Traffic Management for ATM Local Area Networks*

Peter Newman†

Network Equipment Technologies

June 1994

Two fundamental switching mechanisms are employed in today's networks: *capitalist switching*; and *socialist switching*. According to the capitalist approach, you specify exactly how much bandwidth you require, and if it is available, you receive it. The bandwidth is yours, you can use it or waste it as you will, no-one else has access to it — you don't have to share it with anyone. The socialist view of communication takes a very different course. In a socialist switching mechanism we have a big pool of bandwidth — and we share it! You want some bandwidth, you take it, and when you have finished with it, you return it to the pool.

The asynchronous transfer mode (ATM) is the first switching technology to be capable of supporting both of these fundamental approaches to switching (i.e. circuit switching and packet switching) within a single integrated switching mechanism. Indeed, this was one of the research goals that led to the development of ATM. Considerable progress has already been made in implementing constant bit-rate services, similar to conventional circuit switching, over ATM. Local area networks (LANs) however, take a thoroughly socialist view of communication due to the highly bursty nature of data traffic. Here we briefly consider how to support LAN emulation over an ATM network and then explore how to offer the dynamic bandwidth sharing in the local area, that forms the central thesis of the socialist approach.

1 LAN Emulation

Not so long ago the 10 Mb/s of current LANs seemed an almost infinite amount of bandwidth, even when shared among a large group of workstations. But the power of the desktop workstation has increased rapidly to the point where 10 Mb/s dedicated to each workstation seems only a moderate amount of bandwidth for data applications. High-end workstations, and the support of multimedia applications, are beginning to demand more. A number of technologies are being developed to satisfy this demand, some based upon the traditional shared medium (FDDI, DQDB, Fast Ethernet) and others based upon a switched point-to-point topology (Switched Ethernet, ATM).

A shared medium design restricts the total capacity available to the LAN to the bandwidth of the shared medium. Also, each client must access the LAN at the speed of the shared medium which gets expensive at high speeds. On the other hand, ATM can scale from small multiplexers to very large switches in both aggregate capacity and number of access ports. It can accommodate access ports from low speeds (1.5 Mb/s and 6 Mb/s are under discussion) to very high speeds (2.4 Gb/s). ATM is designed to handle multimedia traffic and will be deployed in the public network as the future Broadband-ISDN. It is therefore of considerable interest to investigate how to support traditional LAN services over an ATM network in the local area.

A LAN offers a connectionless service for the transfer of variable size packets at the medium access control (MAC) sublayer. Corrupted packets are not retransmitted at this layer, but packet loss

---

[*] To appear in the IEEE Communications Magazine, special issue on: "Data Communications in ATM Networks," Aug. 1994.

[†] `<newman@net.com>`

due to corruption within the LAN segment itself is rare. The majority of the installed base of LANs conforms to the IEEE 802 family of protocols, and uses the 48-bit, flat MAC address. In contrast, ATM offers a connection oriented service for the transfer of fixed size cells using a hierarchical ATM address. LAN emulation is the process of supporting an IEEE 802 connectionless packet transfer service at the MAC sublayer on top of ATM.

Network layer protocols can be implemented directly on top of ATM. So why go to the trouble of implementing a MAC sublayer with LAN emulation? Because there is a huge installed base of 'legacy' LANs, bridges, and protocol stacks in end stations, all based on the common standard of the service offered at the top of the IEEE 802 MAC sublayer. Also each network layer protocol must be interfaced to ATM individually. If ATM is to succeed as a LAN technology it must interwork with the existing installed base. It must support the existing higher layer protocol stacks without requiring them to be modified. The solution is to design a protocol layer — an ATM MAC sublayer — above the ATM adaptation layer (AAL) that emulates the service offered by an IEEE 802 LAN segment at the MAC sublayer.

An IEEE 802 LAN segment connects end stations using a physical, broadcast, shared medium. ATM emulates the physical shared medium by establishing an ATM multicast virtual connection between all of the end stations that belong to the ATM LAN segment. This multicast connection is the broadcast channel of the ATM LAN segment. Any station may broadcast to all others on the ATM LAN segment by transmitting on the multicast virtual connection.

