MSc. in Software Technology

Dissertation

of

Sender-Adaptive Rate Control for Layered Video Multicast

Academic Supervisor : Dr. Rocky Chang
Student : Lau Chi Wai
Date : May 2001

Abstract

This dissertation proposes the Sender-Adaptive Rate Control for Layered Video Multicast[SARC] to provide a rate control mechanism for the layered video multicast over the internet. Unlike other established approach, SARC will determine at the start of the video session the number of video layers according to the real-time status of the network bandwidth and make adaptation to the video layers so that the convergence time would be minimal.

SARC adopts the hierarchical structure from LVMR[10] to organize the receivers into domains. Intermediate Agent/IA administers the domain and gathers network status information from different subnets within the domain. Subnet Agents/SA are responsible for their own subnets. In SARC the sender gets the real-time network status information from Intermediate Agent. It consolidates various link speeds of the receivers. It then dynamically splits or combines video layers as well as the transmission rate of each layer so that the convergence time would be minimal.

The SARC is evaluated by simulation using network simulator ns. The result is compared with established approach such as RLM[7]. In RLM[7] the sender is not involved in the rate control mechanism, only receivers are involved. SARC is shown to have faster convergence time and lower loss ratio than RLM[7].

Acknowledgement

I would like to thank my supervisor Dr. Rocky K. C. Chang for his valuable advice and guidance during the preparation of the dissertation.

Table of Content

List of Figures

List of Tables

1.0 Introduction

In recent years the Internet has seen significant growth in bandwidth, for example, many internet services providers can provide link speed up to T3. Moreover the computer processing power also advances in fast pace. Nowadays 100-MHz system bus supporting up to 800-MBps data throughput is a norm. Both these two important criteria have helped foster transmission of real-time video data over the internet. The M-Bone also serves as a test bed for the development of multicasting applications such as video conferencing, distance learning, remote presentation, and media-on-demand. This dissertation will focus on the topic of rate control mechanism of layered video multicasting over the internet.

1.1 Background

The transmission of video data is usually differentially compressed because of the redundancy contained in most video sequences. Otherwise the raw uncompressed video bandwidth could be about 60Mbps for NTSC video[8]. Thus the transmission rate varies from very low for still scenes to very high for sequences with many scene changes.

In order to support multicasting of real-time video data over the internet, the following supports are required :

i. Multicast data delivery support

- This is satisfied by multicast routing protocols such as DVMRP, MOSPF, PIM, CBT, …etc.

ii. Real-time requirement of digital video

- The playout time for each frame should be preserved otherwise the synchronization will be lost. Thus the end-to-end network delay jitter needs to be small. This problem can be alleviated by buffering at the receiver side.
- Network congestion can lead to a significant loss of packets. The result could be picture degradation at the receiver side. By decreasing the digital video transmission rate at the sender, the loss could then be minimized.

Some researchers[3], [5] handle the multicast of video in the internet by a resource reservation model of the network. The sender should ask for permission to send and the network would reserve a certain amount of resources. Turletti and Huitema in [8] section V, p.345, point out that this model is based on the assumption of video data flow characteristics known before the start of video session. Moreover the network has enough knowledge to decide if such a flow should be admitted or not. The advantages and disadvantages are :

Advantages

- Guaranteed quality of service

Disadvantages

- Need to couple admission with accounting
- Need to enforce complex reservation schemes
- Need to introduce virtual circuit in the internet which is datagram oriented
- The Resource Reservation Protocol has not been fully deployed

The burst nature of the variable-bit-rate video stream makes it difficult to determine the exact amount of resources required. If resources are reserved according to the average rates of the variable-bit-rate video sources, unacceptable delays or packet losses may result when the sources are transmitting at their peak rates.

Due to the above limitations, other researchers[2], [4], [6], [7], [10] worked on the end-to-end control model. This end-to-end control model explores the feasibility of video distribution over the internet with the best effort delivery support. End-to-end control has the advantage of relieving the impact of network infrastructure changes. Basically it makes use of a rate control mechanism to adjust the video traffic characteristics (e.g. by adjusting the compression ratio and hence the video transmission rate) to meet the current network's capabilities based on feedback information about changing network conditions.

1.1.1 Rate Adaptation Control Mechanism

This rate control mechanism falls into certain categories with examples.

<u>A. Sender-Initiated Approach</u>

Adaptive Congestion Control (ACC) [2] and,
Scalable Feedback Control (SFC) [4],

<u>B. Receiver-Initiated Approach</u>

Receiver-Driven Layered Multicast (RLM) [7].

<u>C. Sender/Receiver-Initiated Approach (Hybrid of A and B approaches)</u>

Destination Set Grouping (DSG) [6]

<u>D. Hierarchical Approach with Agents/Receiver-Based Control</u>

Layered Video Multicast with Retransmission (LVMR) [10]

These approaches are respectively described in section 1.1.1.1.

1.1.1.1 Sender-Initiated Approach

The sender multicasts a single video stream whose quality or transmission rate is adjusted based on feedback information from receivers. In other words, it is the sender who decides if the maximum output rate of the video coder should be changed or not based on a probabilistic feedback from receivers. Both in ACC[2] and SFC[4] the sender transmits at a single rate for the entire group of receivers and the goal is to optimize that single rate so that the overall reception quality of the group is good. In SFC[4], the sender deliberately avoids feedback implosion from receivers by using a randomly delayed reply scheme together with a probabilistic polling mechanism.

1.1.1.2 Receiver-Initiated Approach

The sender multicasts several layers of video (typically a base layer and several enhancement layers) in different multicast groups. The receiver decides on its own whether to drop an enhancement layer or to add one. In RLM[7], a fully distributed approach is advocated in which a receiver makes decision to add or drop an enhancement layer. This decision is enhanced by a “shared learning” process in which information from experiments conducted by other receivers is used to improve performance. RLM multicasts the join-experiment from the beginning till the end of the experiment to every member of the group. Thus every other member can learn the result from one single member’s experiment result. Hence the fraction of time the network is congested due to join-experiments decreases. In this learning process, receivers make decisions based on failed experiments not on successful ones. The success/failure decision is based on local observations, not on a global outcome.

1.1.1.3 Sender/Receiver-Initiated Approach

In DSG[6] the sender sends three replicated streams with different compression ratios targeting at different groups with different bandwidth constraints. Within each stream the receivers provide feedback information to the sender to adjust the video transmission rate. Whereas the receivers also can change to other stream to receive better or worse quality video according to its capabilities and link speed. Therefore within each stream the control is in the sender, it is sender-initiated while across stream the receivers can make the decision to change to high or low quality stream and is invited from the sender to change by means of poll messages.

1.1.1.4 Hierarchical Approach with Agents/Receiver-Based Control

On the other hand in LVMR[10] a hierarchical approach is used in the receivers' dynamic rate control schemes so as to allow receivers to maintain minimal state information and decrease control traffic on the multicast session. In addition LVMR compiles the receivers' success and failure experiment results into a comprehensive group knowledge base. This knowledge base is partitioned intelligently and distributed efficiently to the receivers, and some agents with relevant information. Thus based upon minimal state information at the agents in the network, intelligent decisions can be made about arbitration of concurrent add-layer experiments.

Table 1-1 delineates each of these five schemes.

Table 1-1 Schemes of Video Multicast on the Internet

Video Multicast on the Internet					
Titles	An Adaptive Congestion Control Scheme For Real-Time Packet Video Transport 1993 SIGCOMM [2]	Scalable Feedback Control For Multicast Video Distribution in The Internet 1994 ACM [4]	Receiver-Driven Layered Multicast 1996 SIGCOMM [7]	On the Use of Destination Set Grouping to Improve Fairness in Multicast Video Distribution INFOCOM'96 [6]	Layered Video Multicast with Retransmissions (LVMR):Evaluation of Hierarchical Rate Control INFOCOM'98 [10]
Transmission Source	Single Video Source Sender-based Control	Single Video Source Sender-based Control	Single Video Source Receiver-based Control	Single Video Source/Multiple Streams Sender/Receiver-based Control	Multiple Video Sources Hierarchical approach. Agent/Receiver-based Control
Video quality	352 X 240 @ 30 fps	1. Max Rate = 150kbps 2. Min Rate = 10kbps	Six-layered CBR stream at 32 X 2^m kbps, m = 0..5	G1 : [8..15] 140 kbps G2 : [16..23] 76 kbps G3 : [24..31] 46.5kbps	Packets of 1K bytes at a speed of 24 frames per second
Control Target	Single Video Source Data Generation Rate	Single Optimized Transmission Rate For The Group	Optimal number of video layers subscribed by each receiver	Single Optimized Transmission Rate For Each Video Stream	Optimal number of video layers subscribed by each receiver
Control Parameters	1. Target sending rate 2. Target queue size	1. Rate Increment = 10kbps 2. Threshold of Congestion = 1.4% Coder Refresh Rate 3. Quantization step 4. Motion Detection Threshold	1. Congestion detection timer 2. Join timer 3. Join timer backoff constant 4. Join timer relaxation constant	1.Congestion threshold 2. Unload condition threshold 3. Stream advance decision threshold 4. Advance Invitation	1. Congestion detection timer 2. Unload condition detection timer
Judgment Basis	1. Per-Frame SNR. 2. Average SNR over the entire sequences of frame	Packet Loss Rate	1. Worst-case lost rate over varying time scales. 2. Time taken for system to converge	Receiver Packet Lost Rate.	1. Averaged data rate over every half-second. 2. Packet loss ratio during one GOP
Advantages	Transmission rate adaptation for VBR video in end-to-end control (not for multicast)	Avoidance of feedback implosion in multicast environment	Enhanced system scalability	Improved fairness over single stream variable rate approach	Hierarchical control of layered video subscriptions
Drawbacks	1. Ignorance of network heterogeneity. 2. Broadcast at fixed rate regardless of changing network condition	1. Ignorance of network heterogeneity. 2. Broadcast at fixed rate regardless of changing network condition	1. Increased multicast control traffic especially on low speed link 2. Maintenance of surplus state information by receivers	1. Additional bandwidth overhead incurred in low speed link. 2. Network utilization and video quality not optimized.	1. Rule becomes invalid when network bandwidth changes 2. Latency involved in decision making to allow adding layer

1.1.2 Fairness and Heterogeneity

Internet is a heterogeneous environment where the capabilities of the receivers differ greatly. Moreover the bandwidth of the links connecting them to their senders vary from one to another. Thus when a source multicasts a broadband video signal, not all intended destinations are willing to receive or are capable of receiving the complete signal. As stated by Shacham in [1] section 1, p.2107:

Bandwidth or terminal limitations restrict the rate of information that can be delivered to some, whereas others prefer to pay less and receive only a subset of the information contained in a multicast signal.

1.1.2.1 Heterogeneity and Fairness in Sender-Initiated Approach

In the sender-initiated approach the sender determines the transmission rate according to the conditions in the network capacity feedbacked by the receivers. However in a multicast environment, the network capacity is difficult to define or sometimes it is ill-defined. The heterogeneity of the group of receivers cannot be satisfied by a single video transmission rate because their bandwidth requirements are in conflict. As the number of receivers grows, the sender-initiated approach is also not scalable. Similarly in the sender/receiver-initiated approach the DSG[6] scheme also provides rate adaptation of different streams of the same video targeting different bandwidth constraint. However multiple streams all carrying the same video with different compression ratio may result in inefficient use of network bandwidth.

If a receiver is constrained by a low-speed link, then the video quality received should be low compared with those receivers connected to high-speed links. Then when fairness is addressed among receivers with different capabilities, their processing power and link speed connection should be taken into account. In other words the quality of video received should be commensurate with link speed of the network path together with the receiver's processing power. A single video transmission rate in the sender-initiated approach cannot achieve fairness among receivers. If the video can be separated into basic layer and several enhancement layers, the sender can send each layers in different multicast group address and the receiver can subscribe to the basic layer and a set of enhancement layers depending on its own capability and link speed connection. This scheme could achieve a certain degree of fairness among receivers and is the principle of RLM[7] and LVMR[10].

Comparing with the sender-initiated single/multiple stream approach, the codec requirement on the receiver side would be a bit more complex because different video layers have to be synchronized before decoding. This entails more receiver buffer size and also overhead in encoding and decoding time. However with the advance in VLSI technology, this issue could be well handled by high-speed hardware components and would not be an obstacle for layered video transmission. As far as bandwidth is concerned, layered video transmission is more bandwidth efficient in general, when compared with sender-initiated multiple stream approach such as DSG[6].

1.1.2.2 Heterogeneity and Fairness in Receiver-Initiated Approach

In RLM[7], a fully distributed approach is adopted among receivers by sharing the experiments from the beginning phase to the result phase. However in a heterogeneous network, the experiment done by one receiver may not affect the state of other receivers. These receivers just do not have to know about the results of the experiment. Thus as stated in LVMR[10], section 1, p.1063, each receiver has to maintain a variety of state information which it may or may not require and is thus redundant. As mentioned in RLM[7] the experiment results are also multicasted to others. Hence the bandwidth on low-speed link may decrease and video quality received may decrease as well.

The approach in LVMR[10] is hierarchical in the sense that the region of receivers can be grouped in domain. In each domain there is an Intermediate-Agent (IA) for coordinating add-drop layer experiments from receivers. The next hierarchy down from the domain is subnets. Within each domain, there are many subnets with Subnet-Agent (SA) as coordinator in each subnet. The IA contains rule to determine the possibilities of allowing simultaneous add-drop experiments within or across the same or different subnets, in the same or different video layers.

1.1.3 The Hierarchical Approach of LVMR[10]

In the hierarchical approach the receivers do not give feedback directly to the sender but rather to subnet agents SA in their corresponding subnets. The subnet agents form a hierarchy as shown in figure 1.1 below in a typical domain where there are two subnets. Up next in the hierarchy is the intermediate agent IA. The SA collects the status of its subnet and reports to its IA. The IA compiles the information from all SAs in its domain and passes it down to the SAs. Then each SA multicasts the information to its corresponding subnet. Communication within subnet is through multicast whereas unicast is used between SA and IA. Figure 1.2 shows the hierarchy of LVMR[10] when more domains are involved. Note that the sender can also be the IA at the highest level. In order to simply the figures the receivers inside the subnets are not shown.

