

The Multicast Scheme of ATM Networks for Multimedia Transmissions Based on SD Channel

Wang Ting, Hao Jianzhong, Zhuo Li, Wang Shuling, Wei Yan
Xu Zhou Air Force College, China

Abstract—In this paper, we propose a bi-directional multipoint-to-multipoint multicast scheme, a SD channel-based Multicast with Round robin Access (SDRAM), for ATM networks, which uses a single tree for a multicast group consisting of multiple participants that are either senders, receivers, or a mix of both. We first discuss why the resequencer model will not be suitable for multimedia traffics, and then propose the SDRAM scheme to solve the problems, and finally compare our scheme with the resequencer model through simulation. Results show the mean queuing delays and mean inter-PDU delays of our scheme are not sensitive to mean PDU size while the mean queuing delays and mean inter-PDU delays of the resequencer scheme are very sensitive to mean PDU size. **Index Terms**—ATM, multicast, resequencer, round robin, shared tree.

Keywords—TM Networks, Multicast, SD Channel, PDU Delay

I. INTRODUCTION

Multicast, or selective broadcast, is defined as a subset of all hosts in a network that could communicate logically with each other simultaneously. For each node, multicast is considered as one point-to-multipoint connection. As a whole, multicast can be viewed logically as a multipoint-to-multipoint connection. Therefore, emphasize on the multipoint-to-multipoint multicast architecture. Because just one single copy of data is sent out from sender, multicast has the advantage of saving network bandwidth. In addition, it also has the advantage of logical addressing. With the logical addressing, one sender uses a single multicast group address to transmit and receive data. It is not necessary to identify the locations of all other members.

In this paper, we propose a bi-directional multipoint-to-multipoint multicast scheme, a SD channel-based Multicast with Round robin Access (SDRAM), for ATM networks, which uses a single tree for a multicast group consisting of multiple participants that are either senders, receivers, or a mix of both. We first discuss why the resequencer model will not be suitable for multimedia traffics, then propose the SDRAM scheme to solve the problems, and finally compare our scheme with the resequencer model through simulation. Results show the mean queuing delays and mean inter-PDU delays of our scheme are not sensitive to mean PDU size while the mean queuing delays and mean inter-PDU delays of the resequencer scheme are very sensitive to mean PDU size. **Index Terms**—ATM, multicast, resequencer, round robin, shared tree.

II. SYSTEM ARCHITECTURE

Our architecture is based on a shared tree as shown in Figure 1 and we will focus on the access control of the tree. For a complete multipoint-to-multipoint multicast, there are many other topics that need to be studied. Senders in Figure 1 use the common tree to transmit their cells and the node that merges traffics from different sources is called a synchronizer. The traffic from the common VC will be duplicated in the distribution point (DP). In a leaf node, the dispatcher is connected to its receivers by a number of VCs, i.e., one VC per receiver. Here we apply only one VP connection to bundle these VCs for the purpose of easy management.

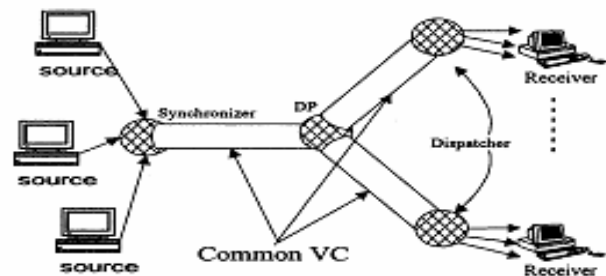


Figure 1. Simple system architecture

Figure 2 shows the inside of a RAM switch that consists of many synchronizer and dispatches. The synchronizer is used to fairly allocate common output resource among sources. The synchronizer multiplexes all source traffic into a SD channel, and the dispatcher de-multiplexes the incoming mixed cell stream from the SD channel and then dispenses them to the ports of out going SD channels.

