****

ATM Forum Document Number: ATM Forum/98-0150

****

Title: Proposed modifications to the baseline text and living list on multipoint ABR behavior

****

Abstract:
We propose modifications to the baseline text and living list on ABR flow control for multipoint connections. Modifications are proposed to section 5.10.8 of the traffic management specifications TM 4.0, and an appendix on implementation examples for multipoint ABR is proposed. In addition, changes are suggested for the living list items on ABR multipoint connections.

****

Source:
Sonia Fahmy, Raj Jain, Rohit Goyal, and Bobby Vandalore
The Ohio State University
Department of Computer and Information Science

Raj Jain is now at Washington University in Saint Louis, jain@cse.wustl.edu http://www.cse.wustl.edu/~jain/

This presentation of this work at the ATM Forum is sponsored by NASA Lewis Research Center.

****

Date: February 1998

****

Distribution: ATM Forum Technical Working Group Members (AF-TM)

****

****

1   Introduction

ATM multicast capabilities are essential for the support of many applications, including LAN emulation and IP multicasting over ATM. UNI 3.1 and 4.0 define point-to-multipoint and leaf-initiated join (LIJ) capabilities. In addition, multipoint-to-point and multipoint-to-multipoint connection support are being discussed at the signaling and PNNI groups, and they are expected to be specified in the near future.

Although a number of contributions [3, 4, 5, 6, 8, 9, 11, 14, 15] and papers [1, 2, 10, 12, 16] have investigated ABR flow control for multipoint connections, the current traffic management specifications document [7] only contains a brief outline of point-to-multipoint ABR behavior in section 5.10.8.

In this contribution, we propose a number of modifications and additions to the baseline text and living list, with respect to ABR multipoint flow control. In the next section, we propose additions to the text of section 5.10.8 of the specifications. Then, we propose the addition of a subsection dealing with ABR multipoint implementation examples in Informative Appendix I. Finally, we propose changes to the living list.

2   [Motion 1] Additions to Section 5.10.8 of TM 4.0

We first provide the reasoning behind our proposal, and then give the proposed modifications.

2.1   Motivation

In point-to-point ABR flow control, the source is controlled to the minimum rate that can be supported by all the switches on the path from the source to the destination [7]. The natural extension of this strategy to point-to-multipoint connections is: controlling the source to the minimum rate that can be supported by the switches on the path from the source to *all of the leaves* in the multicast tree. This is because the minimum rate is the technique most compatible with the typical data requirements: no data should be lost, and the network can take whatever time it requires to deliver the data intact [14]. If the source sends at a rate greater than the rate that can be supported by a branch, then the leaves of that branch will not receive all the cells sent by the source.

The operation of feedback consolidation employed at branch points can be explained with the aid of figure 1. The consolidation operation is required to avoid the feedback implosion problem, where the number of backward RM cells received by the source is proportional to the number of leaves in the multicast tree. In addition, the source allowed cell rate (ACR) should not keep fluctuating due to the varying feedback received from different leaves.

Consolidation noise can occur when feedback from some leaves is not always received in a timely fashion at the time when the RM cells need to be returned by the branch point. Alleviating the consolidation noise problem (by ensuring feedback is received from all leaves before sending a BRM cell) may create additional problems, such as slow transient response and additional complexity.
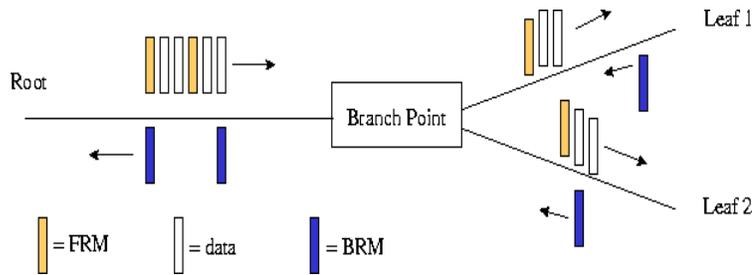
Figure 1: Point-to-multipoint connections

2.2 Proposed Text

The following text should be added at the end of the text for the second item in section 5.10.8.2:

One method of consolidating information from BRM cells is to assign the ER field in returning RM cells to the minimum of the ER values indicated by the branches, the CI to the OR of the indicated CI values, and the NI to the OR of the NI values.

Over the long run, the branch point should send (close to) one BRM cell corresponding to every FRM cell it receives. This can be done by maintaining a flag to indicate the receipt of an FRM cell from the root, or flags to indicate the receipt BRM cells from the branches, or a counter to denote the difference between the number of FRM cells received and BRM cells sent. In addition, the branch point should ensure that each BRM captures feedback from most branches, while also ensuring that transient response to overloads is not excessive because the branch point is waiting for feedback from the various branches.

The BRM consolidation method at the branch points needs to:

1. Scale well with the number of levels and with the number of branches in the multicast tree.

2. Ensure that the ratio of BRMs to FRMs in the network and at the root is maintained close to one.

3. Handle non-responsive branches such that they do not halt the consolidation operation nor cause overload or underload.