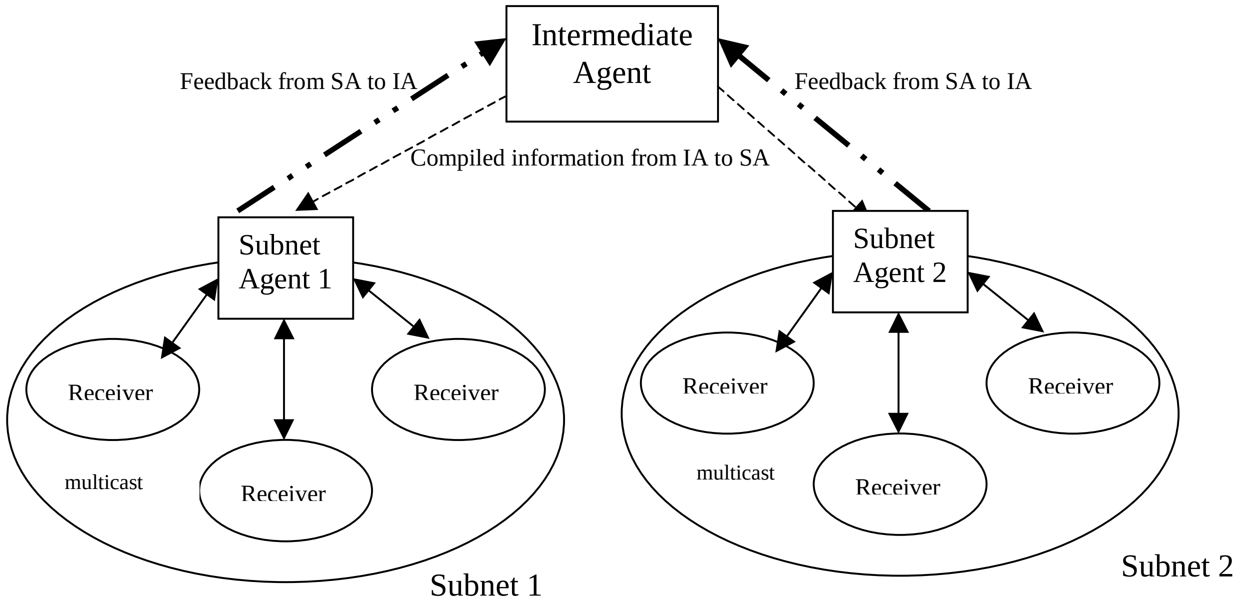


Figure 1-1 IA-SA hierarchy in a domain with 2 subnets

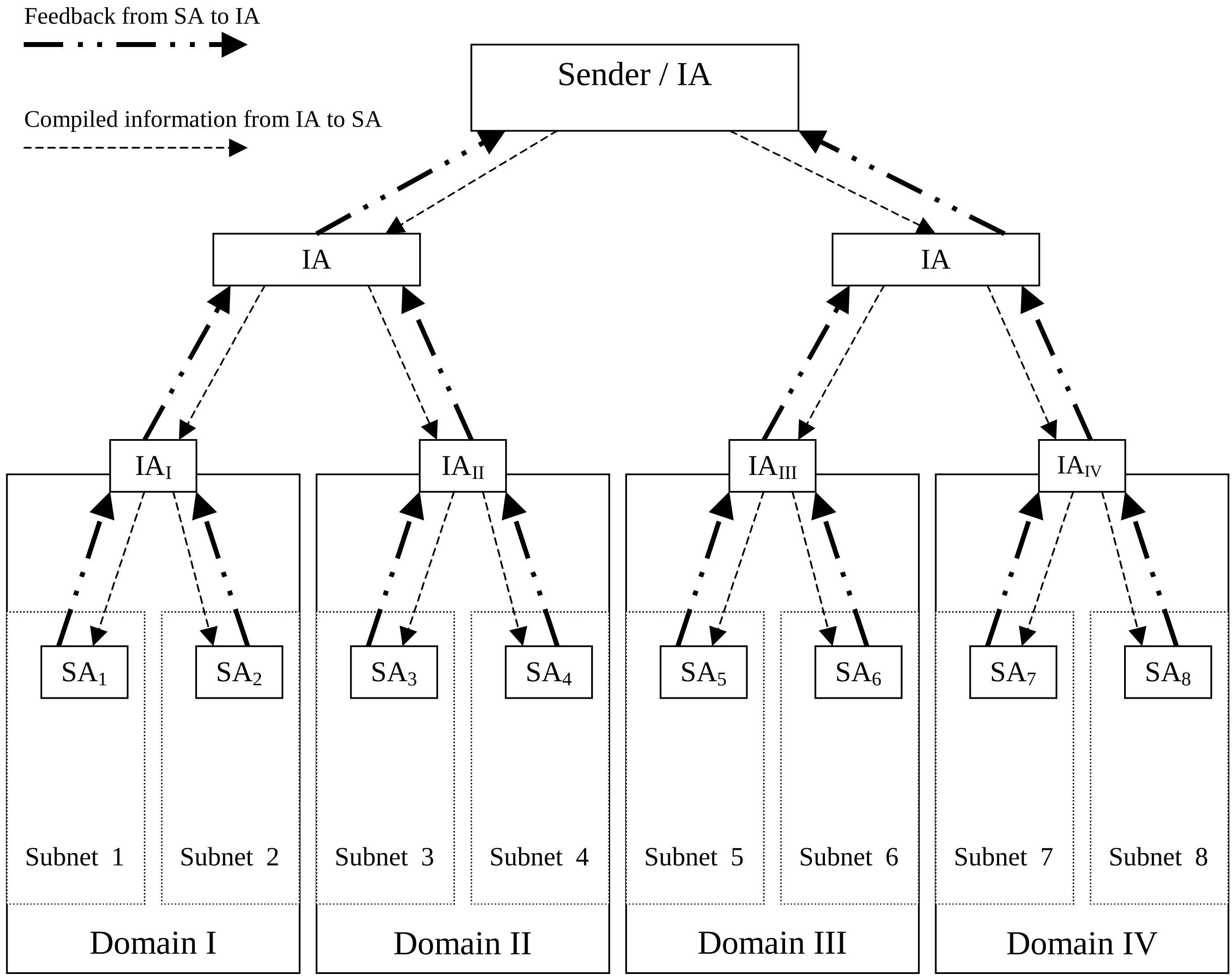


Figure 1-2 Hierarchical control architecture of LVMR[10]

The knowledge base in the IA of the domain will compile rules based on past experiment. For example if receiver A in subnet 1 wants to add layer 4 and its present layer is 3. Then it multicasts its request to subnet 1. SA_1 then forwards this request to IA_I which then infers from its knowledge base whether this operation is permitted or not. If past experiment shows that adding layer from 3 to 4 causes congestion in both subnet 1 and subnet 2, then IA_I will issue a FAIL message to SA_1which then multicasts this FAIL message to subnet 1. Upon receipt of this message, the receiver A will update its T_U^i timer to delay its layer adding operation. If there is no conflict inferred from the knowledge base, then the IA will issue an ADD message to SA_1 to notify receiver A. For details, please refer to the literature in LVMR[10].

1.1.4 Drawbacks of LVMR

LVMR also has its drawbacks.

I. The protocol does not provide fair bandwidth sharing between competing video sessions in a shared link.

II. In addition the authors of LVMR[10] did not mention how to update the knowledge base if the previous failed experiment now becomes possible under the circumstances of link speed upgrade.

III. The knowledge base will grow in complexity as the number of receivers grows. The receivers join and leave. This situation may make the maintenance of the knowledge base formidable.

IV. Moreover the complexity will definitely build up a latency for the layer-adding process of the receivers.

Also referring to section 4.2, p.1065 in LVMR[10], when a new receiver first subscribes, the SA will acknowledge it with SL_{max} meaning that it only can subscribe to the maximum layers in the subnet. The number of layers subscribed will be slowly added until a dynamic equilibrium is reached. In the very beginning of the video session, this approach will impede the video quality obtained even though the receivers have enough capability and link speed to subscribe to the full sets of video layers.

Nevertheless LVMR[10] provides an example of a middle road in the spectrum of rate/congestion control. On the one extreme end of the rate control is in the hand of the sender responsible for collecting information about the network topology and congestion conditions. As mentioned in the above paragraphs, this scheme does not scale well in a heterogeneous network like internet. On the other extreme end is the fully distributed approach in which the receivers do not coordinate their decisions. The approach used in this dissertation is to adopt the hierarchical architecture of LVMR[10] and is described in the next section.

2.0 Sender-Adaptive Rate Control Mechanism - SARC

2.1 Objectives

The objectives are to devise and develop a rate control mechanism for the layered video multicast in the internet. The rate control mechanism should have the following merits.

- Avoidance of feedback implosion from receivers.

  This can be achieved by adoption of the hierarchical control approach. Thus the state of the receivers and information about the network can be obtained by means of multicasting query messages to the subnet. The Subnet Agent (SA) then collects the information and unicasts the status to the Intermediate Agent (IA) in its domain. The sender then collects the status from the IAs through unicast.

- Reduction of latency in the process of adding/dropping video layers

  The rate control mechanism should attempt to minimize or eliminate the latency induced by adding/dropping video layers. Moreover upon starting of the video session receivers with high capacity and fast link speed should have the chance to subscribe to a rate equivalent to that of a full set of video layers.

- Participation of the sender

  The sender no longer just involves in video transmission but have enough information to make decision to avoid congestion by splitting or combining video layers in real-time. The sender collects real-time information about the network conditions through the IAs.

- Heterogeneity

  The sender can split or combine the video layers to meeting the corresponding capacity of the receivers and the link speed from the source to the receivers.

- Required Number of Video Layers at Video Session Start

  The sender will adjust the number of video layers in real-time according to the status of the network bandwidth. It tries to find an optimal number of video layers for a certain amount of receivers at the start of the video session.

2.2 SARC Basic Principles

If the sender can gather the network loading information in real-time, then it can immediately optimize the number of video layers and its transmission rate. This can be realized in the hierarchical approach. SA can poll the receivers' video reception rate by means of multicasting a query message to its corresponding subnet. Then the information about the receiving rate and the number of receivers receiving at this rate are recorded. The information is then sent to the IA through unicast in turn. The sender can then collect the statistics from the IAs. From the statistics, the sender could adjust the number of video layers and transmission rate. The major benefits are the elimination of the add-drop layer mechanism and the chores of maintenance of the knowledge base which might be formidable in size. The following paragraph illustrates the principle with a simple example.

Suppose the sender is sending only one basic video layer at 10Mbps. There are two receivers A and B in subnet A and B each receiving 10Mbps. Later on somehow subnet B gets congested and receiver B starts losing packets. After a while, it can only receive at 5Mbps in stable condition. Thus SA_B can query receiver B through multicast query message and send this information to IA_B. The sender collects the statistics from IA_A and IA_B as shown below :

Receiving Rate	No. of Receivers
10Mbps	1
5Mbps	1

Then the sender can make the decision to split the transmitting video layer into two components, namely one basic video layer transmitting at 5Mbps and one enhancement layer transmitting at 5 Mbps. Then receiver B could then be satisfied by receiving the basic video layer. Receiver A can then get both layers.

This rate control mechanism involves a protocol design and simulation work to prove its functionality. The protocol design is elaborated in Chapter 3.

2.3 Video Encoder and Rate Controller

SARC assumes that there exists a video encoder that can dynamically adjust the number of video layers and can receive a list of target bit rates for each video layer to produce layered video streams at rates that closely follow the target bit rates.

One such video encoder is proposed and used by Brett J. Vickers et al. in “Source-Adaptive Multilayered Multicast Algorithms for Real-Time Video Distribution,” SAMM[12]. The block diagram is reproduced in figure 2-1.

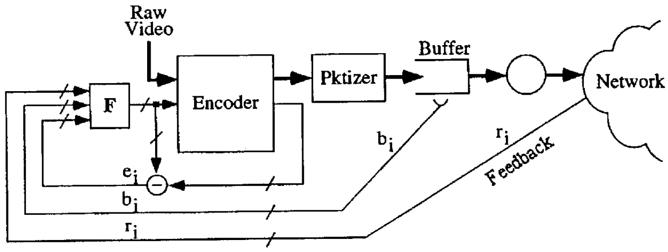


Figure 2-1 Video Encoder and Rate Controller

According to Vickers, the target bit rate for layer i is

$$F_i(r_i, b_i, e_i) = r_i - \left[\alpha e_i + \beta\left(\frac{b_i - T_d r_i}{\tau}\right)\right] \qquad \text{where}$$

F_i Target bit rate for layer i

r_i Rate requested for layer *i* in the most recently received feedback packet

e_i Encoder rate error in layer *i* from the previously encoded block.

b_i Number of bits from layer *i* currently stored in the buffer

T_d Target buffer delay

τ Length of the video block interval

α, β Weighting coefficients

For details, please refer to the literature in SAMM[12].

3.0 SARC Protocol

3.1 SARC Protocol Design

The protocol design involves the following aspects :

- 3.11 Hierarchical architecture of receivers
- 3.12 Role of the Intermediate Agent (IA), Subnet Agent (SA) and the sender
- 3.13 The Video Layer Combining/Splitting Mechanism
- 3.14 Elimination of The Video Layer Add/Drop Process
- 3.15 Starting with the Minimum or the Full Video Rate
- 3.16 System Convergence
- 3.17 Comparison with LVMR
- 3.18 Flowcharts and State diagrams of the Sender and Receivers
- 3.19 Design parameters

3.11 Hierarchical Architecture of Receivers

The hierarchical architecture of receivers is adopted from the LMVR architecture. The group of receivers is divided into domains. Within each domain there is an Intermediate Agent (IA). The IA is responsible for the coordination of the whole domain. Each domain consists of subnets in which receivers resides. Subnet is the at the bottom of the hierarchy. There is a Subnet Agent (SA) inside each subnet. The duty of the SA is to multicast query message to poll the receiving rate of each receiver. The status of the corresponding subnet is thus collected in real time. The IA then periodically poll the SA through a special multicast group for the status information of the subnet. SAs reply through TCP connection to IA. After the IA collects all the status information of its domain, then it can consolidate a link-speed table for its domain stating the receiving rates and the number of receivers receiving at this rate. The sender can then poll each IA through TCP to collect the link-speed table for each domain and makes decision as to whether the video layer should be splitted or combined.

Splitting the video layer into one basic and one or more enhancement layers enables adaptation of different receiving rates by different receivers at their corresponding link speeds. Combining the video layer speeds up the video decoding process for the receiver. Thus the overhead in synchronization of different layers before decoding is reduced. As receivers leave and join the video session, the video layers may be splitted up or combined. It may not be appropriate to combine the video layers immediately after one or some receivers leave the video session as the overhead in splitting and then combining may adversely affect the efficiency of the protocol. The sender should consider the current status of the network before it makes decision to combine the video layers.

Communication within the subnet is done through multicast query message because multicast traffic inside the subnet would not affect those outside. TCP is used in the message communication between the sender and the IA, and also between the IA and the SA to ensure the communication is reliable.

