\text{RLI} = 10 \cdot \log(\text{MSE}).$$

It is interesting to observe the RLI distribution for several test H.263+ sequences. As shown in Figure 3, the RLI distribution varies according to scene characteristics.

Packets with different RLI values should be clustered and assigned to a finite number set of category, which is required to be fit into the finite DS field of DiffServ DS byte. In our approach, a simple uniform quantization of RLI is adopted for this. In VG, packets will be mapped from their categories to even more limited network DS levels. Here each packet is represented by source category k (among a total of K categories). After categorization, all packets in category k may be represented by their average RLI value $\overline{RLI_k}$.

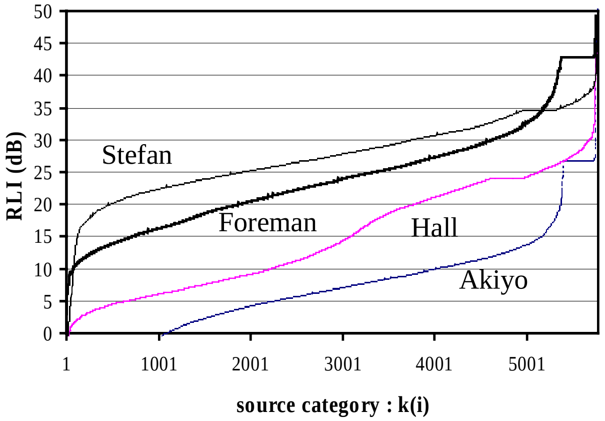


Figure 3: *Cumulative RLI distributions for several H.263+ sequences.*

To summarize, we differentiate a video source down to the packet level, and focus on relative priorities within a video session. The RPI assignment should exhibit some clustering behavior in the the RLI/RDI space so that it can keep its distinction when being mapped to the category (i.e. DS field). Besides, the RPI assignment for each packet should reflect the influence of each packet to the receiving end-to-end quality. Note that, in our approach, the RPI assignment task is isolated from network adaptation, where knowledge on the network status and competing sessions is required.

4 QoS mapping control: problem formulation and solution

To map a loss-rate/delay differentiated video packet represented by RLI/RDI to different service classes represented by DS level q (among total Q DS levels) of a DiffServ network is the key step in our proposed method. In this section, we formulate the QoS mapping problem and then derive its solution under an ideal setting. This solution can be used as a guidance to develop a dynamic QoS mapping control mechanism in Section 5.

4.1 Problem formulation

A typical QoS mapping example is illustrated in Figure 4, where there are two video application requests at the source side denoted by (A) and (B). Based on discussion in the previous section, the RDI values of these two applications are different but stay constant within each application. In contrast, RLI values can vary from one packet to the other.

To achieve service differentiation, RDI is used to select a class-queue range to support the desired delay requirement while RLI is used to decide drop precedence in each class-queue or choose one drop DS level among several

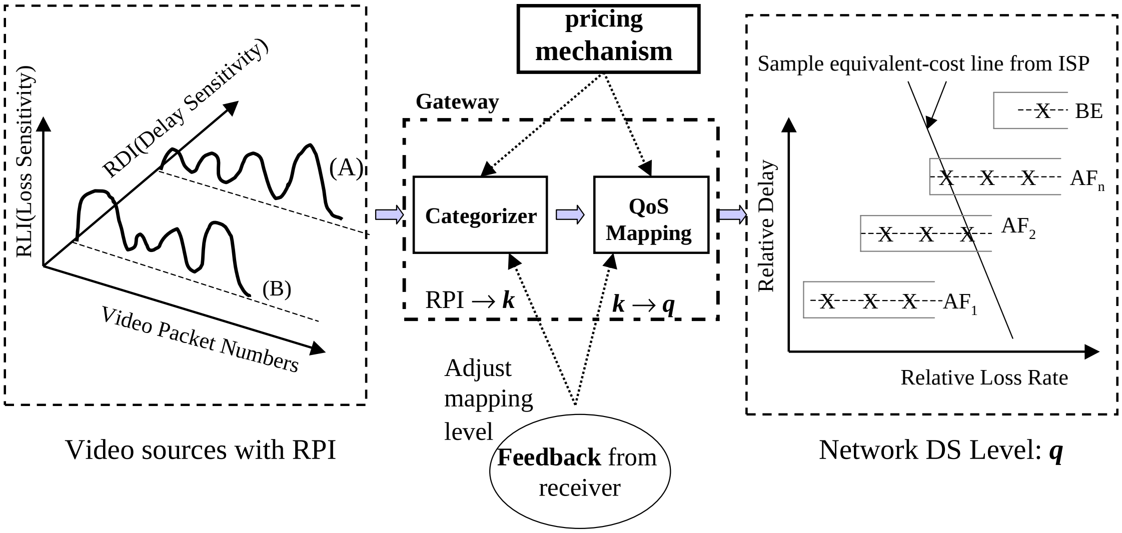


Figure 4: *The QoS mapping control for categorized video packets.*

class-queues. It is relatively easy to take RDI into account in developing a QoS mapping scheme. That is, for each flow f, $RDI^{(f)}$ limits the allowed range of $[q_{min}, q_{max}]$ to which packets of category k can be mapped without violating the delay requirement.

