EFFECTIVE FLOW CONTROL FOR ABR MULTIPOINT-TO-MULTIPOINT SERVICES

Uyen Trang Nguyen

Department of Computer Science, York University, Toronto, Ontario, Canada, M3J 1P3
Email: utn@cs.yorku.ca

ABSTRACT

Although many flow control schemes have been proposed for ABR (Available Bit Rate) multipoint-to-multipoint (mp-mp) services in ATM (Asynchronous Transfer Mode) networks, several issues remain to be resolved: bandwidth allocation definitions (BADs), and the effectiveness of those schemes. No consensus has been reached concerning the issue of criteria for allocating ABR bandwidth among multipoint connections, or between point-to-point and multipoint connections. On the other hand, existing mp-mp flow control schemes implement only one BAD, namely source-based allocation. This inflexibility can limit the application of those schemes. Moreover, our simulation results will show that they suffer from the problems of link under-utilization and/or non-compliance with the selected BAD. We propose a mp-mp flow control algorithm, the FCAM[1] algorithm, that supports a large set of BADs, provides correct rate allocations with fast convergence, and maximizes link utilization. Experimental results are presented to demonstrate the above advantages of the FCAM algorithm.

1. INTRODUCTION

Multipoint communications support the exchange of information among a set of participants in an efficient manner. Multipoint-to-multipoint communications encompass an important class of applications whose examples are synchronization of replicated databases, audio/video conferencing, distributed interactive games, and distance education.

ABR is a best-effort service with no bandwidth guarantee, except a minimum cell rate. The bandwidth left over after CBR (Constant Bit Rate) and VBR (Variable Bit Rate) usage is divided evenly among ABR sources. ABR traffic is regulated using explicit rate (ER) control as follows [1]. Resource management (RM) cells are generated periodically by ABR sources, sent from the source to the destination (downstream) as forward RM (FRM) cells, and turned backed by the destination to the source (upstream) as backward RM (BRM) cells. On the way back, a BRM cell may have its field ER modified by switches on the path from the destination to the source for ER control. Upon receiving a BRM cell, the source adjusts its transmission rate accordingly using the value in field ER (BRM.ER).

Although there exist many mp-mp flow control schemes [2, 13, 15, 19], several issues remain to be resolved: bandwidth allocation definitions (BADs), and the effectiveness and performance of those schemes. Bandwidth allocation definitions refer to criteria for allocating ABR bandwidth among multipoint connections, or between unicast (point-to-point) and multipoint connections. For example, should a source of a multipoint connection with 100 sources be allocated the same rate as a unicast source, or just 1/100th of the rate of a unicast source? Several BADs have been suggested for multipoint communications [5, 10, 12]. However, researchers have not yet agreed on which definitions are most suitable for ABR multipoint applications. Therefore, a mp-mp flow control scheme should support several BADs, i.e., be *flexible*, so that switch vendors can configure their switches quickly and easily to meet clients' requirements.

A mp-mp flow control scheme is *effective* if it allocates ABR bandwidth to sources according to the selected BAD and maximizes the utilization of ABR capacity. Desirable *performance measures* of a multipoint flow control scheme are fast convergence and minimal rate oscillations.

In this paper, we first present a simulation-based evaluation of the existing mp-mp flow control schemes. The evaluation indicates that those schemes suffer from the problems of link under-utilization and/or non-compliance with the selected BAD in many cases. Moreover, they implement only one BAD, namely source-based allocation, which divides ABR bandwidth evenly among sources, unicast or multipoint. This inflexibility can limit the application of those schemes. We then propose a new mp-mp flow control algorithm, the FCAM algorithm, that is effective and supports a large set of BADs. Experimental results are presented to demonstrate the advantages of the FCAM algorithm: correct rate allocations with fast convergence, maximum link utilization, and minimal rate oscillations.

The ABR service was originally proposed for use in ATM networks. However, its congestion controlled traffic characteristic makes it an attractive service for use in high-performance IP networks as well, in order to promote fairness towards connections using TCP, the dominant Internet transport protocol [16]. In fact, Sisalem and Schulzrinne have proposed an ABR-like service for the Internet called Adaptive Load Service (ALS) whose traffic control mechanism is similar to that of ABR [16]. Although the mp-mp ABR flow control scheme we propose in this paper is implemented for ATM networks, its design is applicable to various networks that support rate-based feedback control.

The paper is organized as follows. In section 2, we discuss the issues involved in the design of a mp-mp flow control scheme. Section 3 presents related work and our evaluation of the existing mp-mp flow control schemes. The proposed FCAM algorithm is described in section 4, followed by experimental results. Section 5 concludes the paper.

2. DESIGN ISSUES OF MP-MP FLOW CONTROL

One straightforward approach to implement a mp-mp group with N members is to build N p-mp (or mp-p) physical connections, each having one source (receiver) and $N-1$ receivers (sources) [2].