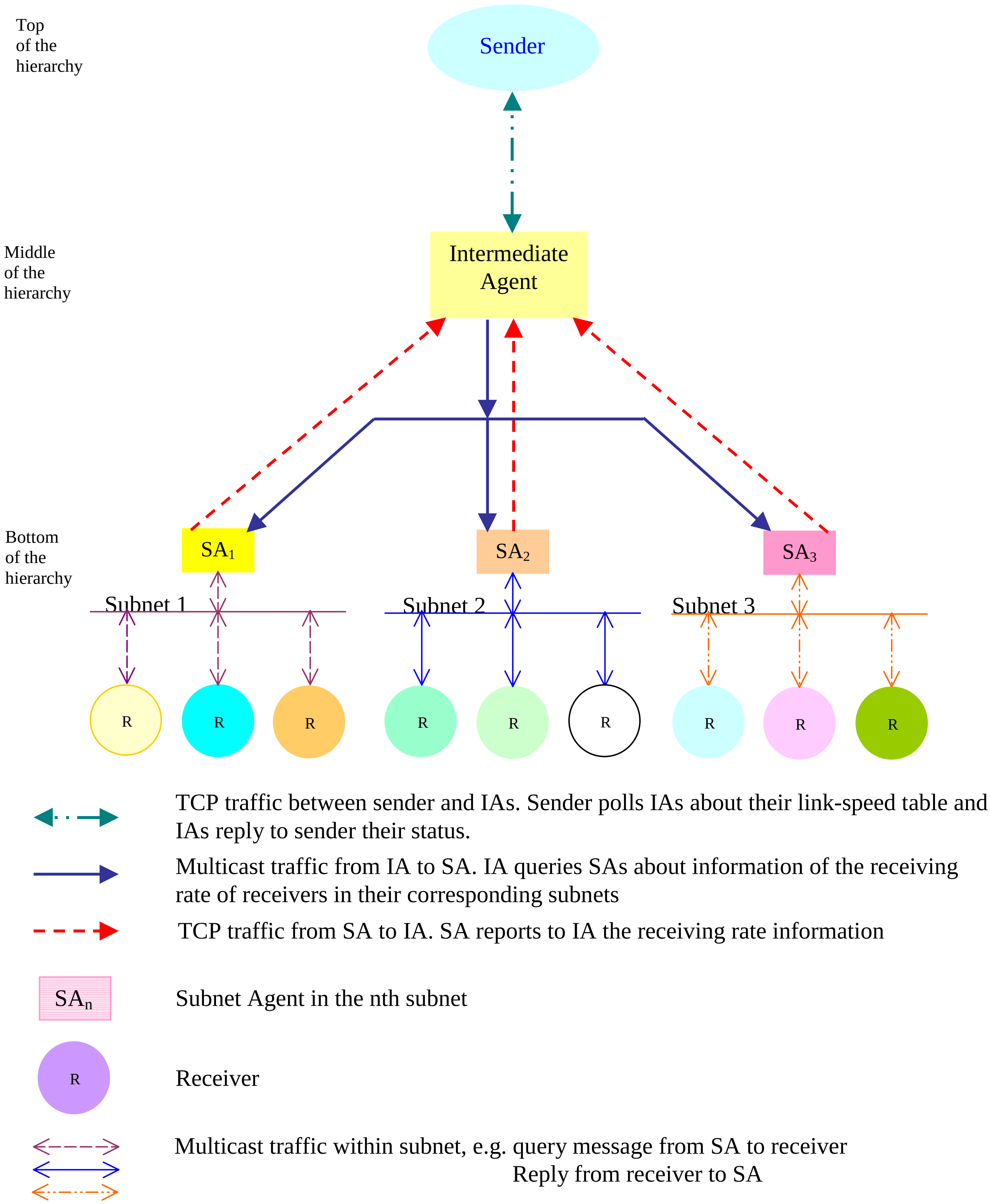
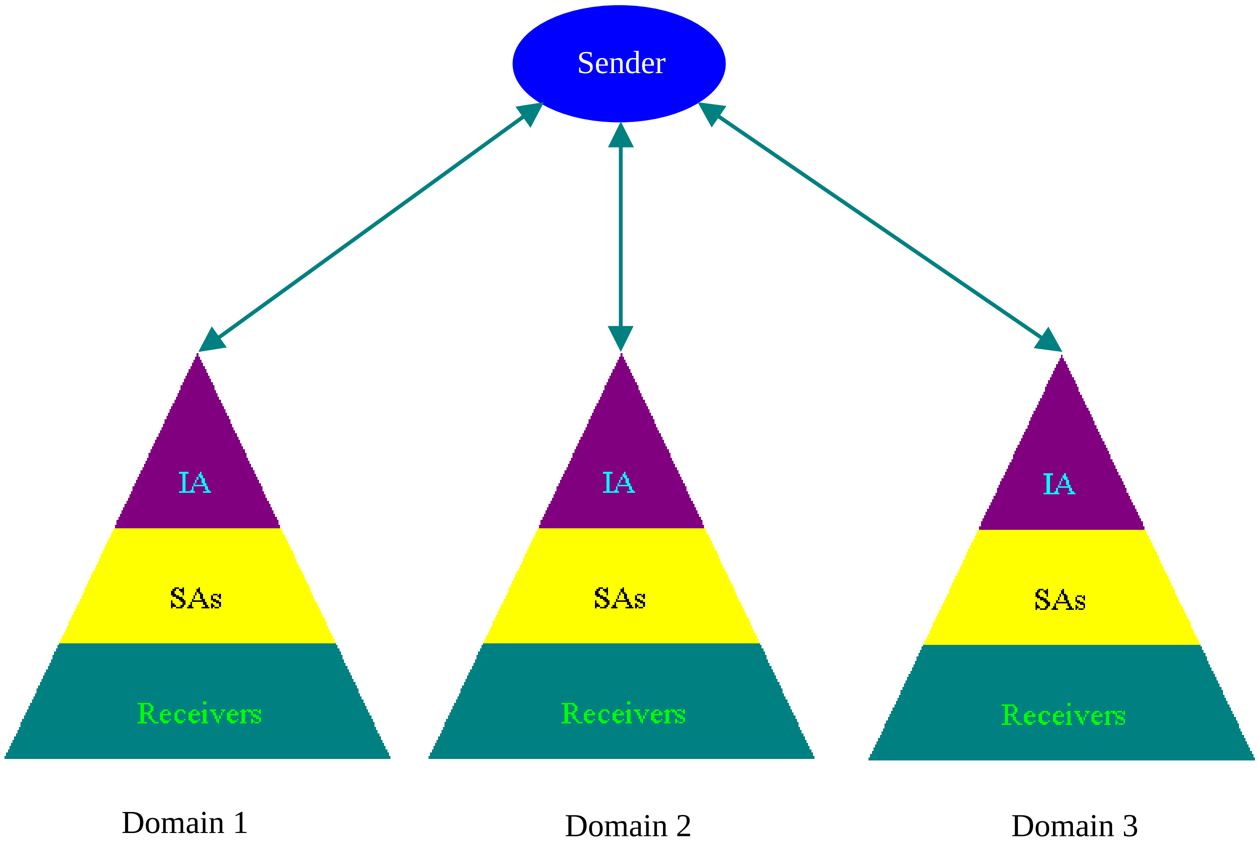


Figure 3-1 Hierarchy of the SARC system in a single domain



TCP traffic between sender and IAs. Sender polls IAs about their link-speed table and IAs reply to sender their status.

Figure 3-2 Hierarchy of SARC with three domains

LVMR[10] uses multiple domains within a multicast group. As stated in p.1063 LVMR[10],

A domain can be a physical region like a geographical area, or a logical region like a corporate intranet or even a logical scope imposed by time-to-live (TTL) field of IP packets.

As the receivers grow in numbers, they can be organized in different domains. The advantage of the use of domain in the hierarchical architecture is that ensures scalability within the multicast group. The reason is that the coordination is done by IA within the domain and the coordination of subnet is done by SA.

3.12 Role of the Intermediate Agent, Subnet Agent and the sender

The Intermediate Agent (IA) coordinates the domain. It periodically multicasts query messages to a special multicast group. All the SAs in the domain subscribed to this group listen to query messages from the IA. The reason to multicast the query message is to improve the efficiency for polling the link-speed status from SAs. It may take a long time to poll from one SA to another until all the SAs are polled especially when the number of SAs is large. The SAs reply through reliable TCP connection to IA about their link-speed status. As far as efficiency is concerned, this approach eliminates the idle time for the IA.

Reply with link speed table

Figure 3-3 shows the multicast query message from IA to SAs

Link Speed	Number Of Receivers
10 Mbps	1
8 Mbps	1

Figure 3-4 shows the link-speed table from SA to IA through TCP

The IA will then collect all the link-speed table from SAs and consolidate them into an integrated link-speed table. Suppose the IA receives the following link-speed tables from SA_1 to SA_3 in its domain,

SA_1

Link Speed	Number Of Receivers
10 Mbps	1
8 Mbps	2

SA_2

Link Speed	Number Of Receivers
10 Mbps	3
5 Mbps	1

SA_3

Link Speed	Number Of Receivers
10 Mbps	2
5 Mbps	1

Then IA will consolidate the link-speed table as follows :

Link Speed	Number Of Receivers
10 Mbps	6
8 Mbps	2
5 Mbps	2

This link-speed table contains the receiving rates and the number of receivers receiving at this particular rate. It also reflects the status of the domain where SA_1, SA_2, SA_3 reside. From this table we can observe that 60% of the receivers in this domain are receiving at full rate (assuming that the full video rate is 10 Mbps), 20% are suffering from packet loss so that the receiving rate decreases to 8Mbps. Moreover 20% are low-capacity receivers that they are receiving at half of the full video bandwidth. Of course the above analysis is based on the assumption that the sender is transmitting a single video stream at 10Mbps.

Now suppose another IA in another domain also consolidates with the following link-speed table :

Link Speed	Number Of Receivers
10 Mbps	2
8 Mbps	8
5 Mbps	1

Now the sender has the information of the two domains, it can further consolidate them into a single table as follows :

Link Speed	Number Of Receivers
10 Mbps	8
8 Mbps	10
5 Mbps	3

From this table we know that nearly half of the receivers are receiving 80% of the full video rate and 14% are receiving at half the video rate. If the video bandwidth can be splitted into layers of different bandwidth, then the capacity of the receivers can then optimally be satisfied. This strategy can also be adopted so that changes in the status of the network can be adapted dynamically in real time. The next paragraph 3.13 delineates the mechanism by which the sender adjusts the video layer to cope with the network status in real time.

3.13 The SARC Video Layer Combining/Splitting Mechanism

The sender will solicit feedback information from each domain through Intermediate Agents (IAs). The IAs will return the consolidated link-speed tables to the sender. The link-speed table are built up periodically through the Subnet Agents (SAs). The SAs poll the receivers periodically about their receiving rates through multicast query message and then construct the link-speed table for IAs' usage. The sender makes use of the link-speed table to determine the state of the network and then combines or splits the video layers and alters the transmission rates accordingly.

The link-speed table consists of tuples of receiving rate and the number of receivers receiving video data at this particular rate. At the bottom layer of the hierarchy, the SAs are responsible for collecting the receiving rate and the number of receivers receiving at this rate by means of multicasting an query message. Each receiver replies the SA with its receiving rate. The SA then builds up the link-speed table for IA's usage.

Consider the following scenario. Domain A consists of Intermediate Agent A (IA_A) and in the beginning only two receivers in subnet A and B are subscribed to the video session. Subnet Agent A and B are each responsible for their subnets. They collect information about the receivers and their corresponding receiving rates by means of multicast query message. When IA_A polls SA_A, SA_A replies with the following table :

Link Speed	Number Of Receivers
10 Mbps	1

When IA_A polls SA_B, SA_B replies with the following table :

Link Speed	Number Of Receivers
10 Mbps	1

The sender S then consolidate the link-speed table as follows :

Link Speed	Number Of Receivers
10 Mbps	2

The sender transmits the video with one basic layer at 10 Mbps during period $0 \le t < T_1$

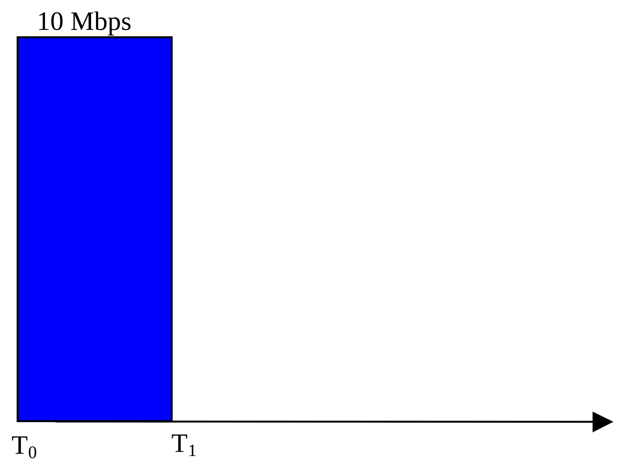


Figure 3-3 A single video base layer of 10 Mbps

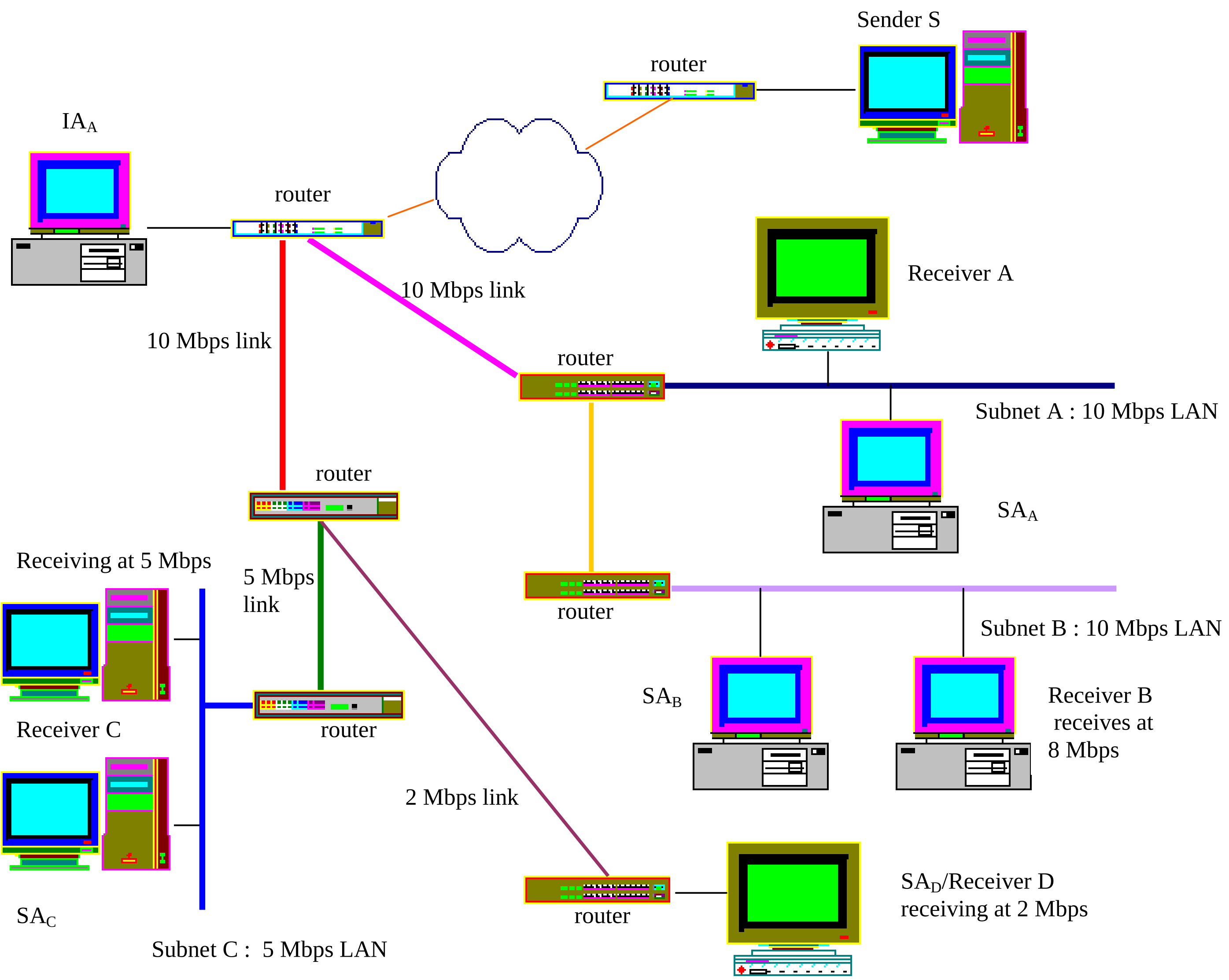


Figure 3-4 A single domain A consisting of subnet A, B, C and D

Suppose after T_1 that the traffic flow from subnet A to subnet B increases such that the receiving rate of receiver B decreases to 8 Mbps. SA_B collects this information and replies with the following table when polled by IA_A.

Link Speed	Number Of Receivers
10 Mbps	1
8 Mbps	1

After receiving this link-speed table, the sender S decides that there are two video transmission rates, i.e., 10 Mbps and 8 Mbps. The rate of base layer is now 8 Mbps because any decrease in this transmission rate will cause significant degradation in picture quality. Therefore the video layer of 10 Mbps is splitted into two layers with a 8 Mbps base layer and an enhancement layer of 2 Mbps. Figure 3 shows the splitting of the video layer after T_1.

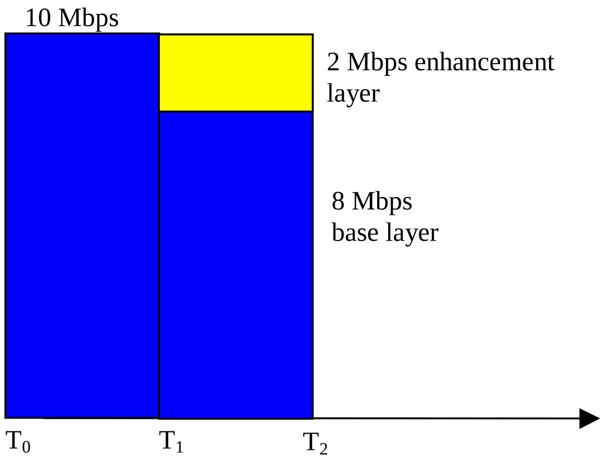


Figure 3-5 During $T_1 \leq t < T_2$ the sender splits the video layer into one of 2 Mbps and one of 8 Mbps (basic layer)

After T_2 receiver C in subnet C and receiver D in subnet D join the video session. Due to their limitation in link speed, receiver C and D achieves receiving rates of 5 Mbps and 2 Mbps respectively. Subnet Agent C and D will reply the query message sent by Intermediate Agent A (IA_A) for domain A. IA_A then consolidates the following link-speed table and replies to sender S when polled.

