Timer Imposed and Priority Supported (TIPS) Congestion Control Scheme for Point-to-Multipoint Connections in ATM

Muddassir TUFAIL and Bernard COUSIN
IRISA, Campus de Beaulieu
35042 Rennes Cedex (France)
{mtufail, bcousin}@irisa.fr

Abstract

The unicast traffic of ABR class in ATM networks has attracted a lot of discussions on its congestion control issue. The development of distributed applications have made the multicast domain equally important. A multicast connection can be simple (point-to-multipoint) or multiple (multipoint-to-multipoint). The diversity in multicast connection's nature is a big problem in designing a global congestion control scheme for them. In this paper we concentrate our attention on the simple multicast ABR traffic in ATM. Our proposed TIPS scheme is to tackle, presently, the congestion control issue in point-to-multipoint connections. Its two key features, $Timer$ and resource $Arbiter$, makes it efficient to solve the complex congestion control problems in simple multicast connections without any significant increase of work load at nodes. The source, following the ATM Forum specifications version 4.0 [12], generates a stream of RM cells for congestion control. The nodes need not to perform tedious calculations of explicit rate. The adjacent nodes inter-communicate and perform buffer management per port. On one hand, the $Timer$ specifies the time instant for each branching node to amalgamate the congestion informations and on the other hand, the $Arbiter$ maintains a good and balanced control over the queue lengths in branching nodes.
Keywords: congestion control, resource management (RM) cell, credit per port, multicast ABR traffic.

Résumé

Dans cet article, nous concentrons notre attention sur le trafic ABR multicast des réseaux ATM. Le mécanisme que nous proposons, TIPS, est chargé d'assurer le contrôle de congestion des connexions point à multipoint. Ses deux caractéristiques clefs, le $Timer$ (temporisateur) et l'$Arbiter$ (gestionnaire de ressources), rendent ce mécanisme efficace dans la résolution du contrôle de congestion des connexions multicast simples tout en ne surchargeant pas de manière significative les commutateurs ATM. Le débit de la source est contrôlé : en accord avec les spécifications de la version 4,0 [12] de l'ATM Forum ; la source génère périodiquement des cellule de gestion (RM) qui sont diffusées vers les destinataires de la connexion multicast. Ces dernières sont chargées de les renvoyer vers la source. Les commutateurs n'ont pas besoin d'effectuer de difficiles calculs du débit explicite : un simple bit d'indication de congestion suffit. Les commutateurs adjacents inter-communiquent en contrôlant globalement l'occupation de leur zone de memorisation des cellules, par lien de sortie. L'$Arbiter$ est chargé de maintenir un équilibre des différentes zones de mémorisation entre-elles.
Mots clefs: contrôle de congestion, cellule de gestion de ressources (RM), contrôle par crédit et par lien de sortie, trafic ABR multicast.

1 Introduction

The development of high speed networks (B-ISDN) has given rise to new application domains. These days the video-conferences interactive/non interactive are becoming an important part of business and educational communications. Moreover the distributed data applications require more efficient traffic and congestion control methods. In this paper, we concentrate our attention on multicast traffic in ATM networks. There are two main classes of multicast connections on the basis of their communication policy [15].

- **Simple Multicast Connection (point-to-multipoint):** The simple multicast connections have one source, which will be referred as root (see figure 1) and have many destinations, designated as leaves (see figure 1). The data cells are transmitted by the root and are received by all the leaves of the connection. As far as the data cells are concerned, the leaves have unidirectional data receptor behavior. They do not emit but the control cells[1] which are destined to root. Leaves do not inter-communicate. There is no guarantee that all the leaves receive the data cells at the same time. It is entirely a function of their distances from the root and the nature of links and switches involved.
  The figure 1 shows a multicast connection which forms a tree, connecting root to the leaves. At the same time, the nodes, shown in the figure 1, forward the cells belonging to unicast or any other multicast connection(s) but they are not shown for the reasons of simplicity. We would like to highlight the important role of node[2] 1 and node 2 in the multicast data transfer. At these node, the cells are replicated on all the branches[3] attached. These nodes are responsible for data cell's replication and managing their proper switching and forwarding. At the same time, these nodes must be equipped with a good buffer manager to serve equally the queues of replicated data.We declare these nodes as branching nodes (cf. section 3).
  An example of simple multicast connection can be a video-conference, being held at a given place

