Network-Driven Layered Multicast with IPv6

Ho-pong Sze and Soung C. Liew

Department of Information Engineering, The Chinese University of Hong Kong, Shatin,
N.T., Hong Kong
{hpsze8, soung}@ie.cuhk.edu.hk

Abstract. Receiver-driven Layered Multicast (RLM) has previously been
proposed to distribute video over the best-effort heterogeneous IP networks.
Although RLM can avoid network overloading in multicast, its rate control
mechanism performs poorly under bursty traffic conditions, which is the
characteristic of today's Internet. In this paper, we propose a new scheme called
Network-Driven Layered Multicast (NLM) which makes use of IPv6, the next
generation IP. Although IPv6 is not currently used in the Internet and MBone, it
will be undoubtedly adopted in the near future due to the running out of IPv4
address space. With the new priority-dropping feature offered by IPv6, the rate
adaptation control process in our scheme is much simplified and the reception
rate at receivers can be optimized even under bursty background traffic
conditions.

1 Introduction

As the capacity of the Internet increases, various multimedia network applications
that provide audio-visual services such as VOD and video conferencing are becoming
increasingly popular. Due to the heterogeneous nature of the Internet, which
interconnects networks of diverse regions and different technologies together, high-
quality video distribution by multicast is particularly challenging. In a video multicast
system, some users may join in from high-speed FDDI networks while others may
participate by slow dial-up telephone lines. The main issue here is that the sender can
only transmit packets at a single rate. If the video is transmitted in a high bit rate, dial-
up users will not able to participate. But if it is sent out at modem speed, users from
high-speed network will then suffer from low-quality service.

To deal with this rate control problem, some researchers have proposed shifting of
rate adaptation task from the sender to the network or receivers. One of such
approaches is the deployment of video gateways [1, 9] at the network which transcode
a high-quality video multicast stream into a low bit rate one before the stream flows
into a low-speed network so that users can still participate with their limited
bandwidth. However, this approach has two shortcomings. First of all, in order to
perform the stream transcoding, the gateway must be able to understand the semantic
content of the stream. This requires additional efforts on the network. Moreover, the
performance of this approach is highly dependent on the placement of gateways at
appropriate locations. But these locations may not be available because of security or
resource restriction.

An alternative approach is known as Layered Video Multicast [9]. This approach attempts to accommodate network heterogeneity by tuning the reception rate at the receiver end to match the network capacity. The reception rate can be controlled by subscribing to a different number of video layers. As this approach is receiver-driven, each individual user is able to optimize to a different transmission rate.

Apart from the network heterogeneity problem, the Internet is based on the best-effort network-service model. Since there is no guarantee that packets can reach the destination or be delivered on time and in order, the Internet itself is not suitable for real-time traffic, which is subject to strict loss rate and latency constraints. To cite an example, MPEG is a widely used video coding scheme, but it is not designed for transmission in an unreliable channel like the Internet. An MPEG video sequence typically consists of three types of frames: I-frames, P-frames and B-frames. In order to achieve high compression efficiency, motion compensation technique is used to remove temporal redundancy in successive frames when coding the P-frames and B-frames. While I-frames are independently decodable, other types of frames can only be decoded from its reference frames. Therefore, a packet loss in a reference frame during transmission not only introduces an error in the current frame but will also propagate additional errors to the subsequent frames which will refer to it for decoding. As a result, the perceived video quality at the receiver is severely degraded. Several techniques such as forward error correction (FEC) and introduction of more intra-frames in a video sequence are proposed to control the packet loss and error propagation problem, but none of them are efficient as extra bandwidth is required to transmit the redundant packets. Moreover, the reliability of these techniques depends on the degree of redundancy added, which should be determined by estimating channel characteristics. However, the time-variant and dynamic traffic condition in the Internet makes it hard to attain an accurate estimate.

Alternatively, a more feasible solution to overcome packet loss in multimedia streaming is to retransmit the packet when loss is detected. Although retransmission-based error recovery has been criticized as being ineffective in wide-area real-time communications because of the incursion of extra latency in retransmitted packets, effective recovery is still possible if 1) packet loss can detected an soon as possible; and 2) an adequate amount of buffering is introduced at receiver to delay the play-out time of presentation. Recent research also suggested that retransmission-based error control work well in real-time multicast if lost packets can be re-sent immediately upon request [3].

