A Rate Based Back-pressure Flow Control for the Internet

Carlos M. Pazos and Mario Gerla
Computer Science Department
University of California, Los Angeles
405 Hilgard Ave., Los Angeles, CA 90024
{pazos,gerla}@cs.ucla.edu

Abstract

The Internet has traditionally relied on end-to-end congestion control performed at the transport layer by TCP. In this paper, we discuss the limitations of this approach to address the large number of flows and the large delay-bandwidth product scenarios typical of next generation Internets. We propose a link layer back-pressure flow control which can be applied to Internet backbones over ATM. More precisely, we use the ABR service and flow control and we turn routers into virtual sources (VSs) and virtual destinations (VDs) for the ABR control loop. We introduce a VS/VD "behavior" that implements a rate based back-pressure flow control and that addresses max-min fairness.

Keywords

Internet Backbones, ABR Service, Back-pressure Flow Control.

1 INTRODUCTION

Following the rapid evolution of Internet services in recent years, the Best Effort (BE) service is no longer adequate to address the requirements of new services, some of which have real time (RT) constraints and thus require resource commitment from the network in order to implement Quality of Service (QoS) guarantees. The IEFT Internet Integrated Services (ISS) working group is investigating new services and features to allow the Internet to transport multimedia traffic (Braden *et al.* 1994).

As a result, we are bound to experience an ever increasing demand for transmission resources, already a reality given the exponential growth of current Internet traffic. The conventional approach to address this problem has consisted on adding faster links to Internet backbones, upgrading the available pool of resources. While this course of action satisfies the Internet craving for bandwidth, it also makes it more difficult to control traffic and prevent congestion. With faster transmission rates, we are unavoidably faced with

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35388-3_42

H. R. van As (ed.), *High Performance Networking*

problems due to large delay-bandwidth products on backbone links (a huge number of packets is in transit on these links).

The RT traffic can reserve resources and it does receive priority service on routers in the backbones. For this traffic, appropriate resource allocation and efficient admission control are probably enough to avoid the effects of congestion. On the other hand, the bulk of BE traffic is transported over the backbones under TCP window flow control applied to individual sessions on the edges of the network. Namely, TCP sources adjust their transmission rates in reaction to congestion in the network, which is detected when packet are lost. Hence, actions to remedy congestion are only taken when congestion sets in. The associated reaction time is in the order of end-to-end round trip times (RTTs).

Furthermore, it has been argued (Morris 1997) that TCP alone cannot fairly and effectively control a very large number of sessions, an increasingly frequent scenario in the Internet. Increasing the bandwidth available on backbone links and/or increasing buffering space on routers can help accommodate the traffic of a very large number of sessions (Morris 1997), but TCP sessions are not generally subjected to admission control.

In this paper we describe a rate based back-pressure flow control that acts on the **aggregate** traffic between each pair of routers. Each router continuously notifies its neighbors of the rate it can accept from them and the back-pressure control propagates all the way to ingress edge routers. Hence, the response time is in the order of the RTT between the edge router and the point of congestion.

Our flow control approach builds on the work in (Pazos *et al.* 1997), where the use of the ATM ABR service was suggested for Internet backbones. Namely, the links interconnecting IP routers are ABR VCs, rather than the more conventional CBR or UBR VCs. With ABR, the backbone VCs are not restricted to the CBR peak rate allocation and statistical multiplexing is improved. As compared with UBR, the ABR service offers a much better protection guarantee. In this paper, we take a step further and we implement a rate based back-pressure flow control scheme. One considerable advantage of this approach is that it is almost entirely implemented at the ATM layer on routers and hence it requires no modifications to either TCP or IP protocols.

The balance of the paper is organized as follows. In section 2 we describe the congestion problem we address in this paper and in section 3 we present the network scenario we consider for the remainder of the paper. In section 4 we review current approaches for congestion control in Internet backbones and we introduce our back-pressure flow control scheme. In section 5 we present simulation results. Finally, in section 6, we make some concluding remarks.

2 CONGESTION IN THE BACKBONE

Consider the model shown in Figure 1. Congestion occurs when the incoming traffic ($\lambda_{ad} + \lambda_{bd}$) feeding an outgoing Virtual Circuit VC_d exceeds its capacity C_d. As a result, packets may be dropped, impacting effective network utilization and compromising performance. In order to avoid such losses, we need to selectively slow down the traffic flows λ_{ad} and λ_{bd} at the respective sources. This is clearly a hard problem to solve unless an end-to-end rate based (or credit based) feedback mechanism such as the ABR flow control (ATM Forum 1996) is employed. However, since most WAN TCP sessions transfer only a few kilo-bytes of data (Paxson 1993), and since convergence to the fair share of the available bandwidth may take a number of RTTs (Jain *et al.* 1996), the efficiency of the ABR flow control may be compromised. The Explicit Congestion Notification (ECN) approach proposed for IPv6 in (Floyd 1995) attempts to provide feedback control from the network layer (IP) to TCP. The approach in (Roche *et al.* 1995) allows an edged router to provide congestion feedback to other sending edge routers.

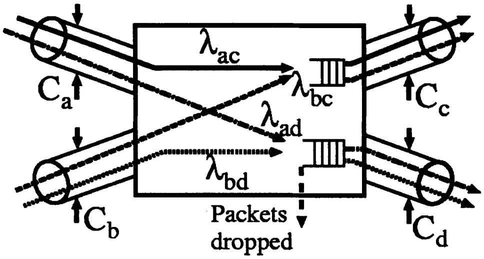


Figure 1 A congested router.

The current approach for congestion control in backbones relies on the TCP window flow control. With TCP, only those sessions suffering packet loss (experiencing congestion) reduce their transmission rate (window). In Figure 1, for instance, the sessions using VC_c are unaffected by the congestion on VC_d and the TCP flow control does not act on them. However, if the flow λ_{ad} is responsible for the abuse of VC_d, the flow λ_{bd} is also penalized in the process and the TCP mechanism cannot selectively slow down the λ_{ad} flow. Namely, TCP alone cannot ensure fairness (Morris 1997). A number of other unfair TCP behaviors have been addressed in (Floyd *et al.* 1991, Floyd 1991).

