Backward Explicit Congestion Notification for ATM Local Area Networks*

Peter Newman†
N.E.T. Adaptive Division

ABSTRACT

The application of backward explicit congestion notification (BECN) to an ATM LAN is investigated in order to support the dynamic sharing of bandwidth between high-speed, bursty, data traffic sources. The results of a simulation study are presented which suggest that a BECN mechanism may provide simple and effective traffic management for an ATM LAN or campus backbone network of up to at least 50 km diameter. Cell loss due to congestion may be prevented and a link utilization in excess of 80% may be maintained with only 0.08% of the link bandwidth, per active source, carrying backward congestion notification traffic during a congestion event.

1 Introduction

Over the last few years the physical topology of local area networks has migrated from the ring and the multidrop bus toward a star configuration — the hub. A star topology is easier to manage and offers higher reliability. However, the technology remains shared medium, resulting in the "LAN-in-a-box". As the power of the desktop workstation increases, so the capacity of the LAN must increase in proportion. Shared medium access in the Gb/s range is significantly more expensive than access at around 100 Mb/s or so. Thus pressure is mounting for the high-performance LAN hub to abandon its internal shared medium architecture and adopt a switched design.

In addition to the requirement for increasing local area bandwidth is the desire to support multimedia applications. These will require the integration of video, voice, image, and data traffic. There is also a significant requirement to achieve closer interworking between local area and wide area networks. ATM has been proposed as a technology capable of offering high capacity switching and supporting multiservice traffic. It has received much attention in the public wide area network as the future broadband integrated services digital network (B-ISDN). ATM is therefore an obvious candidate for the high-performance LAN and campus backbone network [8].

2 Traffic Management in an ATM LAN

In a shared medium LAN, be it ring, bus, or backplane, all of the attached stations share a single resource — the shared medium. To transmit, a station contends for access to the shared medium via the medium access (MAC) protocol. If the network is heavily loaded it will apply backpressure, through the MAC protocol, to stations requesting access. Once a station has successfully gained access to the shared medium it may transmit its data without fear of causing network congestion since all stations can receive at the data rate of the shared medium. But the bandwidth of the shared medium has now become a bottleneck. In seeking a switched solution we are replacing the shared medium by a switch fabric of greater aggregate capacity than any individual input or output port. In so doing, we have removed the MAC protocol, removed the backpressure that the MAC protocol provides, and opened up the network to the possibility of congestion if too much traffic is directed at any single output port over a short time interval.

A number of schemes have been proposed for traffic management in an ATM network [5]. The most frequently discussed approach is to determine the traffic characteristics of the source and to allocate resources accordingly when the call is admitted to the network [1–4]. This approach is best suited to traffic sources that can be accurately characterized in advance of transmission, for example voice and compressed video. Individual data sources are very difficult to characterize in this manner. If we seek to provide a service with the characteristics of a LAN then, at least for data traffic, we must permit each source to burst at high speed and dynamically share the available capacity between active sources.

Within a LAN or campus network based upon ATM the propagation delay across the network is low. So a simple feedback scheme between the point of congestion and the source may be employed to replace the backpressure mechanism that the MAC protocol provided in a shared medium LAN. Traffic sources with well defined characteristics may be handled by call admission and bandwidth reservation. Such traffic is transmitted at a higher delay priority and is not subject to the backpressure mechanism [3]. The remaining bandwidth is then available to be shared dynamically between those sources that require a LAN-like (best-effort) service. Sources that require this service must be subject to the backpressure mechanism in the same way that sources must conform to the MAC protocol if they wish to attach to a LAN.