4. Exhibit: (a) minimal consolidation noise and consolidation delays, (b) fast transient response, (c) low complexity.

3 [Motion 2] Informative Appendix I.9 for TM Specifications: Multipoint ABR Flow Control

We propose the addition of a subsection dealing with ABR multipoint in Informative Appendix I (which gives ABR implementation examples) as follows.

I.9 Multipoint ABR Flow Control

3

Sample branch point algorithms for handling point-to-multipoint connections, and sample merge point algorithms for multipoint-to-point connections are given in this appendix. Multipoint-to-multipoint connections can be handled by combining a point-to-multipoint (branch point) algorithm with a multipoint-to-point (merge point) algorithm.

I.9.1 Sample Branch Point Algorithms

A branch point replicates cells from the root to each branch in the responding state and consolidates their feedback. Sample consolidation algorithms are given next.

In a simple point-to-multipoint ABR algorithm [14][1], the minimum explicit rate indicated by the BRM cells received from the branches is maintained, say as MER. Whenever an FRM cell is received, it is multicast to all branches, and a BRM is returned using the MER value for the BRM explicit rate. MER is then set to PCR. A simple enhancement to reduce noise in this algorithm is to only generate the BRM cell if a BRM has been received from at least one leaf after the last BRM was sent by the branch point [16].

To reduce the complexity of the algorithm, some of the backward RM cells generated by the destinations can be forwarded, instead of turning around the RM cells at the branch points. Whenever an FRM cell is received at a branch point, the algorithm simply sets a flag indicating the receipt of the FRM cell, and multicasts it to all branches. When a BRM cell is received from a branch, it is passed back to the source (after using the minimum allocation), only if the flag was set. The flag and the MER register are then reset [10].

To reduce consolidation noise, the BRM cell can only be passed back when BRM cells from *all* branches have been received after the last feedback. This can be easily implemented by maintaining a separate flag for each branch to indicate if a BRM cell has been received from the branch after the last BRM cell was sent. It is necessary to handle the possible non-responsiveness of a branch by implementing timeouts in this algorithm. In addition, the transient response of this algorithm may be slow due to waiting for feedback from possibly distant leaves. This delay should be avoided when a severe overload situation has been detected. In this case, there is no need to wait for feedback from all the branches, and the overload should be immediately indicated to the source [4]. In the cases when the branch point is itself a switch and queuing point, the branch point can invoke the switch scheme whenever a BRM is received, and not just when a BRM is being sent. Hence, overload at the branch point itself can be detected and indicated according to the fast overload indication idea.

The fast overload indication idea may increase the BRM cell overhead, since the ratio of source-generated FRM cells to BRM cells received by the source can exceed one. To alleviate this problem, a counter (maintained for each multipoint VC) can be incremented whenever a BRM cell is sent before feedback from all branches has been received. When feedback from all branches indicates underload, and the value of that counter is more than zero, this particular feedback can be ignored and the counter decremented [4].

I.9.2 Sample Merge Point Algorithms

Merge points must ensure that BRM cells are sent to the appropriate sources at the appropriate times. These algorithms should maintain the BRM to FRM ratio at the sender and inside the network close to one. They should be simple, scalable, and minimize noise and delays. With

---

[1]References may be removed in the specifications.

multipoint-to-point and multipoint-to-multipoint connections, the implicit assumption that each connection has only one source is no longer valid.

I.9.2.1 Fairness Definitions

Four different types of fairness can be defined for multipoint-to-point and multipoint-to-multipoint connections:

1. **Source-based fairness,** which divides bandwidth fairly among active sources as if they were sources in point-to-point connections, ignoring group memberships.

2. **VC/source-based fairness,** which first gives fair bandwidth allocations at the VC level, and then fairly allocates the bandwidth of each VC among the active sources in this VC.

3. **Flow-based fairness,** which gives fair allocations for each active flow, where a flow is a VC coming on an input link. Formally,
   $NumFlows_j, j \in OutputPorts =$
   $\forall i, i \in InputPorts, \sum_i$ Number of VCs coming on port i and being switched to port j

4. **VC/flow-based fairness,** which first divides the available bandwidth fairly among the active VCs, and then divides the VC bandwidth fairly among the active flows in the VC.

I.9.2.2 Sample Merge Point Algorithm for Source-Based Fairness

An example merge point algorithm for source-based fairness can operate as follows [13]. The algorithm maintains a flag at the merge point for each of the flows being merged. The flag indicates that an FRM has been received from this flow after a BRM had been sent to it. Therefore, when an FRM is received at the merge point, it is forwarded to the root and the flag is set. When a BRM is received at the merge point, it is duplicated and sent to the branches that have their flag set, and the flags are then reset.

Switch Scheme Restrictions for VC Merge Switches:

Source-based fairness algorithms operating in VC merge switches need to consider the following issues:

1. Per source accounting should not be performed. For example, measuring the rates or activity for each source, or distinguishing overloading and underloading sources should not be performed. The algorithm can use the information supplied in RM cells, in addition to aggregate measurements such as load, capacity and queuing delays. If accounting is performed at the VC level or at the flow level, an additional mechanism to divide VC or flow bandwidth among sources is necessary.