Link Speed	Number Of Receivers
10 Mbps	1
8 Mbps	1
5 Mbps	1
2 Mbps	1

Then the sender will split the layer as follows and the layer each receiver should subscribe are as follows :

Video Layer	Video Rate	Receiver A	Receiver B	Receiver C	Receiver D
Enhancement layer 3	2 Mbps	Yes	No	No	No
Enhancement layer 2	3 Mbps	Yes	Yes	No	No
Enhancement layer 1	3 Mbps	Yes	Yes	Yes	No
Basic Layer	2 Mbps	Yes	Yes	Yes	Yes

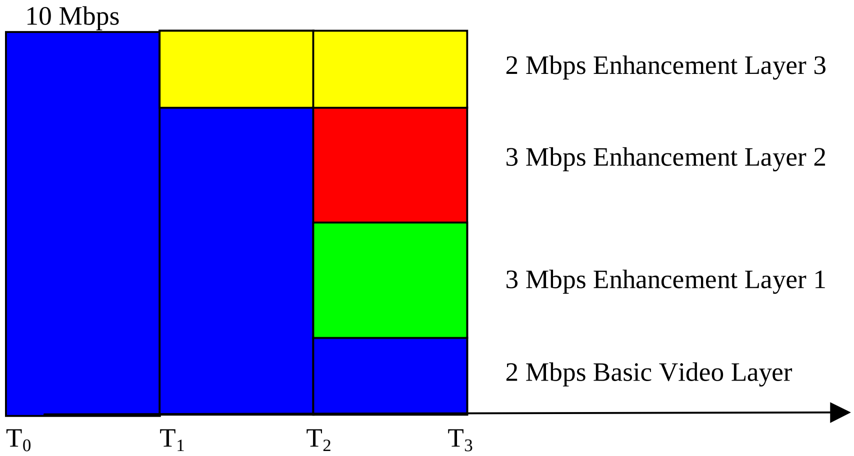


Figure 3-6 The sender splits the video layer during $T_2 \le t < T_3$

Suppose now that receiver D leaves the video session and unsubscribes the basic 2Mbps video layer. The sender will receive the following link-speed table from IA_A :

Link Speed	Number Of Receivers
10 Mbps	1
8 Mbps	1
5 Mbps	1

Then the sender will combine the 2-Mbps basic video layer together with the 3-Mbps enhancement layer as a 5-Mbps basic video layer as follows and the layer each receiver should subscribe now becomes :

Video Layer	Video Rate	Receiver A	Receiver B	Receiver C
Enhancement layer 2	2 Mbps	Yes	No	No
Enhancement layer 1	3 Mbps	Yes	Yes	No
Basic Layer	5 Mbps	Yes	Yes	Yes

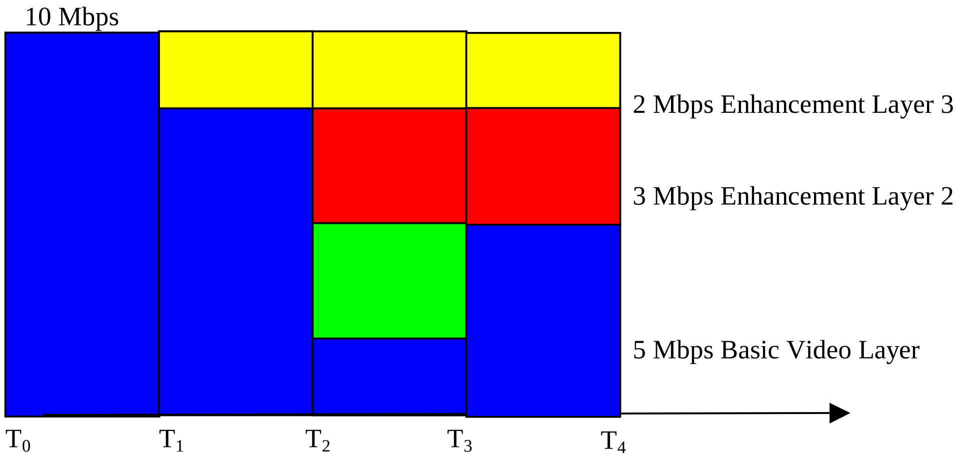


Figure 3-7 The sender combines the video layer during $T_3 \le t < T_4$

Thus the sender can adjust the video transmission rate according to the state of the network by combining/splitting video layers. This can be achieved in real time by means of gathering information from the network through Intermediate Agent in each domain.

3.14 Elimination of the Video Layer Add/Drop Process

When the receiver multicast a "JOIN-LAYER" request to its subnet, the subnet agent (SA) will reply with "ACK" and the link-speed table. The link-speed table contains information of receiving rates and the number of receivers receiving video transmission at this particular rate. Thus the receiver knows the status of the network and will subscribe the number of video layers up to its capacity or the limit of the link capacity.

The information of the link-speed table is made known by the time the receiver receives acknowledgement from SA and before it joins the video layer. It monitors its receiving rate, average frame arrival delay and packet loss rate. From these information it can determine whether it is in the UNLOADED, LOADED or CONGESTED states. The definition of these states is described in paragraph 3.16.

Each video layer is transmitted through a different multicast group so that those receivers subscribed to one of those groups form a multicast tree. If the receiver determines that it is in the UNLOADED state, then it can subscribe the next higher video layer in another multicast group. Afterwards it can monitor its receiving parameters, i.e., receiving rate, average frame arrival delay and packet loss rate to determine whether to further subscribe the next higher video layer or not. The receiver is not required to carry out the video layer adding process if it is in the UNLOADED state.

On the other hand if the receiver is in the CONGESTED state, the receiver may choose to unsubscribe the higher video layer to reduce the packet loss rate. Since the receiver knows the status of the network from the link-speed table through SA, therefore it does not have to go through the video layer dropping process. No experiments like those described in page 3, section 3 in RLM[7] or in page 1065 – 1067 in LVMR[10] are necessary.

3.15 Starting with the Minimum or the Full Video Rate

Given a fixed number of receivers of different capabilities, when the system starts video transmission, there are two ways to begin. It could send out the video at full transmission rate in a single basic video layer. Then the sender will split up the video into appropriate number of video layers after it receives the network information. The sender further adjusts the number of video layer and the transmitting rate of each layer until the entire system converges to an optimal operating point where each receiver satisfies itself by receiving the ample number of video layers.

Alternatively a basic video layer of the minimum transmission rate is transmitted first. Then after the information is collected and the number of video layers is increased progressively until the system converges to the same optimal operating point.

A little thought reveals that the video transmission should be started with the full rate instead of the minimum. The reason is simple. Starting with the full rate, each receiver can receive according to its capability and link speed. If the receiver is of low performance, it cannot receive the video at full rate and packet loss results whereas the high performance one can receive at full rate without video degradation. In other words, the network status is made known at first round. Afterwards the sender collects the information and then splits the video layers accordingly. Then the system should converge to its optimal operating point in two rounds.

On the other hand, starting with the minimum rate, it is highly likely that most of the receivers could be satisfied. The sender then has to add an enhancement layer to further "test" the receivers' capabilities. The receivers may not be entirely satisfied in the second round and have to wait for the third, or even the fourth round up to the physical limit of the number of video layers that the sender can handle. It can be seen that the time it takes for the system to converge on average may exceed two rounds. The best case is two rounds but the worst is more than that.

3.16 System Convergence

It is very important to investigate the time it takes the system from the very beginning to converge to an optimal operating given a fixed number of receivers with different capabilities. This time reflects the settling time of the system and its stability. The shorter it takes, the faster the receivers can reach their optimal operating conditions. Under this optimal operating point each receiver can receive the enough number of video layer commensurate with its processing power, hardware capability and link speed. On the other hand, if the time is long then the system is sluggish in response to network changes and may experience instability as the receivers struggle to subscribe more layers but confront with congestion.

3.17 Comparison with RLM[7] and LVMR[10]

3.171 Disadvantages of RLM[7] and LVMR[10]

In the layered video multicast system of RLM[7] and LVMR[10], the receivers are required to drop video layer when they experience packet loss or congestion above a certain threshold level. On the other hand when the utilization is below a certain threshold level, they are in the UNLOADED state and will add video layer. In the RLM[7] approach, the add/drop video layer process is entirely controlled by the receivers in a distributed manner. Each receiver receives multicast messages from other receivers during their add/drop experiments. If the experiment is successful, the receiver can add a higher video layer otherwise all receivers will exponentially back off their timers. Only failed experiments affect receivers whereas successful ones do not. The disadvantage is that redundant multicast traffic consumes bandwidth. Since the receivers do not know about the status of the network, they rely on this kind of try and error approach to add/drop video layers. It takes time for the entire system to converge to an optimal operating point where each receiver eventually reaches the optimal level of subscription.

As for the LVMR[10], the mechanism is centralized. The IA has a knowledge base which contains rules from heuristics to coordinate the add/drop request from receivers of the same or different subnets. Each receiver has to inform the SA with an ADD message. The SA in turn informs the IA about this request. The IA makes judgement based on past events from the knowledge base. If this ADD request will not cause any conflicts, then the IA will issue a ADD_ACK message to the corresponding SA. The SA in turn informs the receiver to add the video layer. On the other hand if adding the layer causes congestion in its own or other subnet, then the IA will issue FAIL message to the corresponding SA. The SA will in turn inform the receiver that it should back off its timer to add layer.

It is obvious that there is a certain degree of latency from the time the receiver starts sending a ADD request to the time it receives the feedback in this kind of centralized approach. The message travels from receiver to SA, from SA to IA, then after a certain delay of knowledge base inference and decision making, the feedback returns from IA to SA, and then finally from SA to the receiver. For a simple hierarchy, this latency may not be significant. However it is not trivial when the hierarchy is large and the receivers are numerous. As the receivers leave and join the video session, the knowledge base may become very complicated. The rules may be formidable to be processed and analyzed. The delay in decision making is believed to rise non-linearly.

3.172 Counter-Measures for the Add/Drop Process

The approach adopted in this dissertation eliminates the add/drop video layer process so that the latency is reduced to minimum. Since the receiver knows about the state of the network when it receives the acknowledgement from SA, it can immediately start subscribing the video layer. The receiver monitors it receiving rate and the changes in the link condition is then reported to the SA. The SA in turn informs the IA. Finally the sender receives the status of the network and compiles the new link-speed table to be sent to each domain. It takes only a round-trip time for this table to be sent to each receiver. Each receiver now has the latest knowledge of the link condition, it can make decision as to whether more video layers should be added or dropped. It takes the sender a minimal amount of time to consolidate the link-speed table and carry out the video layer splitting/combination.

3.173 Bandwidth Competition in Shared Link

Another problem is that when two receivers competing for the bandwidth in the same link may not receive equal treatment. If the adding of video layer may cause congestion and cause the receiver to loss packets, then the IA will be informed and a rule will be set in the knowledge base. The receiver joining the video session earlier will be less likely to drop more than one higher video layer whereas the late-comer can only subscribe the basic video layer.

The approach in this dissertation would not have this problem. The reason is that the sender will adjust the video layer according to the feedback from the receivers. If the two competing receivers are of each capacity then their receiving rate would be approximately the same. In this way the video layer would be splitted into two halves of equal bandwidth.

The approach in this dissertation enables the sender to respond to constantly changing network conditions by dynamically adjusting the number of video layers it generates as well as the rate at which each layer is transmitted. The feedback information about the network is from the receivers through the coordination effort of SAs and IA in each domain.

3.174 Comparison of the System Convergence

As described paragraph 3.16, the system in this dissertation should converge in about two rounds. As for LVMR[10], the receiver has to subscribe the basic layer and then add one video layer after another. The best case is that all receivers subscribe only the basic layer. It is done in the first round. However it just doesn't make sense for sending layered video if only one layer of video is subscribed. The time it takes to converge depends on the capability of the receivers because receivers of high performance take time to add more layers whereas low performance ones satisfy themselves earlier. Of course the time it takes to resolve the congestion must be accounted for. The LVMR author did not mention the time it takes to converge in her paper.

From page 1068 of LVMR[10] figure 4, the author presented a graph showing the layer adding for the 4 receivers located in four different places. The time it took for the 4 receivers to subscribe stably was about 25 seconds. It may not be representative enough for all cases. The author seemed not to go further into this in this paper.

3.18 Flowcharts and State Diagrams of the Sender and Receivers

The flowcharts of the sender and receiver and the state diagram of the receiver are detailed in figures 3.8 – 3-11.

- Figure 3-8 Flowchart of the Sender
- Figure 3-9 Flowchart of the Receiver
- Figure 3-10 State Diagram of the Sender
- Figure 3-11 State Diagram of the Receiver