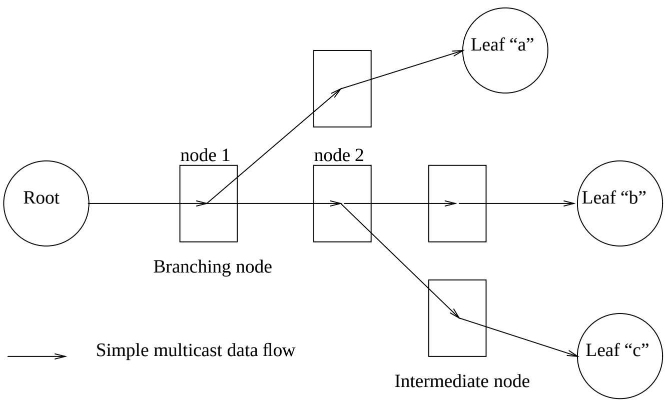

Figure 1: The simple multicast connection

and communicated via a high speed network to a certain remote participants. These remote partici-

[1] We are concerned with the congestion control cells only.
[2] A node, throughout our paper, means an ATM switch.
[3] All the virtual channels, at the exit of a node, carrying the replicated cells of the same simple multicast connection are referred to as branches here after.

pants can watch the conference in real time but they do not have the right to transmit their contributions to the conference.

- **Multiple Multicast Connection (multipoint-to-multipoint):** In this type of configuration, the data cells are transmitted by one of the end stations and all others receive them. Each station in the multicast connection is a receptor at a given instant and can be a emitter at the next instant. The multiple multicast connection is also tree structured connection with the exception that there is no one unique root[4] declared for the connection duration. The root properties are delegated to the different stations, on their demand, during the connection.
  The example of multiple multicast connection can be a video-conference where each participant, in addition to watching the other participant's remarks, can also contribute in the debate. His contribution will be received by all other end stations.
  The issue of multiple multicast connection is still to be brought under the discussion of ATM Forum.

Depending on the QoS required by the multicast application, a proper traffic class[5] is selected. Our work concerns the multicast traffic belonging to available bit rate (ABR) class. We elaborate, in the next section congestion control in simple multicast connection.

2 The Congestion Control in Simple Multicast Connections

Being a new domain in ABR traffic, the congestion control methods for simple multicast connections are in developing phase. The same issue in unicast traffic (point-to-point connection) has gone through an appreciable amount of experimentation and discussion. There have been efforts to solve the congestion control in multicast by extending the techniques proposed for unicast traffic. In [15], the congestion control schemes proposed for unicast traffic (rate based and credit based techniques) are analyzed to control the congestion in ABR multicast traffic. Rate-based technique when implemented for multicast traffic cause the following problems to a root.

1. At what instant the source should increase the rate?
2. How many RM cells source should ignore before reacting?
3. How can a source differentiate the congestion at a branching node from that at one of nodes attached?

The implementation of credit-based technique means dealing each replicated queue at a branching node independently. This independent treatment makes it difficult for the branching node to declare its one unique credit value to upstream node thus increases the buffer requirements at branching nodes. Kai-Yeung Siu has proposed multicast congestion control scheme ensuring max-min fair rate allocation [13]. It adapts the root rate to the minimum rate among all VCs to multiple leaves in a multicast traffic connection thus restricts the multicast flow in the branches where network resources are available.

The ATM forum defines operating policies for ABR point-to-multipoint connections which are implementation specific [12]. The flow control mechanism is ensured by responding to the combined feedback from the leaves. The branching nodes have the option to send fewer RM cells to root than they receive from the leaves but the method to amalgamate the information[6] carried by them is left to be decided by implementation policy maker.

The procedure of amalgamating the information, at a branching node is expected to be logical and efficient

[4] In simple multicast connection there is one unique root for whole connection duration.

[5] Classes defined in ATM (by ATM forum [12]) are constant bit rate (CBR) class, real-time variable bit rate (rt-VBR) class, non-real-time variable bit rate (nrt-VBR) class, available bit rate (ABR) class and unspecified bit rate (UBR) class.

[6] A RM cell bring information to root in at least one of these field and bits [12]: explicit rate (ER) field, congestion indication (CI) bit and no increase (NI) bit.

enough to yield a realistic overall congestion image of all the branches attached. That's why, in a multicast connection, a branching node has an important role in congestion control along with the task of data transfer. Consider the node 2 of figure 1, it duplicates data cells (including RM cells) on two branches leading to leaf "b" and leaf "c". Following the ATM specifications version 4.0 [12], a stream of RM cells, generated by the root, is used for congestion control . The node 2 receives back two RM cells, looped back independently by leaf "b" and leaf "c". Following points reflect the extent of complexity up to which the computations at node 2 may reach:

- The leaves are not necessarily at the same distance from the branching node. Moreover, they are attached, more probably, to branching node through switches and physical links of different speeds. Thus, the arrival of RM cells at node 2 from the leaves will be at different instants. Therefore the process of their evaluation and of amalgamating their information can not be performed but at specific instants.
- The informations carried by RM cells, coming from leaf "b" and leaf "c", are dependent on the network conditions on the respective branches. Evidently, RM cells, arriving at branching node 2, have different values in for explicit rate (ER) field, congestion indication (CI) bit and not increase (NI) bit. A comprehensive policy, which evaluates each RM cell's fields separately and then calculate their mean or averaged values for amalgamated RM cell, becomes indispensable. Such policy is function of network diversity (regarding link speeds, switch speeds and their congestion control mechanisms) among attached branches. The probability of having a complex amalgamating policy is quite high when number of attached branches is significant.

This article proposes a congestion control scheme for simple multicast connection which takes care of complex processing, elaborated above, of RM cells at branching node. Our proposed scheme is in accordance with the ATM Forum specification version 4.0 [12]. We enrich the behavior of branching node by a $Timer$ and a resource $Arbiter$[7].

3 Timer Imposed and Priority Supported (TIPS) Scheme

Following are the important definitions and parameters used in TIPS scheme. These definitions are valid for simple multicast connections.

- **Group:** The group means the set of all the nodes and the end stations, regardless of their nature or position, participating in point-to-multipoint communication.
- **Root:** As described earlier, this is the data emitter node i.e. the source. No other node in the group can be a root.
- **Leaf:** The destination on the extremity of each branch of multicast connection is called as leaf.
- **Branching node:** The node where replication of cells is performed is called as branching node. Root can be a branching node too.
- **Intermediate node:** All other nodes except the root, the branching nodes and the leaves are called as intermediate nodes.
- $Timer$**:** The $Timer$ is associated to the root and to all the branching nodes. This is the time interval during which a node expects to receive the congestion control feed back signal from the downstream nodes. This feed back signal is carried in a resource management (RM) cell. The value of $Timer$ decreases as we move from the root to the leaves. At a node, the value of $Timer$ is calculated as:

[7] This technical term has, originally, been introduced for naming certain sophisticated driver devices in computer operating systems.

$$Timer_i = \max \{ RTT_i^j \text{ such that j } \epsilon \; Leaf_i \}$$

Where:
$Timer_i$= $Timer$ value at node "i".
RTT_i^j= Round trip time to reach leaf "j" from node "i".
$Leaf_i$ = Set of all the leaves accessible through node "i".

- **Nrm:** The number of data cells to be transmitted per RM cell. Nrm is set at the call setup time and controls the frequency of RM cells. Larger values of Nrm reduce line overhead but reduce the tightness of the control loop. Additionally, the action of having transmitted forward Nrm data cells triggers the $Timer_i$ at a branching node "i". The value of Nrm is selected such that the root does not emit less cells than Nrm value in time interval equal to $Timer_{root}$. If R in the root emission rate in bytes per second[8] then:

$$\text{Nrm} \leq \text{R/53} * Timer_{root}$$

This relation shows that the triggering of new $Timer$ is always either before or at the termination of precedent one.

3.1 The $Timer$

It is important to determine the time instants at which the branching nodes amalgamate the information, carried by the RM cells, to create one unique backward RM cell. These instants are specific to each branching node and are determined by $Timer$. This section is devoted to explain the parameter $Timer$. Taking the example of point-to-multipoint connection shown in figure 1, we suppose, in order to calculate $Timer$ values at root, node 1 and node 2, that the maximum corresponding RTT value for all the three nodes is that of leaf "b". That's why we have shown in figure 2 the $Timers$ triggering and their terminations with reference to leaf "b". The Nrm taken, in this case, is 10. The figure shows the flow of cells from the root to the leaf "b" following the node 1 and the node 2. Since the nodes are not equidistant (taking all other parameters, involving switch and link speeds, as uniform), so the cells reach the nodes with different inter-node delays. It is clear, that $Timer_{root}$ is triggered when the root has transmitted Nrm data cells. At the same time a RM cell is generated which arrives back at the root before/at the termination of $Timer_{root}$. The $Timers$ at other nodes are triggered similarly as shown in figure 2. It is to be noted that as we go ahead in connection topology along the data flow direction, the $Timer$ triggering is not only delayed but their termination is earlier too. The second point to highlight is that the triggering of the next $Timer_{root}$ is before the termination of the previous one which guarantees the non-existence of any time zone during which flow of data cells may risk to be un-supervised.