[1] Flow Control for Abr Multipoint-to-multipoint services

This approach, however, is not scalable as N grows. It is desirable to implement a mp-mp group as a single VC for scalability and easy maintenance [13]. In addition, the following three design issues must be addressed [7, 12]:

(1) Bandwidth Allocation Definitions (BADs)

For unicast connections, the most commonly used BAD is max-min "fairness" [8]. The BAD becomes more complicated when mp-mp connections are involved since such a mp VC is shared by several sources. The suggested BADs based on sources are:

− **Source-based** [5]: divides ABR bandwidth on a link evenly among active *sources* sharing the link, ignoring whether a source belongs to a unicast or a mp VC.

− **VC/source-based** [5]: first allocates the ABR bandwidth evenly among the VCs, unicast or multipoint, and then divides the allocated bandwidth of each VC evenly among the active *sources* in the VC.

− **Weighted-source-based** [10]: requires every source to be assigned a weight. The weight of any unicast source is 1. The weight of a source of a mp connection i is w_i, where $0 < w_i \leq 1$. For two different mp connections i and j, we can have $w_i \neq w_j$. ABR bandwidth is allocated to the sources proportionally to their weights. Note that when $w_i = 1$, we have source-based allocation. Weighted-source-based definition provides a compromise between the above two extreme definitions.

There are also BADs based on flows [5, 12]. However, these BADs suffer from the beat-down problem: the farther away a source from the destination in terms of the number of merge points, the smaller the rate it is allocated [5]. We thus do not consider BADs based on flows in this paper.

(2) Rate Calculation Algorithm (RCA)

A unicast RCA such as ERICA [9] or FMMRA [4] can be extended to support multipoint connections. In fact, ERICA has been modified and applied in many mp-p and mp-mp flow control schemes [7, 10, 15]. The resulting mp RCA must take into account the selected BAD to ensure that it allocates bandwidth correctly as dictated by the BAD.

(3) Processing of RM Cells

There are two issues to consider: feedback implosion and feedback consolidation.

If a mp-mp group has N members, an FRM cell will be duplicated $N-1$ times. When the FRM cell is turned around and travels back to the sources as a BRM cell, it would be duplicated again $N-1$ times. To prevent the problem of *feedback implosion*, at a branch point, a switch will send a BRM upstream along a branch only if the switch has received at least one FRM cell from that branch since the last BRM cell was sent along the branch [7, 12, 15, 19].

A mp-mp group with N members can be viewed as N logical p-mp trees, each containing one sender and $N-1$ receivers. In such a logical p-mp tree, the allowed cell rate (ACR) of the source at the root should be less than or equal to the lowest rate that can be supported among all the branches of the p-mp tree. Otherwise, a higher rate would cause congestion, and thus data loss, on the branch with the lowest capacity. Therefore, a switch should consolidate ER values in BRM cells from the fan-out branches and select the minimum value before sending a BRM cell back to the source. This procedure is called *feedback consolidation*. However, when an overload situation is detected, a BRM cell with overload information should be sent immediately to the source, without waiting for feedback from all the branches, in order to alleviate the congestion as quickly as possible [6].

3. SURVEY AND EVALUATION OF EXISTING MP-MP FLOW CONTROL SCHEMES

Cavendish and Gerla propose a mp-mp flow control scheme, the CG[2] algorithm [2], that implements a mp-mp connection with N members as N p-mp VCs, each having one sender and $N-1$ receivers. This scheme is thus not scalable as N grows.

The scheme proposed by Ren *et al.*, RSSS algorithm [19], implements a mp-mp connection as a single VC, and is the combination of a p-mp flow control algorithm [17] and a mp-p algorithm [18]. The mp-p scheme does not provide a specific multipoint RCA. Instead, the authors claim that any unicast RCA can be applied directly to their mp-p scheme without modifications. However, if data forwarding uses VC merge and the unicast RCA uses per-source accounting (e.g., the ERICA algorithm), the mp-p algorithm will not allocate ABR bandwidth correctly, as previously demonstrated by simulation results [10]. The resulting mp-mp scheme is thus not effective.

Other mp-mp flow control schemes include the MPER [13] and Pao's [15] algorithms. The main differences between these schemes are the BADs implemented and the RCAs. Table 1 lists the BADs supported by and other characteristics of these flow control schemes.

The RCA of the MPER scheme requires VC weights to be calculated and maintained. The weight of any unicast VC is 1. The weight W_k of a mp VC k on a link L is $W_k = N_k \times w_k$, where w_k is the weight of a source of VC k, and N_k is the number of sources of VC k merging on link L. The unicast RCA is modified so that each VC is offered an amount of bandwidth proportional to its weight. At a logical mp-p merge point, the bandwidth of VC is then divided among the merging branches using the branch weights [13]. The weight of a branch j of a mp VC k is defined as $w_{j,k} = n_{j,k} \times w_k$, where $n_{j,k}$ is the number of sources of VC k merging on branch j. Each branch is allocated an amount of bandwidth proportional to its weights: $branch\_ER_j = VC\_ER_k \times w_{j,k}/W_k$. Our simulation results presented next show that RCAs using VC weights suffer from the problems of link under-utilization and/or non-compliance with the selected BAD in many cases.

