Quality Requirement Coordination in Rate Adaptation of Multiple Layered Videos

Akihito Hiromori
Graduate School of Eng. Sci.
Osaka University
Toyonaka, Osaka, 560-8531, Japan
hiromori@ics.es.osaka-u.ac.jp

Takeshi Yamashita
Graduate School of Eng. Sci.
Osaka University
Toyonaka, Osaka, 560-8531, Japan
t-yamast@ics.es.osaka-u.ac.jp

Hirozumi Yamaguchi
Graduate School of Eng. Sci.
Osaka University
Toyonaka, Osaka, 560-8531, Japan
h-yamagu@ics.es.osaka-u.ac.jp

Keiichi Yasumoto
Dept. of Info. Proc. and Mng.
Shiga University
Hikone, Shiga, 522-8522, Japan
yasumoto@biwako.shiga-u.ac.jp

Teruo Higashino
Graduate School of Eng. Sci.
Osaka University
Toyonaka, Osaka, 560-8531, Japan
higashino@ics.es.osaka-u.ac.jp

Kenichi Taniguchi
Graduate School of Eng. Sci.
Osaka University
Toyonaka, Osaka, 560-8531, Japan
higashino@ics.es.osaka-u.ac.jp

Abstract

In this paper, we propose a layer adaptation scheme for multiple layered video multicast distribution. Assuming that each user has his/her own quality requirement for the multiple videos, a "quality coordinator" determines the set of layers of videos to be dropped by those users who have detected network congestion on their common bottleneck link so that their requirements are satisfied as much as possible. Based on the proposal scheme, MPEG1 layered video distribution system has been designed and implemented. Through an experiment on an IP multicast network, we have confirmed that the recovery time from network congestion was less than half time and the satisfied requirements are higher than a simple layer adaptation where each user drops layers according to his/her requirement independently of the others.

1. Introduction

As the recent remarkable innovation of the infrastructure of networks and the computing capability of end hosts, users in multimedia applications may receive and play multiple videos on their hosts transmitted from different source hosts. For example, in video conferencing systems, each user may watch the videos of speakers, presentation slide and conference site overview simultaneously. The similar situations can be considered in many other applications such as remote lecturing, multi-site monitoring and multipoint live video streaming.

As the need for these applications is growing rapidly, we should focus our attention on the significant problem *i.e.*, they are bandwidth wasting applications. Since multiple videos often consume much bandwidth, they may cause network congestion even though each video is transmitted via multicast. Therefore, an appropriate rate control scheme is desired. Receiver-driven Layered Multicast (RLM) [6] provides a scalable rate control of a single video where users can adapt the receiving rate to their own resources (*e.g.* network bandwidth) in heterogeneous networks. The source video is hierarchically encoded where each layer enhances the quality of the video decoded from its lower layers, and those layers of the video is transmitted via different multicast groups. Each user who has detected network congestion drops some layers (*i.e.* leaves the multicast groups of the layers) of the video. Therefore, if network congestion occurs on a bottleneck link, it will be avoided within a certain period after the users behind the bottleneck link have detected congestion and dropped some layers of the video.

However, if there are multiple layered multicast videos and the quality requirement of each user for those videos is different from the others, a new problem arises. For example, assume that the receiving rates of two layered videos v_1 and v_2 become falling down at two users u_a and u_b by network congestion on a bottleneck link. Also assume that u_a drops some layers of v_2 to keep the high quality of v_1, and on the other hand u_b drops some layers of v_1 to keep the high quality of v_2. In this case, the rates of v_1 and v_2 on the bottleneck link may not be decreased because v_1 and v_2 are still transmitted in the high quality to u_a and u_b, respectively. Consequently, both users u_a and u_b may think that the congestion has not been avoided yet, and finally drop the layers of v_1 and v_2 respectively, after a long term. This means that in order to avoid network congestion quickly caused by multiple layered videos where users with different requirements for those videos exist, a certain coordination scheme would be significant that determines the layers to be dropped, satisfying those users' quality requirements.