3.2 TIPS's Working Principle

TIPS scheme assigns important functions to the root, to the branching nodes and to the leaves. We will follow the track of RM cells, first in forward direction then in backward direction, to explain the TIPS's working principles at different network elements (root, intermediate nodes, branching nodes and leaf).
Root generates a stream of RM cells, whose frequency is determined by Nrm value. These RM cells are replicated at branching nodes in a similar way as the data cells. On the branching nodes (node 1 and node 2 in figure 1), the cells are replicated to all the branches attached. These branching nodes are equipped with an resource $Arbiter$ which ensures a good service balance between unicast and multicast virtual channels

[8] R is actually the source emission rate the network can guarantee to offer for most of the connection's duration. It can, safely, be taken equal to minimum cell rate (MCR).

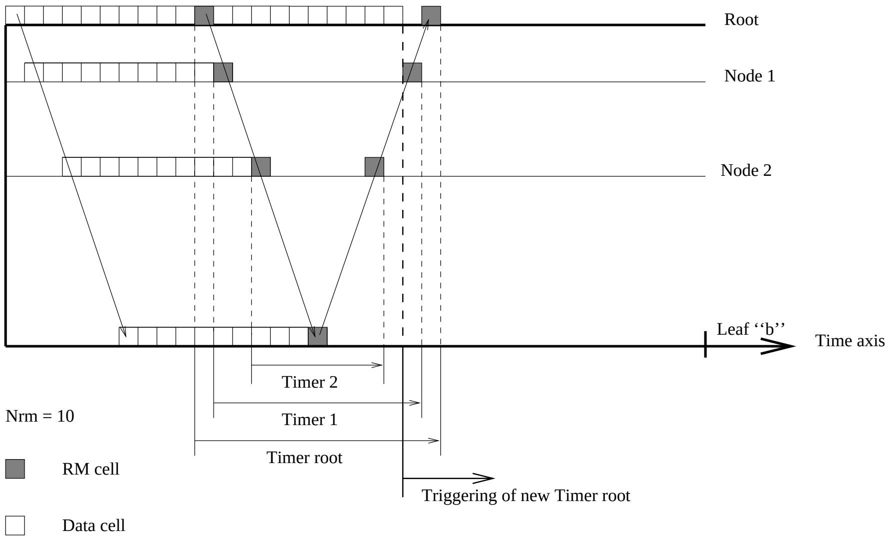

Figure 2: The $Timer$ values at different nodes.

(VC) and within themselves too. This $Arbiter$ implements an algorithm [5] which, dynamically, gives or draws the priority to a multicast queue according to buffer occupation rate of other queues (unicast and/or multicast) at the output port. On the root and on all the branching nodes[9], the $Timer$ is triggered after each Nrm data cells.
All the nodes, participating in the connection, implement at least one of the following methods[10] to control the congestion at queuing points:

- A node may set the explicit forward congestion indication (EFCI)bit in the data cell header.
- A node may set congestion indication (CI) or not increase (NI) bit in forward and/or backward RM cells[11].

At a branching node, in addition to above two cases, the bits (EFCI, CI or NI) can also be marked to ensure the fair allocation among the contending VCs at an output port. The branching nodes keep an eye on the emission rate of each VC: the cells of a VC, whose rate of emission is more than its fair share, are marked regardless of the congestion level at the node. There have been proposals to calculate the fair share of a VC. The max-min fairness [7] and weighted fairness [2] are possible fair allocation criterias. The max-min fair allocation get the VCs an equal share on every link provided, without accounting for guaranteed minimum (MCR) where as weighted fairness determines optimal allocation of resources over and above MCR.
The adjacent nodes inter-communicate at regular intervals. A downstream node (with respect to data flow direction) declares the availability of its buffer to the corresponding upstream port. A node may implement input buffering, output buffering or central buffering. The input queuing discipline causes head of line blocking problem where as central queuing requires an expensive intelligent server. We propose output

[9] Root can also be a branching node.

[10] The options of reducing ER (explicit rate) field and segmenting the control loop, as defined in [12], is presently not considered for multicast for the reasons of simplicity.

[11] Availability of bandwidth in the return path (from leaf to root) is necessary for the propagation of backward RM cells [12].

queuing discipline for our proposed TIPS scheme as it facilitates the memory access [11]. We manage the buffer at the downstream node by the credits per port. Logically, if the credit is zero, the output port cannot emit the data cells on the link.
The data, finally, reaches at the leaf. At the leaf, following actions are performed.