Then, we can address the QoS mapping problem by considering the RLI factor only as follows. It is assumed that each packet i of flow f assigned to a certain network DS level q will get an average packet loss-rate $\bar{l}_q$ by paying unit price p_q. Given the acceptable total cost $P^{(f)}$, the effort to achieve the best end-to-end quality for flow f can be formulated by minimizing the generalized quality degradation $QD^{(f)}$,

$$\min_{\vec{q_i}} QD^{(f)} = \min_{\vec{q_i}} \left(\sum_{i=1}^{N^{(f)}} RLI_i^{(f)} \cdot \bar{l}_{q(i)} \right) \quad (2)$$

$$\text{subject to} \sum_{i=1}^{N^{(f)}} p_{q(i)} \leq P^{(f)}$$

where the total number of packets of flow f is $N^{(f)}$. The QoS mapping can be written as

$$\vec{q_i} = \{q(1), q(2), \ldots, q(N^{(f)})\},$$

where $q(i)$ is the DS level to which the ith packet is mapped.

If we fix the mapping decision for all packets clustered by a finite set of categories $k = 0, \cdots, K-1$, equation (2) can be simplified to

$$\min_{\vec{q_k}} QD^{(f)} = \min_{\vec{q_k}} \left(\sum_{k=0}^{K-1} \overline{RLI_k^{(f)}} \cdot \bar{l}_{q(k)} \cdot n_k^{(f)} \right) \quad (3)$$

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

$$\text{for} \sum_{k=0}^{K-1} n_k^{(f)} = N^{(f)}.$$

where $n_k^{(f)}$ is the number of packets of category k in flow f and the QoS mapping can be written as

$$\vec{q_k} = \{q(0), q(2), \ldots, q(K-1)\}.$$

Note that the quality degradation (QD) in equation (3) is an expected value, and the loss effect of a packet belonging to category k is represented by the average value $\overline{RLI_k}$ as well.

It is worthwhile to point out that the traffic conditioning agreement specified in SLA between an application and the DiffServ domain should also be considered. Thus, the problem formulated above has to be constrained further by the traffic conditioning agreement in reality.

4.2 Ideal QoS mapping solution

Under the assumption that proportional differentiation is maintained for different DS levels of a DiffServ network throughout time, we will derive an optimal solution in this section. Delay can be proportionally differentiated according to the DS level with some proprietary well-configured scheduling [6, 7] or adaptive WFQ (weighted fair queueing) [14]. Generally speaking, the average packet loss-rate per DS level (denoted by $\bar{l}_q$) should decrease as the DS level (denoted by q) increases. Even though the loss-rate and the DS level may have various relationships in real situations, a reasonable approximation is that the increase of loss-rate l_q is inversely proportional to that of DS level q as shown in figure 5(b). Furthermore, the unit price p_q is assumed to be proportional to the DS level q. The solution to this idealized situation can be used as a guidance to determine a dynamic solution for practical situations.

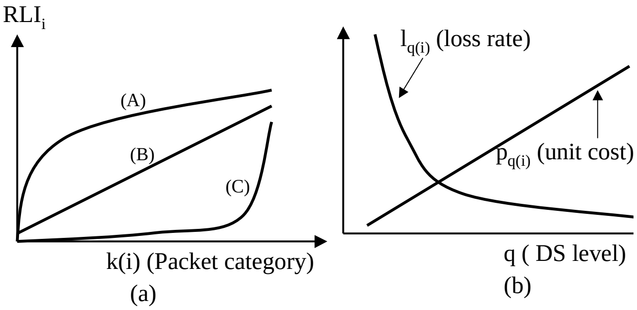


Figure 5: *(a) Several RLI distribution patterns to characterize categorized RLI distributions, and (b) typical packet loss-rate l_q and unit cost p_q relationship with respect to DS level q.*

The optimal (or effective) mapping solution $\vec{q_i^*} = \{q^*(1), q^*(2), \ldots, q^*(N^{(f)})\}$ to Equation (2) can be determined as follows. This constrained optimization problem can be solved by finding the QOS mapping $\vec{q_i^*}$ that minimizes the Lagrangian formula

$$J_i(\lambda) = RLI_i \cdot \bar{l}_q(i) + \lambda \cdot p_{q(i)} \,. \qquad (4)$$

By further assuming the video source RLI distribution is available, one can get the solution to Equation (2). For this, we approximate the RPI distribution of sequences shown in Figure 3 as one of three typical patterns of Figure 5(a): $C_1 k(i)^2$ for 'Akiyo', $C_2 k(i)$ for 'Hall' and $C_3\sqrt{k(i)}$ for 'Stefan' and 'Foreman'. With loss-rate function $\bar{l}_{q(i)} = L/q(i)$ and unit price function $P_l \cdot q(i)$ as shown in Figure 5(b), the Lagrangian formula of Equation (4) becomes

$$J_i(\lambda) = C_2 \; k(i) \cdot \frac{L}{q(i)} + \lambda \cdot (P_l \cdot q(i)), \qquad (5)$$

for a RLI pattern of the form $C_2 k(i)$.

Then, by searching around the convex hull from the graph of QD vs. cost, we can get the optimal mapping solution. We can even get a closed-form solution by using $\frac{\partial J_i(\lambda)}{\partial q(i)}$ and the constraint with the equal sign in Equation (2). The resulting closed-form solution can be written as

$$\begin{aligned} \lambda &= C_2 \cdot L \cdot P_l / P^2 \cdot [\sum_i \sqrt{k(i)}]^2 \\ q(i) &= P/P_l \cdot \sqrt{k(i)}/[\sum_i \sqrt{k(i)}]^2. \qquad (6) \end{aligned}$$

If RLI_i has other distribution patterns such as $C_1 k(i)^2$ and $C_3\sqrt{k(i)}$, then the corresponding optimal QoS mapping is of forms that are proportional to $k(i)$ and $\sqrt[4]{k(i)}$, respectively.

