10

Dynamics of an Explicit Rate Allocation Algorithm for ATM Networks

*L. Kalampoukas, A. Varma**
Computer Engineering Department
University of California, Santa Cruz, CA 95064, USA
E-mail: {lampros,varma}@cse.ucsc.edu

K. K. Ramakrishnan
AT&T Bell Laboratories, Murray Hill, NJ 07974, USA
E-mail: kkrama@research.att.com

Abstract

In this paper we study the performance of an explicit rate allocation algorithm for ATM networks using the available bit-rate (ABR) class of service. We examine the behavior of ABR traffic with simple cell sources, and demonstrate that the allocation algorithm is fair and maintains network efficiency in a variety of network configurations. We also study the behavior of TCP/IP sources using ABR service in a network of switches employing the rate allocation algorithm; the results show substantial improvements in fairness and efficiency in comparison with the performance of TCP in an underlying datagram-based network. We study the performance of ABR traffic in the presence of higher-priority variable bit-rate (VBR) video traffic and show that the overall system achieves high utilization with modest queue sizes in the switches, and the ABR flows adapt to the available rate in a fairly short interval. We also demonstrate the scalability of the rate allocation algorithm with respect to the number of connections.

Keywords

Explicit rate allocation, congestion control, TCP over ATM.

1 INTRODUCTION

Asynchronous Transfer Mode (ATM) networks are being developed with the intent of providing a single common technology to carry voice, video and data traffic. Networks based on ATM combine the flexibility of packet-switched networks with the service guarantees and predictability offered by circuit-switched networks.

Several service classes have been defined in the context of ATM networks. The *Available-Bit-Rate* (ABR) service class [Giroux, 1995], defined to support delay tolerant best-effort applications, uses rate-based feedback mechanisms to allow them to adjust their transmission rates to make full utilization of the available bandwidth [Bonomi, 1995]. Compliant connections are also assured of a low loss rate, and if needed, a minimum bandwidth allocation. The ATM Forum Traffic Management Committee is currently defining a rate-based

*Supported by the Advanced Research Projects Agency (ARPA) under Contract No. F19628-93-C-0175 and by the NSF Young Investigator Award No. MIP-9257103.

L. Mason et al. (eds.), *Broadband Communications*

congestion control framework to meet this objective. This framework allows a number of options for the switches to signal their congestion state to the source. With the *explicit-rate marking* option that is the focus of our work here, the source of each connection periodically transmits a special *resource management* (RM) cell. The source specifies the bandwidth demand and the current transmission rate of the connection in each transmitted RM cell. With the explicit rate scheme, switches communicate in the RM cell, the amount of instantaneous bandwidth it can allocate to each connection to the source of the connection. The goal of the allocation is to arrive at an efficient allocation that is also *max-min fair* [Bertsekas, 1992].

Several rate allocation algorithms using the explicit-rate option have been proposed [Charny, 1994, Kalampoukas, 1995a, Jain, 1995]. In this work we study the dynamics and evaluate the performance of the rate allocation algorithm proposed in [Kalampoukas, 1995a We consider the behavior of the rate allocation algorithm in both ATM-layer-generated ABR traffic and TCP-controlled ABR traffic. In the first case we show that, in the network configurations being analyzed, the algorithm converges to a steady state, allocates the available bandwidth fairly among competing connections, and has modest buffer requirements. We demonstrate its scaling capabilities by increasing the number of active connections by a factor of more than 10.

We also study the behavior of TCP-controlled ABR traffic. We demonstrate that the use of an explicit rate allocation scheme enhances the fairness achieved for TCP/IP traffic compared to its performance in traditional datagram networks.

The paper is organized as follows: Section 2 briefly reviews the rate-based congestion control framework and describes the proposed rate allocation algorithm. Section 3 provides a description of the simulation models used in this work. Section 4 discusses the dynamics of the described algorithm in a network configuration consisting of connections with widely-different round-trip times. We study the behavior of the network first with only ATM-layer generated traffic, and subsequently with ABR traffic that is flow-controlled by TCP. Section 4.3 studies the performance of ABR connections when mixed with VBR traffic. Section 4.4 investigates the performance of TCP in a dynamic environment, where the total bandwidth available to TCP connections is varied dynamically. Finally, Section 5 summarizes the results and proposes directions for future work.

2 SOURCE AND SWITCH BEHAVIOR

We describe in this section the source and destination algorithms used in our study. Due to lack of space we provide just an outline of the source and switch behavior. Detailed information on the ATM Forum's source policy as well as a complete description of the rate allocation algorithm under investigation can be found in [Bonomi, 1995, Kalampoukas, 1995b].

