利用子群播為主的錯誤回復機制在可靠及連續性媒體資料傳播上之探討

王信仁[1] 林志彥[2]

1. 國立宜蘭技術學院電子工程系講師
2. 國立台灣科技大學電子工程系研究生

摘 要

在許多以重傳為主的可靠群播機制中，子群播(subcast)為主的錯誤回復機制是一種較新的做法。在此機制中，如何將一個大的群播群組切成子群組，再將這些子群組組織起來成為某一類型的邏輯子群組結構〔又稱為虛擬子群組結構〕，會對錯誤回復機制的效能有很大的影響。在本論文中，我們將探討在可靠及連續性媒體資料的傳播中，不同的子群組架構對於子群播為主的錯誤回復機制的影響。從實驗模擬結果中我們可了解一個好的子群組架構之組成要素。

關鍵詞：群播, 子群播, 子群組, 錯誤回復, 子群組結構

Issues in the Use of Subcast-based Multicast Error Recovery for Reliable and Continuous Media data Dissemination

Hsin-Jen Wang[1] and Chih-Yen Lin [2]

1. Lecturer, Department of Electronic Engineering, National Ilan Institute of Technology
2. Graduate Student, Department of Electronic Engineering, National Taiwan University of Science and Technology

Abstract

Subcast-based error recovery strategy is a recent and very promising development in retransmission-based error recovery strategies. In this strategy, organizing a large multicast group into logical subgroups and structuring them into some kind of logical subgroup order, called virtual subgroup structure (VGS), have important effects on system performance. In this paper, we investigate effects of subgroup structures on the performance of this class of subcast-based error recovery strategies to understand the underlying principles of subgroup formation for subcast-based error recovery, both for reliable and continuous media (CM) type of data dissemination. We use extensive simulations in our work. From the results of simulations, we demonstrate what constitutes good subgroup structures in general. We also give general statements of good retransmission-based continuous media content delivery strategies for both unicast-based error recovery strategy (such as STORM [9]), and subcast-based error recovery strategies.

Key Words：multicast, subcast, subgroup, error-recovery, subgroup structure

I. Introduction

With improved communication infrastructure and popularity of home PCs on the Internet, more and more applications are designed that place more stringent resource requirement on today's networks, such as video-conferencing, video broadcasts, audio broadcasts, collaborative applications, etc. Compared with IP unicast, IP multicast [10] can deliver the same information to multiple end users while consuming significantly less network resources. However, since IP multicast is based on the same underlying best-effort IP infrastructure, it doesn't provide any reliable data delivery guarantee. To ensure reliable delivery of data, a reliable multicast (RM) transport layer entity for error control has to be designed in the IP multicast protocol stack.

A number of previous works [1,2] showed that receiver-initiated RM protocols can alleviate the feedback implosion problem inherently presented in sender-initiated RM protocol. However, they also have serious problem of incurring potential deadlock conditions if they operate with only finite amount of buffer memory [4, 5, 6]. In TMTP [6], the infinite buffer requirement issue was solved by having receivers periodically send positive feedback acknowledgments (ACKs) to their designated representative (DR), which collects feedback information and represents the multicast subtree underneath the DR. In their case, the DRs are the receivers' immediate parents in the multicast delivery tree. The TMTP work marks the beginning of the RM design approach in which representatives are used to merge positive or negative feedbacks for retransmission-based error recovery strategies. These tree-based RM protocols [6, 7, 8, 9] divide the multicast group members into logical subgroups and distribute the error recovery responsibility over the tree structure. It has been shown that tree-based VGS strategies have the best performance and are the most scalable[8]. Clearly, how to organize the subgroups into efficient recovery structures is of vital importance in this approach. Previous study has obtained preliminary results on preferred organizations of subgroup structures [3] and provides guidance on the placement DRs on the multicast delivery tree to aggregate the feedback messages (ACKs and negative acknowledgments, NACKs).

A very recent proposal in the tree-based VGS family of algorithms utilizes a newly recognized functionality in multicasting – subcasting. Specifically, subcasting is multicasting over a subtree of the whole multicast delivery tree. Since packet losses are always spatially correlated [4] -- when error occurs in one location, the whole subtree must also be affected. Therefore, it is almost always worthwhile to multicast repair packets over the whole subtree under error occurrence point, rather than trying to repair packet losses individually one receiver at a time. However, in this case, there has been no systematic investigation of how logical subgroups should be organized to achieve the best error recovery performance. This topic is the focus of the first part of this paper. In this paper, we will systematically and quantitatively explore this issue in the very promising subcast-based error recovery and a very interesting unicast-based (STORM) error recovery strategy for tree-based VGS.

In [9,11,12,13], employment of retransmission-based error recovery strategies has also been investigated for continuous media (CM) applications. Because CM applications can tolerate a certain amount of packet loss, but have more stringent delay sensitivity, how to devise a strategy that minimizes packet loss and delay jitter for audio/video playback quality is an interesting direction attracting many researchers. In STORM [9], the authors proposed to use an approach similar to tree-based VGS error recovery in an attempt to achieve scalability and low error recovery latency for CM applications. However, their results are far from perfect from a practical standpoint. In this paper, we will show that subcast-based error recovery is a much more efficient even for CM data delivery. It can both minimize playout buffer size and maximize the effective goodput, and is much more practical than approaches such as STORM like tree-based error recovery strategies.

The rest of this paper is organized as follows. In section II, we will provide a brief review of the techniques of RM error recovery and subcasting. In section III, we will discuss two significant phenomena contributing to subcast-based RM error recovery strategies: early start effect and cascading delay. In section IV, we will demonstrate that performance of subcast-based error recovery is much better efficiency than other approaches and much more suitable for CM applications using retransmission based error control.

II. Background

1. Multicast subtree, subgroup, and VGS

A multicast delivery tree underlies a multicast group with the multicast sender at the root of tree and receivers distributed among the intermediate and leaf nodes. A multicast subtree is a portion of the multicast delivery tree. If a node is not in the leaf (a intermediate node) of tree then there exists a subtree with such node at the root of subtree.