Figure 3-8 Flowchart of the Sender

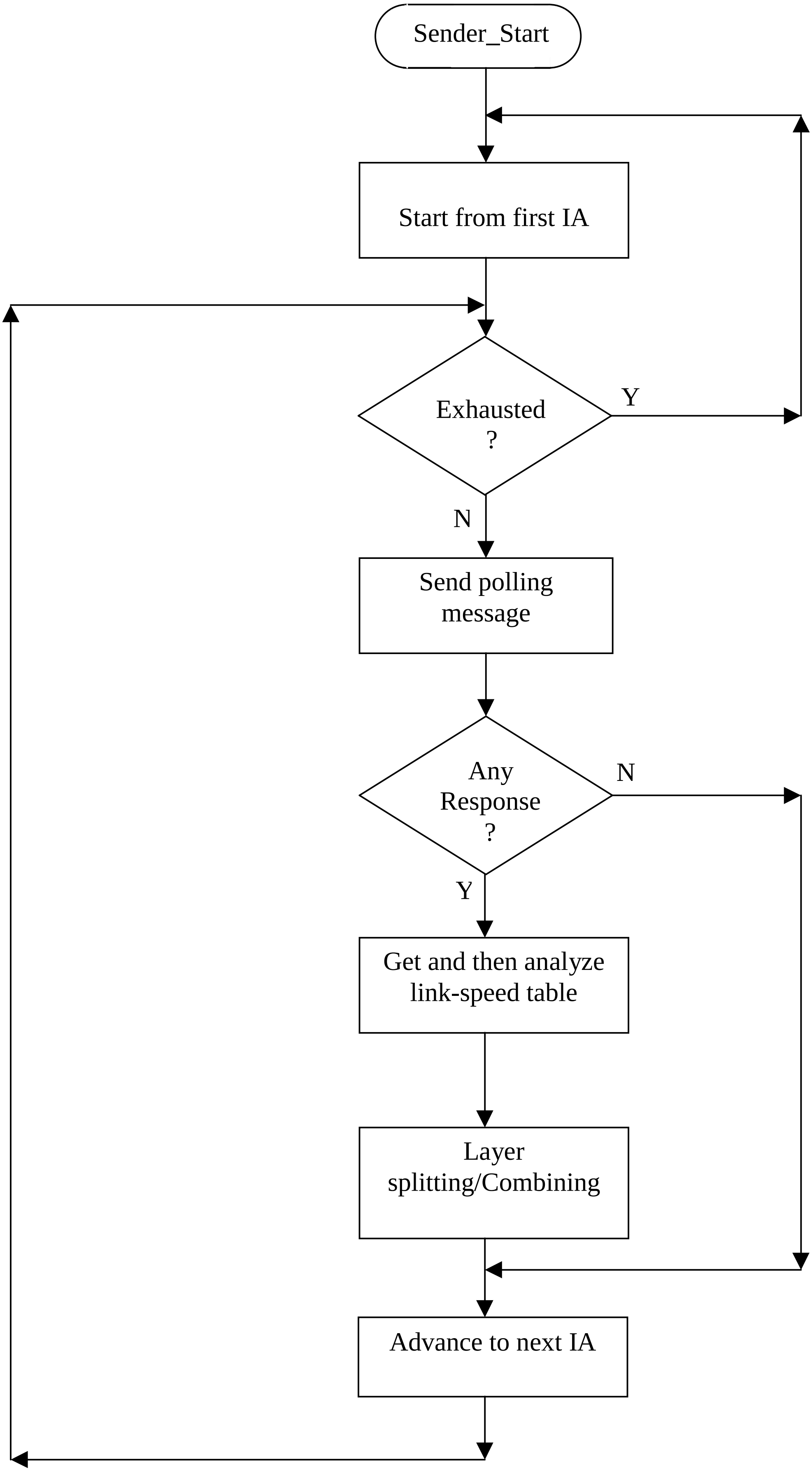
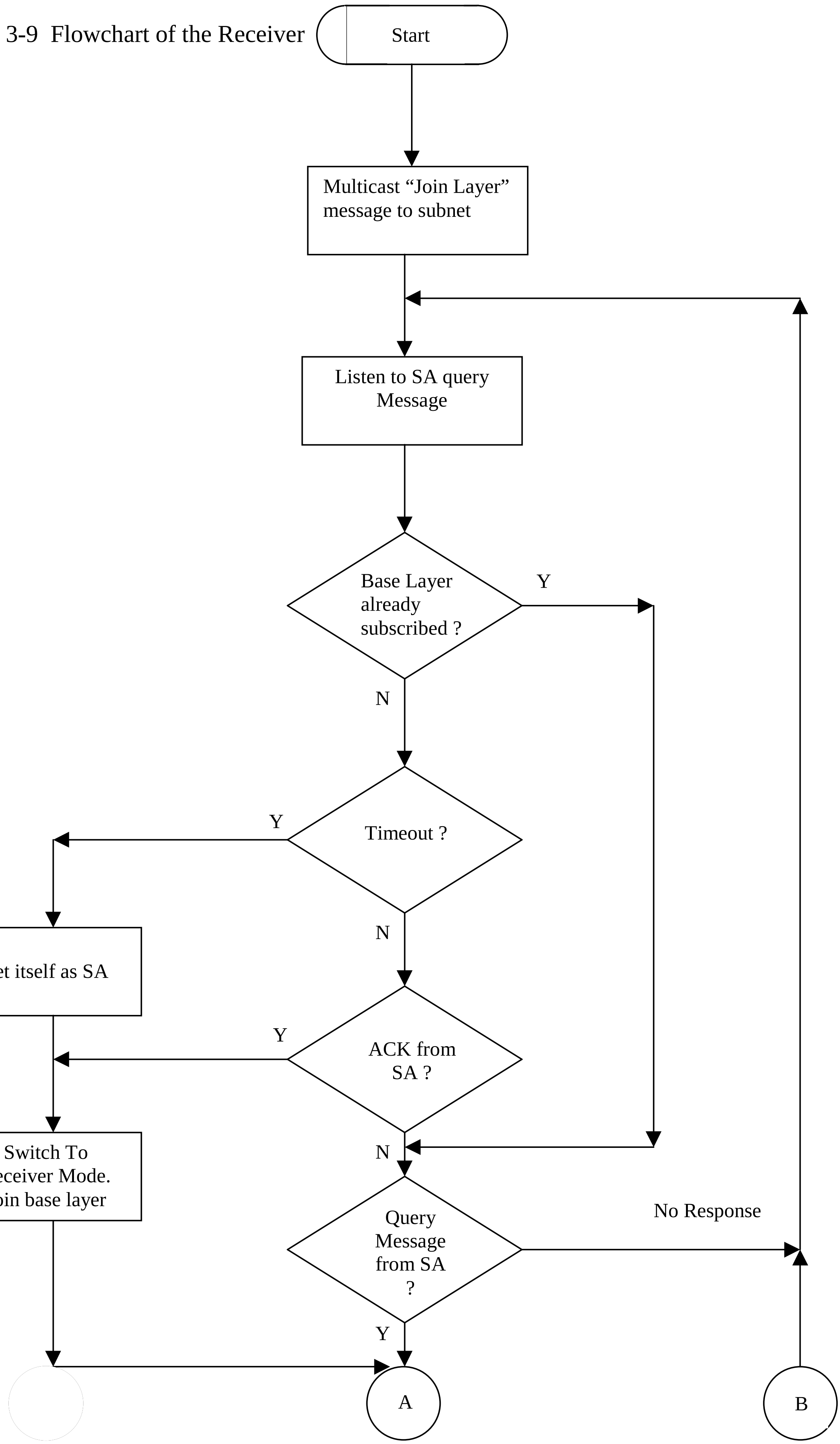


Figure 3-9 Flowchart of the Receiver



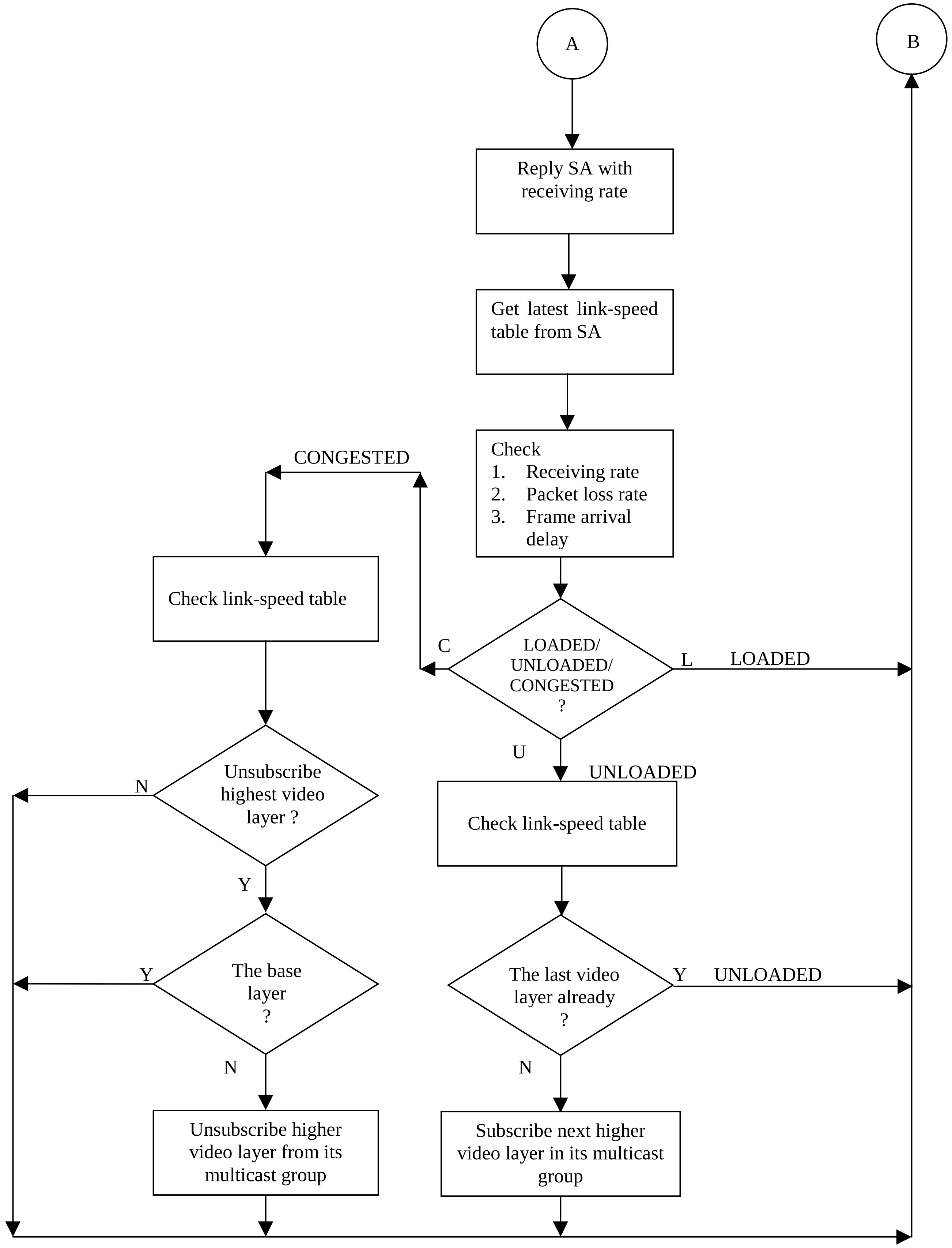

A
B
Reply SA with receiving rate
Get latest link-speed table from SA
Check
1. Receiving rate
2. Packet loss rate
3. Frame arrival delay
CONGESTED
C
LOADED/ UNLOADED/ CONGESTED ?
L
LOADED
U
UNLOADED
Check link-speed table
Check link-speed table
Unsubscribe highest video layer ?
N
Y
The base layer ?
Y
N
The last video layer already ?
Y
UNLOADED
N
Unsubscribe higher video layer from its multicast group
Subscribe next higher video layer in its multicast group

Figure 3-10 State Diagram of the Sender

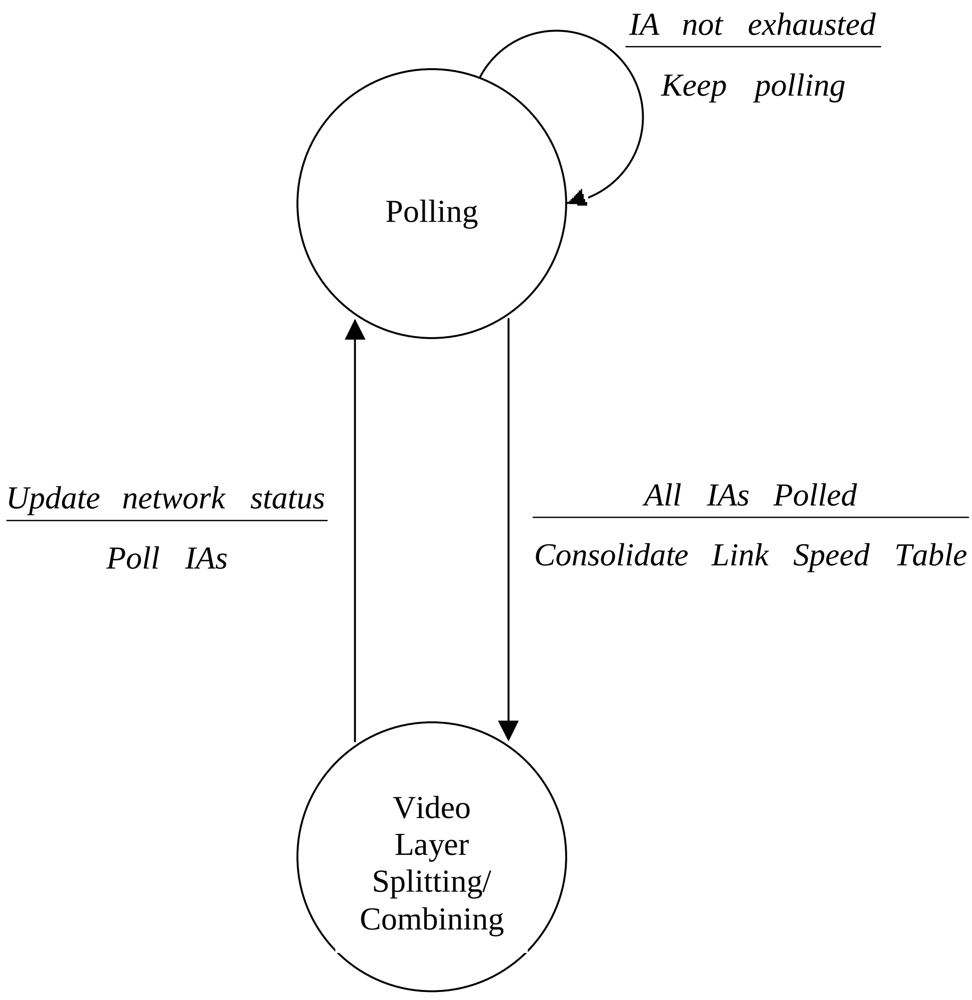
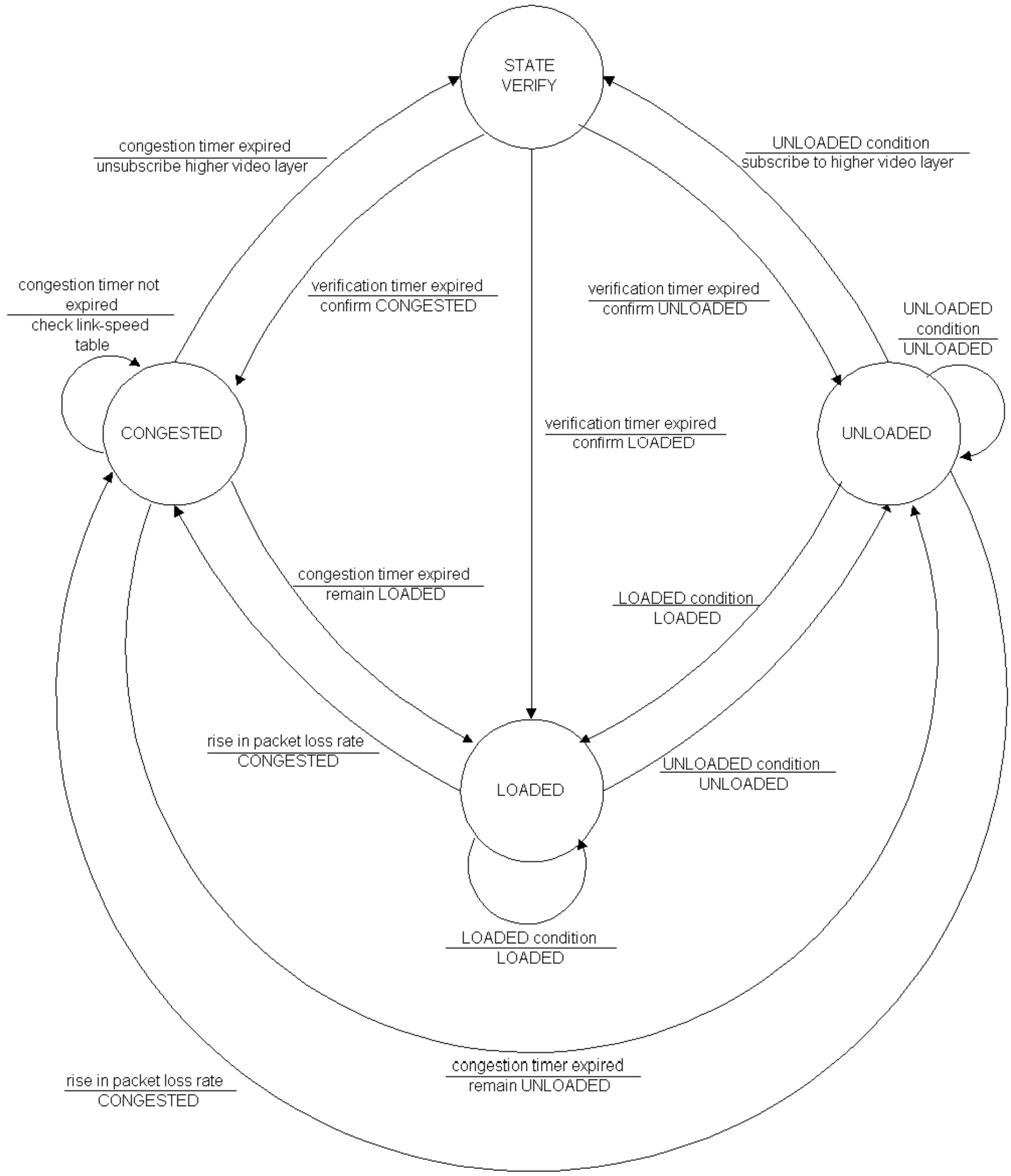