In this paper, we propose a new video multicast approach which can overcome network heterogeneity and packet loss simultaneously. To do so, we adopt IPv6 in the layered video multicast scheme. The new *Traffic Class* field in IPv6 provides an excellent priority-dropping feature to distribute layered video in the IP networks without the effort from sender or individual users for adaptation. Although recent research results suggested that priority dropping of packets without any additional error control measures may not have a significant performance improvement over uniform dropping in layered video distribution under smooth background traffic, it does give a modest gain, especially under bursty traffic conditions like the Internet [10]. By using our loss recovery scheme, error control can be even more effective and a better performance can be achieved. The paper proceeds as follow: Section 2 gives an overview of RLM and IPv6. Section 3 presents our proposed scheme known as Network-Driven Layered Multicast with IPv6 (NLM). Section 4 analyses the performance of the new scheme. Section 5 concludes our work.
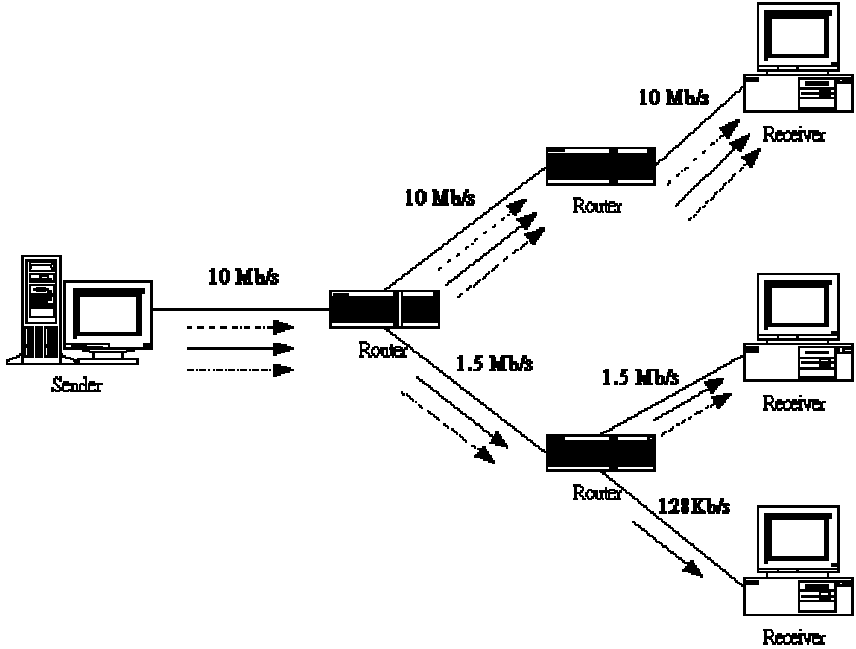
Fig. 1. An Example of RLM

2   Background

2.1   Receiver-Driven Layered Multicast (RLM)

As mentioned in the last section, one approach for real-time video multicast on the IP networks is the use of layered video multicast. One typical example of such schemes is Receiver-driven Layered Multicast [2]. In this scheme, the video source is first encoded into multiple layers. The base layer provides the basic quality of video and each higher enhancement layer progressively improves the signal quality. Depending on the compression schemes, a video sequence can be layered by picture resolution, quantization factor and frame rate. At the sender, each layer is transmitted to a distinct multicast group. In turn, based on its network capacity, each receiver subscribes to the number of layers which can obtain the best possible quality. As the bandwidth between the source and receiver may vary from time to time, dynamic rate adaptation is carried out by the receiver joining/leaving multicast groups during the subscription period. When congestion is detected in the network, a receiver drops the highest enhancement layer to reduce reception rate and hence alleviate congestion. When congestion frees up, it subscribes to an additional layer. Figure 1 shows an example of RLM. Three layers of a video are represented by different types of arrows. By subscribing to different number of layers according to available bandwidth, all receivers in the group are able to optimize their reception rates.

While network congestion can be easily detected by high packet-loss rate at the receiver, the detection of unused network capacity is more difficult. In RLM, a receiver tries to detect extra available bandwidth by periodical probing. That is, in every predefined period, the receiver subscribes to an additional layer to observe the reception quality. If the quality is degraded rather than improved, this indicates network congestion results and hence the receiver should immediately drop the newly added layer. How often should the receiver do the probing is an issue. A recent research work used the decrease of measured RTT as an indication of potential extra bandwidth [12], eliminating the need for blind, periodic probing. Nevertheless, the coordination among a group of receivers in rate adaptation remains an issue.