A subgroup is a set of nodes, and we can arbitrarily assign nodes into such set. The smallest subgroup just contains one node that is a multicast group member. Each subgroup has a unique representative, called designated representative (DR). DR is responsible for handling the task of error recovery for members within the subgroup as much as it can. An example of a possible subgroup clustering is shown in Figure 1. In this Figure, members are divided into eleven subgroups and form a VGS. The eleven subgroups are also listed in table 1.

Each node within a VGS has to participate in two subgroups, one is the subgroup responsible by itself, and the other one being a upper-level subgroup, and A special case is that a subgroup just only one node and such node representation itself, e.g. node H, I, M, K, and N as shown in Figure 1.

2. Local error recovery using DR

Local error recovery for receivers is a mechanism through which a receiver can recover lost packets from a special node in the subgroup, its designated representative (DR), which supposedly should be another receiver closer to the source. In the case when the DR has the requested packet, the error can be repaired by the DR directly, without requesting a retransmission from the source. In this sense, errors can be repaired *locally*. In all DR-based RM protocols, it is important to assign DRs at strategic positions, either manually or algorithmically, for handling feedback from receivers and as a source of retransmission when the DRs possess the requested packets. This local error recovery mechanism has the potential of drastically reducing error recovery latency and alleviating the feedback implosion issue. For instance, in the example in Figure 1, node E serves as the DR for nodes K, L, and N. When node K, L, or N detects a packet loss, they try to repair the loss by sending a request to node E. If E has the requested packet, E can respond by sending the packet directly to the packet requester. Considerable less recovery latency can be obtained in this way than having the packet requesters try to repair the error by sending the request directly to the source node.

Obviously, the algorithm for formation of this receiver and DR relationship is of vital importance in this architecture since it has direct bearing on the error recovery performance. In table 1, we show the eleven DR-receiver relationships.

3. Subcasting

Subcasting is a facility that multicasts a packet over a portion of multicast tree, i.e. a subtree and can be implemented by a form of IP encapsulation [15]. Figure 2 illustrates the subcasting operation. Assume that a node in the subtree rooted at A detects a loss and makes a repair request, which propagates upward through C to B. After node B obtains the repair packet, it tunnels the packet via unicasts to the turning point of the subgroup rooted at node C, which multicasts the packet to the entire subtree under node C. Note that B's repair operation may repair multiple packet losses under the subtree, if there are other losses.

Subcast-based error recovery strategies, such as OTERS, utilize the property of multiple repairs with a single repair operation to achieve a much better performance among RM protocols. In these strategies, an additional notion of turning point (TP) is required. When a loss occurs, the repairer has to know where in the multicast delivery tree to send (subcast) the requested packet. To convey this information to the repairer, each node has an associated turning point, informing the repairer from which the repair packet should be subcast. For this purpose, a TP is associated with each subgroup to indicate to higher level subgroups the best place to inject the repair packet.

In OTERS and other DR-based RM protocols, members are clustered into subgroups. For the subtree rooted at node C, there are eleven subgroups as shown in Table 2. Each subgroup has associated with it two special nodes, DR, and DR's DR (myDR for short). DR is a member of the subgroup, and acts as a representative of the subgroup.

Even though the role of a DR in error recovery is different from those of the group members, the actual operation of DR's is no different from those of the group members in error recovery. Therefore, the variable myDR is introduced to uniformly represent the relationship between a subgroup member and its DR.

In Table 2, the column of "myDR" represents "the DR's DR" and "myTP" represents "the DR representation subtree". (TP stands for turning point that is a node at the root of subtree; the subtree notion only make sense for subcast-based error recovery strategy, otherwise is not make any sense).

Details of the operation for OTERS-class error recovery protocols using subcasting is demonstrated using the pseudo code listed in algorithm 1. Note that the error recovery stage occurs after tree-like VGS subgroups are already formed.

= *Algorithm **1** --- The algorithm for error recovery of subcasting*=

The variable **PakcetID** denotes the sequence number assigned to a packet, the variable **Packet** denotes the received packet, variable **TP** denotes a turning point node, variable **DR** denotes a unicast address of designated receiver. Each receiver has a timer handler for executing events when previously set timers expire. **myTP** is a TP that such a nodes is responsible for representative. **myDR** indicates the DR that is responsible for represents such subgroup member.

Furthermore, the following functions are used in the error recover algorithm.

HavePakcet(**PacketID**): For checking whether such node still keeps that packet, if it is then returns TRUE. Otherwise, it returns FALSE.

InIgnore(**packetID**,**TP**): When the repair packet is already sent and any requests for such a TP within the ignore period, $RTT_{requester-to-repairer}$, will be ignore, and this function will return TRUE. Otherwise, it returns FALSE.

SendRepairPkt(**PacketID,TP**): sends a IP encapsulation packet to TP. Then such TP will transform such packet into a multicast packet and multicast over subtree.

SendRequestPkt(**PacketID,myTP,myDR**): sends a request packet to its DR, **myDR**, and such packet contains **PacketID** and **myTP** information for repairing loss packet.

Reword(**Packet.**receiver,**Packet.**msg_multicast_Addr): reword the IP packet's receiver part information into the multicast address and port.

SendPacket(**Packet**): send the **Packet** out.

1. /* *Sender or DR side**/

[**If it received the requested packet then it sends the repair packet to the requester's TP, else it invokes ForwardRequest(PacketID).**]

```
RepairLoss(PacketID,TP){
  If (HavePakcet(packetID)){
    If (InIgnore(packetID,TP))/* within NACK suppression */
      /* do nothing */
    Else {
      SendRepairPkt(PacketID,TP);
      Set TimerRepair[PacketID][TP] = RTT_{requester-to-repairer};
      ScheduleEvent(TimerRepair[PacketID][TP],ignore);
    }
  }Else {
    If (AreadyRequest(PacketID))
      /* do nothing */
    Else {
      ForwardRequest(PacketID);
    }
  }
}
```

2. /* *DR side* */

[**Forwarding the request to upper layer DR and set the TP to its TP.**]

```
ForwardRequest(PacketID){
  RequestLoss(PacketID);
}
```

3. /* *Turing Point**/

[**When it Receive the IP encapsulation packet then it rewords to such multicast group address and sends the packet out.**]

```
ReMulticast(Packet){
   Reword(Packet.sender,Packet.msg_multicast_Addr);
   SendPacket(Packet);
}
```

4. /* *Receiver side or DR side* */

[**When the receiver detects packet loss, e.g. selective repeat, it will invoke the RequestLoss(PacketID).**]

```
RequestLoss(PacketID){
   SendRequestPkt(PacketID,myTP,myDR);
  /* exponential backoff */
   round[PacketID] =  round[PacketID] + 1; /* round[PacketID] is initiated to 0 */
  Set TimerRequest[PacketID] = 2^round × RTT_requester-to-repairer;
   ScheduleEvent(TimerRequest[PacketID],re-request);
}
```

= *End of Algorithm* =

III. Early Start Effect and Cascading Delay

There are two key factors for why subcast-based error recovery can outperform unicast-based error recovery.

