Rate Control Over Repair Traffic in Multicast

Jun Peng and Biplab Sikdar
Electrical, Computer and Systems
Engineering Department
Rensselaer Polytechnic Institute
110 8th Street
Troy, New York 12180
e-mail: pengj2,sikdab@rpi.edu

Abstract —

Although the rate of repair traffic is generally well controlled in unicast, little attention has been paid to the rate control over repair traffic in the current literature of distributed multicast loss recovery. In this paper we show that rate control over repair traffic in multicast is necessary for traffic stability and congestion alleviation. We also propose a general method for controlling the rate of repair traffic in multicast, which can be used to enhance any of the existing distributed multicast loss recovery schemes.

I. Introduction

Packet loss recovery is necessary for achieving reliable network communications, while rate control is indispensable to keep networks in efficient condition. In general, congestion is the dominant source of packet losses in current networks. When congestion occurs, packets get lost due to queue overflow. The lost packets must be resent for those receivers that do not receive the lost packets but require reliable data transmission. Meanwhile, rates of flows going through the congested link must also be controlled to alleviate the congestion. Because of the routing protocols implemented in current networks, the repair packets for losses at a congested link usually need to go through the same link to reach receivers that experienced the losses. Therefore, the rate of repair traffic also needs to be controlled properly if high network utilization is to be maintained.

In unicast, the repair traffic and the original traffic are usually controlled together by a single scheme. This is possible because the repair traffic and the original traffic usually originate from the same source in unicast. In multicast, repair traffic does not necessarily come from the same source as original traffic does. It has been shown that distributed multicast loss recovery schemes usually outperform concentrated schemes in many aspects such as efficiency and recovery latency [1]. In distributed multicast loss recovery schemes such as [3] [2] [4] [11] [5], each receiver or each agent/server across the multicast tree plays an active role in loss recovery. In some schemes, receivers or agents/servers are also potential repair sources. In this case, the rate control over repair traffic must also be distributed if it is implemented. Existing distributed multicast loss recovery schemes generally do not specifically control the rate of the repair traffic generated by repair sources across the multicast tree (except for avoiding duplicate repairs). Additionally, almost all existing multicast congestion control schemes such as [7] [8] also do not consider the rate control over repair traffic when combined with a distributed multicast loss recovery scheme to form a complete reliable multicast transport protocol. Further, these congestion control schemes also can not be simply extended to control the rate of repair traffic. This is because with these schemes the original multicast traffic is usually specially organized at the multicast source in assisting rate control, while repair traffic can hardly be organized in a similar way in distributed loss recovery.

In this paper we show that in order to alleviate network congestion and achieve stable throughput for multicast receivers, a complete multicast transport scheme that combines congestion control and distributed loss recovery must have a mechanism to control the rate of repair traffic. We show that uncontrolled repair traffic may worsen the situation of a congested link and consequently decrease network utilization. Furthermore, the congestion control in the combined transport scheme may over-respond and render higher fluctuation in traffic. We also propose a general and effective mechanism in controlling repair traffic rate in multicast, which is to control the submission of repair requests.

The rest of the paper is organized as follows. We analyze the issue of rate control over repair traffic in multicast in Section 2. In section 3 we show some simulation results. Summaries appear in Section 4.

II. The Issue and The Solution

A The Issue

In general, both rate control and loss recovery need to be applied to a session when the traffic of the session goes through some congested links. The goal of the former is to alleviate congestion, while the latter recovers lost packets for receivers who require reliable data transmission. So a transport protocol usually incorporates both a rate control scheme and a loss recovery scheme. For example, in TCP lost packets are resent from the original source and the repair traffic is controlled together with the original traffic by an AIMD scheme. This kind of combination of the control over repair traffic with the control over original traffic generally can be introduced to any unicast transport protocols, where the repair traffic and the original traffic usually originate from the same source.

