Global Experiment on Dynamic Admission Control of UDP Flows

Atsushi Takahara and Takashi Shimizu†

Abstract

Video conferencing applications are causing congestion problems on the Internet. These applications mostly use UDP (User Datagram Protocol) for video and audio transmission, which unlike TCP (Transmission Control Protocol), does not have any standardized mechanisms for congestion control. Therefore, they can cause severe congestion that can significantly degrade network utilization. To overcome this problem, we introduce a new flow-aware forwarding policy to enable dynamic admission control of UDP flows and propose the MXQ (MaXimal Queuing) mechanism to achieve this policy. In a large-scale trans-Pacific field trial conducted at the Fall 2002 Internet2 Member Meeting, we tested MXQ by applying it to video conferencing traffic. The results show that its dynamic admission control could effectively control congestion in a real environment, suggesting that we can make better use of best-effort networks.

1. Introduction

With the progress in personal computer technologies and growth in the number of broadband access users, video conferencing is expected to be one of the killer applications of the Internet. Most video conferencing applications, for example, videophones based on H323 [1] or SIP (Session Initiation Protocol) use UDP (User Datagram Protocol) for the transmission of video and audio data since they require timeliness rather than reliability in data delivery. Unlike TCP (Transmission Control Protocol), UDP does not have any standardized mechanisms for congestion control, and some researchers have warned that this lack might cause severe congestion, leading to congestion collapse, in which the network is overloaded by packets that are useless to applications [2]. Thus, service providers need to have some network solutions to address the problems.

To overcome this problem, NTT Network Innovation Laboratories proposed the MXQ (MaXimal Queuing) mechanism [3] to achieve dynamic admission control of UDP flows. A number of experiments were conducted using our prototype implementation and initial experimental results show that the dynamic admission control achieved by MXQ could appropriately control the congestion.

It is usually difficult to estimate the effect of a congestion avoidance method by using simulation or an experimental network because real traffic patterns are not well simulated in such an evaluation environment, so it should be evaluated in a real network with real applications. We had the opportunity to evaluate MXQ in a real network in October 2002 as part of the joint work with the University of Illinois at Chicago and University of Southern California.

In this paper, we report the capabilities of MXQ based on the results of this large-scale trial, discuss potential congestion problems caused by UDP flows, and show the MXQ mechanism, which resolves them.

2. Congestion caused by UDP flows

The following experiment shows one typical example of such congestion and clarifies the problems that can arise.

In the experimental system (Fig. 1), two IP routers were connected by a link whose bandwidth was set to 70 Mbit/s. A packet generator generated five UDP flows with different sending rates (5, 10, 15, 20, and 25 Mbit/s), which were input to the left-hand-side IP

† NTT Network Innovation Laboratories
Yokosuka-shi, 239-0847 Japan
E-mail: shimizu.takashi@lab.ntt.co.jp

router. These flows were routed through the link and output from the right-hand-side IP router, where the received packets were measured. Each flow had a constant sending rate, and the UDP packets all had the same frame size of 1480 bytes.

Figure 2 shows the number of packets sent and received for each UDP flow during the test period of three minutes. Since the sum of the sending rates of the five UDP flows (75 Mbit/s) exceeded the bandwidth of the ATM link (70 Mbit/s), congestion occurred at the left IP router and some packets were discarded. Since the IP routers used the FIFO (First-In First Out) policy, packet loss was observed in every UDP flow, and the number of packets discarded from a given flow was purely random.

Most video conferencing applications are packet loss sensitive; exceeding some level of loss in each UDP flow triggers a substantial degradation in video and audio quality. Therefore, Fig. 2 implies that every flow will be affected by congestion, and users of applications will perceive the congestion as abnormal behavior such as frame skipping, or interruption of audio. In addition, since it is difficult for the receiving application to adequately regenerate the original video or audio from incomplete data, the partial receipt of packets may not contribute to the user's communication at all, that is, most of the packets forwarded could be useless ones. Therefore, the utiliza-

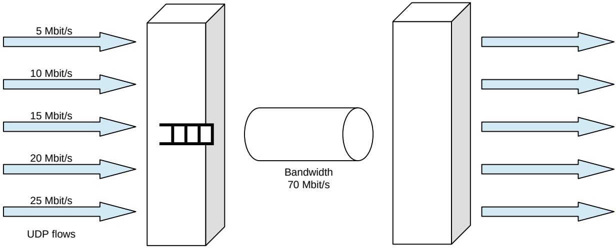

Fig. 1. Experimental system.