1. Early Start Effect

The early start effect [8] means that the error recovery latency is shorter than one round trip time between repairer and requester. Such effect can significantly reduce the error recovery latency. In Figure 3, there is a chain like topology and a long delay link locates between node A and node B.

Figure 4 demonstrates the error recovery behavior for both unicast error recovery (the dotted lines) and subcast error recovery (the black thick lines). As we can see in the Figure 4, the same packet loss condition incurs difference error recovery latency, subcast is t1 and unicast is t2. Subcast outperforms than unicast, because of the subcast can "early start" to recovery the loss packet in preceding node for successive nodes, e.g. node B receive the node A repair packet, rep(n).

2. Cascading Delay Effect

Cascading delay is a normal phenomenon in retransmission based error recovery strategy. The multiple level hierarchy error recovery will incur this phenomenon since the downstream node have to wait the upper node for receiving the more upper node recovery. Such cascading delay effect is called "cascading delay" for short. Such an effect is illustrated in Figure4 as cascading delay tD.

The impact of cascading delay is different to early start effect. The early start effect can alleviate the variance of error recovery latency but the cascading delay is just the opposite. The worst case of subcast error recovery is equal performance with unicast error recovery, that is, out of the benefit of early start effect. Cascading delay poses the unicast-based error recovery are not suitable for Internet applications because packet loss rate and long delay link presence are disorderly. As a result, it is hard to devise an algorithm achieving well performance. Nevertheless, the early start effect alleviates the variance. Therefore, we can conclude principles for applying such strategy to Internet applications. These contentions will be explored on section IV.

IV. Evaluation

In this section, we designed three kinds of experiments to demonstrate several effects, namely subgroup structure effect, error recovery latency distribution, and error recovery effectiveness. Our experiments were evaluated on the NS2 [14], a network simulator, and the simulation code was extended from OTERS [8].

The general experiment environment about subgroup distribution, presence and locations of the long delay links is described in appendix A.1.

1. Subgroup Structure Effect

VGS is formed by the subgroups in the multicast group. As we can know the tree-based VGS, that is we form the subgroups into a tree like of hierarchy, has best performance in error recovery. Intuitively, the subgroup characteristic will be the immediate factor to influence the error recovery efficiency. Thus, we mean the error recovery efficiency is the error recovery latency, so the propagation delay in a link is the first characteristic and the diameter of subgroup (the hop count between furthest two nodes) is the second characteristic. Following this, we designed experiments as shown in appendix.

Form the result of experiment, we known that the long delay link and the diameter of subgroup influence the error recovery efficiency indeed. From Figure 5 we can observe three essentials. First, *when subgroup size increased incurs the average error recovery latency increased too*, because the request messages and repair messages have to travel longer distance when the subgroup size is growing. Second, *when the long delay link is closer to the source then it incurs the average error recovery latency also increased*, as most of subgroup distribution types in the average error recovery latency are L1>L2>L3>L4>identical. Thirdly, *when the long delay link is closer to DR also incurs the efficiency of error recovery reduction*, such as L3 at 1+1+2 subgroup distribution, L3 at 2+2 subgroup distribution, or L2 at 1+3 subgroup distribution.

Therefore, the subgroup formation on subcast-based error recovery should follow the listed principles for achieving better performance on RM or retransmission-based CM applications. 1) Tree-based VGS will has best performance and scalability, also agree with the guideline in [3], 2) subgroup diameter should be as small as possible, 3) the source and DRs of multicast tree shouldn't be located behind long delay link.

From the previous paragraph, we deem that the long delay link presence and diameter of subgroup are two significant factors for the subgroup formation. Thus, from the delay point of view, the diameter of subgroup and the long delay link presence are similar views since the diameter of subgroup indicates the request packet will travel long distance (long delay) to the DR of the subgroup.

The average error recovery latency against with subgroup distribution in unicast-based error recovery is shown in Figure 6. We could find that the three essentials, we described in previous paragraph, are not properly for unicast-based error recovery. But compare Figure 5 with Figure 6, the subcast-based error recovery has lower average error recovery latency than unicast-based one.

To analyze the Figure 6, we found that the smaller subgroups do not imply that the average error recovery latency is lower than the bigger subgroups, because subgroups may incur the cascading delay (without early start effect). That is to say, unicast-based error recovery has to trade-off among the diameter of subgroup, long delay link presence and degree of VGS level. Therefore, unicast-based error recovery is difficult to design an optimal subgroup structure formation algorithm for VGS.

Hereafter, we adopted L4 topology to evaluate later experiments because that is an Mbone-like topology (and this case is the worst case for subcasting when the links on periphery of multicast tree are very lossy). In our topology, we just only simulate the routing node so that the leaf nodes just like nodes outside U.S., and such links are crossed the Pacific Ocean or .the Atlantic Ocean that will be a long propagation delay link.

The experiment, the result was shown in Figure 7, was designed to stress the unicast-based error recovery is not suitable for Internet if we want to achieve a simple enough and optimal solution.

