Performance Study of Real-Time Scheduling Techniques under Multicasting Traffic in an ATM Multiplexer

Khalid H. Sheta and Mukesh Singhal
Department of Computer and Information Science
The Ohio State University
Columbus, OH 43210
email: sheta@cis.ohio-state.edu, singhal@cis.ohio-state.edu
Phone: 614-292-5839
Fax: 614-292-2991

Abstract

Multimedia applications in broadband ISDN present various types and classes of traffic streams, i.e., connections such as audio, video, and file transfers. Output multiplexers at an ATM switching node are responsible for scheduling cells belonging to these traffic connections. A cell loss in a traffic stream due to a cell missing its deadline results in the loss the entire packet that should have been delivered to the destination node. Multicasting is an important application in broadband networks. A cell loss of a multicast stream at an ATM multiplexer not only affects one node, but also it affects a group of nodes in the multicasting group. This packet loss leads to inconsistency in the information received among different members of the multicast group. Moreover, if the lost packet can be retransmitted by the root node of the multicast tree, the retransmission will result in an excessive load to the network. In this paper we address the problem of scheduling connections that contain real-time multicasting/non-multicasting and non-real-time traffic. Traffic on real-time connections has a deadline. Each cell has a deadline associated with it (in terms of cell transmission time slots) which the cell has to be scheduled at the multiplexer before it expires. We propose a new scheduling technique that makes use of the information available at multicast connection setup time to give appropriate priority to multicast traffic. In particular we use the multicast tree size (number of nodes) and the multicast subtree sizes at each multiplexer. We show how this information coupled with cell arrival time, can be used to proportionally weigh the actual cell deadline of multicast cells to give them priority over other classes of real-time non-multicast and non-real-time cells. We assume an On-Off traffic source model for real-time connections containing two types of deadlines, tight and relaxed. We compare, through extensive simulation, our technique with two well known real-time scheduling techniques. We show how the new technique outperforms the other techniques by improving cell loss for multicast connections while attempting to maintain the same cell loss or better for non-multicasting connections over a wide range of parameter values. We also show that the new technique does not increase the delay of non-real-time connections compared to the other scheduling techniques. Finally, we discuss how the new technique can be incorporated into an ATM switching node.

Key words: Multicast, Real-Time, Scheduling, ATM Switches.

1 Introduction

Multimedia applications in broadband ISDN present various types and classes of traffic streams, i.e., connections such as audio, video, and file transfers. ATM is the underlying transport protocol for broadband ISDN. Information packets transmitted between two communicating hosts are partitioned into smaller fixed size packets called cells. An ATM switching node routes cells from its input port to one or more its output ports (multiplexers). Cells routed to the output multiplexers represent different traffic streams (connections). Output multiplexers at an ATM switching node are responsible for scheduling cells belonging to these traffic connections. Cell scheduling at the multiplexer decides which cell of what connection is to be transmitted next. The scheduling decision tries to satisfy the connection requirements, such as deadlines.

A cell that cannot be scheduled (serviced) before its deadline expires is lost. A cell loss in a traffic stream due to a cell missing its deadline results in the loss of the entire packet that should have been delivered to the destination node. Multicasting connections represent an important class of traffic streams. A cell loss of a multicast stream at an ATM multiplexer not only affects one node, but also it affects a group of nodes in the multicasting group. A packet loss in a multicast connection leads to an inconsistency in the information received among different members of the multicast group. Moreover, if the lost packet needs to be retransmitted by the root node of the multicast tree, the retransmission will result in excessive load to the network.

Most of the scheduling techniques in the literature [3, 4, 6, 7, 9, 10, 11, 13, 20] do not take into account the class of multicast traffic at an ATM multiplexer. In addition, the scheduling techniques are designed to make the decision at an ATM multiplexer independent of other multiplexers in the ATM switching node.

In this paper we consider traffic streams (connections) that are real-time multicasting/non-multicasting and non-real-time. We address the problem of scheduling cells at an output ATM multiplexer that receives these traffic streams. Traffic on a real-time connection has a deadline. Each cell has a deadline associated with it in terms of cell transmission time slots. A cell has to be transmitted by the multiplexer before the cell deadline expires. We propose a new scheduling technique that makes use if the information available at multicast connection setup time to give appropriate priority to multicast traffic.

In particular we use the multicast tree size (number of nodes) and the multicast subtree size rooted at each multiplexer. We show how this information can be used along with the cell arrival time to weigh the actual cell deadline of multicast cells to give them priority over other classes of real-time and non-real time cells. We also show how our proposed scheduling technique correlates the decision taken at each output multiplexer to give the multicast connection an appropriate priority in the multiplexers it is spanning.

We assume an On-Off traffic source model for real-time connections containing two types of deadlines, tight and relaxed. We use extensive simulations to study and compare our technique with two well known real-time scheduling techniques. We show how the proposed technique outperforms the other techniques by improving cell loss for multicast connections while attempting to maintain the same cell loss or better for non-multicasting connections over a wide range of parameters values. A reduction in multicast traffic cell loss is a very desirable feature since it reduces the overhead of retransmission and minimizes the inconsistency of the information received by the multicasting group members. We also show that the proposed technique does not increase the delay of non-real time connections compared to the other scheduling techniques. We also point out how the proposed technique can be incorporated into and ATM switching node.

The rest of this paper is organized as follows: Section 2 presents a description ofexisting scheduling techniques at an ATM multiplexer. Section 3 describes the switch and traffic model we used in our performance study and also describes the details of the proposed scheduling technique. Section 4 presents the results and discussion of the performance study. Section 5 points out the implementation issues of the new proposed technique. Finally, Section 6 presents the conclusions.

2 Related Work

Scheduling of real-time traffic in an ATM multiplexer can be classified into the following categories two [6, 10, 20, 7]:

1. Throughput based
2. Delay based

1.Throughput based: In this category, the capacity of the output link of a multiplexer is partitioned among different types of traffic. Each incoming traffic type (connection) to a multiplexer declares its bandwidth requirements at connection setup time. The scheduling policy, that decides which cell from which stream is to be transmitted on the output link, then tries to satisfy the connection requirement. It either guarantees that the connection meets its requirements, or applies a best effort strategy.

Two such scheduling policies are Weighted Fair Queueing (WFQ) [10] and Stop-and-Go queueing [8]. WFQ can be viewed as the realization of round robin service policy. Each connection has a dedicated queue and cells are serviced from each queue in cyclic order. The service is modulated by applying weights to determine the rate at which the connection can be served during the cycle. An advantage of this policy is that it adapts to the rate at which the connection is sending at and it enforces the required bandwidth by each connection, thus protecting each connection from the others. The disadvantages are that the policy uses separate queues which could be a wastage of buffer space and it is complex to implement. Also, there is no guarantee of satisfying individual connection deadlines since the scheduling policy is purely a rate based strategy.

In the Stop-and-Go policy, the time axis is divided into frames of constant length such that each frame is a multiple of cell lengths. The input frames are mapped into the output frames according to a delay criteria. A frame of cells is not allowed to depart until the beginning of time of the next frame. The advantage of this policy is similar to the WFQ policy in terms of guaranteeing the bandwidth required per connection. It also limits the traffic pattern distortion and help reshape the traffic throughout the network. However, since the output link can be idle when there are frames waiting at the input to be transmitted, the policy is non-work conserving and is subject to wastage of bandwidth. In addition, the disadvantages of WFQ mentioned above apply here as well.

2. Delay Based: In this category, scheduling of cells at a multiplexer is based either on *Local delay* or *end-to-end delay* criteria at the multiplexer.

Local delay: Scheduling of cells at a multiplexer is based on local delay information that is used to give priorities to different traffic classes [14]. Examples of such policies are FCFS, Earliest Due Date First (EDD), and FIFO+ [6].

FCFS is the simplest policy since cells are serviced in the order of their arrival regardless of their class. It has the advantages of giving the same low delay to all traffic classes and it efficiently uses the buffer space. Its disadvantages are a high cell loss rate for real-time traffic and it has no individual control over different real-time traffic classes.

In the EDD policy, the cell with the smallest deadline is always serviced first. This policy minimizes the delay and cell loss for those classes of traffic having small deadlines. On the other hand, non real-time traffic and real-time traffics with larger deadlines suffer higher cell loss and delays. Some variations in the literature [17, 11, 14] add static thresholds to the policy. These thresholds are assigned to traffic classes having higher deadlines. When the number of cells of a certain class exceeds a threshold value, cells of that class are serviced until they fall below the threshold value. The main advantage of EDD is its simplicity. The disadvantage is that the scheduling decision uses only the deadline values and the number of cells of a certain class, does not give appropriate priority of aging cells.

In FIFO+, the average cell delay is estimated at the multiplexer for all cells of a given class. A priority value is assigned to each cell of a traffic class. Each cell is timestamped when it joins the queue. When the cell departs the multiplexer, the current time and the cell arrival time are used to compute the delay experienced by the cell. This delay is used to refine the estimated cell delay to a new value at the multiplexer for all cells of each class. The new estimated delay value is used to raise or lower the old priority assigned to each traffic class. The advantage of this policy is the adaptivity to traffic changes and fluctuation. The disadvantages are the vulnerability of the policy to being unstable in case of transient traffic fluctuation and the complexity of implementing the policy in a high speed switching environment.

End-to-end delay: Scheduling decision made at the multiplexer to service the next cell takes into account some of the end-to-end requirements of each traffic class. Examples of such a category are Transit Priority (TP) and Hop Laxity (HL) [6].

In TP, cells that originate at the current multiplexer give way to those cells passing through (called transient) the multiplexing point. The policy tries to increase the probability of aged cells in the network get to their final destination as opossed to those cells just entering the network. This policy is simple to implement but has limited performance results since it doesn't distinguish between various traffic classes requirements (deadlines).