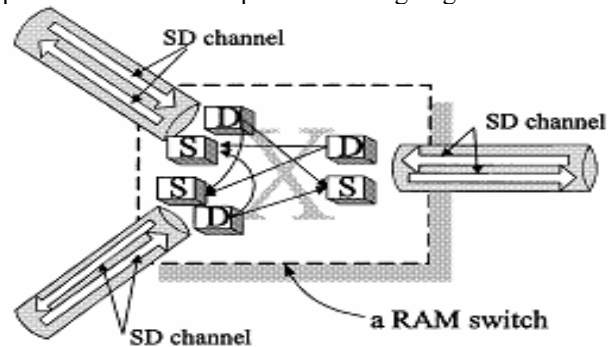


Figure 2. A tree-branch node

For the synchronizer, since its duty is to let all senders transmit cells in a common VC, it must allocate multiple

queues for buffering their individual traffic. That is, we reserve a receiving queue for each source. In addition, the synchronizer applies a Round-Robin (RR) scheme to fairly allocate the shared bandwidth among all sources. Every time a source can continuously send at most M cells (an M-cell burst). The M is an important variable. It can be a dynamic or static value. If a source has X cells to send, each time the source can send out min (X, M) cells, and then the control right of the resource is changed to next source by the RR order.

For the dispatcher, it uses the M value and a dispatching table to de-multiplex cells stream in the ingress common VC (SD channel) by the RR order, and then switches them into dedicated egress ports. The connections among dispatchers and end-users are full connectivity in the design. So, we don't have to modify the functions of machines at end systems. This is unlike the resequencer model that needs the end machines to participate in identifying incoming packets.

In order to transmit and receive cells, the synchronizer and the dispatcher must communicate with each other by control cells (in RM cell format). Control cells here should be as few as possible in order not to increase the CPU loads and not to occupy band width that can be used for transmitting general user cells. Five kinds of control cells are needed in the design. The description of them is in Table I. Here we simply just only use 3 bits in FSF field of a RM cell to recognize control cells of different types.

III. THE UPPER BOUND OF THE INTER-PDU DELAY AND ITS RELATIONSHIP WITH SOURCE RATE AND BUFFER SIZE

A cell has 53 bytes and each byte consists of 8 bits. Each source can send out at most M_k cells in the SD channel k each time. So, we can calculate the maximum time needed for each member to transmit once in the SD channel by the following equation:

$$(53 * 8 / R_{out}^k) * M^k N_m^k \quad (1)$$

Here, R is the number of multicast members that share the specific SD is the allocated bandwidth of a SD channel. By (1), we can derive the upper bound of the inter-PDU delay in the synchronizer the inter-PDU delay as the time between the beginning transmission of the last cell in the current PDU and the beginning transmission of the first cell in the following PDU, The upper bound of the inter-PDU delay can be written as the following:

$$(53 * 8 / R_{out}^k) * [M^k (N_m^k - 1) + 1] \quad (2)$$

IV. SIMULATION RESULTS

In our simulations, we study queuing delays, inter-PDU delays, and throughputs in the synchronizer.

Figure 3 is the comparisons of inter-PDU delay between our scheme and the resequencer scheme. In this figure, we find that, with the mean PDU size increasing, the gap between our scheme and the resequencer model is more and more obvious. This is due to our RR method for resource allocation, while in the resequencer model, one sender could send its PDUs only after another sender has finished transmission of an entire PDU. It will have serious influence

on performance when the PDU size increases. The mean inter-PDU delays for our scheme are between 0.018001sec and 0.019016 sec if we let the mean PDU size be 900 cells while the mean inter-PDU delays for the resequencer scheme are between 0.031354 sec and 0.034008 sec for the same mean PDU size. For the mean PDU size of 900 cells, the mean inter-PDU delay of our model is less than 60% of the mean inter-PDU delay in the resequencer model.

Additionally, we could also figure out a phenomenon that in the resequencer model, the sender who has the lowest output bandwidth would affect the performance of other senders who have higher output bandwidth. That is, it will increase the inter-PDU delay of the system. This is because the lowest band width sender would in general spend more time to receiver an EoP cell. Therefore, the resequencer model is only suited well for all senders with almost the same bandwidths while our scheme would not be constrained by the senders with lower bandwidths.