Hence, the best to alleviate the congestion on the router of Figure 1 and to avoid packet losses is to selectively back-pressure the flow λ_{ad}. Such preventive action can effectively address the large delay-bandwidth product case if we make back-pressure propagate from the congestion point all the way to the sources. Even if this is very difficult to implement, since routers do not have end-to-end information, back-pressuring aggregate input port traffic is enough to alleviate congestion in many instances as discussed in this paper.

3 THE NETWORK SCENARIO

We study the congestion control problem in the IP over ATM backbone scenario. Nonetheless, we are not actually assuming ATM connectivity end-to-end, rather, we assume that Internet users reside on legacy LANs. Edge routers, with ATM connectivity, function as ingress routers for the traffic crossing from the LAN to the backbone (and vice versa), see Figure 2. Hence, in our model, the edge routers are the effective sources and sinks of Internet traffic.

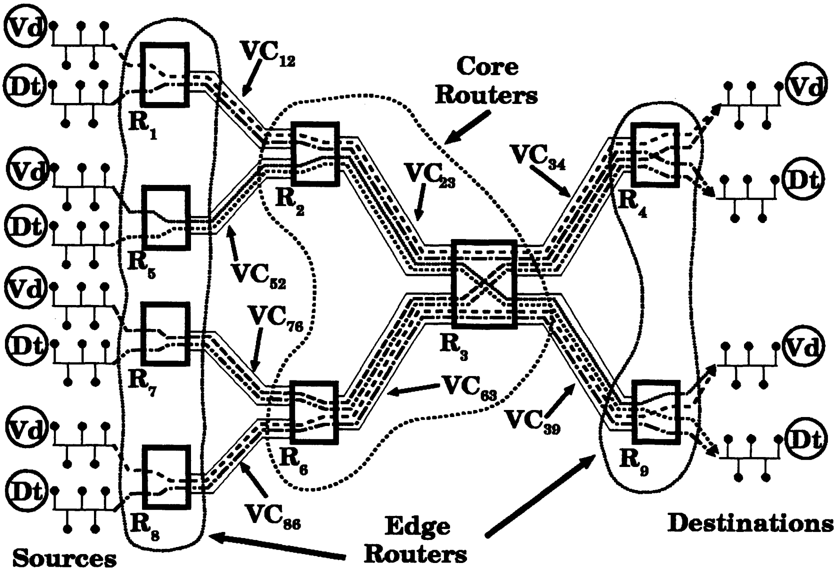


Figure 2 The network scenario.

As Figure 2 also illustrates, we consider a scenario in which the backbone transports Data (Dt) and video (Vd) traffic. In our simulation studies, we consider the Dt traffic being file transfers over TCP and the Vd traffic being compressed video sent over UDP. We place Vd and Dt sources and sinks in different legacy LANs for simplicity since we are not concerned with the individual traffic flows. Our interest is on the aggregate streams for each class on which we exercise back-pressure.

The Vd traffic is given priority over the Dt traffic on edge and core routers by a Class Based Queueing (CBQ) (Floyd *et al.* 1995) scheduler. We also assume that RSVP and some form of flow admission control are employed to allocate the Vd traffic flows appropriate bandwidth on VCs such that RT constraints can be guaranteed. Finally, we use the ABR service on the VCs interconnecting the routers in Figure 2 where a Minimum Cell Rate (MCR) is allocated to implement bandwidth guarantees to the backbone traffic. This approach, as described in (Pazos *et al.* 1997), improves ATM network utilization and throughput for the Internet traffic as compared to the more common approach of using the CBR service.

4 FLOW CONTROL APPROACH

Let us now elaborate a little further on the congestion issue we are addressing in this paper. First of all, Vd sessions are likely to last for more than a few minutes, they reserve resources in the network and they are given priority over the Dt traffic by the CBQ scheduler. Hence, in this scenario if the Vd traffic is well behaved or policed, it is unaffected by congestion in the network. On the other hand, TCP sessions are likely to be responsible for the majority of the BE traffic. The TCP flow control tests for congestion in the network by increasingly sending more packets. Packets must then be dropped for the transmission window to be reduced and the source rate to be controlled. However, the dropping of packets is very undesirable because it can lead to global synchronization and throughput collapse, and it compromises the performance of delay sensitive traffic (e.g., telnet) (Floyd 1995, Morris 1997).

Approaches such as Random Early Detection (RED) (Floyd *et al.* 1993) and Explicit Congestion Notification (ECN) (Floyd 1995) try to make the packet dropping more "effective and fair" and to enhance the congestion feedback, respectively. RED uses queue length information to randomly decide when to drop packets. ECN improves on RED by marking the packets RED would normally drop as a means to indicate congestion to the sources. In any case, the source's reaction to the congestion signal is felt at the congested router only after one RTT. Even then, the reduction in the source rate may not have been sufficient, in which case packets are still dropped and a new RTT has to go by before the rate is reduced again.

The equivalent of ECN has also been proposed in ATM under the name of FECN (Forward ECN) for the flow control of ABR connections (Newman 1994). In this case, the sources implement a multiplicative decrease of their sending rates upon congestion indication, and an additive increase when there is no congestion in the network. Such binary feedback may reduce implementation complexity, but it is well known to lead to unfairness, oscillation, and slow response to congestion (An *et al.* 1997). Hence, many ABR flow control schemes employ explicit rate indication (ATM Forum 1996, Jain *et al.* 1996) or relative rate indications (Chiussi *et al.* 1997), as a means to solve or at least alleviate these problems.

Challenges also arise in the transport of TCP traffic over ATM using the UBR service. The UBR traffic is not subjected to flow control at the ATM layer and cells are dropped when a congestion threshold is exceeded. In this scenario, the ATM switches usually employ Early Packet Discard (EPD) mechanisms (Romanow *et al.* 1994, Wu *et al.* 1997) to alleviated the source synchronization problem caused by tail dropping. Fairness problems also arise and selective dropping approaches such as the "Virtual Queueing" (Wu *et al.* 1997) address this issue. Since RED and ECN are also binary indication mechanisms which feedback congestion indications based on the state of a common FIFO queue, they are bound to experience similar fairness problems.