The derived closed-form solutions match well with actual numeric solutions for three test sequences, i.e. 'Akiyo', 'Hall' and 'Stefan', as depicted in Figure 6. Note that these mapping functions resemble those shown in Figure 3, which are represented by RLI patterns of $C_1 k(i)^2$, $C_2 k(i)$ and $C_3\sqrt{k(i)}$, respectively.

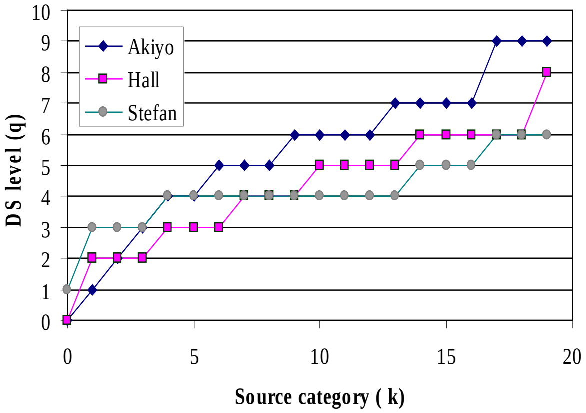


Figure 6: *Effective QoS mappings for different RLI source distributions.*

5 Dynamic QoS mapping control for aggregated flows

Based on the QoS mapping guidance functions derived in Section 4.2, let us examine the dynamic case with time-varying network service quality. In real situations, simplifying assumptions stated earlier have to be relaxed. First, the full distribution of RPI is not available, and VG should try to do the best mapping of a flow just with the observed RPI's or categorized RPI's. Next, the ideal proportional DS level provisioning of DiffServ domain may not exist, and we should be satisfied with reasonably consistent proportional differentiation. Furthermore, we have to limit the resource allocation based on the traffic conditioning agreement specified in SLA with this DiffServ domain. Thus, some traffic conditioning tool is required to shape/police non-conforming users and their aggregation. The dynamic QoS mapping control is required to address these issues. As shown in Figure 1, dynamic QoS mapping is done for aggregated flows at VG in feedforward/feedback mode.

5.1 Dynamic feedforward QoS control

The overall feedforward QoS control algorithm at VG is summarized in Figure 7.

Packets generation with RLI/RDI at the sending end-system

1. Assign $RDI^{(f)}$ to flow f for its delay sensitivity. /* $RDI^{(f)}$ restricts the q range */
2. Assign RLI_i to each packet i based on video-content as described in section 3.
3. Categorize RPI into k category by discretizing RLI_i of packets within a flow and get $\overline{RLI_k^{(f)}}$.

Feedforward QoS control at VG

Session/Packet-based

1. Each user (with one or several sessions) gets assigned TB with specified parameters.
/* TB enforces each user to keep the rate contract between the user and VG */
2. Down-grade non-conformant packets to the best-effort class marking in the DS level.
3. Perform the effective per-flow QoS mapping with the guidance solution given in section 4.2.
/* mapping of $k \to q$ to meet equation (3) with constraints */

Class(or DS level)-based

1. Class-based TB (i.e., inter-connected trTCM in section 5.1.2) enforces all incoming traffics according to traffic conditioning agreement of SLA.
/* Re-mark incoming packets to prevent service inversion among DS levels due to class-load unbalance. */

Figure 7: *Dynamic feedforward QoS control algorithm.*

Receiving end-system

1. The receiver requests service with a tolerable range
/* e.g., 400msec $\pm 10\%$ $\leq$ average delay $\leq$ 500msec $\pm 10\%$ variation, and/or 3% $\leq$ average loss rate $\leq$ 5% */
2. Send the feedback information such as the average delay and the packet loss rate back to VG periodically
/* When the average delay/loss rate deviates from desired ranges, the necessary QoS change is requested. */

VG reaction to feedback QoS control

1. VG monitors and compares the feedback with the requested service range.
2. VG changes the mapping of $k \to q$ if the desired service is not met.

Figure 8: *Dynamic feedback QoS control algorithm.*

5.1.1 Session/Packet-based Feedforward QoS control

A video packet is categorized into k category by RLI/RDI at the video sender, without knowing about other competing flows. An assigned RDI may limit the mapping range of q as discussed earlier. Then, how to perform practical and effective QoS mapping based on Equation (3) is the key issue. First, from Figures 5 and 6, we can obtain a practical mapping guidance for $k \rightarrow q$ mapping by considering session/packet-based granularity. Also within the customer network, the traffic conditioning per end-user is handled by VG through assigning TB with different parameters (e.g. the token rate, the bucket size, etc.). The total sending rate is restricted by the pre-assigned total bandwidth usage per end-user. In addition, the total cost constraint may be applied per DS level, which altogether covers per-user QoS control in session/packet-based granularity. Note that the proposed dynamic QoS mapping scheme helps end-users to minimize quality distortion under the total price paid.

5.1.2 Traffic conditioning for class-based Granularity

The dynamic mapping for class-based granularity is performed by re-marking through TM. By degrading the $k \rightarrow q$ mapping level when a flow or a class traffic volume exceeds the allowed bandwidth, we can regulate the traffic while trying to meet per-flow efficiency as detailed below.