Another problem of RLM is its poor rate adaptation performance in best-effort networks in which packet losses can occur even when transmission rate is well below the bottleneck rate. As RLM uses packet loss as an indication of congestion, heavy random packet drop will prevent optimal transmission rate from being attained. Moreover, under bursty background traffic, the available bandwidth can be time-varying and fluctuates rapidly. It is difficult for RLM to have a fast enough response for rate adjustment. The simulations in [10] have studied the performance of RLM. The results show that RLM operates far from the optimal level under bursty traffic.

An alternative scheme, Layered Video Multicast with Retransmission (LVMR) was proposed later to improve RLM [5] [6]. With this scheme, retransmission of lost packets is introduced. However, the issue of slow response in adaptation to rapid traffic condition changes remains.

2.2   New Features in IPv6

Internet Engineering Task Force has recently drafted a new version of Internet protocol known as IPv6 [7]. Apart from increasing the address space from 32 bits to 128 bits, several new features have been added in the new protocol to support real-time traffic and multicast. First of all, a range of IPv6 addresses is reserved for multicast purposes. Each IPv6 multicast address consists of a *Flag* field, a *Scope* field and a *Group* identifier. The flag field indicates whether the group is permanent or transient. The scope field limits the scope of the multicast group. The group identifier specifies the multicast group. In addition, a new *Traffic Class* field is added in the IPv6 packet header to provide QoS support. We now discuss the use of this class field.

IP packets arriving at a router are queued for processing. However, when the buffer of a router is full, the incoming packets will be discarded. The new class field allows a source to specify the desired priority/class of its packets. When a link is congested and the buffer overflows, the incoming and queued packets will be dropped from low to high class. Therefore, by assigning important real-time traffic to a higher class, certain degree of QoS guarantee can be attained.

3   Network-Driven Layered Multicast

Both RLM and LVMR attempt to adapt to network congestion and heterogeneity at the receiver side. It is true that receivers can successfully adjust to a more or less

optimal reception rate through these two approaches when the background traffic is smooth, but in practice, the bursty Internet traffic has limited their performance [10]. We propose to solve this problem using a Network-Driven Layered Multicast (NLM) scheme, which shifts the adaptation burden from receivers to the network. In our scheme, receivers are only responsible for handling packet loss recovery.

3.1   Video Layering Scheme

As other schemes mentioned in the previous section, our system makes use of layered video for multicast. It should be noted that our system is applicable to any layered encoding format. Here, we choose MPEG video as an example for illustration. In general, an MPEG video can be encoded into layers by scaling the video SNR, frame rate or a combination of both. For simplicity, we select a layering scheme similar to [5]; that is, we layer the video from the frame types. This simple layering technique requires a minimal additional effort from the encoder. A typical GOF pattern in a MPEG video sequence is $I_1B_2B_3P_4B_5B_6P_7B_8B_9P_{10}B_{11}B_{12}P_{13}B_{14}B_{15}$. For layering, I-frames, which can be decoded independently, is assigned to the base layer. P-frames ($P_4P_7P_{10}P_{13}$), which must be decoded from previous I-frames or P-frames, make up the first enhancement layer. B-frames, which should be decoded from both previous and subsequent I-frames or P-frames, constitute the second enhancement layer. By using this layering scheme, each successive subscribed layer can improve the video quality through an increase in frame rate. Finer layering is also possible by subdividing B and P-frames into more layers.

3.2   Rate Adaptation Scheme

The main difference between our scheme and RLM or LVMR is that our rate adaptation is totally handled by the network instead of receivers. To handle different priorities of IP packets, routers must be capable of processing the class field in the IPv6 header. Although nowadays most routers in the Internet are designed for IPv4, the classical IP, the existing routers will very likely be upgraded to become IPv6-compatible in the near future in order to support real-time traffic and the new address space. We also assume that the routers will be able to support a fair sharing of bandwidth among all the existing flows. That is, each flow will occupy only a certain proportion of the available bandwidth.

In NLM, rather than multicasting each video layer to a different IP address, we transmit all layers of the video source with a single multicast address, but each layer will be assigned with a different traffic class/priority. As each user must receive the base layer to acquire the minimal perceived quality, a higher priority is assigned to the base layer. For enhancement layers, priorities decrease up the layers since each additional layer only further refines the quality. In this way, the sender will transmit the video at a full rate to all receivers regardless of their network capacities.