In this paper, we propose a layer adaptation scheme for multiple layered video multicast distribution. In our scheme, assuming that (a) users receive multiple layered videos from different sources, (b) each user has a quality requirement between his/her receiving layers and (c) there exists a "quality requirement coordinator" on a certain network node, the coordinator receives the congestion notifications from some users, calculates the number of layers of each video to be dropped by those users to avoid the congestion on their common bottleneck link, and sends the calculation result (a set of layers to be dropped) in response. The quality requirement of each user is defined as a set of *priority relations,* each of which represents whether one layer should be more important than another or not. The collection of the quality requirements of those users are translated into the *priority values* of layers, and the set of layers is determined to be dropped by those users to avoid the congestion so that the total loss of the priority values can be small enough.

Based on the proposal scheme, MPEG1 layered video distribution system has been designed and implemented. Through an experiment on IP multicast where three MPEG1 layered videos are transmitted to two users with opposite quality requirements, we have confirmed that network congestion was avoided within less than half time and the satisfied requirements are higher compared with an adaptation algorithm where each user drops layers according to his/her requirement independently of the others.

1.1. Related Work

Many researches for the rate adaptation of a single layered video have been investigated[14, 15, 16, 17, 18, 19], however, few methods are proposed for the rate adaptation of multiple layered videos on IP multicast. Ref. [7] proposes a method to decide the priority of layers of multiple videos from the collection of all users' preferences, and aggregate a certain set of the layers with the highest priorities into one layer. This method can adapt the transmission rate according the bandwidth on the bottleneck link. However, it is basically a sender-initiated rate adaptation where the sender controls the layers to be transmitted. The more precise rate control is required in large-scaled networks. In [21], Aoyama at.el proposes a receiver-initiated (user-oriented) rate adaptation, however, this method does not mention the different requirements of users. Shacham at.el [9, 10] and our research group [11, 12] have proposed coordination methods for multiple layered multicast. However, these methods assume that the coordinator knows the transmission rates of each video on all links, and it is not easy to apply them to the rate adaptation on IP multicast.

2. Layer Control for Multiple Layered Video Distribution

In our proposed scheme, we assume multiple servers s_1, ... and s_n, a *quality coordinator* on a certain node, and multiple receivers r_1, ... and r_m who receive those layered videos. At each server s_i, a video v_i is hierarchically encoded and they are sent via different multicast groups from s_i. Hereafter, $v_{i,l}$ denotes the l-th layer of v_i where the 1st layer is the basic (indispensable) layer. $(l+1)$-th layer is used to enhance the quality of video decoded from l layers, *i.e.* 1st, ... and l-th layers. Each receiver can establish a connection with the quality coordinator on demand.

2.1. Quality Requirement

Each receiver r_k specifies his/her quality requirement for the receiving layered videos as a set of priority relations, each of which represents whether one layer should be more important than another or not. Hereafter, each priority relation between two layers $v_{i,x}$ and $v_{j,y}$ is denoted as $v_{i,x} > v_{j,x}$ if $v_{i,x}$ is important than $v_{j,x}$.

This is an intuitive way for receivers to specify their requirements. For example, in remote lecturing, a participant often wants to see the video of the lecturer (say v_1) with higher quality than those of the other participants (say v_2, ... and v_h). Moreover, he/she may want to keep at least the lowest quality for v_2, ... and v_h. In this case, this receiver specifies his/her quality requirement as follows.

$$v_{1,lcur_1} > v_{i,2}, \;\; v_{1,2} < v_{i,1} \quad (2 \leq i \leq h)$$

Note that $lcur_1$ denotes the current highest layer of v_1 that this receiver receives. The first relation represents that the highest layer of v_1 should be kept prior to the second layers of the other videos. The second relation represents that the basic (1st) layers of v_2, ... and v_h should be kept prior to any layer of v_1 except the basic layer. Consequently, if this requirement is satisfied, the layers of v_2, ... and v_h except their basic layers are dropped prior to any layer of v_1, and the layers of v_1 except its basic layer are dropped prior to the basic layers of v_2, ... and v_h.