According to the ATM Forum's framework, the source of a connection (VC) transmits cells at a rate allowed by the network, termed the *allowed cell rate* (ACR), until it receives new information in an RM cell that it had transmitted previously. The source sends an RM cell every $Nrm-1$ data cells transmitted. This proportional transmission of RM cells is to ensure that the amount of overhead for RM cells is a constant, independent of the number of VCs in the network or their rates.

The switches use the *explicit rate* option. Whenever an RM cell from connection is received at a given switch, the switch determines the allocation for the VC based on the

bandwidth being requested in the explicit rate (ER) field. If the maximum bandwidth that can be allocated is less than the value in the ER field, then the ER field is updated to reflect the new maximum possible allocation on the path of connection so far.The explicit rate option assumes the existence of an algorithm within the switch that allocates the available bandwidth on each outgoing link among the connections sharing it. In this paper we consider the rate allocation algorithm described in [Kalampoukas, 1995a].

When the RM cell returns back to the source, the transmission rate (allowed cell rate, ACR) is updated based on the value indicated by the ER field in the returned RM cell. The value in the ER field reflects the bandwidth that can be allocated at the bottleneck link in the path of the connection. If the current transmission rate is above the value in the ER field, the source immediately reduces its rate to this value. However, if the current rate is less than the returned ER value, the transmission rate ACR is increased gradually by a constant amount $(Nrm \cdot AIR)$ to the current ACR. In addition, the source is never allowed to exceed the rate specified by the ER field of the returned cell. Thus, if $r(t)$ is the transmission rate of the source the instant just before the arrival of an RM cell, and $r(t+)$ the rate after the update, then

$$r(t+) = \min(r(t) + Nrm \cdot AIR, ER).$$

3 SIMULATION ENVIRONMENT

In this section, we provide an overview of the simulation models used in the paper. More detailed description of a specific topology used in a simulation will be given in the corresponding sections describing the simulation results. The simulations were performed using the OPNET tool.

The links in the network are full-duplex with a capacity of 155 Mbits/sec each, unless otherwise specified. The switches are nonblocking, output-buffered crossbars. There is one queue per output port for ABR traffic and its scheduling policy is FIFO, with each output queue being shared by all the virtual circuits (VCs) that are sharing the outgoing link. We assume that all the switches support the explicit rate allocation algorithm of Section 2 and the sources follow the source policy described. The parameters for the source-end systems are set as follows:

- Nrm = 32 cells.
- ICR = PCR/50 $\approx$7300 cells/sec. This is the initial cell rate of the source
- AIR = 180 cells/sec per cell transmitted (about 5760 cells/sec maximum rate increase every Nrm cells).
- PCR = 365,566 cells/sec. This is the peak rate for all VCs (link rate).

An important observation we make is that the same set of parameters are used for both WAN and LAN configurations which have widely different characteristics.

When simulating TCP over ATM, we use the ATM Adaptation Layer Type 5 (AAL 5) [I363]. AAL 5 performs segmentation and re-assembly between IP packets and ATM cells. Each IP packet is extended by eight bytes to accommodate the AAL header. Thus, the number of ATM cells produced by the original IP datagram is given by $\left\lceil \frac{\text{IP packet size}+8}{48} \right\rceil$.

The model for TCP used in the simulations is based on the TCP-Reno version. It supports the congestion control mechanism described by Jacobson [Jacobson, 1988], ex-

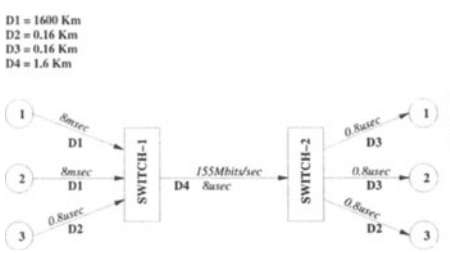

Figure 1 Configuration R1.

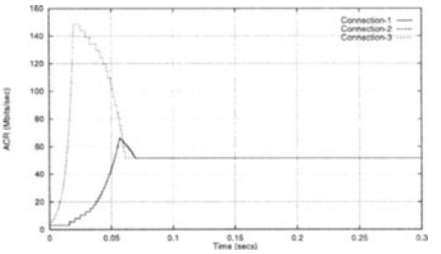

Figure 2 Transmission rates (ACRs) of the three connections in the R1 configuration.

ponential back-off, enhanced round-trip (RTT) estimation based on both the mean and the variance of the measured RTT, and the *fast retransmit and fast recovery* mechanisms.

