A New Approach for the Construction of ALM Trees using Layered Coding

Yohei Okada, Masato Oguro, Jiro Katto and Sakae Okubo

Department of Computer Science, Waseda University
3-4-1 Okubo, Shinjuku, Tokyo 169-8555, Japan
E-mail: {okada, masato, katto}@katto.comm.waseda.ac.jp , sokubo@waseda.jp

Abstract— **This paper presents a novel approach for efficient tree construction in ALM (Application Layer Multicast) using layered video coding. This system is suitable for the users in heterogeneous environment such as xDSL and CATV, where upload and download speeds are different. We point out some problems in these environment of the ALM and show their solutions at two stages. First, we redefine the "degree parameter" which originally specifies the number of single rate streams the host can transmit, to reflect asymmetric link properties and to accommodate layered streams. Second, we develop an efficient tree construction algorithm using the renewed degree parameter, which maximizes total throughput of all hosts. Simulations using NS-2 prove improvement in throughput, delay and overhead by our proposal. Software implementation is also carried out, and subjective quality improvement is verified.**

Keywords—**application layer multicast, overlay**

I. INTRODUCTION

AS network evolves into "broad-band", multimedia contents like music and video tend to spread over the internet. In the conventional client/server model assuming multiple unicast connections, however, some problems are inevitable such as network congestion around the server or on bottleneck links and too much burden on the server due to large content sizes and a huge number of clients. IP multicast has been thought to be the most valid system that solves these problems completely [1]-[3], but its deployment is quite slow due to the complexity of route control, necessity of exchanging routers for multicasting and so on.

Instead, ALM (Application Layer Multicast) has emerged as an alternative to the IP multicasting, in which each host executes route control and packet forwarding not on IP layer but on application layer. A video stream transmitted from the source is received, played by some hosts and forwarded to other hosts simultaneously. There are some merits such as load-balancing and cost-down because ALM does not need any high performance servers.

On the other hand, ALM also has some issues to be solved. For example, hosts which are located on a lower-position of overlay network (or ALM tree) can not receive data when its higher-position hosts leave the tree suddenly. In addition, when layered coding is applied, receivable layers by the host might be reduced due to limitation

of its higher-position hosts. Therefore, it is important how to construct the ALM trees and to adapt to dynamic behavior of the ALM trees.

To construct ALM trees, delay and bandwidth are popular metrics utilized in general [4]-[9]. For interactive communication like IP video phone and TV-conference, much delay causes fatal performance degradation and the delay metric should be applied. For one-way-communication like video streaming, the bandwidth metric should be utilized to achieve to receive high delivered quality of contents [4]-[6].

We propose an approach to construct the ALM trees for layered video streaming. Thus, we give priority to bandwidth over delay. In this case, we consider that the validity can be demonstrated only by making each host receive at various rates according to their environment and by transmitting not a single rate but multiple rates such as in layered coding. Our goal is to make total throughput of all hosts which are in various environment to be maximized by using layered coding.

Moreover, we consider the problem of dynamic behavior of the ALM trees such as parent hosts' departures. When a host in higher-position leaves the tree, its descendants cannot receive the data. Thus, they need to find new parents that are not affected by the departing of the host. Instead of simply re-joining the tree, we propose an efficient process that shortens the recovery time and saves the number of control packets.

II. ISSUES AND OVERVIEW OF OUR SOLUTIONS

There are mainly two issues of the ALM using layered coding. First, a child host can not receive the rate that is more than the rate its parent host receives at due to characteristics of streaming technology [5]. Thus, in case of Figure 1, host B which is a child of host A is not able to receive at high rate even if B has rich bandwidth to download data because A receives at poor rate. In such case, layered coding is not used adequately and each host must choose its parent host carefully. The second issue is complexity to find a proper parent host [6]. Usually, a new participant host joins the ALM tree as a leaf host. Then it searches for its neighbor hosts by trading their bandwidth information and find an adequate host that forwards stream data at the rate which it desires. However, this process is complex and takes much time. Still more, it may be difficult to attain this

end host in an optimum manner because it can get only local area information of the overlay network once it joins the tree.
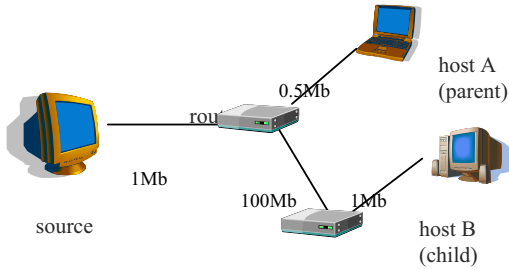