The result of this experiment was demonstrated that unicast-based error recovery is subject to influence by increased propagation delay on long delay link (we just increased the delay of link delay link from 1ms to 9ms propagation delay in topology L4, and all link's loss rate are 1%; the dotted lines represent unicast-based error recovery, and the black thin lines represent subcast-based error recovery). Therefore, the unicast-based error recovery is difficult to adaptive subgroup structure to achieve good performance since the nature of Internet is not regulation and changeability. On the other hand, unicast-based error recovery is more variation (sensitively) than subcast-based and is difficult to devise a simple VGS formation algorithm to achieve efficiently to recover the loss packet. Thus, subcast-based error recovery is more suitable for Internet applications than unicast-based error recovery.

2. Error Recovery Latency Distribution

Next, we will design experiment to explore subcast-based error recovery and unicast-based error recovery on CM like traffic (with time constraint). In the following experiments, we will explore the error recovery latency distribution. These experiments were done on L4 topology (as shown in appendix) with different link loss rate within each level. For instance, 2-2-2-18 means that level 1 link with 2% loss rate, level 2 link with 2% loss rate, level 3 link with 2% loss rate, and level 4 link with 18% loss rate. The corresponding loss rate on each level of node is about 2%, 3.96%, 5.88%, and 22.82% respectively. The Figure 8 demonstrates the results of these three kinds of loss distributions, namely 2-2-2-2, 2-2-2-18, and 5-5-5-5. The black lines indicate the CDF of subcast-based and the gray lines indicate the CDF of unicast-based.

Figure 8 shows that the early start effect not only alleviate the error recovery latency but also reduce the loss packet (as shown in Figure 8 – 2-2-2-2 and 5-5-5-5).

3. Error Recovery Effectiveness

We proposed an algorithm for evaluating the effectiveness on time constraint traffic. This algorithm is following the notion in paper [11, 13] and listing an algorithm in the algorithm 2. Our experiments were done L4 topology and the long delay link is 9 ms. The following experiments will demonstrate the subcast-based error recovery can both minimize the playout buffer size and maximum the effectiveness than unicast-based error recovery (e.g. STORM).

Figure 9 illustrated the related parameters in algorithm 2, as listed below. In our experiments we used the CBR with a pre-configuration transmission, i.e. ProfileRate, be the source traffic generator (here, we used CBR for convenience to evaluation). Figure 9 demonstrated that the packet a6 (the arrival time of **PacketID** equal to 6) was late for playback, and was *invalid*. The packets from a1 to a5 and a7 were *valid*, since they have arrived before (or equal to) the playback time.

= *Algorithm **2** --- The validation function for effectiveness* =

ProfileRate: the sending rate (e.g. PCM generates 160 bytes of data every **20 ms**).

BufferSize: the playout buffer size in receiver side (in our experiment we set it to a constant value during the experiment).

PropagationDelayOfEach: the propagation delay from the sender to receiver at such receiver first time receiving the data packet.

PacketID: each packet with a unique ID for identification the order of packet and it is a monotonous function.

ArrivalTime: the packet arrival time.

```
Set SetSPT = FALSE;
Set ValidCnt = 0;
Set InvalidCnt =0;

/*
When receiver receives a packet it can use this function to verify this packet is effectiveness or not.
*/
ValidateFuncation(ArrivalTime, PacketID){
If (SetSPT == FALSE) {
   /*initiate the Start Playout Time (SPT)*/
   PropagationDelayOfEach := ArrivalTime;
  SPT = PropagationDelayOfEach + BufferSize;
  ValidCnt = 1;
  SetSPT = TRUE;
} Else {
 If (PropagationDelayOfEach + BufferSize+ PacketID* ProfileRate) ≥ ArrivalTime) {
        /* The playout time of received packet is greater than or equal to packet arrival time*/
   ValidCnt = ValidCnt + 1;
```

```
 } Else {
     /*The playout time of received packet expired*/
   InvalidCnt = InvalidCnt + 1;
       }
}
```

= End of Algorithm =

We used previous algorithm to verify the effectiveness in a given playout buffer size. The effectiveness is defined as following:

$$Effectiveness = \frac{ValidCnt}{total\_transmitted\_packet} \quad \text{..............................(1)}$$

Figure 10 demonstrates the results of eight subgroup distributions in the loss distribution 5-5-5-5. It shows that the subcast-based can achieve a higher effectiveness than unicast-based at the same playout buffer size. In Figure 11, we also explored those three kinds of loss distributions, 2-2-2-2, 2-2-2-18, and 5-5-5-5. The subcast-based strategy outperforms unicast-based one among these three kinds of distributions. Therefore, the subcast-based error recovery strategy is a more suitable retransmission-based error recovery than unicast-based error recovery strategy for CM application over Internet.

V. Conclusions

Dissecting a large multicast group and organizing the resulting subgroups into some kinds of subgroup structure has important consequences on the performance of the error recovery strategies. In this paper, we mainly demonstrate the effects of network characteristics on tree-based VGS at subcast-based error recovery and unicast-based error recovery. Subcast-based error recovery can easily achieve the optimal VGS formation than unicast-based error recovery. It also outperforms than unicast-based error recovery in the error recovery efficiency.

We also demonstrated that the subcast-based error recovery strategy could both minimize the playout buffer size and maximize the effectiveness than unicast-based error recovery strategy.

Appendix

A.1 Experiment Design

We want to investigate several network characteristics, such as the reciprocal effect between locations of long delay and subgroup distribution types, the reciprocal effect between locations of lossy links with subgroup distribution types, and the effect of network size. We adopted a tree-like topology for our experiments because we only model the network routing nodes rather than the member distribution. The root of tree is the sender and other branch nodes and leaf nodes are receivers.