Figure 3-11 State Diagram of the Receiver



3.181 Legendary

Terminology	Descriptions
CONGESTED	The receiver is in the congested state. It is asserted when the packet loss rate is above threshold T_C.
UNLOADED	The receiver is in the unloaded state. It is asserted when the frame arrival delay statistics is below threshold T_U.
LOADED	The receiver is in the loaded state. In other word the receiver is neither in the unloaded or congested state. It is asserted when the packet loss rate is below threshold T_C and the frame arrival delay statistics is below threshold T_U.
STATE VERIFY	This is an indeterminate state in which the status of the receiver has to be observed over time. It arises when the receiving rate is not steady due to video layer joining/leaving by other receivers or itself.
T_C	Packet loss rate threshold value
T_U	Threshold value of percentage of video frame missing deadline

Table 3-1 Description of states of the receiver

3.182 Design Parameters

Parameters	Definitions
T_d	The time the receiver takes to be continuously in a particular state.
T_{sv}	The time the receiver takes to be in the STATE VERIFY state
T_c	The time the receiver takes to be in the CONGESTED state
R	Packet loss rate during a period
Fb_s	The feedback polling rate that the sender collect status of the network from IAs
fb_{IA}	The feedback polling rate that the IA collect status of the domain from SAs
fb_{SA}	The feedback polling rate that the SA collect status of the subnet from receivers in their subnets

4.0 Simulation

4.1 Simulation Network Model to Test the Convergence Time

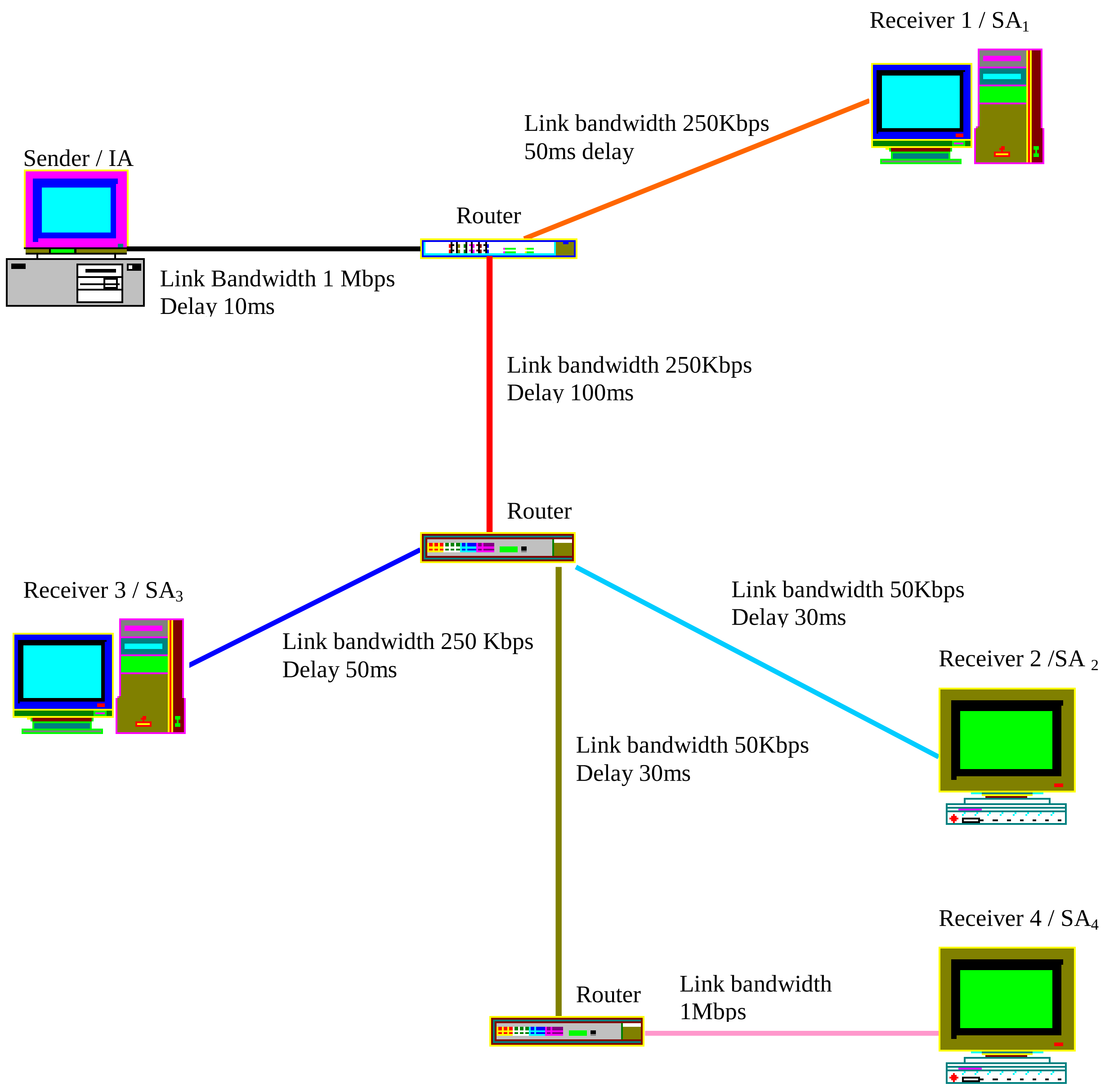


Figure 4-1 Topology of the simulation model to test the convergence time

In figure 4.1 there was one sender with 4 receivers/SAs. The link bandwidths and delays were made different to provide a heterogeneous environment to test for the convergence of the SARC mechanism. The packet size was 1000 bytes. For the sake of simplicity the sender sent a constant bit rate of 250 Kbps data stream. The sender enquired the network status information from the receivers/SAs through multicast and then adjusted the number of video layers to achieve a minimal convergence time.

This same network model was also simulated using RLM[7] so as to compare the convergence time. In the first part of this simulation twenty five layers each of 10 Kbps transmission rate were used to give the same 250 Kbps sending rate. In the second part seven layers were used starting with 32 Kbps as the first layer and subsequent layers were 64K, 128K, 256K, 512K, 1024K, 2048K. The packet size is also 1000 bytes. Under two different criteria, the RLM[7] showed different behaviour.

Three simulations had the same simulation time duration starting from 0 to 1000 seconds. Afterwards, the data was collected from the trace file and put in Microsoft Excel software for analysis. Mainly the convergence time and loss ratio were studied.

4.2 Simulation Tool - *ns*

The simulation tool for SARC is ***ns***. ***ns*** is a discrete event simulator targeted at networking research. ***ns*** provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks.

ns began as a variant of the REAL network simulator in 1989 and has evolved substantially over the past few years. In 1995 ***ns*** development was supported by DARPA through the VINT project at LBL, Xerox PARC, UCB, and USC/ISI.

ns is available at the web site http://www.isi.edu/nsnam/ns It is an object-oriented written in C++ and uses the Otcl interpreter as the command and configuration interface. For SARC simulation, the all-in-one version 2.1b7a was installed on the Red Hat Linux 7.0 platform. The following modules were written for the SARC protocol :

- Test-SARC.tcl

  This is a module written in OTcl script of about 250 lines containing the SARC simulation topology in figure 4-1 to test the SARC mechanism. It specifies all the necessary parameters and records all the event into the trace file for analysis.

- SARC-sender.tcl

  This is a module written in OTcl script of about 300 lines. It defines the mechanism of the sender as depicted in figure 3-10, such as :

  - Collection of feedback from Intermediate Agent,

- Consolidation of the link-speed tables
- Adjustment of the constant bit rate generator
- Transmission of the link-speed table to the Intermediate Agent

- SARC-recr.tcl

  This is a module written in OTcl script of about 400 lines. It defines the mechanism of the receiver as depicted in figure 3-11 such as :

  - Calculation of the packet loss rate
  - Calculation of the packet receiving rate
  - Reply to the Intermediate Agent of the link-speed
  - Subscription of the optimal number of video layers according to the real-time status report of the number of video layers and their receiving rates from Intermediate Agent

- SARC-LossMonitor.tcl

  This is a module written in OTcl script of about 150 lines. It is used to record the time between two successive packets. Hence the packet receiving rate and packet loss rate can then be logged. The function in this module is attached as an agent to each receiver. Each receiver can then call the function in this module to know the real-time receiving rate and packet loss rate.

- SARC-cbr.cc

  This is a module written in C++ of about 100 lines. It is a constant bit rate generator with functions such as :

  - Start/stop
  - Initialization of the packet size
  - Packet inter-arrival time
  - Packet sending rate

  The sender calls these functions to configure the sending rate of each video layer.

- SARC-debug.tcl

  This is a module written in Otcl of about 50 lines. The sender can call the functions in this module to show the status such as :

  - The link-speed table from the Intermediate Agent
  - Consolidated link-speed table content of the sender
  - Sending rate of each video layer

  The receiver can call this debug module to show the status such as :

  - The calculated packet receiving rate
  - The calculated packet loss rate
  - The link-speed table of the receiver
  - The report of the consolidated link speed table from Intermediate Agent

The RLM protocol is readily available in the ***ns*** software package. It consists of three Otcl modules namely test-rlm.tcl, rlm-ns.tcl and rlm.tcl.

4.3 Simulation Result

4.3.1 Simulation Result - SARC

The figure 4.2 shows the simulation result of the sender transmission rate from time 0 to 1000 seconds.

Figure 4-2 Sender transmission rate from time = 0 to time = 1000 second

Originally from time = 0, the sender was transmitting at 250 Kbps. At time = 1 second approximately , the sender had collected the link-speeds information from 4 receivers, it then splitted its basic sending rate from 250Kbps into 3 components, i.e.

1. Basic layer of 50Kbps
2. Enhancement Layer 1 at 100Kbps
3. Enhancement Layer 2 at 100Kbps

It then announced these video layers to those 4 receivers/SAs. These 4 receivers/SAs then subscribed to various layers according to their individual link-speed capacity.

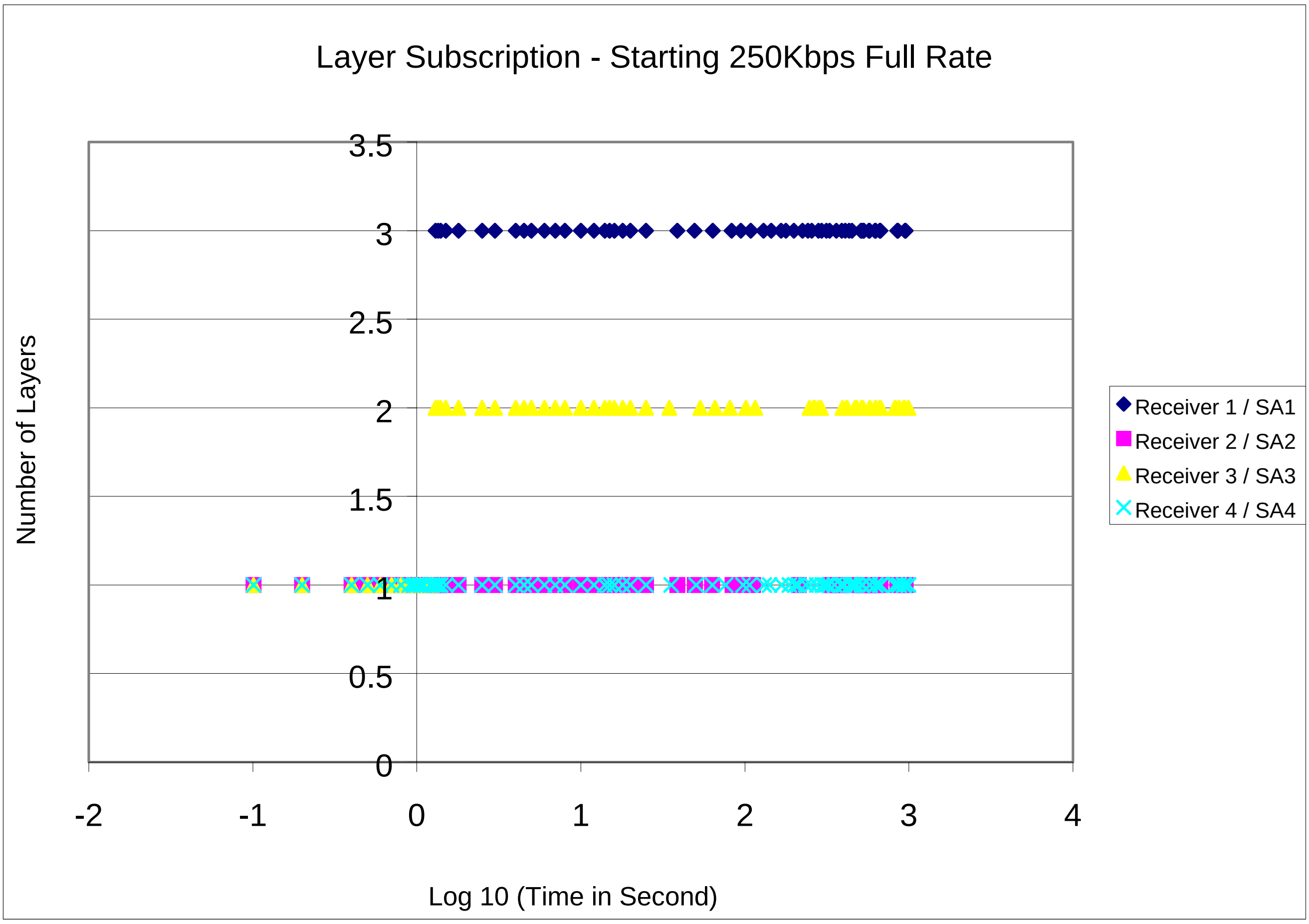


Figure 4-3 The layer subscription of the 4 receivers/SAs

Originally receiver 1 subscribed the basic layer at 250 Kbps. After the sender splitted the video layer at time = 1 second, receiver 1 then subscribed to all 3 layers, i.e., the basic layer - 50Kbps, enhancement layer 1 - 100 Kbps and layer 2 - 100Kbps. The aggregate receiving rate of receiver 1 was 250Kbps which was the full rate from sender.