1. When a data cell arrives, the leaf saves the EFCI state (per VC).
2. On receiving the forward RM cell, the leaf changes the cell direction and sends the RM cell back to the root. The RM cell fields shall be unchanged except:
   (a) If the saved EFCI state is set, then the leaf sets CI=1 in the RM cell and saved EFCI state to reset.
   (b) If the leaf has internal congestion then it may set CI=1 or NI=1.
3. If a leaf cannot transmit a waiting backward RM cell before it receives a subsequent forward RM cell to be turned around on the same VC, it overwrites the old RM cell information with the more recent information.

In present version of TIPS scheme, RM cell just serve as an indication for the source either to modify its rate or to continue transmitting with the current rate. We do not propose the idea of having explicit rate (ER) evaluation for multicast connection as it becomes very complex at branching nodes to evaluate a collective response of all the branches. Actually, having an $Arbiter$ at branching nodes gives us the convenience of performing efficiently the congestion control even with a simpler structured RM cell.
On the other hand, the branching nodes are in waiting mode (initiated by their respective $Timers$) for receiving backward RM cells. They expect to receive one RM cell per branch, from all the branches attached, during the interval the $Timer$ stays activated. If they do not receive RM cell from one or more branches among the attached ones, this is an indication of congestion on the silent branch(es). In this case, the branching node assumes that as if a RM cell with CI bit set is received from each silent branch. It is possible that a branching node receives two or more RM cells (with in the same $Timer$ interval) returned by the same leaf. In that case the recently received RM cell from the leaf will be considered for congestion evaluation. The branching node amalgams the information of all the received RM cells and it generates, at the termination of each $Timer$ triggered, one unique backward RM cell carrying the amalgamated information to the root. The RM cells information are amalgamated as follows:

- The branching node generates a RM cell with CI bit set if at least one cell, among the RM cells received, has CI bit set.
- The branching nodes generates a RM cell with NI bit set if at least one cell, among the RM cells received, has NI bit set and none of them has CI bit set.
- The branching node generates a RM cell with both CI and NI bits un-set if all of the RM cell received has both the bits un-set.

If the root is a branching node too, then an amalgamated RM cell is generated at the termination of $Timer_{root}$. If the backward RM cell, arrived/amalgamated at the root, has CI bit set then the root will decrease its rate by a reduction factor (RDF) which is specified at call setup. If it has NI bit set then the root will not modify its current emission rate. A backward RM cell, with both CI and NI bits un-set, will let the root to increase its rate by additive increase rate (AIR) specified at call setup.
We talked of an $Arbiter$ at branching nodes. Following section describes the need of such $Arbiter$ and its different functions.

3.3 The Branching Node

According to the definition of a branching node (cf. section 3), it is responsible for the replication of data cell on all the branches attached to it. Replications means disposing and maintaining a separate queue for each data copy created. The cells from these queues are forwarded on different links, more probably, having different speeds and leading to switches of diverse commutations speeds. This makes the role of branching node more important and influential. We take a simple example in the following paragraph to show that how complex and decisive role, a branching node is expected to have.
The figure 3 shows a part of point-to-multipoint connection where node B is a branching node. We concentrate our attention on the behavior of branching node B and are not bothered about the nature of other nodes. The node B receives the data cells from A and copies it on three channels to C, D and E. This result into three different VCs which are controlled by their RM cells independently. The data cells, coming from node A through a single VC, are served at different rates at the output ports of node B which, evidently, leads to have different queue lengths of replicated data cells. We assume that the branching node B per-

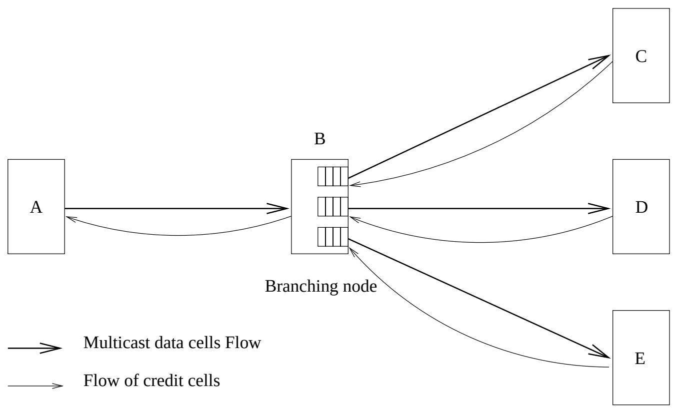

Figure 3: The data replication at the branching node

forms a static buffer allocation[12] and thus has disposed a certain buffer space to each queue of replicated data cells. Serving these queues at different rate, due to different link speeds or due to different traffic loads on branches, means liberating their buffer at different paces. For example, in the figure 3, the data cell queue for node C is being served at a slower rate. The buffers are liberated earlier by the other two queues at branching node B. Eventually, a situation arrives when the queue for node C is full whereas the other two queues are totally vacated. Any data cell arriving now can not be copied on all the branches. In this situation, branching node B has one of the following two options to react:

1. It delays the credit declaration to upstream node A until the occupied queue liberates buffers. It results the under-utilization of network available resources.
2. It drops the cells from the queue for node C and sends the credit. This timely sending of credit is at the cost of an additional cell loss.