2.2. Layer Adaptation by Quality Coordinator

We assume that each receiver measures the *effective rates* of receiving videos at the receiver's host and detects network congestion through the degradation of the effective rates. An effective rate is the total size of the data transmitted to an end host within a permitted delay during a unit of time. The permitted delay depends on applications. For example, it must be small enough in video conferencing systems to reduce the delay in conversation. Once a receiver detects network congestion, it establishes a connection with the quality coordinator and sends a notification that includes (a) the current receiving status (the current numbers of layers of the receiving videos), (b) shortage bandwidth for the transmission of those layers, and (c) his/her quality requirement. Here, we assume that each receiver estimates the shortage bandwidth from the required (transmission) rate and the effective rate of each video v_i.

The quality coordinator receives the notifications from some receivers and regards that those users share the same bottleneck link. Then the coordinator considers the maximum shortage bandwidth reported from the receivers as the shortage bandwidth on the bottleneck link, and determines the set of layers that should be dropped to avoid the congestion, so that the requirements of receivers can be satisfied as much as possible. For this purpose, it translates the set of priority relations given by the receivers into *priority values* of layers. By this translation, each layer has a priority value that represents the priority of the layer among all the layers (the details are explained in Section 2.4). The quality coordinator determines the set of layers where their total transmission rate is larger than the shortage bandwidth and the sum of their priority values is small enough. The decision is sent to those receivers as reply messages in response. Each receiver who has received the reply message drops layers as specified in the reply message. Consequently the congestion will be avoided within a certain period since the receivers behind the bottleneck link drop the layers equally.

2.3. Example

In Fig. 1(a), receivers r_1, r_2 and r_3 receive three videos v_1, v_2 and v_3, each of which is hierarchically encoded into three layers. Each video requires 1,200 kbps for its transmission (suppose that each layer requires 400 kbps for simplicity of discussion). r_2 receives two layers of v_1, three layers of v_2 and two layers of v_3, and r_3 receives three layers of v_1, two layers of v_2 and three layers of v_3. Now assume that a network congestion occurs on a link L. In this case, r_2 detects this congestion by measuring that the effective rates of v_2 and v_3 have been falling down to 80% of the required rates. Then r_2 estimates the shortage bandwidth as 1,200 kbps * (1-0.8) + 800 kbps * (1-0.8) = 400 kbps and sends a notification message to the quality coordinator to report (a) the shortage bandwidth (400 kbps), (b) the numbers of currently receiving layers of v_2 and v_3 (three and two, respectively), and (c) his/her quality requirement. Similarly, r_3 detects the same congestion by measuring that the effective rate of v_1 has been falling down to 70% and v_2 and v_3 to 80%. r_2 estimates the shortage bandwidth as 1,200 kbps * (1-0.7) + 800 kbps * (1-0.8) + 1,200 kbps * (1-0.8) = 760kbps and sends a notification message to the quality coordinator to report (a) the shortage bandwidth (760 kbps), (b) the numbers of currently receiving layers of v_1, v_2 and v_3 (three, two and three, respectively) and (c) his/her quality requirement.

The quality coordinator regards that three layers of v_1, v_2 and v_3 (the maximum numbers of layers) compete for the limited bandwidth on a bottleneck link. Therefore, it determines a set of layers that should be dropped to avoid this congestion. Now assume that the requirement of r_2 is following.