Rate control and loss recovery in multicast are much more complicated than in unicast. In unicast, there is only a single receiver to cater to, while in multicast there are possibly thousands of receivers awaiting data and these receivers may spread across the earth. Few existing multicast congestion control schemes specifically consider controlling the repair traffic generated by distributed loss recovery when combined with a distributed loss recovery scheme. Furthermore, these schemes usually can not be directly extended to control repair traffic in multicast. This is because with these schemes the original multicast traffic is usually organized in a special way to assist rate adjustment, while repair traffic generated by distributed loss recovery generally can not be organized in a similar way. In addition, all existing distributed multicast loss recovery schemes generally do not specifically control the rate of the repair traffic generated by them across a multicast tree (except for avoiding redundant repairs). This is possibly because rate control is usually considered the job of a congestion/flow control scheme but not the job of a loss recovery scheme. In summary, the rate control over repair traffic generated by distributed loss recovery is considered neither by existing multicast loss recovery schemes nor by existing multicast congestion control schemes.

Without proper rate control over repair traffic, there may be significant problems if a multicast transport scheme is to be created by combining one of the existing multicast congestion control schemes and one of the existing distributed multicast loss recovery schemes. The repair traffic produced by a distributed loss recovery scheme may be heavy in periods of severe congestion, because in severe congestion there are huge losses. Furthermore, existing multicast congestion control schemes are usually slow in reducing rate. There are two main reasons for this:

- Pruning in multicast routing is usually slow because of the IGMP protocol implemented [12], while most multi-rate multicast congestion control schemes use pruning to drop layers in reducing traffic rate.
- Most multicast congestion control schemes try to filter transient congestion for stabilizing traffic, so observation time is usually considerably long before rate control is applied to traffic.

With uncontrolled repair traffic and delayed congestion control over original traffic, congestion at bottlenecks can not go away fast. The consequence is low network utilization. In addition, the congestion control over original traffic may over-respond because of the interference from uncontrolled repair traffic, so the original traffic may show higher fluctuation in rate. High traffic fluctuation may be a serious disadvantage for some applications such as streaming media.

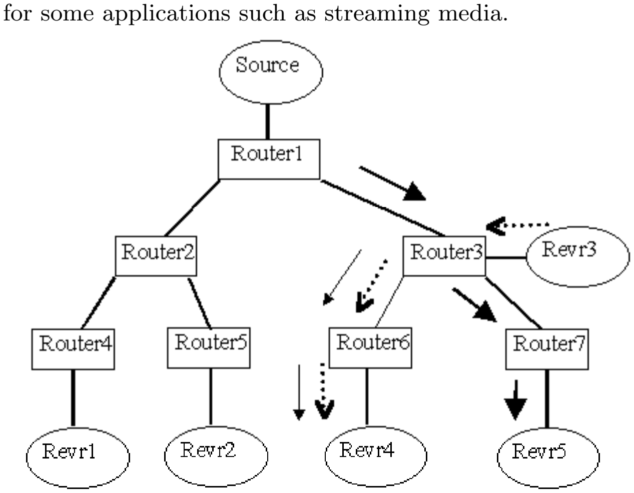

Fig. 1: Issue Analysis

We can further analyze the issue by trying to form a multicast transport scheme by combining one of the existing multicast congestion control schemes and one of the existing multicast loss recovery schemes. For example, let us consider the scenario of a streaming video multicast on the multicast tree in Fig. 1. We may choose RLM [7] as the congestion control scheme and encode the original video (the original traffic, OT) into M layers . For loss recovery, we only consider protecting some high-priority layers such as layer 1 to layer N, since streaming video multicast can generally tolerate some losses and network resources are usually limited. We may choose SRM [3] as the loss recovery scheme for protecting layer 1 to layer N. Although SRM is not necessarily an ideal loss recovery scheme for streaming video because of its relatively long recovery latency, our choice does not weaken our analysis of the issue. As a matter of fact, with a more responsive loss recovery scheme, repair traffic will reach the congested link more quickly, so it is more likely that the congestion will be worsened by repair traffic. Now we consider what will happen if the link between Router3 and Router6 (LK36) becomes congested for a period of time. Receiver3 is the receiver that is located closest and upstream to LK36, so most probably Receiver3 will be the repair source. In this case, Receiver3 will generate repair traffic (RT) that spreads across the multicast tree and also goes through the already congested link LK36 to reach Receiver4. In general, RLM does not control the rate of the repair traffic generated by Receiver3, while SRM also does not specifically control the repair traffic rate. So what will be the consequence? We can assume that the congestion event begins at $t1$ and ends at $t2$. Some packets then get lost after $t1$. Packet losses bring the loss recovery scheme, SRM, into action. So Receiver4 reports losses to the group and Receiver3 starts to generate repair traffic some time later. We can assume that the repair traffic arrives at LK36 at $t1 + \delta$. Usually δ is small because of the introduction of distributed loss recovery, which enables local receivers/agents to provide repair packets. So it is highly possible that $t1 + \delta$ is less than $t2$:

