RECEIVER-DRIVEN LAYERED MULTICAST USING ACTIVE NETWORKS

Lechang Cheng and Mabo R. Ito
Department of Electrical & Computer Engineering
University of British Columbia
Vancouver, Canada

ABSTRACT
This paper proposed a receiver-driven layered multicast scheme using active networks (RLM-AN). The multicast tree is regarded as a set of virtual links. TCP-friendly congestion control and FEC-based error control are performed on each virtual link. In order to solve the loss path multiplicity problem, the intermediate active nodes also perform error recovery. By introducing congestion control and error mechanisms in active nodes inside the network, we obtain smoother and more TCP-friendly throughputs and quicker response to congestion inside the network and much lower packet loss rate.

1 INTRODUCTION
Multicast has become a promising technology for saving resource in delivering multimedia over Internet. In order to address the problem of bandwidth heterogeneity of receivers, a new cumulative layered multicast scheme has been proposed in [4]. In this method, the video is encoded into a basic layer and one or more enhancement layers. The lowest layer contains the basic information of the video and high layers provide enhancement to video quality. Receivers subscribe to cumulative layers according to their available bandwidth.

For multicast transport protocols, there are two important issues that need to be addressed: congestion control and error control. Multicast transport protocols can be classified into two categories: pure end-to-end schemes and router-assisted scheme. In a pure end-to-end layered multicast scheme, only the sender and the receiver are responsible for collecting network information and perform congestion control or error control. Although the existing pure end-to-end layered multicast schemes have presented some good results, they have many problems such as a) the need to allocate and manage several multicast groups, b) slow adaptations, c) subscribe unsubscribe oscillation, d) difficulty to synchronize among receivers [9].

Recently, video communication using active networks has drawn more and more attentions [9]. Active networks add intelligence inside the network. With the computation ability presented in intermediate nodes inside the network, network problems can be treated within the network rather than at the network endpoint. Therefore, one can anticipate that multicast schemes using active networks can provide better solution to multicast routing, quality of service support, congestion control, and error control.

In this paper, a receiver-driven layered multicast scheme using active networks (RLM-AN) scheme for real-time video streaming is proposed. Active nodes inside the network not only rely the streaming packets, but also perform congestion control and error control. While most of the current congestion control algorithms are end-to-end, we design congestion control mechanism from another perspective. The overall multicast tree is regarded as a set of virtual links and congestion algorithm is applied to each of the virtual links. In order to improve reliability of the data transmission scheme, a distributed FEC (forward error correction)-based error recovery mechanism is also provided to protect video quality from packet loss.

2 STRUCTURE OF RLM-AN SYSTEM
A RLM-AN system consists of three parts: a video streaming server, a set of clients and a number of intermediate nodes (active or non-active). The active node will recognize active packets, process them then replicate them and deliver them to their child nodes. Non-active nodes simply pass the active packets on as regular IP packets. The IP tunnel from a father node to its child node is regarded as a virtual link.

2.1 Functionality of the server
The server compresses the video streaming into a number of layers, marks the generated data packet as active packets and sends them to their children using IP service. During the multicast session, the server receives subscribe packets from active nodes or clients. The server registers them as its child nodes and keeps records of their subscription level and error protection level. During the video streaming session, the server also receives feedback packets from its child nodes and adjusts the number of layers and error protection level for each child accordingly.
In this work the notion of a server is extended to include both the original source servers as well as the pseudo servers associated with active nodes as described in Section 2.3.

2.2 Functionality of the client
The client receives video packets from its father node in the multicast tree. The client also estimates the round trip time (RTT) and packet loss rate of the virtual link from its father to itself. For every period of time, it estimates a fair rate using TCP response function, eqn. (1) below, calculates the subscription level and error protection level and sends a feedback packet to its father node.

2.3 Functionality of the active node
Each of the active nodes can be regarded as a combination of a client and a pseudo server. On one hand, it receives packets from its father node and sends feedback information to its father node. On the other hand, it receives feedback information from its child nodes and adjusts the number of layers and error protection level for each of the children accordingly. In addition, if packet loss occurs, the active node will perform loss recovery. If the error protection level of its child node is higher than that of its upstream link, the active node will generate extra redundant FEC packets based on the received packets.

3 PROTOCOL DETAILS

3.1 Distributed TCP-friendly congestion control
In the Internet, majority of the Internet traffic is TCP flow. In order to achieve fairness, it is important to design the congestion mechanism to be TCP-friendly.