The access network operator can contract with the Internet service provider (ISP) for a traffic conditioning agreement in SLA. For example, it can be specified for each different class-queue with tuning parameters like the committed burst size (CBS), the committed information rate (CIR), the peak burst size (PBS), the peak information rate (PIR), etc. However, since end-users are blind to other aggregate flows coming into VG, VG needs class-based traffic conditioning on all aggregate flows.

To achieve the class-based policing for aggregated flows, a two-rate three-color-marker (trTCM) [15] can be adopted for each class queue. These trTCMs are then interconnected. Each trTCM is composed of two token buckets (C and P for CBS and PBS, respectively) to meter incoming packets, and mark them by one of the three colors (i.e. green, yellow and red). Let us consider an example where each assured forwarding (AF) class queue has three drop precedences for three colors in the loss priority order:

- green - corresponding to DS level AF_{x1}[2] ;
- yellow - corresponding to DS level AF_{x2} ;
- red - corresponding to DS level AF_{x3}.

In the color-aware mode, packets arrive with some pre-assigned color. The initial assignment is respected, and the drop precedence can only be increased. Token buckets P and C are initialized to PBS and CBS in the beginning. Token bytes B_C (for the C token bucket) and B_P (for the P token bucket) are incremented by CIR and PIR up to their upper bounds, CBS and PBS, respectively.

If a packet of size B has been colored as red, it will remain red. A yellow packet is re-marked as red if there is no token available in both C and P buckets. Otherwise, it is allowed to remain yellow and B_p becomes $B_P - B$ if there is a token in the P bucket. Finally, a green packet becomes yellow with $B_P \leftarrow B_P - B$ if a token is available only in the P bucket. Or it becomes red if there is no token available in either buckets. Otherwise, a green packet remains green with $B_C \leftarrow B_C - B$ and $B_P \leftarrow B_P - B$.

To coordinate the inter-connected function between trTCMs, we need to measure the aggregated ingress rate into each class. If the aggregated rate reaches the maximum assigned rate C_j for each class queue with a typical WFQ scheduler as shown in figure 9, we hand over the incoming packet randomly. The aggregated ingress rate is calculated by using the exponentially weighted moving averaging to smooth the estimation of fluctuation noise. To avoid possible sensitivity due to the packet length distribution, a dynamic weighting as a function of inter-packet arrival time is utilized. Thus, the average rate is measured via

$$avg\_rate = (1 - e^{inter\_pkt\_time/T}) \cdot current\_rate$$
$$+ e^{inter\_pkt\_time/T} \cdot avg\_rate,$$

where T is a constant. We refer to [4] for more details regarding this choice.

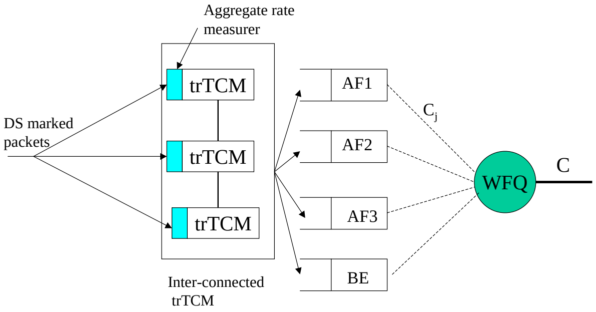


Figure 9: *Inter-connected trTCM with aggregated rate measurements.*

With the above schemes in place, VG performs traffic conditioning at boundary nodes in a DiffServ architecture [1] to prevent service inversion, where a lower class flow happens to receive sometimes better service than a higher class one. If the aggregated rate for the queue of class j is greater than C_j, the incoming packets into this queue is re-directed by trTCM to lower the class queue with the probability of $(avg\_rate - C_j)/avg\_rate$ and so on. This class-based traffic conditioning in VG enables the DS domain to provide more reliable differentiated services.

[2] DS levels $BE, AF_{33}, AF_{32}, AF_{31}, AF_{23}, AF_{22}, AF_{21}, AF_{13}, AF_{12}, AF_{11}$ are equivalent to numbers of $q = 0 - 9$ in this work.

5.2 Dynamic feedback QoS control

Feedback control is performed as per-flow basis by VG (or end-systems) to adjust the QoS mapping level of category k to DS level q(or RPI to k) for delay quality upon the feedback of the video receiver. The receiver sends a report of delay/packet loss to the sender whenever necessary and request for mapping adjustment when the received quality is not satisfactory. This feedback control enables the fine-tuning of rather coarse feedforward QoS control. The video receiver may even ask for the decrease of QoS mapping level q when it feels the current received quality is more than needed and wants to reduce the charging bill. This feedback mechanism enables the whole QoS control to be better adjusted while the operating DS level remains stable by sustaining the end-to-end quality within an acceptable range. The algorithm for the feedback QoS control is summarized in Figure 8.

6 Experimental results

The proposed dynamic QoS mapping control scheme for aggregated streaming video flows is evaluated by simulations in several steps. For video streaming, an error-resilient version of the ITU-T H.263+ stream is utilized and decoded by a robust video decoder.

6.1 Packet/session based QoS mapping for the ideal case

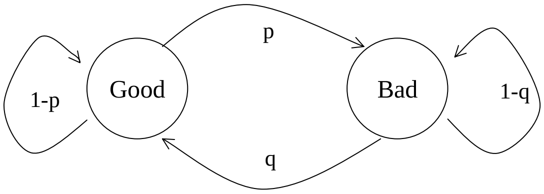


Figure 10: *Markov model for loss-rate modeling for ideal case.*

First, the proposed RPI performance is evaluated with H.263+ video under the ideal network condition as shown in Figure 5(b). The result shows the effectiveness of packet/session-based differentiation over several typical sequences. The idealized packet loss-rate for each DS level is simulated by the two-state Markov model known as the Gilbert model with transition probabilities from ‘Bad (i.e., packet loss)’ to ‘Good’ as $q = 0.9$ and $p = \frac{\bar{l}_{q(k)} \cdot q}{1-\bar{l}_{q(k)}}$, which gives a little bit burst packet loss as shown in Figure 10.