2. CCR values from BRM cells should not be used in computing rate allocations for sources in multipoint connections, since the CCR value can be that of another source that does not go through the switch performing the computation. CCR values from FRM cells can be used to compute rate allocations for sources in multipoint connections, even though the CCR used to compute the rate for a source may not actually be the CCR value of the source. The maximum CCR value seen during an interval can also be used instead of the CCR of the source.

4 [Motion 3] Proposed Modifications to Item 96-004 of LTD-TM (Parameter Negotiation for Multipoint-to-Multipoint ABR Service)

The source of the baseline text for this item is contribution 97-0832, which is more concerned with fair bandwidth allocation, than with parameter negotiation. A separate item for flow control for multipoint-to-multipoint connections should be added as follows:

- *Title:* Flow control for ABR multipoint-to-multipoint connections

- *Problem Statement:* Define the desirable forms of fairness for multipoint connections, and extend current switch algorithms for multipoint connections. Conduct a performance analysis to examine the fairness, complexity, overhead, transient response, delays, and scalability tradeoffs involved. Interoperability must also be studied.

- *Solution Requirements:* Fairness, low overhead, fast response, scalability.

- *Item Introduced:* February 1998

- *Last Updated:* February 1998

- *Current Status:* Under Study

- *Other Working Groups:* SIG, PNNI

- *Contribution Log:* 97-0832, 97-1085R1, 98-0150

- *Work To Be Done:*

- *Baseline Text:* [Same as current text for 96-4 in Part II of LTD-TM-01.07].

References

[1] D. Cavendish, S. Mascolo, and M. Gerla. Rate based congestion control for multicast ABR traffic. In *Proceedings of IEEE GLOBECOM '96*, volume 2, pages 1114–1118, November 1996.

[2] You-Ze Cho and Myong-Yong Lee. An efficient rate-based algorithm for point-to-multipoint ABR service. In *Proceedings of the IEEE GLOBECOM '97*, November 1997.

[3] Sonia Fahmy, Raj Jain, Rohit Goyal, and Bobby Vandalore. A switch algorithm for abr multipoint-to-point connections. ATM Forum/97-1085R1, December 1997.

[4] Sonia Fahmy, Raj Jain, Rohit Goyal, Bobby Vandalore, Shivkumar Kalyanaraman, Sastri Kota, and Pradeep Samudra. Feedback consolidation algorithms for ABR point-to-multipoint connections. ATM Forum/97-0615, July 1997. Extended version to appear in the Proceedings of IEEE INFOCOM '98.

[5] Sonia Fahmy, Raj Jain, Rohit Goyal, Bobby Vandalore, Sastri Kota, and Pradeep Samudra. Fairness for ABR multipoint-to-point connections. ATM Forum/97-0832, September 1997.

[6] Sonia Fahmy, Raj Jain, Shivkumar Kalyanaraman, Rohit Goyal, Bobby Vandalore, Xiangrong Cai, and Seong-Cheol Kim. Performance analysis of ABR point-to-multipoint connections for bursty and non-bursty traffic with and without VBR background. ATM Forum/97-0422, April 1997.

[7] The ATM Forum. The ATM forum traffic management specification version 4.0. ftp://ftp.atmforum.com/pub/approved-specs/af-tm-0056.000.ps, April 1996.

[8] Eric Gauthier, Jean-Yves Le Boudec, and D. Dykeman. SMART: A many-to-many multicast protocol for ATM. ATM Forum/96-1295, October 1996.

[9] Doug Hunt. Open issues for ABR point-to-multipoint connections. ATM Forum/95-1034, August 1995.

[10] Wenge Ren, Kai-Yeung Siu, and Hiroshi Suzuki. On the performance of congestion control algorithms for multicast ABR service in ATM. In *Proceedings of IEEE ATM'96 Workshop, San Francisco*, August 1996.

[11] Wenge Ren, Kai-Yeung Siu, and Hiroshi Suzuki. Performance evaluation of multipoint-point ABR and UBR. ATM Forum/96-1402, October 1996.

[12] Wenge Ren, Kai-Yeung Siu, and Hiroshi Suzuki. Multipoint-to-point ABR service in ATM networks. In *Proceedings of the International Conference on Communications, ICC'97, Montreal*, June 1997.

[13] Wenge Ren, Kai-Yeung Siu, Hiroshi Suzuki, and Masayuki Shinihara. Multipoint-to-multipoint ABR service in ATM. *Submitted to the Journal of Computer Networks and ISDN Systems*, 1997.

[14] Lawrence Roberts. Rate based algorithm for point to multipoint ABR service. ATM Forum/94-0772R1, November 1994.

[15] Lawrence Roberts. Point-to-multipoint ABR operation. ATM Forum/95-0834, August 1995.

[16] Kai-Yeung Siu and Hong-Yi Tzeng. Congestion control for multicast service in ATM networks. In *Proceedings of the IEEE GLOBECOM '95*, volume 1, pages 310–314, 1995.

All our papers and ATM Forum contributions are available through http://www.cis.ohio-state.edu/~jain/