Many of the existing multicast congestion control mechanisms are conducted in a receiver-driven way such that the client estimates an appropriate transmission rate and feedbacks this information to the server which will adapt its transmission rate accordingly. However, one problem of receiver-driven congestion control schemes is feedback explosion. When the number of clients increases, the feedback packets may overwhelm the server and cause congestion of the links around the server. Besides, the network condition varies greatly from area to area. The occurrence of congestion in one area does not mean the congestion in another area. Therefore, response to network condition changes should also be directed to the congested region.

Based on the above discussion, a distributed TCP-friendly congestion control scheme for RLM-AN is proposed. Congestion control is performed between the sender and the receiver of each of the virtual links. The receiver (a client or an active node) estimates the packet loss rate and round trip time from its father node and computes a fair appropriate transmission rate using TCP response function, eqn. (1) below. The receiver then finds the appropriate number of layers and sends the information to its father node. The father node adjusts its sending rate accordingly.

3.1.1 Rate Adjustment
For TCP flows, the relationship between the packet loss rate $p$, the round trip time $t_{RTT}$, packet size $s$ and the throughput $R_{fair}$ is [6]

$$R_{fair} = \frac{s}{t_{RTT}(\sqrt{2p/3} + (12\sqrt{3p/8})p(1 + 32p^2))} \quad (1)$$

As in TCP, there are two algorithms used: slow-start and congestion-avoidance. Before any packet loss happens, the slow start algorithm is used. For every round trip time, the receivers double the sending rate. After packet loss happens, the congestion avoidance algorithm is used instead. The fair rate is estimated through equation (1). The new transfer rate is calculated as:

$$R_{i+1} = \begin{cases} \min(R_i + s/t_{RTT}, R_{fair}), & \text{when } R_i \leq R_{fair} \\ \max(R_i - s/t_{RTT}, R_{fair}), & \text{when } R_i > R_{fair} \end{cases} \quad (2)$$

where $R_{i+1}$ is the new transmission rate and $R_i$ is the previous transmission rate. With the transmission rate $R_{i+1}$, the number of layers can be calculated as:

$$N = \sum_{j=1}^{N} B_k < R_{i+1} \leq \sum_{k=0}^{B_k} B_k \quad (3)$$

3.1.2 Estimation of packet loss rate and RTT
The packet loss rate is measured at the receiver (a client or an active node). In order to obtain a less noisy measurement of the packet loss rate, the Average Loss Interval method [6] is used. A lost event is defined as one or more packet losses during a round trip time. The average loss interval can be computed as the weighted average of several recent loss intervals. The packet loss rate is calculated as the inverse of the average loss interval size [7].

The round trip time is also measured at the receiver. For every period of time, the receiver sends the feedback packet to its father node. The father node sends an echo packet to the receiver. Using the echo packet, the receiver can measure the RTT periodically. In order to obtain relatively smooth RTT estimation, exponential smoothing is used to reduce the noise in the RTT measurement.

3.2 Distributed FEC-based Error Control Algorithm
Error control mechanisms are usually provided to recover lost packets and improve reliability. FEC-based methods are usually used in video multicast schemes to avoid the problem of feedback implosion. Therefore, FEC-based error control is used in RLM-AN.

One challenge of designing an error control mechanism for video multicast is the problem of multi-path loss. A packet loss that occurs in the multicast tree will cause packet loss in many clients. If the lost packet could be recovered by the very first active node that
detects this loss, this packet loss will no longer propagate to all downstream clients or active nodes. The packet loss recovery operation is performed once instead of multiple times at all clients without increasing the error recovery latency. Furthermore, by conducting error recovery for each virtual link, the packet loss rate for the virtual link will be decreased. Thus, the overall packet loss rate from the server to the client will be greatly reduced. As the link capacity also depends on the packet loss rate, the effective bandwidth from the server to the client can also be improved. Therefore, we proposed a distributed error control mechanism for RLM-AN. In this error control mechanism, FEC-based error control is conducted on each of the virtual links in the multicast tree. Details of this mechanism are as follows.

- For each virtual link, the receiver (an active node or a client) estimates the link loss rate. Based on the target loss rate, it calculates the packet loss protection level. The receiver will send the information to its upstream node;
- The sender receives both source packets and redundant packets from its upstream node. If packet loss occurs, loss recovery is performed. If the error protection level of its child node is higher than that of its upstream link, the active node will generate extra redundant packets based on the received redundant packets.