Figure 11 gives the performance comparison between RPI-aware and RPI-blind mappings under the total cost constraint. The loss-rate and the cost curves of each DS level are $p_{q(i)}$ and $\bar{l}_{q(i)}$ as shown in Figure 5(b). It is interesting to note that the gain in PSNR varies somewhat according to the nature of the sequence. For ‘Akiyo’, the gain is smaller than other sequences, which is partly due to its small movement involved in the scene. For other sequences, we can observe a clear advantage of packet-based differentiation. Overall, the obtained result confirms the benefit of RPI-aware QoS mapping in the DiffServ network.

6.2 Dynamic QoS mapping control

Next, We investigate the VG functionality in dynamic feedforward/feedback QoS control with traffic conditioning by using the *ns* (network simulator) tool [16]. The network topology shown in Figure 12 with VG having the inter-connected trTCM in figure 9 is used to generate the underlying network dynamics with several video traces. In each link of the DS domain in Figure 12, queuing with WFQ and multiple RED is applied for link sharing of scheduler and the drop precedence of queue management. By giving different relative shares (relative to the average arrival rate) for different AF classes, the DS domain in this example provides relative differentiated services through ‘capacity differentiation’ [17]. In this example, the class order from high to low is AF_1, AF_2, AF_3, and BE.

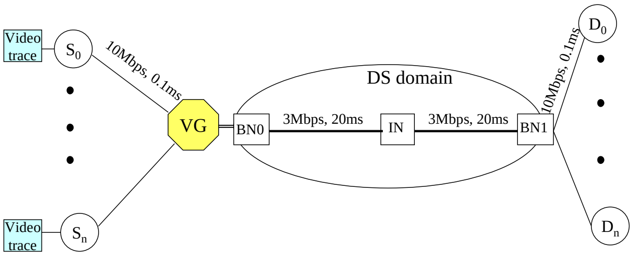


Figure 12: *The network topology model used in simulation.*

6.2.1 Feedforward QoS mapping

In the first experiment, we investigate the functionality of VG using the interconnected trTCM for dynamic feedforward QoS control of aggregated video flows. The overall end-to-end video quality is assessed by the objective PSNR measurement after video decoding. Also the network service quality is evaluated by the packet loss-rate, the throughput, and the average delay/jitter.

Table 1: *The network load condition for performance evaluation of inter-connected trTCM.*

class queue	subscription level(%)	assigned link BW by WFQ(kbps)	imposed load(kbps)
AF_1	136	1200	1632
AF_2	123	900	1104
AF_3	104	600	624
BE	N/A	300	240