We used the ATM network simulator version 4.1 developed by the American National Institute of Standards and Technology (NIST) [11], and added code to implement the MPER flow control scheme. The simulation was conducted on the configuration depicted in Figure 1. In this configuration, the mp-mp connection consists of four sources S_1, S_2, S_3 and S_4 that exchange data with each other, and the unicast source S_u sends data to the destination D_u. We use the following setting in the experiments presented in this paper. All the links have a capacity of 150 Mbps. The distance between switches is 1000 km. Terminal-switch distance is 1km. VC merge is simulated, and the packet size is 512 bytes (maximum TCP segment size). The sources are persistent, have an initial cell rate (ICR) of 8 Mbps, and start sending data at the same time. The duration of each experiment is 1000 msec. Given the above configuration and link bandwidths, Table 2 lists the sources' rates for different BADs. In every experiment, we recorded the ACRs of the sources and the utilization of links L_{1b} and L_{3a}, the most bottlenecked links in the given configuration.

The following experiment, which simulates source-based BAD,

[2] Named using the first letters of the authors' last names. We will use this naming convention throughout the paper to denote different mp-p and mp-mp flow control schemes.

Characteristic	CG	RSS	MPER	Pao's	FCAM
A single mp-mp VC	No	Yes	Yes	Yes	Yes
Base unicast RCA	(any)	(any)	FMMRA	ERICA	ERICA*
MP-MP RCA	(n/a)	(n/a)	using VC weights		using source CCRs
Source-based	x	x	x	x	x
VC/source-based					x
Weighted-source-based					x

ERICA* is based on the ERICA algorithm, but all the steps that need per-source accounting are eliminated in order to support VC merge.

Table 1. Characteristics of the existing multipoint flow control algorithms

Bandwidth Allocation Definition	S_1	S_2	S_3	S_4	S_u
Source-based	37.50	37.50	37.50	75.00	37.50
VC/source-based	25.00	25.00	25.00	100.00	75.00
Weighted-source-based (MP source weight = 0.75)	34.62	34.62	34.62	80.76	46.15

Table 2. ACRs (in Mbps) of the sources in Figure 1

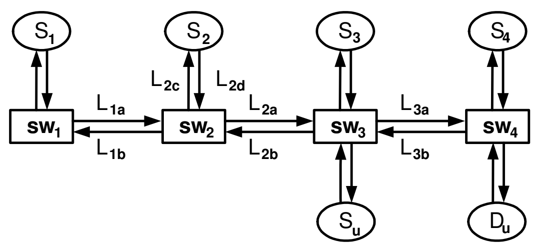


Fig. 1. Configuration for experiments

illustrates the problem of link under-utilization of the MPER scheme. The graph of source ACRs in Figure 3 obtained from simulation indicates that S_4 transmits at 50 Mbps, instead of 75 Mbps according to source-based allocation (see Table 2). As a result, the utilization of link L_{1b} varies around 83%, instead of being 100% had S_4 been able to claim the unused bandwidth from S_2 and S_3 on link L_{1b}. The reason for this under-utilization is that switch sw_2 is not aware of the bottleneck on link L_{3a} and cannot re-allocate the unused bandwidth from S_2 and S_3 to S_4.

The RCA by Pao [15] attempts to alleviate the problem of link under-utilization by dividing the bandwidth at merge points using a combination of weights and aggregate CCRs (current cell rates) of the branches. The aggregate CCR of a branch is the sum of the CCRs of the sources (belonging to the mp VC) whose traffic merging on this branch. The ER of a branch is computed using two ER values: one proportional to the weight of the branch, and the other, to the aggregate CCR of the branch, as follows: $branch\_ER_j = VC\_ER_k \times (w_{j,k}/W_k + branch\_CCR_{j,k}/VC\_CCR_k)/2$.

Our simulation results show that using both branch weights and aggregate CCRs indeed helps improve link utilization. However, it still does not ensure the maximum possible link utilization due to the partial use of branch weights for dividing the VC ER. This is illustrated by the following experiment that was run on the same configuration in Figure 1 with source-based allocation. The ACR graph in Figure 4(a) shows that the ACR of S_4 is higher than in the previous experiment, at 59 Mbps, but still does not reach the optimal 75 Mbps. The utilization of link L_{1b} concentrates in the range of 80% - 92%, instead of being 100%.

Using *only* aggregate CCRs of branches for dividing the VC ER would allow link utilization to be maximized. However, the rate allocation may not comply with the selected BAD in many cases, as shown in our previous work [14].

There also exist multipoint-to-point RCAs that use VC weights [10, 12]. Similarly to the above mp-mp RCAs, these mp-p RCAs are also affected by the problems of link under-utilization and/or non-compliance with the selected BAD in many cases, as proven by previous simulation results [14]. These mp-p RCAs thus cannot be extended in any way to support mp-mp connections.

4. THE FCAM FLOW CONTROL SCHEME

To give effective rate allocations, our FCAM flow control scheme uses an RCA based on source CCRs and not VC weights. It supports any of the existing BADs based on sources: source-based, VC/source-based and weighted-source-based.

4.1. Key Ideas and Novel Features

The FCAM flow control scheme treats a mp-mp group as a single VC for scalability and easy maintenance. RM cells are processed as described in section 2, i.e., with feedback implosion avoidance and feedback consolidation to minimize traffic overheads, rate oscillations and cell loss rates.