It might seem at first glance that the multicast traffic under this approach will occupy all the bandwidth in low-speed links and prevent other flows from acquiring adequate bandwidth. However, with advanced routers, the available bandwidth in the links will be partitioned and reserved for different services. The multicast traffic will occupy only an allocated amount of bandwidth when a link is also used by other

services. Since each layer is multicast with a different priority, when the flow rate exceeds the allocated bandwidth portion, packets in the flow will be discarded by the routers starting from those of the lowest priority, which correspond to the highest enhancement layer. As a result, only those layers which the network capacity can sustain can traverse the links and reach receivers. The effect of this process is indeed similar to dropping a layer in RLM when congestion is detected at the receiver. Figure 2 depicts an example of NLM in a heterogeneous network.
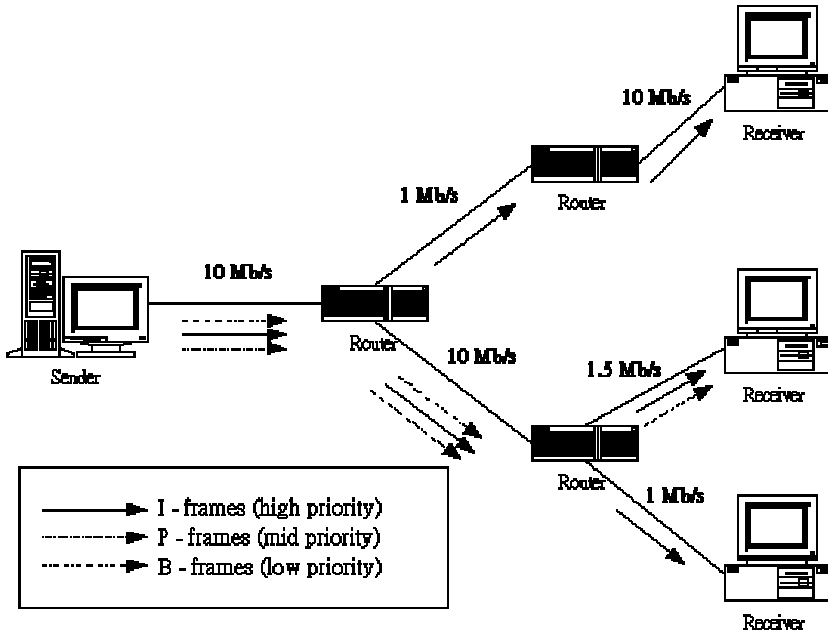

Fig. 2. An example of NLM

In practice, a user may not exactly receive a complete set of layers. For example, the network capacity may allow a user to receive only the base layer and part of an enhancement layer. In that case, we should decide whether to display the partially received layer since display of an incomplete enhancement layer can result in a poorer video quality than just playing the base layer. In NLM, we set a threshold α on the proportion of layer received. If the proportion of layer received r is less than α, the incomplete layer will be omitted in the presentation; otherwise it will be played with other lower layers. We will not suggest a value for α here because its value depends on the particular video coding technique. If the video is vulnerable to data loss, then a threshold, say, as large as 90% - 95% should be set. In case a video can withstand some missing data without severe deterioration in perceived quality, a smaller value such as 70% - 80% may be assigned to α. For the MPEG video in our example, the first enhancement P-frame layer should have a large α since any missing data will be propagated to the subsequent P-frames and significantly degrade the perceived video quality. But the second enhancement B-frame layer can have a smaller threshold

because a B-frame will not be referenced by other frames for decoding. Any data loss in the layer will affect only the corresponding frame. Using a suitable error concealment technique, the impact of packet losses can be minimized.

During a video session, the traffic condition of the network may vary and hence a receiver should continuously monitor the proportion of a layer being received r in order to optimize the perceived video quality. Suppose at the beginning $r < \alpha$ and the layer is not displayed (but is still being received). If now there is spare bandwidth, more packets of the layer are received, so r may then increase beyond α. This will immediately trigger the receiver to start displaying the layer as well. Similarly, when network congestion occurs, r may drop below α. Then the receiver should stop displaying the layer until available bandwidth increases again.