We focus on the following performance measures: 1) The ACR, or transmission rate, at the source. In all the graphs we show the ACR value on every change, without any filtering of the collected information. 2) The utilization of the links. We present the utilization averaged over 5 millisecond intervals. 3) The queue length at the switch for individual links. We present in the plots the queue length, which is the maximum value observed during a 5 millisecond interval. 4) In the case of TCP traffic, we also show the TCP sequence number growth for each individual TCP connection and the corresponding window size in bytes, where appropriate.

4 SIMULATION RESULTS

In this section we consider the dynamics of the rate allocation algorithm in a network configuration where connections with widely different feedback delays interact. We first study the behavior of the algorithm with ABR traffic from cell sources, and subsequently characterize its behavior with TCP-generated ABR traffic.

4.1 Performance with ABR Traffic

We begin the evaluation of our rate-allocation algorithm with a simple configuration, referred to from now on as the *R1 configuration* , shown in Figure 1. It consists of three connections which open simultaneously and request peak bandwidth. Data flows from the end-systems on the left to the ones on the right. Connections are set up between corresponding end-systems, identified by the same index within circles. The link propagation delays and capacities are as shown in the figure. The reason we find the configuration R1 interesting is because of the large difference (three orders of magnitude) between the round-trip times of the different connections. The round-trip delay of connection 3 (to be referred from now on as the *short connection*) is 11.2 μseconds while that of connections 1 and 2 (from now on to be referred as *long connections*) is 16.076 milliseconds. We expect D4 to be the bottleneck link in the configuration. All the sources follow the source policy outlined in Section 2. The sources are assumed to be greedy, that is, they always set the ER field of every transmitted RM cell to the peak link capacity of 155 Mbits/sec.

Because of the large difference in the feedback delay between the long and short connections, and the small initial rates of the flows (ICR), we expect that the short connection (connection 3) will quickly ramp up to acquire a larger than fair share of the bottleneck

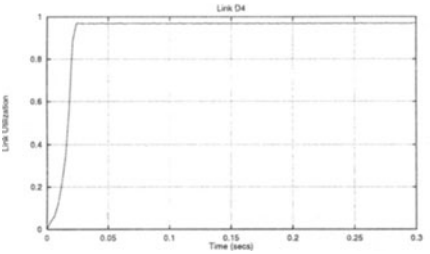

Figure 3 Total utilization of link D4 in the R1 configuration with three connections.

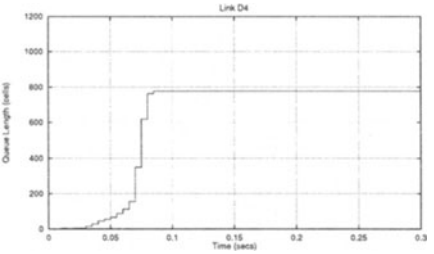

Figure 4 Queue length at the bottleneck link in switch-1.

link bandwidth. This initial start-up transient is clearly seen in Figure 2 which shows the exact evolution of the transmission rate at the source (ACR) for the three VCs. However, as time passes, the returned RM cells for the long connections allow them to acquire their fair share, and the short connection releases the bandwidth it acquired during the slow start-up of the long connections. Eventually, all the rates converge to their final allocation of one-third of the link bandwidth (about 51.5 Mbits/sec per connection).

The behavior of the rate decrease for the short connection during the start-up phase is gradual, rather than in a small number of discrete steps. This is due to the fact that the transmission rates for the long connections, which are being carried by the CCR field, increase gradually, thus resulting in a gradual decrease to the available rate to the short connection until the steady state is reached.

It is important to note the small overshoot in the transmission rates of the long connections before convergence is finally reached. This overshoot is a direct result of the allocation based on CCR values of the connections, and can be explained as follows: Assume that each of the two long connections transmits at time t_1 a forward RM cell with its CCR field containing transmission rates $r_1(t_1)$ and $r_2(t_1)$, respectively. These RM cells arrive at Switch 1 at time $t_2 = t_1 + 8$ msecs. Let $A_3(t_2)$ be the current allocation for the short connection in Switch 1 at that time. A computation for rate allocation is performed at time t_2 for each of the long connections. Assume that the RM cell from Connection 1 is the first seen by the switch. The updated ER value in its RM cell will now be $B - (A_3(t_2) + r_2(t_1))$, where B is the link capacity. It is easy to observe that, if $(A_3(t_2) + r_2(t_1))$ is less than $2B_{eq}$, the ER value signaled to Connection 1 can be larger than B_{eq}. The same ER value is also signaled to Connection 2 when its RM cell is processed. When these RM cells reach the sources of the long connections, the sources attempt to gradually increase their rates to the new ER values signaled, resulting in the rates exceeding the fair value B_{eq} temporarily. This overshoot is soon corrected when the increased CCRs of the long connections reach the switch, which clamps their allocations to B_{eq}. However, because of the long feedback delay, convergence to B_{eq} occurs slowly at the sources of the long connections.