Our RCA is based on the method of using source CCRs adapted from the ERICA algorithm and first used in the FJ algorithm, a mp-p flow control scheme proposed by Fahmy *et al.* [7]. At the end of every measurement interval, the switch calculates the current *load factor* of a link, which is the ABR input rate divided by the ABR capacity of the link (pseudo-code in Figure 2(d)). For every output link L, the switch maintains the CCR of every VC (carried in FRM cells), as well as the maximum ER allocated in the previous interval, $max\_ER\_previous\_interval$. Depending on the current value of the load factor on link L, every VC must scale up or down its CCR by the load factor in order to maintain an optimal load of 1 on the link (Figure 2(b), lines $\{4\}-\{5\}$). In addition, if the load factor is less than $1 + \delta$ (underload), a VC can raise its rate to $max\_ER\_previous\_interval$ to be able to catch up with the VC having the highest rate. The value of $max\_ER\_previous\_interval$ thus serves as a "reference point" for sources to aim at. If the VC is a mp-mp VC, there will be different CCRs from various sources of the VC on the same output link. In that case, we select the maximum of the CCR values seen in the current interval; the pseudo-code is given in Figure 2(a). This helps reduce rate fluctuations [7].

Previous simulation results [7, 14] show that RCAs based on source CCRs are effective in terms of correct bandwidth allocation and link utilization. The FJ algorithm, however, supports only source-based allocation, and only for mp-p connections. Moreover, it suffers from prolonged rate fluctuations in many cases [14]. We propose an RCA using source CCRs without the problem of excessive rate oscillation. Specifically, our contributions include: (1) minimizing rate fluctuations in the FJ algorithm, (2) supporting source-based, VC/source-based and weighted source-based, and (3) applying the source CCRs approach to mp-mp flow control.

The reason for prolonged rate oscillations in the FJ algorithm is that the ER of a VC is calculated several times while the load factor is calculated only once *per measurement interval* [7]. If the load factor is not exactly one, the ER values will keep increasing or decreasing during that interval, causing unnecessary oscillations in the subsequent interval. To solve this problem, we permit a switch to compute *only one* ER feedback value per VC per measurement interval. The new code is outlined in Figure 2(b), lines $\{1\}-\{3\}$.

We employed the idea of a "reference point" to support multipoint sources with weights. Since sources of different multipoint VCs carry different weights, their ERs should not be used as $max\_ER\_previous\_interval$, the "reference point". Instead, ERs of unicast VCs are considered, because all unicast sources have the same weight of 1. Therefore, switch variables $max\_ER\_previous\_interval$ and $max\_ER\_current\_interval$ of every output port are maintained with respect to unicast sources. When the ER of a mp-mp source is calculated, variable $max\_ER\_previous\_interval$ is scaled by the weight of the source (recorded in variable $source\_weight[k]$), as shown in the algorithm in Figure 2(b), lines $\{7\}$ and $\{11\}$. A BAD is implemented by setting variable $source\_weight[k]$. For instance, if the mp-mp VC m in Figure 1 has $source\_weight[m] = 1$, the rate allocation will be source-based; if $source\_weight[m] = 1/3$, that is VC/source-based BAD.

The new RCA is then incorporated into the mp-mp flow control scheme as shown in Figure 2. The following section explains the FCAM algorithm in more detail.

4.2. Protocol Description

A switch in the routing tree of a mp-mp VC can always view itself as either a logical p-mp branch point or a logical mp-p merge point. For example, in Figure 1, when a cell arrives at switch sw_2 from link L_{1a}, it is duplicated and sent out along links L_{2a} and L_{2c}: this is a p-mp cell forwarding operation. Link L_{1a} is called the *supplying link*, and links L_{2a} and L_{2c} are *fan-out links*. Conversely, switch sw_2 can also view links L_{2b} and L_{2d} as *contributing links*, and L_{1b} as the *merged link* of a mp-p merge point: data from L_{2b} and L_{2d} is merged on L_{1b}.

When an FRM cell arrives at a switch s from the incoming link l_{ji} of a branch j of VC k, variable $ccr[k][l_{ji}]$ is updated using the CCR value in the FRM cell (pseudo-code given in Figure 2(a)). If this is a mp-mp VC, the maximum of the CCR values seen in the current interval is recorded. The switch is then considered as a logical p-mp branch point with the supplying link being l_{ji}. The FRM cell is duplicated, and sent along each fan-out link l_{fo} of the supplying link l_{ji} (e.g., sent along links L_{2a} and L_{2c} if $l_{ji} = L_{1a}$).

When a BRM cell arrives at switch s from the incoming link l_{ji} of a branch j of VC k, the ER of the VC on link l_{ji} is calculated. The ER calculation is outlined in Figure 2(b). The switch then considers l_{jo}, the outgoing link associated with l_{ji}, as the merged link of a logical mp-p merge point. For instance, if the BRM cell arrives at switch sw_2 from link L_{1a}, then L_{1b} is the merged link, and L_{2b} and L_{2d} are contributing links of L_{1b}. For every contributing link l_{ci} whose merged link is l_{jo}, feedback consolidation is performed using the new ER of the VC on link l_{ji}, and fields BRM.ER and BRM.CI of the BRM cell.