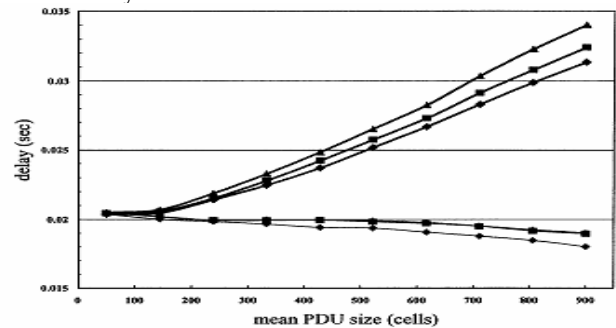


Figure 3. Mean inter-PDU delays

Figure 4 is the comparisons of queuing delays between these two approaches. We could also obtain better performance in our scheme than in the resequencer model. Again, our SDRAM scheme is not sensitive to the mean PDU size. As the PDU size increases, the distinction between our scheme and the resequencer approach becomes more and more apparent. The mean queuing delays for our scheme are between 0.002405 sec and 0.004736 sec if we let the mean PDU size be equal to 900 cells while the mean queuing delays for the resequencer scheme are between 0.036332 sec and 0.045476 sec for the same mean PDU size. The mean queuing delay in our scheme is less than 13% of the mean queuing delay in the resequencer model. This is because when we dump cells in a queue of the resequencer, other queues have to wait until a complete PDU has arrived. In this figure, we also find that the lowest-bandwidth sender would have little effect on the other senders in our scheme. This is due to the cyclic distribution of the transmission right among all senders whether an active sender has received an EoP cell or not.

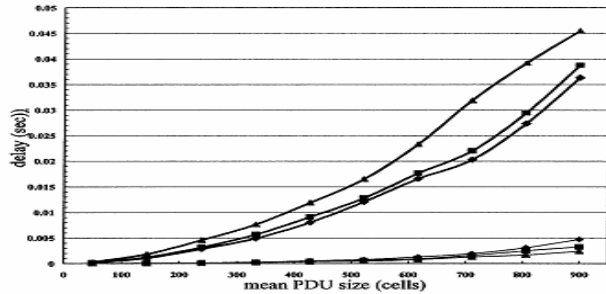


Figure 4. Mean queuing delays

Figure 5 shows the throughputs of the two schemes. The black and white boxes represent our scheme and the resequencer model, respectively. Here, we find that the two schemes have almost the same throughputs because of the rare appearance of the control cells. In addition, we observe that the throughputs would get saturated when the PDU size increases. The maximum throughput is limited by the speed of the switches in handling cells.

In the second simulation, the VBR traffic is generated from a generator of self-similar network traffic. The traffic loads are between 0.1 and 0.9, the packet length is 400 cells, and the simulation time for each traffic load is 20 seconds. The lines with the square symbols on them mean that all senders have 25Mbps of transmission rates and the lines with diamond symbols on them show that one sender has 20Mbps of transmission rates and other four senders have 25Mbps of transmission rates.

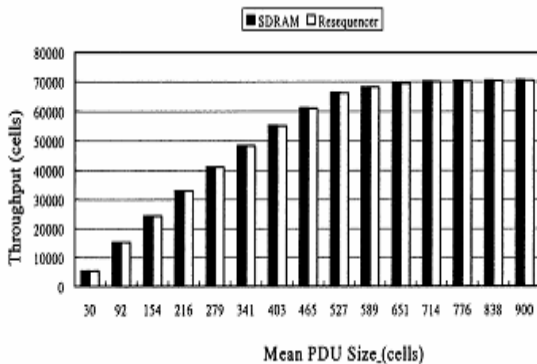


Figure 5. Throughput comparison

Figure 6 shows mean Inter-PDU delay versus various traffic loads. We can see Inter-PDU delay is decreased as the load increases for both the resequencer model and our scheme. This is because we have a fixed PDU size (400cells) and there are more possibilities of PDUs waiting for transmissions as the transmissions of current PDU in each source are complete. We can also see that with one slow source, Inter-PDU delays for both schemes are increased and our SDRAM scheme still has much smaller Inter-PDUs, which is similar to the situation in CBR traffic.