To sum up, the TCP flow control is reactive in nature with actions taken only after congestion sets in. Congestion notification based on packet drop is inadequate to preserve network throughput, to maintain fairness, or to deliver a quality service to delay sensitive traffic. Schemes are available to alleviate the consequences of dropping packets, but the response time is still in the order of RTTs. With high transmission rates becoming increasingly more common and with the number of simultaneous sessions rapidly increasing, such traditional approach may not react fast enough to alleviate congestion. In the next section we introduce a back-pressure approach with response time in the order of the RTT between the source and the point of congestion.

4.1 Back-pressure Flow Control

Our goal here is to address the congestion problem of Figure 1 as discussed in section 2. However, from now on we will consider the more general scenario of Figure 2 in which router R_3 can experience the congestion illustrated in Figure 1. The rationale of our approach is that since the VCs interconnecting the routers use the ABR service, the Internet traffic can make a better utilization of the available ATM resources (than with CBR) and still allow the routers to enforce a back-pressure flow control. In addition, routers at the end points of a VC function as virtual sources (VSs) and virtual destinations (VDs) for the aggregate traffic flowing between them through the VC. Figure 3(a) illustrates the VS and VD roles played by each router port as seen by the traffic flowing in the direction indicated.

Figure 3(a) illustrates how the ABR rate-based flow control is applied to each VC. A VS is continuously informed of the bandwidth available along its associated VC path through Explicit Rate (ER) indications on a stream of Resource Management (RM) cells (ATM Forum 1996). While ABR protects the VC path from congestion, it does not prevent congestion at the terminating router acting as VD. Hence, a back-pressure flow control can be implemented by having VDs reduce the ER indications on the RM cell stream, before sending them back to the VSs.

It is then critical to determine how to dynamically adjust the ER indications

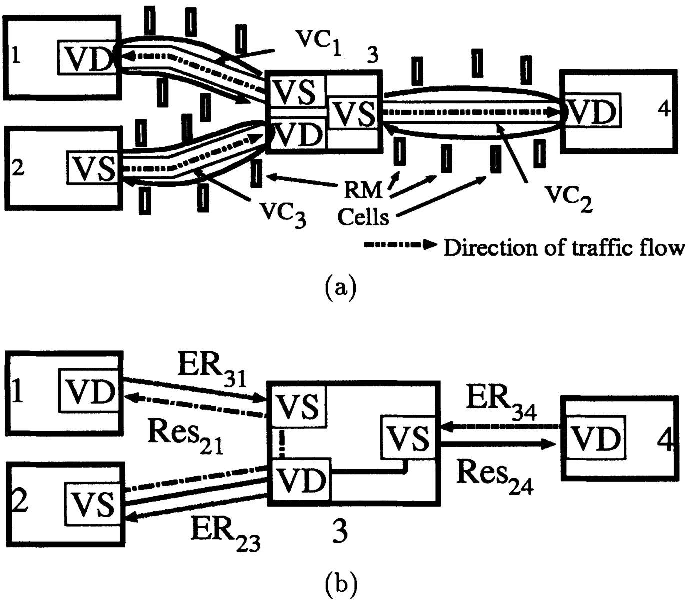


Figure 3 The ABR flow control applied to VCs.

and to assess the resulting impact on throughput, cell loss and fairness. Let us illustrate the issues involved by considering Figure 3(b). In this figure router 3 is an intermediary hop for the bidirectional traffic flowing between any two of the other routers shown. Let us further focus on the traffic from router 2 to routers 1 and 4. A portion of this traffic (RS) has made resource reservations Res_{21} and Res_{24} on the VCs traversed by the respective streams, as illustrated in Figure 3(b). These reservations are guaranteed by the MCR allocated to the VCs traversed by the RS flows and they are enforced by the CBQ scheduler on each output port. The traffic without reservation (NRS) is entitled to a fair share of the unused MCR bandwidth, the unassigned bandwidth and the bandwidth leftover by other ATM connection along the paths of the respective VCs. For the sake of argument, we further assume that ER_{31} and ER_{34} are the rates advertised by routers 1 and 4 and by the corresponding VCs. The issue is then to determine ER_{23} (i.e., the rate router 3 should advertise to router 2) such that the traffic from router 2 does not congest router 3.

In our approach, such back-pressure is actually provided as part of the VS and VD behaviors to be implemented by the ATM cards in the router ports. In (Pazos *et al.* 1997b), a VS and VD behavior was proposed for a simple tandem topology. Here we consider the more general topology of Figure 2 and we use the measured bandwidth utilization and the available bandwidth to slow down sending routers when their offered traffic would make downstream routers congested. In the following, we first disregard, for simplicity, the notion of selective back-pressure and at the end of this section we discuss the resulting implications on throughput and fairness.

4.2 The Virtual Source and Destination Behaviors

For max-min fairness (Bertsekas *et al.* 1992), we measure the NRS traffic contribution from each input port and we allocate fair shares to input ports based on whether they have traffic to use the fair shares. In our studies, we consider a variation of the approach in (Bertsekas *et al.* 1992), which involves measuring used and available bandwidth. This does not necessarily lead to more complexity in our scenario since our CBQ scheduler knows how much bandwidth has been assigned to each class and it measures how much bandwidth each class is actually using in order to enforce the link sharing policy. A minor modification can then be made to allow the scheduler to keep track of bandwidth usage by individual input ports.

Note, however, that the CBQ queues contain packets. Determining the input port a packet comes from may not be possible given that it would involve the same amount of processing overhead as that needed to make packet routing decisions. In our studies, though, we make the input ports stamp their own local identification on packets, using a sort of internal encapsulation header, before sending them to the output port which then strips this extra header off before sending the packet onto the outgoing link (such approach is not uncommon on the architecture of some ATM switches).