This paper presents simulation results of a backward explicit congestion notification (BECN) mechanism. When a queue in an ATM switch exceeds a threshold it sends congestion notification (BECN) cells back to the sources of the virtual channels currently submitting traffic to it. On receipt

*To be presented at IEEE Globecom, Houston, TX, Dec. 1993.
†<newman@adaptive.com>

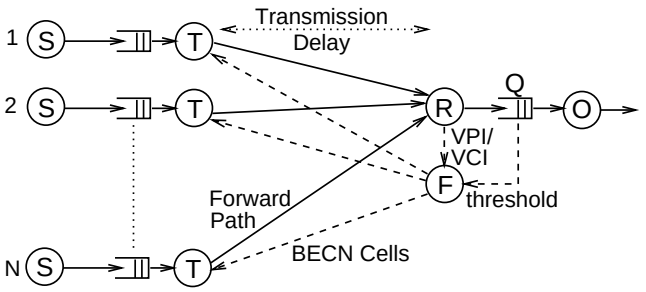

Figure 1: The simulation model.

of a BECN cell on a particular virtual channel, a source must reduce its transmission rate for the indicated virtual channel. If no BECN cells are received on a particular virtual channel for a certain period of time, a source may gradually restore its transmission rate on that virtual channel.

3 The Simulation Model

The simulation model is illustrated in fig. 1. Each source (S) generates traffic at saturation so that the queue feeding its transmitter is never empty. Each of the transmitters (T) independently removes cells from its source queue and transmits them to the receiver (R) at the peak cell rate. There is a transmission delay between each transmitter transmitting a cell and the cell being received by the destination receiver (R). The transmission delay represents the combination of the propagation delay and the switching delay from the source to the point of congestion. For simplicity all sources are assumed to experience the same transmission delay.

If the length of the destination queue (Q) exceeds a threshold, the filter (F) will generate BECN cells. With no filtering, one BECN cell will be generated for each incoming cell and returned to the source transmitter (T) of the incoming cell. This is a very simple mechanism to implement in hardware — simply latch the VPI/VCI of the incoming cell; copy it into the header of the fixed BECN cell format; and insert the resulting BECN cell into the cell stream in the reverse direction. It will return to the source because VPI/VCI values are identical in both directions on a virtual connection. Two filter designs are investigated to reduce the amount of BECN traffic while maintaining an acceptable performance. No limit was placed on the size of the destination queue (Q). Under stable conditions the BECN mechanism limits the maximum length of the queue so cells are never dropped at the destination queue. The destination queue server (O) removes one cell from the queue every cell time (i.e. all measurements are normalized to the rate of the destination queue server which represents the output link rate). BECN cells are subject to the same transmission delay in the return direction as cells in the forward direction.

When a transmitter (T) receives a BECN cell it will reduce its cell transmission rate to one half of the current rate. If further BECN cells are received it will ignore them until it has transmitted at least one cell in the forward direction. Successive BECN cells will cause a transmitter to reduce its cell transmission rate to: 50%, 25%, 12%, 6%, after which a further BECN cell will cause it to stop transmitting. A transmission rate recovery mechanism is built into each transmitter. If no BECN cells are received within the source recovery time period, the current transmission rate for that transmitter will be restored to the previous level, once each recovery time period, until the transmission rate is restored to its original peak value. This algorithm is not difficult to integrate into the AAL segmentation and reassembly silicon [10].

4 Simulation Results

For all of the simulations reported the arriving traffic load exceeds the rate at which the server (O) can remove traffic from the queue, so the system is in saturation. When the system is in saturation the length of the queue grows until it passes the threshold. BECN cells are then generated that reduce the amount of arriving traffic. The queue length then declines until the sources recover, increase the incident traffic, and the queue length once more increases. Thus at saturation the length of the queue oscillates. An analytical study of an alternative BECN mechanism in which the queue oscillates at saturation is presented in [3]. Performance at saturation gives some insight into the interaction of the various system parameters and illustrates the steady-state performance.

For all of the measurements reported, the congestion control algorithm prevented any loss of traffic from the destination queue. The peak transmission rate and amount of BECN traffic are expressed as a percentage and normalized to the output cell rate of the destination queue server (O). All delays are expressed in cell times normalized to the output rate of the destination queue server (at 155 Mb/s one cell time is about 2.7 μs). After reaching steady-state each simulation ran until the output server had transmitted 2 million cells. This represents a simulated time of about 5 seconds at 155 Mb/s.