[12] We may have the branching nodes doing dynamic buffer allocation but it is quite tedious to manage and the management complexity increases significantly as number of VC increases.

As on the output port of branching node B to node C, there are other VCs[13] (unicast or multicast) too, so an efficient service balance among these VCs can avoid the catastrophe caused by the blocked queue. This service balance, looking at the serving rate of other queues of a multicast connection, can favor or de-favor certain VCs and thus ensures that cells from replicated data queues are forwarded at a homogeneous rate. The $Arbiter's$ algorithm ensures a good service balance among the contending VCs on an output port while guaranteeing the best utilization of available resources. This algorithm is also capable of managing several multicast VCs on a single output port. This algorithm (refers to as an arbitration algorithm here after) is also equally efficient to manage good service balance among unicast queues only if there is no multicast queue at the output port. Additionally, it assures the fair allocation of resources among the contending VCs.

3.3.1 The Arbitration Algorithm

This section describes the above mentioned arbitration algorithm [5]. The credits per port (cf. section 3.2, figure 3) gives a certain amount of liberty to the arbitration algorithm to select a queue, among those at an output port, to be served for the available cell slot. In a branching node, the incoming cells of an input port are buffered in different queues. A queue corresponds to an output port to which are destined its cells regardless of their virtual channel identification/virtual path identification (VCI/VPI). All the multicast channels have to go through a multicast scheduler, for replication, before approaching the corresponding output port. At an output port, each unicast queue is assigned with a Normal Priority (NP) variable where as each multicast queue is assigned with the NP variable as well as a Multicast Priority (MP) variable. The NP and MP variables are calculated/updated at output port and at multicast scheduler respectively.

At the Multicast Scheduler: The multicast scheduler creates the required number of data copies of multicast connection data cells and buffer them in corresponding output queues. A MP value is evaluated for each queue of the multicast connection. The calculation of MP variable is as follows:

- If a queue length crosses the max_threshold value ("a" in fig 4), it attains positive MP value.
- If a queue length is shorter by the threshold_diff value ("b" in fig 4) than the longest queue of same multicast session, it attains negative MP value.
- If a queue falls in none of above two cases, its MP value is zero.

For a multicast connection, the values of max_threshold and threshold_diff parameters are selected such that:

max_threshold $>$ (the maximum possible queue length - threshold_diff)

At the Output Port: The cells from different queues are contending for the available cell slot. Among these queues, there may be some queues belonging to multicast connection along with those of unicast connections. The output port evaluates a NP value for each of its queues regardless of their nature (unicast or multicast). The NP variable is function of following parameters and is updated every cell slot:

1. The percentage of buffer occupied by the queue (queue length).
2. The number of times that a cell of the queue has been refused for the available cell slot.
3. The rate at which the queue length grows.
4. The parameters declared at the connection establishment time e.g. MCR, PCR, and cell delay variation tolerance (CDVT) etc.

[13]We talk of ABR traffic VCs only.

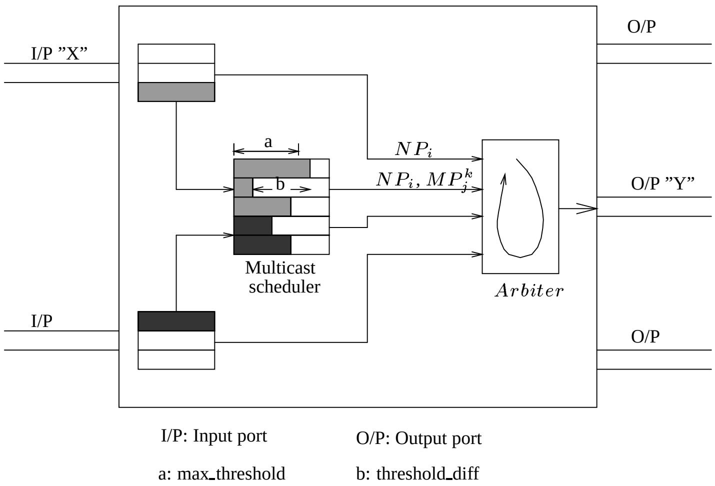

Figure 4: The implementation of arbitration algorithm in an ATM branching node.