In HL, a laxity parameter is defined as a measure of the time remaining until the cell end-to-end deadline expires in the network. The laxity parameter is updated at each hop which the cell traverses. The per-hop-laxity is the individual cell laxity weighted by the number of hops the cell has to travel until it reaches its final destination. The scheduling decision in HL selects the cell with the minimum per-hop-laxity. The advantage of this policy is that it compensates those cells that have experienced larger delays at earlier nodes by raising their priorities. The disadvantages is that the scheduling decision is purely based on the laxity values and does not take into account other traffic classes that may need priorities.

3 The Scheduling Model

In Section 3.1, we present a description of the ATM multiplexers in an ATM switching node. We also describe the characteristics of the traffic model used in the performance study. In Section 3.2, we present the proposed scheduling technique in detail.

3.1 System Description

We assume a nonblocking output buffered ATM switch with N multiplexers at its output ports as shown in Figure 1. We assume that time is slotted with each slot equal to one cell transmission time over an output link. Hence, for a link of capacity C_l bps, a cell of length 53 bytes takes $53 \times 8 \times 1000/C_l$ ms to be transmitted. The service time at an output multiplexer is equal to one cell transmission time.

Each of the N input port receives its input from a traffic stream consisting of a group of connections. A connection is a stream of cells arriving at an input port and routed to one or more multiplexers. Each cell in a connection has a deadline expressed in time slots. A cell arriving at an input port j belonging to connection i at time a_i with an associated deadline d_i must depart the output multiplexer at time $t \leq a_i + d_i$.

We define three types of connections:

1. Non Real-Time Connection.
2. Real-Time Non-Multicast connection.
3. Real-Time Multicast connection

1. Non Real-Time Connection: Each arriving cell in the stream is routed to exactly one output multiplexer. The cell stream is assumed to follow a Bernoulli type arrival pattern

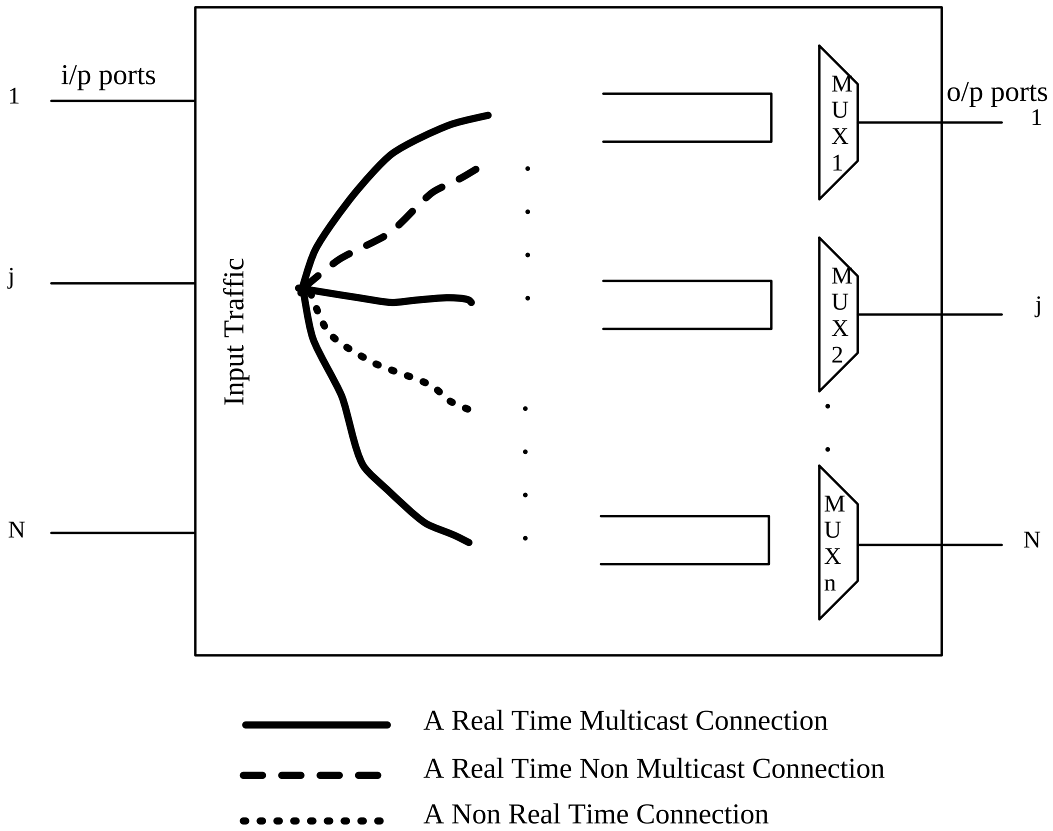

Figure 1: An ATM Switch Model

such that the probability of a cell arriving in any given slot is p. A non real-time connection has an infinite deadline.

2. Real-Time NonMulticast Connection: Each arriving cell in the stream has a n associated deadline and is routed to exactly one output multiplexer. The cell stream is assumed to follow a classical On-Off traffic source [1, 19] as shown in Figure 2. The cell stream for the connection consists of arrivals at fixed intervals T during an active period having a mean a^{-1}. The number of cells in an active period is geometrically distributed with a mean of $(aT)^{-1}$. The idle period is exponentially distributed with a mean b^{-1}. We approximate the time in an idle period by rounding to the nearest time divisible by the cell transmission time $53 \times 8/C_l$.

3. Real-Time Multicast Connection: Each arriving cell in the stream has a n associated deadline and is replicated to exactly dest_i multiplexers, $1 \leq \text{dest}_i \leq N$. We assume that the cost of replicating a cell is negligible. The cell stream for such a connection has the same On-Off traffic source model as in the real-time nonmulticasting stream described above.

We also assume that a real-time multicast connection i has a subtree of size T_{ij} rooted at multiplexer j. Hence, a cell belonging to real-time multicast connection i at multiplexer j is replicated, after departing the multiplexer to all nodes within subtree T_{ij}. The total tree

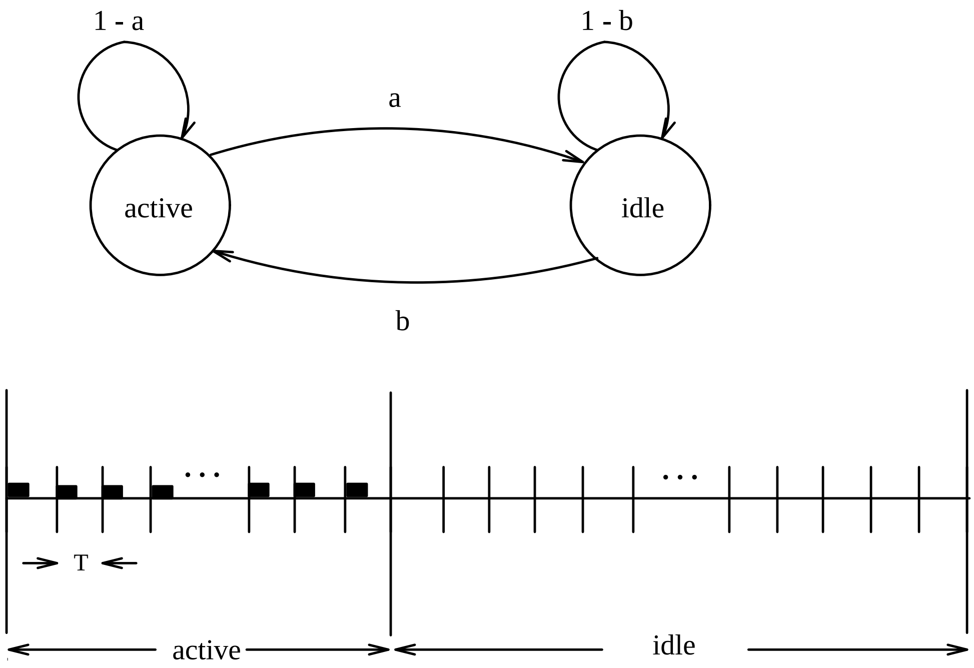

Figure 2: Cell Stream Model

size of connection i is thus defined as:

$$T_i = \sum_{j=1}^{N} T_{ij}.V_{ij} \tag{1}$$

where

$$\begin{aligned} V_{ij} &= 1 \quad if\ connection\ i\ spans\ multiplexer\ j \\ &= 0 \quad otherwise. \end{aligned}$$

Figure 3 shows two multicast connections i and k where connection i spans multiplexers $1, 2, j, N$ and connection k spans multiplexers $1, j, N$.

We assume real-time connections, to consist of two classes of traffic having deadlines d_1 and d_2 such that $d_1 << d_2$. Connections having deadline d_1 are called Tight DeadLine (T.D.) connections and connections having deadline d_2 are called Relaxed Deadline (R.D.) connections. We assume that there are N_c connections feeding multiplexer j where $N_c = N_{nr} + N_m + N_{nm}$ and N_{nr} is the number of non real-time connections, N_m is the number of real-time multicast connections, and N_{nm} is the number of real-time nonmulticast connections. We also assume that $N_m = N_{mt} + N_{mr}$ and $N_{nm} = N_{nmt} + N_{nmr}$ where N_{mt} and N_{mr} are the number of tight deadline and relaxed deadline multicast connections respectively. N_{nmt} and N_{nmr} are the number of tight deadline and relaxed deadline nonmulticast connections.