Figure 1. Improper relation between host A and B

Our solution of these matters is to use "degree" parameter. This parameter originally specifies how many hosts a parent host can maintain [10]. The value depends on the parent's forwarding capacity. For example, when the forwarding capacity of host i is Fi and the single streaming rate is R, host i's degree Di is calculated as follows;

$$Di = Fi / R \qquad (1)$$

We will redefine this parameter in the next section for multiple streaming rates so that each host can decide its parent properly. One of the best advantages of using this is that the number of parameter is known in advance. Thus, each new participant can decide its parent host when it joins the ALM tree and there is no necessity of looking for an adequate parent later.
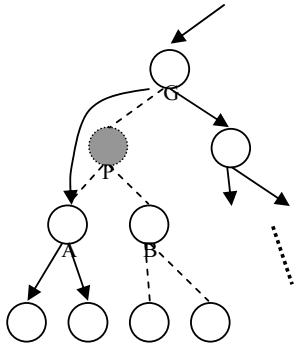

Figure 2. Recovery phase caused by the leaving of host P

Then, it is obvious by the degree parameter that not all descendant hosts can decide the new parent host easily when the parent host leaves. In Figure 2, since all hosts' degree is 2, host B (or host A) cannot reconnect to host G which is a parent host of departing host P, and it must search for another host which has an unused degree.

The simplest way to do that is to rejoin as a new participant host. However, it is considered that this takes a lot of recovery time and overhead. Therefore, we propose a devised method for efficient recovery of the tree.

III. PROPOSAL SYSTEM

A. Redefining the degree parameter

On the Internet, bandwidth of access link, especially up-link, between an edge router and each host of that area is generally much narrower than that of the trunk link. Thus, it is considered that up-link of a sender host is to be the bottleneck link on peer-to-peer networking [6]. Based on this, the degree constraints parameter which represents how many hosts the parent host can have is defined as (1). It is easy to decide whether a new participant host can join the parent host or not by referring to this parameter. In ALM using layered coding, the streaming rate R is not fixed value and we should redefine it.

At first, we use R1 which represents the rate of the base layer stream instead of R in (1).

$$Di = Fi / R1 \qquad (2)$$

Second, R1 is assumed to be '1', and the 'accumulative' rate of each layer's stream is shown by real number as the ratio to R1. As a result, the new degree represents the number of streams which the parent host can have. For example, when rates of multiple streams are {50, 50, 100} (kbps) and host i's sending capacity is 300 (kbps), the degree of the host is **6** according to (2), and the ratio of accumulative rates are **1:2:4**. In such case, this host i has streams of '4 base layers and 1 second layer (**1***4+**2***1=**6**)' or '1 second layer and 1 third layer (**2***1+**4***1=**6**)' and so on. We call this degree parameter 'out-degree.'

Moreover, this approach also applies to the receiving capacity of the hosts. That is, the number of layers which the child host desires is shown with the real number. We call this degree parameter 'in-degree' to distinguish from out-degree.

Finally, we add an important constraint (incentive) to the relation of 'in' and 'out' degrees. In ALM video streaming system, each host contributes a portion of the bandwidth to the overlay in exchange for the receiving service. A host which is able to forward little data should not receive much.Otherwise, other hosts have less streams in return for satisfaction of the host. Thus, any host's in-degree must be limited equal to or less than out-degree as follows:

$$\text{out-degree} \geqq \text{in-degree} \quad \text{(at each host)} \qquad (3)$$

B. Construction of the ALM trees

We build the ALM trees by using these degree parameters. Our goal is to construct the tree which satisfies all hosts' in-degree. There are following 4 key points.