The only real issue is that this bandwidth bookkeeping has to be done at the IP layer (we implement it as part of the CBQ scheduler code) and periodically communicated to the ATM layer, as described below. In our implementation, the CBQ scheduler on each output port keeps track of the traffic offered (in bytes) by each class and each input port i (Dt_i and Res_i), and it keeps track of the traffic actually sent (Tf_{out}, also in bytes) through the output port during a fixed sampling period. Tf_{out} reflects the ER indication (the available bandwidth) for the outgoing VC, while MCR_{out} (in bytes) is the traffic that would be sent if ER = MCR. Finally, $RESV_i = \alpha_i \times MRC_{out}$ reflects the bandwidth reserved for the traffic from input port i.

The Virtual Source behavior is carried out by the ATM layer and the CBQ scheduler in the following way:

1. At the beginning of a new sampling period:

   (a) The CBQ scheduler computes the fair share fpr the Dt class as:

$$Av_{Dt} = Tf_{out} - \sum_{i}^{N} RESV_i, \qquad FS = \frac{Av_{Dt}}{N},$$

Av_{Dt} is the available resources (in bytes) for the Dt traffic and N is the number of input ports contributing traffic to the output port.

(b) If an input port offered traffic is such that $Dt_i + Res_i <$ FS $+RESV_i$, the traffic from input port i is constrained somewhere else and the respective input port is marked as unconstrained.

(c) The remaining input ports j are constrained on the router and they are assigned an actual share:

$$AS_j = \frac{\dfrac{Tf_{out} - \sum_{i}^{N_{un}}(Dt_i + Res_i) - \sum_{k}^{N-N_{un}} RESV_k}{N - N_{un}} + RESV_j}{Tf_{out}}$$

N_{un} is the number of input ports unconstrained at the output port.

(d) Less obvious is the possibility that $Dt_i + Res_i > AS_i \times Tf_{out}$. This means that input port i is actually sending too much traffic to the output port. Let $Ex_i = Dt_i + Res_i$ - $AS_i \times Tf_{out}$ be the excess traffic sent from input port i. Since this excess traffic on the previous sampling time was caused by reasons outside the scope of the output port, we expect at least another Ex_i in the next sampling period. Hence, we make:

$$AS_i^{(new)} = \frac{AS_i^{(old)} \times Tf_{out} - 2Ex_i}{Tf_{out}}.$$

(e) If an input port is constrained somewhere else, assigning it its fair share does not congest the router while the input port is constrained. However, when the input port is no longer constrained somewhere else before getting to the router, this allows the corresponding traffic to reclaim its fair share. Hence, all unconstrained input ports are assigned an actual share:

$$AS_i = \frac{FS + RESV_i}{Tf_{out}}$$

(f) Finally, the CBQ scheduler sends the vector AS to the ATM layer through a control primitive.

2. At the ATM layer, a flow of backward RM cells (ATM Forum 1996) is continuously being returned as part of the ABR flow control. The ER indication on these cells advertise the bandwidth actually available along the VC and supported by the router on the other end. Hence, upon the arrival of a new forward RM cell, the ATM layer uses the current vector AS and sends to the ATM interface on each input port i in the router a

control message with AS_i*RM.ER as the bandwidth available for the input port i to send traffic to the output port.

The Virtual Destination behavior carried out by the ATM layer on input ports is:

1. A vector Bw_i keeps track of the bandwidth available for the input port traffic on each other output port i, and an entry Bw_i is updated every time a new control message arrives from the output port i.
2. Whenever a forward RM cell arrives, we test if RM.ER is smaller than the sum of all Bw_i. If it is, the ER indication is left intact, otherwise it is replaced by the sum of all Bw_i.
3. The RM cell is sent back to the VS as a backward RM cell.

4.3 Max-min Fairness Considerations

In our simulation results we demonstrate the effectiveness of this approach to implement back-pressure and to achieve max-min fairness. Now, let us elaborate on the implications to the congestion control problem of Figure 1. If we apply the VS and VD behaviors above to the scenario of Figure 1, we would get for instance an ER_b, the bandwidth available to the router feeding link b. However this ER indication does not reflect AS_{bc} and AS_{bd}. Rather it reflects $\mathrm{AS}_{bc} + \mathrm{AS}_{bd}$ and the sending router has no way of enforcing AS_{bc} and AS_{bd} selectively. If λ_{bc} and λ_{bd} are smaller than or equal to AS_{bc} and AS_{bd}, respectively, we have no problems.

However, if both flows λ_{bc} and λ_{bd} have enough traffic load, they would be assigned half of $\mathrm{AS}_{bc} + \mathrm{AS}_{bd}$, which is clearly a problem if $\mathrm{AS}_{bc} \neq \mathrm{AS}_{bd}$. In this case, it is easy to see that one link will get congested while the other will be under utilized. However, with appropriate buffering, we can accommodate the excess traffic and, from the VS behavior action 1(d), we try to alleviate this effect by reducing the actual share for the input port creating congestion.

4.4 Explicit Rate and the Last Hop

As mentioned earlier, the Explicit Rate indications provided by ABR cannot be communicated to the TCP sources without ATM connectivity "to the desk top". The issue is then the usefulness of the ER indications if the actual sources do not receive this information. This problem was already addressed in (Kalyanaraman *et al.* 1996, Narcáez *et al.* 1997). Since, with the arrival of acknowledgments, the TCP sources can double their transmission window and send more packets, and since the sources connected to a Legacy LAN are not subjected to any flow control other than TCP, the traffic will reach the edge router at a rate higher than that supported by the ATM network.

If the number of active TCP sessions is high, one expects that the edge router may run out of buffers and may start dropping packets. On the positive side, no network resource is consumed by packets that would be eventually dropped inside the network. An alternative approach proposed in (Kalyanaraman *et al.* 1996) consists on equipping edge devices with enough buffers to hold many TCP-receiver-windows worth of packets.