Notice here that the transient bandwidth over-allocation may be reduced if we update the ER field of the RM cells going in the backward direction also. In that case, if the value in the ER field carried by an RM cell is larger than the most recent value of A_{max}, we update the ER field with the new A_{max}. This might improve the convergence of the rate allocation process in the general case; however, the modification would have little effect in this specific example because the congested switch is very close to the destination.

In Figure 2, the transmission rates of the sources converge to their final values within

70 msecs. Considering the 16 msecs round-trip delay of the long connections, this is quite reasonable, After the rate allocation process is completed, the transmission rates remain constant and the overall behavior is stable as long as the network state remains unchanged.

Although the feedback delay affects responsiveness of long connections to network changes, the utilization of the congested link is less affected. This is because of the short connection is able to utilize the excess bandwidth of the link while the long connections are gradually increasing their rates. Figure 3 shows the utilization of the link D4. Note that the utilization reaches its maximum value within approximately 25 msecs and remains constant thereafter. The maximum link utilization reached is about 97%, the theoretical maximum achievable after accounting for the overhead due to RM cells.

The transient bandwidth over-allocation causes a queue build-up in switch-1, as illustrated by Figure 4. The length of the queue is a function of the amount of the bandwidth over-allocation and the duration of the transient phase. As shown in Figure 4, however, even in a network with a round-trip delay of the order of 16 msecs, the built-up queue was relatively small, approximately 780 cells (approx. 40 Kbytes). Once built up, the queue size remains steady until a change in network state occurs, because our target link utilization is set at 100 %.

To examine how the allocation algorithm scales with the number of connections, especially with respect to its convergence time, the required amount of buffering and the bottleneck link utilization, we slightly extended configuration R1. The new configuration (the figure is omitted due to space constraints) contains 5 nodes on each side. Every source node on the left now originates eight connections, thus increasing the total number of VCs to 40. The round-trip delay for the two new sources was chosen identical to that of the long connections in Figure 1. Thus, the configuration consists of 32 connections with 16 msecs round-trip delay (long connections), and 8 connections with very small (about 11.2 μsecs) round-trip delay (short connections).

The transmission rates (ACR) for the connections in this modified configuration are shown in Figure 5. For simplicity, we have plotted only the transmission rate for a single connection in the set of connections originating at each source node. The behavior of the transmission rates is almost identical to that of the original R-1 configuration with three connections. However, convergence of the transmission rate to the final values is faster than before, taking only about 45 msecs (compared to 70 msecs with 3 connections). Therefore, the convergence time scales well with increasing number of connections.

As before, setting the target link utilization at 100% can lead to queue buildup at the bottleneck switch during the transient phase. However, the behavior of the queue size at the bottleneck link in switch 1, shown in Figure 6, indicates that the queue at the bottleneck link does not grow rapidly with the number of active connections. An increase by a factor of about 13 for the number of connections results in increasing the queue size by only a factor of 3. When we increased the number of short connections rather than the number of long connections in the R1 configuration, the increase in queue size was even smaller, about 30%.

The utilization of the bottleneck link achieves its maximum value somewhat quicker when the number of connections is increased. This is done in about 15 msecs, compared to 25 msecs for the same configuration with three connections. This is due to a combination of the relatively large aggregate initial rate for all the long connections and the fast ramp-up by the short connections due to their small feedback delay.

From our observations of the limited increase in the queue size, fast convergence and

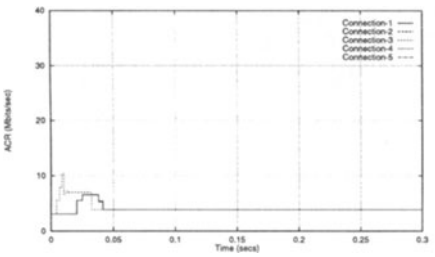

Figure 5 Transmission rates for the five sets of connections in the modified R1 configuration with 40 connections.