$$v_{2,x+1} < v_{3,x}, \;\; v_{2,x} > v_{3,x+1} \;\; (x = 1, 2)$$

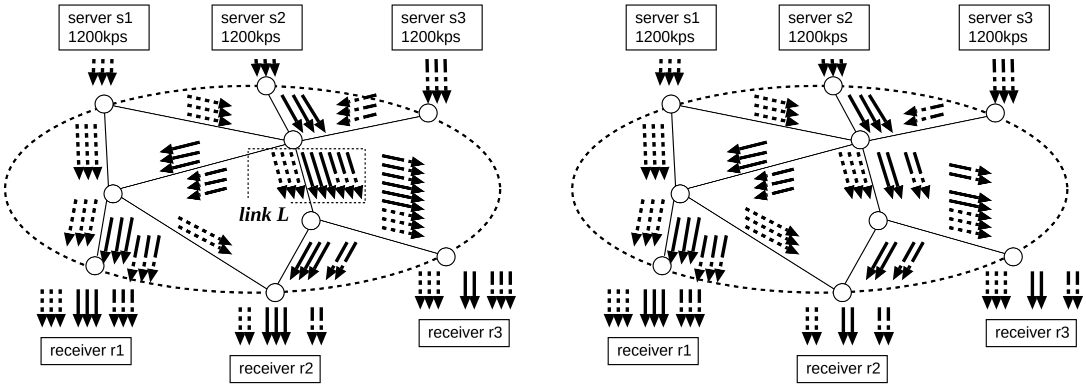
This means that x-th layers of v_2 and v_3 are more important for r_2 than their $(x + 1)$-th layers. On the other hand, the requirement of r_3 is following.

$$v_{1,3} > v_{2,1}, \;\; v_{2,2} > v_{3,1}$$

This means that v_1 is most important and v_3 is least important for r_3. The quality coordinator considers that the shortage bandwidth is the maximum shortage bandwidth reported from the receivers (in this case 760 kbps), and that two layers (400 kbps * 2) should be dropped. The quality coordinator translates the above requirements into priority values and determines the set of layers to be dropped according to the algorithm in Section 2.4. Consequently the layers $v_{2,3}$ and $v_{3,3}$ are determined to be dropped, and r_2 and r_3 drop $v_{2,3}$ and $v_{3,3}$, respectively. Fig. 1(b) shows the situation after the congestion is avoided. Note that the multicast tree that has transmitted $v_{3,3}$ is now pruned and the packet of the multicast group is not forwarded beyond the local subnet of s_3, because there is no group member.

2.4. Problem Formulation and Algorithm

Hereafter, R denotes the set of receivers who have detected the same congestion, and V_j denotes the set of videos



(a) Congestion **(b) Layer Adaptation**

Figure 1. Example

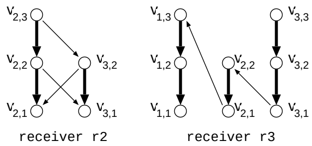


Figure 2. Directed Graphs Generated from Requirements of Receivers

by which receiver r_j has detected the congestion. Each receiver $r_j \in R$ may be behind the same bottleneck link L to receive the videos of V_j.

Here, we assume that every receiver knows the required (transmission) rate of each $v_{i,x}$ (denoted as $B_{i,x}$). Each receiver r_j estimates his/her shortage bandwidth based on the effective rate of each $v_{i,x}$ (denoted as $P_{i,x,j}$) as follows; $Bshort_j = \sum_{v_i \in V_j} \sum_{1 \leq x \leq lcur_{i,j}} (B_{i,x} - P_{i,x,j})$ where $lcur_{i,j}$ denotes the number of layers of v_i that r_j receives. The quality coordinator considers the maximum shortage bandwidth reported from receivers $Bshort = \max_{r_j \in R} Bshort_j$ as the shortage bandwidth on the bottleneck link L. Moreover, the number of layers of video v_i on the bottleneck link L (denoted as $Lmax_i$) is the maximum number of layers reported from the receivers; $Lmax_i = \max_{r_j \in R}\{lcur_{i,j}\}$. The set of videos on L (denoted as V) can also be defined as follows; $V = \bigcup_{r_j \in R} V_j$.

Then we transform the requirements given as priority relations between two layers $v_{i,x}$ and $v_{j,y}$ into *priority values* of layers according to the following policy.

- For each receiver r_j, $|V_j| * C$ is given where C is a common constant for all the receivers. This is the sum of priority values for the layers of the videos V_j.