A more interesting idea was presented in (Narcáez *et al.* 1997) as a means of "effectively extending the ABR flow control all the way to the TCP source". In their approach the edge router uses an acknowledgment bucket that translates the ER indications into a sequence of TCP acknowledgments to prevent the TCP source from sending at rates higher than the ER indications. A potential difficulty with this approach is that the TCP acknowledgments may in fact be returned piggy-backed on the packets from the reverse data flow. Thus, implementing this approach may require altering the acknowledgment field in the TCP header.

Another approach would use ECN (Floyd 1995). Again, with the arrival of acknowledgments the sources would increase their transmission windows and the buffers on the edge routers would fill up. Hence, using ECN in the **forward** direction has the same problems as discussed in section 4. However, we can stamp the ECN bit on acknowledgment packets **returning** through the edge device.

5 SIMULATION RESULTS

In this section we present simulation results to illustrate the effectiveness of our back-pressure approach. For our study, we used the topology illustrated in Figure 2, in which all links are 150Mbps, the propagation delay on the indicated VCs is 400μs, and the propagation delay from a host to an edge router is 10μs. Each Vd stream is composed of seven individual H.261 video (Turletti *et al.* 1996) flows transported over UDP at a target rate of 1.5Mbps and with 15 consecutive H.261 frames sent on every IP packet. Each Dt stream is made up of ten individual ftp session transferring large files (persistent sources). The routers have 100-packet and 30-packet queues for the aggregate Dt and Vd streams, respectively. Our simulator code implements the TCP Tahoe version and we consider a maximum segment size of 1024 bytes.

Furthermore, each VC in Figure 2 traverses two ATM switches (not shown to avoid overloading the figure) and one of the links traversed by each VC is shared by other CBR traffic. We chose the load for the background CBR traffic such that the residual bandwidth C_{ij} available along the VC_{ij} paths in Figure 2 is 70 Mbps, except for C_{23} and C_{63} which are 95Mbps. The MCR_{ij} allocated to the VC_{ij}s terminating in R_3 is 40Mbps, the others are allocated an MRC of 20Mbps. In our experiments, we modify the CBR load to test different congestion scenarios. Finally, the MCR_{ij} for a VC is taken by the CBQ scheduler to be the link bandwidth for the purpose of imposing the link sharing policy of 70% of MCR dedicated to the video traffic, and 30% dedicated to the data traffic. The data traffic can of course use all of the bandwidth actually available on the links.

5.1 Using Back-pressure

We study the effectiveness of our scheme using the traces in Figures 4 and 5. These traces show the Dt, Vd and Total load offered to the VC connected to the routers indicated. Figure 4 plots the traces for the core routers, while Figure 5 plots the traces for the edge routers of Figure 2. All connections start at different times, but the actual starting time is uniformly distributed over an initial time interval.

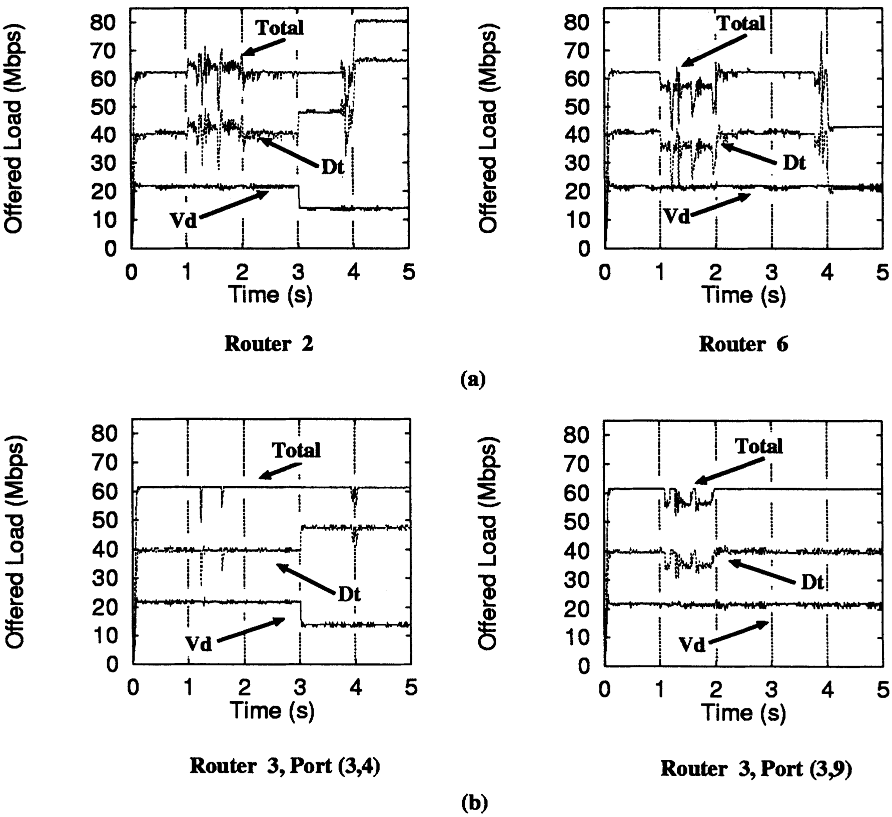
(a) A Load Balanced Scenario

We study a different load scenario in each of the five seconds of Figures 4 and 5. So, **before $t = 1$s**, we have all connections active, and the initial available bandwidth on the links are the ones indicated above. However, as the traces in Figure 4(b) indicate, the actual rate available on the respective links is roughly 60Mbps, not the 70Mbps mentioned above. This is so because we plot the effective rate, discounting all the different protocol (UDP, TCP, IP and ATM) overheads. Note that 60Mbps is also the actual rate at which R_2 and R_6 transmit to R_3, even though the VCs connecting them to R_3 have 95Mbps of available bandwidth. This means that the VD entities on R_3 are appropriately reducing the ER indication they receive on forward RM cells, before returning them to the VS entities in the sending routers. Hence, R_3 is effectively back-pressuring R_2 and R_6.