For point-to-point communication an address resolution operation is first required to translate the 48-bit MAC address into an ATM address. Address resolution may be implemented with a broadcast mechanism, by a server, or using both. Once the ATM address of the destination has been discovered, a point-to-point ATM virtual connection may be established to the destination using the ATM signaling protocol. The result of the address resolution, and the VCI of the established connection, are cached in a table on the assumption that further communication to that destination is likely. This mechanism operates entirely within the LAN emulation protocol layer and is totally transparent to the LLC and higher layer protocols in the end station. LAN emulation is discussed in much greater detail in [1] and a standard for LAN emulation will be completed by the ATM Forum towards the end of this year.

But this is only half of the story. LANs take a thoroughly socialist view of communication. To support a LAN service on top of ATM the available bandwidth must be dynamically shared between all active users. Considerable effort is currently being expended upon this aspect of traffic management.

2 Traffic Management

There are two fundamental classes of traffic: *capitalist traffic* and *socialist traffic*. Perhaps we are stretching the analogy too far — the two classes of traffic are generally called guaranteed and best-effort. Guaranteed traffic is traffic for which an explicit guarantee of service has been given by the network. The guarantee forms a contract between the traffic source and the network. Prior to connection setup the source must describe its traffic characteristics and request a specific quality of service from the network. The network must police the arriving traffic, to ensure that it conforms to the traffic descriptor. This ensures that an errant traffic source cannot invalidate the guarantee given to other clients by submitting excess traffic into the network.

For a constant bit rate (CBR) ATM connection the traffic will be described in terms of peak cell rate and cell delay variation tolerance (maximum permitted jitter). For a variable bit rate (VBR) ATM connection we also add the sustainable cell rate and the burst tolerance (which defines the maximum burst length that may be transmitted at the peak rate). The quality of service may be described in terms of, for example, the maximum cell loss probability and some function of delay. When a call request for the guaranteed service is received by the network, the call acceptance control determines the resources that the call will require [2, 3, 4]. If the required resources are available the call is accepted, else it is rejected. Examples of traffic that may require a guaranteed service include real-time traffic such as circuit emulation, voice, and video.

The vast majority of existing data networking applications are incapable of predicting their own bandwidth requirements. Indeed, the application is generally interfaced to communications bandwidth via the host's operating system. The application cannot know how many other applications are concurrently demanding communications bandwidth. Nor can the operating system, in general, predict how the user will make use of the application. Therefore it is unrealistic to expect the majority of data applications to predict their bandwidth requirements in advance. Since the traffic characteristics are unknown, an explicit guarantee of service cannot be given. Rather, data applications require a service that dynamically shares the available bandwidth between all active sources. This service is loosely referred to as a best-effort service, although the name 'available bit rate' (ABR) service has recently been adopted as being more descriptive in that the network dynamically shares the bandwidth available for this service between the users.

The switch hardware needs to ensure that at no time will the quality of service of the guaranteed traffic be adversely affected by best-effort traffic. The simplest approach to ensure this is to separate the cell buffering in the ATM switch into at least two traffic classes implemented in separate physical or logical FIFO queues, fig. 1. Guaranteed traffic is placed in one queue and best-effort in the other. The queue service algorithm always serves the guaranteed traffic in preference to the best-effort traffic. More complex queueing structures and service algorithms have been investigated [5, 6] which offer greater efficiency or isolation within different subclasses of traffic.

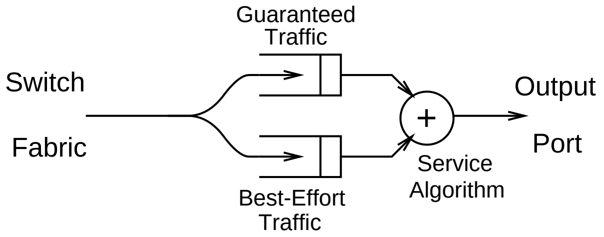

Figure 1: Output buffer with two classes of traffic.

A single FIFO queue may not be sufficient to handle all of the classes of traffic that will require a best-effort service. For example, X-windows traffic and interactive client-server applications are likely to be more delay sensitive than large file transfers. So to prevent large file transfers from disturbing the performance of interactive applications, a small number of separate FIFO queues may be required, one for each of the classes of best-effort traffic. In the limit it is possible for every individual virtual connection to have its own FIFO queue to ensure absolute isolation between different applications [7]. With round-robin scheduling this offers a very close approximation to fair queueing [8] which ensures that all connections get identical service during overload. Selection of the most cost-effective queueing and scheduling scheme is by no means clear, and the choice will significantly influence both the design of the ATM switch and the preferred control mechanism for best-effort traffic.