4.1 No Filter

The performance of the system in the absence of a BECN cell filter is discussed in [9]. In summary, without a filter the system transmits more BECN traffic than is necessary. If the peak transmission rate of all sources is limited to a maximum of 25%, acceptable performance may be attained for limited transmission delays. (Less than 20% BECN cells and a maximum queue length of 500 cells for transmission delays of up to 30 cell times). This is acceptable for a single switch LAN but the addition of a BECN cell filter offers greatly improved performance.

4.2 A Simple Filter

In the following simulations the performance of a simple BECN cell filter is investigated. When the destination queue is past threshold the filter permits one BECN cell to be generated every F cell times. If the filter permits the generation of a BECN cell, the next incoming cell will generate a BECN cell, after which no further BECN cells will be generated until F cell times have passed. This filter may be implemented

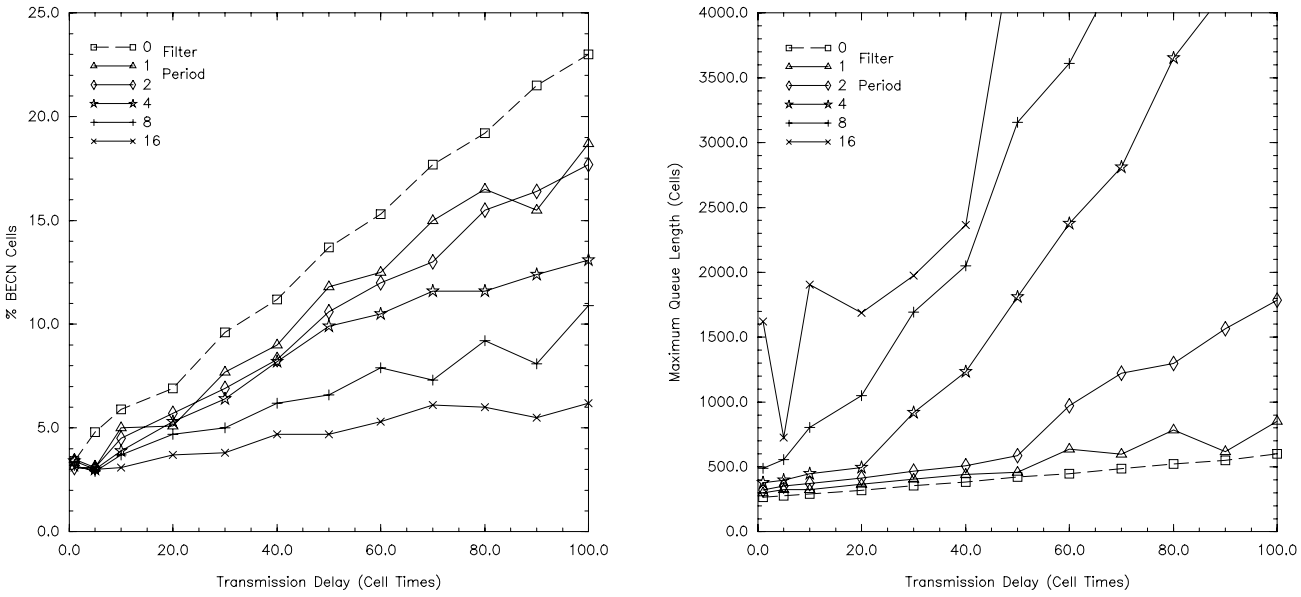

Figure 2: BECN cells and maximum queue length against transmission delay for different filter periods with a simple BECN cell filter and divide-by-two rate throttling.

in hardware with little more than a flip-flop, and a clock to define the filter period F.