5. The value of Cell Loss Priority (CLP) bit in ATM cell header.

We describe the $Arbiter's$ working scenario in following two categories as:

1. **Normal Condition**: When all the multicast queues at an output port have MP value zero or when there is no multicast queue at the output port, the $Arbiter$ picks the cell from a queue of the highest NP value and forward it in available cell slot. The unserved queues get their NP value updated accordingly.
2. **Particular Condition**: This condition occurs whenever one or more multicast queues tend to either overfill or underfill their buffers. The $Arbiter$ selects the multicast queue with the highest MP value and do as follows:
   (a) If the highest MP value is positive, the cell from this queue is forwarded in the available cell slot regardless of the queue NP value.
   (b) If the highest MP value is negative, the $Arbiter$ picks a queue with the highest NP value among the unicast queues only. A cell from this queue will be forwarded in the available cell slot.
   (c) If the highest MP value is zero, the $Arbiter$ selects a queue of highest NP value among the unicast queues along with the multicast queues with MP value zero. A cell from the selected queue is forwarded.

This way the $Arbiter$ decides dynamically the queue to be served and maintains a service balance between unicast and multicast queues. The description of the arbitration algorithm is given in the following section.

3.3.2 The Arbitration Algorithm Description

As presented in section 3.3.1, the arbitration algorithm, at a branching node, is functionally distributed at the multicast scheduler and at the output ports. The following description is with reference to figure 4.

Multicast Scheduler Behavior

If nms $>$ 0
Do k=1 ... nms
L_H^k = max(L_i^k where 1 $\leq$ i $\leq$ $ncms_k$)
If L_H^k $>$ max_threshold
MP_H^k = L_H^k - max_threshold
Do j= 1 ... $ncms_k$
If (j $\neq$ H)
If ((L_H^k - L_j^k) $>$ threshold_diff)
MP_j^k = threshold_diff - (L_H^k - L_j^k)
If ($\forall$ i, 1 $\leq$ i $\leq$ $ncms_k$ such that $Forward_i^k \geq 1$)
then $Forward^X$ = $Forward^X$ + 1

Output Port Behavior

If ($credit^Y$ $>$ 0 and n $>$ 0 and m $>$ 0)
a) If ($\exists$ i, 1 $\leq$ i $\leq$ m such that $MP_i^k \neq 0$)
then follow case "c".
else follow case "b".
b) NP_J = max (NP_i where 1 $\leq$ i $\leq$ n)
$credit^Y$ = $credit^Y$ - 1 /* a cell from VC_J is forwarded downstream */
If ($VC_J \, \epsilon \, S^Y$(U))
then $Forward^X$ = $Forward^X$ + 1
If ($VC_J \, \epsilon \, S^Y$(M))
then $Forward_J^k$ = $Forward_J^k$ + 1
Do i= 1 ... n
If (i $\neq$ J) then update NP_i
c) If m $>$ 1
MP_R^k = max (MP_i^k where 1 $\leq$ i $\leq$ m)
If m=1
R=m
If n $>$ m
If MP_R^k = 0
NP_J = max {max (NP_i such that $VC_i \, \epsilon \, S^Y$(U)),
max (NP_i such that $VC_i \, \epsilon \, S^Y$(M) and MP_i^k=0)}
$credit^Y$ = $credit^Y$ - 1 /* a cell from VC_J is forwarded downstream */
If ($VC_J \, \epsilon \, S^Y$(U))
then $Forward^X$ = $Forward^X$ + 1
If ($VC_J \, \epsilon \, S^Y$(M))
then $Forward_J^k$ = $Forward_J^k$ + 1
If MP_R^k $<$ 0
NP_J = max(NP_i such that $VC_i \, \epsilon \, S^Y$(U))
$credit^Y$ = $credit^Y$ - 1 /* a cell from VC_J is forwarded downstream */
$Forward^X$ = $Forward^X$ + 1
Do i= 1 ... n
If (i $\neq$ J) update NP_i
If (n=m or MP_R^k >0)
$credit^Y$ = $credit^Y$ - 1 /* a cell from VC_R is forwarded downstream */

$Forward_R^k$ = $Forward_R^k$ + 1
Do i= 1 ... n
If (i $\neq$ R) update NP_i
If ($credit^Y > 0$ and n > 0 and m = 0)
Follow case "b" described above