2.1 Loss Mechanisms

In an IEEE 802 LAN the shared medium provides the bandwidth and the MAC protocol is the arbitration mechanism by which the bandwidth is shared between the contending users. In an ATM switch each output port is a pool of bandwidth that needs to be shared dynamically between all connections currently sending best-effort traffic across it. There are three proposed solutions: use large buffers and let the higher layer protocols deal with congestion; use a loss mechanism; use a delay mechanism — generally a feedback control scheme.

Some transport layer protocols, notably TCP and DECnet, employ an adaptive window flow control scheme to handle congestion in the network. If the best-effort buffer in the ATM switch is large enough to hold a reasonable number of packets one might expect these protocols to achieve adequate performance. Simulation studies suggest that this very much depends upon one's definition of adequate performance [9, 10]. Depending upon the buffer size, TCP packet size, number of connections sharing the buffer, etc., TCP will achieve a useful throughput of 60%–90%. For ten saturated TCP connections on 155 Mb/s links sharing a 1000 cell buffer with 9180 byte packets and a 64 kbyte window the useful throughput is slightly less than 60% [9]. The remaining 40% of the bandwidth available to the best-effort service is occupied by cells belonging to packets already corrupted by cell loss. In a packet network this bandwidth is not wasted because the entire packet is discarded when a buffer becomes congested. In ATM, if the loss mechanism simply discards arriving cells when the buffer is full, each discarded cell is likely to belong to a different packet.

An improvement is to implement a loss mechanism in the best-effort buffer of the ATM switch that will discard the remainder of a packet should it become necessary to discard one cell from the packet. The studies suggest that this enhancement offers only marginal performance improvement. Early packet discard, a scheme that can completely discard all new arriving packets if the buffer exceeds a threshold, has shown much better performance, close to that of pure packet switching for TCP, in a simple topology [9]. However, simulation studies of more complex topologies have shown some unfairness in the distribution of bandwidth by TCP with the early packet discard scheme.

The Fast Reservation Protocol [11] is a burst loss scheme designed to operate regardless of the upper layer protocol. The source sends a request into the network, along the virtual connection, asking permission to transmit a burst at a specified peak bit rate. All switches along the path examine the request and if the requested bandwidth is available they reserve the bandwidth and forward the request. If all switches along the path accept the request an acknowledgement is returned to the source from the far end. The scheme has the disadvantage that bandwidth is wasted, once reserved, until the request completes the round trip.

An alternative scheme treats the burst itself as a request [12]. If the bandwidth is available the burst is forwarded else it is completely discarded. In these schemes the source knows what it wants — the requested peak bit rate, and requests it of the network. In a LAN the source typically does not know what it wants — it wants as much as is currently available. Also, if no-one else is using the LAN, a source would like access to the entire bandwidth. Analysis has shown that the burst reservation schemes suffer high levels of burst blocking (loss) when the requested peak bit rate is high compared to the link rate [13]. It would seem that a more natural scheme for LAN applications is one in which the network informs the sources of the amount of bandwidth currently available.

2.2 Feedback Mechanisms

Feedback schemes use a closed-loop feedback control mechanism that allows the network to control the cell emission process at each source. Each virtual connection must have an independent control loop since each connection may follow a different path through the network. Two classes of feedback scheme have been proposed: credit-based; and rate-based.

The credit approach is a link-by-link window flow control scheme [14]. Each link in the network independently runs the flow control mechanism, fig. 2. A certain number of cell buffers are reserved for each virtual connection at the receiving end of each link. One round trip's worth of cell buffers must be reserved for each connection, so the amount of buffering required per connection depends upon the propagation delay of the link and the maximum required transmission rate of the virtual connection. To permit a connection to attain a maximum bit rate of 155 Mb/s, four cell buffers per km per connection are required, due to the propagation delay, plus further buffering to account for any switching delay. A credit balance is maintained at the sending end of the link where each unit of credit rep-