$$t1 + \delta < t2$$

Furthermore, it is also highly possible that congestion control has not been in effect at $t1+\delta$, because δ is small and existing multicast congestion control schemes such as RLM usually are slow in reducing traffic rate. In this case, the total traffic rate, R, at the bottleneck is increased by the repair traffic rate, R_{RT}. So now the total traffic rate at the bottleneck becomes:

$$R = R_{OT} + R_{RT}$$

With a heavier traffic at LK36, the congestion there is worsened. With a worsened congestion, more packets get lost. With more losses, the repair traffic rate, R_{RT}, becomes higher. Therefore, R is further increased by R_{RT}. The increased R worsens the congestion further and causes more losses, which means heavier repair traffic (higher R_{RT}) a moment later. This devastating recursive process repeats until the congestion control scheme succeeds in reducing the rate of the original traffic, R_{OT}. Network utilization is reduced in this process because heavier congestion generally means lower network utilization.

Moreover, because the traffic rate, R, at the bottleneck, LK36, has been increased recursively by the repair traffic before the congestion control scheme, RLM, is in effect, RLM may drop more layers of the original video traffic than necessary. For example, without the interference of repair traffic, the congestion can be alleviated by dropping m layers. Because of the recursively increasing repair traffic, it is possible that another n layers also have to be dropped to alleviate the congestion, so the number of dropped layers increases from m to $m + n$:

$$m \Longrightarrow m + n$$

This excessive dropping of layers usually causes higher traffic fluctuation for receivers, in our case, for Receiver4. For streaming video multicast, higher traffic fluctuation leads to poorer viewing experience.

In our analysis above we only use a very simple scenario. In reality the situation may be much worse. There are two reasons:

- There may be many concurrent congestion events on a multicast tree, so a congested link may be under the influence of several repair sources.
- Control traffic of loss recovery schemes and congestion control schemes may also spread across a multicast tree, so bottlenecks may also be under the influence of control traffic.

B The Solution

There are two directions in seeking solutions to the problem of rate control over repair traffic in multicast. The first one is to modify the existing multicast congestion control schemes, while the second one is to modify the existing multicast loss recovery schemes.

The first direction seems reasonable because rate control is generally regarded as the responsibility of a congestion/flow control scheme, but actually it would not be easy to work out a solution in this direction. Existing single-rate multicast congestion control schemes are usually concentrated, so it is almost impossible for them to control the distributed repair traffic generated by distributed loss recovery. In existing multi-rate multicast congestion control schemes, the original data are usually organized in a special way at a single multicast source to help receivers adjust their receiving rate distributedly. With distributed loss recovery, the repair traffic generally can not be organized in a similar way. Mainly there are 3 reasons:

- With distributed loss recovery, there may be hundreds of potential repair sources instead of a single one.
- In some schemes, the repair sources may be unknown before they actually generate repair traffic with the occurrence of losses.
- It can not be predicted which packets will get lost before losses actually occur.

Therefore, it is almost impossible to organize the repair traffic generated by distributed loss recovery in a special way.

Although there is a type of multicast loss recovery schemes with some exceptions, these schemes have their own problems. With layered FEC loss recovery schemes such as [10] [9], the multicast source is also the repair source, and the only one. Additionally, the repair traffic is also organized in a similar way as the original traffic is organized. Therefore, the rate of repair traffic and the rate of original traffic can be controlled in similar ways with these layered FEC schemes. However, these schemes have a serious problem. Because they also use branching and pruning in adjusting repair traffic rate, these schemes are also affected adversely by the pruning delay in multicast. The pruning delay generally affects traffic rate adjustment seriously with these schemes.