3.3   Retransmission-Based Error Recovery Scheme

Even when the network can support the delivery of different video layers by multicast, due to the best-effort nature of IP networks, packets can still be lost in transmission. Therefore, apart from the rate adaptation scheme discussed in the previous section, we further improve the performance of our multicast system by a retransmission-based error recovery scheme. We choose ARQ rather than FEC for error recovery because ARQ has more benefits than FEC as described in Section 1. Important issues such as retransmission requests, local recovery and priority of retransmitted packets are considered in our scheme.

a. Triggering of Retransmission Request

The function of retransmission is to recover a randomly lost packet. For packet loss due to heavy network congestion, a packet should not be retransmitted as this would only lead to congestion collapse. Thus, in our system, retransmission requests will only be triggered by packet losses in the displayed layers (i.e., layers with $r > \alpha$) but not the undisplayed layers, since the losses in these layers are mainly caused by network overloading.

To detect packet loss, the popular approach is to check for gaps in received packet sequence numbers. Once a gap is discovered in the sequence space, it indicates that the expected in-sequence packet is probably lost. If packet losses are independent and not consecutive, this indeed offers a fast method to detect packet loss. However, it is recently found that bursty losses occur frequently in the Internet [8], which means that with this approach, packet loss may be detected at receiver only when a new packet arrives after several lost packets' inter-arrival times. To improve the response, we use both packet disorder and a time-out mechanism to trigger retransmission requests. Our mechanism works as follow: packets of each layer will have their own set of sequence numbers. When the packets in a layer are received out of sequence, and if the layer is a displayed layer, retransmission requests will be issued immediately. In addition, we also set a time-out instant for each coming packet. MPEG video has a variable encoding rate and it is difficult to predict the time of arrival for each packet, but by layering the video by frame types in our example, each layer can have a more regular bit rate. Moreover, we can use traffic shaping technique to smooth the video traffic so that packets in each layer can be transmitted more regularly. Thus, for each layer, it is possible to estimate the arrival time of packets and set the time-out instant

accordingly. We apply the streaming protocol proposed in [11]. Assume for layer i, the first packet (packet 0) is received at time t_o^i. If the mean inter-arrival time of packets in this layer is R^i, then the expected arrival time for packet j will be

$$e_j^i = T_j^i - \tilde{T}_j^i \ . \tag{1}$$

If the packet actually arrives at time T_j^i, then the error of expected arrival time is given by

$$\tilde{T}_j^i = t_0^i + jR^i; j \geq 0 \ . \tag{2}$$

In order to adapt to delay jitter and clock asynchronization between the sender and receiver for estimation of the packet arrival time, e_j^i is smoothed by an equation similar to the one used in TCP for estimating RTT:

$$e_{j\,smooth}^i = (1-\lambda)e_{j-1\,smooth}^i + \lambda e_j^i \ . \tag{3}$$

where λ is a smoothing constant between 0 and 1. Therefore the estimated arrival time for packet j in layer i is

$$A_j^i = \tilde{T}_j^i + e_{j-1\,smooth}^i \ . \tag{4}$$

In our scheme, the time-out instant for packet j is set to be A_j^i plus a safety margin which is related to the deviation of estimated packet arrival time from the actual arrival time. The deviation for packet j is given by

$$\sigma_j^i = \left| T_j^i - A_j^i \right| \ . \tag{5}$$

We can use a smoothed value of σ_j^i to set the time-out instant. Specifically,

$$\sigma_{j\,smooth}^i = (1-\lambda)\sigma_{j-1\,smooth}^i + \lambda\sigma_j^i \ . \tag{6}$$

Therefore the time-out instant for packet $j+1$ in layer i is

$$TO_{j+1}^i = A_{j+1}^i + \beta\sigma_{j\,smooth}^i \ . \tag{7}$$

for some constant $\beta > 1$. β controls the sensitivity of the time-out triggering. From TCP recommendations, a typical value of 4 can be used. Accordingly, if a packet is dropped in the network and a time-out occurs or packet disorder is detected, a retransmission request will be issued. However, rather than multicasting the request back to sender for retransmission, the local recovery technique described in [4] is deployed in order to reduce the RTT of retransmitted packets.

b. Local Recovery

In NLM, designated receivers (DRs) are used to retransmit lost packets in each subnet. A DR can be considered as a representative for a subnet. Its role is to reduce the load of network/source in retransmissions and the latency of retransmitted packets. When a member in the subnet detect a packet loss by the mechanism in *(a)*, it will first multicast a NACK to its subnet so that the DR and all other receivers can get the request message. The NACK scope can be confined to the subnet only by setting the scope field in IPv6 multicast address. In this way, when other receivers get the NACK, if they also undergo the same packet loss, they need not issue the same request again and so prevent NACK implosion. In response to the request, the DR will then multicast the requested packet to its subnet so that the same loss at different receivers can be recovered simultaneously by a single packet. But if the DR misses the packet also, it will in turn multicast the retransmission request back to original source for recovery, and the source will multicast the missing packet to the whole multicast group.