- The set of priority relations between two layers given by each r_j forms the partial order of those layers. Here, it is natural to consider the "depth" of a layer in the partial order as the priority of the layer. Therefore, we generate a directed graph where each node corresponds to a layer and a directed edge from node $v_{i,x}$ to $v_{j,y}$ represent a priority relation $v_{i,x} < v_{j,y}$. In the graph, let $d_{i,x,j}$ denote the maximum length of directed path from a leaf node to node $v_{i,x}$ in the graph of r_j. For example, Fig. 2 shows the directed graphs generated from the requirements of r_2 and r_3 in the example of Section 2.3. In this example, $d_{3,1,2} = 2$ and $d_{1,1,3} = 7$. Note that the thick arrows in the graph represent the layer dependency of a video (each layer should be dropped earlier than its lower layer) and this dependency should also be treated as priority relations.

- Finally, we define the priority value of each layer $p_{i,x}$ as follows.

$$p_{i,x} = \sum_{r_j \in R} |V_j| * C * \frac{d_{i,x,j} + 1}{\sum_{r_j \in R} (d_{i,x,j} + 1)}$$

For given $Bshort$, V, L_i, $B_{i,x}$ and $p_{i,x}$, the problem is to find a set of layers where the sum of their priority values is the minimum. This problem is a combinatorial optimization problem (a kind of the knapsack problem) where the complexity is NP-complete. In this paper, we adapt a greedy method where we iteratively select the layer with the minimum priority value per unit of bandwidth from the currently the highest layers, until the sum of the bandwidth of selected layers reaches the shortage bandwidth $Bshort$.

Figure 3. Snapshot of Layer Control System

3. Layer Control System for MPEG1 and Experiment

3.1. Layer Control System for MPEG1

We have designed and implemented a prototype of MPEG1 layer adaptation system, using a developer's kit for MPEG1 layered multicast called *Vidaris* provided by KDDI R&D Laboratories.

The system consists of a quality coordinator and a set of clients (receivers). The quality coordinator is written in C and designed as a daemon program on UNIX. It waits multiple requests to establish connections from clients (receivers) within a certain period and makes a copy of itself to process their requests and wait new requests in parallel. The client is written in Visual C++ using Vidaris library. It establishes a TCP connection with the quality coordinator, and playbacks multiple videos sent from multiple MPEG1 layered video servers. As video servers, we have used a prototype program of Vidaris that encodes each MPEG1 file into six layers and sends them via different multicast groups. The session information and the location of the quality coordinator are given by XML description placed on a certain URL. Each client has a built-in function of Vidaris that measures the effective rate of each video. Fig. 3 shows a snapshot of the client program.

Then we had an experiment on an IP network. The network consists of three PC routers A, B and C (CPU:Pentium 90MHz, OS:FreeBSD3.4) as shown in Fig.4. On routers A and C, multicast routing daemons *mrouted ver3.9b* are running and a virtual multicast connection is established between them by the tunneling functionality over UDP of *mrouted*. Using rate_limit functionality of *mrouted*, router C limits the rate of multicast packets transmitted to the LAN between routers B and C to 5,000 kbps. The quality coordinator is located on router A.

On this network, three servers s_1, s_2 and s_3 on the local subnet of router C transmit MPEG1 layered videos v_1, v_2 and v_3, respectively. Each video is 352x240 30fps MPEG1 video of constant bit rate (about 1,200 kbps) and encoded