The second direction for seeking solutions is to modify existing multicast loss recovery schemes. Although this direction is not traditional, it may be a prospective one. The reason is that a loss recovery scheme has more knowledge about its loss recovery process than any other independent congestion control schemes. In a loss recovery scheme, repair traffic sources can control their repair traffic directly. Additionally, request sources can also control repair traffic, but indirectly, by controlling the frequency and time of submitting repair requests. With the latter method, a repair request buffer is required at each request source. We believe request sources are the better candidates for controlling the rate of repair traffic. There are two reasons:

- A request source can retrieve congestion information from the traffic coming from the bottleneck, while a repair source does not have the traffic to analyze. So congestion feedbacks are not needed if request sources are responsible for controlling the rate of repair traffic.
- If repair sources are responsible for controlling repair traffic, they have to introduce some delay in responding to repair requests for adjusting the repair traffic rate. Without additional procedures, this delay in response may cause problems for request sources, because in this case request sources have no way to know whether their requests or corresponding repairs have been lost in transmission or the response to their requests has been intentionally delayed by repair sources.

The key mechanism in controlling the rate of repair traffic at request sources is to control the frequency and time of submitting repair requests. In almost all existing multicast loss recovery schemes, repair requests are sent as soon as possible after losses have been detected (for some other reasons such as filtering out redundant requests but not for controlling the rate of repair traffic, repair requests may be intentionally delayed). On one hand, the quick response to losses may reduce recovery latency. But on the other hand, uncontrolled quick response to losses may worsen the situation of the congested link during periods of severe congestion. When a receiver detects huge losses, it is highly possible that severe congestion is going on somewhere along the path reaching the receiver. If the receiver sends repair requests immediately, the heavy repair traffic may arrive at the congested link before the congestion has been eased by the congestion control scheme. With this observation, some control over the sending of repair requests can help in decreasing the interference from repair traffic on the congested link. The mechanism is to buffer repair requests and send them out only after congestion has been eased. In fact, the rate of repair requests should be low when the rate of losses is high, since heavy losses usually mean severe congestion.

We can take a look at the example in the subsection above. If the submission of repair requests is properly controlled, Receiver4 will hold the requests when the congestion is still severe (sensed with heavy losses). Only after the congestion control scheme has been in effect and the congestion at LK36 has been alleviated, Receiver4 starts to send out repair requests. In this way, the total traffic rate, R, at LK36 will not devastatingly increase during congestion. Instead, the rate will decrease after the congestion control has been in effect. When the repair traffic, RT, from Receiver3 arrives, the rate of the original traffic, R_{OT}, has already been properly reduced. Therefore, the congestion at LK36 is effectively controlled.

One thing worth mentioning is that in some applications such as streaming media, a repair packet will be useful only if it arrives at a receiver within a time range. Although this requirement on recovery latency seems contradictory to the proposed mechanism of delaying repair requests, it is not, actually. If request sources do not pay attention to the current condition of congestion and send out repair requests as soon as possible to reduce recovery latency, as introduced above, the congestion that causes the losses may be worsened. With worsened congestion, queuing delay will increase and more packets, including repair packets, will get lost. Therefore, in this case, not only is the goal of reducing recovery latency not achieved, but the quality of the communication is further degraded.

In summary, rate control of repair traffic in multicast is a special issue. Because of the introduction of distributed loss recovery in multicast, repair traffic may originate from hundreds or thousands of potential sources. So rate control of repair traffic in this case should also be distributed. Existing multicast loss recovery schemes and congestion control schemes usually do not specifically control the rate of repair traffic if distributed loss recovery is used. We analyzed above that uncontrolled repair traffic in multicast may worsen congested links and consequently cause lower network utilization and higher fluctuation in traffic. We also showed that to modify existing loss recovery schemes is a good direction. Specifically, to control the submission of repair requests is a general and feasible method. In next section we use simulations to support our analysis.

III. Simulation Results

In this section we use simulations to further illustrate the issue of rate control over repair traffic in multicast. The streaming video multicast example introduced in the last section is used in our simulations, but the network topology is simplified here to only keep the parts related to our analysis. The simplified topology is shown in Fig. 2.