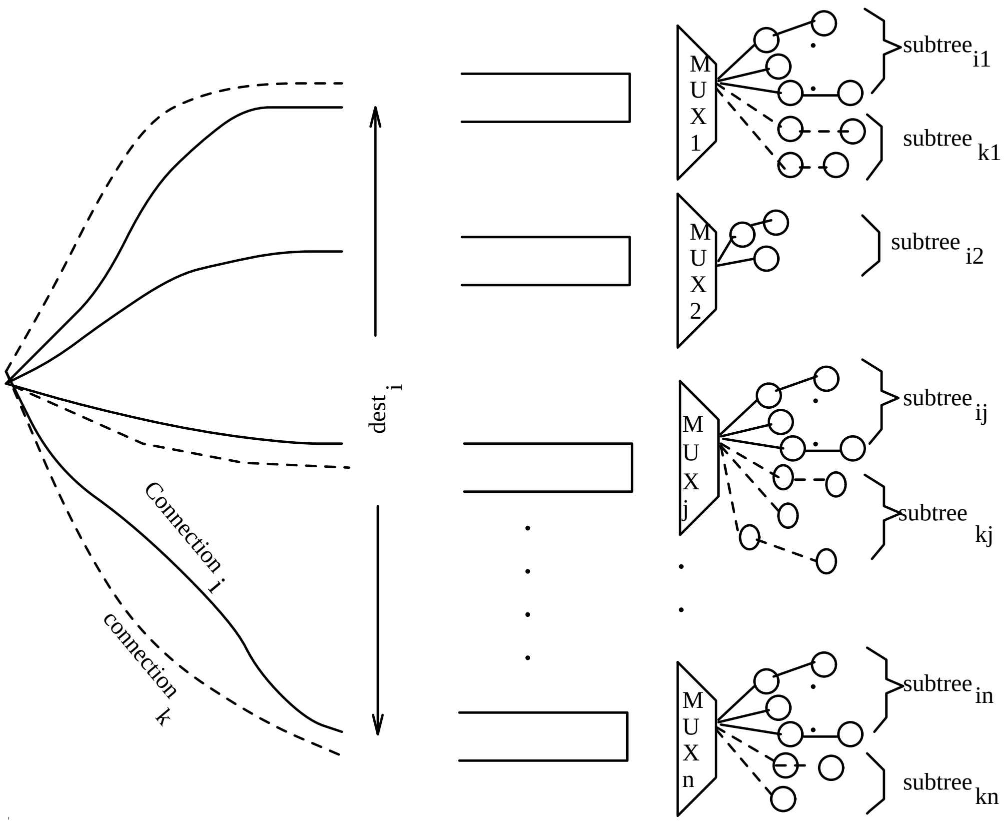

Figure 3: Multicast Connections Model

The input load at multiplexer j is thus defined as $\lambda = \lambda_1 + \lambda_2 + \lambda_3$ such that λ_1, λ_2 and λ_3 are the arrival rates of all non real-time, real-time multicast and real-time non multicast connections. λ_1, λ_2 and λ_3 are defined as:

$$\lambda_1 = N_{nr} * p \tag{2}$$

Assuming a link capacity of C_l at multiplexer j, the cell transmission time $ct = 53 \times 8/C_l$ and assuming the number of cells in an active period to be geometrically distributed with mean $(aT)^{-1}$, then:

$$\lambda_2 = N_m \frac{a^{-1}}{a^{-1} + b^{-1}} \frac{T}{ct} \tag{3}$$

similarly,

$$\lambda_3 = N_{nm} \frac{a^{-1}}{a^{-1} + b^{-1}} \frac{T}{ct} \tag{4}$$

The service rate per cell at multiplexer j is fixed to one time slot (i.e. the cell transmission time). Therefore, the overall system load (utilization) ρ is equal to the arrival rate λ divided by the service rate

$$\rho = \lambda \tag{5}$$

3.2 Proposed Scheduling Technique

In this Section, we propose a new scheduling technique for an ATM multiplexer. We compare its performance with two existing scheduling techniques, namely FCFS* and EDD (Earliest Deadline Due). In the FCFS* technique, cells join the tail of the multiplexer buffer queue in the order of their arrival. At each time slot, the deadline d_i associated with the cell at the head of the queue is examined with respect to the cell arrival time a_i and the current clock time t. If $t - a_i < d_i$ (i.e. the cell has not missed its deadline) the cell is serviced, otherwise the cell is dropped. FCFS* differs from plain FCFS in not allowing a cell that has missed its deadline to be serviced. This makes the comparison of FCFS* (as the reference line) to the other techniques more fair.

In the EDD technique, like FCFS*, cells join the tail of the multiplexer buffer queue in the order of their arrival. At each time slot, the deadline field associated with all cells of all connection types is scanned for the minimum value. The cell with the minimum deadline value (the earliest deadline due) is serviced (as in FCFS* above) only if $t - a_i < d_i$, where t is the current clock time, a_i is the cell arrival time and d_i is the cell deadline.

Both FCFS* and EDD described above do not distinguish between different connection types when making the scheduling decision at the multiplexer. EDD has been shown to be an optimal policy for one class of traffic [20]. It is no longer optimal if packets of one class are to be given priority over other packets. In addition, the scheduling decision at an individual multiplexer is taken independently from other multiplexers at the ATM switching node.

Multicasting connections are significant since a cell loss due to a cell missing its deadline at a multiplexer, results in the loss of the entire packet that should have been delivered to the destination node. Consequently, all nodes in the subtree rooted at the multiplexer where the loss happened will lose the packet. This packet loss results in an inconsistency of the information received among different members of the multicasting group. Moreover, if the lost packet need to be retransmitted by the higher application level in the multicast root node, these retransmissions will result in excessive load on the network leading to a ripple effect of longer delays and packet losses. The larger the subtree where the packet loss happens, the more damaging the retransmission effect.

The above shortcomings motivated the development of a new technique. We call this new technique MC-EDD (Multicast Earliest Deadline Due). In this technique, we keep the simplicity of the EDD technique and enhance it by using additional pieces of information that are available when the connection is setup in addition to the cell arrival time. In particular, we use:

1. The multicast tree size for connection i, T_i, as defined in Equation 1.

2. The subtree size T_{ij} for connection i at multiplexer j as define in Section 3.1.

We add an additional field to the cell information header (which usually contains information such as cell deadline, arrival time, etc.) in the switch. We call this field *cell attribute*. A *cell attribute* is computed for each cell m belonging to connection i at multiplexer j as follows:

$$cell\ attribute = w_i \times d_i + a_{im} \tag{6}$$

$$\begin{aligned} w_i &= 1 - \frac{T_{ij}}{T_i} \qquad connection\ i\ is\ real-time\ multicast \\ &= 1 \qquad\qquad Otherwise \end{aligned} \tag{7}$$

where, a_{im} is the arrival time of cell m belonging to connection i having deadline d_i at multiplexer j. The weight w_i in Equation 7 is computed only once for connection i at connection setup time.

MC-EDD scheduling algorithm works as follows: at each time slot, MC-EDD picks the cell having the minimum value of *cell attribute* from the buffer queue at multiplexer j. The cell is serviced only if $t - a_{im} < d_i$, where t is the current clock time and a_{im} is the arrival time of cell m belonging to connection i.

The *cell attribute* is composed of two terms as per the definition in Equation 6. The first term, $w_i \times d_i$, is the weighed deadline of each cell by a factor w_i as defined in Equation 7. The weighting factor w_i is less than one for multicast connections, and is equal to one for connections other than multicast. This weighting factor reduces the value of the cell deadline for multicast connections relative to other connections. Since the algorithm looks for the minimum *cell attribute*, cells belonging to multicast connections are selected first. This selection gives preference or higher priority to multicast connections over other connection types. [1]

The weighting factor w_i in Equation 7 is chosen such that it if two multicast connections i and k having the same multicast tree size, i.e., $T_i = T_k$, the connection with the larger subtree size at multiplexer j will produce a smaller weighting factor giving that connection more priority. Hence, cells destined to larger subtrees of a multicast tree get higher priority. This higher priority increases the likelihood of a larger number of the multicast group members receiving the same information. Consequently, inconsistency of the information received by the multicast group members is minimized.

In addition, the weighting factor correlates the decision taken at each multiplexer spanned by a multicast connection i. This correlation is a result of the weighting factor creating a priority proportional to the size of the subtree at the multiplexer relative to the total tree

[1] The value of *cell attribute* for non-real-time connections is impelmented as a very large value (larger than the largest possible value of *cell attribute*.

size. For example, if connection i spans multiplexers j and k such that $T_{ij} < T_{ik}$, the connection will get a smaller weighting factor, hence higher priority, at multiplexer k than at multiplexer j. The correlation enables the scheduling decision at each multiplexer give the multicast connection appropriate priority it needs in coordination with other multiplexers that are spanned by the same connection.

The second part in Equation 6 is the cell arrival time a_{im} for the connection under consideration. This part is necessary since the first part, $w_i \times d_i$, by itself gives higher priority only to those multicasting connections having smaller w_i values. This higher priority results in multicast cells that has just arrived to get serviced while other cell types have aged in the queue and are about to be lost. By adding the cell arrival time to the cell weighted deadline, the priority of the cell gets higher as the cell ages in the queue. Therefore, the adding of the cell arrival time helps compensate cells having larger weighting factor and gives them a fair share of scheduling.

4 Performance Results and Discussion

An extensive simulation study was done using CSIM simulation package to simulate an ATM multiplexer as described in Section 3.1. The simulation study is used to compare the proposed technique MC-EDD with EDD and FCFS* techniques as described in section 3.2.