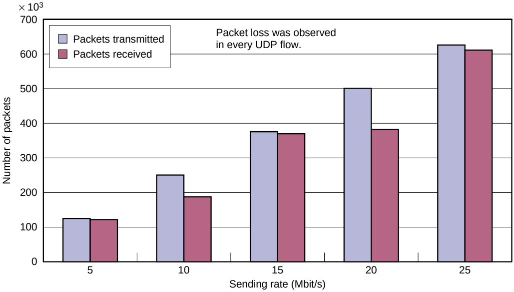

Fig. 2. Packet loss behaviors of UDP flows (FIFO).

tion of the network could be heavily degraded.

Some video conferencing applications use application-level control as a form of congestion control: they reduce their frame rate or frame size when packet loss is detected. If this mechanism were used in all applications, packet loss would result in a reduction of frame rate or frame size, resulting in an excessive reduction overall, leading to networks being underutilized.

3. Dynamic admission control of UDP flows

The network mechanism should discard packets more intelligently by being aware of the notion of flows. By analogy with conventional telephone networks, we need a router mechanism to achieve dynamic admission control of UDP flows. The goal of dynamic admission control is to minimize the number of UDP flows affected by congestion. This policy for packet discarding could avoid the degradation of network utility and generate a comprehensive indication that allows applications to react appropriately to congestion.

In this section, we discuss the network mechanism in general and introduce a flow-based forwarding policy that enables dynamic admission control in best-effort networks.

3.1 Conventional queuing mechanisms

Since IP routers are devices that forward IP packets, an IP router is, by definition, unaware of a flow, i.e., the stream of packets between end-to-end applications. However, there are lots of proposed implementations of queue management and scheduling mechanisms that are aware of the flow. They are roughly categorized into the following two groups in terms of how bandwidth is allocated.

(1) Give priority to certain flows

The rules defining flows are specified beforehand, and an IP router prioritizes their packets by a predefined amount. Priority queuing and CBQ (Class-Based Queuing) [4], [5] are examples of this approach.

(2) Distribute loss fairly among arbitrary flows

In this scheme, an IP router manages the packet forwarding so that the bandwidth is distributed fairly among arbitrary flows. One example is WFQ (Weighted Fair Queuing) [6], [7] that is used with assigning equal weight to all flows.

Dynamic admission control could be achieved using a mechanism in the former category. However, to prioritize specified flows, it is necessary to configure all routers along the path to set up the rules beforehand by using some control protocol such as RSVP (Resource reSerVation Protocol). There are known to be several difficulties in achieving this protocol economically [8], and dynamic use of UDP flows during video conferencing sessions might exhibit a substantial control overhead.

To distribute bandwidth fairly among arbitrary flows, we do not need any control protocol. However, the distribution of bandwidth is enforced without regard to the minimum requirements of each UDP flow. Therefore, mechanisms of this type cannot achieve dynamic admission control. The congestion could affect many applications, because most of the flows would be affected by the fair sharing of bandwidth.

The same system as in Fig. 1 can evaluate how congestion affects each UDP flow when WFQ is used. We used WFQ in the left router in Fig. 1. Figure 3 shows the number of packets sent and received for each UDP flow during the test period of three minutes. With WFQ, the higher-rate flows (20 and 25 Mbit/s) suffered from a lot of packet loss independent of their sending rates. This is because WFQ forced those flows to have the same bandwidth. It is clear that WFQ cannot be used for dynamic admission control, even through the lower-rate flows (5, 10, and 15 Mbit/s) were protected from packet loss in this specific example.