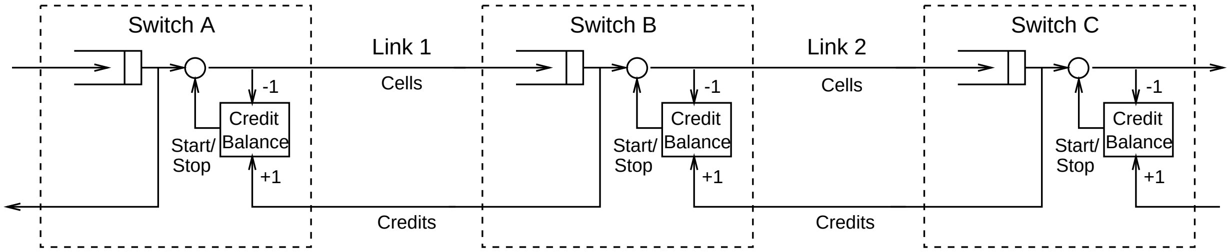

Figure 2: A link-by-link credit based flow control scheme.

resents one empty cell buffer at the receiving end. As each cell is transmitted by the sending end the credit balance for that connection is decremented. When a connection runs out of credit on a particular link it must stop transmitting cells. As cells are removed from the buffer at the receiving end of the link, credit is returned upstream on each connection. Current proposals suggest that a credit update is returned upstream for every ten user cells transmitted in the forward direction. Credit information for up to six connections may be contained in a single credit update cell. A background audit process is also required to protect against loss of credit, and therefore loss of throughput, caused by cell loss.

Rate-based schemes use feedback information from the network to control the rate at which each source emits cells into the network on every virtual connection. Three types of rate-based scheme have been proposed: explicit rate control, forward explicit congestion notification (FECN) [15, 16, 17]; and backward (BECN) [18, 19]. With explicit rate control the network periodically determines at what rate each source should be transmitting and sends a message to each source informing them of the new rate. Few details are yet available on this type of rate-based scheme.

FECN is an end-to-end scheme in which most of the control complexity resides in the end systems. When a path through a switch becomes congested the switch marks a bit in the header of all cells on that path in the forward direction to indicate congestion. (This function is already required by the existing ATM standards.) The destination end system monitors the congestion status of each active virtual connection and sends congestion notification cells in the reverse direction on each active virtual connection to inform the source of the congestion status. The source uses this feedback to increase or decrease the cell transmission rate on each virtual connection.

In BECN congestion information is returned directly from the point of congestion back to the source for each virtual channel. The source adjusts its cell transmission rate on each virtual connection in a similar manner to FECN. BECN requires more hardware in the switch to detect and filter the congestion state, and to insert cells indicating congestion into the return path, but it is capable of reacting to congestion faster than FECN. Also, since the network itself generates the congestion feedback information, it is more robust against end systems that do not comply with the requirements of the scheme. The policing mechanism at the entrance to the network can itself perform the rate adjustment in response to the congestion feedback so that traffic from non-compliant sources can be discarded. In the following section we present a simple BECN mechanism for an ATM LAN with a brief summary of some simulation results from a feasibility study. Details of the simulation model and further results are available in [19].

3 A BECN Scheme

The simulation model of the BECN scheme is illustrated in fig. 3. The source (S) generates packets that are queued for transmission by the transmitter (T). The transmitter represents the segmenta-

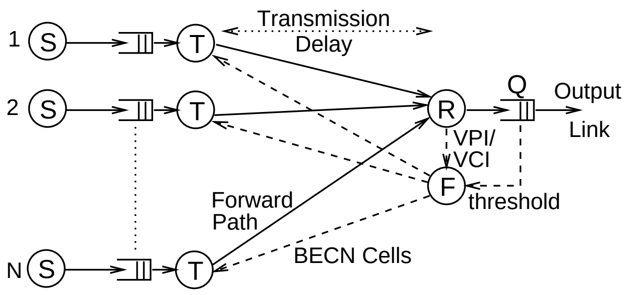

Figure 3: The BECN scheme.

tion process for a particular virtual connection in the host's ATM network interface, which segments packets into cells for transmission over the ATM network. (R) is a point of congestion somewhere in the network, for example, an output buffer on an ATM switch. There is a transmission delay between each transmitter transmitting a cell and the cell being received at (R). The transmission delay represents the combination of the propagation delay and the switching delay from the source to the point of congestion.