4.1 Assumptions

We assume that the ATM multiplexer has a T1 output link of 1.544 Mbps giving a cell transmission of one time slot of 0.275 ms. We simulate each real-time connection by a classical On-Off source model [19], as shown in Figure 2, where the mean active period a^{-1} and the mean idle period b^{-1} (as defined in section 3.1) are 352 ms and 650 ms respectively. The cell separation time T is 16 ms. Hence, as defined in Equations 3 and 4 λ_2 and λ_3 become:

$$\begin{aligned}
\lambda_2 &= N_m \cdot \frac{352}{352+650} \cdot \frac{1}{58} \\
&= 0.0061 \times N_m, \qquad\qquad \text{and} \\
\lambda_3 &= N_{nm} \cdot \frac{352}{352+650} \cdot \frac{1}{58} \\
&= 0.0061 \times N_{nm}
\end{aligned}$$

We simulate a non-real-time connection by a Bernoulli distribution with parameter $p = 0.03$, giving $\lambda_1 = 0.03 * N_r$ as defined in Equation 2. We vary the system load ρ (as defined in Equation 5) by varying the number of connections N_m, N_{nm}, and N_r.

To select reasonable values for deadlines of the two classes of real-time traffic we are simulating, Tight Deadline (T.D.) and Relaxed Deadline (R.D.), we consider the NeVot application in [5]. In this application a packet consisting of 3 cells has a permissible one-way delay budget of $\simeq$ 120 ms between two communicating nodes. Assuming a hop count of $\simeq$ 10 nodes and a T1 link speed, the local deadline per cell at an intermediate switching node is $\simeq$ 15 time slots. For a tighter deadline value, e.g., farther communicating nodes, a value of 10 time slots is reasonable.

The performance indices we have used in the study are the cell loss (ratio of the number of cells lost due to missing the deadline to the number of cells generated of the cell class under consideration) and the cell delay (average number of time slots a cell spends from the time it joins the buffer queue till it departs the multiplexer). In particular, we considered the cell loss and delay for: all classes, real-time multicast, real-time multicast tight deadline, real-time multicast relaxed deadline, real-time non-multicast, real-time non-multicast tight deadline, real-time non-multicast relaxed deadline, and non-real-time.

We refer to the multicast class in the results tables as MCAST, the non-multicast class as NMCAST, and the non-real-time class as NRT.

In the performance study, we varied the number of connections to obtain a system load ρ of 0.75 through 0.95. For loads lower than these, cell loss become insignificant and all techniques exhibit basically the same performance. To study the performance of the techniques we are comparing under a certain system load, we fixed the number of non-real-time connections and varied the percentage of multicast and non-multicast connections relative to the total number of real-time connections, i.e., $N_m/(N_m + N_{nm})$ and $N_{nm}/(N_m + N_{nm})$, respectively. We used the percentage combinations (25%,75%), (50%,50%), and (75%,25%) for (multicast, non-multicast), respectively, to study the effect when a traffic class has a small, equal or large effect on the overall real-time traffic class.

For the subclasses of MCAST (T.D., R.D.) and NMCAST (T.D., R.D.), we used percentages of (25%, 75%) in all different combinations, e.g., MCAST (T.D., R.D.) = (75%, 25%), NMCAST (T.D., R.D.) = (75% , 25%), and MCAST (T.D., R.D.) = (25%, 75%), NMCAST (T.D., R.D) = (25%, 75%). These different combinations show the effect when one subclass, e.g., T.D. has a larger effect over the other, e.g., R.D. We report the results for $\rho = 0.9$ and $\rho = 0.8$ as representative of the load range. Tables 1 through 6 report the results of MCAST (T.D., R.D.) = (75%, 25%), NMCAST (T.D., R.D.) = (75%, 25%) and MCAST (T.D., R.D.) = (25%, 75%) , NMCAST (T.D., R.D.) = (25%, 75%) All other combinations are reported in Tables 7 through 12 in the appendix. Each table contains the cell loss and delay in two consecutive rows for each class.

We have studied two MC-EDD variations, MC-EDD1 and MC-EDD2. In MC-EDD1 we used a weighting factor w_i of 0.6 for half the connections in either the MCAST(T.D.) or MCAST(R.D.) and 0.9 for the other half. These values show the effect when we have multicast connections at the multiplexer with subtree sizes as small as 10% of the total multicast tree size ($w_i = 0.9$) and intermediate subtree sizes of 40% of the total multicast tree size ($w_i = 0.6$). In MC-EDD2 we did the same as in MC-EDD1 except that we used values of w_i of 0.3 and 0.9. The lower value of w_i, 0.3, shows the effect when we have multicast connections at the multiplexer with subtree sizes as large as 70% of the total multicast tree size.

The results reported here and in the appendix are for 10 connections of NRT traffic class, i.e., 30% of the total system load ρ. The results for a number of non-real-time connections smaller than and larger than 10, representing NRT system loads smaller than and larger than 30%, exhibit the same performance.

All simulation experiments were run several times with different seeds to obtain 95% confidence interval. The interval was found to be very narrow for all experiments.

4.2 Results and Discussion

In Table 1, when the MCAST class has a smaller percentage (25%) of real-time traffic relative to the percentage of NMCAST (75%), and the subclass T.D. has a higher percentage (75%) over R.D.(25%), MC-EDD1 has substantially reduced cell loss for the MCAST class compared to EDD and FCFS* for both $\rho = 0.9$, 0.8. This reduction is due to the priority given to MCAST cells by weighting their deadlines by the factor w_i. At the same time, MC-EDD1 has reduced the NMCAST cell loss, and consequently the overall cell loss, compared to EDD and FCFS*. This reduction is due to the second part in Equation 6 of the MC-EDD technique which incorporate the cell arrival time into the scheduling decision at the multiplexer giving a fair share to aging cells. We observe that the overall delay in MC-EDD1 has increased slightly compared to EDD. This increase is due to the fact that more cells are being saved under MC-EDD than in EDD causing more cells to stay in the queue leading to delay build up. For the NRT class also, the delay has slightly been increased in MC-EDD1 over EDD for the same reason of saving more cells. Note that FCFS* has got the best overall delay and NRT delay compared to EDD, MC-EDD, but this is due to the fact that FCFS* does not distinguish between any traffic class leading to big cell loss that reduces the overall delay.

In MC-EDD2 at $\rho = 0.9$, as the weighting factor w_i decreases (meaning bigger subtree sizes), MCAST cell loss is reduced even over MC-EDD1. NMCAST cell loss is higher in MC-EDD2 compared to MC-EDD1. The reduction in MCAST cell loss and the increase NMCAST cell loss are attributed to the fact that higher priority is given to MCAST because of the smaller w_i, or equivalently larger subtree sizes, which shifts the reduction in cell loss from NMCAST more towards MCAST for this system load and traffic mix. However, MC-EDD2 still improves the cell loss for NMCAST compared to EDD. We also note that the overall cell loss in MC-EDD2 has slightly increased compared to MC-EDD1 (due to the increase in NMCAST cell loss) but is still less than that of EDD.

For lower load $\rho = 0.8$, both MC-EDD1 and MC-EDD2 exhibit the same performance as with $\rho = 0.9$ except that the amount of increase in NMCAST cell loss in MC-EDD2 relative to MC-EDD1 is small due to the lighter load. All of the above analysis apply for Table 7 in the appendix, where MCAST and NMCAST has the same relative percentage of (25%) and

(75%), respectively, but with NMCAST (T.D. , R.D.) having a percentage of (25% , 75%).

In Table 2, at $\rho = 0.9$, the percentage of the T.D. subclass is lower (25%) than the R.D. subclass(75%) for both MCAST and NMCAST, that is, the proportion of relaxed deadline cells is large. MC-EDD1 still achieves a substantial reduction in MCAST cell loss over EDD. But, since the proportion of NMCAST R.D. has increased to (75%), weighted deadlines of MCAST T.D. and R.D. cells make these cells always preferred by MC-EDD over NMCAST since the majority of NMCAST are of the R.D. type. This preference leads to cell more losses of the NMCAST class in MC-EDD1 than in EDD. This effect can be seen even more in MC-EDD2 which has a smaller weighting factor. However, the reduction of MCAST cell loss by MC-EDD2 more than offsets the increase in NMCAST leading to improvement in the overall cell loss in MC-EDD1, MC-EDD2 over EDD. This reduction of MCAST cell loss is desirable since a cell loss of an MCAST class is more devastating than in the case of NMCAST. Note that when the majority of the NMCAST class is of T.D. type (Table 8 in the appendix), the NMCAST cell loss is decreased for MC-EDD1 and MC-EDD2 compared to EDD.

For lower load $\rho = 0.8$, both MC-EDD1 and MC-EDD2 have managed to keep the improvement in MCAST cell loss. To the contrary of the results at $\rho = 0.9$, both MC-EDD1 and MC-EDD2 decreased the NMCAST cell loss compared to EDD. This decrease is due to the fact that at lower loads and lower percentage of MCAST traffic, the cell arrival time part of the *cell attribute* in MC-EDD has more chance of saving aging cells of NMCAST. In addition, the preference of MCAST weighted deadline cells is not prevalent at this lighter level of load. The same behavior is seen in Table 8 in the appendix. We will see that this behavior does not occur later on when the percentage of MCAST traffic increases.

It should be noted that typically, the percentage of multicast connections relative to non-multicast connections is lower. Hence to the credit of MC-EDD, it manages to improve cell loss for both MCAST and NMCAST at lower loads.

Table 3 depicts the results when MCAST and NMCAST have equal percentage of real-time traffic, and the subclass T.D. has higher percentage (75%) over R.D.(25%). In this case, both MC-EDD1 and MC-EDD2 substantially reduce MCAST cell loss compared to EDD and FCFS* for both $\rho = 0.9$ and $\rho = 0.8$. This reduction is again due to the priority given to MCAST cells by weighting their deadlines by the factor w_i. The reduction caused by MC-EDD2 is more than that caused by MC-EDD1 due to the lower w_i in MC-EDD2. Both MC-EDD1 and MC-EDD2 reduce the NMCAST cell loss compared to EDD for both $\rho = 0.9$ and $\rho = 0.8$ due to the fair share of scheduling given to aging cells by the MC-EDD technique. Note that the reduction of NMCAST cell loss in MC-EDD1 compared to EDD is larger than the relative reduction of MC-EDD2 compared to EDD. This is due to the fact that priority is shifted more towards MCAST than NMCAST. We also note that MC-EDD1 and MC-EDD2 have reduced the overall cell loss compared to EDD and FCFS*. This is because both MCAST and NMCAST cell loss have been reduced as mentioned above.