Subgroup Distribution

Various subgroup distributions are investigated. The subgroup distribution follows the binomial enumeration of number as demonstrated below. For example: for a tree of height 4 (4-level):

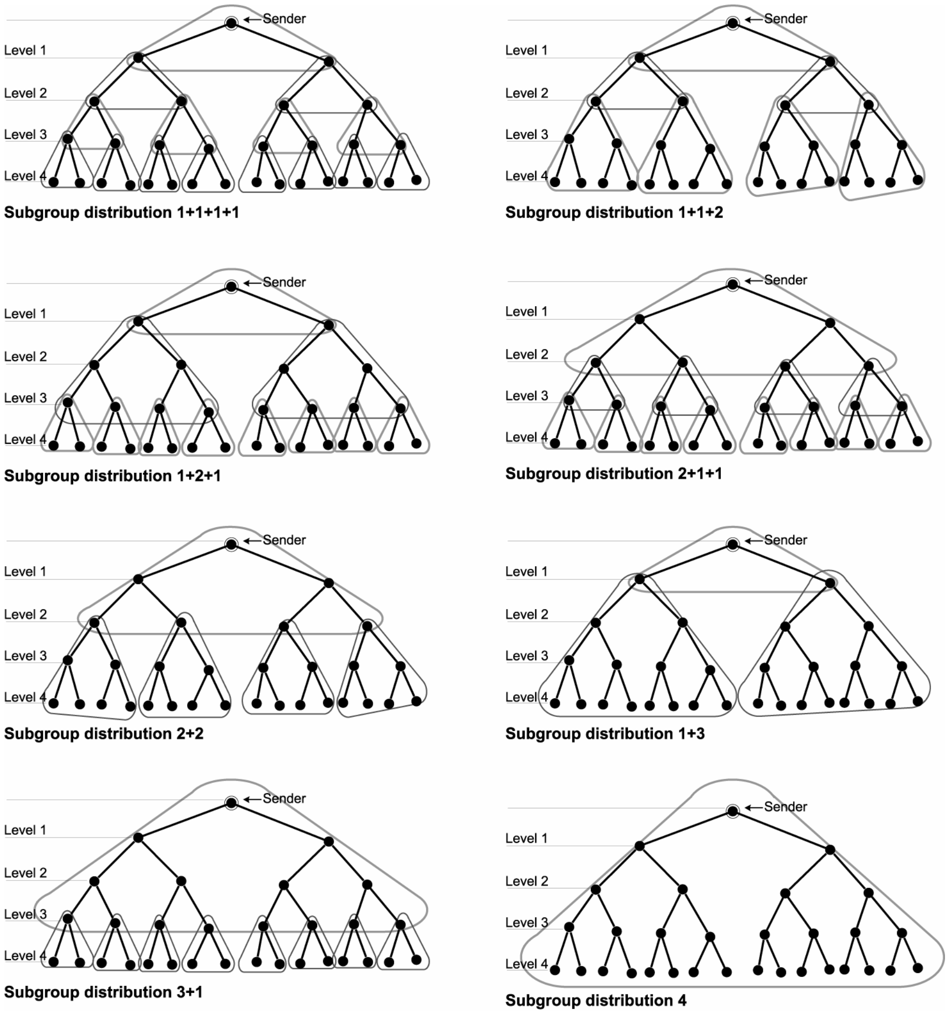

Fig A.1　The subgroup distribution illustration.

Presence and Locations of Long Delay Links

We make successive links at each level of the tree to be a link with a longer propagation delay, in 4-level tree topology resulting in four different kinds of topologies to experiment with it as shown in Figure A.2. The thick lines in each figure are long delay links.

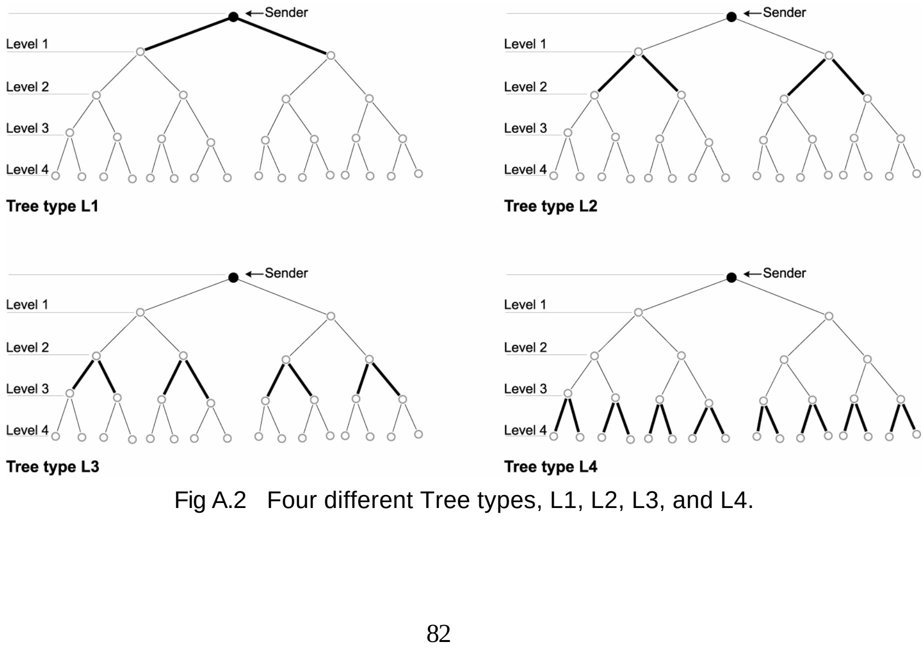

Fig A.2　Four different Tree types, L1, L2, L3, and L4.