Receiver 3 could subscribe to the basic layer and enhancement layer 1. The aggregate receiving rate of receiver 3 was (50+100) Kbps = 150Kbps. Note that due to the link bandwidth of 50Kbps for receiver 2, it could only subscribe to the basic layer of 50Kbps. Even though receiver 4 had 1Mbps bandwidth from node 3 to node 6, it was restricted by the link bandwidth from node 2 to node 3 which was 50Kbps. Therefore receiver 4 could only subscribe to the basic layer of 50Kbps. The receiving rates of each receivers were delineated in figure 4.4

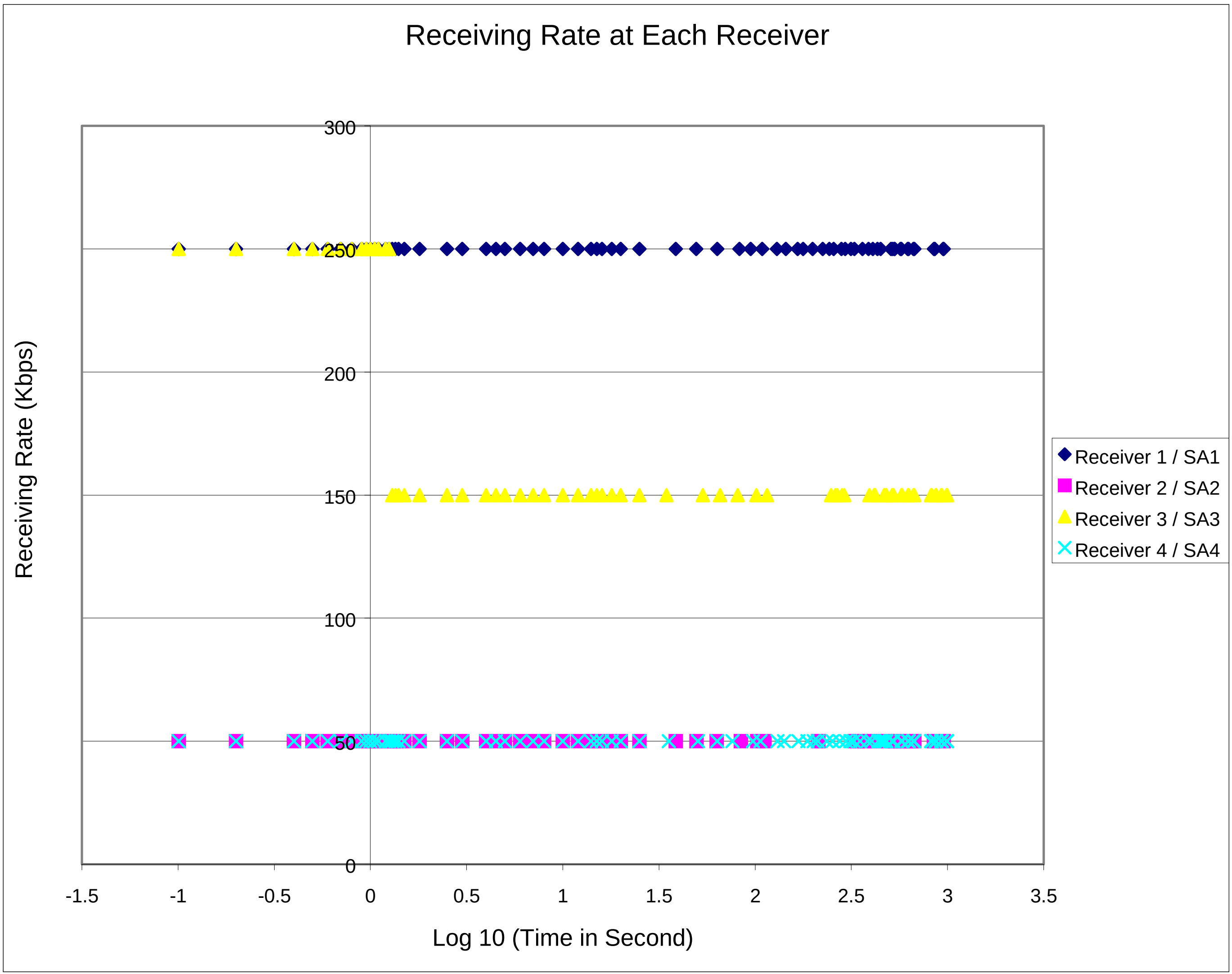


Figure 4-4 The receiving rate of each receiver with SARC protocol

4.3.2 Simulation Result - RLM

The RLM[7] experiments were performed under two different scenarios :

1. Seven layers of 32 Kbps layer exponential distribution :

   32Kbps basic layer,

   64 Kbps enhancement layer 1,

   128 Kbps enhancement layer 2,

   256 Kbps enhancement layer 3,

   512 Kbps enhancement layer 4,

   1024 Kbps enhancement layer 5,

   2048 Kbps enhancement layer 6,

   This was the nominal case to test the convergence time taken by RLM protocol.

2. Twenty five layers of 10Kbps, i.e.,

   (10 Kbps basic layer,

   10 Kbps enhancement layer 1,

   10 Kbps enhancement layer 2,

   10 Kbps enhancement layer 3, …….. and so on up to the 25^{th} layer)

   This was the tough case to test the convergence of the time taken by the RLM protocol.

These two cases of RLM[7] simulation were used as a comparison to SARC protocol. The candidates for comparison were the loss ratio and the convergence time taken by the SARC protocol and RLM[7].

4.3.2.1 RLM Layer Subscription – 32 Kbps Exponential Distribution

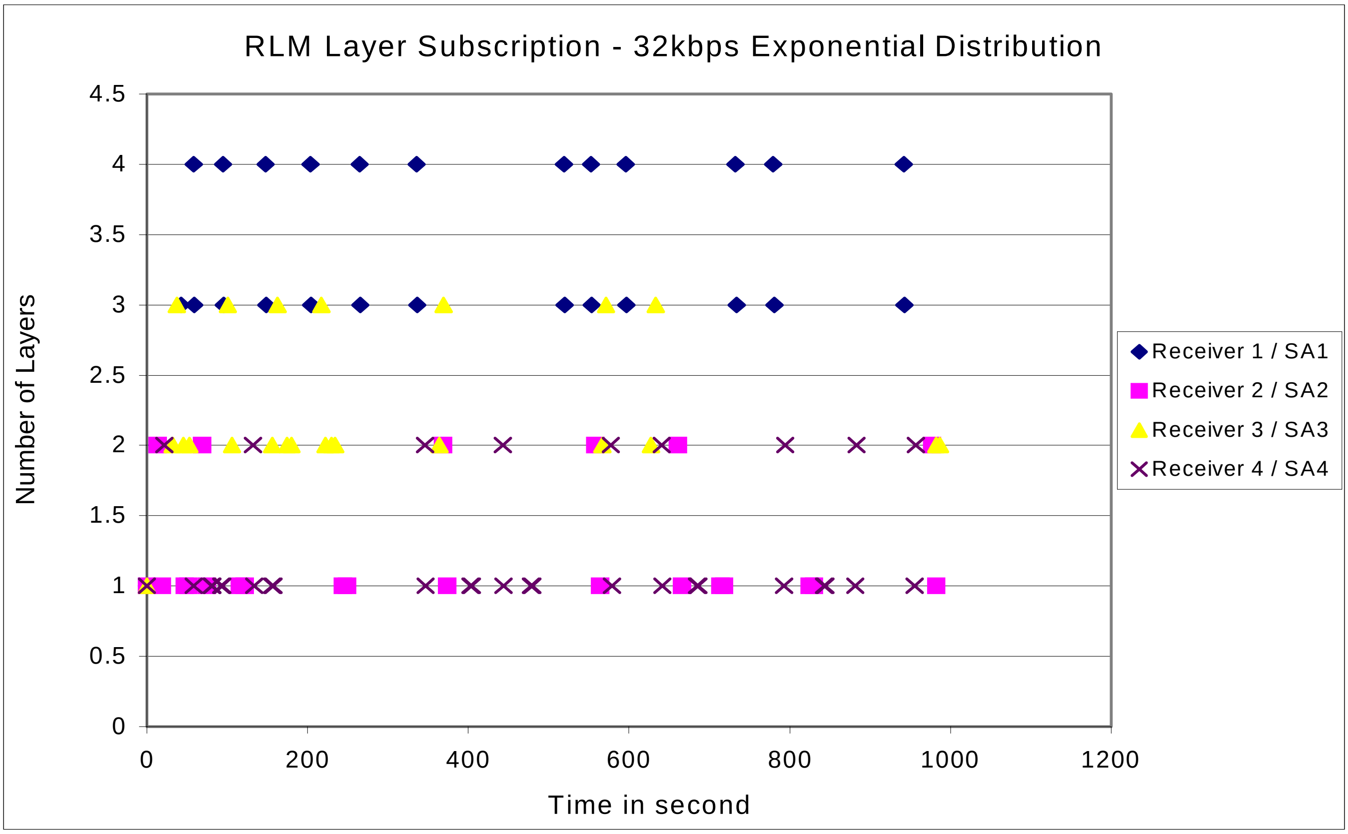


Figure 4-5 The response of the RLM[7] layer subscription

As shown from the figure, receiver 1 subscribed to layer 3 at about 40 seconds. The reason was that there were join experiments from lower layers taking place as receivers 2, 3, and 4 were trying to subscribe the basic and enhancement layer. These join experiments from lower layers delayed receiver 1 in joining layer 3. Receiver 2 and 4 could subscribe to the basic 32 Kbps layer and occasionally attempt the enhancement layer 1 without success. Receiver 3 could subscribe the basic 32 Kbps and enhancement layer 1 – 64 Kbps. It also attempted the enhancement layer 2 – 128 Kbps layer without success. As can be seen the overall response was much slower than the SARC approach.

4.3.2.2 RLM Receiving Rate – 32 Kbps Exponential Distribution

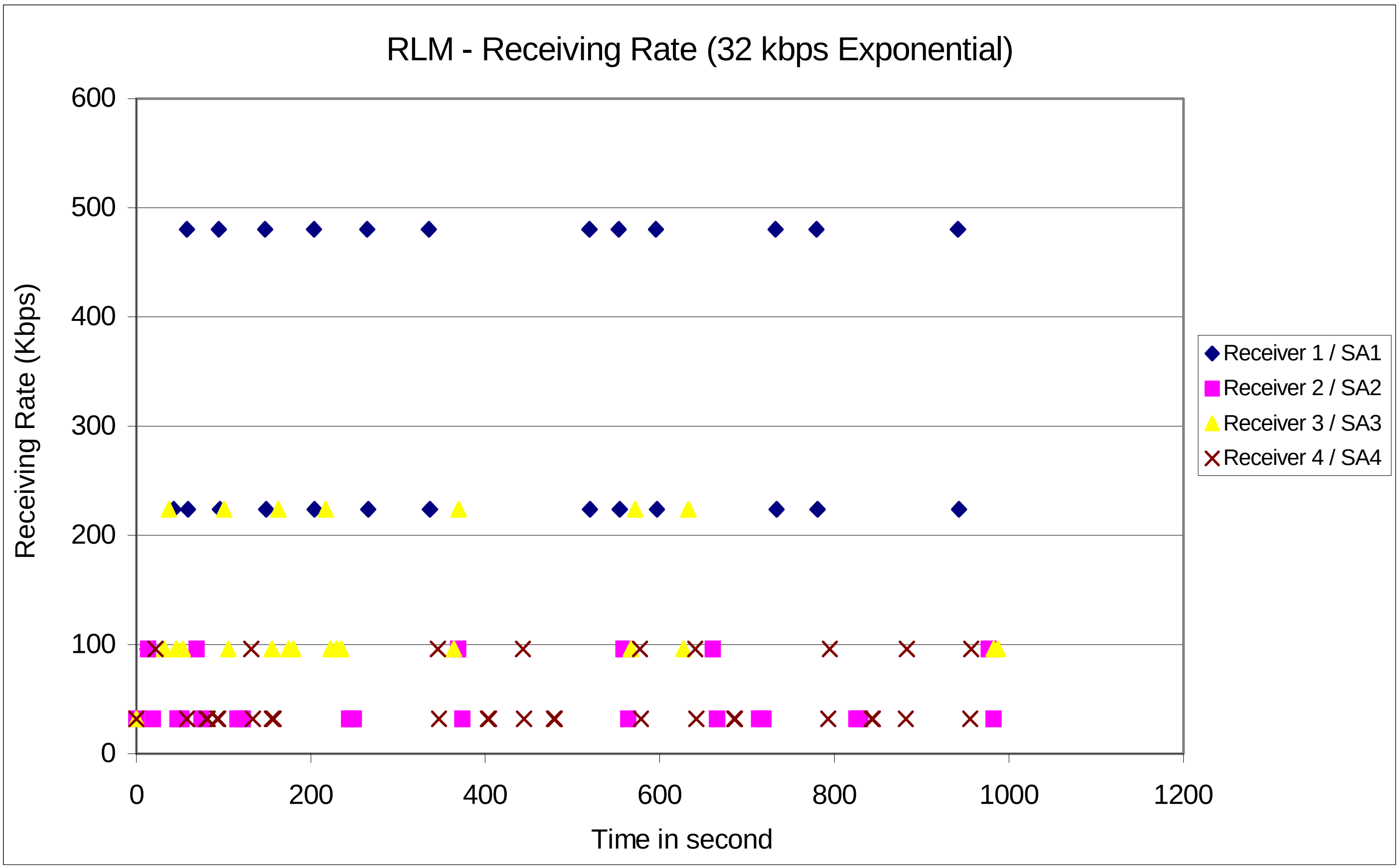


Figure 4-6 The RLM receiving rate with 32 Kbps exponential distribution layer

Receiver 1 subscribed three video layers at about time = 40 seconds. The aggregate receiving rate was (32+64+128) = 224Kbps.

Receiver 3 subscribed enhancement layer 1 so its aggregate receiving rate was (32+64) = 96 Kbps. Of course this was less half of its link bandwidth capacity. Therefore it tried to subscribe to enhancement layer 2 – 128 Kbps. However if really succeeded, it would have been receiving at 224Kbps. By then the total aggregate receiving rate from receiver 2, 3, and 4 would sum up to (32+32+224) = 288Kbps. This was much larger then the link bandwidth capacity 250Kbps offered from node 1 to node 2.

The aggregate receiving rate of receiver 4 was only 32 Kbps because it was effectively limited by the 50 Kbps link bandwidth from node 2 to node 3.

Receiver 2 was also receiving at 32 Kbps.

4.3.2.3 RLM Layer Subscription – 10 Kbps Thin Layer

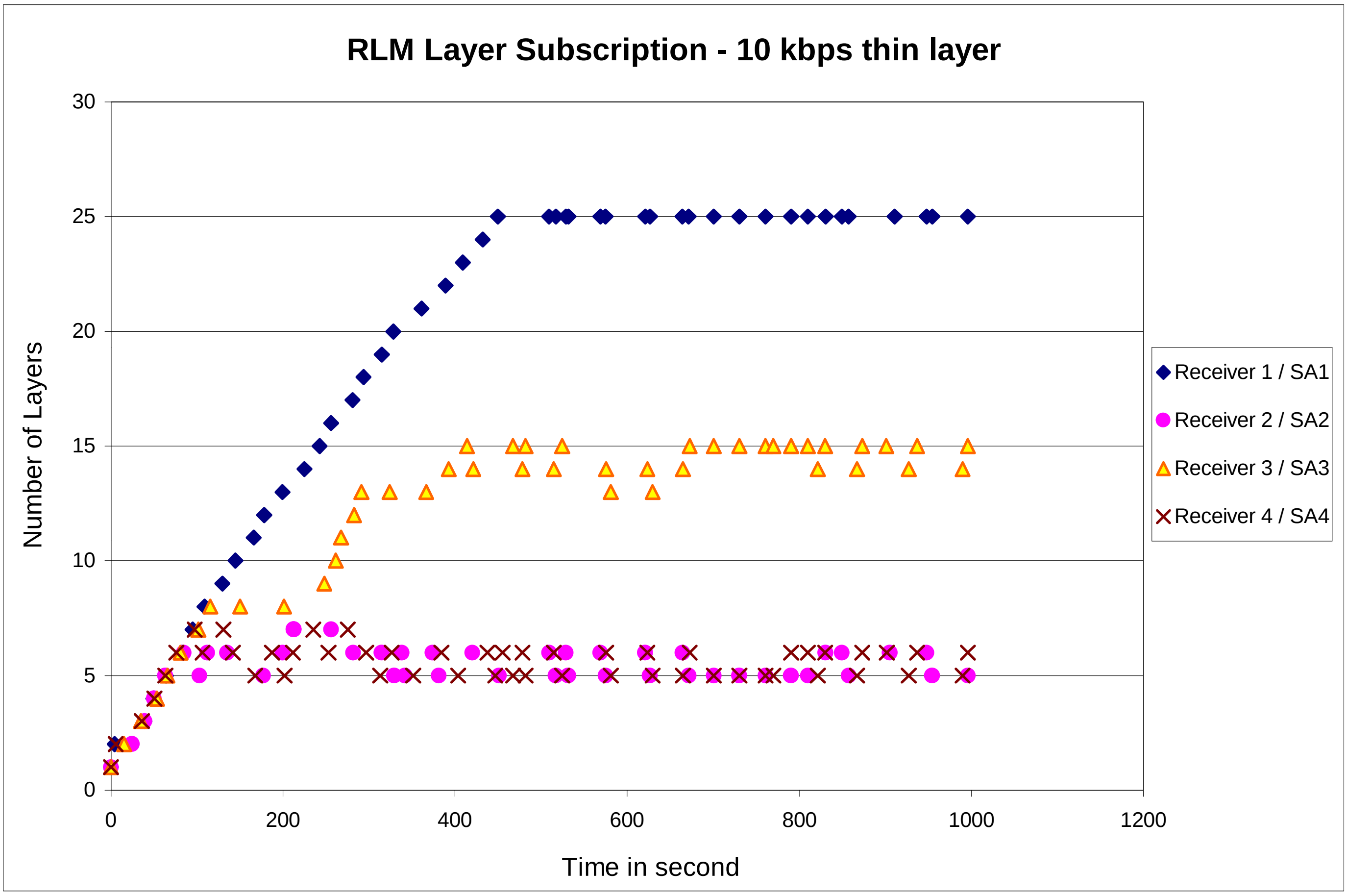


Figure 4-7 The RLM[7] layer subscription with 10 Kbps thin layers

It was shown from figure 4.7 that it took more than 440 seconds for receiver 1 to subscribe to the 25th layer. The minimum value of the join timer was fixed to 5 second in RLM[7] protocol. The smaller the layer granularity, the slower the convergence would be. Moreover the loss threshold of RLM[7] was set to 25% of the packets, so with such small layers, receiver 1 would never enter into a congestion period where it experienced more than 25% of the packets.