If the length of the congested queue (Q) exceeds a threshold, the filter (F) will send congestion notification (BECN) cells back on the virtual channels currently sending traffic through the congested queue. BECN cells are subject to the same transmission delay in the return direction as cells in the forward direction. When a transmitter receives a BECN cell it reduces its cell transmission rate by half for the indicated virtual connection. If no BECN cells are received within a recovery time period, the current transmission rate for that virtual connection is doubled, once each recovery time period, until it reaches the peak rate. To bias the scheme toward fairness between all sources, the source recovery period was made proportional to the current transmission rate so that the lower the transmission rate the shorter the source recovery period. This algorithm was selected to simplify the implementation of the ATM adapter silicon [20]. (Fairness is usually achieved in a binary feedback rate-based scheme by using multiplicative decrease and additive increase [21].)

The filter is required to prevent excess BECN cells being generated. There is no point in sending another BECN cell back to each source until the previous feedback has had time to take effect. So, when the queue is congested, the filter should transmit no more than a single BECN cell to each active source during each filter time period. To allow the previous BECN cell to take effect, the filter time period should be of the same order of magnitude as the maximum propagation delay for which the system is designed.

3.1 Simulation Results

For the following steady-state simulation results, the congestion control algorithm prevented any loss of traffic from the destination queue. The amount of BECN traffic is expressed as a percentage of the output link rate. The transmission delay is expressed in cell times normalized to the output link rate (at 155 Mb/s one cell time is about 2.7 μs). Curves of BECN traffic and mean and maximum queue length against transmission delay are given in fig. 4 with up to 20 greedy sources each attempting to transmit at 100% of the output line rate. 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. The throughput remains mostly in excess of 80% for transmission delays up to 100 cell times. (A propagation delay of 100 cell times represents about 50 km at 155 Mb/s.) 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 greedy sources each attempting to transmit at 100% of the line rate the amount of BECN traffic is less than 9% and for 20 sources it is just over 1%. For 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. The results of a simple analytical model of an ideal BECN scheme derived from [22] are shown as dashed lines in fig. 4. Agree-

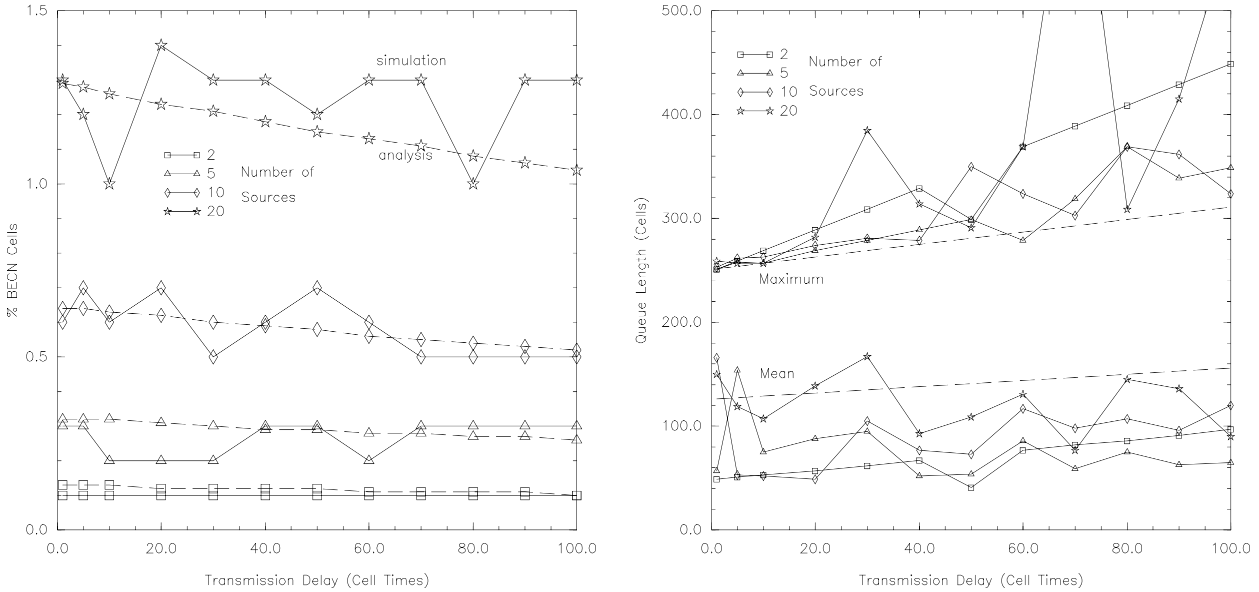

Figure 4: BECN traffic and queue length against transmission delay for number of sources.

ment is surprisingly close up to about 20 sources considering the simplicity of the analytical model.