References

1. Sridhar Pingali, Don Towsley, and James F. Kurose. (1994) “A comparison of sender-initiated and receiver-initiated reliable multicast protocols” In Proc. of ACM SIGMETRICS '94, volume 14, pp. 221-230
2. B. N. Levine, and J. J. Garcia-Luna-Aceves. (1998) “A Comparison of Known Classes of Reliable Multicast Protocols”, Multimedia Systems Journal(ACM/Springer),6(5).
3. M. Hofmann, M. Rohrmler. (1997) “Impact of Virtual Group Structure on Multicast Performance”. Fourth International COST 237 Workshop, December 15-19, 1997, Lisboa, Portugal, Ed.: A. Danthine, C. Diot: From Multimedia Services to Network Services, Lecture Notes in Computer Science, No. 1356, pp. 165-180, Springer Verlag,.
4. B.N. Levine, S. Paul, and J.J. Garcia-Luna-Aceves. (1998) “Organizing multicast receivers deterministically according to packet-loss correlation”. Proc. Sixth ACM International Multimedia Conference (ACM Multimedia 98).
5. Brian Neil Levine, (1999) “Network Support for Group Communication”, PhD dissertation.
6. Rajendra Yabatkar, James Griffioen, and Madhu Shdan. (1995) “A Reliable Dissemination Protocol for Interactive Collaborative Applications”. Proc. the ACM Multimedia'95 Conference.
7. S. Paul, K. K. Sabnani, J. C. Lin, and S. Bhattacharyya. (1997) "Reliable Multicast Transport Protocol (RMTP)," IEEE Journal on Selected Areas in Communications, special issue on Network Support for Multipoint Communication, vol. 15, no. 3, pp. 407-421.
8. D. Li and D. R. Cheriton. (1998) “OTERS (On-Tree Efficient Recovery using Subcasting): A Reliable Multicast Protocol”, Proceedings of 6th IEEE International Conference on Network Protocols(ICNP'98). Austin, Texas, pp 237-245.
9. R. X. Xu, A. C. Myers, H. Zhang, and R. Yavatkar. (1997) "Resilient multicast support for continuous media applications", Proc. NOSSDAV '97 , pp. 183-194.
10. S. Deering. (1989). “Host extension for IP multicasting”, RFC-1112.
11. M. T. Lucas, B. J. Dempsey, and A. C. Weaver, (1997) "MESH: Distributed Error Recovery for Multimedia Streams in Wide-Area Multicast Networks", International Conference on Communication, ICC'97, vol. 2, pp. 1127-1132, Montreal, Canada.
12. S. Pejhan, M. Schwartz, and D. Anastassiou, (1996) "Error control using retransmission schemes in multicast transport protocols for real-time media", IEEE/ACM Transactions on Networking, vol. 4, no. 3, pp. 413-427.
13. B. Dempsey, J. Liebeherr, and A. Weaver, (1996) “On Retransmission-Based Error Control for Continuous Media Traffic in Packet-Switching Networks”, Comp. Networks and ISDN Sys. Vol. 28, no.5, pp. 719-736.
14. http://www-mash.cs.berkeley.edu/ns/
15. C. Perkins, (1996) “IP Encapsulation within IP”, RFC 2003.

91 年 08 月 28 日投稿
91 年 09 月 15 日接受

Table 1　Eleven subgroups within the subtree rooted at node A of Fig. 1.

Subgroup	DR	TP	Group member
1	**H**	H	H
2	**I**	I	I
3	**K**	K	K
4	**M**	M	M
5	**N**	N	N
6	**G**	G	G, H, I
7	**L**	J	L, M
8	**F**	D	F, G
9	**E**	E	E, L, K, N
10	**C**	C	C, E, F
11	**B**	A	B, C

Table 2　myTP and myDR for each multicast member in Fig. 1.

Node	myTP	myDR
H	H	G
I	I	G
K	K	E
M	M	L
N	N	E
G	G	F
L	J	E
F	D	C
E	E	C
C	C	B
B	A	-

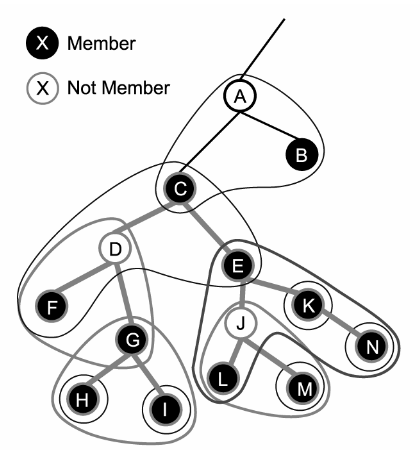

Fig 1　The VGS within a portion of multicast tree.

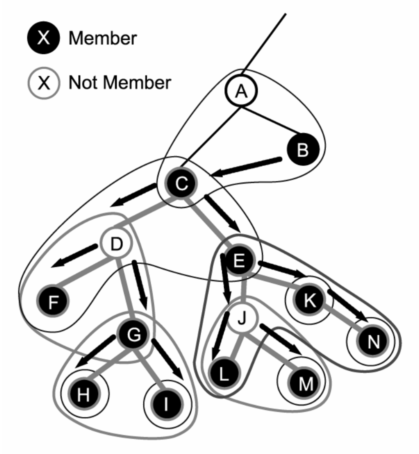

Fig 2　An example of subcasting.

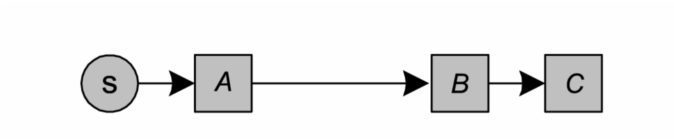

Fig 3　There is a chain like topology and a long delay link locates between node A and node B.

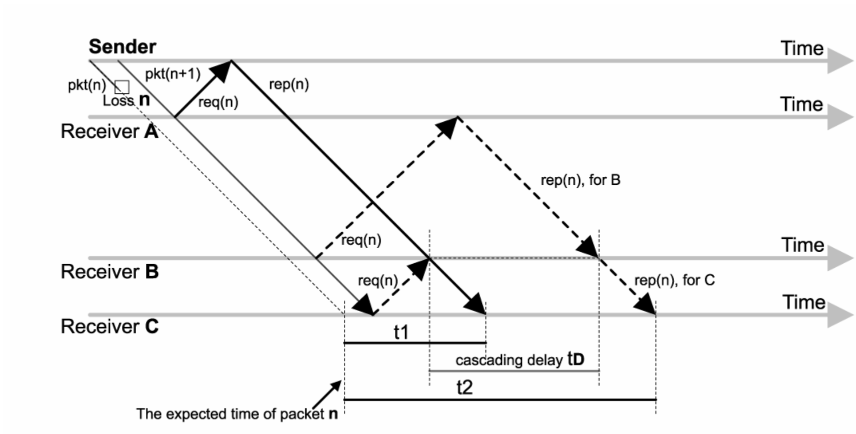

Fig 4　The timing diagram about the error recovery behavior both unicast (the dotted lines) and subcast (the black thick lines).