4.3.2.4 RLM Receiving Rate – 10 Kbps Thin Layer

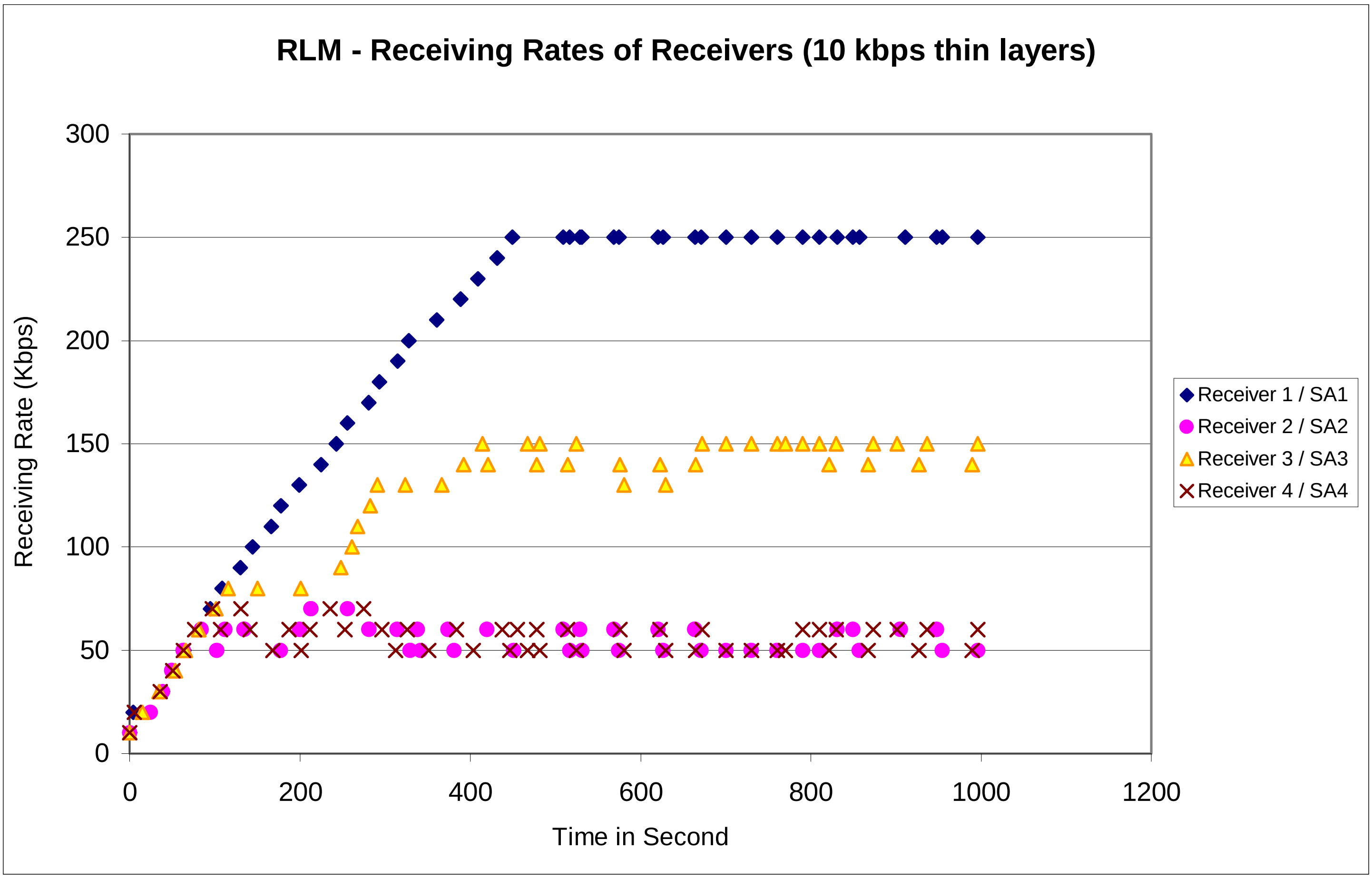


Figure 4-8 The RLM[7] receiving rates with 10 Kbps thin layer

It was obvious that it took about 440 seconds for receiver 1 to achieve its optimal receiving rate at 250 Kbps. It took Receiver 3 almost the same time to reach a receiving rate of 150 Kbps. Receiver 2 and 4 took over 100 seconds to reach at an optimal rate of 50 Kbps.

This situation could never happened in SARC protocol because the real-time status was made known to the sender from the receivers/SAs. The layer splitting information was transmitted immediately back to the receivers/SAs. Therefore the receivers/SAs could subscribe their optimal layers right the way.

4.3.3 Simulation Result – Loss Ratio

Loss ratio for a particular flow *i* is defined to be

$$= \frac{The \quad number \quad of \quad packets \quad dropped \quad in \quad flow \quad i}{Total \quad packets \quad received \quad in \quad flow \quad i} \%$$

This is the performance index to be compared with RLM[7].

4.3.3.1 Loss Ratio - The SARC Approach

In the experiment of SARC, there were three layers after the 250K bps video was splitted. The first flow was the basic layer – 50 Kbps. The second flow was the enhancement layer 1 – 100K bps. The third flow was the enhancement layer 2 – 100K bps. The data were obtained from the trace file as in table 4-1.

	Number of Packets Received	Number of Packets Dropped	Loss Ratio
Basic Layer – 50Kbps	34893	6	0.000171954
Enhancement Layer 1 – 100Kbps	15019	0	0
Enhancement Layer 2 – 100Kbps	8956	0	0

Table 4-1 Loss ratio for the SARC approach.

The number of packets dropped was very low compared to the number of packets received in all three video layers. It showed that receiver 1 was stable after subscribing all three layers. Receiver 3 was stable after subscribing two layers whereas receiver 2 and 4 were stable at the basic layer. The mean loss ratio across 3 video layers is 0.0000573 or 0.00573%.

4.3.3.2 Loss Ratio – RLM[7] 32Kbps Exponential Distribution

In the case of RLM[7] 32Kbps exponential distribution approach, the following data were obtained from the trace file.

	Number of Packets Received	Number of Packets Dropped	Loss Ratio
Basic Layer – 32Kbps	4014	158	0.03936223
Enhancement Layer 1 – 64Kbps	7886	253	0.03208217
Enhancement Layer 2 – 128Kbps	15228	77	0.00505647
Enhancement Layer 3 – 256Kbps	328	10	0.0304878
Enhancement Layer 4 – 512Kbps	0	0	0
Enhancement Layer 5 – 1024Kbps	0	0	0
Enhancement Layer 6 – 2048Kbps	0	0	0

Table 4.2 Loss Ratio – RLM[7] 32Kbps Exponential Distribution

From table 4-2 the mean loss ratio across all video layers is 0.0152841 or 1.5%

4.3.3.3 Loss Ratio – RLM[7] 10Kbps Thin Layers

In the case of RLM[7] 10Kbps thin layer approach, the following data were obtained from the trace file.

	Number of Packets Received	Number of Packets Dropped	Loss Ratio
Basic Layer – 10Kbps	4896	489	0.099877451
Enhancement Layer 1 – 10Kbps	4837	392	0.081041968
Enhancement Layer 2 – 10Kbps	4642	255	0.054933218
Enhancement Layer 3 – 10Kbps	4585	174	0.037949836
Enhancement Layer 4 – 10Kbps	4594	168	0.036569438
Enhancement Layer 5 – 10Kbps	2933	120	0.04091374
Enhancement Layer 6 – 10Kbps	2261	13	0.005749668
Enhancement Layer 7 – 10Kbps	2236	6	0.002683363
Enhancement Layer 8 – 10Kbps	2172	2	0.00092081
Enhancement Layer 9 – 10Kbps	2109	4	0.001896633
Enhancement Layer 10 – 10Kbps	2101	2	0.000951928
Enhancement Layer 11 – 10Kbps	2051	8	0.003900536
Enhancement Layer 12 – 10Kbps	2033	0	0
Enhancement Layer 13 – 10Kbps	1968	8	0.004065041
Enhancement Layer 14 – 10Kbps	1898	8	0.004214963
Enhancement Layer 15 – 10Kbps	1886	11	0.00583245
Enhancement Layer 16 – 10Kbps	1793	5	0.002788622
Enhancement Layer 17 – 10Kbps	1782	13	0.007295174
Enhancement Layer 18 – 10Kbps	1736	7	0.004032258
Enhancement Layer 19 – 10Kbps	1638	17	0.01037851
Enhancement Layer 20 – 10Kbps	1618	3	0.001854141
Enhancement Layer 21 – 10Kbps	1548	7	0.004521964
Enhancement Layer 22 – 10Kbps	1522	8	0.005256242
Enhancement Layer 23 – 10Kbps	1438	6	0.004172462
Enhancement Layer 24 – 10Kbps	700	10	0.014285714

Table 4-3 Loss Ratio – RLM[7] 10Kbps Thin Layer

From table 4-3, the mean loss ratio for the RLM[7] 10 Kbps thin layer approach is 0.017443445 or 1.7% across all video layers.

5.0 Conclusions and Future Work

5.1 Comparison between SARC and RLM[7]

The following table illustrates the performance of two approaches :

Sender-Adaptive Rate Control Mechanism - SARC and RLM[7]

	SARC	RLM[7]	
		10Kbps Thin Layer	32Kbps Exponential Distribution
Convergence Time Taken	1 second	440 Seconds	40 Seconds
Loss Ratio	0.00573%	1.7%	1.5%

Table 5-1 Comparison between SARC and RLM[7]

As shown in this table that the SARC approach proposed in this dissertation converges much faster than the RLM[7] approach. The loss ratio is also very low compared with RLM[7]. The receivers can quickly reach their optimal receiving rate without layer subscription oscillations. In contrast RLM[7] exhibits inconsistent behaviour in the 10Kbps thin layer and 32Kbps exponential distribution simulations. As compared to SARC, the receivers in RLM[7] exhibit much higher loss ratio and are not stable in their layer subscription. They oscillate between upper and lower video layers about their optimal layer from time to time the receivers add or drop video layer.

5.2 Sender-Adaptive Rate Control Mechanism - SARC

Multi-layered video is not sufficient to provide ideal network bandwidth utilization or video quality. The SARC approach advocates the participation of the sender in order to improve the bandwidth utilization of the network. In addition, the sender must respond to constantly changing network conditions by dynamically adjusting the number of video layers it generates as well as the rate at which each layer is transmitted to optimize the quality of video received by each of the destinations. The SARC mechanism avoids the feedback implosion by communicating with the Intermediate Agent. The sender finds the optimal number of video layers at the start of the video session after an enquiry of the real-time status of the network bandwidth. This dissertation also evaluated the performance of the SARC approach with that of RLM[7]. The simulation result shows a very low loss ratio and a fast convergence time as compared to that of RLM[7].

5.3 Future Work

This dissertation shows the Sender-Adaptive Rate Control mechanism. The simulation is based on one sender and 4 receivers. However in a multicast video session, the potential number of receivers may be very large such that the link speed could be very diverse. This diversity in the link speed could cause a large number of video layers more than the sender can generate. Further research should be carried out to study the effects of scalability of the video quality as the number of video layers is larger than the sender can generate.

Secondly the end-to-end propagation delay has a certain impact on the loss ratio. Variation of end-to-end propagation delay means different network size. So how SARC mechanism would react to network size should also be studied further.

Thirdly receivers joining lately in the video session may have different link speed requirements such that the sender has to change the video layer combination. It may not be stable for the entire system if the sender constantly splits or combines the video layers. If so those receivers already joined the multicast group have to drop the old video layers and re-subscribe the new video layers. Further study should be conducted to investigate this problem.

Reference

[1] Nachum Shacham, "Multipoint Communication By Hierarchically Encoded Data," Proc. of IEEE INFOCOM' 92, 1992, pp.2107-2113.

[2] H. Kanakia, P.Mishra, A.Reibman, "An Adaptive Congestion Control Scheme For Real-Time Packet Video Transport," Proc. ACM Sigcomm'93, San Francisco, CA, September 1993, pp. 20–31.

[3] L. Zhang, S. Deering, D. Estrin, S. Shenker and D. Zappala, "RSVP: A New Resource Reservation Protocol," IEEE Network Magazine, September 1993.

[4] J-C. Bolot, T. Turtelli, and I. Wakeman,"Scalable Feedback Control For Multicast Video Distribution in The Internet," Proc. ACM SIGCOMM 1994, September 1994, pp. 58-67.

[5] Domenico Ferrari, Anindo Banerjea and Hui Zhang. "Network Support For Multi-media," Computer Networks and ISDN Systems, Vol.20, 1994, pp. 1267-1280.

[6] S. Cheung, M. H. Ammar, and X. Li, "On the Use of Destination Set Grouping to Improve Fairness in Multicast Video Distribution," Proc. of IEEE INFOCOM'96 (San Francisco, CA), Mar. 1996, pp.553-560.

[7] S. McCanne, V. Jacobson and Martin Vetterli, "Receiver-Driven Layered Multicast," Proc. ACM SIGCOMM, August 1996, pp.117-130.

[8] Thierry Turletti, Christian Huitema, "Videoconferencing on the Internet," IEEE/ACM Transactions on networking, Vol. 4 No. 3 June 1996, pp.340–351.

[9] Sanjoy Paul, Multicasting On The Internet And Its Applications, Kluwer Academic Publishers, pp.287–324, 1998.

[10] X. Li, S. Paul, and M. H. Ammar, “Layered Video Multicast with Retransmissions (LVMR):Evaluation of Hierarchical Rate Control,” Proc. of IEEE INFOCOM’ 98 April 1998, pp.1062-1072.

[11] Bin Wang, and Jennifer C. Hou, “QoS-Based Multicast Routing for Distributing Layered Video to Heterogeneous Receivers in Rate-based Networks,” Proc. of IEEE INFOCOM 2000, pp. 480–489.

[12] Brett J. Vickers, Celio Albuquerque, Tatsuya Suda, “Source-Adaptive Multilayered Multicast Algorithms for Real-Time Video Distribution,” IEEE/ACM Transactions on Networking, Vol. 8, No. 6 December 2000, pp.720 –733.

[13] Dapeng Wu, Yiwei Thomas Hou, WenWu Zhu, Ya-Qin Zhang, Jon M. Peha, “Streaming Video over the Internet : Approaches and Directions,” IEEE Transactions on Circuits and Systems For Video Technology, Vol. 11, No. 3 March 2001, pp. 282–300.