3.2 Implementation

The implementation of the BECN scheme is illustrated in fig. 5 for an output buffered switch. (Implementation for other switch architectures is similar.) The VPI/VCI of every cell arriving at the output buffer is latched. If the queue is above threshold the VPI/VCI is submitted to the filter initiating a BECN cell generation request. If the filter passes the request, the VPI/VCI is inserted into the header of a pre-defined BECN cell pattern and the resulting BECN cell injected into the cell stream in the reverse direction before the header translation hardware. Because VPI/VCI values are bi-directional in ATM, the BECN cell will be returned to the source of the virtual connection.

The filter enforces an upper limit on the amount of BECN traffic that can be generated. It requires one bit of state per virtual connection to keep a record of the connections on which it has transmitted a BECN cell during each filter period. It may be implemented by writing the VPI/VCI into a content addressable memory and clearing the memory at the beginning of each filter period. Alternatively, 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.

BECN cells may be transmitted at a higher priority than best-effort traffic to reduce the transmission delay. BECN cells themselves must not be permitted to generate further BECN cells should the reverse path also become congested. BECN cell generation is a property of the connection and may be disabled in any node on a per connection basis at connection set-up.

BECN cells may be coded to use a resource management (RM) cell format (payload type 6). (This is one of the non-user cell formats reserved for control functions in an ATM network.) The simulation study assumes that each BECN cell carries congestion information regarding a single virtual connection back to its source. We can reduce the amount of BECN traffic on network trunks by a factor of up to 15 by packing congestion information regarding up to 15 virtual connections into a single RM cell. The RM cell would contain the VPI/VCI values of up to 15 congested virtual connections. This technique requires more hardware because these packed RM cells require unpacking at the far end of the link, however the extra hardware

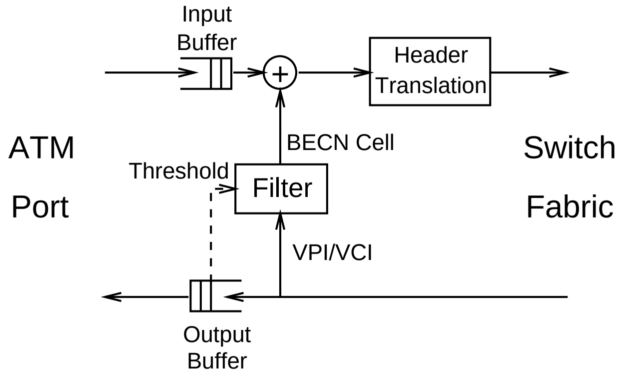

Figure 5: Implementation of BECN on an ATM switch port.

may well be worthwhile in order to reduce control traffic on network trunks. Also, BECN cells passing through multiple nodes may be routed through the filter at every node they pass through as there is no need to duplicate a BECN cell if an earlier node on the connection has just issued a BECN cell. This will prevent unnecessary generation of BECN traffic in a network with multiple simultaneous congestion events.

4 Discussion

The simulation results for the BECN scheme suggest that for the local area a simple rate-based feedback scheme is worthy of further investigation. At least in the steady-state a large number of greedy sources, each capable of filling the entire link bandwidth, may be controlled with a small amount of control traffic and moderate size buffers, to achieve an acceptable utilization. Consideration must be given to the method by which a source initiates a new transmission. One possibility is to initiate transmission at 100% of the link rate, but to drop to a much lower rate if a BECN cell is received within a time period of a few round trip delays. Another possibility is to define an initial rate at which to begin new transmissions. Further work needs to focus on transient behavior, fairness of resource sharing, and more complex topologies.