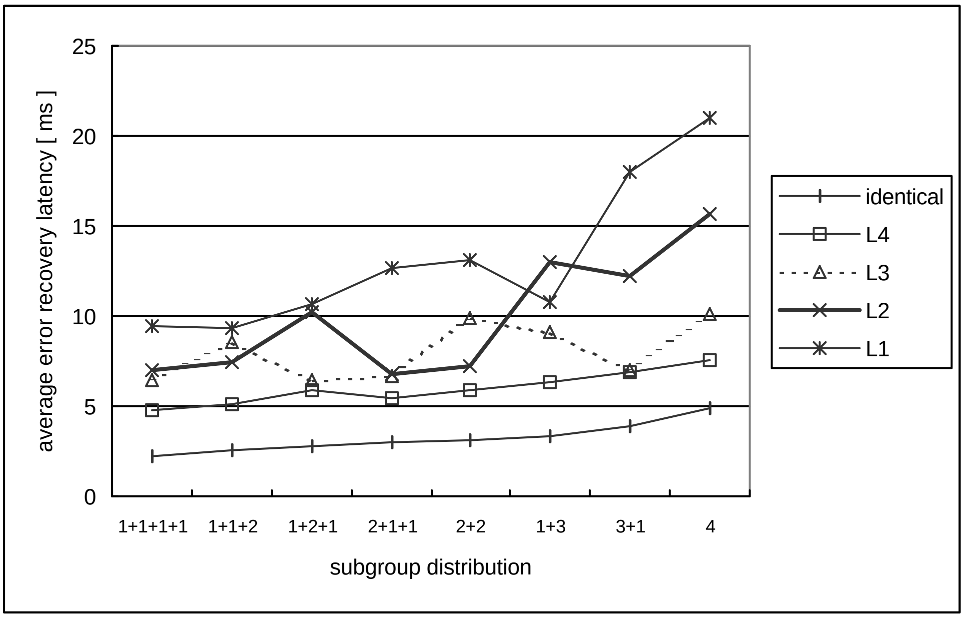

Fig 5 The average error recovery latency against with subgroup distribution in subcast-based error recovery.

Fig 6 The average error recovery latency against with subgroup distribution in unicast-based error recovery.

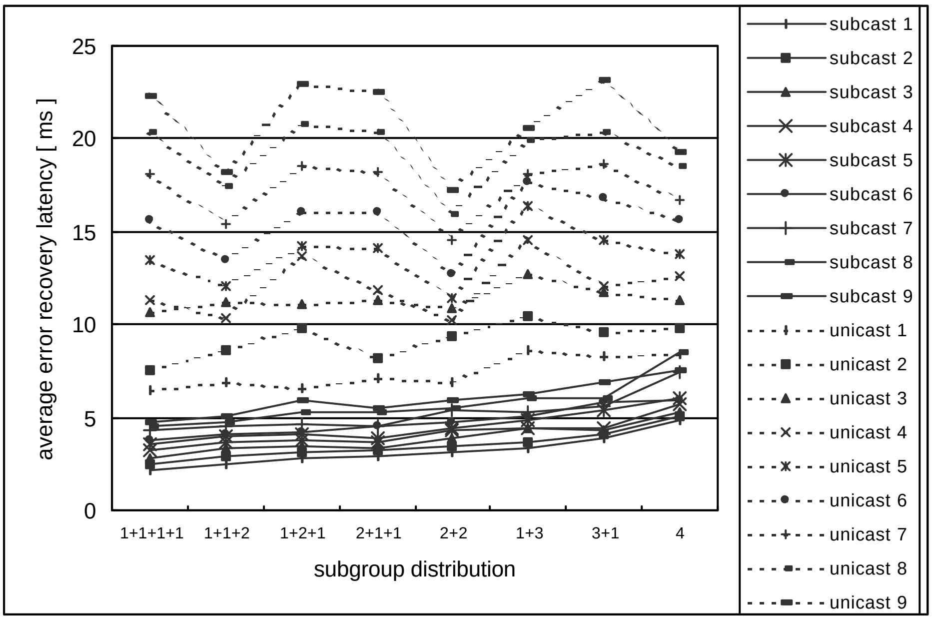

Fig 7 The average error recovery with increasing delay at the link in L4.

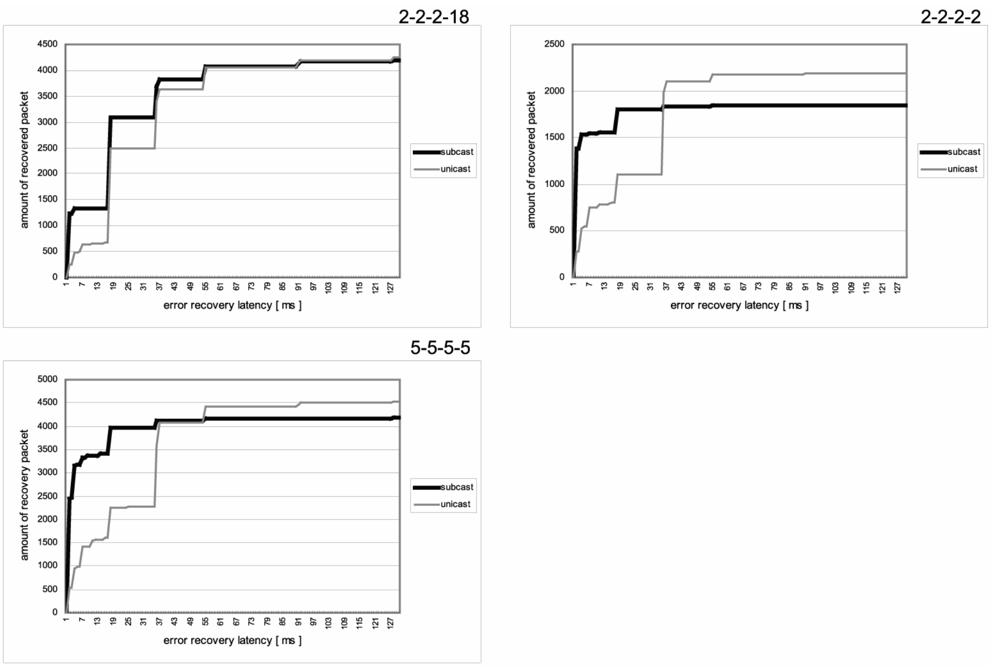

Fig 8 The CDFs of error recovery latency.