· **A participant finds the proper parent when joining the tree**
    In order to avoid the problem of complexity that a participant finds its proper parent after it has joined the tree, it should be able to decide the host at the same time when joining the tree.

- **Parent's out-degree ≧ child's in-degree**

  To satisfy each host's in-degree, its parent must have out-degree which is bigger than in-degree of the child.
- **Find not only a parent but its own child**

  In relation to the above two points, a participant joins the tree not only as a leaf host but also as an intermediate host of the tree. Thus, it must find its own child simultaneously.
- **Adopt 'delay' as the second metric**

  To avoid the problem of bandwidth waste by connecting hosts that are far from each other in real network, we use a RTT (Round-Trip-Time) metric, too.

Based on the above key points, we explain the process of constructing ALM trees concretely as Figure 3 shows.
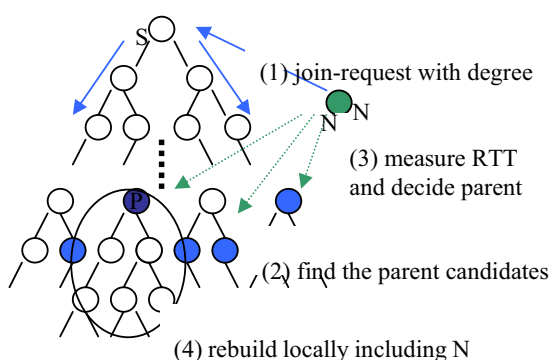
Figure 3. The process of ALM tree construction

Step.1 Send a join-request massage to the source

New participant host N sends a join-request massage to the source with its in and out degree information. The source address and degree information are already known.

Step.2 Search for candidates of the parent host

All hosts maintain the degree information of itself and its children and grand-children. The host receiving the join-request (at first, the source) refers to its degree information and N's one, it becomes the parent candidate of host N when each of the following two conditions is satisfied at least. Otherwise, it forwards the request message to its children and this process might be repeated.

1. N's in-degree ≦ its (remaining) out-degree
2. N's out-degree ≧ out-degree of its children hosts

   The second condition is based on the concept that hosts which have big out-degree should be in higher-position of the tree.

Step.3 Decide the parent

The parent candidates send their response messages to host N and host N measures RTT respectively. Then, host P which has minimum RTT is decided to be the parent host of N.

Step.4 Exchange link connections locally

Under the condition 1 at Step 2, host N is connected to P simply and N becomes a leaf host. If under the condition 2,

P needs to exchange connections in the local area which includes hosts from P to P's grand-children based on the degree and N can join any host of them. At this time, total out-degree of them is equal to or bigger than total in-degree because of (3). Thus, this process can be completed locally.

C. Recovery system

Node departures are of two kinds: graceful departure and sudden failure. In the former case, the departing host notices its will to the source and neighbors. The descendants of the host can decide new parents that have unused out-degree beforehand and reconnect to them seamlessly. In the latter case, the departing host leaves suddenly and cannot give his descendants the time to prepare for the reconnection. This may happen due to a computer crashing or congestion of the network links.
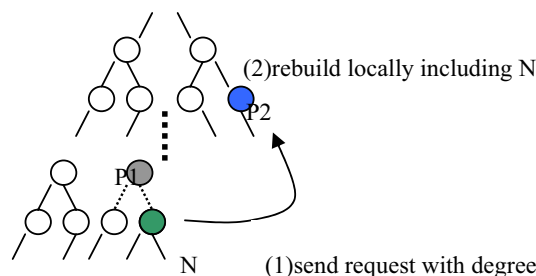

Figure 4. Recovery phase

In either case, the descendant hosts of the departing host needs to find his new hosts. The simplest way is to rejoin the source after being aware of his parent's departure and decide a new parent as a result of Step 1~4 as described above. However, we consider that this takes too much recovery time and overhead.