The behavior of a rate controlled source is similar for both BECN and FECN so the two rate-based schemes may be combined. FECN requires very little additional hardware in the switch which makes it attractive to the public carriers, who envisage deploying very large switches. Also FECN naturally collects and filters congestion information along the entire forward data path while BECN requires explicit filtering within each switch. BECN offers a faster response and allows connections in the local area faster access to the available bandwidth which is important for LANs. BECN is also easier to police across a network interface as the network itself generates the control messages. This is an important consideration for a wide area network, offering service to a private LAN. One possible approach to ensure tight policing is to divide the network into multiple segments. Each segment can then implement its own internal feedback control system which can be either rate or credit based. A BECN rate control scheme is used across the interface between segments. This approach allows a network segment in the middle to apply strict policing on arriving traffic but requires per VC queueing at the interface. It results in a network-by-network (edge-to-edge) control scheme rather than an end-to-end scheme.

An alternative implementation for a combined FECN/BECN mechanism is to use positive control messages. In this case a control message is sent periodically, on each connection, from the destination to the source when the forward path is not congested. This scheme is more robust against loss of control cells, link failure, and mis-behaving users, because the transmission rate decreases on the loss of control information. Also BECN may be implemented simply by deleting control cells. A problem with periodic transmission of control information occurs when a large number of connections each wish to transmit at a rate approaching that of the frequency of the control messages. In this situation the amount of control traffic can approach the amount of forward data traffic. To avoid this problem a rate-based scheme may be designed where the control traffic on each connection is proportional to the forward data traffic. Such a scheme is under investigation by the ATM Forum.

Simulation studies of the link-by-link, per virtual connection, credit scheme show very high link utilization with zero cell loss [10]. A single active connection can acquire near instantaneous access to all of the available bandwidth yet when multiple connections are active the bandwidth is shared fairly. In the local area about 10–20 cell buffers will need to be reserved for each virtual connection to maintain a maximum rate of 155 Mb/s with about 4% of the bandwidth being consumed by credit cells. With fixed buffer reservation even a modest number of virtual channels will result in large buffer requirements. In the local area 1000 connections will require a 10k–20k cell buffer per port. In the wide area, buffer requirements become much larger due to the increased propagation delay. Adaptive credit allocation has been proposed to solve this problem whereby buffers are shared and connections only receive credit as they become active [23].

Probably the most significant difference between the credit-based and rate-based control mechanisms is one of complexity. On each switch port the credit scheme requires individual FIFO queues for each virtual connection and a scheduler to select cells for transmission from connections that have both traffic and credit. It also requires a background audit process to detect and restore credit loss. The rate schemes are less deterministic and slower to adjust the bandwidth on each connection. Under extreme variations of load it is possible for cell loss to occur or under utilization of the available bandwidth. Rate schemes have a number of parameters in the source that must be set by the network for each connection whereas a credit-based scheme can be auto-configuring. However, the rate-based approach places very modest requirements on the switch hardware and requires a similar complexity to the credit scheme in the ATM adapter at the end station.

5 Conclusion

At present, only the very high end workstations and ATM adapter cards can sustain transmission in excess of 100 Mb/s. But, given the pace of development in ATM LANs, this will not remain so for very long. Transport protocols with an adaptive window congestion control mechanism will operate successfully over ATM LANs without congestion control at the ATM layer (assuming reasonably large switch buffers). But as soon as end stations achieve transmission rates around 100 Mb/s, up to 40% of the bandwidth will be wasted due to cell loss and packet retransmission. Also, the parameters of the protocol may need to be tuned in each end station to achieve this performance. Switches that implement packet discard rather than cell discard under congestion conditions will exhibit much better performance for protocols with an adaptive window mechanism [9]. However, protocols that do not employ an adaptive window, for example IPX or NFS, will exhibit very poor performance over an ATM LAN if there is no traffic management scheme at the ATM level to share the available bandwidth between contending sources [24].

Several traffic management schemes are under consideration by the ATM Forum, including rate-based schemes, and a credit-based scheme. Both approaches seem capable of supporting low loss and high utilization within the local area. The credit scheme is most attractive to those who believe that per virtual connection buffering is required to ensure isolation between the different applications sharing the best-effort service. Those that see no need to implement per virtual connection buffering are more likely to prefer the rate-based approach.