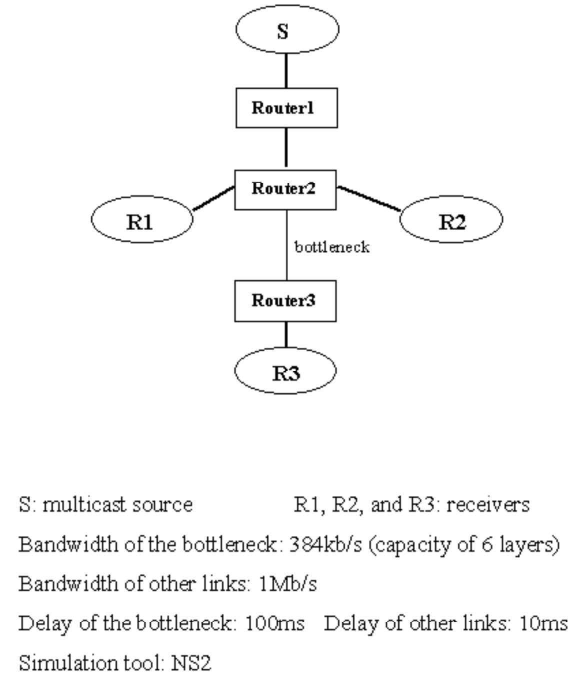

Fig. 2: The Network Topology for Simulations

In our simulations, the original multicast source S has 9 layers of video data and each layer has a rate of 64kb/s. The lowest 2 layers, layer 1 and layer 2, are protected from losses by SRM. As indicated before, although SRM is not necessarily an ideal candidate for our scenario for its relatively long recovery latency, the choice of it does not weaken our analysis, because a more responsive loss recovery scheme will let repair traffic arrive at the congested link faster and worsen the congestion with higher possibility. The protection of two layers also does not weaken our analysis, because to protect more layers means heavier repair traffic and therefore heavier interference. The bottleneck between Router2 and Router3 is set to a capacity of 6 layers: 384kbs. To generate a period of congestion at the bottleneck, a UDP session from S to R3 starts at the 150th second of the simulation and stops at the 180th second. In addition, the UDP session has a constant rate of 120kb/s.

For comparison, first we conducted a simulation in which no loss recovery is applied to any layer of the data. The number of layers at R3 is shown in Fig. 3. The number of layers climbs to 6 first. Then after the UDP session starts at the 150th second, it drops to 5. The number of layers finally recovers to 6 some time later after the UDP session stops. So in this case the number of layers at R3 is relatively stable.

Then we did a simulation in which SRM protects the 2 lowest layers. Fig. 4 gives the result. If we compare this figure with Fig. 3, we can find that the number of layers becomes much more unstable when loss recovery is applied to the lowest 2 layers. Even before the UDP session starts, the number of layers alternates between 6 and 5. This is because the join-experiments of RLM and the control messages of SRM can also cause losses. During the period when the UDP session is alive, the number of layers even drops to 4 and stay there for some time. This shows that the repair traffic generated by SRM for the lowest 2 layers worsens the congestion at the bottleneck and forces RLM to over-respond and drop another layer.

We also conducted a simulation in which repair requests are

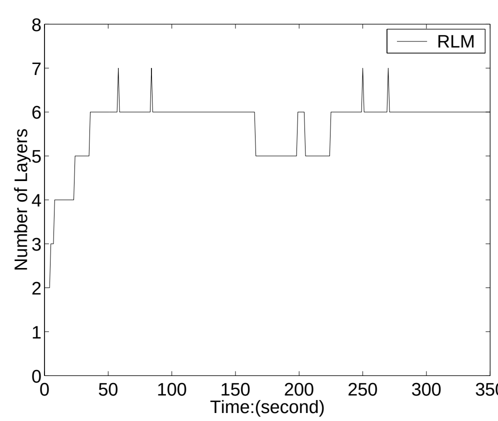

Fig. 3: Number of Layers When No Loss Recovery Applied: Relatively Stable; No Drop to 4 Layers

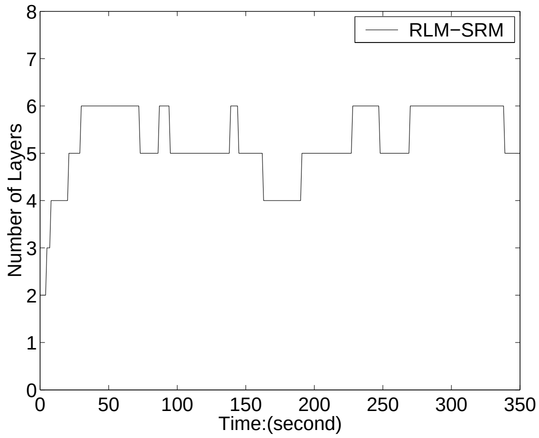

Fig. 4: Number of Layers When SRM Applied to the Lowest 2 Layers: Higher fluctuation; Drop to 4 Layers During Congestion (Over-response)