### 3.2.1 Calculation of the Error Protection Level

In FEC, the data stream is divided into blocks of \( N \) packets and \( K \) redundant packets are added to each block. The more redundant packets added to each block, the more source packets that can be recovered resulting in a lower unrecoverable packet loss rate. However, the redundant packet consumes the available bandwidth. In this section, an algorithm is provided to find the minimum number of redundant packets that can reduce the unrecoverable packet loss to the predefined level.

Let \( I \) be the number of received packets in one block and \( M \) is the number of packets that can be recovered. When \( I \) is larger than \( N \), all the \( N \) original packets can be recovered and \( E(M | I = i, i \geq N) = N \). When \( I \) is less than \( N \), only the source packets are useful and the redundant packets will be discarded. Therefore

\[
E(M | I = i, i < N) = \begin{cases} 
\sum_{i=K}^{\infty} C_{K}^{i} (1-p)^{i} p^{K-i} \cdot j & \text{when } i > K \\
\sum_{i=0}^{K} C_{K}^{i} (1-p)^{i} p^{K-i} \cdot j & \text{when } i \leq K 
\end{cases}
\]

Therefore, the expectation of the number of source packets that can be recovered is

\[
E(M) = \sum_{i=0}^{N+K} E(M | I = i) \cdot P(I = i) 
\]

where

\[
P(I = i) = C_{K}^{i} (1-p)^{i} p^{K-N-i} \]

The packet loss rate \( \bar{p} \) after packet recovery is

\[
\bar{p} = \frac{E(M)}{N} 
\]

\( \bar{p} \) is the function of \( N \), \( K \) and \( p \). Given \( N \) and \( p \), simulation can be done to find the minimum \( K \) that can reduce the packet loss to a certain level \( p_0 \).

\[
\min_{K} \bar{p}(N, K) < p_0
\]

### 3.2.2 Priority Dropping Mechanism

Another challenge in designing the error control algorithm for layered multicast is the dependency among packets from different layers. The decoding of the higher layer packets relies on the packets of the low layers in the same frame. If the packets in the lowest layer are lost, it is meaningless to recover packets in the high layers. In order to add extra protection to source packets in lower layers, a priority dropping mechanism is used in RLM-AN. The priority dropping mechanism is performed at the intermediate active node. When the active node detects congestion in the queues of its output interface, it will drop packets of high layers before packets of lower layers.

### 4 SIMULATION RESULTS

RLM-AN has been implemented in ns-2. Extensive simulations have been done to assess the performance of the congestion control and error control mechanism of RLM-AM. In [1], simulation results show that the distributed congestion control scheme of RLM-AN is scalable and more TCP-friendly than RLM. In this part, some of the simulation results related to the error control mechanism of RLM-AN are discussed.

![Figure 1 Network tops used to test the scalability of the error control algorithm](image)

The main purpose of this simulation study is to test the scalability of the error control scheme of RLM-AN. Two different tops (figure 2) are used. In top (a), the number of clients is varied and in top (b) the number of intermediate active nodes is varied. If the proposed scheme works well with these two topologies, one can conclude that it will work well with different numbers of intermediate active nodes and clients. During the simulation, the block size is set to be 8. The targeted loss rate is 0.1%.

In top (a) and (b), the links are exactly the same (2Mb, 20ms, 0.5% packet loss rate). Therefore, the packet loss rate and end-to-end delay for every client are almost the same as in the simulations for top (a). The first node (from the top) in top (a) will be used as the representative in performance comparison.
• **Packet loss rate**

Figure 2-upper shows that when the number of clients increases from 1 to 40, the packet loss rate with or without FEC does not vary with the number of clients. Figure 2-lower shows that without FEC, the packet loss rate increases when the number of intermediate active nodes increases. When the error control mechanism is applied, the packet loss rate drops to below 0.2% and does not vary when the number of active nodes changes. Therefore, the proposed algorithm is scalable in terms of packet loss recovery when the number of intermediate active nodes or the number of clients increases.

![Figure 2 Packet loss rate](image)

5 **Summary**

In this paper, a RLM-AN scheme is provided for video streaming multicast. In addition to the basic data dissemination mechanism, a distributed TCP-friendly congestion control algorithm and a FEC-based error control algorithm are also proposed. Simulation studies show that the error control mechanism greatly reduces the packet loss rate without increasing the end-to-end delay significantly. It is also scalable when the number of clients or intermediate active nodes increases.

**REFERENCES**