The rate-based approach seems more natural for public wide area networks. The broadband-ISDN must offer a universal information transport service supporting many different types of traffic for many years to come. It seems unreasonable to require a significant amount of specialized hardware down at the ATM layer in each switch to support a single class of traffic. Also public networks deal in rate — it is more difficult to offer a best-effort service with a minimum guaranteed rate using a credit-based scheme. The public carriers are likely to support a rate-based approach because of complexity issues in large switches although it is unclear how many carriers plan on providing such services in the near future.

While it would be possible for both schemes to work together (rate in one network interfaced to credit in another) the desire for multivendor interoperability mandates the selection of a single scheme. From the investigations conducted so far it would seem that both schemes can be engineered to offer acceptable performance in the local area. Considering the rate at which the industry is moving, the ATM Forum must soon select a single feedback control scheme for best-effort (ABR) traffic. Once over that hurdle we can get on with the engineering of making it work.

References

[1] P Newman. ATM Local Area Networks. *IEEE Commun. Mag.*, pages 86–98, Mar. 1994.

[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] 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.

[4] Y Sato and K-I Sato. Virtual path and link capacity design for ATM networks. *IEEE J. Select. Areas in Commun.*, 9(1):104–111, Jan. 1991.

[5] J M Hyman, A A Lazar, and G Pacifici. Real-time scheduling with quality of service constraints. *IEEE J. Select. Areas in Commun.*, 9(7):1052–1063, Sep. 1991.

[6] D D Clark, S Shenker, and L Zhang. Supporting real-time applications in an integrated services packet network. In *Proc. ACM SIGCOMM*, pages 14–26, Baltimore, MD, Sep. 1992.

[7] S Shenker. A theoretical analysis of feedback flow control. In *Proc. ACM SIGCOMM*, pages 156–165, Philadelphia, Sep. 1990.

[8] A Demers, S Keshav, and S Shenker. Analysis and simulation of a fair queueing algorithm. In *Proc. ACM SIGCOMM*, pages 1–12, Austin, Tx, Sep. 1989.

[9] A Romanow and S Floyd. Dynamics of TCP traffic over ATM networks. Submitted to ACM SIGCOMM, London, Sep. 1994.

[10] C Fang, H Chen, and J Hutchins. Simulation analysis of TCP performance in congested ATM LAN using DEC's flow control scheme and two selective cell-drop schemes. ATM Forum/94-0119, Tahoe, Jan. 1994.

[11] P E Boyer and D P Tranchier. A reservation principle with applications to the ATM traffic control. *Computer Networks and ISDN Systems*, 24:321–334, 1992.

[12] J S Turner. Managing bandwidth in ATM networks with bursty traffic. *IEEE Networks Mag.*, pages 50–58, Sep. 1992.

[13] Y Z Cho and A Leon-Garcia. Performance of burst-level bandwidth reservation in ATM LANs. In *Proc. IEEE Infocom*, Toronto, Jun. 1994.

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

[15] O Aboul-Magd and H Gilbert. Incorporating congestion feedback in B-ISDN traffic management strategy. In *Proc. Int. Switching Symp.*, volume 2, pages 12–16, Yokohama, Oct. 1992.

[16] B A Makrucki. On the performance of submitting excess traffic to ATM networks. In *Proc. IEEE Globecom*, Dec. 1991.

[17] N Yin and M G Hluchyj. On closed-loop rate control for ATM cell relay networks. In *Proc. IEEE Infocom*, Toronto, Jun. 1994.

[18] A W Berger, F Bonomi, and K W Fendick. Proposal for backward congestion feedback at the ATM UNI. ATM Forum/93-839R1, Bedford, MA, Sep. 1993.

[19] P Newman. Backward explicit congestion notification for ATM local area networks. In *Proc. IEEE Globecom*, volume 2, pages 719–723, Houston, TX, Dec. 1993.

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

[21] D M Chiu and R Jain. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. *Computer Networks and ISDN Systems*, 17(1):1–14, Jun. 1989.

[22] 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.

[23] H T Kung, T Blackwell, and A Chapman. Adaptive credit allocation for flow-controlled VCs. ATM Forum/94-0282, Durham, NC, Mar. 1994.

[24] S Keung et al. IPX performance with flow control in congested ATM LANs. ATM Forum/94-0272, Durham, NC, Mar. 1994.