3.2 Proposed forwarding policy

To achieve dynamic admission control, we propose a forwarding policy that gives high priority to smaller-rate flows. This policy is summarized as follows.

- Giving Low Priority to Higher Rate Flows

  Prioritize incoming packets according to the estimated sending rate of the flows they belong to, so that packets of smaller-rate flows take higher priority, those of higher-rate flows take lower priority.

Under this policy, smaller-rate flows can be protected even when congestion occurs, because they get a higher priority. The higher-rate flows do suffer from packet loss: the degree of packet loss depends on the estimated sending rate. Accordingly, packets are discarded starting with the highest-rate flows and continuing in descending order of sending rate, until the network has recovered from the congestion.

The characteristics of this policy are shown by the following example. Consider 51 flows: 50 flows of 1 Mbit/s and 1 flow of 50 Mbit/s. When the total avail-

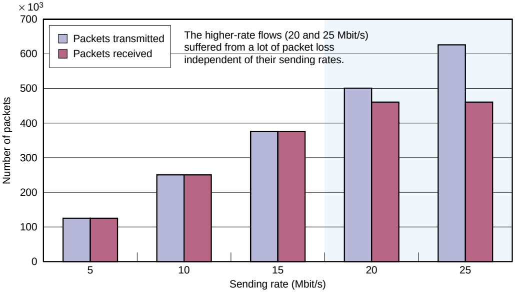

Fig. 3. Packet loss behaviors of UDP flows (WFQ).

able bandwidth is 70 Mbit/s, the policy discards packets of the 50-Mbit/s flow. The other 50 flows continue to use their bandwidth without any problems even though the network is close to congestion. A smaller rate flow has higher priority over a greedy, higher-rate flow.

This policy achieves dynamic admission control in a best-effort packet-based network. Discarding packets from the highest bandwidth flow is reasonable, because high-bandwidth flows often exceed the fair share bandwidth and they have enough margin to react to congestion, for example, by reducing the frame size. Under this policy, the packet loss can be regarded as a comprehensive indication that the sending rate of this flow is the highest among the flows. This indication can be used by applications to react to congestion, which could avoid the network being under-utilized.

4. MXQ mechanism

The MXQ mechanism [3] is a flow-aware active queue management technique that can achieve dynamic admission control of UDP flows. It consists of a rate estimation function and a selective discard function. These functions can be added to a typical IP router architecture as shown in Fig. 4. They are described briefly below. For details, see ref. [3].

4.1 Rate estimation function

The incoming packets are classified by five information fields in the packet header: the source IP address, destination IP address, protocol number, source port number, and destination port number. The combination of information in these fields defines the flow. The state information about a flow is maintained at the ingress interface, and the sending rate is estimated from the inter-arrival time of consecutive packets and their length.

Two methods can be used for estimation. The choice depends on the performance requirements and the degree of traffic variation. The exponential averaging method [9] can estimate a stable average value over a certain period but the computation load is rather high. The time-sliding window [10] is a simpler method that can estimate a stable average value in a rather longer period.

The sending rate is estimated every time a packet is received by an interface card, and its value is attached to the packet in a dedicated internal header, so that the selective packet discard function can refer to the value.

4.2 Selective packet discard function

Figure 5 shows the block diagram of the selective packet discard function. Each packet destined for the output port is evaluated in the packet discard block before it is queued in the packet buffer. The decision to forward the packet or discard it is based on a probability function calculated from three values: the estimated sending rate, the average rate, and the queue length. The policy of prioritizing smaller rate flows is enforced by this packet discarding function block.

The average rate is derived by calculating statistical

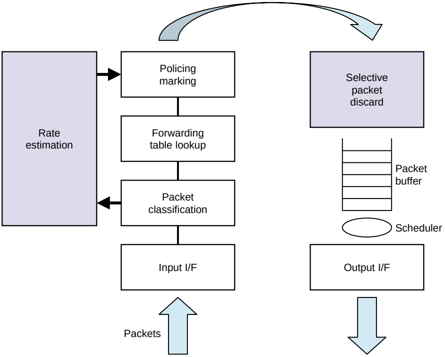

Fig. 4. Block diagram of MXQ.