If we now look at the traces in Figure 5, we see that each edge router is roughly transmitting at a maximum aggregate rate of 30Mbps, when the bandwidth on the VCs connecting the edge routers to the core routers R_2 and R_6 is actually 70Mbps. The reason is that the traffic from R_1 and R_5 and from R_7 and R_8, must share VCs with 60Mbps bandwidth constraints from routers R_2 and R_6 to router R_3, respectively. With this, we observe that the core routers effectively back-pressure the edged routers.

(b) The Effect of Unbalanced Load

At **time $t = 1$s**, we simulate the scenario in which the CBR traffic is increased on the trunk (8,6) thus reducing C_{86} to roughly 30Mbps. As Figure 5(b) shows, the Dt traffic through R_8 is reduced accordingly while the Vd traffic is unaffected since we have the CBQ scheduler. Moreover, for R_6, the traffic stream fed by R_8 becomes constrained somewhere else. Thus the VS on R_6



Router 2

Router 6

(a)

Router 3, Port (3,4)

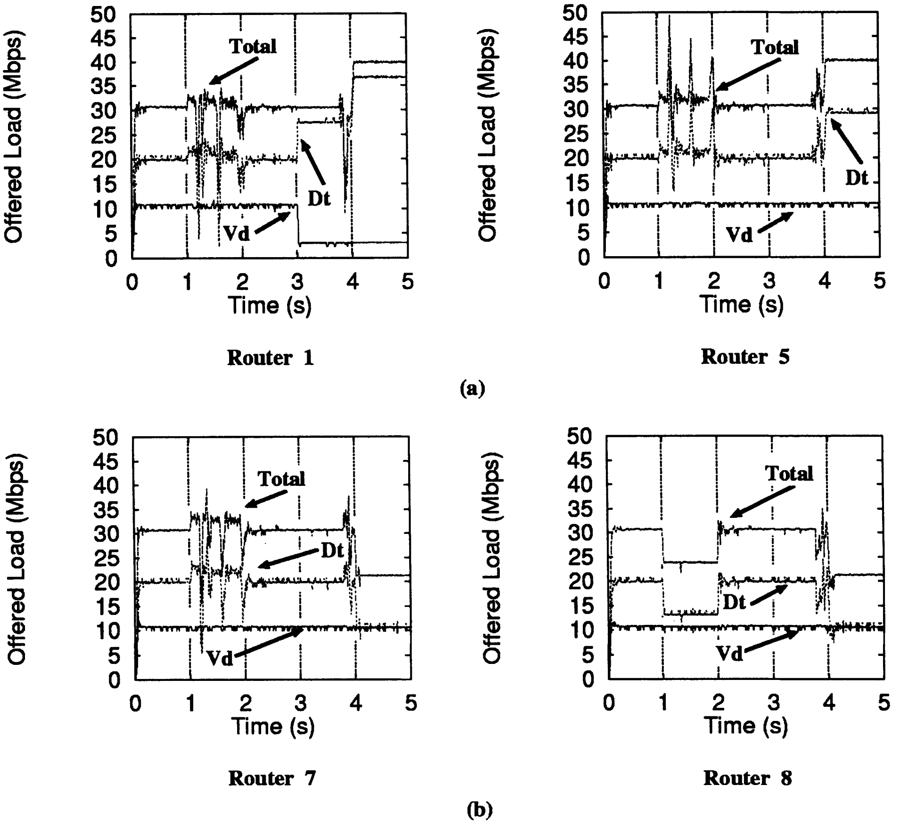
Router 3, Port (3,9)

(b)

Figure 4 The load offered by core routers.

increases the Actual Share (AS) of the VC_{63} bandwidth to the input port fed by R_7. This is consistent with the max-min fairness policy. However, the bandwidth left over by the Dt traffic from R_8 cannot be claimed by the Dt from R_7 because, as Figure 2 indicates, these streams only share VC_{63}. The Dt from R_7 is still constrained by the bandwidth available on VC_{34}. Hence, some of the excess packets are unavoidably lost on R_3.

Then, after the reduction of the traffic load from R_8 is felt on R_3, the VS acting on VC_{39} deems VC_{63} constrained somewhere else and it increases the Actual Share for R_2, again in an attempt to implement max-min fairness. However, the newly available bandwidth is split in R_2 between the traffic from R_1 and R_5. The traffic from R_5 could recover the bandwidth released by R_8, but we cannot enforce this in R_2 and the result are the oscillation we see in the interval [1,2]. The excess traffic sent from R_5 goes through unharmed



Router 1

Router 5

(a)

Router 7

Router 8

(b)

Figure 5 The load offered by edge routers.

while the excess traffic from R_1 collides with the excess traffic from R_7 and losses cannot be prevented if the scenario persists for a long time.

(c) Using the Allocated Fair Shares

At **time** $t = 2\mathrm{s}$, the CBR extra load on trunk (8,6) is removed, C_{86} returns to its original value and the Dt traffic from R_8 recovers the bandwidth that is available along the path to R_9. This is due to the fact that the R_3 output port (3,9) regarded the input port (6,3) constrained but it still allocated this input port its fair share (see VS behavior 1(e)). So, when the R_8 traffic started using its fair share, the AS for input port (2,3) is also brought back to the fair share and the oscillations present in the interval [1,2] are terminated.

At **time** $t = 3\mathrm{s}$ we illustrated another type of traffic load variation: the aggregate Vd traffic load offered through R_1 is reduced. This simulates the case in which some of the Vd sessions are closed. However, this does not reduce the reservations that are made on CBQ schedulers for the Vd flows. As a result, the Dt traffic from R_1 benefits from the bandwidth released by the Vd sessions.

(d) Testing Max-min Fairness

Finally, at **time t = 4s** another increase in the CBR load reduces now the bandwidth C_{63} to 50Mbps. As Figure 4(a) illustrates, the Dt traffic from R_6 is reduced accordingly. With the VC_{63} constrained by the excess CBR traffic, the BE traffic from R_7 and R_8 are also reduced in response to the back-pressure from the VS in R_6. The Dt reduction affects the loads on VC_{34} and VC_{39}. In particular, the respective output ports regard the input port (6,3) constrained and they increase the AS for input port (2,3). This extra bandwidth is then notified by the back-pressure mechanism to R_1 and R_5 which can use the extra bandwidth. Hence their load is adjusted accordingly as illustrated in Figure 4(a), for core router R_2, and in Figure 5(a), for edge routers R_1 and R_5.