Then, our proposal method is that all new participant hosts memorize their candidates of the parent which are found at Step 2, and request them to reconnect directly when their ancestors depart the tree as Figure 4 shows. In this way, descendants of the departing host only execute Step4 and may shorten the recovery time and overhead.

IV. EVALUATION

A. Construction phase

We evaluate our proposal system in ns-2 simulator [11]. The streaming rates of each layer are {125, 125, 250} (kbps), the streaming protocol is UDP, and the number of hosts of which degrees are assigned randomly is changed from 20 to 100. Under this condition, we measure three values which are average throughput of all hosts, average delay of each link and the number of control packets (overhead). We adopt the two

systems as object of comparison with our proposal method, RTT method (Figure 5) and Round-Robin (RR) method (Figure 6). The former is that the join-request is forwarded repeatedly until the leaf hosts which have at least one remaining out-degree and the participant decides its parent by measuring their RTTs. The latter is that the join-request is forwarded to a single child in turn until a leaf host is found which has at least 1 out-degree so that the load is not biased and the control packets are reduced.

Figure 7 shows importance of the in-degree restriction described in Section 3.1. When in-degree is not limited, the total throughput decreases as the number of nodes increases. This is because the hosts which have small out-degree are located in a higher-position of the tree, and the descendants of them cannot receive at the rate which they desire. Consequently, as the number of host increases, the ratio of the influenced hosts to the whole tree grows. Therefore, it is clear that in-degree of each host should be restricted.

Figure 8 shows that our proposal method has validity in terms of throughput in comparison with others. In our system, all hosts have proper parents and receive at the rate which they desire by using the redefined degree parameter. On the contrary, other systems have the described problem of 'improper relation', and some hosts can not be satisfied with the receiving rate. The more hosts join the tree, the less average throughput is obtained because the hosts which receive the high rate are only in higher-position of the tree. In this point, the proposal method has an advantage in scalability.
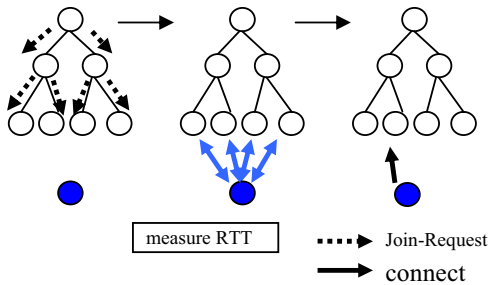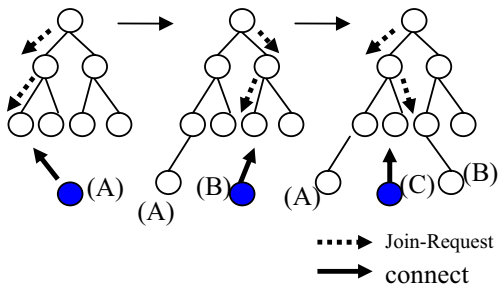

Figure 5. RTT method

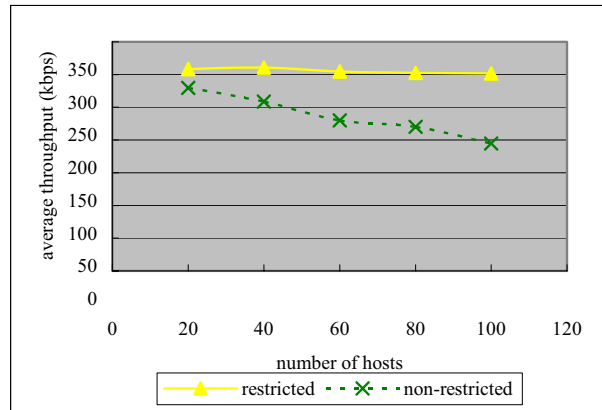Figure 6. Round-Robin method

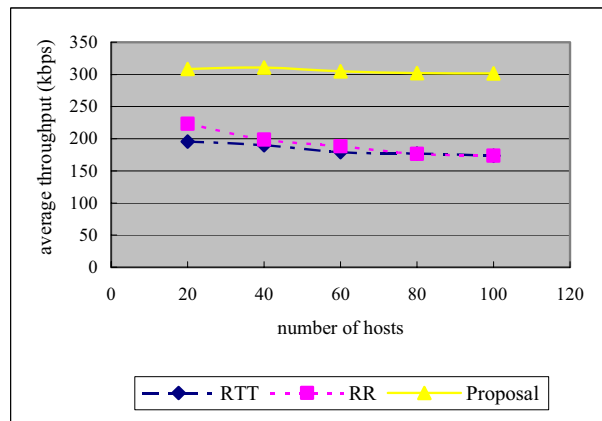
Figure 7. Average throughput comparison (1)