To isolate the effect of VG’s functions, we do not di-

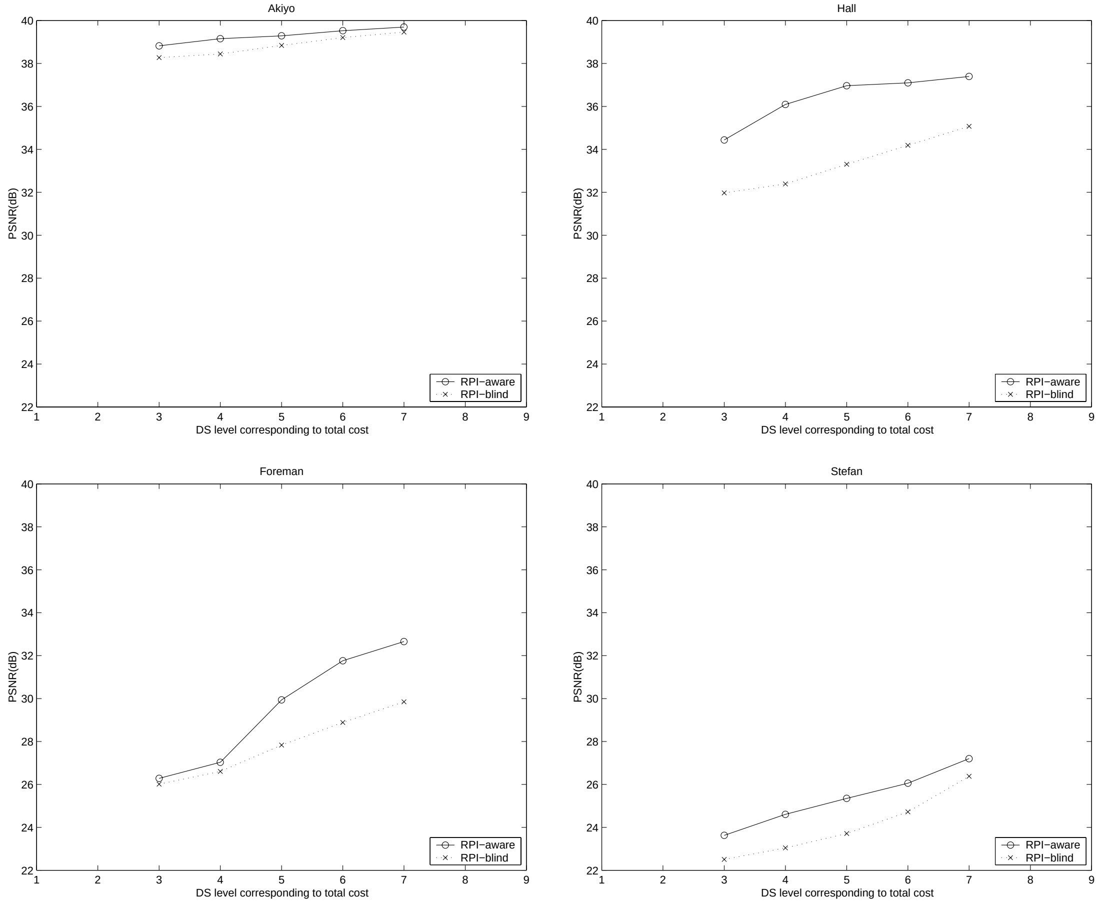


Figure 11: *Performance comparison between RPI-blind and RPI-aware mapping under the ideal loss/cost assumption of Figure 5(b). The average objective video quality (in PSNR) is compared for several sequences with different patterns as shown in Figure 3.*

rectly use the guidance QoS mapping as described in Section 4.2 in this experiment. Instead, an unbalanced video traffic load against the traffic conditioning agreement is imposed. Three 'Hall' (with an average rate per flow equal to 288kbps) and two 'Foreman' sequences (384kbps) are assigned to the AF_1 class queue. One 'Foreman' and three 'Akiyo' (240kbps) sequences are assigned to AF_2, one 'Foreman' and one 'Akiyo' are assigned to AF_3, and one 'Akiyo' to BE class queue, respectively. The load conditions of different class queues are tabulated in Table 1. Note that higher class queues are deliberately loaded heavy (compared to the assigned bandwidth portions by the scheduler) to induce service inversion. This is to demonstrate the promising regulation effect of inter-connected trTCM for dynamic network load conditions. Results in Table 2 show that inter-connected trTCM in VG is useful in the intended traffic conditioning. It meets the traffic conditioning goal of VG by ensuring that higher priority-marked packets get better service than lower-priority ones.

In the second experiment, we demonstrate the superiority of using RPI in sender's pre-marking scheme as described in Section 4.2 and evaluate the scalability of feed-

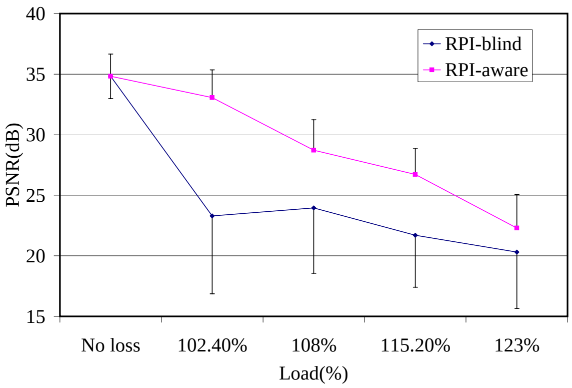


Figure 13: *Performance comparison between RPI-blind and RPI-aware mapping over differently imposed network loads (end-to-end video PSNR performance).*

Table 2: *Performance evaluation for inter-connected trTCM by comparing service quality parameters of total aggregated flows in each class queue. Note: *) from the TB enforcement of each user, **) from traffic conditioning of inter-connected trTCM.*

flows in class queue	without inter-connected trTCM				with inter-connected trTCM			
	achieved throughput (kbps)	packet loss rate(%)	delay/jitter (msec)	re-marking* rate (%)	achieved throughput (kbps)	packet loss rate(%)	delay/jitter (msec)	re-marking** rate (%)
AF_1	1223.6	28.43	142.0/8.19	2.20	1577.6	1.48	67.1/8.33	44.35
AF_2	918.7	20.59	92.3/7.24	5.44	960.0	8.79	70.3/15.70	40.91
AF_3	612.0	2.70	127.3/19.14	3.11	331.5	46.38	117.6/16.06	36.75
BE	240.6	0.41	138.0/18.00	0	125.1	49.78	138.0/17.99	0

forward QoS mapping with the increase of the number of RPI-aware flows. We compare the loss-rate for RPI-aware and RPI-blind mappings. In different runs, we simulate different numbers of RPI-aware 'Foreman' flows with one reference-to-be-compared flow (f_r) assigned to a DS level (e.g. all k to $q = AF_{23}$). The number of flows with RPI-aware QoS mapping is increased to several subscription degrees (i.e. total network loads). In each run, VG uses the interconnected trTCM as described in Section 5.1. We then select one sample flow (f_s) among RPI-aware flows and compare it with the RPI-blind flow (f_r).

Table 3: *Comparison between RPI-blind and RPI-aware mappings over different imposed loads in terms of the packet loss-rate and the throughput).*

imposed load(%)	RPI-blind mapping		RPI-aware mapping	
	loss rate (%)	throughput (kbps)	loss rate (%)	throughput (kbps)
< 100	0	384	0	384
102.4	2.16	370.79	4.98	366.7
108	5.25	357.46	9.18	349.87
115.2	8.57	340.66	13.71	332.57
123	13.2	321.77	22.3	300.34

As shown in Figure 13, the network load increases in accordance with the increasing number of RPI-aware flows for different simulations runs, and the overall PSNR for selected RPI-aware flow f_s is degraded gracefully compared to that of RPI-blind flow f_r. Also, the RPI-bind flow experiences mapping irregularity reflected in the average/standard deviation of PSNR, where the vertical bar stands for the standard deviation. The corresponding throughput and packet loss-rate are provided in Table 3.