c. Priority Assignment for Different Packets Types

As for the first transmission of packets, retransmitted packets in different layers from the source should also have different priorities. Nevertheless, as retransmitted packets should reach the receivers as soon as possible before their play-out times, they should have an even higher priority than original packets in the same layer. Hence, we can assign priorities to the packets as follow: First, each layer can have a different priority. Then for each layer, we further subdivide the priorities according to whether the packet is an original or retransmitted one from the source. The resulting priority assignment for different packet types is shown in figure 3. Note that if a layer is only partially received (undisplayed layer), the lost packets will not be recovered by retransmission.
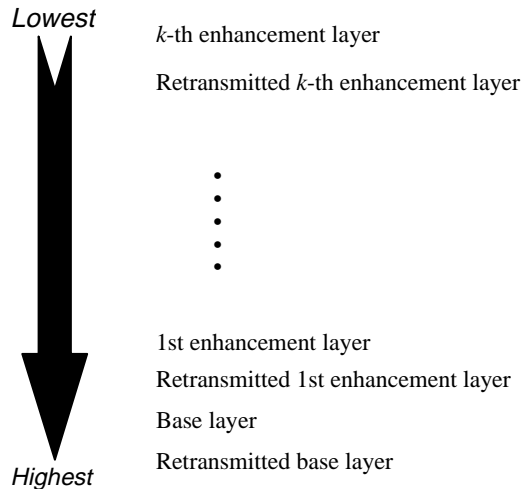
Lowest

k-th enhancement layer

Retransmitted k-th enhancement layer

．
．
．
．
．
．

1st enhancement layer

Retransmitted 1st enhancement layer

Base layer

Retransmitted base layer

Highest

Fig. 3. Priority Assignment for Different Packet Types

4   Performance Study of NLM

NLM is more robust at rate adaptation and error recovery than RLM. We provide our rationale and give a brief performance analysis of NLM here. A detailed study by simulations will be left as future work.

4.1   Rate Control

One obvious benefit of NLM over RLM is that only one IP multicast address is required to transmit all the layers in multicasting. In RLM, each layer requires one address for multicast. Therefore, NLM has effectively reduced the IP address consumption in rate control, especially when the number of layers is large. However, the benefits of NLM are not limited to this. In receiver-driven approaches, rate adaptation requires the interaction between different receivers for coordination of the probing experiments (known as shared learning) [2]. This process involves a lot of overhead and complicated control messages. Although LVMR attempts to suppress the control traffic, it requires the installation of Intermediate Agents which will consume extra resources. A particular bad situation can arise in receiver-driven approaches is when one receiver misbehaves by subscribing more layers than available bandwidth in the subnet, all other users in the same subnet will be then affected and undergo network congestion. In NLM, this does not occur.

  Another improvement of NLM over receiver-driven approaches is the way to detect spare bandwidth. In RLM or LVMR, periodical probing is used to check for any extra bandwidth. This method is inefficient because the reception quality at receivers is affected if no spare bandwidth is found during the probing experiment. Also, it is difficult to estimate the optimal probing period so as to minimize the frequency of failed experiments but can still provide a quick response to spare bandwidth. However, in NLM, when there is any unoccupied bandwidth, low priority packets will immediately be able to flow through the routers and the number of layers received will increase automatically without any effort from the receiver. This greatly reduces the response time in rate adjustment. These are also the reasons why we do not just combine RLM with IPv6 priority dropping feature. Although this combination can surely outperform RLM as base layer packets can have more protection against loss to ensure a basic video quality, the problem of complicated interaction between receivers and inefficient spare bandwidth detection will continue to exist in such an approach. Nevertheless, there is one drawback in our bandwidth detection approach. If the free bandwidth found is not located in the bottleneck link, then any increase of number of layers in a flow will not help in improving the video quality at receiver side as the additional layers will eventually be discarded in the bottleneck link.

4.2   Error Recovery

Apart from rate control, NLM also outperforms the receiver-driven approaches in error recovery. First of all, we use a fast retransmission technique to detect packet loss and send NACK. Suppose the average inter-arrival time for packets in layer i is t^i ms.