Function Generate_BRM() is then called to generate a BRM cell, if possible, to be sent along link l_{co}, the outgoing link associated with l_{ci} (e.g., sent along links L_{2a} and L_{2c} if $l_{ji} = L_{1a}$). Specifically, a BRM cell may be generated if either one of the following two conditions is satisfied (pseudo-code given in Figure 2(c)):

(1) Switch s has received at least one FRM cell from l_{ci} since the last BRM cell was sent along l_{co}. In addition, switch s has received at least one BRM cell from every branch b ($b \neq l_{ci}$) since the last BRM cell was sent along l_{co} (e.g., switch sw_2 has received BRM cells from both L_{1a} and L_{2b} for $l_{co} = L_{2c}$). A time-out mechanism is also implemented to ensure that the BRM cell flow is not stopped due to non-responsive branches.

(2) Switch s detects an overload condition on link l_{ci}. Overload is determined by comparing the newly calculated ER, new_ER, with $previous\_ER[k][l_{ci}]$, the last ER value sent along link l_{co} to the source(s). Link l_{ci} is said to be overloaded if ($new\_ER < previousER[k][l_{ci}] \times \tau$), where $0 < \tau \leq 1$ is an overload threshold. In general, the smaller the value of τ, the lower the traffic overhead, but the slower the reaction to congestion [6].

Figure 2(d) also outlines the algorithm for updating switch variables at the end of every measurement interval, as done in the ERICA algorithm [9].

4.3. Experimental Results

We added code to the NIST ATM network simulator [11] to implement the FCAM flow control scheme. Several experiments were conducted on various configurations with mp-mp groups of 3-15 sources. We present here a set of three experiments on the configuration depicted in Figure 1. The setting of the experiments is as described in section 3.

The first two experiments simulate source-based and VC/source-based BADs respectively; all the sources have a low ICR of 8 Mbps. The ACR graphs in Figures 4(b) and (c) show that bandwidth is allocated properly to the sources according to each BAD (refer to Table 2). The convergence time[3] is in each case is short, within 250 msec or less. The ACRs remain stable after the sources reach their correct allocations. The feedback consolidation procedure at branch points helps maintain this stability. The utilization of links L_{1b} and L_{3a} are 100% after sources' ACRs converge.

[3]In general, the convergence time depends on the network topology, the number of connections, mp-mp group sizes, and source-to-destination distances. Also, RCAs based on source CCRs take somewhat longer to converge than RCAs using weights, but are more effective.

In the third experiment, the sources' ICR is increased to 100 Mbps in order to test the robustness of the FCAM algorithm. Figure 4(d) shows that the sources' ACRs converge to the correct allocation quickly, within 250 msec or less. Both links L_{1b} and L_{3a} are 100% utilized after sources' rates stabilize . This experiment shows that the FCAM algorithm adjusts well to high traffic demand, provides fast and correct bandwidth allocation, and maximizes bandwidth utilization.

5. CONCLUSION

In this paper we first present an evaluation of the existing mp-mp flow control schemes. Simulation results show that in many cases they under-utilize ABR capacity and do not comply with the selected BAD. Also, they implement only source-based BAD. We propose a mp-mp flow control scheme, the FCAM algorithm, that can support any of the existing BADs based on sources, namely source-based, VC/source-based, and weighted-source-based allocations. This flexibility allows ATM networks to meet various bandwidth requirements of a wide range of ABR applications. The FCAM RCA is based on maximum source CCRs (as opposed to VC weights), a method that has been shown to give correct rate allocation and maximize link utilization. Furthermore, the FCAM RCA requires less storage and processing overheads than those using VC weights (e.g., the MPER and Pao's algorithms): there is no need to maintain VC weights, branch weights, or aggregate CCRs of branches. Feedback consolidation and feedback implosion avoidance are used at branch/merge points to minimize rate fluctuation during transient periods and cell loss rates. Simulation results have confirmed that the FCAM algorithm is effective, allows sources' rates to converge quickly, and adjusts well to high traffic demands.

Although the FCAM flow control scheme is designed in the context of ATM networks, the design principles discussed are applicable to various networks that use feedback control, and useful for developing rate-based congestion-controlled services in high performance IP networks.

6. REFERENCES

[1] ATM Forum, "ATM Forum Traffic Management Specification Version 4.0," April 1996.

[2] D. Cavendish and M. Gerla, "Rate Based Congestion Control for Many-to-Many Multicast ABR Traffic," *SPIE Conference on Performance and Control of Network Systems II*, vol. 3530, November 1998, pp. 122-130.

[3] A. Charny, D. Clark, and R. Jain, "Congestion Control with Explicit Rate Indication," *Proceedings of IEEE ICC '95*, June 1995, pp. 1954-1963.