5.2 Comparing Results With and Without Back-pressure

The simulation results presented so far were designed to illustrate some of the features as well as limitations of our back-pressure control. Next, we compare these results with those of a network without back-pressure. For the latter case, we present the simulation results in Figure 6 for edge routers only. First of all, the TCP sources become active with a slow-start and the offered traffic increases exponentially (doubling every RTT). Since we no longer employ the back-pressure, this traffic is always admitted into the network. Hence, the aggregate offered traffic achieves a 50Mbps peak **after t = 0s** and many packets are dropped.

Comparing these results with the results in Figure 5, it is clear that back-pressure is effective in avoiding these losses. Furthermore, since TCP sources continuously increase their transmission windows, packets are periodically lost and the windows shrink to W = 1. This accounts for the oscillations throughout the simulated time. Since these losses are prevented by our back-pressure, the traces in Figure 5 are smoother.

At t = 1s, the available rate on VC_{86} is reduced. This affects only the traffic from R_8, which drops many packets and keeps a persistent backlog allowing full utilization of the available bandwidth on VC_{86} (Figure6(b)). Note, though, that there is no side effects on the traffic offered by R_1 and R_7 **through t = 3s** since their respective flows do not share a common path with traffic from R_8. The traffic from R_5, on the contrary, can claim the bandwidth leftover. This is the type of scenario in which the back-pressure would affect the offered traffic from sessions not affected by the congestion.

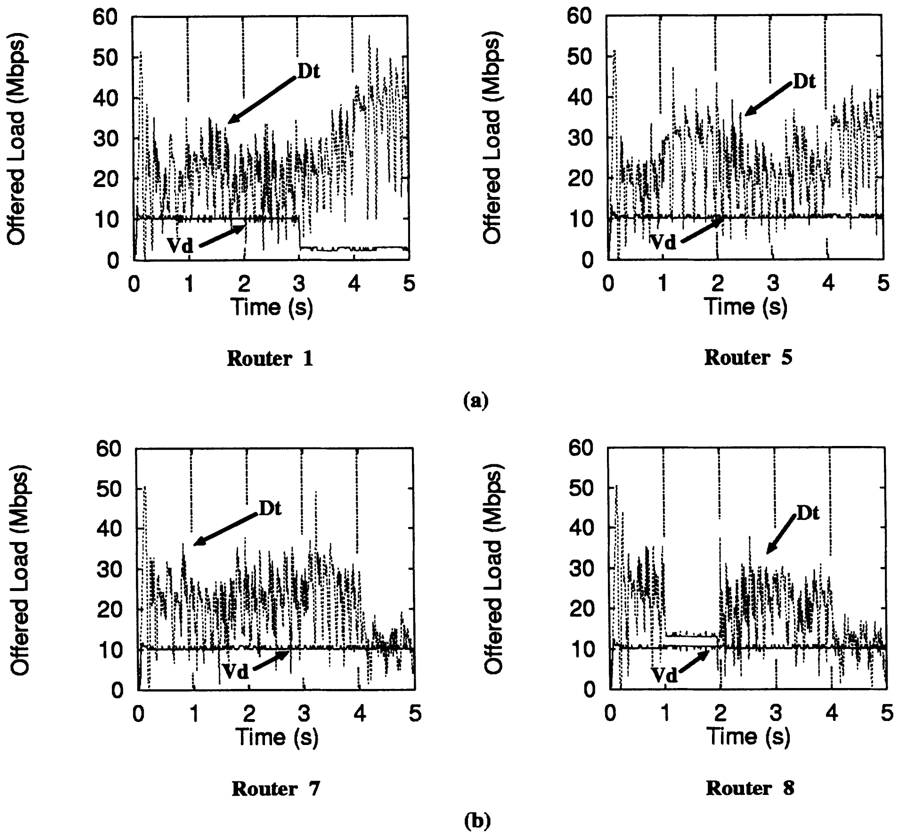


Figure 6 The load offered by edge routers under no back-pressure.

At t = 2s, the available rate on VC_{86} is increased again and the backlog in R_8 is dumped into R_6, causing buffer overflow in R_3. Here again, in the back-pressure scenario of Figure 5, the new offered load reclaims its fair share while the load from R_5 gets back-pressured and these losses are avoided. Finally, note that in the interval [2,3] the bottleneck for all connections are VC_{34} and VC_{39}. But at t = 3s, the reduction on the Vd load through R_1 allows the Dt streams from R_1 and R_7 to claim the leftover bandwidth. However, since the Dt stream from R_8 cannot claim this newly available bandwidth, this traffic suffers with the extra traffic sent from R_7. With the use of back-pressure this is avoided.

5.3 Other Performance Metrics

In Table 1 we present the number of packets lost and the aggregate throughput accumulated over the five-second interval considered in the above study. As these results show, the use of back-pressure is effective in reducing the number of packets lost. The throughput performance, on the other hand, is almost identical. The reason for this is that the number of active TCP sessions is large compared to the buffering and delay-bandwidth product. This is an indication that in the presence of a vary large number of sessions, it is not likely that all of them will be periodically synchronized leading to the loss of TCP throughput, which RED and ECN try to avoid. However, average TCP throughput alone does not tell the whole story. The bursty behavior observed in Figure 6 implies considerable fluctuations in TCP end-to-end delays over individual sessions, which is a serious performance problem for delay sensitive applications such as telnet and web browsing. The study of these effects is left for future research.

Table 1 Packet loss and aggregate throughput.