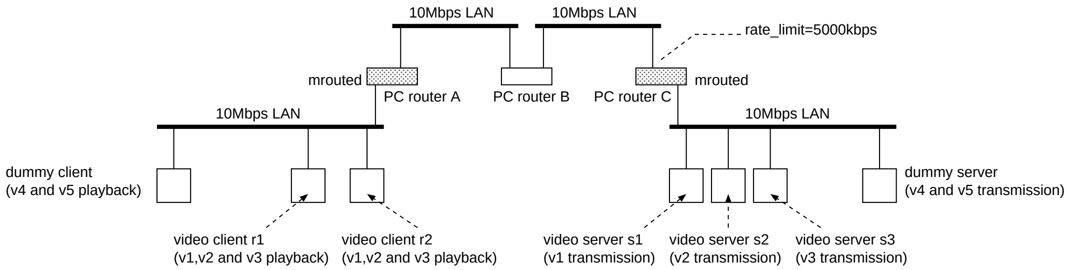
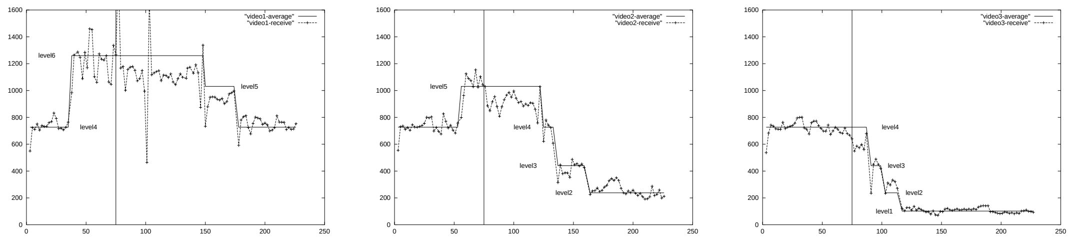
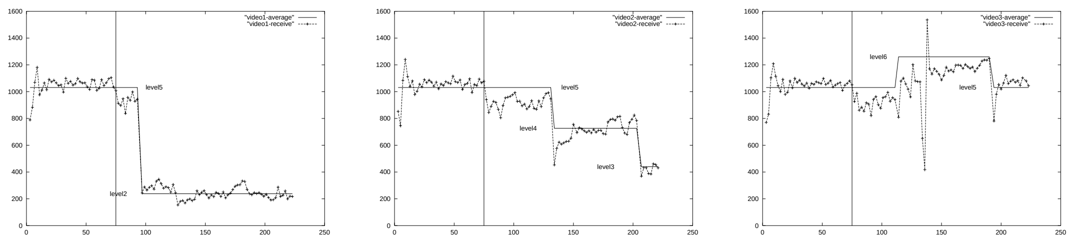


Figure 4. IP Network

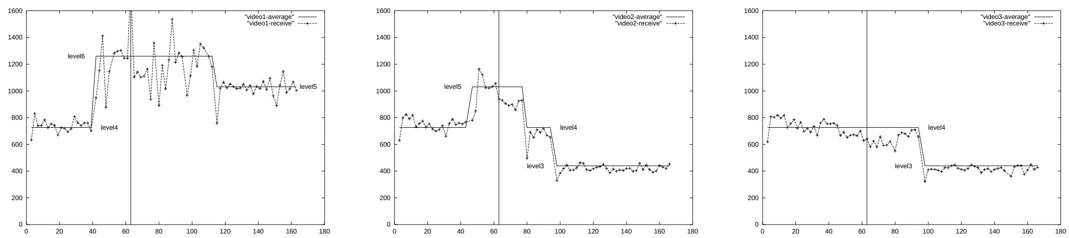


(1-1) v_1 at client r_1 (1-2) v_2 at client r_1 (1-3) v_3 at client r_1

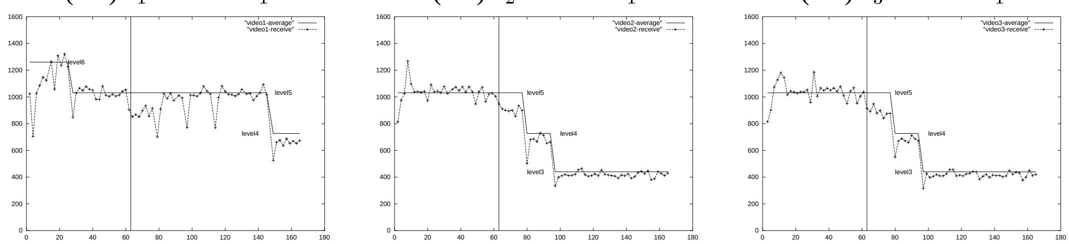


(2-1) v_1 at client r_2 (2-2) v_2 at client r_2 (2-3) v_3 at client r_2

Figure 5. Effective Rates of v_1, v_2 and v_3 at Clients r_1 and r_2 in Independent Adaptation