[4] F.M. Chiussi, A. Arulambalam, Y. Xia, and X. Chen, "Explicit Rate ABR Schemes Using Traffic Load as Congestion Indicator," *Proceedings of 6th International Conference on Computer Communications and Networks (ICCCN '97)*, Las Vegas, Nevada, September 1997, pp. 76-84.

[5] Sonia Fahmy, Raj Jain, Rohit Goyal, and Bobby Vandalore, "Fairness for ABR multipoint-to-point connections," *SPIE Symposium on Voice, Video and Data Communications*, vol. 3530, Boston, MA, November 1998, pp. 131-142.

[6] Sonia Fahmy, Raj Jain, Rohit Goyal, Bobby Vandalore, and Shivkumar Kalyanaraman, "Design and Evaluation of Feedback Consolidation for ABR Point-to-Multipoint Connections in ATM Networks," *Computer Communications*, vol. 22, no. 12, July 1999, pp. 1085-1103.

[7] Sonia Fahmy, Raj Jain, Rohit Goyal, and Bobby Vandalore, "Fair Flow Control for ATM-ABR Multipoint Connections," *Computer Communications*, vol. 25, no. 8, May 2000, pp. 741-755.

[8] J. M. Jaffe, "Bottleneck Flow Control," *IEEE Transactions on Communications*, vol. 29, no. 7, July 1981, pp. 954-962.

[9] Shivkumar Kalyanaraman, Raj Jain, Sonia Fahmy, Rohit Goyal, and Bobby Vandalore, "The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks," *IEEE/ACM Trans. on Networking*, vol. 8, no. 1, Feb. 2000, pp. 87-98.

[10] W.M. Moh and Y. Chen, "Design and Evaluation of Multipoint-to-Point Multicast Flow Control," *Proceedings of SPIE '98 Conference on Performance and Control of Network Systems II*, vol. 3530, November 1998, pp. 143-154.

[11] The NIST ATM/HFC Network Simulator, Version 4.1, National Institute of Standards and Technology, http://w3.antd.nist.gov/Hsntg/prd_atm-sim.html.

[12] U.T. Nguyen and Irene Katzela, "A Flexible Multipoint-to-point Traffic Control Algorithm for ABR Services in ATM Networks," *Proceedings of the IEEE Conference on High Performance Switching and Routing (ATM 2000)*, Heidelberg, Germany, June 2000, pp.185-194.

[13] U.T. Nguyen, "Congestion Control for ABR Multipoint Services in ATM Networks," Ph.D. thesis, Department of Computer Science, University of Toronto, 2003, http://www.cs.yorku.ca/~utn/publications.html.

[14] U.T. Nguyen, "Evaluation of Flow Control Algorithms for ABR Multipoint Services," *9th IEEE Symposium on Computers and Communications*, Alexandria, Egypt, June 2004.

[15] Derek C. W. Pao, "A Congestion Control Algorithm for Multipoint-to-Multipoint ABR Service in ATM Network," *Proceedings of the IEEE Conference on High Performance Switching and Routing (ATM 2000)*, Heidelberg, Germany, June 2000, pp.167-175.

[16] D. Sisalem and H. Schulzrinne, "The Adaptive Load Service (ALS): An ABR-Like Service for the Internet," http://citeseer.nj.nec.com/271051.html.

[17] W. Ren, K.-Y. Siu, and H. Suzuki, "On the Performance of Congestion Control Algorithms for Multicast ABR Service in ATM," *Proceedings of IEEE ATM Workshop '96*, San Francisco, Aug. 1996.

[18] W. Ren, K.-Y. Siu, and H. Suzuki, "Multipoint-to-Point ABR Service in ATM Networks," *IEEE International Conference on Communications*, Montreal, Canada, June 1997, pp. 1185-1190.

[19] W. Ren, K.-Y. Siu, H. Suzuki, and M. Shinhara, "Multipoint-to-Multipoint ABR Service in ATM Networks," *Computer Networks and ISDN Systems*, October 1998, vol. 30, no. 19, pp. 1793-1810.

(a)
Upon receiving an FRM cell from incoming link l_{ji} of branch j of VC k:
$received\_FRM[k][l_{ji}]$ = true;
if ($first\_FRM\_in\_this\_interval[k][l_{ji}]$) {
  $ccr[k][l_{ji}] =$ FRM.CCR;
  $first\_FRM\_in\_this\_interval[k][l_{ji}] =$ false;
}
else $ccr[k][l_{ji}] = \max(ccr[k][l_{ji}]$, FRM.CCR);
for every fan-out link l_{fo} whose supplying link is l_{ji} {
  Duplicate the FRM cell;
  Forward FRM cell to next switch along link l_{fo};
} /* end for */
Generate_BRM(k, l_{jo});
/* l_{jo} is the outgoing link associated with l_{ji} */