Moreover, let us observe the loss rate of f_r and f_s at 115.2% load level from Table 3. The objective PSNR quality measure of f_s is better than that of f_r over all time even though f_s experiences a worse total packet loss rate as shown in Figure 14. This is due to the fact that f_s tends to lose relatively unimportant packets due to RPI-aware mapping.

In Figure 14, we show the loss rates of f_s for different source categories. Recall that all k source categories of f_r is mapped into $q = AF_{23}$. With this cost constraint, f_s is mapped as follows: $k = 0 \rightarrow AF_{32}$ with loss-rate 33.3%, $k = 1 \sim 5 \rightarrow AF_{31}$ with loss-rate 30.5%, $k = 6 \sim 11 \rightarrow AF_{23}$ with loss-rate 8.66%, $k = 12 \sim 17 \rightarrow AF_{22}$ with loss-rate 1.42%, $k = 18 \rightarrow AF_{21}$ with no loss, and $k = 19 \rightarrow AF_{12}$ with no loss, respectively. Thus, the total loss rate of f_s is 13.71% while that of f_r is 8.57%. The objective PSNR quality measure of f_s is however better than that of f_r over all time even though f_s experiences a worst total packet loss rate as shown in Figure 14. It is again verified by the presented snap shots of a decoded video frame as shown in Figure 15. RPI-aware mapping is a clear winner in the average or instance sense over time-varying network load conditions.

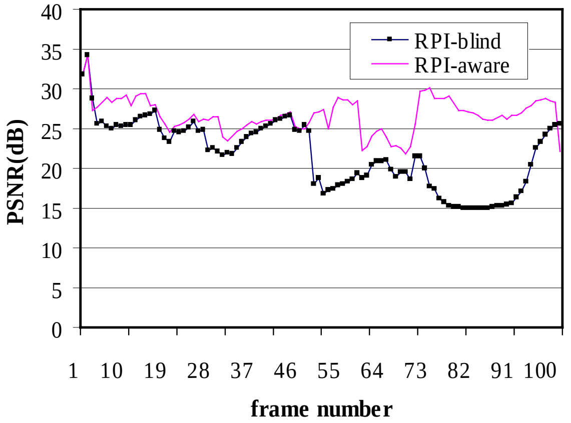
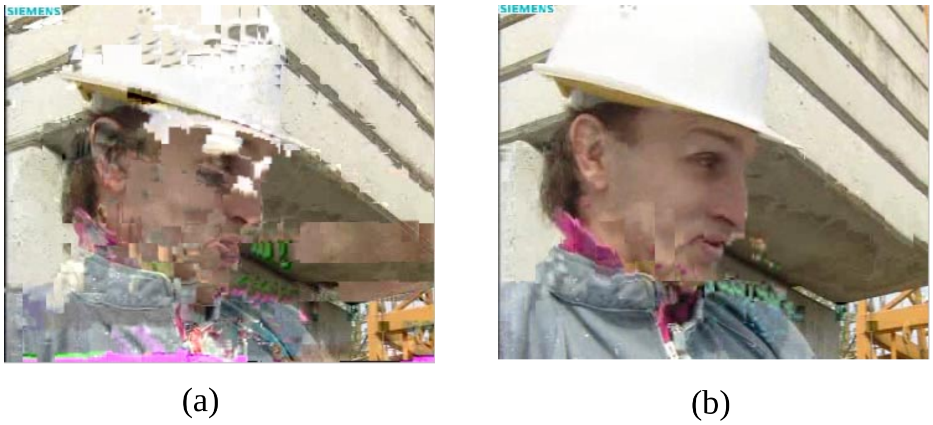


Figure 14: *PSNR comparison for RPI-blind and RPI-aware mappings for the 'Foreman' sequence at 115.2% load.*