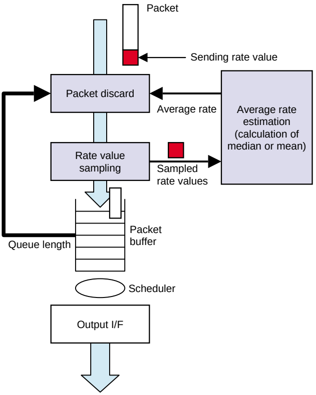

Fig. 5. Selective discard function.

measures, such as the mean and median, of sampled sending rate values attached to the packets. Only packets that have been chosen to be forwarded have their sending rate values sampled, so the statistical measures can be regarded as estimates of the average flow rate passing through the output port.

Even when the sending rate is high, we do not have to discard packets if there is no congestion. The packet discard decision incorporates a parameter that reflects the existence of congestion. When congestion occurs, the queue length quickly increases toward the maximum. The target queue length parameter is introduced and congestion is defined as existing when the queue length exceeds the target queue length. In our prototype, the target queue length was set to half the maximum queue length.

4.3 Prototype system and admission control capability by MXQ

We have developed a software-based prototype system of an MXQ-enabled PC-based router. The software platform is FreeBSD 4.1 Release, and Alternate Queuing [11] is used. The MXQ mechanism is implemented as a module for Alternate Queuing. All of the experiments described in this paper were conducted on this prototype system.

The same system as in Fig. 1 was used to evaluate the dynamic admission control capability of MXQ.

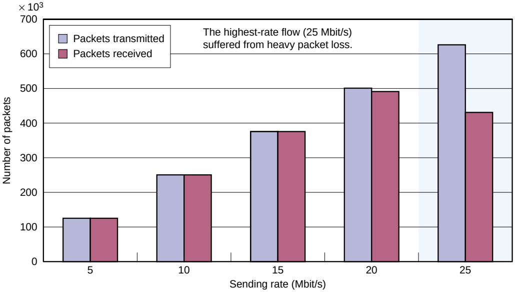

Fig. 6. Packet loss behaviors of UDP flows (MXQ).

We used MXQ in the left router in Fig. 1.

Figure 6 shows the number of packets sent and received for each UDP flow during the test period of three minutes. It is clear that the highest-rate flow (25 Mbit/s) suffered from heavy packet loss. However, the second highest flow (20 Mbit/s) did not suffer very much. The lower-rate flows (5, 10, and 15 Mbit/s) were protected from packet loss. MXQ affected only the small number of highest-rate flows when congestion occurred; thus, MXQ achieved dynamic admission control of UDP flows.

Note that the 20-Mbit/s flow suffered the small level of packet loss. Selective discarding of the highest-rate flow resulted in a decrease in the actual throughput of the flow. This led to the second highest flow temporarily becoming the highest one. When this happened, the 20-Mbit/s flow suffered packet loss.

5. Field experiment involving actual video conferencing applications

We conducted a field experiment using the long-distance network that spans Asia and North America. The experiment established video conferences between parties on different sides of the Pacific. As shown in Fig. 7, the trans-Pacific link operated by GEMnet [12] and the Abilene Network, which is operated by the Internet2 [13], a consortium led by over 200 universities in the United States, connected the two sites: our Laboratory in Yokosuka, Japan and the campus of the University of Southern California. The bandwidth of the transpacific link was about 700 kbit/s, and the RTT (Round Trip Time) was about 130 ms.

Two H.323 video conferencing applications were used in the experiment. One (App1) transmitted a video flow at 240 kbit/s (average) and 610 kbit/s (peak) and an audio flow at a constant rate of 69 kbit/s. The second (App2) transmitted a video flow at 360 kbit/s (average) and 990 kbit/s (peak) and an audio flow at a constant rate of 13 kbit/s. To compare their behaviors, we connected a video source to the two terminals to send the same video and audio synchronously.