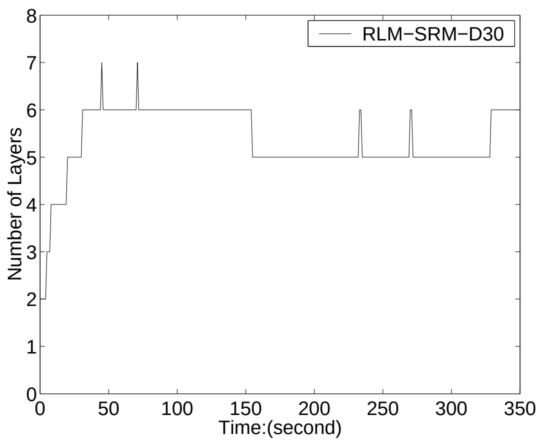

Fig. 5: Number of Layers When Repair Requests Are Delayed: Higher Stability; No Drop to 4 Layers (No Over-response)

held to a point of time when congestion has been eased. The result is shown in Fig. 5. When comparing Fig. 5 with Fig. 4, we can find that the number of layers is much stabler when the repair requests are delayed. Furthermore, the number of layers never drops to 4 in Fig. 5. This means the interference at the bottleneck from repair traffic is lowered and over-response of congestion control disappears.

IV. Summary

In this paper we showed that rate control over repair traffic in multicast is necessary for improving the performance of a multicast transport scheme where distributed loss recovery is adopted, although generally existing multicast congestion control schemes and multicast loss recovery schemes do not specifically control the rate of the repair traffic generated by distributed loss recovery. We showed that if there is no appropriate control over repair traffic rate, network utilization may be lowered and traffic may have higher fluctuation. Higher traffic fluctuation means poorer experience for some application users such as streaming media users. In our paper we also showed that to modify the existing multicast loss recovery schemes to enable them to control the rate of repair traffic is a good direction. One of the specific methods is to control the submission of repair requests.

References

[1] J. Nonnenmacher et al, “How bad is reliable multicast without local recovery?,” *Proc of IEEE INFOCOM*, March 1998.

[2] X. Xu, A. Myers, H. Zhang and R. Yavatkar, “Resilient multicast support for continuous-media application,” *Proc of IEEE NOSSDAV*, New York, May 1997.

[3] S. Floyd, V. Jacobson, S. McCanne, C. G. Liu and L. Zhang, “A reliable multicast framework for light-weight sessions and application level framing,” *Proc of ACM SIGCOMM*, October 1995.

[4] S. Paul, K. Sabnani, J. Lin and S. Bhattacharyya, “Reliable multicast transport protocol (RMTP),” *IEEE J. on Select. Areas Commun.*, April 1997.

[5] R. Yavatkar, J. Griffioen and M. Sudan., “A reliable dissemination protocol for interactive collaborative applications,” *Proc of ACM Multimedia*, 1996.

[6] H. W. Holbrook, S. K. Singhal and D. R. Cheriton, “Log-based receiver-reliable multicast for distributed interactive simulation,” *Proc of ACM SIGCOMM*, August 1995.

[7] S. McCanne, V. Jacobson and M. Vetterli, “Receiver-driven Layered Multicast,” *Proc of ACM SIGCOMM*, Aug. 1996.

[8] L. Vicisano, L. Rizzo and J. Crowcrof, “TCP-like Congestion Control for Layered Multicast Data Transfer,” *Proc of IEEE INFOCOM*, March 1998.

[9] W. Tan and A. Zakhor, “Video Multicast using Layered FEC and Scalable Compression,” *IEEE Trans. on Circuits and Systems for Video Technology*, Februrary 2001.

[10] P.A.Chou and A.E.Mohr and A. Wang and S. Mehrotra, “Error Control for Receiver-Driven Layered Multicast of Audio and Video,” *IEEE Trans. on multimedia*, March 2001.

[11] R. Kermode, “Scoped hybrid automatic repeat request with forward error correction (SHARQFEC),” *Proc of ACM SIGCOMM*, September 1998.

[12] L. Wu, R. Sharma and B. Smith, “ Thin Streams: An Architecture for Multicasting Layered Video,” *Proceedings in NOSSDAV*, May 1997.