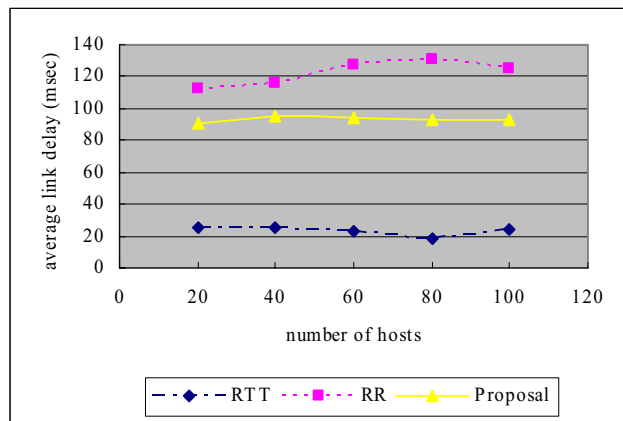Figure 8. Average throughput comparison (2)
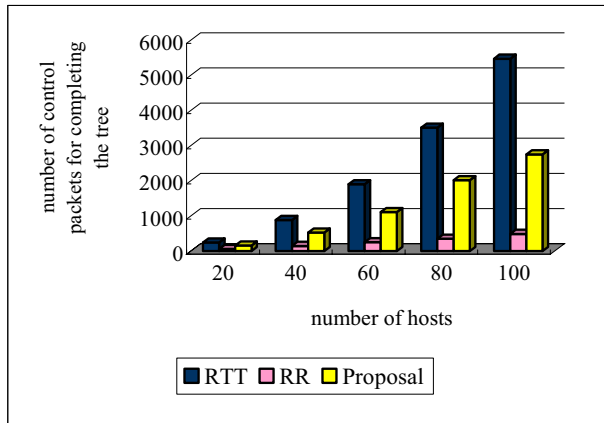

Figure 9. Average delay comparison

Figure 10. Overhead comparison

Figure 9 and Figure 10 show that the proposal does not necessarily sacrifice the delay and overhead. In terms of these points, we consider that our method has validity, too.

B. Recovery phase

Next, we evaluate the average recovery time and the overhead which affects the ALM network. The former is the time after the new parent of the descendants is found and reconnected when a host in higher position of the tree departs the ALM. The latter is the number of control packets for exchanging information of new relationship with the neighbors.
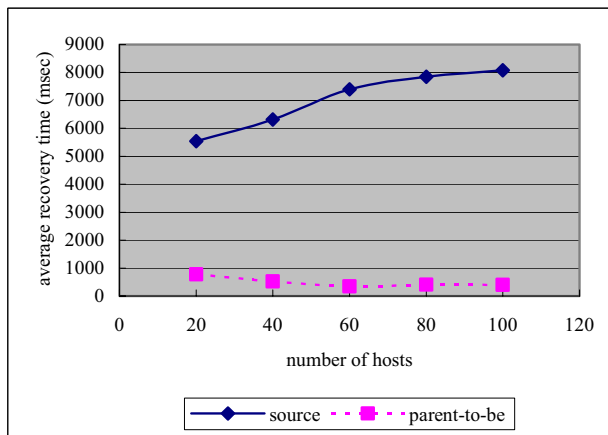

Figure 11. Recovery time comparison

As described in Section 3.3, we compared the two cases. First, all new participant hosts memorize their parent candidates as the parent-to-be which are found at Step 2, and request them to reconnect directly when their ancestors depart the tree. Second,

the descendants rejoin the source after being aware of his parent's departure and decide a new parent as a result of Step 1~4. We here assume the probability of host departing is 10%.