The BECN traffic and maximum queue length against transmission delay are given in fig. 2 for 20 sources at a peak transmission rate of 25% with different filter periods. The recovery time constant of each source is 128 cell times and the queue threshold 250 cells. With no filter (a filter period of zero) the throughput is about 70%. As the filter period is increased the throughput increases, due to an increase in the mean queue length, and the BECN cell traffic is reduced. The rate of increase of the queue length with transmission delay increases as the filter period is increased. Depending upon the acceptable maximum queue length and the anticipated transmission delay, a filter period of up to 8 gives improved performance, compared to the performance with no filter, at a peak transmission rate of 25%. At a peak rate of 100%, however, with 20 sources at saturation, a BECN cell filter causes the maximum queue length to grow too rapidly to be useful even for filter periods as small as 1.

In employing the simple BECN cell filter we are reducing the amount of feedback information getting back to the sources and thus increasing the time taken for the sources to respond to an overload event. The time spent in overload is increased, thus the mean and maximum queue lengths are increased, and the period of oscillation at saturation is also increased. The increased mean queue length and the increased period of oscillation imply that the queue empties less frequently which explains the improved throughput with increased filter period.

The simple BECN cell filter with divide-by-two rate throttling does not reduce the incoming traffic quickly enough if sources are transmitting at a peak rate of 100%. One way to deal with this is to increase the effect of each BECN cell. This may be achieved by causing each BECN cell to throttle the source all the way down to zero and letting the source recover in the usual manner. Curves of BECN traffic and maximum queue length for this manner of operation are given in fig. 3 for 5 sources transmitting at a peak rate of 100% with different filter periods. The recovery time constant of each source is 256 cell times and the queue threshold 250 cells. Filter periods between 2 and 8 achieve a similar throughput (in excess of 80%) and maximum queue length. Filter periods of 4 and 8 will limit the BECN traffic to below 10% for a transmission delay of up to 100 cell times. This configuration works well if sources prefer to transmit at a peak rate of 100%. It generates less BECN traffic and is capable of operation at higher transmission delays than the divide-by-two throttling scheme with a simple BECN filter. However, it can only cope with a limited number of sources simultaneously bursting at a peak rate of 100% before the maximum queue length increases to an unacceptable value even at reasonable transmission delays.

4.3 A Per Virtual Channel Filter

A major problem with the simple filter is that many of the BECN cells generated are redundant and are discarded by the source because it has very recently been informed of the congestion event by a previous BECN cell. The optimum design of filter is one that transmits only a single BECN cell to each active source during each filter time period if the queue is congested. The optimum filter period is of the same order of magnitude as the maximum propagation delay for which the system is designed. This is because there is little point in sending further control feedback to the sources until previous feedback has had time to take effect. If the source recovery time constant is made slightly longer than the filter period, during overload each source will reduce its transmission rate by half every filter period until the total incident traffic is reduced below the available bandwidth. To ensure fairness between all sources, the source recovery period should be proportional to the transmission rate (throttling level) so that the lower the transmission rate the shorter the source recov-