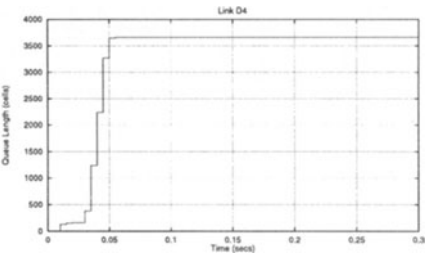

Figure 6 Queue length for the bottleneck link in switch-1 in the modified R1 configuration with 40 connections .

maintenance of high utilization, we believe that the allocation algorithm scales well with the number of connections.

4.2 Dynamics of TCP Traffic over ABR Service in a Network Configuration with Unequal Feedback Delays

The ABR traffic will not, in general, consist of ATM-layer-generated data only. Many applications use a transport protocol to provide reliable end-to-end transmission of data. Since TCP is currently the most widely used reliable transport protocol, ATM will likely be used widely as the datalink layer for the TCP/IP Internet as a means of evolving from the current infrastructure. In this subsection we study how the rate allocation algorithm at the ATM layer influences the behavior of TCP.

Of particular interest is to study how the TCP congestion control mechanisms affect the behavior of the rate allocation algorithm. The TCP congestion control algorithm is based on end-to-end windows and consists of several components. Key components are the *slow-start* algorithm, a congestion-avoidance mechanism, and an algorithm to estimate round-trip delays [Jacobson, 1988]. The TCP Reno Version, introduced in 1990, added the *fast retransmit and fast recovery* algorithm to avoid performing slow-start when the level of congestion in the network is not severe to warrant congestion recovery by slow-start.

For this study, we use the same R1 configuration considered in the previous subsection with two long connections and one short connection. The only difference is that the traffic of each ABR connection is now flow-controlled by TCP.

In addition to studying the initial start-up phase and the steady-state behavior of the connections, we also examine the dynamics of the connections when a packet loss occurs. This is achieved by dropping a cell from a TCP segment from connection 1 at time $t = 0.5$ seconds. In this case the AAL5 layer at the receiving end will detect a corrupted packet and discard all the remaining cells from that packet. The segment loss is later detected by the TCP source, which then retransmits the segment.

Figure 7 shows the ACR values at the sources of the three connections. The source rate behavior during the start-up phase is similar to that with cell sources, except for the more abrupt increase and decrease steps. This change in behavior is due to the TCP slow-start algorithm which increases the window size by doubling it every round-trip time. This produces intervals of time during which sources have no data to transmit. Since the source rate is allowed to increase only on the receipt of an RM cell, the idle intervals produce

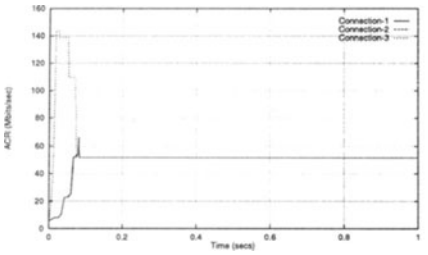

Figure 7 ACRs for the three connections in R1 configurations with TCP-generated traffic (in Mbits/second).

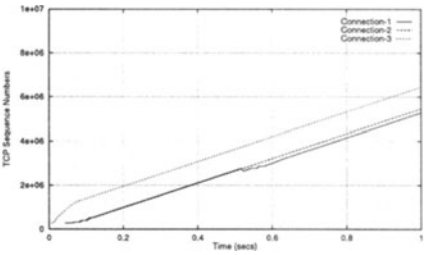

Figure 8 TCP sequence numbers for the three connections.

breaks in the rate increase process. However, the ACRs eventually converge to the fair values and remain steady. Note also that once the rate allocation algorithm converges, the behavior is similar to each connection operating over a dedicated link with no interference from other connection.

Figure 8 shows the increase in the sequence numbers of TCP segments transmitted by the three connections as a function of time. The plot for the short connection has a substantially higher slope during the slow-start phase, owing to its much smaller round-trip delay. However, in steady state, the rate of increase for all the connections is identical, demonstrating the effectiveness of the rate allocation scheme in providing fairness among connections with widely different round-trip delays.

The simulation results regarding the congestion windows (the graphs are omitted due to lack of space) show that the congestion windows for all three connections open to their maximum size of 150 Kbytes within 100 ms and remain at that state until 0.5 secs when a packet loss is simulated by discarding a single ATM cell from connection 1. The source TCP of connection 1 soon detects the segment loss and enters the fast-retransmit and fast-recovery phase, while the other two connections remain completely unaffected owing to the isolation provided by the rate allocation algorithm.