Variables and Parameters

- MP_j^k: multicast priority defined for j^{th} queue of k^{th} multicast connection.
- NP_i: normal priority defined for i^{th} queue at the output port.
- n: total number of queues present at an output port.
- m: total number of multicast queues present at an output port.
- nms: total number of multicast connections present in the multicast scheduler.
- $ncms_k$: total number of copies of k^{th} multicast connection's cells, created by multicast scheduler.
- L_j^k: length of j^{th} queue of k^{th} multicast connection.
- max_threshold: the maximum threshold value of queue length of a multicast connection in multicast scheduler.
- threshold_diff: the threshold difference of a queue length from the longest queues of the same multicast connection.
- $Forward^X$: number of cells forwarded by the port "X".
- $Forward_j^k$: number of cells forwarded by j^{th} queue of multicast session k.
- $credit^Y$: the remaining buffer of downstream node of output port "Y".
- S^Y(U): set of all the unicast queues present at the output port "Y".
- S^Y(M): set of all the multicast queues present at the output port "Y".

4 Conclusion

The paper discusses multicast connections in ATM networks. It presents, briefly, simple (point-to-multipoint) and multiple (multipoint-to-multipoint) multicast connections. We focus our attention on simple multicast connections in ABR traffic class. We propose a novel method to control efficiently the congestion control for simple multicast ATM connections.

Our proposed technique has two important features. The first one is the $Timer$ which is triggered at the root and the branching nodes and the second one is the $Arbiter$ implemented at the branching nodes only. These features enhances the performance of TIPS scheme. The use of $Timer$ facilitates the branching nodes to determine the time instant of generating and sending back a RM cell whose value is specific to each branching node. The $Arbiter$ works on the basis of priority(ies) (normal and/or multicast priority(ies)) assigned to each queue at an output port and ensures a service balance between them. This leads to the best utilization of network available resources and helps the source rate to fluctuate less before reaching the stability.

The simulation of TIPS scheme is under progress which will help to determine exactly its different parameters and to evaluate its performance. This evaluation will then be compared with the performance of existing methods.

References

[1] Rainer Händel, Manfred N. Huber, Stefan Schröder. ATM Networks, Concept, Protocols, Applications. *Second Edition 1995, Addison-Welsey Pub. Co., pp. 1-19.*

[2] Raj Jain. Congestion Control and Traffic Management in ATM Networks: Recent Advances and a Survey. *ATM Forum/95-0177, Oct. 1995.*

[3] Pierre Rolin. Rseaux Haut Dbit. *Edition Herms, Paris, 1995*.

[4] Flavio Bonomi, Kerry W. Fendick. The Rate-Based Flow Control Framework for The Available Bit Rate Service. *IEEE Network, March/April 1995.*

[5] M. Tufail, B. Cousin. ABR Congestion Control Schemes and the Multicast Problem. *The first NDSU Workshop on ATM Networking, Fargo, USA, August 7-9 1996.*

[6] K. K. Ramakrishnan, R. Jain. A Binary Feedback Scheme for Congestion Avoidance in Computer Networks. *ACM Trans. Comput. Sys., vol. 8, no. 2, 1990, pp. 158-181.*

[7] K. K. Ramakrishnan, Peter Newman. Integration of Rate and Credit Schemes for ATM Flow Control. *IEEE Network, March/April 1995.*

[8] Hong-Yi (Henry) Tzeng, Kai-Yeung (Sunny) Siu. Enhanced Credit-Based Congestion Notification (ECCN) Flow Control for ATM Networks. *ATM Forum/94-0450, May 1994.*

[9] ITU-T Recommendation I.371. Traffic Control And Congestion Control in B-ISDN. *March 1993.*

[10] H. T. Kung Harvard University. The FCVC (Flow Controlled Virtual Channels) Proposal for ATM Networks. *Proc. International Conf. on Network Protocols, San Francisco, California, Oct. 1993, pp. 116-127, Modified April 8, 1994.*

[11] Martin de Prycker. Asynchronous Transfer Mode solutions for broadband ISDN. *Ellis Horwood Limited, Edition 1992.*

[12] ATM Forum Management Specification Version 4.0. *ATM Forum/95-0013R6, June 7,1995.*

[13] Kai-Yeung Siu, Hong-Yi Tzeng. Congestion Control for Multicast Service in ATM network.*Dept. of Electrical & Computer Engineering, University of California, Ivrine. Technical Report Number: 94-03-01.*

[14] A. Romanov. A Performance Enhancement for Packetized ABR and VBR+ Data. *AF-TM 94-0425, March 1994.*

[15] M. Rabah, B. Cousin. Le contrôle de congestion dans le réseau ATM pour les applications de téléconférence. *DEA project report, ADP/IRISA Rennes, Sept 96.*

[16] P. Barri, G. De Smet. Distributed Connection Control and Maintenance for an ATM Multistage Interconnection Network. *IEEE International Conference on Communication ICC91.*