The total average bandwidth did not exceed the bandwidth of the trans-Pacific link. However, the video conferencing applications generated bursts of traffic at scene changes, which was sufficient to trigger congestion. The impact of packet loss depended on the application's encoding method and implementation. With App1, packet loss was observed as frame skipping, while with App2 it was seen as image corruption. Noise, interruption or delay, was observed in the audio stream with both applications.

FIFO allowed image corruption in App2's video, and frame skipping in App1's video. The audio streams of both applications were noisy, and sometimes interrupted. It was confirmed that congestion degraded the quality of both video conferencing applications. MXQ restricted most of the quality degradation to App2's video; frame skipping was infrequent in App1's video. Moreover, the audio streams of both applications were clear, with no interruption or delay. Thus, the communication quality of

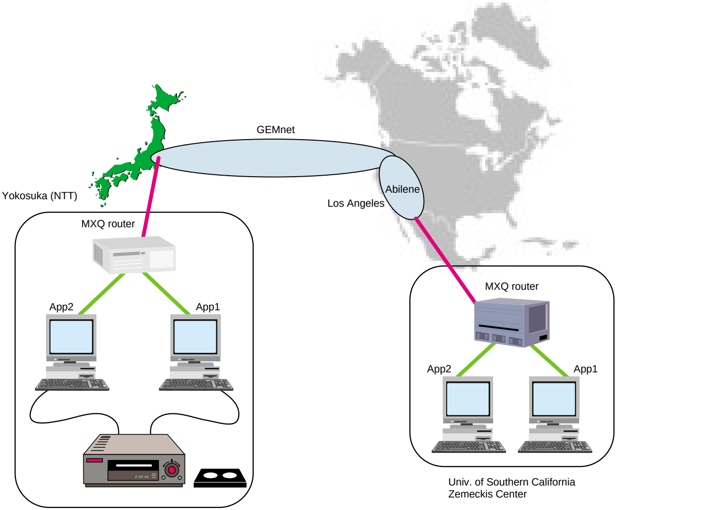

Fig. 7. Field trial at Internet2 meeting.

both applications was greatly improved, which implies that the network utilization was improved.

To determine the precise relationship between packet loss and application behavior, we conducted an additional experiment by emulating the field experiment in our laboratory. The precise packet loss was analyzed by capturing packet data. Figure 8 shows the number of packets discarded from each flow in a certain period, when using FIFO, WFQ, and MXQ. It is clear that the MXQ discarded packets mostly from the video flow of App2 (the highest-rate flow), while FIFO discarded packets of all flows. This result agrees with the observed behavior of the applications.

WFQ discarded packets from the video flows of both App1 and App2. However, the total number of discarded packets was higher than in the case of MXQ. This result indicates that WFQ failed to generate a useful indication that enabled applications to make an appropriate reaction to congestion.

The dynamic admission control of the MXQ mechanism can avoid the degradation of utilization and provide a comprehensive indication to applications to enable them to make an appropriate reaction to congestion.

6. Conclusion

We proposed a new forwarding policy in which the small rate flows get high priority. This policy can achieve one type of dynamic admission control in best-effort networks and solves some problems caused by video conferencing applications.

To achieve this policy, we proposed the MXQ mechanism, and we evaluated its effectiveness through a number of experiments conducted using a software-based prototype system, including a large-scale trans-Pacific field trial conducted at the Fall

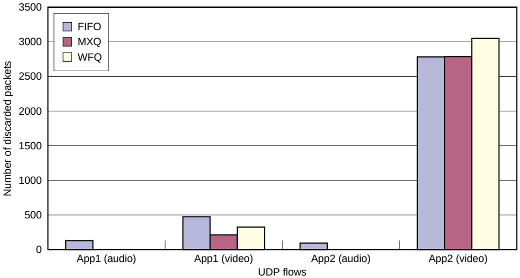

Fig. 8. Number of discarded packets for each UDP flow.