The simulation results for the overall utilization of the bottleneck link D4 (again the graphs are omitted due to lack of space) show that the maximum utilization of 97% is reached within 200 ms after start-up. The long period required for the link utilization to get maximized is caused by the TCP slow-start process. At time 0.5 second when the simulated cell loss happen, the utilization is transiently reduced because the ATM source policy we have implemented does not incorporate any provisions for recovering bandwidth from an idle source and therefore, the unused bandwidth of connection 1 during its recovery phase is not made available to other connections.

In summary, the simulation results in this subsection show that substantial improvements in fairness and efficiency in the operation of TCP can be obtained by the use of ABR service in conjunction with our rate allocation algorithm.

4.3 Performance of ABR Traffic Mixed with VBR Traffic

Up to now, we have focused on the rate allocation process when considering only ABR traffic. It is important to examine the ability of the scheme to adapt to changes in the available bandwidth, when there is a mix of high-priority traffic such as video and voice.

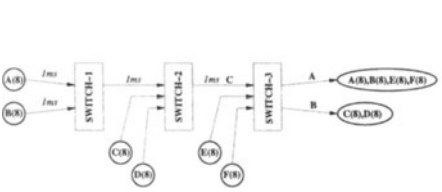

Figure 9 Configuration R2.

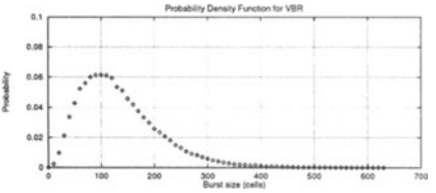

Figure 10 Probability density function for the burstiness of VBR traffic.

In this section, we study the effects of variable-rate real-time traffic (VBR traffic) on ABR traffic that is rate-controlled using our rate allocation algorithm.

In this set of simulation we use configuration R2 shown in Figure 9. In this configuration each source consists of four VBR and four ABR connections. Each of the sources originates eight connections, providing a total of 48 connections. All the ABR connections request peak bandwidth. The VBR connections have an allocated bandwidth, which is based on the average rate for the video data generated by the application.

The VBR traffic is based on the model described in [Heyman, 1992]. One frame of video data is generated approximately every 1/25 seconds (the model assumes a PAL system, not an NTSC system which transmits 30 frames/sec) and the size of the frame expressed in number of cells follows the probability density function given in Figure 10. The resulting process has a distribution that generates data with an average rate of about 1.5 Mbits/secs. We reserve, in all the links on the path from the source to the destination, a bandwidth of 1.9 Mbits/sec for each VBR connection. Thus, the average utilization of the reserved bandwidth for a VBR connection is expected to be about 75%.

The VBR traffic generated with the model described earlier may exhibit very bursty behavior. In order to limit the burstiness of each VBR connection, we shape its traffic using a token bucket. The bucket size is set to 50 cells and the rate of token arrival was set to be equal to the bandwidth reserved to each connection, that is 4,500 cells/sec=1.9 Mbits/sec. In order to avoid any synchronization between the video streams as much as possible, the corresponding VBR connections open at random times that are uniformly distributed in the interval (0,50 msecs).

We assume that the VBR and ABR classes of traffic are buffered in the switches in separate queues. Scheduling between the two classes is based on static priorities, with the VBR traffic always taking higher priority. Thus, the switch transmits an ABR cell only when the VBR queue is empty. Since we are primarily interested in the effect of VBR service on the ABR class, we use a single FIFO queue for all VBR traffic. As always, we assume that the ABR traffic share a single common FIFO queue at each outgoing link of a switch.

Figure 12 shows the ACR for a representative ABR connection from each source node (other connections from the same source exhibit similar behavior). The rate allocation process converges quickly (within 50 msecs) to the final allocation. After convergence, the rate allocated to each connection is a fair share of the available bandwidth: this is the total bandwidth minus the bandwidth reserved for VBR traffic. Here, the bandwidth available to ABR traffic on links A and C is 124.6 Mbits/sec and therefore each ABR connection has an available capacity of about 7.8 Mbits/sec.

The expected link utilization is close to 100% for link A and 50% for link B. However, the instantaneous link utilization (averaged over 5 msec intervals) exhibits spikes as shown

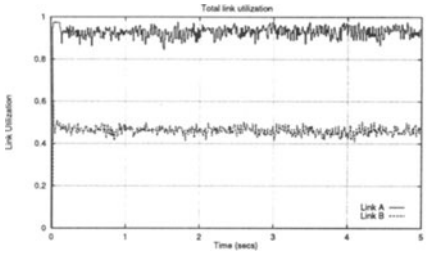

Figure 11 Utilization of links A and B.