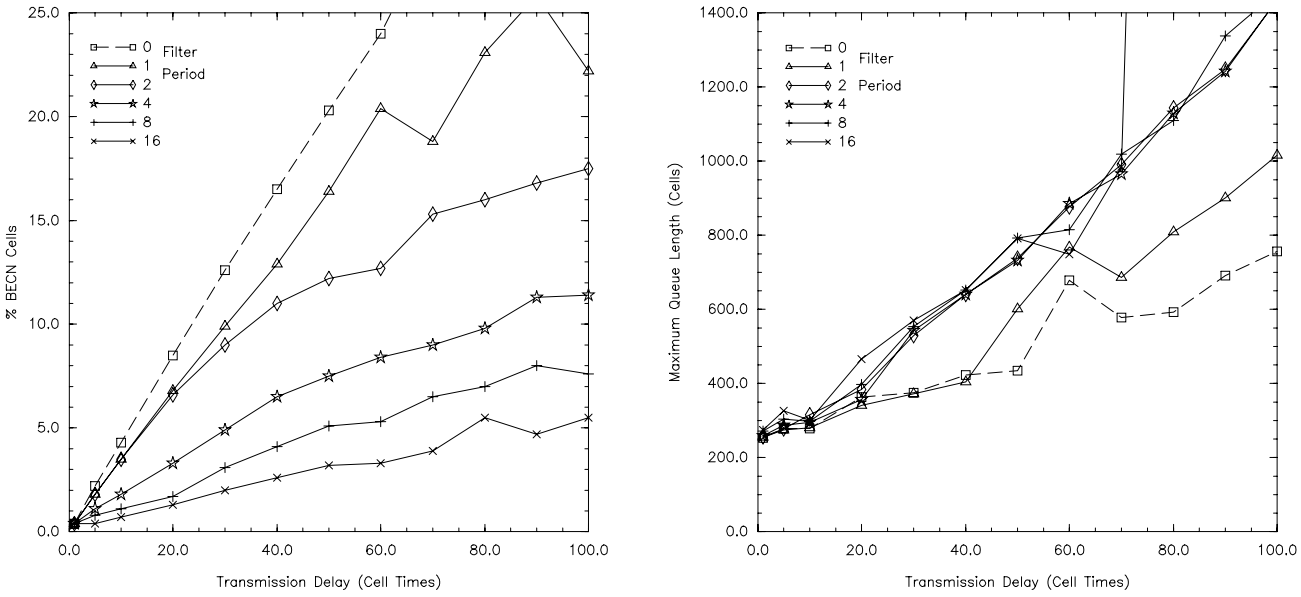

Figure 3: BECN traffic and maximum queue length against transmission delay for different filter periods with a simple BECN cell filter and throttling back to zero.

ery period. A factor of four difference between the highest transmission rate and the lowest was found to be effective, with the shortest source recovery period being approximately equal to the filter period. In the steady-state this causes each source to center at a transmission rate that is approximately its fair share of the available bandwidth. It toggles between this rate, the rate above, and the rate below, during successive filter time periods as the queue length oscillates during a congestion event.

This is a per virtual channel filter since it must keep a record of the sources (i.e. virtual channels) to which it has transmitted a BECN cell during each filter time period. This may be implemented by writing the virtual path and channel identifier into a content addressable memory and clearing the memory at the beginning of each filter period. (The expense of a CAM may be avoided if the VPI/VCI address space is compressed in the input translation and a one bit wide memory is used with a simple state machine to reset each address once during the filter period.)

Curves of BECN traffic and mean and maximum queue length against transmission delay are given in fig. 4 for the per virtual channel filter with up to 100 sources each attempting to transmit at a peak rate of 100%. The filter period is 768 cell times (about 2 ms at 155 Mb/s), the recovery time constant of each source at the lowest transmission rate is also 768 cell times and the queue threshold is 250 cells. In this simulation the number of transmission rates (throttling levels) of each source was increased by three more levels (to a total of nine levels including the 100% and zero levels) such that the minimum non-zero transmission rate was 0.78% (i.e. 2^{-7}). This gives better results for large numbers of sources attempting to transmit at 100% of peak rate. The throughput remains mostly in excess of 80% for transmission delays up to 100 cell times and any number of sources. The amount of BECN traffic is about 0.08% per active source. It remains approximately constant with respect to transmission delay and is approximately proportional to the number of sources. Even for 100 sources transmitting at 100% of peak rate the amount of BECN traffic is less than 9% and for 20 sources it is just over 1%. The mean queue length for 10 sources is also shown in fig. 4. For any number of sources, and transmission delays of up to 100 cell times, the mean queue length is below 200 cells. Also, for up to about 20 sources the maximum queue length remains below 400 cells growing slowly with transmission delay. This is a vast improvement upon the simple filter.