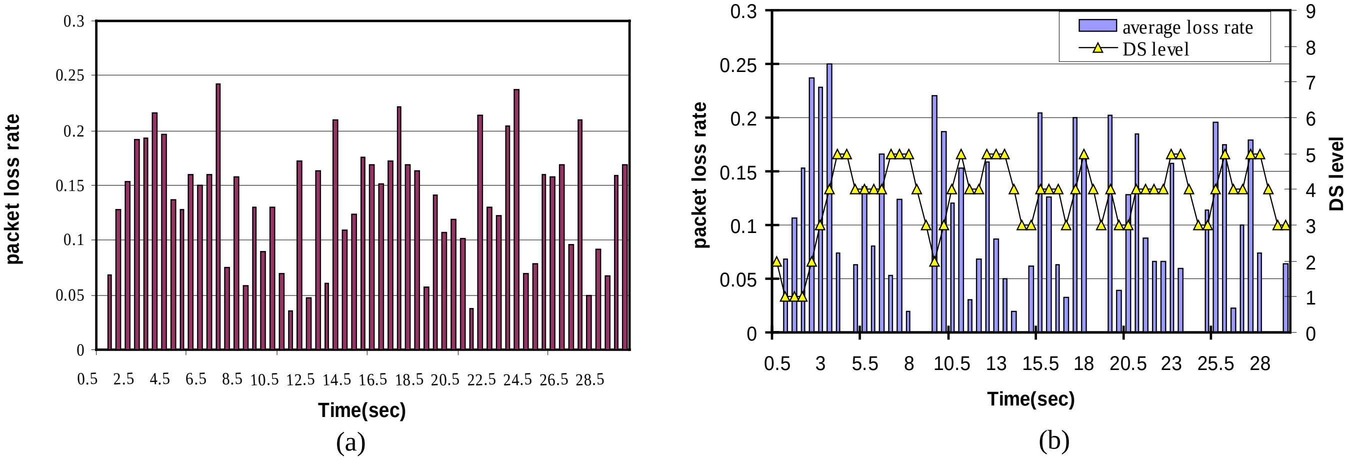
6.2.2 *Feedback QoS control*

Finally, we verify the effectiveness of the proposed feedback QoS control with RPI-aware mappings. The $k \rightarrow q$ QoS per-flow mapping at VG is adjusted based on periodic feedback from the receiver with a desired loss-rate range. For this simulation, the range is set to [5, 15%] and we expect to get the average loss rate as the middle value (i.e. 10%) of this range. Obtained results illustrate how feedback QoS control affects the average loss-rate dynami-



(a) (b)

Figure 15: *Visual quality comparison of a sample frame (i.e. the 52th frame of the Foreman sequence) at 115.2% load: (a) RPI-blind and (b) RPI-aware mapping.*



(a) (b)

Figure 16: *Illustration of the feedback control effect by comparing the average loss-rate: (a) the reference case with a video flow staying to a fixed DS level ($q = 3$) and (b) the feedback case with a video flow adjusting the DS level.*

cally. As shown in Figure 16, VG can re-mark the DS level dynamically upon receiver's feedback. The average total loss-rate of a reference case is 13.4% and the loss-rate of the feedback case is 9.5%, respectively. The time interval of QoS feedback and re-marking is 0.5 sec for both cases, and the time interval for the average loss-rate calculation in Figure 16 is the same.

7 CONCLUSION AND FUTURE WORK

Our previous work in [18, 14] showed that the QoS mapping control between categorized-packet video and the DS level can give gains in end-to-end video quality in terms of delay and loss under the total cost constraint to use Diff-Serv. However, previous results were limited to per-flow feedforward QoS mapping. In this research, we extended results to the full scope by including the three-layer granularities (packet-based, session-based, and class-based) and feedback QoS control. Extensive simulations have been run to demonstrate the superior behavior of the proposed scheme. Our focus has been the QoS mapping of unicast flows. How to generalize it to multicast flows is still an open question.

Manuscript received on June 17, 2000.

REFERENCES

[1] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W.Weiss. An architecture for differentiated services. *Internet Engineering Task Force*, RFC2475, December 1998.

[2] V. Jacobson, K. Nichols, and K. Poduri. An expedited forwarding PHB. *Internet Engineering Task Force*, RFC2598, June 1999.

[3] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski. Assured forwarding PHB group. *Internet Engineering Task Force*, RFC2597, June 1999.

[4] I. Stoica, S. Shenker, and H. Zhang. Core-stateless fair queueing: Achieving approximately fair bandwidth allocations in high speed networks. In *Proc. of ACM SIGCOMM'98*, September 1998.

[5] I. Stoica and H. Zhang. Providing guaranteed services without per flow management. In *Proc. of ACM SIGCOMM*, pages 81–94, September 1999.

[6] C. Dovrolis, D. Stiliadis, and P. Ramanathan. Proportional differentiated services: Delay differentiation and packet scheduling. In *Proc. of ACM SIGCOMM*, September 1999.

[7] C. Dovrolis and P. Ramanathan. Proportional differentiated services, part II: Loss rate differentiation and packet dropping. In *Proc. of IEEE/IFIP International Workshop on Quality of Service*, pages 52–64, June 2000.

[8] A. Banchs and R. Denda. A scalable share differentiation architecture for elastive and real-time traffic. In *Proc. of IEEE/IFIP International Workshop on Quality of Service*, pages 42–51, June 2000.

[9] Z. Wang. User-share differentiation(USD): Scalable bandwidth allocation for the Internet. In *Proc. of IFIP Conference on High Performance Networking*, Vienna, September 1998.

[10] 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. *Computer Networks*, Vol. 32, No. 2, pages 185–209, February 2000.

[11] H. R. Shao, W. Zhu, Y. Q. Zhang, and Y. Wu. User-aware object-based video multicast over the Internet. In *Proc. of Packet Video Workshop*, May 2000.

[12] J. Shin, J. Kim, D.C. Lee, and C.-C.J. Kuo. Adaptive packet forwarding for relative differentiated service and categorized packet video. In *Proc. of International Conference on Communications*, June 2001 (to be appeared).

[13] 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. Visual Communications and Image Processing*, January 2001.

[14] J. Shin, J. Kim, and C.-C. J. Kuo. Relative priority based qos interaction between video applications and differentiated service networks. In *Proc. of IEEE International Conference on Image Processing*, September 2000.

[15] J. Heinanen and R. Guerin. A two rate three color marker. *Internet Engineering Task Force*, RFC2698, September 1999.

[16] UCB/LBNL/VINT. Network simulator - ns (version 2). http://www.isi.edu/nsnam/ns, 1998.

[17] C. Dovrolis and P. Ramanathan. A case for relative differentiated services and the proportional differentiation model. *IEEE Network*, Vol. 13, No. 5, pages 26–35, September 1999.

[18] J. Shin, J. Kim, and C.-C. J. Kuo. Content-based packet video forwarding mechanism in differentiated service networks. In *Proc. of Packet Video Workshop*, Sardinia, Italy, May 2000.