Figure 11 shows that using "parent-to-be" takes much less time to recover the tree. This is because it needs only the time for exchanging packets directly to the parent-to-be regardless of the network size. To the contrary, in the "source" method case, the new participant host stands by for about 5 seconds to collect sufficient number of candidates of the parent host. Although we can shorten the latency in return for decrease of the number of the parent candidates, quality of available routes decreases.
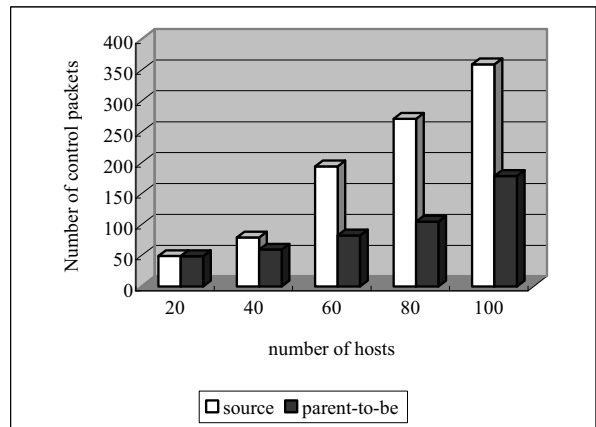

Figure 12. Overhead comparison (recovery phase)

Figure 12 compares overheads of the two methods in the recovery phase. The control packets are two kinds; one is for a join request message to rejoin the tree. This is redirected to the whole tree until the node's new parent is found. The other is to exchange information on the change of network due to joining and departing of hosts with the neighbors. The latter packet is needed for both ways and the former packets are little in small trees, then the number of control packets is about the same. To the contrary, when the number of hosts increases, more packets are necessary in the "source" method because the join request message spread the whole ALM tree.

C. Implementation

Moreover, we implemented our proposal into several computers and observed the time lag caused by tree reconstruction. In this experiment, video stream is encoded by H.263+ and the layering is carried out in a temporal scalable manner, by simply splitting I-picture packets and P- picture packets. Host computers are located inside our university campus.

Figure 13 shows an example, in which new participant C sends a join request to its parent host A and tree reconstruction begins. When the degree constraint is satisfied, host A temporally sends the stream to both B and C until host B starts

IEEE
COMPUTER
SOCIETY

to receive it from host C. The time lag happens due to tree reconstruction, and its length depends on sending interval of I-picture. In our experiment, a host transfers 10 frames per second, and inserts two I-pictures per second. Thus, the average latency until I-picture arrives is 0.25 sec (min is 0 sec, max is 0.5 sec). Observed video does not make us to feel serious quality degradation.
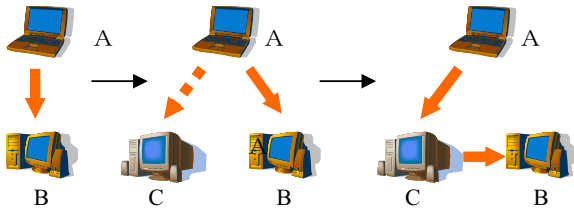
Figure 13. Measurement of the time lag caused by tree reconnection in implementation experiments

We then compared the proposal method and the RR method both of which are implemented into 20 hosts. The streaming rates of each layer are {125, 125} (kbps), and the host degree is assigned 1 or 2 randomly. Measured average throughput of all hosts is shown in Figure 14, where the simulation results on the same condition are displayed for the comparison. The results of the implementation are almost similar to those of the simulation.