We observe that both the overall delay and the NRT delay in MC-EDD1 and MC-EDD2 have slightly increased compared to EDD. This is because MC-EDD is saving more cells and hence slightly increase the delay. Again the delay for FCFS* is low due to the indiscriminate nature of FCFS* scheduling among different classes of traffic leading to big cell loss that reduces the delay.

Note that the analysis given above for Table 3 applies for Table 5 and Tables 9 and 11 in the appendix.

In Table 4 when the percentage of the T.D. subclass is lower (25%) than the R.D. subclass for both MCAST and NMCAST, i.e., the proportion of relaxed deadline cells is large, both MC-EDD1 and MC-EDD2 reduce the MCAST cell loss compared to EDD for both $\rho = 0.9$ and $\rho = 0.8$. The analysis given for table 2 for $\rho = 0.9$ applies here for Table 4 and also applies for Table 10 in the appendix.

Finally, in table 6 both MC-EDD1 and MC-EDD2 reduce the MCAST cell loss and increase the NMCAST cell loss compared to EDD. This is because of the shifting of priority by the deadline weighting factor towards MCAST and that the majority of NMCAST cells are of R.D. type. For MC-EDD1, the decrease in MCAST cell loss more than offsets the increase in NMCAST resulting in a decrease in overall cell loss compared to EDD. However MC-EDD2 increases the overall cell loss compared to EDD. This is because of priority shifting towards MCAST type is more than in MC-EDD1 due to the lower value of w_i coupled with the large percentage of MCAST (75%) relative to NMCAST (25%).

$\rho = 0.9$	MCAST = 25%		NMCAST = 75%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 75%	R.D. = 25%	
FCFS*	1.35e-2				
	4.25				
	1.99e-2		2.04e-2		0
	4.11		4.11		
	2.27e-2	1.14e-5	2.72e-2	1.04e-5	4.52
	4.06	4.25	4.06	4.25	
EDD	4.44e-4				
	12.7				
	7.04e-4		6.58e-4		0
	2.4		2.39		
	4.62e-6	2.6e-3	1.62e-6	2.56e-3	33.0
	2.14	3.12	2.14	3.12	
MC-EDD1	1.42e-5				
	13.34				
	1.11e-6		2.86e-6		0
	2.14		2.5		
	1.52e-6	0	3.33e-6	1.51e-5	34.87
	1.98	2.55	2.28	3.14	
MC-EDD2	1.91e-5				
	13.33				
	8.12e-7		3.86e-5		0
	2.11		2.51		
	9.94e-7	3.17e-7	4.54e-5	1.91e-5	34.84
	1.98	2.46	2.29	3.14	

$\rho = 0.8$	MCAST = 25%		NMCAST = 75%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 75%	R.D. = 25%	
FCFS*	1.55e-3				
	2.81				
	2.44e-3		2.5e-3		0
	2.69		2.69		
	3.36e-3	1.33e-6	3.36e-3	1.25e-6	2.99
	2.69	2.72	2.69	2.72	
EDD	4.35e-5				
	3.06				
	7.29e-5		6.87e-5		0
	1.98		1.97		
	4.76e-3	2.66e-4	2.07e-8	2.66e-4	4.86
	1.84	2.33	1.85	2.32	
MC-EDD1	2.82e-6				
	3.07				
	7.95e-7		5.85e-6		0
	1.81		2.02		
	0	2.92e-6	7.06e-6	2.34e-6	4.89
	1.74	2	1.91	2.34	
MC-EDD2	2.79e-6				
	3.07				
	3.78e-7		5.93e-6		0
	1.8		2.03		
	0	1.39e-6	6.97e-6	2.92e-6	4.9
	1.74	1.97	1.92	2.35	

Table 1: 25% Multicast, 75% Non-Multicast (75% T.D., 25%R.D.) for $\rho = 0.9, 0.8$.

$\rho = 0.9$	MCAST = 25%		NMCAST = 75%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 25%	R.D. = 75%	
FCFS*	1.15e-2				
	4.55				
	1.8e-2		1.69e-2		0
	4.42		4.43		
	6.3e-2	1.48e-3	6.31e-2	1.48e-3	4.8
	4.06	4.55	4.05	4.55	
EDD	8.84e-6				
	13.44				
	1.42e-5		1.3e-5		0
	2.4		2.4		
	0	1.95e-5	0	1.75e-5	35.01
	1.89	2.59	1.89	2.59	
MC-EDD1	7.22e-6				
	13.35				
	4.27e-8		1.47e-5		0
	2.05		2.53		
	1.59e-7	0	2.11e-5	1.25e-5	34.91
	1.87	2.12	1.99	2.72	
MC-EDD2	7.59e-6				
	13.35				
	0		1.55e-5		0
	2.04		2.54		
	0	0	1.92e-5	1.42e-5	34.96
	1.89	2.1	2	2.72	

$\rho = 0.8$	MCAST = 25%		NMCAST = 75%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 25%	R.D. = 75%	
FCFS*	1.21e-3				
	2.83				
	3.68e-3		1.32e-3		0
	2.7		2.73		
	5.05e-3	2.93e-5	5.04e-3	2.42e-5	3
	2.69	2.74	2.69	2.74	
EDD	3.4e-6				
	3.07				
	4.46e-6		4.95e-6		0
	1.97		1.97		
	0	6.13e-6	0	4.69e-6	4.88
	1.71	2.06	1.71	2.07	
MC-EDD1	2.58e-6				
	3.06				
	0		4.6e-6		0
	1.77		2.04		
	0	0	4.53e-6	5.97e-6	4.89
	1.69	1.8	1.76	2.13	
MC-EDD2	3e-6				
	3.07				
	0		4.7e-6		0
	1.77		2.04		
	0	0	4.99e-6	4.13e-6	4.9
	1.69	1.8	1.77	2.14	

Table 2: 25% Multicast, 75% Non-Multicast (25% T.D., 75%R.D.) for $\rho = 0.9, 0.8$

$\rho = 0.9$	MCAST = 50%		NMCAST = 50%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 75%	R.D. = 25%	
FCFS*	1.09e-2				
	4.03				
	1.65e-2		1.65e-2		0
	3.9		3.9		
	2.23e-2	9.23e-6	2.24e-2	8.97e-6	4.28
	3.86	4.03	3.86	4.02	
EDD	4.53e-4				
	12.74				
	6.83e-4		6.84e-4		0
	2.39		2.39		
	6.13e-6	2.61e-3	1.67e-6	2.62e-3	33.09
	2.14	3.11	2.13	3.12	
MC-EDD1	2.15e-5				
	13.51				
	7.42e-6		5.75e-5		
	2.21		2.61		
	9.16e-6	2.48e-6	6.18e-5	4.53e-5	0
	2.05	2.64	2.39	3.22	35.39
MC-EDD2	4.18e-5				
	13.47				
	3.96e-6		1.22e-4		0
	2.1		2.72		
	6.13e-6	1.78e-6	1.55e-4	8.88e-5	35.29
	1.98	2.22	2.35	3.08	

$\rho = 0.8$	MCAST = 50%		NMCAST = 50%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 75%	R.D. = 25%	
FCFS*	1.56e-3				
	2.81				
	2.52e-3		2.5e-3		0
	2.69		2.69		
	3.41e-3	1.33e-6	3.38e-3	1.21e-6	2.99
	2.69	2.72	2.69	2.72	
EDD	4.46e-5				
	3.06				
	7.36e-5		6.92e-5		0
	1.97		1.97		
	4.61e-7	2.8e-4	2.15e-7	2.64e-7	4.88
	1.85	2.32	1.84	2.32	
MC-EDD1	3.08e-6				
	3.08				
	1.17e-6		8.69e-6		0
	1.86		2.08		
	1.42e-6	4.55e-7	1e-5	4.84e-6	4.9
	1.78	2.09	1.97	2.38	
MC-EDD2	4.38e-5				
	3.07				
	1.09e-6		1.29e-5		0
	1.85		2.09		
	1.26e-6	6.07e-7	1.46e-5	8.18e-6	4.91
	1.79	2.03	1.99	2.39	

Table 3: 50% Multicast, 50% Non-Multicast (75% T.D., 25%R.D.) for $\rho = 0.9, 0.8$

$\rho = 0.9$	MCAST = 50%		NMCAST = 50%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 25%	R.D. = 75%	
FCFS*	9.11e-3				
	4.27				
	1.37e-2		1.38e-2		0
	4.16		4.16		
	4.98e-2	1.05e-3	5.02e-2	1.03e-3	4.5
	3.86	4.26	3.87	4.26	
EDD	1.26e-5				
	13.43				
	2.13e-5		1.67e-5		0
	2.41		2.4		
	0	2.88e-5	0	2.25e-5	35.12
	1.89	2.59	1.89	2.58	
MC-EDD1	7.95e-6				
	13.54				
	2.19e-6		2.17e-5		0
	2.15		2.67		
	3.5e-6	1.74e-6	2.24e-5	2.15e-5	35.48
	1.91	2.24	2.08	2.87	
MC-EDD2	1.09e-5				
	13.53				
	2.56e-6		3.02e-5		0
	2.14		2.68		
	5.73e-6	1.44e-6	3.52e-5	2.84e-5	35.46
	1.95	2.21	2.11	2.88	
$\rho = 0.8$	MCAST = 50%		NMCAST = 50%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 25%	R.D. = 75%	
FCFS*	1.04e-3				
	2.84				
	1.67e-3		1.67e-3		0
	2.73		2.73		
	6.2e-3	6.76e-3	6.21e-3	6.34e-5	3
	2.69	2.75	2.69	2.75	
EDD	2.8e-6				
	3.07				
	5.15e-6		3.8e-6		0
	1.97		1.97		
	0	6.97e-6	0	5.16e-6	4.9
	1.71	2.06	1.71	2.06	
MC-EDD1	2.4e-6				
	3.08				
	7.14e-7		6.98e-6		0
	1.84		2.11		
	1.21e-6	5.38e-7	1.05e-5	5.73e-6	4.91
	1.71	1.88	1.81	2.22	
MC-EDD2	2.57e-6				
	3.08				
	8.13e-7		7.4e-6		0
	1.83		2.11		
	1.74e-6	4.84e-7	8.18e-6	7.12e-6	4.91
	1.73	1.86	1.83	2.22	