The simulation results are also compared to the results from a simple analytical model of an ideal BECN scheme derived from [3]. In the analytical model the aggregate incoming traffic alternates between 130% and 65% of the available bandwidth and the source recovery timer is tuned such that the minimum queue length is exactly one cell. While the two schemes are far from identical, and considerable simplifying assumptions have been made in the analysis, the measurements of BECN traffic and mean and maximum queue length show remarkable correspondence for up to 20 sources.

5 Conclusions

Backward explicit congestion notification may be employed to support the dynamic sharing of ATM switch port capacity between high-speed bursty sources in the local area. For an ATM LAN the BECN mechanism performs a similar backpressure function to the MAC protocol in a shared medium LAN. If BECN cells are transported at a higher delay priority than data traffic, the maximum delay through an ATM switch is likely to be no more than a few cell times. At 155 Mb/s a propagation delay of one cell time is equivalent to about 0.6 km of fiber. So a single switch ATM LAN is likely to have a transmission delay of around 5 cell times. In this case the simple filter will offer adequate performance. Since sources in the local area prefer to transmit at 100% of peak

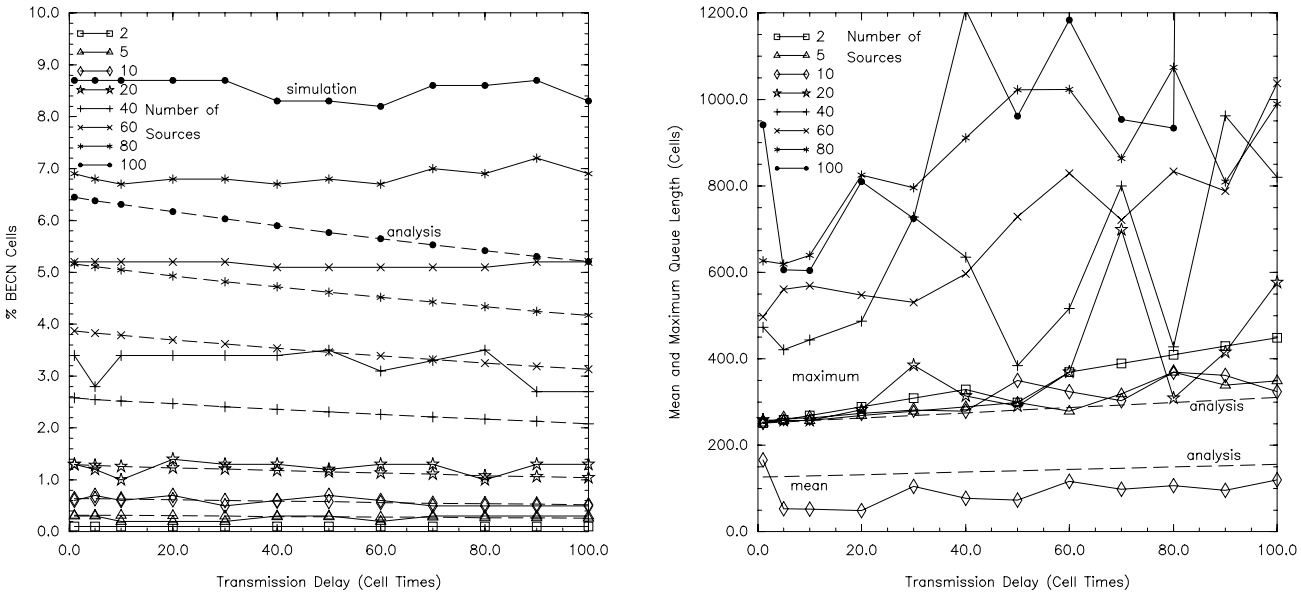

Figure 4: BECN traffic and mean and maximum queue length against transmission delay for number of sources with a per VC BECN cell filter and divide-by-two rate throttling (analysis shown as dashed line).

rate (i.e. at the full line rate) the throttle back to zero algorithm would be the most effective. For a limited number of simultaneously active sources a throughput in excess of 85% with a maximum queue length of less than 500 cells may be attained for transmission delays of up to 30 cell times or so with less than 5% BECN traffic.