(1-1) v_1 at client r_1 (1-2) v_2 at client r_1 (1-3) v_3 at client r_1



(2-1) v_1 at client r_2 (2-2) v_2 at client r_2 (2-3) v_3 at client r_2

Figure 6. Effective Rates of v_1, v_2 and v_3 at Clients r_1 and r_2 in Our Adaptation

into six layers. There exist two clients r_1 and r_2 on the local subset of router A, and r_1 receives six layers of v_1, five layers of v_2 and four layers of v_3, and r_2 receives fives layers of v_1, v_2 and v_3. r_1 has a requirement where $v_{1,6} > v_{2,1}$ and $v_{2,5} > v_{3,1}$, *i.e.*, v_1 is the most important and v_3 is the least important. r_2 has the opposite requirement where $v_{3,5} > v_{2,1}$ and $v_{2,5} > v_{1,1}$, *i.e.*, v_3 is the most important and v_1 is the least important.

In this situation, both receivers can receive all the videos in stable effective rates. Then the dummy client on the local subset of router A starts to receive two videos v_4 and v_5 from the dummy server on the local subset of router C. Although at least 6,000 kbps is required to transmit all the videos, the output rate at router C is limited to 5,000 kbps. Therefore, this causes congestion on router C.

We have also implemented a simple layer control policy where each client drops the layers of less important video prior to the others, independently of the other clients (called independent adaptation). We have measured the effective rates of three videos v_1, v_2 and v_3 at both clients r_1 and r_2 until the congestion is avoided, in our adaption and the independent adaptation.

Fig. 5 shows the effective rates of v_1, v_2 and v_3 at clients r_1 and r_2 in the independent adaptation. In these graphs, the required (transmission) rates at servers are also shown. The dummy receiver started to receive v_4 and v_5 at time 75. Then according to the requirement, r_1 first dropped the layers of v_3. However, r_2 dropped the layers of v_1, the number of layers on the bottleneck link (between routers A and C) was not decreased. Then r_1 and r_2 dropped the layers of v_2 and the congestion was avoided at time 200.

Fig. 6 shows the effective rates in our adaptation. The dummy client started to receive v_4 and v_5 at time 65. r_1 and r_2 detected the congestion at around time 80, and the quality coordinator ordered to drop $v_{2,5}$ and $v_{3,5}$ to both clients. However, since the effective rates are still instable, r_1 and r_2 detected the congestion again at around time 100, and the quality coordinator ordered to drop $v_{2,4}$ and $v_{3,4}$. After that both clients detected congestion again at around time 110, the quality coordinator ordered to drop $v_{1,6}$ and the congestion was avoided at time 120.

In our adaptation, it took 55 seconds to avoid the congestion, while it took 125 seconds in the independent adaptation. Also, the total number of layers in our adaptation are larger than that in the independent adaptation. Consequently, our adaptation can quickly avoid the congestion, satisfying the requirements of receivers as much as possible.

4. Conclusion

In this paper, we have proposed a layer control scheme for multiple layered video multicast distribution. Through an experiment on IP multicast, we have confirmed that network congestion was avoided within less than half time and the satisfied requirements are higher compared with a simple adaptation by each individual receiver. As future work, in order to confirm the scalability of our adaptation method, we are planning to conduct simulation assuming the large number of users on MBONE topology.

Acknowledgment

We would like to thank Mr. Shigeyuki Sakazawa and Dr. Masahiro Wada in KDDI R&D Laboratories for their useful comments. We would also like to thank KDDI R&D Laboratories that could offer the Vidaris SDK for this work.

References

[1] C. Diot, W. Dabbous and J. Crowcroft, “Multipoint Communication: A Survey of Protocols, Functions, and Mechanisms,” *IEEE Journal on Selected Areas in Communications*, Vol. 15, No. 3, pp. 277-290, April 1997.

[2] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu and L. Wei, “An Architecture for Wide-Area Multicast Routing,” *Proc. of ACM SIGCOMM’94*, pp. 126-135, 1994.