Table 4: 50% Multicast, 50% Non-Multicast (25% T.D., 75%R.D.) for $\rho = 0.9, 0.8$

$\rho = 0.9$	MCAST = 75%		NMCAST = 25%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 75%	R.D. = 25%	
FCFS*	1.36e-2				
	4.25				
	2.06e-2		2e-2		0
	4.12		4.12		
	2.74e-2	9.39e-6	2.74e-2	1.14e-5	4.51
	4.07	4.27	4.07	4.27	
EDD	4.47e-4				
	12.68				
	6.61e-4		7.09e-4		0
	2.39		2.4		
	4.18e-6	2.56e-3	1.23e-6	2.63e-3	33.01
	2.14	3.12	2.13	3.13	
MC-EDD1	2.29e-5				
	13.29				
	1.35e-5		9.43e-5		0
	2.29		2.73		
	1.47e-5	1.02e-5	1e-4	7.75e-5	34.79
	2.13	2.75	2.51	3.32	
MC-EDD2	7.96e-5				
	13.2				
	3.77e-5		3.53e-4		0
	2.28		2.77		
	4.53e-5	0	3.91e-4	2.51e-4	34.52
	2.16	2.03	2.56	3.33	

$\rho = 0.8$	MCAST = 75%		NMCAST = 25%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 75%	R.D. = 25%	
FCFS*	1.55e-3				
	2.8				
	2.51e-3		2.43e-5		0
	2.69		2.69		
	3.39e-3	9.99e-7	3.34e-3	4.44e-7	2.98
	2.67	2.72	2.69	2.72	
EDD	4.26e-5				
	3.04				
	6.91e-5		6.62e-5		0
	1.97		1.97		
	2.69e-7	2.67e-4	1.89e-7	2.42e-4	4.8
	1.84	2.32	1.84	2.32	
MC-EDD1	2.66e-6				
	3.06				
	2.19e-6		1.01e-5		0
	1.91		2.15		
	2.69e-6	7.29e-7	1.05e-5	9.03e-6	4.86
	1.83	2.14	2.05	2.43	
MC-EDD2	7.65e-6				
	3.05				
	3.7e-6		3.64e-5		0
	1.9		2.17		
	4.75e-6	6.77e-7	4.23e-5	2.06e-5	4.85
	1.84	2.06	2.07	2.43	

Table 5: 75% Multicast, 25% Non-Multicast (75% T.D., 25%R.D.) for $\rho = 0.9, 0.8$

$\rho = 0.9$	MCAST = 75%		NMCAST = 25%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 25%	R.D. = 75%	
FCFS*	1.15e-2				
	4.56				
	1.7e-2		1.88e-2		0
	4.44		4.43		
	6.36e-2	1.49e-3	6.38e-2	1.46e-3	4.79
	4.06	4.56	4.07	4.56	
EDD	9.24e-6				
	13.31				
	1.42e-6		1.32e-5		0
	2.41		2.4		
	0	1.9e-5	0	1.81e-5	34.87
	1.89	2.59	1.89	2.59	
MC-EDD1	7.96e-6				
	13.32				
	6.19e-6		2.85e-5		0
	2.27		2.81		
	8.53e-6	5.37e-6	2.49e-5	2.98e-5	34.88
	1.96	2.37	2.18	3.04	
MC-EDD2	1.77e-5				
	13.29				
	7.6e-6		8.07e-5		0
	2.26		2.84		
	1.35e-5	5.56e-6	2.55e-6	7.9e-5	34.82
	2.02	2.34	2.25	3.05	
$\rho = 0.8$	MCAST = 75%		NMCAST = 25%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 25%	R.D. = 75%	
FCFS*	1.04e-3				
	2.84				
	1.64e-3		1.72e-3		0
	2.73		2.73		
	6.17e-3	6.91e-5	6.16e-3	5.76e-5	3
	2.69	2.75	2.69	2.75	
EDD	2.11e-6				
	3.05				
	3.82e-6		2.16e-6		
	1.97		1.97		
	0	5.15e-6	0	2.98e-6	4.84
	1.71	2.06	1.71	2.06	
MC-EDD1	1.83e-6				
	3.06				
	1.52e-6		6.93e-6		0
	1.89		2.19		
	2.66e-6	1.12e-6	6.53e-6	7.08e-6	4.86
	1.74	1.95	1.87	2.3	
MC-EDD2	2.44e-6				
	3.06				
	1.67e-6		1.02e-5		0
	1.89		2.2		
	3.07e-6	1.18e-6	8.06e-6	1.1e-5	4.86
	1.77	1.93	1.9	2.31	

Table 6: 75% Multicast, 25% Non-Multicast (25% T.D., 75%R.D.) for $\rho = 0.9, 0.8$

5 Implementation Issues

The MC-EDD policy is a simple policy to implement. Basically, it is an enhancement to the EDD policy which has readily been implemented [16]. The addition of the *cell attribute* field to the internal cell header in the switch is easy since it is another piece of information similar to the deadline value and cell arrival time. Since the weighting of the deadline is done at connection setup time, the host computer that is establishing the connection can compute the weighted value and pass it to the switch. The switch can store the weighted value in its internal cell header in a way similar to storing the deadline value in EDD. The weighted value can be implemented as an integer value which is computed by a table lookup in the host for every possible deadline value and a range of weighting values. This computation is possible, for example, if there are 100 possible deadline values and the possible weighting values w_i are 100 (0.01, 0.02, ..., 0.98, 0.99), a two dimensional table of increasing integer values can be established having 10^4 elements. The host looks up the deadline and the weighting factor and passes the integer weighted deadline valueto the switch. This scheme assumes that weighting factors can be rounded to two decimal places, which is a reasonable assumption for most applications. The *cell attribute* then becomes the addition of this weighted deadline integer value to the current value of the clock at the cell arrival. The implementation of clocks in ATM switches has been discussed in the literature[16, 4].

With the above implementation approach, we believe that the MC-EDD technique can be easily and efficiently implemented.

6 Conclusions

We presented a new scheduling technique in an ATM multiplexer which considers real-time multicast, real-time non-multicast, and non-real-time traffic. We showed how the scheduling technique uses the information available at multicast setup time to give appropriate priority to real-time multicast traffic. We also showed that the new technique correlates the scheduling decision taken at each multiplexer in an ATM switching node by using the multicast tree size and the subtree sizes rooted at each multiplexer. We compared, using simulation, the performance of the new buffering technique with respect to cell loss and delay with two of the well known real-time scheduling techniques. We showed, using a real-time traffic model having two deadlines, tight and relaxed, that the new technique outperforms the other two techniques by reducing the cell loss for real-time multicast traffic over all the range of parameter values. We also showed that the new technique improves the the non-multicast cell loss when the percentage of multicast traffic class is less than, equal to, greater than (in many subcases) the non-multicast traffic. We also showed that the new technique does not affect the delay for non-real-time. Finally, we discussed a how to practically and efficiently implement that new technique.

Appendix

$\rho = 0.9$	MCAST = 25%		NMCAST = 75%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 25%	R.D. = 75%	
FCFS*	1.22e-2				
	4.43				
	3.54e-2		1.24e-2		0
	4.17		4.34		
	4.83e-2	4.78e-4	4.82e-2	4.98 e-4	4.69
	4.06	4.43	4.06	4.43	
EDD	1.91e-5				
	13.32				
	9.6e-6		3.56e-5		0
	2.14		2.5		
	0	3.57e-5	0	4.79e-5	34.83
	1.94	2.69	1.94	2.69	
MC-EDD1	7.44e-6				
	13.35				
	9.4e-7		1.48e-5		0
	1.98		2.56		
	1.29e-6	0	1.92e-5	1.33e-5	34.91
	1.90	2.18	2.04	2.74	
MC-EDD2	7.85e-6				
	13.35				
	6.41e-7		1.58e-5		0
	1.97		2.56		
	7.6e-7	3.17e-7	2.04e-5	1.42e-5	34.9
	1.91	2.15	2.05	2.74	

$\rho = 0.8$	MCAST = 25%		NMCAST = 75%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 25%	R.D. = 75%	
FCFS*	1.21e-3				
	2.83				
	3.68e-3		1.32e-3		0
	2.7		2.73		
	5.05e-3	2.93e-5	5.04e-3	2.42e-5	3
	2.69	2.74	2.69	2.74	
EDD	2.75e-6				
	3.07				
	1.82e-6		5.33e-6		0
	1.84		2.02		
	0	6.67e-6	0	7.18e-6	4.89
	1.74	2.12	1.74	2.12	
MC-EDD1	2.6e-6				
	3.07				
	1.25e-6		5.2e-6		0
	1.74		2.05		
	0	4.58e-6	5.47e-6	5.11e-6	4.89
	1.71	1.83	1.79	2.14	
MC-EDD2	2.03e-6				
	3.07				
	4.54e-7		5.29e-6		0
	1.74		2.06		
	0	3.33e-6	5.26e-6	3.91e-6	4.9
	1.71	1.93	1.79	2.15	

Table 7: 25% Multicast(75% T.D., 25%R.D.), 75% Non-Multicast(25% T.D., 75%R.D.) for $\rho = 0.9, 0.8$.