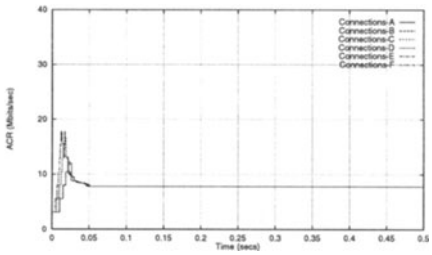

Figure 12 ACR for each connection set.

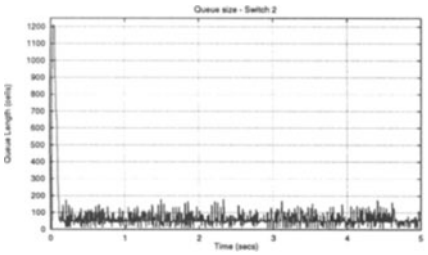

Figure 13 Queue size for ABR traffic – Switch 2.

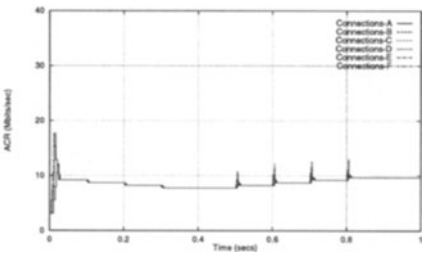

Figure 14 ACR for each connection set with staggered opening and closing of VBR connections at 100 millisecond intervals.

in Figure 11. Accounting for the overhead of 3% of the available bandwidth for RM cells, the utilization of the links is close to the maximum attainable for both the links A and B. The difference is simply because of the over-allocation of bandwidth that we did for the VBR connections: on the average, the VBR traffic should utilize only 75% of its allocated bandwidth. However, this conservative over-allocation for the VBR connections has the desirable side effect of maintaining the queue sizes small. Although the utilization is kept high, the queue sizes for ABR traffic remain small even in the presence of VBR traffic. The queue length for ABR traffic in switch 2 is shown in Figure 13. The queue behavior for the other switches is similar. The queue sizes have an average of about 50 cells (which is also the size of the token bucket of a VBR connection) and a maximum of about 200 cells in steady state.

Figure 14 shows the behavior of the ACRs of the ABR connections when the VBR connections open in a staggered fashion. Six VBR connections (one from each source) open every 100 msecs starting at time $t = 0$ seconds. Then, at time $t = 0.5$ seconds, these VBR connections start closing in a similar manner. As shown in the figure, the convergence to the final allocation after each change in available bandwidth is rapid. In the worst case, the rate allocation process is completed within 20 msecs.

Our results suggest that the rate allocation algorithm will perform well even in the presence of different service classes. For the specific configuration considered, its performance was efficient and scaled well with the number of connections. However, further work is needed to understand in greater detail the dynamics of the algorithm in more general network configurations.

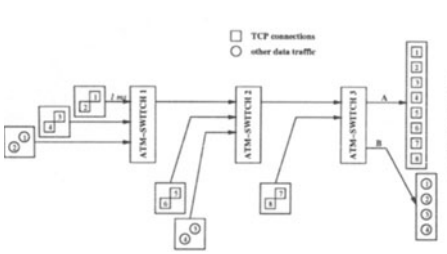

Figure 15 The R3 network configuration.

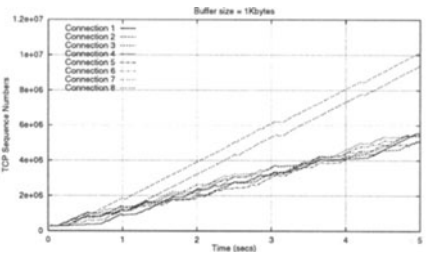

Figure 16 TCP sequence numbers (1 Kbytes buffer size).

4.4 TCP in a Dynamic Environment

In this section, we study the performance of TCP in a network configuration in which the bandwidth allocated to TCP connections is not constant.

The configuration we use is shown in Figure 15 and will be referred to as *R3*. Two types of connections are simulated: TCP-controlled connections and ABR cell sources. The TCP connections are kept open throughput the simulation. To simulate a dynamic environment, the ABR cell sources open and close frequently according to an exponential ON/OFF model. The mean duration for both the ON and OFF periods is set to 100 ms.