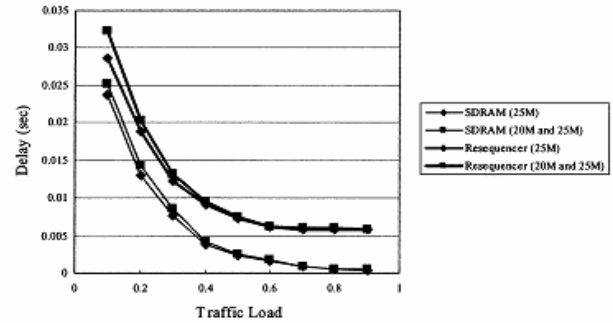


Figure 6. Mean inter-PDU delays (VBR traffic)

Figure 7 shows mean queuing delay versus various traffic loads. We can see that with VBR traffic, the benefit of the SDRAM scheme is even more obvious in terms of mean queuing delays. Since the queuing delays of our SDRAM scheme is very small compared to the resequencer model (Figure 7(a)), we show the details of our SDRAM scheme in Figure 7(b). Again, with one slow source, queuing delays for both schemes are increased.

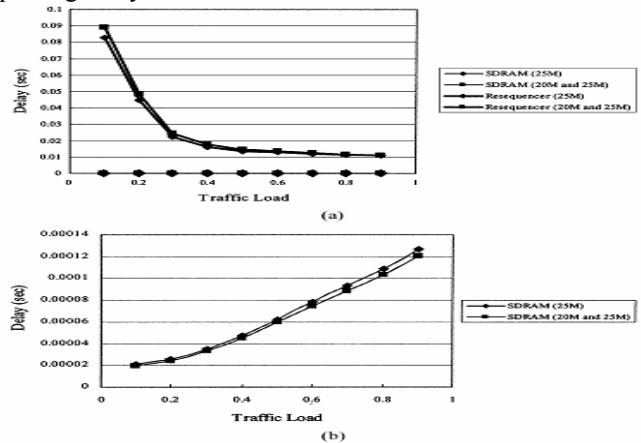


Figure 7. Mean queuing delays (VBR traffic) for both (a) the resequencer model and our SDRAM scheme and (b) the details of our SDRAM scheme

Figure 8 shows the throughputs of the two schemes. The black and white boxes represent our scheme and the resequencer model, respectively. Here, we find again that the two schemes have almost the same throughputs because of the rare appearance of the control cells. In addition, we observe that the throughputs would get saturated when the traffic load increases. The maximum throughput is also limited by the speed of the switches in handling cells.

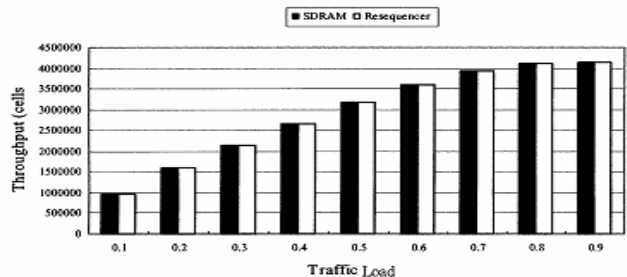


Figure 8. Throughput comparison (VBR traffic)

V. CONCLUSIONS

In this paper, we have developed a multicasting scheme, which supports a shared tree in ATM networks and can be implemented in ATM layer. Our mechanism has very low inter-PDU delays and very low queuing delays compared to the resequencer model. In addition, our scheme allocates the shared bandwidth fairly among source traffic and has almost the same throughputs as the resequencer model. Simulation results also show that the scheme is not sensitive to mean PDU size and effectively reduces inter-PDU delays and queuing delays. Hence, it is very suitable for multimedia applications.

Although a single shared tree has many advantages, e.g., resources saving and scalability, some congestion will occur in some links as the number of senders grows. Hence, it needs flow control mechanisms to avoid congestion. In addition, in order to use buffer efficiently, it might need a dynamic buffer allocation scheme. A priority of bandwidth assignment for senders' traffic in the same channel can also be applied to our approach.

REFERENCES

- [1] Jia Z F, Varaiya P. Heuristic methods for delay constrained least cost routing using k-shortest -paths[C]. Proc. of IEEE INFOCOM '01. 2001.
- [2] Waxman B M. Routing of multipoint connections [J] IEEE Journal on Selected Areas in Communications, 1988, 6(9):1617. 1622.
- [3] Bellman R E. Dynamic Programming[M]. NJ : Princeton University, 1997.