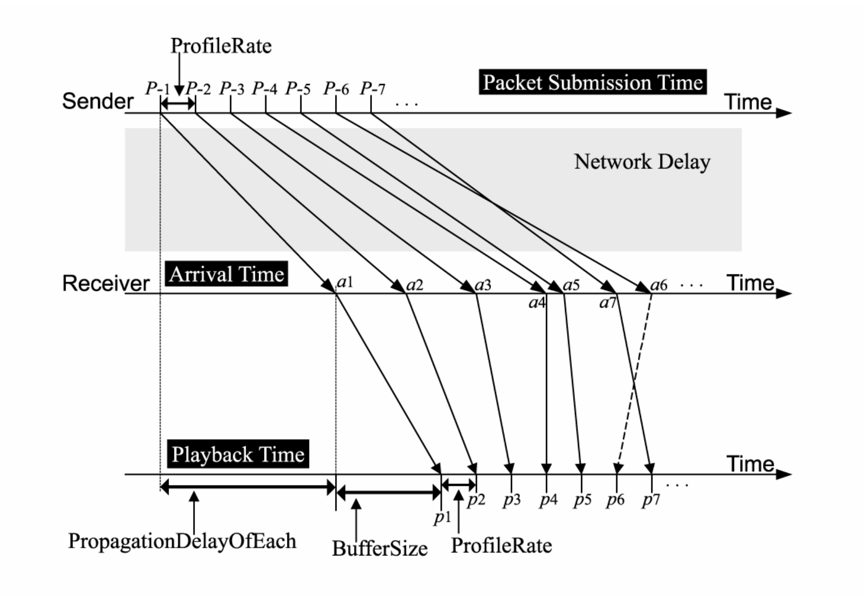

Fig 9　The transmission model for evaluating the effectiveness.

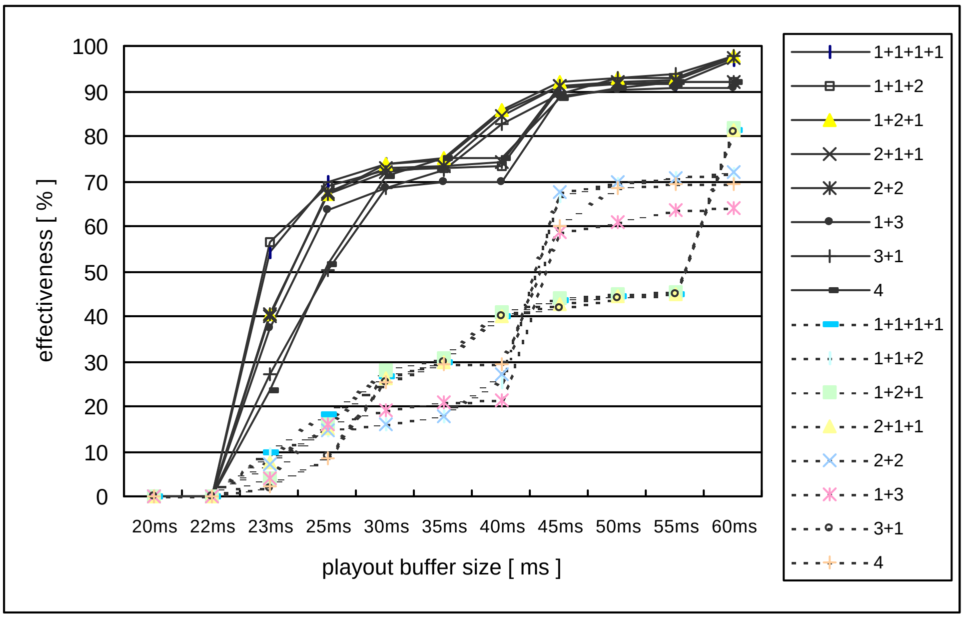

Fig 10　The effectiveness error recovery in L4, the dotted lines represent unicast-based, and the black thin lines represent subcast-based.

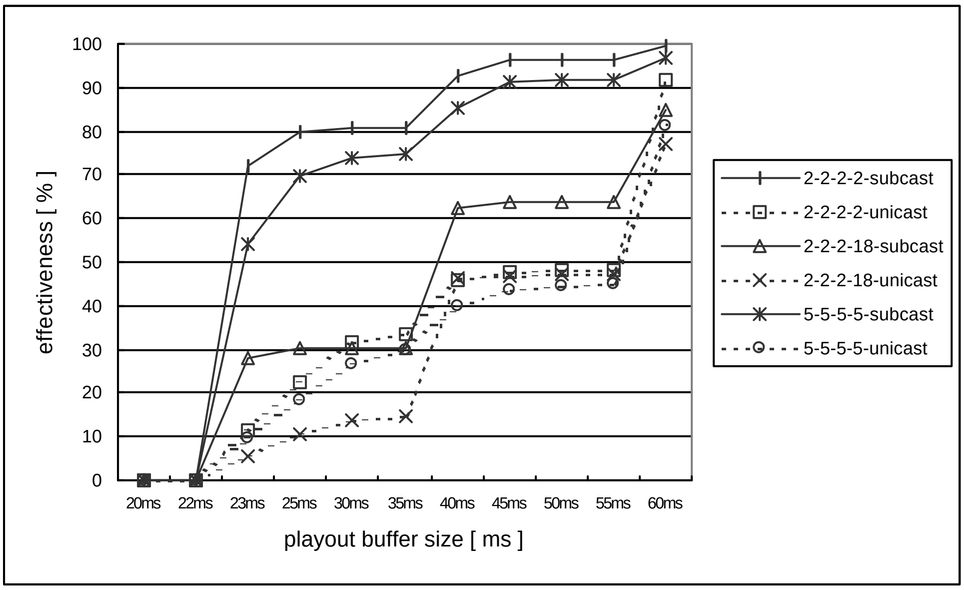

Fig 11　The effectiveness error recovery in L4 at 1+1+1+1 subgroup distribution.