[3] M. Ghanbari, “Two-Layer Coding of Video Streams for VBR Networks,” *IEEE Journal on Selected Areas in Communications*, Vol. 7, No. 5, pp.771–781, 1989.

[4] N. Shacham, “Multipoint Communication by Hierarchically Encoded Data,” *Proc. of INFOCOM’92*, pp. 2107–2114, 1992.

[5] S. McCanne, M. Vetterli and V. Jacobson, “Low-Complexity Video Coding for Receiver-Driven Layered Multicast,” *IEEE Journal on Selected Areas in Communications*, Vol. 15, No. 6, pp. 983-1001, August 1997.

[6] S. McCanne and V. Jacobson, “Receiver-driven Layered Multicast,” *ACM SIGCOMM ’96*, 1996.

[7] E. Amir, S. McCanne and R. Katz, “Receiver-driven Bandwidth Adaptation for Light-weight Session,” *ACM Multimedia*, November 1997.

[8] N. Shacham and J.S. Meditch, “An Algorithm for Optimal Multicast of Multimedia Streams,” *Proc. of INFOCOM’94*, pp.856–864, 1994.

[9] N. Shacham and H. Yokota, “Admission Control Algorithms for Multicast Sessions with Multiple Streams,” *IEEE Journal of Selected Areas in Communications*, Vol. 15, No. 3, pp. 557-566, 1997.

[10] N. Shacham, “Preemption-Based Admission Control in Multimedia Multiparty Communications,” *Proc. of INFOCOM’95*, pp.827–834, 1995.

[11] H. Sakate, H. Yamaguchi, K. Yasumoto, T. Higashino and K. Taniguchi, "Resource Management for Quality of Service Guarantees in Multi-party Multimedia Application," *Proc. of the IEEE 1998 Int. Conf. on Network Protocols (ICNP'98)* , pp. 189–196, 1998.

[12] H. Yamaguchi, K. Yasumoto, T. Higashino and K. Taniguchi, "Receiver-Cooperative Bandwidth Management for Layered Multicast," *Proc. of the IEEE 1999 Int. Conf. on Network Protocols (ICNP'99)* , pp. 43–50, 1999.

[13] X. Li, M. H. Ammar, and S. Paul. "Video multicast over the Internet," *IEEE Network Magazine*, Vol.12, No.2, pp. 46–60, 1999.

[14] A. Legout and E. Biersack. "PLM: Fast convergence for cumulative layered multicast transmission schemes," *Proc. of ACM SIGMETRICS '2000*, 2000.

[15] L. Vicisano, L. Rizzo, and J. Crowcroft. "TCP-like congestion control for layered multicast data transfer," *IRTF RM Workshop*, 1997.

[16] D. Rubenstein, J. Kurose, and D. Towsley. "The impact of multicast layering on network fairness," *Proc. of ACM SIGCOMM '99*, 1999.

[17] D. Sisalem and F. Emanuel. "QoS control using adaptive layered data transmission," *IEEE Multimedia Systems'98*, 1998.

[18] W. Zhao, M. W, and L. Tiwari. "Efficient adaptive media scaling and streaming of layered multimedia in heterogeneous environment," *Proc. IEEE Intl. Conf. on Multimedia Computing and Systems*, Vol.2, pp. 377–381, 1999.

[19] G. Riley, M. Ammar and L. Clay, "Receiver-Based Multicast Scoping: A New Cost-Conscious Join/Leave Paradigm," *Proc. of the IEEE 1998 Int. Conf. on Network Protocols (ICNP'98)*, 1998.

[20] S. Fischer, A. Hafid, G.v. Bochmann and H.d. Meer, "Cooperative QoS Management for Multimedia Applications," *Proc. of Int. Conf. on Multimedia Computing and Systems (ICMCS'97)*, pp. 303–310, 1997.

[21] M. Kawada, H. Morikawa and T. Aoyama, "Cooperative Inter-stream Rate Control Scheme for Layered Multicast," *Proc. of 2001 Symp. on Applications and the Internet (SAINT 2001)*, pp. 147–154, 2001.