From above-mentioned two points, we think that our proposal method clearly satisfies users' demands from the practical viewpoint.
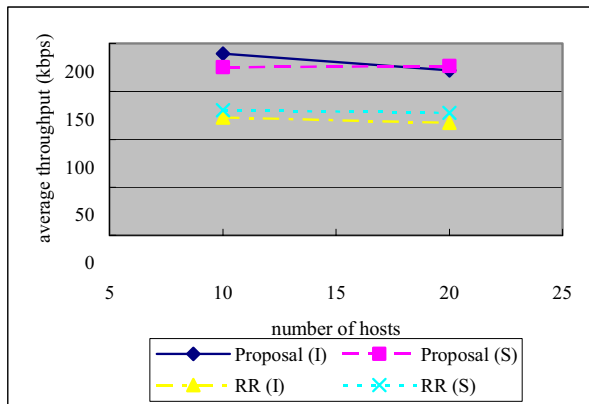
Figure 14. Average throughput comparison (3) (implementation vs simulation experiment, (I) is implementation and (S) is simulation)

V. CONCLUSIONS AND FUTURE WORK

We described a new approach to construct ALM trees for layered video stream. By adopting newly defined degree parameters and constructing trees according to those parameters, we verified that all hosts are easily satisfied with receiving rates which they desire. We have investigated its validity in terms of throughput, delay and overhead. Especially, throughput which should be given priority in streaming applications is much better than that of traditional way. Furthermore, notice that our method can be directly applied to the system using Layered-MDC (Multiple Description Coding) [13] [14], which is very flexible for network congestion control and heterogeneity of receiving environment. As future work, it is important not only to improve the stationary throughput performance but also to provide the robustness against the dynamic behavior of the ALM trees. Introduction of robust route maintenance approaches such as [10] [12] and network-wide evaluation via implementation are now considered.

REFERENCES

[1] A.Legout and E.W.Biersack, "PLM: Fast Convergence for Cumulative Layered Multicast Transmission Schemes," ACM SIGMETRICS 2000.
[2] I.Yamaguchi, K.Homma and J.Katto, "A Study on Receiver-Driven Layered Multicast with TCP-Friendly Rate Control," IEICE Tech Report, IN 2002-105 (in Japanese).
[3] Gu-In.Kwon and John.W.Byers, "Smooth Multiple Multicast Congestion Control," IEEE INFOCOM 2003.
[4] T.S.Eugene.Ng, Yang-hua.Chu, Sanjay.G.Rao, Kunwadee Sripanidkulchai and Hui Zhang, "Measurement–Based Optimization Techniques for Bandwidth-Demanding Peer-to-Peer Systems," IEEE INFOCOM 2003.
[5] John.Jannotti, David.K.Gifford, Kirk.L.Johnson, M.Frans.Kaashoek and James.W.O'TooleJr, "Overcast: Reliable Multicasting with an Overlay Network," Cisco Systems, Oct.2000.
[6] Reza Rejaie and Shad Stafford, "A Framework for Architecting Peer-to-Peer Receiver-driven Overlays," ACM NOSSDAV 2004.
[7] Reza Rejaie and Antonio Ortega, "PALS : Peer-to-Peer Adaptive Layered Streaming," ACM NOSSDAV 2003.
[8] Eli Brosh and Yuval Shavitt, "Approximation and Heuristic Algorithms for Minimum Delay Application-Layer Multicast Trees," IEEE INFOCOM 2004.
[9] Duc A.Tran, Kien A.Hua and Tai T.Do, "A Peer-to-Peer Architecture for Media Streaming," IEEE JSAC Special Issue on Advances in Overlay Networks, 2003.
[10] Mengkun Yang and Zongming Fei, "A Proactive Approach to Reconstructing Overlay Multicast Trees," IEEE INFOCOM 2004.
[11] "ns-2" http://www.isi.edu/nsnam/ns/.
[12] Y.Kunichika, T.Kusumoto, J.Katto and S.Okubo, "Application Layer Multicast with Proactive Route Maintenance over Redundant Overlay Trees," 2004 Packet Video Workshop.
[13] V. N. Padmanabhan, H. J. Wang, and P. A. Chou. "Layered Multiple Description Coding." 2003 Packet Video Workshop.
[14] V. N. Padmanabhan, H. J. Wang, and P. A. Chou. "Supporting Heterogeneity and Congestion Control in Peer-to-Peer Multicast Streaming," IPTPS, February 2004.