(b)
Upon receiving a BRM cell from incoming link l_{ji} of branch j of VC k:
/* Calculate ER on link l_{ji} */
if ($seen\_BRM\_in\_this\_interval[k][l_{ji}]$) {
  $er\_calculated = last\_calculated\_ER[k][l_{ji}]$;
else {
  if ($load\_factor > 1 + \delta$)
    $er\_calculated = ccr[k][l_{ji}]/load\_factor$;
  else $er\_calculated = \max(ccr[k][l_{ji}]/load\_factor$,
    $max\_ER\_previous\_interval$ * $source\_weight[k]$);
  $er\_calculated$= min($er\_calculated$, $ABR\_capacity$);
  $max\_ER\_current\_interval$ =
      max($max\_ER\_current\_interval$,
      $er\_calculated/source\_weight[k]$);
  $last\_calculated\_ER[k][l_{ji}] = er\_calculated$;
  $seen\_BRM\_in\_this\_interval[k][l_{ji}]$ = true;
}
$er\_calculated$ = min ($er\_calculated$, BRM.ER);
l_{jo} = outgoing link associated with incoming link l_{ji};
for every contributing link l_{ci} whose merged link is l_{jo} {
  $received\_BRM[k][l_{co}][l_{ji}]$ = true;
  /*Next 2 steps: part of feedback consolidation*/
  $min\_ER[k][l_{ci}] = \min(min\_ER[k][l_{ci}]$, er_calculated);
  $aggr\_CI[k][l_{ci}] = aggr\_CI[k][l_{ci}]$ OR BRM.CI;
  Generate_BRM(k, l_{co});
  /*l_{co} is the outgoing link associated with l_{ci}*/
} /* end for */

(c)
Generate_BRM(k, l_{co}) {
/* l_{ci} is the incoming link associated with l_{co} */
$new\_ER = \min(last\_calculated\_ER[k][l_{ci}], min\_ER[k][l_{ci}])$;
if ((($received\_FRM[k][l_{ci}]$) AND ($received\_BRM[k][l_{co}][l_b]$
    for all contributing links l_b)
    OR ($new\_ER < previous\_ER[k][l_{ci}] \times \tau$)) {
  Generate a BRM cell;
  BRM.ER = new_ER;
  $previous\_ER[k][l_{ci}] = BRM.ER$;
  BRM.CI = $aggr\_CI[k][l_{ci}]$;
  Forward the new BRM cell to next switch along l_{co};
}
/*Reset consolidation variables*/
$received\_FRM[k][l_{ci}] =$ false;
for every contributing link l_b of l_{co}
  $received\_BRM[k][l_{co}][l_b] =$ false;
$min\_ER[k][l_{ci}] = MAX$; /*a very big value*/
$aggr\_CI[k][l_{ci}] = 0$;
}

(d)
At the end of every measurement interval:
Recalculate $ABR\_target\_capacity$ and $input\_rate$ (refer to reference [9]);
$load\_factor = input\_rate/ABR\_target\_capacity$;
$max\_ER\_previous\_interval =$
    $(1-\alpha) \times max\_ER\_current\_interval$
    $+\alpha \times max\_ER\_previous\_interval$;
$max\_ER\_current\_interval = 0$;
for every VC i
  for every branch b {
    $first\_FRM\_in\_this\_interval[i][b]$ = false;
    $seen\_BRM\_in\_this\_interval[i][b]$ = false;
  }
/* Note: recommended value of δ is 0.1;
recommended value of α is 0.1 [7] */

Fig. 2. The FCAM mp-mp flow control algorithm

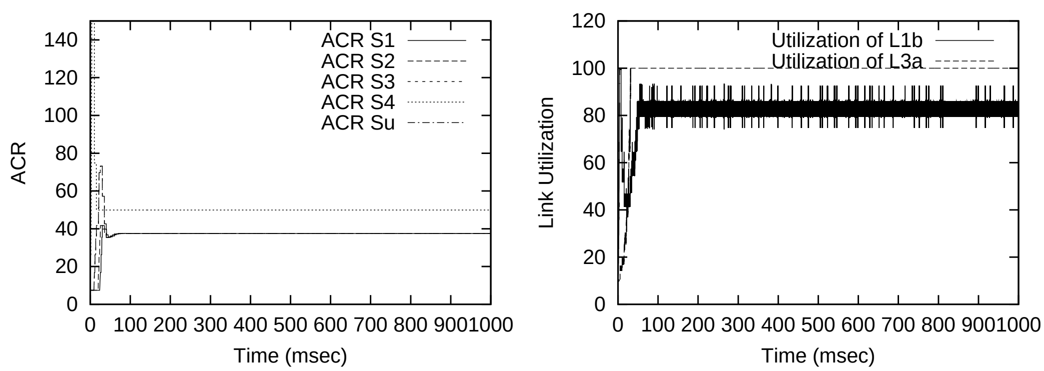
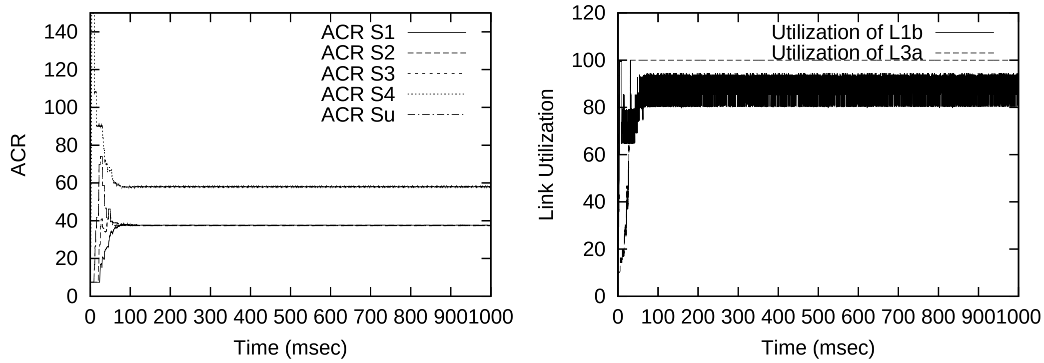
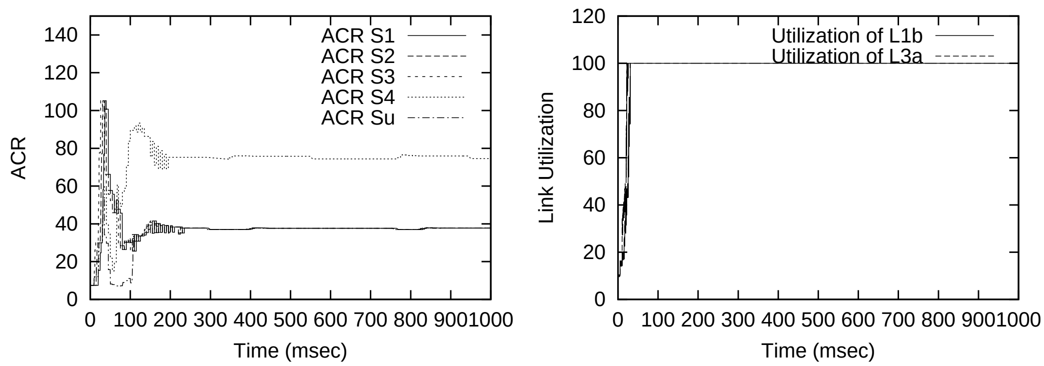