The random opening and closing of the ATM-layer connections trigger frequent re-computations of the allocations at the switches. To observe the effect of possible transient over-allocations during the convergence of the algorithm, we chose a very small buffer size of 1 Kbyte for the switches, making a packet loss very likely if a transient over-allocation occurs during convergence of the algorithm after a change in the connection states.

Figure 16 shows the progress of the sequence numbers for the eight TCP connections. All of the connections make steady progress. Connections 7 and 8, being closest to the destination, are able to use more than their fair share during transient periods. Note that this behavior does not represent any inherent unfairness in the rate allocation algorithm, but is due to the delay of the control loop. The aggregate throughput sustained by the TCP connections even with only 1 Kbyte of buffering is about 60% of the maximum attainable.

5 CONCLUSION

In this paper, we evaluated the performance of the explicit rate allocation algorithm presented in [Kalampoukas, 1995a] in a variety of network configurations and with a diverse range of workloads. Traditionally, feedback-based congestion control mechanisms have shown a bias towards flows with shorter round-trip times when they co-exist with long round-trip time flows. Because of the rate allocation mechanism we use here, this bias is eliminated, and we see a fair allocation even under the extreme condition where the round-trip times differ by three orders of magnitude.

We also show that the proposed allocation algorithm retains several desirable characteristics as we scale in the number of connections. Increasing the number of the connections by a factor of 13, the queueing requirements go up only by a factor of 3 with the increase

from 2 to 32 of the long-round-trip time connections (with 16 milliseconds round-trip time).

An important requirement for the ABR service is that it mesh well with traditional higher-layer protocols such as TCP/IP. It is well known that TCP exhibits unfairness when multiple TCP connections sharing a bottleneck link have widely different round-trip times. We show that TCP running over ABR avoids this unfairness, there is a dramatic reduction in the queueing requirements at the bottleneck link, and no packet losses occur due to congestion.

Another important need is for ABR flows to operate well when there are higher-priority VBR flows co-existing with the ABR traffic. In the *parking lot* configuration with 48 connections (equally divided as 24 ABR and 24 VBR flows), we achieved fairness, and full utilization of the bottleneck links. In spite of the burstiness of the VBR traffic and the greediness of the ABR flows, the queue sizes were of the order of only 100 cells, which is quite reasonable.

In the future we plan to study the interaction of the proposed rate allocation algorithm with non-cooperative sources. Also, we would like to examine the system performance with not all flows use their stated bandwidth, as specified in the CCR field.

REFERENCES

[Giroux, 1995] N. Giroux and D. Chiswell, "ATM-layer traffic management functions and procedures," in *Proceedings of INTEROP '95 Engineer Conference*, March 1995.

[Bonomi, 1995] F. Bonomi and K. W. Fendick, "The Rate-Based Flow Control Framework for the Available Bit Rate ATM Service," *IEEE Network*, vol. 9, no. 2, pp. 25–39, March/April 1995.

[Bertsekas, 1992] D. Bertsekas and R. Gallager, *Data Networks.* Prentice Hall, 2nd ed., 1992.

[Charny, 1994] A. Charny, "An Algorithm for Rate Allocation in a Packet-Switching Network with Feedback," Master's thesis, Massachusets Institute of Technology, May 1994.

[Kalampoukas, 1995a] L. Kalampoukas, A. Varma, and K. K. Ramakrishnan, "An efficient rate allocation algorithm for ATM networks providing max-min fairness," in *Proceedings of 6th IFIP International Conference on High Performance Networking, HPN'95*, September 1995.

[Jain, 1995] R. Jain, "Congestion Control and Traffic Management in ATM Networks: Recent Advances and A Survey." submitted to Computer Networks and ISDN Systems.

[Kalampoukas, 1995b] L. Kalampoukas, A. Varma and K. Ramakrishnan, "Dynamics of an Explicit Rate Allocation Algorithm for Available Bit-Rate (ABR) Service in ATM Networks," Tech. Rep. UCSC-CRL-95-54, University of California, Santa Cruz, December 1995.

[I363] CCITT, *Draft Recommendation I.363.* CCITT Study Group XVIII, Geneva, January 1993.

[Jacobson, 1988] V. Jacobson, "Congestion avoidance and control," in *Proceedings of ACM SIGCOMM'88*, pp. 314–329, 1988.

[Heyman, 1992] D. P. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in ATM networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 2, no. 1, pp. 49–59, Mar. 1992.