Metric	No back-pressure	Back-pressure
Packet Losses	1957	135
Aggregate Throughput	79.27Mbps	79.66Mbps

6 CONCLUSIONS AND FUTURE RESEARCH

In this paper we address the effectiveness of the TCP flow control to deal with large delay-bandwidth products and with very many simultaneous connections (Morris 1997) observed in Internet backbones. Since ATM has been extensively used as a link layer for Internet backbones, and this trend is likely to persist for the near future, we address this congestion control problem in the IP over ATM scenario. Our approach follows the suggestions in (Pazos *et al.* 1997) to use the ABR service in the backbones and we define an appropriate Virtual Source and Virtual Destination behavior for the IP routers terminating the ABR control loop. Such approach effectively implements a back-pressure mechanism that addresses max-min fairness. In addition, it is also more effective in handling the large RTTs and the large number of TCP sessions because we back-pressure from the point of congestion to the sources and because we back-pressure the aggregate traffic between routers.

As for future research, we are looking into the use of Label Swapping Routers (LSR) over ATM to implement this back-pressure mechanism. The use of LSR should lead to a more elegant mechanism because all queueing, scheduling and bandwidth bookkeeping can be done entirely at the ATM layer. This removes the need for passing control messages across layer boundaries as described in section 4.2 as part of the Virtual Source behavior.

ACKNOWLEDGMENTS

This research was partially funded by GTE, NSF and CNPq.

REFERENCES

Braden, R. and Clark, D. and Shenker, S. (1994) Integrated Services in the Internet Architecture: an Overview. *Request for Comments 1633.*

Morris, R. (1997) TCP Behavior with Many Flows. *Proc. of ICNP '97.*

Pazos, C. M. and Gerla, M. (1997) ATM Virtual Private Networks for the Internet Data Traffic. *Proc. of MMNS '97.*

ATM Forum Technical Committee (1996) Traffic Management Specifications, Version 4.0.

Paxson, V. (1993) Empirically-Derived Analytic Models of Wide-Area TCP Connections: Extended Report. *Lawrence Berkeley Laboratory Technical Report.*

Jain, R. and Kalyanaraman, S. and Goyal, R. and Fahmy, S. and Viswanathan, R. (1996) ERICA Switch Algorithm: A Complete Description. *ATM Forum Contribution.* **AF-TM 96-1172**.

Floyd, S. (1995) TCP and Explicit Congestion Notification. *ACM Computer Communication Review,* **24(4)**.

Roche, C. and Plotkin, N. (1995) The Converging Flows Problem: an Analytical Study. *Proc. of INFOCOM '95.*

Floyd, S. and Jacobson, V. (1991) Traffic Phase Effects in Packet-Switched Gateways. *ACM Computer Communication Review,* **21(2)**.

Floyd, S. (1991) Connections with Multiple Congested Gateways in Packet-Switched Networks Part 1: One-way Traffic. *ACM Computer Communication Review,* **21(5)**.

Floyd, S. and Jacobsen, V. (1995) Link-sharing and Resource Management Models for Packet Networks. *IEEE/ACM Transactions on Networking,* **3(4)**.

Floyd, S. and Jacobsen, V. (1993) Random Early Detection gateways for Congestion Avoidance. *IEEE/ACM Transactions on Networking,* **1(4)**.

Newman, P. (1994) Traffic Management for ATM Local Area Networks. *IEEE Communications Magazine,* **32(8)**.

An, L. and Ansari, N. and Arulambalam, A. (1997) TCP/IP Traffic over ATM Networks with ABR Flow and Congestion Control. *Proc. of GLOBECOM '97.*

Chiussi, F. and Xia, Y. and Kumar, V. (1997) Virtual Queueing Techniques for ABR Service: Improving ABR/VBR Interaction. *Proc. of INFOCOM '97.*

Romanow, A. and Floyd, S. (1994) Dynamic of TCP Traffic Over ATM Networks. *Proc. of SIGCOMM '94.*

Wu, Y. and Siu, K-Y. and Ren, W. (1997) Improved Virtual Queueing and Dynamic EPD Techniques for TCP over ATM. *Proc. of ICNP '97.*

Pazos, C. M. and Gerla, M. (1997b) Bandwidth Efficiency in Internet AVPNs. *Proc. of GLOBECOM '97.*

Bertsekas, D. and Gallager, R. (1992) Data Networks. *Prentice Hall.*

Kalyanaraman, S. and Jain, R. and Fahmy, S. and Goyal, R. and Jiang, J. (1997) Performance of TCP over ABR on ATM Backbone and with Various VBR Traffic Patters. *ATM Forum Contribution 96-1294.*

Narcáez, P. and Siu, K-Y. (1997) An Acknowledgment Bucket Scheme for Regulating TCP Flows over ATM. *Proc. of GLOBECOM '97.*

Turletti, T. and Huitema, C. (1996) RTP Payload Format for H.261 Video Streams. *Request for Comments 2032.*

BIOGRAPHY

Carlos Marcelo Dias Pazos received his graduate degree in Electrical Engineering and his M. S. degree in Computer Science from Federal University of Pernambuco, Brazil in 1990 and 1993, respectively. Since 1993, he has been a Ph. D. student with the Computer Science Department at University of California, Los Angeles. His research interests include high speed communication systems and networks, ATM networks, multi-service Internet, and Internet over ATM.

Dr. Mario Gerla received his graduate degree in Electrical Engineering from Politecnico di Milano, Italy in 1966 and his M. S. and Ph. D. degrees in Computer Science from University of California, Los Angeles in 1970 and 1973, respectively. ¿From 1973 to 1976, Dr. Gerla was a manager in Network Analysis Corporation, Glen Cove, NY, where he was involved in several computer network design projects for both government and industry, including performance analysis and topological updating of the ARPANET under a contract from DoD. From 1976 to 1977, he was with Tran Telecommunication, Los Angeles, CA, where he participated in the development of an integrated packet and circuit network. Since 1977, he has been on the Faculty of the Computer Science Department of UCLA. His research interests include the design, performance evaluation, and control of distributed computer communication systems and networks. His current research projects cover the following areas: topology design and bandwidth allocation in ATM networks; design and implementation of optical interconnects for supercomputer applications, and; network protocol design and implementation for a mobile, integrated services wireless radio network.