For ATM LANs consisting of more than a single switch, distributed across a larger area of up to 50 km or so, or for networks in which a large number of sources are likely to attempt to transmit simultaneously at the full link rate, the per virtual channel filter offers excellent performance. A throughput mostly in excess of 80% may be maintained with only 0.08% of BECN traffic per source, independent of transmission delay, up to about 50 km. For a moderate number of sources (up to about 20) simultaneously attempting to transmit at the full link rate the maximum queue length remains below 400 cells. For any number of sources the mean queue length is less than 200 cells. Work in progress suggests that this performance may be extended to a transmission delay of several hundred kilometers with some slight loss of throughput and increase in the mean and maximum queue lengths. Adjusting the queue threshold (250 cells in the simulations reported) allows throughput to be traded for delay by adjusting the mean queue length and the proportion of time the queue remains empty. The implementation complexity of the BECN scheme with the per VC filter is much less than any of the hop-by-hop per VC credit or feedback schemes, e.g. [6, 7]. It does not require the buffer to be partitioned into individual queues per virtual channel. It does not require a count to be maintained of cells in the buffer on a per virtual channel basis. Also, it is simple to implement in any design of ATM switch: input buffered, output buffered, or anything in between.

This congestion control scheme also provides a simple traffic management mechanism for access to a virtual path into the public network (or a private wide area network). If the BECN mechanism is combined with a virtual path traffic shaper at the access point to the wide area network, we have a mechanism for multiplexing any number of sources of unknown traffic characteristics into a virtual path of specified statistical traffic characteristics. If a traffic burst exceeds the capacity of the virtual path, the sources may be throttled back by the use of BECN cells. This offers much better performance than dropping excess traffic. It also offers much greater statistical gain and is much easier to manage than trying to specify the traffic characteristics of every individual data source in order to calculate the required parameters of the virtual path. Bandwidth across the wide area network may therefore be reserved on a statistical basis, for each virtual path, and shared efficiently between all active users instantaneously requiring access to any virtual path.

References

[1] I Cidon, I Gopal, and R Guerin. Bandwidth management and congestion control in plaNET. *IEEE Commun. Mag.*, pages 54–64, Oct. 1991.

[2] H Esaki. Call admission control methods in ATM networks. In *Proc. IEEE Int. Conf. Commun.*, volume 3, pages 1628–1633, Chicago, Jun. 1992.

[3] A Gersht and K J Lee. A congestion control framework for ATM networks. *IEEE J. Select. Areas in Commun.*, 9(7):1119–1130, Sep. 1991.

[4] R Guerin, H Ahmadi, and M Naghshineh. Equivalent capacity and its application to bandwidth allocation in high-speed networks. *IEEE J. Select. Areas in Commun.*, 9(7):968–981, Sep. 1991.

[5] D Hong, T Suda, and J J Bae. Survey of techniques for prevention and control of congestion in an ATM network. In *Proc. IEEE Int. Conf. Commun.*, volume 1, pages 204–210, Denver, Jun. 1991.

[6] M G H Katevenis. Fast switching and fair control of congested flow in broadband networks. *IEEE J. Select. Areas in Commun.*, 5(8):1315–1326, Oct. 1987.

[7] H T Kung et al. Use of link-by-link flow control in maximizing ATM networks performance: Simulation results. In *Proc. IEEE Hot Interconnects Symp.*, Palo Alto, CA, Aug. 1993.

[8] P Newman. ATM LANs: The customer premises ATM network. In *3rd B-ISDN Technical Workshop*, Nice, France, Apr. 1993. (Submitted to IEEE Commun. Mag.).

[9] P Newman. Simulation results for a backward explicit congestion control scheme. ANSI T1S1.5/93-047, Raleigh NC, Feb. 1993.

[10] FRED Chipset Technical Manual, v2.0. N.E.T. Adaptive, Oct. 1992.