2002 Internet2 Member Meeting. The results show that the dynamic admission control achieved by MXQ could effectively control the congestion and promote better use of best-effort networks.

This mechanism will benefit service providers whose customers are heavy users of UDP-based applications, such as video conferencing.

References

[1] ITU-T, "H.323 packet-based multimedia communications systems," ITU-T Recommendations, Nov. 2000.

[2] S. Floyd and K. Fall, "Promoting the use of end-to-end congestion control in the Internet," IEEE/ACM Trans. Networking, Vol. 7, No. 4, pp. 458-472, Aug. 1999.

[3] T. Shimizu, M. Nabeshima, I. Yamasaki, and T. Kurimoto, "MXQ (MaXimal Queuing): A network mechanism for controlling misbehaving flows in best effort networks," IEICE Trans. Inf. & Syst., Vol. E84-D, No. 5, 2001.

[4] S. Floyd and V. Jacobson, "Link-sharing and resource management models for packet networks," IEEE/ACM Trans. Networking, Vol. 3, No. 4, pp. 365-386, Aug. 1995.

[5] M. Shreedhar and G. Varghese, "Efficient fair queuing using deficit round robin," Proc. ACM SIGCOMM '95, Cambridge, MA, U.S.A., Nov. 1994.

[6] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," Proc. ACM SIGCOMM '89, pp. 1-12, Sep. 1989.

[7] P. E. McKenney, "Stochastic fairness queueing," Proc. INFOCOMM '90, Jun. 1990.

[8] Lee Breslau and S. Shenker, "Best-effort versus reservations: a simple comparative analysis," Proc. ACM SIGCOMM '98, Vancouver, BC, Canada, Sep. 1998.

[9] I. Stoica, S. Shenker, and H. Zhang, "Core-stateless fair queuing: achieving approximately fair bandwidth allocation in high speed networks," Proc. ACM SIGCOMM '98, pp. 118-130, Vancouver, BC, Canada, Sep. 1998.

[10] D. Clark and W. Fang, "Explicit allocation of best-effort packet delivery service," IEEE/ACM Trans. Networking, pp. 362-373, Aug. 1998.

[11] K. Cho, "A Framework for alternate queueing: towards traffic management by PC-UNIX based routers," Proc. USENIX '98, pp. 247-258, New Orleans, LA, U.S.A., Jun. 1998.

[12] K. Okada, J. Kishigami, and N. Miyaho, "Utilization of GEMnet and its Functional Enhancement," NTT R&D, Vol. 50, No. 3, pp. 174-182, 2001 (in Japanese).

[13] http://www.internet2.edu/

Atsushi Takahara

Director, Service Development & Operations Department, Visual Communications Division, NTT Bizlink Inc.

He received the B.S., M.S., and Dr.E. degrees from Tokyo Institute of Technology, Tokyo, in 1983, 1985, and 1988, respectively. In 1988, he joined the LSI Laboratories, NTT, Kanagawa, Japan, and he has been researching in formal methods of VLSI Design, reconfigurable architecture and Inernet protocol processing. He is currently a Director of Service Development and Operations Department, Visaul Communications Division of NTT Bizlink Inc. His current research and development interests are in IP networking for bi-directional real time communication applications and IP infrastructure technologies. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE), Association for Computing Machinery (ACM), and Information Processing Society of Japan (IPSJ).

Takashi Shimizu

Research Engineer, Media Networking Laboratory, NTT Network Innovation Laboratories.

He received the B.S., M.S., and Dr.E. degrees in electrical engineering from Tokyo Institute of Technology, Tokyo, in 1991, 1993, and 1996, respectively. In 1996, he joined the NTT Optical Network Systems Laboratories (now NTT Network Innovation Laboratories), Kanagawa, Japan. Since then he has been engaged in researching packet switching technology for high speed computer communications. His research interests include packet-based network architecture, hardware and software system design technique, and distributed interactive applications. He is a member of IEICE, and the IEEE Communications Society.