$\rho = 0.9$	MCAST = 25%		NMCAST = 75%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 75%	R.D. = 25%	
FCFS*	1.33e-2				
	4.28				
	8.44e-3		2.37e-2		0
	4.23		4.12		
	3.12e-2	4.35e-5	3.16e-2	3.9e-5	4.56
	4.06	4.29	4.06	4.29	
EDD	1.87e-4				
	13.05				
	5.5e-4		1.87e-4		0
	2.72		2.29		
	7.94e-7	7.53e-4	3.64e-7	7.26e-4	34.04
	2.06	2.96	2.06	2.96	
MC-EDD1	1.1e-5				
	13.34				
	2.56e-7		2.23e-5		0
	2.31		2.44		
	7.94e-7	5.84e-8	2.54e-5	1.36e-5	34.89
	1.95	2.45	2.21	2.21	3.1
MC-EDD2	1.48e-5				
	13.33				
	8.55e-8		3.02e-5		0
	2.26		2.46		
	3.17e-7	0	3.42e-5	1.84e-5	34.87
	1.96	2.38	2.23	3.11	

$\rho = 0.8$	MCAST = 25%		NMCAST = 75%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 75%	R.D. = 25%	
FCFS*	1.46e-3				
	2.81				
	1.02e-3		2.8e-3		0
	2.71		2.7		
	3.75e-3	2.92e-6	3.78e-3	3.42e-6	2.99
	2.69	2.72	2.7	2.72	
EDD	1.6e-5				
	3.06				
	5.18e-5		1.64e-5		0
	2.12		1.92		
	0	7.12e-5	0	6.35e-5	4.87
	1.8	2.24	1.8	2.25	
MC-EDD1	2.77e-6				
	3.07				
	0		6e-6		0
	1.9		1.99		
	0	0	6.3e-6	5.15e-6	4.89
	1.72	1.97	1.87	2.33	
MC-EDD2	2.9e-6				
	3.07				
	0		6.3e-6		0
	1.88		2		
	0	0	6.23e-6	3.23e-6	4.9
	1.73	1.94	1.89	2.34	

Table 8: 25% Multicast(25% T.D., 75%R.D.), 75% Non-Multicast(75% T.D., 25%R.D.) for $\rho = 0.9, 0.8$.

	MCAST = 50%		NMCAST = 50%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 25%	R.D. = 75%	
FCFS*	1.03e-2				
	4.12				
	2.29e-2		8.16e-3		0
	3.93		4.04		0
	3.08e-2	1.13e-4	3.11e-2	1.1e-4	4.35
	3.86	4.1	3.87	4.35	
EDD	6.58e-5				
	13.33				
	5.68e-5		1.41e-4		0
	2.2		2.6		
	4.5e-8	2.18e-4	0	1.91e-4	34.81
	1.99	2.81	1.99	2.82	
MC-EDD1	1.02e-5				
	13.53				
	3.82e-6		2.69e-5		0
	2.08		2.74		
	4.89e-6	7.69e-7	2.93e-5	2.6e-5	35.46
	1.98	2.36	2.2	2.92	
MC-EDD2	1.57e-5				
	13.52				
	4.73e-6		4.25e-5		0
	2.07		2.74		
	6.04e-6	1.03e-6	4.88e-5	4.3e-5	35.42
	1.99	2.29	2.22	2.93	

$\rho = 0.8$	MCAST = 50%		NMCAST = 50%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 25%	R.D. = 75%	
FCFS*	1.36e-3				
	2.82				
	3.22e-3		1.15e-3		0
	2.7		2.72		
	4.35e-3	8.85e-6	4.35e-3	9.03e-6	2.99
	2.69	2.73	2.69	2.73	
EDD	5.92e-6				
	3.07				
	6.19e-6		1.28e-5		0
	1.87		2.07		
	0	2.36e-5	0	1.73e-5	4.9
	1.18	2.07	1.77	2.17	
MC-EDD1	2.48e-6				
	3.08				
	9.32e-7		7.02e-6		0
	1.8		2.14		
	1.1e-6	4.55e-7	1.03e-5	5.86e-6	4.91
	1.75	1.94	1.88	2.33	
MC-EDD2	2.78e-6				
	3.08				
	7.93e-7		8.09e-6		0
	1.8		2.14		
	7.8e-7	8.33e-7	1.02e-5	7.33e-6	4.91
	1.76	1.91	1.89	2.23	

Table 9: 50% Multicast(75% T.D., 25%R.D.), 50% Non-Multicast(25% T.D., 75%R.D.) for $\rho = 0.9, 0.8$.

$\rho = 0.9$	MCAST = 50%		NMCAST = 50%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 75%	R.D. = 25%	
FCFS*	1.03e-2				
	4.11				
	8.08e-3		2.29e-2		0
	4.04		3.93		
	3.07e-2	1.16e-4	3.09e-2	1.23e-4	4.35
	3.86	4.1	3.86	4.1	
EDD	7.16e-5				
	13.3				
	1.64e-4		5.18e-5		0
	2.6		2.21		
	0	2.22e-4	0	1.99e-4	34.74
	1.99	2.81	1.99	2.81	
MC-EDD1	1.2e-5				
	13.53				
	2.31e-6		3.41e-5		0
	2.34		2.48		
	4.7e-6	1.47e-6	3.43e-5	3.35e-5	35.45
	1.96	2.47 2.24	3.15		
MC-EDD2	2.41e-5				
	13.5				
	2.6e-6		6.99e-6		0
	2.3		2.52		
	5.98e-6	1.44e-6	7.29e-5	6.13e-5	35.38
	2	2.41	2.29	3.16	

$\rho = 0.8$	MCAST = 50%		NMCAST = 50%		NRT
	T.D. = 25%	R.D. = 75%	T.D. = 75%	R.D. = 25%	
FCFS*	1.36e-3				
	2.82				
	1.15e-3		3.22e-3		0
	2.72		2.7		
	4.37e-3	1.02e-5	4.36e-3	9.7e-6	2.99
	2.69	2.73	2.69	2.73	
EDD	6.4e-6				
	3.07				
	1.58e-5		8.4e-6		0
	2.07		1.87		
	0	2.14e-5	0	1.81e-5	4.89
	1.77	2.18	1.77	2.17	
MC-EDD1	2.63e-6				
	3.08				
	5.75e-7		7.84e-6		0
	1.93		2.01		
	1.14e-6	3.76e-7	8.57e-6	5.76e-6	4.91
	1.73	2	1.89	2.36	
MC-EDD2	3.25e-6				
	3.08				
	7.34e-7		9.68e-6		0
	1.91		2.03		
	1.44e-6	4.83e-7	1.01e-5	8.41e-6	4.91
	1.76	1.97	1.91	2.36	

Table 10: 50% Multicast(25% T.D., 75%R.D.), 50% Non-Multicast(75% T.D., 25%R.D.) for $\rho = 0.9, 0.8$.

$\rho = 0.9$	MCAST = 75%		NMCAST = 25%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 25%	R.D. = 75%	
FCFS*	1.33e-2				
	4.29				
	2.38e-3		8.62e-3		0
	4.13		4.24		
	3.18e-2	3.97e-5	3.19e-2	3.92e-5	
	4.07	4.31	4.07	4.3	4.54
EDD	1.88e-4				
	13.02				
	1.97e-4		5.27e-4		0
	2.29		2.72		
	4.44e-7	7.66e-4	0	7.21e-4	33.99
	2.06	2.96	2.06	2.96	
MC-EDD1	1.46e-5				
	13.3				
	9.62e-6		5.72e-5		0
	2.22		2.95		
	1.02e-5	7.89e-6	6.95e-5	5.27e-5	34.84
	2.09	2.59	2.41	3.15	
MC-EDD2	4.34e-5				
	13.25				
	2.35e-5		1.84e-4		0
	2.21		2.96		
	2.84e-5	1.8e-5	2.94e-4	1.61e-4	34.69
	2.12	2.89	2.45	3.15	

$\rho = 0.8$	MCAST = 75%		NMCAST = 25%		NRT
	T.D. = 75%	R.D. = 25%	T.D. = 25%	R.D. = 75%	
FCFS*	1.46e-3				
	2.81				
	2.87e-3		1.03e-3		0
	2.69		2.71		
	3.79e-3	2.92e-6	3.77e-3	1.58e-6	2.98
	2.69	2.73	2.69	2.72	
EDD	1.54e-5				
	3.05				
	1.79e-5		4.37e-5		0
	1.92		2.12		
	0	6.95e-5	0	6e-5	4.8
	1.8	2.24	1.8	2.24	
MC-EDD1	2.24e-6				
	3.06				
	1.95e-6		8.18e-6		0
	1.87		2.25		
	2.39e-6	6.77e-7	6.53e-6	8.8e-6	4.86
	1.81	2.06	1.99	2.34	
MC-EDD2	4.4e-6				
	3.06				
	2.58e-6		1.96e-5		0
	1.87		2.25		
	3.35e-6	3.65e-7	2.4e-5	1.8e-5	4.86
	1.83	2	2.01	2.34	

Table 11: 75% Multicast(75% T.D., 25%R.D.), 25% Non-Multicast(25% T.D., 75%R.D.) for $\rho = 0.9, 0.8$.