If the loss burst length is k, then using error recovery by gap detection only, it takes $(k+1)t^i$ ms to trigger a retransmission request. However, with our time-out mechanism, assume we have an accurate estimation of packet arrival time, then it requires approximately $t^i+\beta\sigma^i$ ms for the triggering, where σ^i is the mean deviation of packet arrival time in layer i and β is a constant. For the time-out mechanism to be effective, we require

$$t^i+\beta\sigma^i < (k+1)t^i . \tag{8}$$

$$\beta\sigma^i < kt^i . \tag{9}$$

Assume β is equal to 4 and the burst length k is 2, we have

$$2\sigma^i < t^i . \tag{10}$$

For a typical Internet video running at 128kbps with packet size of 1000bytes, t^i is larger than 60 ms and increases up the layers. And under normal Internet traffic conditions, the mean deviation of packet arrival time should lie between 10 - 20 ms. Hence, the inequality is valid in general. As a loss burst length of 2 packets is common in today's Internet traffic, this retransmission scheme works well together with the gap detection scheme. Moreover, the deployment of local recovery technique in our system further reduces the latency of retransmitted packets, which means that lost packets can have a higher chance of recovery. Even if the losses cannot be recovered by local DRs, the high priority of retransmitted packets from the source still offers a reliable retransmission channel for recovery.

5   Conclusions

We have proposed a scheme for Network-Driven Layered Multicast of video over the Internet. Our system made use of traffic class in IPv6 to overcome network heterogeneity and packet losses simultaneously. To perform rate adaptation, each video layer is assigned with a different priority. The priorities are assigned in such a way that only base video signal can flow through the network in times of congestion. We also suggested a retransmission-based error recovery scheme to deal with packet losses. Our scheme offers a faster packet loss detection technique by an integrated approach and a more effective retransmission method by the local recovery technique. A novelty with our error-recovery scheme is that we set a threshold on the proportion of data received in a layer to decide whether to perform loss recovery in that layer and display it. This cuts down unnecessary retransmissions of data of layers that will not be displayed and prevent exacerbating network congestion further.

References

1.   E. Amire, S. McCanne, and H. Zhang, "An Application-level Video Gateway," in *Proceedings of ACM Multimedia '95* (San Francisco, CA, Nov. 1995), ACM, pp. 255-265

2. S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven Layered Multicast," in *Proceedings of SIGCOMM '96* (Stanford, CA, Aug. 1996), ACM, pp. 117-130
3. S. Pejhan, M. Schwartz, and D. Anastassiou, "Error Control Using Retransmission Schemes in Multicast Transport Protocols for Real-Time Media," in *IEEE/ACM Transactions on Networking*, Vol.4, No.3, Pages 413-427, June 1996.
4. S. Floyd, V. Jacobson, C.-G. Liu, S. McCanne, and L. Zhang, "A Reliable Multicast Framework for Light-Weight Sessions and Application Level Framing," in *Proceedings of SIGCOMM '95* (Boston, MA, Sept. 1995), ACM, pp.342-256.
5. X. Li, S. Paul, P. Pancha, and M.H. Ammar, "Layered Video Multicast with Retransmission (LVMR): Evaluation of Error Recovery Schemes," in *Proceedings of NOSSDAV '97*.
6. X. Li., S. Paul, and M.H. Ammar, "Layered Video Multicast with Retransmission (LVMR): Evaluation of Hierarchical Rate Control," in *Proceedings of IEEE INFOCOM '98*.
7. S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," *RFC2460,* Dec. 1998.
8. V. Paxson, "End-to-End Internet Packet Dynamics," in *Proceedings of SIGCOMM '97*.
9. T. Turletti and J.-C. Bolot, "Issues with Multicast Video Distribution in Heterogeneous Packet Networks," in *Proceedings of Sixth International Workshop on Packet video* (Portland, OR, Sept. 1994).
10. S. Bajaj, L. Breslau and S. Shenker, "Uniform versus Priority Dropping for Layered Video," in *Proceedings of SIGCOMM '98*.
11. P. Wu and S.C. Liew, "A Streaming-Protocol Retransmission Scheme without Client-Server Clock Synchronization," in *IEEE Communications Letters*, Vol.3, No.7, July 1999.
12. C.W. Fung and S.C. Liew, "End-to-End Frame-Rate Adaptive Streaming of Video Data," in *Proceedings of ICMCS '99*.