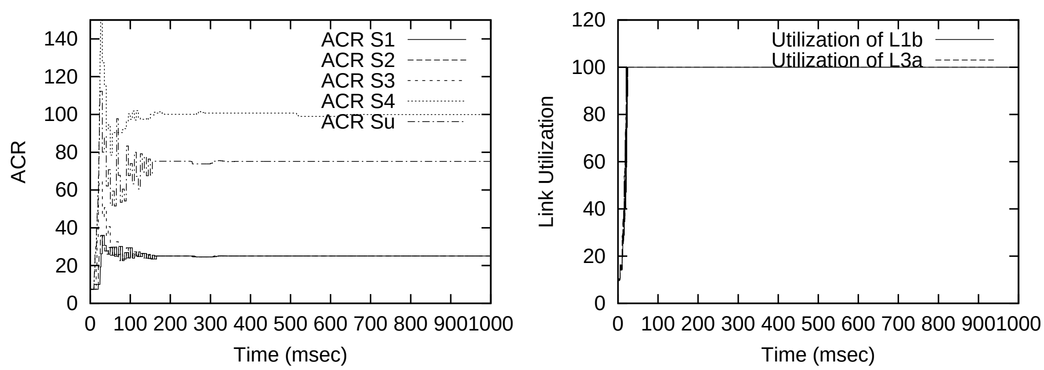
Fig. 3. MPER algorithm; source-based allocation; link L_{1b} is under-utilized.



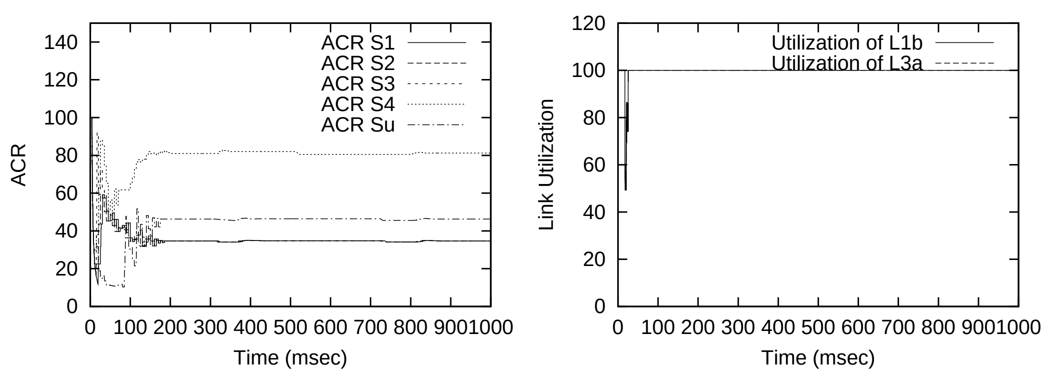
(a) Pao's algorithm; source-based allocation; link L_{1b} is under-utilized.



(b) FCAM algorithm; source-based allocation; ICR = 8 Mbps.



(c) FCAM algorithm; VC/source-based allocation; ICR = 8 Mbps.



(d) FCAM algorithm; weighted-source-based allocation; $source\_weight[m] = 0.75$; ICR = 100 Mbps.

Fig. 4. Source ACRs and link utilization