<table>
<tr><td rowspan="2">$\rho = 0.9$</td><td colspan="2">MCAST = 75%</td><td colspan="2">NMCAST = 25%</td><td rowspan="2">NRT</td></tr>
<tr><td>T.D. = 25%</td><td>R.D. = 75%</td><td>T.D. = 75%</td><td>R.D. = 25%</td></tr>
<tr><td rowspan="6">FCFS*</td><td colspan="5">1.23e-2</td></tr>
<tr><td colspan="5">4.43</td></tr>
<tr><td colspan="2">1.25e-2</td><td colspan="2">3.57e-2</td><td rowspan="2">0</td></tr>
<tr><td colspan="2">4.35</td><td colspan="2">4.18</td></tr>
<tr><td>4.86e-2</td><td>5.04e-4</td><td>4.86e-2</td><td>4.9e-4</td><td rowspan="2">4.67</td></tr>
<tr><td>4.07</td><td>4.44</td><td>4.07</td><td>4.44</td></tr>
<tr><td rowspan="6">EDD</td><td colspan="5">1.91e-5</td></tr>
<tr><td colspan="5">13.29</td></tr>
<tr><td colspan="2">3.46e-5</td><td colspan="2">1.25e-5</td><td rowspan="2">0</td></tr>
<tr><td colspan="2">2.5</td><td colspan="2">2.14</td></tr>
<tr><td>0</td><td>4.65e-5</td><td>0</td><td>4.65e-5</td><td rowspan="2">34.81</td></tr>
<tr><td>1.94</td><td>2.69</td><td>1.94</td><td>2.69</td></tr>
<tr><td rowspan="6">MC-EDD1</td><td colspan="5">9.53e-6</td></tr>
<tr><td colspan="5">13.31</td></tr>
<tr><td colspan="2">6.38e-6</td><td colspan="2">3.7e-5</td><td rowspan="2">0</td></tr>
<tr><td colspan="2">2.37</td><td colspan="2">2.52</td></tr>
<tr><td>9e-6</td><td>5.47e-6</td><td>3.63e-5</td><td>3.92e-5</td><td rowspan="2">34.87</td></tr>
<tr><td>1.98</td><td>2.5</td><td>2.86</td><td>3.19</td></tr>
<tr><td rowspan="6">MC-EDD2</td><td colspan="5">2.95e-5</td></tr>
<tr><td colspan="5">13.28</td></tr>
<tr><td colspan="2">1.03e-5</td><td colspan="2">1.41e-4</td><td rowspan="2">0</td></tr>
<tr><td colspan="2">2.34</td><td colspan="2">2.58</td></tr>
<tr><td>1.88e-5</td><td>7.37e-6</td><td>1.49e-4</td><td>1.10e-4</td><td rowspan="2">34.76</td></tr>
<tr><td>2.06</td><td>2.44</td><td>2.35</td><td>3.22</td></tr>
</table>

<table>
<tr><td rowspan="2">$\rho = 0.8$</td><td colspan="2">MCAST = 75%</td><td colspan="2">NMCAST = 25%</td><td rowspan="2">NRT</td></tr>
<tr><td>T.D. = 25%</td><td>R.D. = 75%</td><td>T.D. = 75%</td><td>R.D. = 25%</td></tr>
<tr><td rowspan="6">FCFS*</td><td colspan="5">1.21e-3</td></tr>
<tr><td colspan="5">2.82</td></tr>
<tr><td colspan="2">1.33e-3</td><td colspan="2">3.69e-3</td><td rowspan="2">0</td></tr>
<tr><td colspan="2">2.72</td><td colspan="2">2.7</td></tr>
<tr><td>5.08e-3</td><td>2.74e-3</td><td>5.06e-3</td><td>2.13e-5</td><td rowspan="2">2.99</td></tr>
<tr><td>2.69</td><td>2.73</td><td>2.69</td><td>2.74</td></tr>
<tr><td rowspan="6">EDD</td><td colspan="5">2.4e-6</td></tr>
<tr><td colspan="5">3.05</td></tr>
<tr><td colspan="2">4.87e-6</td><td colspan="2">9.96e-7</td><td rowspan="2">0</td></tr>
<tr><td colspan="2">2.01</td><td colspan="2">1.84</td></tr>
<tr><td>0</td><td>6.56e-6</td><td>0</td><td>3.65e-6</td><td rowspan="2">4.84</td></tr>
<tr><td>1.74</td><td>2.11</td><td>1.74</td><td>2.11</td></tr>
<tr><td rowspan="6">MC-EDD1</td><td colspan="5">1.98e-6</td></tr>
<tr><td colspan="5">3.06</td></tr>
<tr><td colspan="2">1.53e-6</td><td colspan="2">1.53e-6</td><td rowspan="2">0</td></tr>
<tr><td colspan="2">1.95</td><td colspan="2">2.04</td></tr>
<tr><td>2.5e-6</td><td>1.19e-6</td><td>8.07e-6</td><td>6.94e-6</td><td rowspan="2">4.86</td></tr>
<tr><td>1.75</td><td>2.02</td><td>1.91</td><td>2.38</td></tr>
<tr><td rowspan="6">MC-EDD2</td><td colspan="5">3.38e-6</td></tr>
<tr><td colspan="5">3.06</td></tr>
<tr><td colspan="2">1.76e-6</td><td colspan="2">1.57e-5</td><td rowspan="2">0</td></tr>
<tr><td colspan="2">1.94</td><td colspan="2">2.07</td></tr>
<tr><td>3.18e-6</td><td>1.27e-6</td><td>1.69e-5</td><td>1.25e-5</td><td rowspan="2">4.86</td></tr>
<tr><td>1.79</td><td>1.99</td><td>1.95</td><td>2.39</td></tr>
</table>

Table 12: 75% Multicast(25% T.D., 75%R.D.), 25% Non-Multicast(75% T.D., 25%R.D.) for $\rho = 0.9, 0.8$.

References

[1] D. Yates, J. Kurose, D. Towsley, and M.G.Hluchyj. "On per-session end-to-end delay and the call admission problem for real time applications with QOS requirements". In *Journal of High Speed Networks*, volume Vol 3, pp 429-458, December 1994.

[2] Debanjan Saha. "Supporting Distributed Multimedia Applications on ATM Networks". In *Ph.D. Dissertation*, 1995.

[3] Geping Chen and Ioannis Stavrakakis. "ATM Traffic Management With Diversified Loss and Delay". In *Proceedings of INFOCOM 96*. IEEE, 1996.

[4] H. Jonathan Chao and Daein Jeong. "Generalized Priority Queue Manager Design for ATM Switches". In *Proceedings of ICC 96*. IEEE, 1996.

[5] H. Schulzrinne. "Voice Communication Across the Internet: A Network Voice Terminal". In *Research Report, Dept. of C.S.* Umass, Amherst, July 1992.

[6] Henning Schulzrinne, J.Kurose, and D.Towsley. "An Evaluation of Scheduling Mechanisms for Providing Best-Effort, Real-Time Communication in Wide-Area Networks". In *Proceedings of INFOCOM 94*. IEEE, June 1994.

[7] Hui Zhang and Dominico Ferrari. "Rate-Controlled Static-Priority Queueing". In *Proceedings of INFOCOM 93*. IEEE, 1993.

[8] Hui Zhang and Edward W. Knightly. "RCSP and Stop-and-Go:A Comparison of Two Non-Work-Conserving Disciplines for Supporting Multimedia Communication". In *Multimedia Systems Journal*. ACM/Springer-Verlag, September 1995.

[9] Jin Li and Chuan-lin Wu. "Design and Implementation of a Multicast-Buffer ATM Switch". In *Proceedings of ICNP 95*. IEEE, 1995.

[10] J.V.Nielsen, L.Dittman, J.k.Madsen and P.S.Lassen. "VLSI Implementation of a Fairness ATM Buffer System". In *Proceedings of ICC 96*. IEEE, 1996.

[11] M. Vishnu and J.W. Mark. "HOL-EDD: A Flexible Service Scheduling Scheme for ATM Networks". In *Proceedings of INFOCOM 96*. IEEE, 1996.

[12] Partho P. Mishra and Hemant Kanakia. "On Hop by Hop Rate-based Congestion Control Scheme". In *IEEE J. Transactions on Networking*. IEEE, September 1996.

[13] Pramod Pancha and Mark Karol. "Guaranteeing Bandwidth And Minimizing Delay In Packet-Switched(ATM) Networks". In *Proceedings of GLOBCOM 96*. IEEE, 1996.

[14] R. Chiplkatti, J.F.Kurose and D. Towsley. "Scheduling Policies for Real-Time and Non-Real Time Traffic in a statistical Multiplexer". In *Proceedings of INFOCOM 89*. IEEE, 1989.

[15] Riccardo Bettati and Amit Gupta. “Dynamic Resource Migration for Multiparty Real-Time Communication”. In *Technical Report TR-95-060*. IEEE, October 1995.

[16] R.Y. Awdeh and H.T. Mouftah. “Survey of ATM Switch Architectures”. In *Computer Networks and ISDN Systems*, November 1995.

[17] S.A.Banawan, and G.Radhakrishnan. “An Adaptive Threshold Based Scheduling Policy for ATM Networks”. In *Proceedings of IPCCC*. IEEE, 1996.

[18] Seok-Kyu Kweon and Kang G.Shin. “Traffic-Controlled Rate-Monotonic Priority Scheduling of ATM Cells”. In *Proceedings of INFOCOM 96*. IEEE, 1996.

[19] Sridhar Pingali. “Protocol and Real-Time Scheduling Issues For MultiMedia Applications”. In *Ph.D. Dissertation*, September 1994.

[20] Ting-li Ling and Ness Shroff. “Scheduling Real-Time Traffic in ATM Networks”. In *Proceedings of INFOCOM 96*. IEEE, 1996.

[21] Victor Firoiu and Don Towsley. “Call Admission and Resource Reservation for Multicast Sessions”. In *Proceedings of INFOCOM 96*. IEEE, 1996.